Artificial Vision

Course Summary - Master's Degree

Carlos Alberto Botina Carpio
Universidad Internacional de la Rioja
carlos.botina621@comunidadunir.net

November 25, 2025

Abstract

This document contains a summary of the Artificial Vision course syllabus for the Master's Degree. It includes a summary of the main topics covered during the sessions, as well as additional explanations and extensions of the concepts and techniques referenced in class. The purpose of this document is to serve as study material and reference for the course contents.

Contents

Lecture 003

1 Introduction

This document presents a summary of the **Artificial Vision** course syllabus for the Artificial Intelligence Master's Degree. The content includes a structured summary of the main topics covered during the course sessions, as well as additional explanations and extensions of the concepts, algorithms, and techniques referenced in class.

The main objective is to provide a comprehensive reference that complements the in-person sessions, facilitating the study and understanding of the fundamentals and applications of artificial vision. It includes detailed explanations of the most relevant topics, practical examples, and bibliographic references that allow for deeper exploration of the aspects covered during the course.

2 Sampling

Sampling is the process of converting a continuous-time signal into a discrete-time signal by measuring the signal's value at specific, uniformly spaced time instants. In the context of digital signal processing and computer vision, sampling is fundamental because real-world signals (such as images, sounds, or sensor measurements) are continuous in nature, but computers can only process discrete, finite sets of values.

The sampling process involves taking "snapshots" of a continuous signal at regular intervals, creating a sequence of discrete values that represent the original signal at those specific moments in time. The rate at which these samples are taken is called the **sampling frequency** or **sampling rate**, typically denoted as f_s and measured in samples per second (Hz). The time interval between consecutive samples is called the **sampling period** $T_s = 1/f_s$.

A critical question in sampling theory is: *How fast must we sample a signal to ensure that we can perfectly reconstruct the original continuous signal from its discrete samples?* This question is answered by the Nyquist-Shannon sampling theorem, which establishes the minimum sampling rate required for perfect reconstruction.

2.1 The Nyquist-Shannon Sampling Theorem

The Nyquist-Shannon sampling theorem, also known as the sampling theorem, is a fundamental principle in signal processing and digital image processing. It establishes the conditions under which a continuous signal can be perfectly reconstructed from its discrete samples.

2.1.1 Statement of the Theorem

Theorem 2.1 (Nyquist-Shannon Sampling Theorem)**.** If a function $x(t)$ contains no frequencies higher than B hertz, it is completely determined by giving its ordinates at a series of points spaced $\frac{1}{2B}$ seconds apart. In other words, a band-limited signal can be perfectly reconstructed from its samples if the sampling frequency f_s satisfies:

$$f_s \geq 2f_{\max} \tag{1}$$

where $f_{\max}$ is the highest frequency component in the signal. The frequency $f_N = \frac{f_s}{2}$ is called the **Nyquist frequency**, and $2f_{\max}$ is called the **Nyquist rate**.

Curious Fact: Frequency and Period Relationship

The relationship between frequency f and period T is fundamental in signal processing:

$$f = \frac{1}{T} \tag{2}$$

$$T = \frac{1}{f} \tag{3}$$

where:

- f is the frequency (measured in hertz, Hz, or cycles per second)
- T is the period (measured in seconds, s, or time per cycle)

This means that frequency and period are inversely related: higher frequency corresponds to shorter period, and vice versa. For example, if a signal has a frequency of $f = 10$ Hz, its period is $T = \frac{1}{10} = 0.1$ seconds. In the context of sampling, the sampling period T_s and sampling frequency f_s are related by $T_s = \frac{1}{f_s}$.

2.2 Understanding Sampling in the Time Domain

To understand the theorem, consider a continuous signal $x(t)$ that we wish to sample at regular intervals. The sampling process converts the continuous-time signal into a discrete-time signal by taking samples at uniformly spaced time instants. This relationship is mathematically expressed as:

$$x[n] = x(t_n), \quad t_n = nT_s, \quad n \in \mathbb{Z}, \quad T_s \in \mathbb{R} \tag{4}$$

where:

- $x[n]$ is the discrete-time signal (sequence of samples)
- $x(t_n)$ is the value of the continuous signal at time instant t_n
- n is an integer index representing the sample number
- T_s is the sampling period (time interval between consecutive samples)
- $f_s = \frac{1}{T_s}$ is the sampling frequency

Figure 1 illustrates this sampling process, showing how a continuous signal is converted into a discrete sequence of samples.

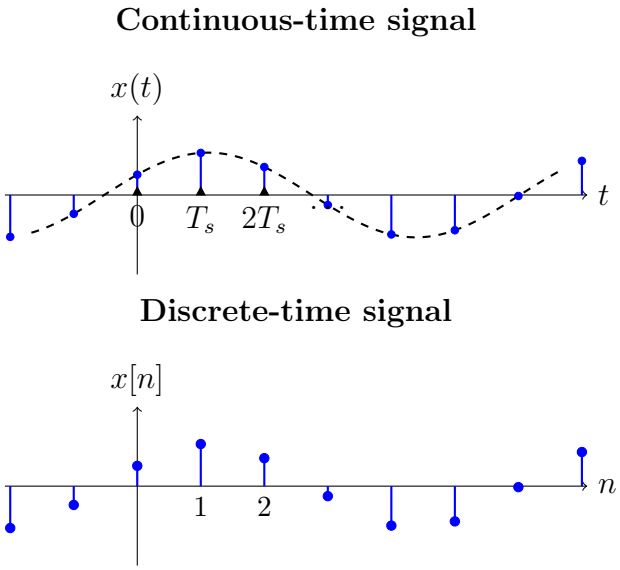

Figure 1: Sampling process: conversion from continuous-time signal $x(t)$ to discrete-time signal $x[n]$. The top plot shows the continuous signal with sampling instants marked, and the bottom plot shows the resulting discrete sequence.

2.3 Signal Reconstruction with Sinc Interpolation

Once a signal has been sampled according to the Nyquist-Shannon theorem, the original continuous signal can be perfectly reconstructed from its discrete samples. This reconstruction is achieved through **sinc interpolation**, which uses the sinc function to interpolate between sample points.

The sinc function is defined as:

$$\text{sinc}(t) = \frac{\sin(\pi t)}{\pi t} \tag{5}$$

with the special case $\text{sinc}(0) = 1$ (by L'Hôpital's rule).

The reconstruction formula, also known as the **Whittaker-Shannon interpolation formula**, expresses the continuous signal $x(t)$ as a weighted sum of sinc functions centered at each sample point:

$$x(t) = \sum_{n=-\infty}^{\infty} x[n] \cdot \text{sinc}\left(\frac{t - nT_s}{T_s}\right) \tag{6}$$

where:

- $x[n]$ are the discrete samples
- T_s is the sampling period
- Each sinc function is centered at a sampling instant nT_s
- The sinc function has zeros at all other sampling instants, ensuring that $x(t)$ equals $x[n]$ at $t = nT_s$

This reconstruction works because:

1. At each sampling instant $t = nT_s$, only the sinc function centered at that point contributes (all others are zero), so $x(nT_s) = x[n]$.

2. Between sampling points, the sinc functions smoothly interpolate the signal values.

3. In the frequency domain, the sinc function acts as an ideal low-pass filter, removing all frequency components above the Nyquist frequency while preserving those below it.

Curious Fact: Band-Limited Signals and Perfect Reconstruction

Since sinc interpolation acts as a low-pass filter (removing all frequencies above the Nyquist frequency $f_N = f_s/2$), if we have a band-limited signal with maximum frequency $f_{\max}$ lower than the Nyquist frequency, then no frequencies are going to be removed and therefore, the result is a theoretically perfect reconstruction.

2.4 Aliasing

Aliasing is a distortion phenomenon that occurs when a signal is sampled at a rate that is too low (below the Nyquist rate). When aliasing occurs, high-frequency components of the signal are "folded back" or "aliased" into lower frequencies, making them indistinguishable from actual low-frequency components in the sampled signal.

2.4.1 Why Aliasing Occurs

In the frequency domain, sampling creates periodic replicas of the signal's spectrum at integer multiples of the sampling frequency. When the sampling frequency f_s is less than $2f_{\max}$, these replicas overlap. The overlapping high-frequency components appear as lower frequencies in the sampled signal, causing aliasing.

For example, consider a signal with frequency $f = 8$ Hz sampled at $f_s = 10$ Hz:

- The Nyquist frequency is $f_N = f_s/2 = 5$ Hz
- The signal frequency (8 Hz) is above the Nyquist frequency
- The aliased frequency is $f_{\text{alias}} = f_s - f = 10 - 8 = 2$ Hz
- The sampled signal incorrectly appears to have a 2 Hz component instead of the original 8 Hz

2.4.2 Consequences of Aliasing

Once aliasing occurs, the original signal cannot be perfectly reconstructed because the high-frequency information has been irretrievably mixed with lower frequencies. This is why the Nyquist-Shannon theorem requires sampling at or above the Nyquist rate to ensure perfect reconstruction.

Curious Fact: The Wagon Wheel Effect

A classic example of aliasing in everyday life is the **wagon wheel effect** (also known as the stroboscopic effect) seen in videos. When a wheel with spokes rotates at a certain speed and is filmed at a fixed frame rate, the wheel can appear to rotate backward, slowly, or even stand still. This occurs because the wheel's rotation frequency is being undersampled by the camera's frame rate. The high-frequency rotation is aliased into a lower apparent frequency, creating the illusion of reverse or slow motion. This is a temporal aliasing effect, where time (rather than space) is being sampled.

These are the three different sampling scenarios:

- **Adequate sampling** ($f_s > 2f_{\max}$): The signal can be perfectly reconstructed.
- **Nyquist rate sampling** ($f_s = 2f_{\max}$): The minimum sampling rate that theoretically allows perfect reconstruction.
- **Insufficient sampling** ($f_s < 2f_{\max}$): Aliasing occurs, and the original signal cannot be recovered.

2.5 Exercise: Determining Minimum Sampling Rate

Consider the following signal:

$$x(t) = \cos(100\pi t) + \sin(200\pi t) + \cos(500\pi t + \pi/4) + 7 \tag{7}$$

Problem: Determine the minimum sampling rate required to perfectly reconstruct this signal.

Solution:

To find the minimum sampling rate, we need to identify the maximum frequency component in the signal. Let's analyze each term:

- $\cos(100\pi t)$:

$$\begin{aligned}
\text{Angular frequency: } \omega_1 &= 100\pi \text{ rad/s} \\
\text{To convert to frequency: } f_1 &= \omega_1 \times \frac{1 \text{ cycle}}{2\pi \text{ rad}} \\
&= 100\pi \text{ rad/s} \times \frac{1 \text{ cycle}}{2\pi \text{ rad}} \\
&= \frac{100\cancel{\pi \text{ rad}}/\text{s} \times 1 \text{ cycle}}{2\cancel{\pi \text{ rad}}} \\
&= \frac{100}{2}\frac{\text{cycle}}{\text{s}} = 50 \text{ cycles/s} = 50 \text{ Hz}
\end{aligned}$$

- $\sin(200\pi t)$: $f_2 = \frac{200\pi}{2\pi} = 100$ Hz
- $\cos(500\pi t + \pi/4)$: $f_3 = \frac{500\pi}{2\pi} = 250$ Hz
- 7: This is a constant (DC component) with frequency $f_0 = 0$ Hz

The maximum frequency in the signal is $f_{\max} = 250$ Hz (from the $\cos(500\pi t + \pi/4)$ term).

According to the Nyquist-Shannon theorem, the minimum sampling rate (Nyquist rate) is:

$$f_s \geq 2f_{\max} = 2 \times 250 = 500 \text{ Hz} \tag{8}$$

Therefore, the minimum sampling rate required is **500 Hz**.

Figure 2 demonstrates the signal reconstruction process using **sinc interpolation** (Whittaker-Shannon interpolation formula) for three different sampling scenarios. The reconstruction is performed using the formula:

$$x(t) = \sum_{n=-\infty}^{\infty} x[n] \cdot \text{sinc}\left(\frac{t - nT_s}{T_s}\right) \tag{9}$$

where $\text{sinc}(t) = \frac{\sin(\pi t)}{\pi t}$ and $T_s = 1/f_s$ is the sampling period.

Each row of the figure shows three plots: (1) the original continuous signal, (2) the sampled signal with sample points marked, and (3) the reconstructed signal using sinc interpolation overlaid with the original for comparison. The three rows correspond to:

- **Adequate sampling** ($f_s = 600$ Hz $> 2f_{\max}$): The reconstructed signal perfectly matches the original, demonstrating perfect reconstruction when sampling above the Nyquist rate.
- **Nyquist rate sampling** ($f_s = 500$ Hz $= 2f_{\max}$): The reconstructed signal matches the original, showing that the Nyquist rate is the theoretical minimum for perfect reconstruction.
- **Insufficient sampling** ($f_s = 300$ Hz $< 2f_{\max}$): The reconstructed signal does not match the original due to aliasing, demonstrating that perfect reconstruction is impossible when sampling below the Nyquist rate.

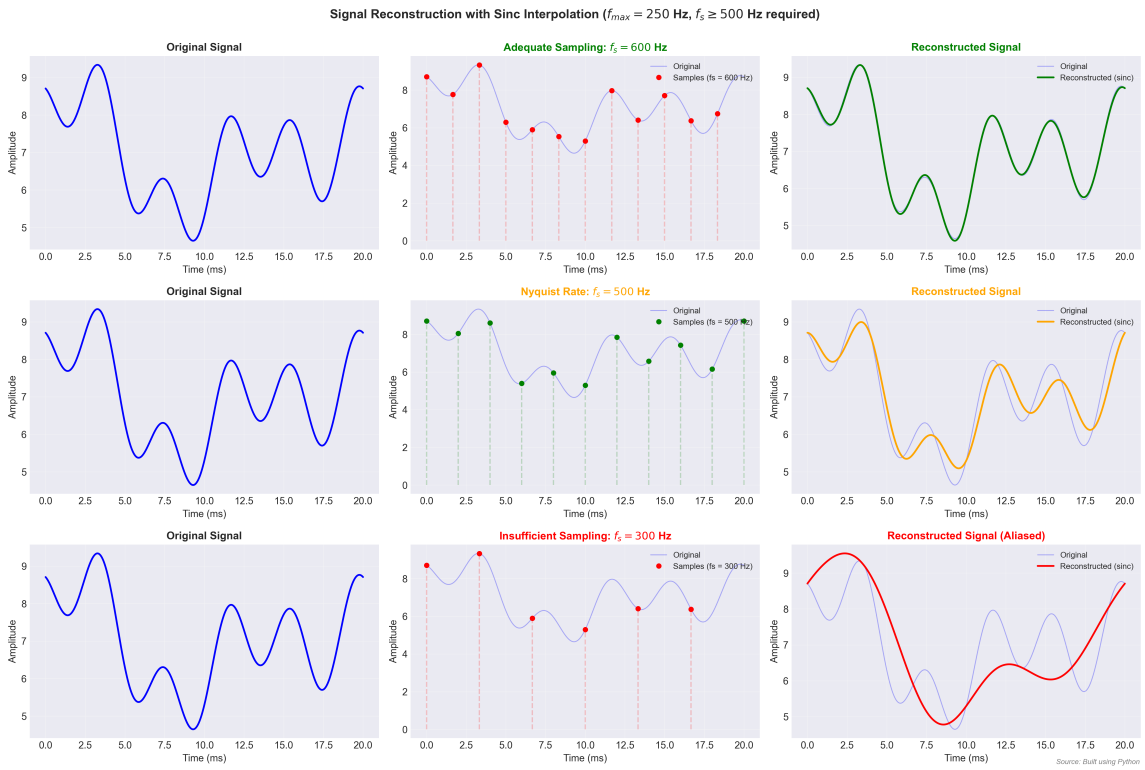

Figure 2: Signal reconstruction using sinc interpolation for $x(t) = \cos(100\pi t) + \sin(200\pi t) + \cos(500\pi t + \pi/4) + 7$ with $f_{\max} = 250$ Hz. Each row shows: original signal (left), sampled signal (center), and reconstructed signal using sinc interpolation (right). The three rows demonstrate adequate sampling (600 Hz), Nyquist rate (500 Hz), and insufficient sampling (300 Hz) where aliasing prevents perfect reconstruction.

Lecture 004

3 Entropy: Concept and Estimation

3.1 Noise

Noise is any unwanted signal of random nature that modifies the intensity of the original signal to be perceived.

In the real world, signals are affected by uncontrollable elements that generate noise. This noise is typically superimposed as **additive noise**:

$$S(t) = f(t) + r(t) \tag{10}$$

where $S(t)$ is the received signal, $f(t)$ is the original signal, and $r(t)$ is the noise component.

The first stage in signal processing focuses on identifying and eliminating noisy artifacts, though complete elimination is usually not feasible. The random nature of noise means that signals with noise are not deterministic but rather **stochastic processes**, where repeated measurements of the same signal produce different results.

3.1.1 Types of Noise

Atmospheric Noise Atmospheric noise comes from electrical signals derived from natural discharges that occur under the ionosphere. Storms or electrical charges in clouds are sources of this type of noise, which generally affects communication systems using the radio spectrum more significantly. Approximately, the power of atmospheric noise is inversely proportional to frequency. Thus, atmospheric noise has greater impact on low and medium frequency bands, while lower power noise affects VHF and UHF bands. As a result, atmospheric noise affects AM communication bands and decreases significantly at TV and FM frequencies. Beyond 30 MHz, atmospheric noise has less negative impact than the receiver's own noise.

Man-Made Noise This refers to electrical artifacts generated by sources such as automobiles, electric motors, switches, high-voltage lines, etc. It is also known as **industrial noise**. The intensity of these noisy signals is greater in large urban centers and industrial areas. In these areas, noise of this nature prevails over other noise sources in the frequency range between 1 MHz and 600 MHz.

Impulsive or Shot Noise This type of noise causes the appearance of anomalous values (outliers) in the signal. It is characterized by a sudden increase in intensity during a short period of time. Generally, its origin is an external agent to the information system: a lightning strike or interference from a motor spark. However, it should not be confused with atmospheric or man-made noise, as the duration of these is more prolonged in time.

Galactic Noise It originates from disturbances produced beyond the Earth's atmosphere. The main sources of galactic noise are the sun and other stars.

- **Solar**: The sun is a major source of energy emission in the form of electromagnetic radiation. These signals affect telecommunications systems. The frequency range of these emissions is very wide, including bands commonly used for radio communication systems. The intensity of the emission produced by the sun varies cyclically, with a period of approximately eleven years. At the highest levels, this radiation can make some frequency bands unusable.
- **Cosmic**: Like the sun, other stars near our planet emit energy in the form of electromagnetic radiation that can affect our signals and communication systems.

Thermal Noise This noise source is due to the random agitation of electrons in the elements of an electronic circuit. This movement could only be canceled under absolute zero temperature conditions. Therefore, it is an unavoidable noise source that will always be present in a signal acquisition and processing system. The movement of electrons increases as the temperature of the conductor increases, giving rise to small electrical currents. This noisy signal is distributed over a wide range of frequencies, so it will always affect the system to some degree, despite carrying out different filtering stages.

Flicker Noise or 1/f Noise It is called 1/f because its power decays below 1 kHz when frequency increases. Therefore, it has greater impact on low frequencies. The physical causes of this type of noise are not entirely clear. It originates in elements such as

transistors or resistors, and it is hypothesized that it is due to intermodulation processes in these elements.

3.1.2 Signal-to-Noise Ratio (SNR)

When an information source is affected by noisy artifacts, the **Signal-to-Noise Ratio (SNR)** quantitatively indicates the quality of the signal of interest. This ratio is defined as the quotient between the power of the received signal and the estimated noise power. A value greater than unity (1) indicates a greater presence of the signal compared to the noise. The relationship between these power terms is generally expressed in decibels (dB).

$$\text{SNR} = 10 \log_{10} \left(\frac{P_S}{P_N} \right) \tag{11}$$

where:

- P_S corresponds to the signal power.
- P_N corresponds to the noise power.

Example: Consider a communication system where the signal power is $P_S = 100$ and the noise power is $P_N = 10$. The SNR is calculated as:

$$\begin{aligned} \text{SNR} &= 10 \log_{10} \left(\frac{P_S}{P_N} \right) \\ &= 10 \log_{10} \left(\frac{100}{10} \right) \\ &= 10 \log_{10}(10) \\ &= 10 \times 1 = 10 \text{ dB} \end{aligned}$$

This means the signal power is 10 times greater than the noise power (a ratio of 10:1), resulting in an SNR of 10 dB.

Interpretation of SNR values:

- **SNR greater than 0 dB**: Signal power exceeds noise power (good quality)
- **SNR = 0 dB**: Signal and noise powers are equal
- **SNR lower than 0 dB**: Noise power exceeds signal power (poor quality)
- **SNR = 20 dB**: Signal is 100 times stronger than noise (excellent quality)
- **SNR = 3 dB**: Signal is approximately 2 times stronger than noise (minimum acceptable for many applications)

3.2 Entropy

Signals contain information and are affected by various noise sources. In this context, the concept of **entropy** arises. Similar to physics, the term refers to the complexity of the signal. The addition of noise increases the degree of complexity of a signal, resulting in higher entropy.

In information theory, **entropy** is defined as the amount of information from a random source (on average). Therefore, entropy serves to **characterize a random variable**. Signals can be modeled as a sequence of realizations of a random variable over time (stochastic process), so we will see how to extend the definition of entropy to random elements of this nature.

3.2.1 Shannon's Entropy Definition

Given a discrete random variable X that takes values from the set $\{x_1, x_2, \ldots, x_M\}$ with probability distribution $P(X = x_i) = p_i$, Shannon defined entropy as:

$$H(X) = E\{-\log_2[P(X)]\} = \sum_{i=1}^{M} -\log_2[P(x_i)] \cdot P(x_i) = \sum_{i=1}^{M} -p_i \log_2(p_i) \tag{12}$$

where $-\log_2[P(x_i)]$ is interpreted as the **quantity of information** (or **self-information**) associated with outcome x_i.

Curious Fact: What Does E Mean?

The capital E denotes the **expected value** (also called expectation or mean). For a discrete random variable, the expected value of a function $g(X)$ is calculated as:

$$E[g(X)] = \sum_{i=1}^{M} g(x_i) \cdot P(x_i) \tag{13}$$

In the entropy formula, $E\{-\log_2[P(X)]\}$ means we take the expected value of the information content $-\log_2[P(X)]$, which gives us the average information across all possible outcomes.

3.2.2 Understanding the Formula

The key insight is that **less probable values carry more information** (surprise effect) compared to more probable values. For example:

- If an event is very likely ($p_i \approx 1$), then $-\log_2(p_i) \approx 0$: we learn little new information.
- If an event is very unlikely ($p_i \approx 0$), then $-\log_2(p_i)$ is large: we learn a lot of new information.

Entropy $H(X)$ is the **expected value** (average) of this information content across all possible outcomes.

3.2.3 Example: Bernoulli Distribution

Consider a random variable X with only two possible outcomes, $\{x_1, x_2\}$ (a **Bernoulli distribution**). Let $P(X = x_1) = p$ and $P(X = x_2) = 1 - p$. The entropy is:

$$H(X) = -p \log_2(p) - (1 - p) \log_2(1 - p) \tag{14}$$

The entropy reaches its **maximum value of 1** when $p = 0.5$. In this case, both events have equal probability, and on average we obtain the same amount of information from X. When p approaches 0 or 1, the entropy approaches 0, meaning we can almost predict the outcome with certainty, so we learn little new information.

Concrete Example: Consider a fair coin flip where $p = 0.5$:

$$\begin{aligned} H(X) &= -0.5 \log_2(0.5) - 0.5 \log_2(0.5) \\ &= -0.5 \cdot (-1) - 0.5 \cdot (-1) \\ &= 0.5 + 0.5 = 1 \end{aligned}$$

This means each coin flip provides an entropy of 1 on average. If the coin is biased (e.g., $p = 0.9$ for heads), then:

$$\begin{aligned} H(X) &= -0.9 \log_2(0.9) - 0.1 \log_2(0.1) \\ &\approx 0.469 \end{aligned}$$

The entropy is lower because we can predict the outcome more easily (heads is very likely), so we learn less information.

Figure 3 shows the variation of entropy $H(X)$ as a function of the probability $P(X = x_1) = p$ for a Bernoulli distribution. The curve is symmetric and reaches its maximum value of 1 when $p = 0.5$, demonstrating that uncertainty (entropy) is highest when both outcomes are equally probable.

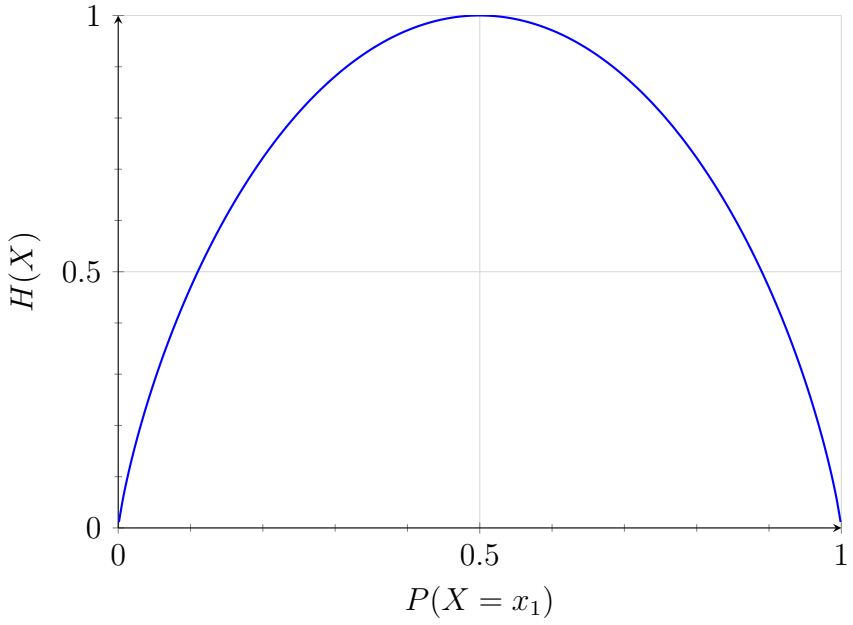

Figure 3: Entropy $H(X)$ as a function of probability $P(X = x_1) = p$ for a Bernoulli distribution. The entropy reaches its maximum value of 1 when $p = 0.5$ (equal probability for both outcomes) and approaches 0 when p approaches 0 or 1 (certain outcomes).

3.3 Signals as Stochastic Processes

Signals can be modeled mathematically as a set of random variables (a **stochastic process**). For example, a voice signal of a certain duration can be viewed as a finite time series, where each sample represents a realization of a random variable.

Including new samples in the series increases the information content, showing that process entropy depends on its length. Therefore, it makes sense to measure the variation of signal entropy due to the inclusion of a new sample. This is called the **entropy rate** or **differential entropy**.

3.3.1 Entropy of a Stochastic Process

Consider a signal of length N as a sequence of N random variable realizations: $\mathbf{x} = x_1, x_2, \ldots, x_N$. Figure 4 illustrates this concept, showing a signal where each sample x_i represents a realization of a random variable at time index i.

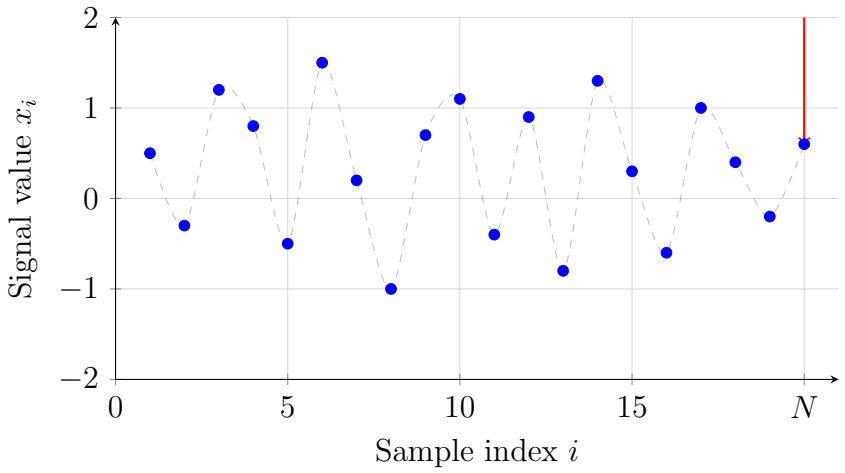

Figure 4: A signal of length $N = 20$ as a sequence of random variable realizations. Each point (i, x_i) represents a sample where i is the time index and x_i is the realization of the random variable at that time.

The entropy H_N of this stochastic process is:

$$\begin{aligned} H_N &= E\{-\log_2[p(x_1, x_2, \ldots, x_N)]\} \\ &= -\int_{-\infty}^{\infty} \log_2[p(x_1, x_2, \ldots, x_N)] \cdot p(x_1, x_2, \ldots, x_N)\, dx_1 \ldots dx_N \end{aligned} \tag{15}$$

where $p(x_1, x_2, \ldots, x_N)$ is the joint probability density function (PDF) of the variables composing the stochastic process.

Note: Computational Complexity

Computing entropy using the stochastic process formula is **much more computationally expensive** than ApEn:

- **Stochastic process entropy**: Requires estimating an N-dimensional joint PDF and computing an N-dimensional integral. The complexity grows exponentially with N (curse of dimensionality), making it impractical for long signals.
- **ApEn**: Works with fixed-length subseries (m is typically 2-3), requiring only pattern matching. Complexity is approximately $O(N^2)$, which is manageable even for long signals.

This computational advantage is one of the main reasons why ApEn is widely used in practice.

3.3.2 Entropy Rate

The **entropy rate** E_N of the signal is defined as:

$$E_N = \lim_{N\to\infty} (H_{N+1} - H_N) \tag{16}$$

This represents the change in entropy when a new sample is added to an infinitely long sequence.

Note

The entropy rate measures how much entropy increases (on average) when you add one more sample to a long signal.

3.3.3 Approximate Entropy (ApEn)

There are various methods for estimating signal entropy. **Approximate Entropy (ApEn)** is one such estimation procedure. The algorithm involves estimating the entropy of subseries of length m and $m+1$. The final entropy value is obtained by taking the difference between these two estimations.

Methodology: Consider an original time series $\mathbf{x} = [x_1, x_2, \ldots, x_N]$.

Step 1: Extract subseries. Extract all subseries of length m, denoted as $x_i^{(m)} = [x_i, x_{i+1}, \ldots, x_{i+m-1}]$ for $i = 1, 2, \ldots, N-m+1$.

Step 2: Find similar subseries. Given a tolerance r, count the number of subseries $N^{(m)}(i)$ that are similar to $x_i^{(m)}$, where similarity is determined by a distance metric $d[x_i^{(m)}, x_j^{(m)}] \leq r$.

Step 3: Calculate probability. The probability of finding a subseries similar to $x_i^{(m)}$ in the original series is:

$$C^{(m)}(i) = \frac{N^{(m)}(i)}{N-m+1} \tag{17}$$

where $N - m + 1$ is the total number of subseries of length m that can be extracted from the original series.

Step 4: Estimate entropy. The term $C^{(m)}(i)$ provides a discrete estimation of the probability density function. Using Shannon's entropy definition, the entropy of the process represented by $x^{(m)}$ is:

$$H_N^{(m)} = -\frac{1}{N-m+1} \sum_{i=1}^{N-m+1} \log_2[C^{(m)}(i)] \tag{18}$$

The Approximate Entropy is then calculated as:

$$\text{ApEn}(m, r, N) = H_N^{(m)} - H_N^{(m+1)} \tag{19}$$

Note: ApEn vs. Entropy Rate

Approximate Entropy (ApEn) is **not exactly the same** as the entropy rate, but they are related concepts:

- **Entropy rate** $E_N = \lim_{N\to\infty}(H_{N+1} - H_N)$ measures the change in entropy when adding one more **sample** to the sequence.
- **ApEn** $= H_N^{(m)} - H_N^{(m+1)}$ measures the change in entropy when increasing the **subseries length** from m to $m + 1$.

ApEn is an **approximation** of the entropy rate. Instead of computing the theoretical entropy rate (which requires the full joint PDF), ApEn estimates it by analyzing patterns in subseries of different lengths. Both measure how entropy changes with sequence length, but ApEn uses a practical, pattern-based approach rather than the theoretical limit.

Example: Consider a time series $\mathbf{x} = [1.0, 1.2, 0.9, 1.1, 1.3, 0.8]$ with $N = 6$. Let $m = 2$ and $r = 0.2$.

Step 1: Extract subseries of length $m = 2$:

- $x_1^{(2)} = [1.0, 1.2]$
- $x_2^{(2)} = [1.2, 0.9]$
- $x_3^{(2)} = [0.9, 1.1]$
- $x_4^{(2)} = [1.1, 1.3]$
- $x_5^{(2)} = [1.3, 0.8]$

Step 2: Find similar subseries. Using the Chebyshev distance (maximum absolute difference between corresponding elements), we count how many subseries are within tolerance $r = 0.2$ of each $x_i^{(2)}$. Two subseries are similar if $d[x_i^{(m)}, x_j^{(m)}] = \max_k |x_{i+k} - x_{j+k}| \leq r$.

For example, comparing $x_1^{(2)} = [1.0, 1.2]$ with $x_4^{(2)} = [1.1, 1.3]$:

$$\begin{aligned} d[x_1^{(2)}, x_4^{(2)}] &= \max(|1.0 - 1.1|, |1.2 - 1.3|) \\ &= \max(0.1, 0.1) = 0.1 \leq 0.2 \end{aligned}$$

Since $0.1 \leq 0.2$, these subseries are similar.

Results for all subseries:

- For $x_1^{(2)} = [1.0, 1.2]$: $N^{(2)}(1) = 2$ (matches itself and $x_4^{(2)} = [1.1, 1.3]$)
- For $x_2^{(2)} = [1.2, 0.9]$: $N^{(2)}(2) = 1$ (only itself)
- For $x_3^{(2)} = [0.9, 1.1]$: $N^{(2)}(3) = 1$ (only itself)
- For $x_4^{(2)} = [1.1, 1.3]$: $N^{(2)}(4) = 2$ (matches itself and $x_1^{(2)}$)
- For $x_5^{(2)} = [1.3, 0.8]$: $N^{(2)}(5) = 1$ (only itself)

Step 3: Calculate probabilities. With $N - m + 1 = 6 - 2 + 1 = 5$ total subseries:

$$C^{(2)}(1) = \frac{2}{5} = 0.4$$
$$C^{(2)}(2) = \frac{1}{5} = 0.2$$
$$C^{(2)}(3) = \frac{1}{5} = 0.2$$
$$C^{(2)}(4) = \frac{2}{5} = 0.4$$
$$C^{(2)}(5) = \frac{1}{5} = 0.2$$

Step 4: Estimate entropy.

$$\begin{aligned} H_6^{(2)} &= -\frac{1}{5}\sum_{i=1}^{5} \log_2[C^{(2)}(i)] \\ &= -\frac{1}{5}[\log_2(0.4) + \log_2(0.2) + \log_2(0.2) + \log_2(0.4) + \log_2(0.2)] \\ &\approx -\frac{1}{5}[-1.32 - 2.32 - 2.32 - 1.32 - 2.32] \\ &\approx 1.92 \end{aligned}$$

Similarly, we would compute $H_6^{(3)}$ for subseries of length $m + 1 = 3$, and then:

$$\text{ApEn}(2, 0.2, 6) = H_6^{(2)} - H_6^{(3)}$$

Takeaway: Why is ApEn Useful?

Approximate Entropy is a practical and powerful tool for signal analysis because:

- **Practical estimation**: It provides a computationally feasible way to estimate entropy without requiring knowledge of the full probability distribution, making it applicable to real-world signals.
- **Pattern detection**: By analyzing subseries patterns, ApEn can detect regularity and predictability in signals, which is useful for characterizing signal complexity.
- **Noise robustness**: The tolerance parameter r allows ApEn to be robust to noise, focusing on overall patterns rather than exact matches.
- **Wide applications**: ApEn is widely used in biomedical signal processing (EEG, ECG), time series analysis, and any domain where quantifying signal complexity or regularity is important.
- **Comparative analysis**: It enables comparison of entropy between different signals or different segments of the same signal, helping identify changes in signal characteristics.

3.4 Entropy in Images

The entropy value of a signal can be interpreted as its degree of uncertainty. Equivalently, it reflects the capacity to predict a future state or value from the knowledge or observation of previous signal values. A higher entropy value reflects greater complexity and chaos in the signal under study.

As a result, entropy gives us an idea of the level of noise impact on a signal. If we take a sample of the same signal under the same conditions but at different time instants, the signal with higher entropy will be the one with a higher noise level.

3.4.1 Images vs. One-Dimensional Signals

The nature and mathematical modeling of images are different from one-dimensional time-dependent signals. An image does not have an implicit time variable, as occurs in a voice signal or an electrocardiogram, but rather represents light captured at each spatial position. Additionally, image information is represented in two dimensions.

Therefore, in images, just as in time series we characterized the rate of entropy increase with respect to new samples, we could think of an entropy rate with respect to the unit area represented. For entropy estimation in an image, the histogram of intensity levels is used. The final estimation is obtained as the entropy of the random variable characterized by this histogram.

As with one-dimensional signals, entropy will tend to increase, or at least remain the same, if the image area considered for estimation is enlarged. Thus, lower entropy values will be associated with repetitive patterns in the image that lead to a histogram with marked peaks (texture). In contrast, entropy increases if there is greater variability in the intensity values observed in the image, with no marked patterns producing a flatter

histogram. In this sense, it follows that noise contributes to increasing the entropy of the image, as it causes the variability of the observed intensity levels to increase.

3.5 Mathematical Characterization of Noise: Stochastic Processes

The term **stochastic process** has been previously used in this topic to refer to a random signal. In our case, any signal will be the result of the combination of the signal of interest and an unwanted signal, of random and chaotic nature, which contributes to increasing entropy. This unwanted signal is noise.

Therefore, the resulting signal is, in itself, a random signal. Just as happens with a random variable, from which we take a sample and obtain values according to a probability density function, the signals we handle are realizations of a stochastic process. Each time we extract a sample from the information source, we obtain a different signal.

In this section, a formal definition of stochastic process is provided that allows understanding the modeling and characterization of noise in signal processing.

3.5.1 Random Variables

A random variable is characterized by the following three elements:

- **Sample space**: The set of all possible outcomes that can be observed in the realization of an experiment.
- **Set of events**: Subset of the sample space.
- **Probability law**: Assignment of probability to each of the observable events.

Example: Rolling a Fair Die Consider the experiment of rolling a fair six-sided die:

- **Sample space**: $\Omega = \{1, 2, 3, 4, 5, 6\}$ (all possible outcomes)
- **Set of events**: Examples include:
  - Event A: "Rolling an even number" $= \{2, 4, 6\}$
  - Event B: "Rolling a number greater than 4" $= \{5, 6\}$
  - Event C: "Rolling a 3" $= \{3\}$
- **Probability law**: For a fair die, each outcome has equal probability:
  - $P(1) = P(2) = P(3) = P(4) = P(5) = P(6) = \frac{1}{6}$
  - $P(A) = P(\{2, 4, 6\}) = \frac{3}{6} = \frac{1}{2}$
  - $P(B) = P(\{5, 6\}) = \frac{2}{6} = \frac{1}{3}$

Example: Signal Intensity Measurement In signal processing, consider measuring the intensity of a signal at a specific time:

- **Sample space**: $\Omega = [0, 255]$ (all possible intensity values, e.g., for an 8-bit image)
- **Set of events**: Examples include:
  - Event D: "Intensity between 100 and 150" $= [100, 150]$
  - Event E: "Intensity greater than 200" $= (200, 255]$
- **Probability law**: Defined by a probability density function (PDF) $f(x)$ that assigns probabilities to intervals, such as:
  - $P(D) = \int_{100}^{150} f(x)\, dx$
  - $P(E) = \int_{200}^{255} f(x)\, dx$

3.5.2 Stochastic Processes

A stochastic process can be viewed as a random variable for which the result of an experiment is given in the form of a signal. In the same way as a random variable, it is characterized by the three elements mentioned: sample space, set of events, and probability assignment law.

In practice, as previously mentioned in this topic, we will have noisy signals that, from a mathematical point of view, will be modeled as a stochastic process. The noise component will be assumed to be additive, so the captured signal will have the following form:

$$y(t) = x(t) + \varepsilon(t) \tag{20}$$

where:

- $x(t)$ reflects the signal of interest.
- $\varepsilon(t)$ corresponds to the noise.

Example: Sinusoidal Signal with Gaussian Noise Consider, for example, that the signal of interest corresponds to a tone of frequency f and that the noise component follows a Gaussian distribution with zero mean and variance σ^2.

In Figure 5, we can see this target signal (top) and a realization of the stochastic process that corresponds to the observed signal (bottom).

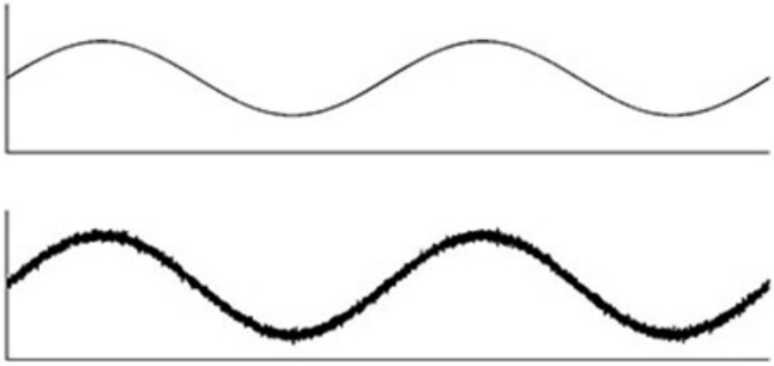

Figure 5: Comparison of a clean sinusoidal signal (top) and a noisy realization of the stochastic process (bottom). The noise component gives the signal a random nature that prevents us from knowing its exact value at any instant t.

As can be seen, the noise component gives the signal a random nature that prevents us from knowing its exact value at any instant t. In order to characterize the stochastic process, the objective will be to know its statistical properties. Probability distribution and density functions allow us to model the process statistically.

These functions are given as follows:

- **Distribution function**: $F_X(x,t) = P(X(t) \leq x)$
- **Probability density function**: $f_X(x,t) = \frac{dF_X(x,t)}{dx}$

From these functions, the stationarity of the process can be defined:

- A process is stationary in **strict sense** if the probability density function that characterizes the process does not vary with time. That is, for a constant c such that $c > 0$, the following will hold: $f_X(x,t) = f_X(x,t+c)$
- A process is stationary in **wide sense** if the statistical moments that characterize it (mean, variance, etc.) do not vary with respect to time.

Example: Strict-Sense Stationarity Consider a process $X(t) = A\sin(2\pi ft + \phi)$, where A and ϕ are random variables, and f is a constant frequency. If A and ϕ are independent, with A having a fixed distribution and ϕ uniformly distributed on $[0, 2\pi]$, then the PDF of $X(t)$ at any time t is the same (it depends only on the distribution of A and ϕ, not on t). This process is stationary in strict sense because $f_X(x,t) = f_X(x,t+c)$ for any c.

Example: Wide-Sense Stationarity Consider white noise $\varepsilon(t)$ with zero mean and constant variance σ^2:

- Mean: $E[\varepsilon(t)] = 0$ (constant, independent of t)
- Variance: $\text{Var}[\varepsilon(t)] = \sigma^2$ (constant, independent of t)
- Autocorrelation: $E[\varepsilon(t)\varepsilon(t+\tau)] = \sigma^2\delta(\tau)$ (depends only on τ, not on t)

This process is stationary in wide sense because its statistical moments (mean and variance) do not vary with time, even though we may not know the full PDF.

Example: Non-Stationary Process Consider a process $Y(t) = t + \varepsilon(t)$, where $\varepsilon(t)$ is white noise. The mean of this process is $E[Y(t)] = t$, which clearly varies with time. Therefore, this process is **not stationary** (neither in strict sense nor in wide sense) because its statistical properties change over time.

Example: Non-Stationary Signal with Trend Let us return to the previous example. In this case, the captured signal shows, in addition to Gaussian noise, another component that causes a clear trend over time. Figure 6 shows this new example.

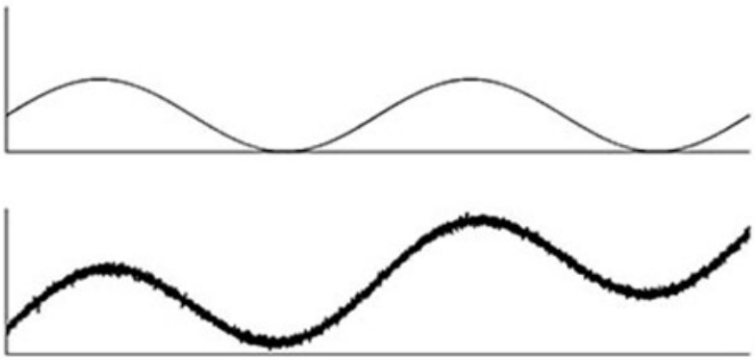

Figure 6: Non-stationary signal with a trend component. The signal exhibits both Gaussian noise and a clear trend over time, making its statistical properties vary along the temporal axis.

As a result of this trend, the statistical properties of the signal do not remain constant along the temporal axis, so it cannot be considered a stationary signal. It will be necessary to eliminate the noise component that causes this trend to remove the non-stationarity present in our information.

Lecture 005

4 Anomaly Detection and Cancellation

4.1 Definition of Anomaly

Anomaly detection aims to identify atypical values in the information source, commonly known as **outliers**. These are defined as unusual patterns that do not conform to expected behavior. The appearance of outliers in a signal or image reflects the existence of noise, typically impulsive noise caused by, for example: a peak value in a nearby electric field, or instabilities in the capture procedure, such as sudden camera movement.

Anomaly detection has direct applications in various practical scenarios:

- **Intrusion detection in networks**. Identification of atypical patterns in network traffic that may indicate an attack.
- **Medical diagnosis**. Recognition of lesions with low incidence in the population that may indicate the existence of some pathology.

- **Fraudulent transaction detection**. The vast majority of transactions are legitimate, and only a small proportion correspond to fraudulent activities.
- **Customer churn prediction in large companies**. In banking, insurance, and telecommunications sectors, a small portion of customers abandon the company, so the identification of these behaviors can be performed using anomaly detection techniques.

4.2 Types of Anomalies

4.2.1 Point Anomalies

When an individual sample can be considered notably different from the rest of the data, it can be taken as an outlier. This type of anomaly is the simplest and the focus of most research work on this topic.

A clear example corresponding to a real scenario would be credit card fraud. If we look at a variable such as the transaction amount, those transactions for which the amount is very high compared to the average of previous transactions are susceptible to being point anomalies and, therefore, suspicious of fraud. Thus, a point anomaly is expressed by the appearance of peak values that deviate excessively from the set of values we observe.

Figure 7 shows a signal in which one of the samples takes a value that is not observed in any other sample. This is clearly a candidate sample for being an anomaly.

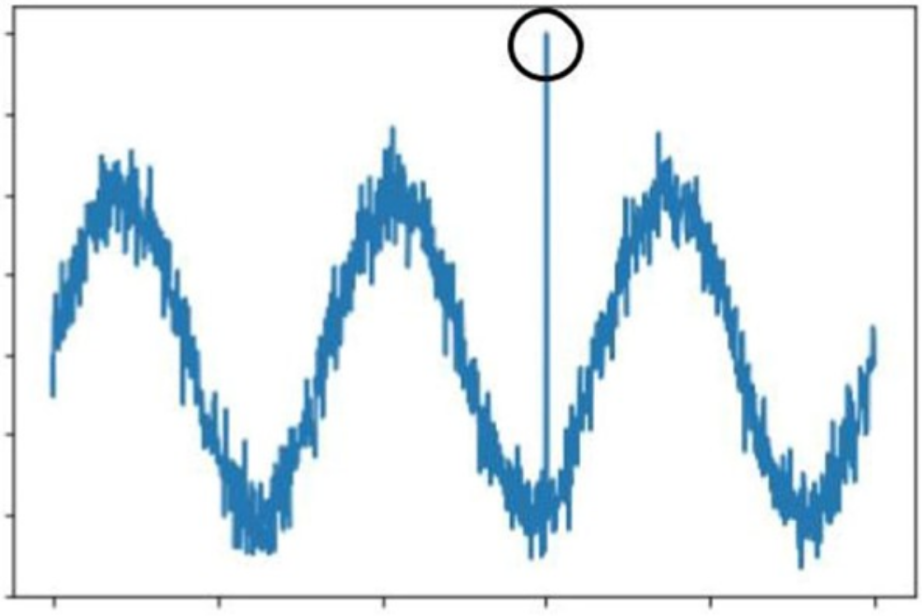

Figure 7: Example of a point anomaly in a signal: a sharp spike (highlighted in black) that deviates significantly from the normal signal pattern.

4.2.2 Contextual Anomalies

If a data sample is anomalous in a specific context (but not otherwise), it is called a **contextual anomaly**. The notion of context is given by the nature of the data. Each data sample is defined considering the following attributes:

- **Contextual attributes**: These are used to determine the context (or neighborhood) for that sample. They are given by the nature of the data source. For example, in temperature monitoring systems, the time of day or season are contextual attributes. In network traffic analysis, the timestamp or day of the week serve as contextual attributes. In image processing, the spatial coordinates (x, y) of a pixel are contextual attributes.

- **Behavioral attributes**: These define the non-contextual character of an instance. That is, they represent the value of the sample. In the temperature monitoring example, the actual temperature reading is a behavioral attribute. In network traffic analysis, the number of packets or bytes transmitted is a behavioral attribute. In image processing, the pixel intensity or color values are behavioral attributes.

Anomalous behavior is determined using the values of behavioral attributes within a specific context. A data instance could be a contextual anomaly in a given context, but an identical data instance (in terms of behavioral attributes, i.e., its value) could be considered normal in a different context. This property is key to identifying contextual and behavioral attributes for a contextual anomaly detection technique.

Unlike point anomalies, where only a comparison of available data samples is performed to identify an atypical value, in temporal signals (time series) and images, context is taken into account to define an abnormal value. For example, in an image, it is possible to identify an anomalous pixel if its intensity is very different from that of neighboring pixels. Similarly, in a time series, the neighborhood of a point provides the contextual information necessary to identify an anomalous value, as exemplified in Figure 8. In this figure, we can see that the series takes similar values to the anomaly at some point, but the context indicates that in this case it is an atypical sample.

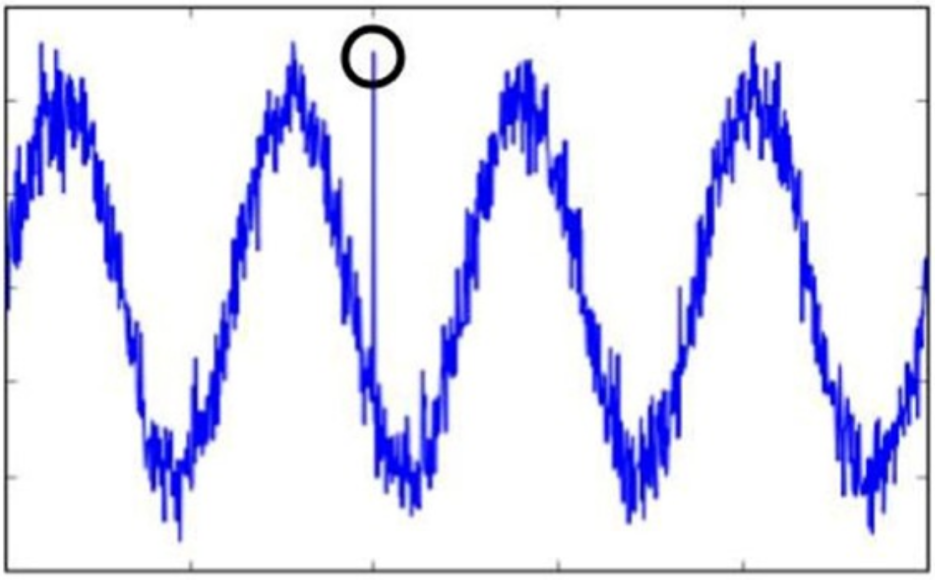

Figure 8: Example of a contextual anomaly in a time series: a spike (highlighted in black) that appears anomalous given its context, even though similar values occur elsewhere in the signal.

4.2.3 Collective Anomalies

If a collection of related data instances is anomalous with respect to the entire dataset, it is called a **collective anomaly**. The individual data instances in a collective anomaly may not be anomalies by themselves, but their joint occurrence as a collection is anomalous.

Figure 9 illustrates an example of a collective anomaly in an electrocardiographic signal. The highlighted region denotes an anomaly because the signal takes approximately the same value for an unusually long time. However, that value itself is not an anomaly.

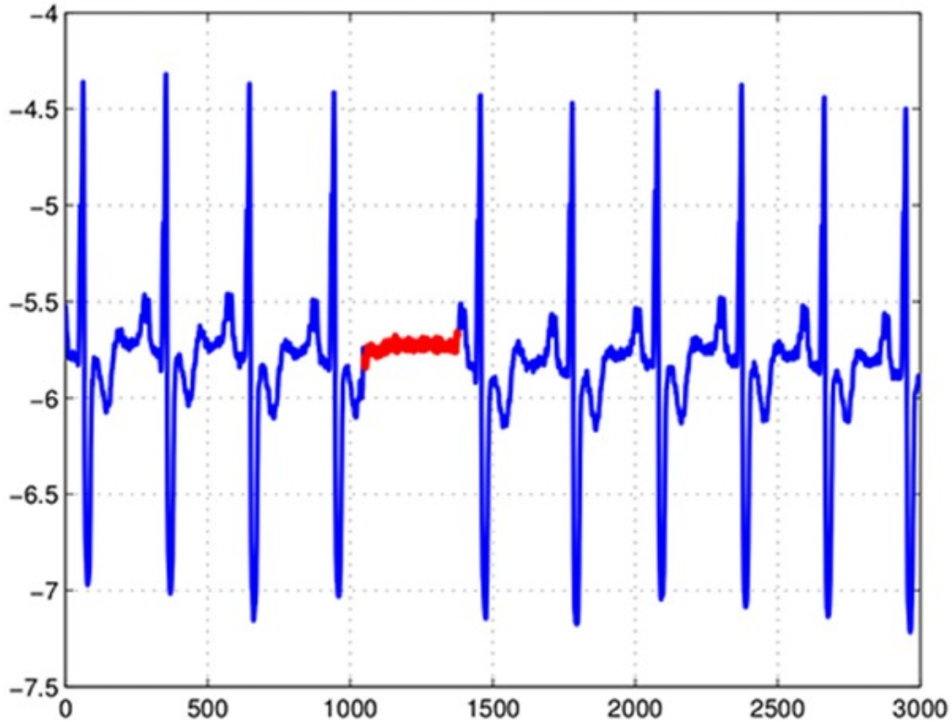

Figure 9: Example of a collective anomaly in an ECG signal: a highlighted region (in red) where the signal maintains approximately the same value for an unusually long duration, forming an anomalous pattern.

4.3 Anomaly Detection Methods

Unlike conventional classification problems, where a labeled training set and a test set are available, anomaly detection has multiple possible configurations depending on the labels available in the dataset. We can distinguish between three main types:

4.3.1 Supervised Methods

Training data contains labeled samples (both normal and anomalous). A classifier is trained to learn patterns that distinguish anomalies from normal data, then classifies new unlabeled data. This approach requires known and correctly labeled anomalies, which limits its applicability. Common algorithms include Support Vector Machines (SVM) and Artificial Neural Networks (ANN). Typically used in applications like fraud detection or medical diagnosis where anomalies are well-defined.

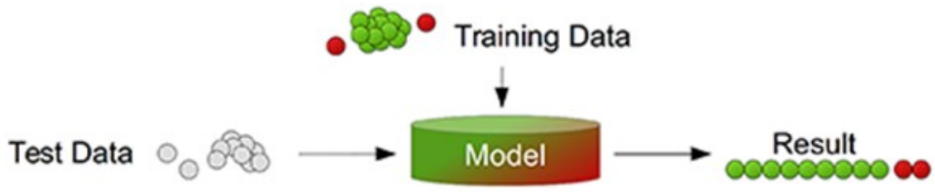

Figure 10: Supervised anomaly detection workflow

4.3.2 Semi-Supervised Methods

Training data contains only normal (non-anomalous) samples. The model learns the normal pattern, then identifies anomalies as deviations from this learned pattern. This approach is known as **one-class classifiers**. Common algorithms include one-class SVM, autoencoders, Gaussian mixture models, and kernel-based density estimation. More practical than supervised methods since it only requires normal samples, not labeled anomalies.

Figure 11: Semi-supervised anomaly detection workflow

4.3.3 Unsupervised Methods

No labeled data is required. The algorithm scores data based solely on intrinsic properties (distances or densities) to identify what is normal versus atypical. This is the most flexible approach. The output is typically a continuous score reflecting the degree of abnormality, allowing instances to be ranked by their anomaly score. Semi-supervised methods also produce continuous scores for ranking suspicious cases.

Figure 12: Unsupervised anomaly detection workflow

4.4 Anomaly Removal

The following explains the most common unsupervised procedures for removing anomalies in signals.

4.4.1 Median Filter

The median filter has been commonly employed on 1D and 2D signals for removing impulsive noise. These artifacts are easily recognized through visual inspection of the signal, as they are associated with peak values that stand out notably from the rest of the signal (point anomaly) or from the immediate neighborhood (contextual anomaly, since the atypical value would be inconsistent with those in its environment).

In images, this type of anomaly is known as **salt & pepper noise**, as the effect it generates is that of randomly placed pixels that take extreme intensity values (1 or 0).

The median filter is an operation applied point by point using a sliding window. The size of this window is determined by the user. For one-dimensional signals such as time series, it is a window of length N, while in images the window is defined in both coordinates and is of size $N \times N$. The value of N is odd, since the window is centered on the point of the signal to be filtered. Thus, the resulting value at this point is given by the median of the points considered by the window.

As can be seen from its definition, the filter does not create new signal values, but selects one of the incoming values as output. The median filter is very similar to an average filter that would obtain, for each window, the mean value of the pixels or points considered. This operation is equivalent to using a low-pass filter in frequency, so rapid variations of the signal, reflected as significant contrasts in an image, are smoothed by the filter.

4.4.2 Statistical Techniques

A common technique for detecting and correcting anomalies relies on using the **probability density function** of the data. Given the density function $f(x)$, where x is one of the values the corresponding random variable can take, a measure quantifying the degree of anomaly for a sample x_1 can be obtained as the inverse of $f(x_1)$.

Values that are very improbable will tend to be identified as atypical (outliers). Therefore, an appropriate strategy must be used for their treatment, such as eliminating them or estimating their value as the mean of neighboring points.

Curious Fact: What is a Probability Density Function?

The **probability density function** (PDF) $f(x)$ describes how the probability is distributed over the possible values of a continuous random variable. For a given value x, the function $f(x)$ tells us the relative likelihood that the random variable will take that value.
Key properties:

- The PDF is always non-negative: $f(x) \geq 0$ for all x
- The area under the entire curve equals 1: $\int_{-\infty}^{\infty} f(x)\,dx = 1$
- Higher values of $f(x)$ indicate that x is more likely to occur
- Lower values of $f(x)$ indicate that x is less likely (more unusual/anomalous)

In anomaly detection, values with very low probability density (low $f(x)$) are considered outliers, as they represent rare or unusual occurrences in the data.

Example: Consider a signal with values following a normal distribution with mean $\mu = 0$ and standard deviation $\sigma = 1$. For this specific distribution, the probability density function is:

$$f(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} \tag{21}$$

Note that this is the PDF formula for a **standard normal distribution**. Different probability distributions have different PDF formulas. The general form for a normal distribution with mean μ and standard deviation σ is:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{22}$$

Figure 13 illustrates the probability density function for the standard normal distribution, showing how the PDF value decreases as we move away from the mean, making outliers easier to identify.

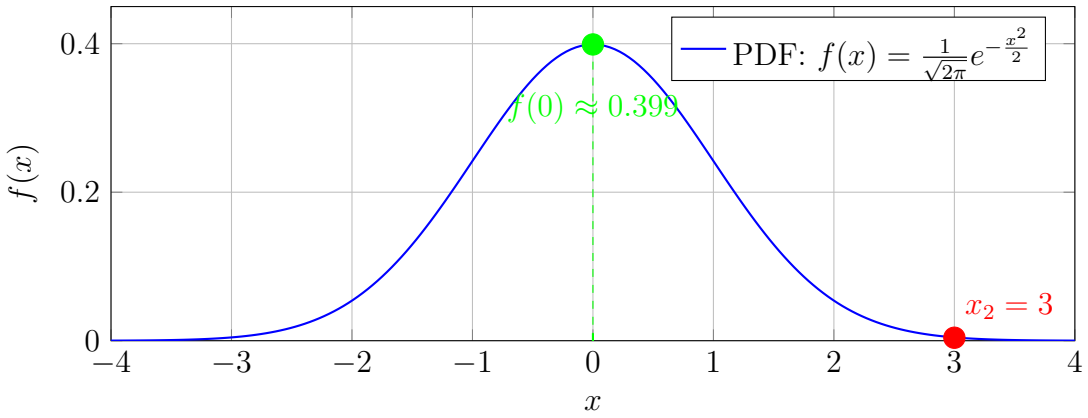

Figure 13: Probability density function of a standard normal distribution ($\mu = 0$, $\sigma = 1$). The green point at $x_1 = 0$ shows a normal sample with high probability density ($f(0) \approx 0.399$), while the red point at $x_2 = 3$ shows an outlier with very low probability density ($f(3) \approx 0.004$).

Step 1: Evaluate the probability density for a normal sample $x_1 = 0$ (at the mean):

$$f(0) = \frac{1}{\sqrt{2\pi}} e^0 \approx 0.399 \tag{23}$$

Step 2: Calculate the anomaly degree for x_1:

$$\text{Anomaly degree} = \frac{1}{f(0)} \approx \frac{1}{0.399} \approx 2.5 \quad \text{(low anomaly)} \tag{24}$$

The anomaly degree of 2.5 is considered "low" because it corresponds to a value at the mean of the distribution (most likely value). In practice, anomaly degrees are evaluated **relative to other samples** in the dataset or compared to a threshold. Common thresholds are based on:

- **Standard deviations**: Values beyond 2-3 standard deviations from the mean
- **Percentiles**: Values below the 1st percentile or above the 99th percentile

- **Relative comparison**: Comparing anomaly degrees across all samples and identifying those significantly higher than the median

Step 3: Evaluate the probability density for a potential outlier $x_2 = 3$ (three standard deviations away):

$$f(3) = \frac{1}{\sqrt{2\pi}} e^{-\frac{9}{2}} \approx 0.004 \tag{25}$$

Step 4: Calculate the anomaly degree for x_2:

$$\text{Anomaly degree} = \frac{1}{f(3)} \approx \frac{1}{0.004} \approx 250 \quad \text{(high anomaly)} \tag{26}$$

Step 5: Since x_2 has a very high anomaly degree (250 vs 2.5), it is identified as an outlier. The comparison shows that x_2 has an anomaly degree **100 times higher** than x_1 ($250/2.5 = 100$), indicating it is extremely unlikely and anomalous. In practice, a threshold would be established (e.g., anomaly degree > 50 or > 100) to automatically flag outliers. Treatment options include: (1) eliminating the value, or (2) replacing it with the mean of its neighboring points.

4.4.3 Threshold-Based Outlier Detection

An alternative statistical strategy identifies outliers as values located at the extremes of the domain of a function $f(x)$. This approach defines two threshold values x_a and x_b such that:

$$P(x \leq x_a) = P(x \geq x_b) = P_{\min} \tag{27}$$

where $P_{\min}$ is a predefined minimum probability threshold. A sample x_1 is considered an anomaly if $x_1 < x_a$ or $x_1 > x_b$.

Global Application (Point Anomalies): When applied globally, the function $f(x)$ represents the entire set of available samples (e.g., all points in a time series or all pixels in an image). This approach detects **point anomalies**, where an individual sample is notably different from the overall dataset.

Local Application (Contextual Anomalies): To identify **contextual anomalies**, the method is applied within a local environment or neighborhood of the point being studied. Here, $f(x)$ is defined solely based on the neighborhood of the point being evaluated. This involves using a **sliding window** centered on the target point, similar to how a median filter operates, to define the local context and establish local thresholds x_a and x_b for that specific neighborhood.

4.4.4 Practical Considerations

Additionally, both methods based on $f(x)$ require initially setting a **decision threshold**:

- For the inverse density method: to compare the obtained anomaly score against a threshold
- For the threshold-based method: to define the $P_{\min}$ value that identifies the extreme values of a distribution

Important: User-Defined Threshold

This threshold determines the definition of anomaly in our dataset and must be established by the user. The choice of threshold directly affects which samples are classified as anomalies, making it a critical parameter in the detection process.

In the definition of methods based on the use of the probability density function $f(x)$, knowledge of this function has been assumed. However, in practice, this function is typically **unknown**, so it is necessary to apply **estimation techniques** to obtain an approximation of it. The following are some techniques that can be used for estimating this function.

4.4.5 Estimation Techniques

Histogram This represents the simplest technique. From the samples of a variable, it is discretized by dividing its domain into a limited number of intervals of equal size, identified by their midpoint. These points represent the discrete values that the variable can take. Thus, the frequency (probability) associated with each possible value is obtained from the total initial dataset by counting the number of samples of the variable that fall into each interval.

The choice of the number of intervals used for discretizing the variable has a very significant influence on the obtained approximation. A number that is too small will result in an excessively simple approximation that does not capture the particularities of the target distribution. However, an excessive number of intervals leads to the resulting estimation presenting discontinuities (null values) and abrupt changes in its profile.

Example: Consider a dataset with 20 samples: $\{1.2, 1.5, 1.8, 2.1, 2.3, 2.4, 2.6, 2.7, 2.9, 3.0, 3.1, 3.2, 3.4$

Step 1: Divide the domain into intervals. For example, using 5 intervals of width 1.0:

- Interval 1: $[1.0, 2.0)$ with midpoint 1.5
- Interval 2: $[2.0, 3.0)$ with midpoint 2.5
- Interval 3: $[3.0, 4.0)$ with midpoint 3.5
- Interval 4: $[4.0, 5.0)$ with midpoint 4.5
- Interval 5: $[5.0, 6.0)$ with midpoint 5.5

Step 2: Count samples in each interval:

- Interval 1: 3 samples $\{1.2, 1.5, 1.8\} \rightarrow$ frequency $= 3/20 = 0.15$
- Interval 2: 7 samples $\{2.1, 2.3, 2.4, 2.6, 2.7, 2.9, 3.0\} \rightarrow$ frequency $= 7/20 = 0.35$
- Interval 3: 6 samples $\{3.1, 3.2, 3.4, 3.5, 3.7, 3.9\} \rightarrow$ frequency $= 6/20 = 0.30$
- Interval 4: 3 samples $\{4.2, 4.5, 4.8\} \rightarrow$ frequency $= 3/20 = 0.15$
- Interval 5: 1 sample $\{5.1\} \rightarrow$ frequency $= 1/20 = 0.05$

Step 3: The estimated probability density function assigns to each midpoint the frequency of its interval. For example, $f(2.5) \approx 0.35$ and $f(5.5) \approx 0.05$. A new sample $x = 5.3$ would fall in interval 5, which has low frequency (0.05), indicating it is likely an anomaly.

Figure 14 visualizes the histogram estimation, showing the frequency bars for each interval. The low frequency in interval 5 (highlighted in red) indicates that values in this range are anomalous.

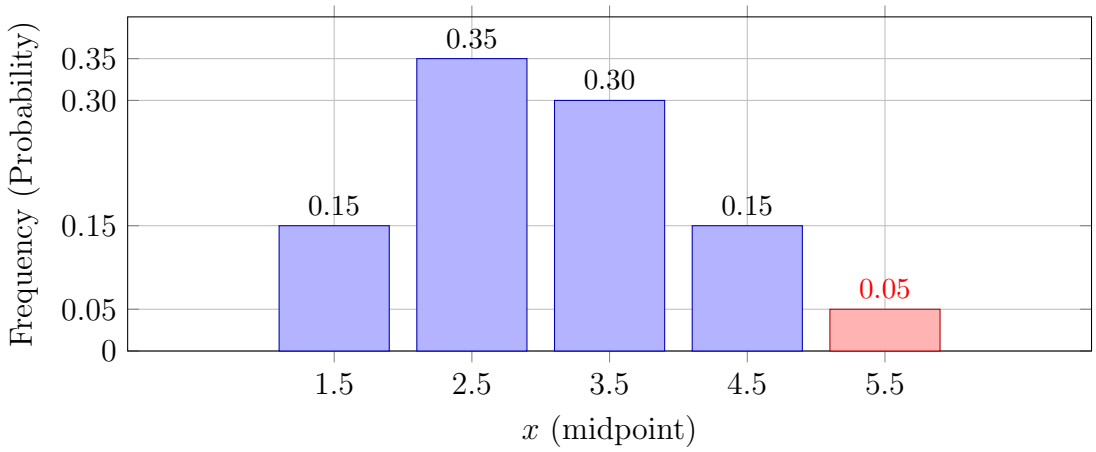

Figure 14: Histogram estimation of probability density function. Each bar represents the frequency (probability) of samples in that interval. The red bar (interval 5) has very low frequency (0.05), indicating anomalous values.

Note: If we had used only 2 intervals, we would lose detail about the distribution shape. If we used 20 intervals (one per sample), many intervals would have zero frequency, creating discontinuities.

Kernel Density Estimation Kernel Density Estimation (KDE) is a non-parametric method that provides a smooth estimate of the probability density function. Instead of using fixed intervals like histograms, KDE places a "kernel" (a smooth function, typically a Gaussian) centered at each data point and sums these kernels to create a continuous density estimate.

The estimated density function is given by:

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \tag{28}$$

where n is the number of samples, h is the bandwidth (smoothing parameter), x_i are the data points, and K is the kernel function. A common choice is the Gaussian kernel: $K(u) = \frac{1}{\sqrt{2\pi}} e^{-\frac{u^2}{2}}$.

Curious Fact: What is a Kernel?

A **kernel** is a smooth, symmetric function that assigns weights to nearby data points. Think of it as a "bump" or "hill" centered at each data point that spreads influence to nearby regions.
Key properties of kernels:

- **Symmetric**: $K(-u) = K(u)$ for all u
- **Non-negative**: $K(u) \geq 0$ for all u
- **Integrates to 1**: $\int_{-\infty}^{\infty} K(u)\,du = 1$ (ensures the density estimate is valid)
- **Peak at center**: Maximum value occurs at $u = 0$

The Gaussian kernel is most common because it's smooth and has infinite support, meaning it gives non-zero weight to all points (though very small for distant points). Other kernel types include uniform, triangular, and Epanechnikov kernels. The kernel essentially "smears" each data point's influence across its neighborhood, creating a smooth density estimate when all kernels are summed together.

The bandwidth h plays a crucial role: a small h creates a detailed but noisy estimate, while a large h produces a smoother but potentially oversimplified estimate.

Example: Using the same dataset as before: $\{1.2, 1.5, 1.8, 2.1, 2.3, 2.4, 2.6, 2.7, 2.9, 3.0, 3.1, 3.2, 3.4, 3.$

Step 1: Choose a bandwidth. For this example, let $h = 0.5$ (moderate smoothing).

Step 2: For any point x, calculate the density estimate by summing Gaussian kernels centered at each data point. For example, at $x = 2.5$:

$$\hat{f}(2.5) = \frac{1}{20 \times 0.5} \sum_{i=1}^{20} \frac{1}{\sqrt{2\pi}} e^{-\frac{(2.5-x_i)^2}{2\times 0.5^2}} \tag{29}$$

This gives a smooth, continuous estimate of the density.

Step 3: For an anomalous point $x = 5.3$, the density estimate will be very low because it is far from most data points. The kernels centered at nearby points (like $x_i = 5.1$) contribute little, and kernels from distant points contribute almost nothing, resulting in $\hat{f}(5.3) \approx 0.02$, indicating an anomaly.

Figure 15 compares the histogram and KDE estimates, showing how KDE provides a smooth continuous curve without discontinuities.

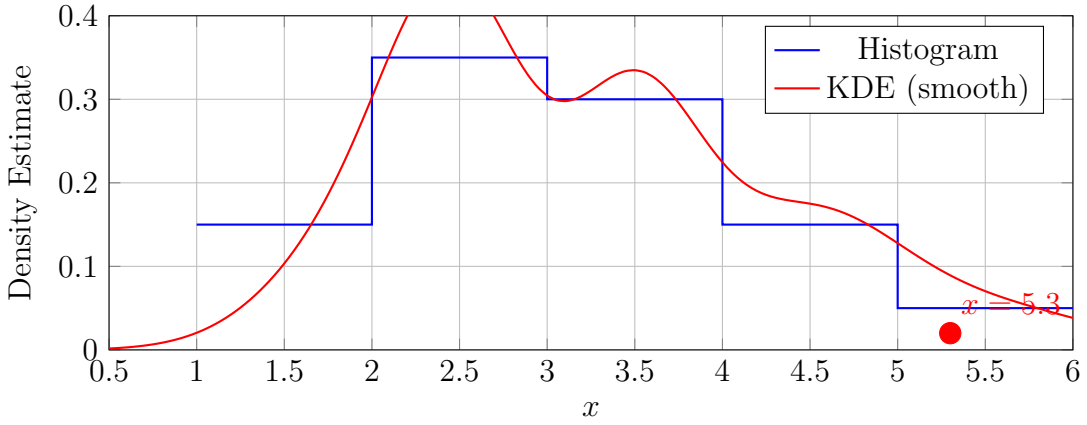

Figure 15: Comparison of histogram and kernel density estimation. The histogram (blue) shows discrete intervals, while KDE (red) provides a smooth continuous estimate. The point $x = 5.3$ has very low density in both methods, indicating an anomaly.

Parametric Estimation It is assumed that the probability density function that statistically characterizes the variable is of normal type. Therefore, the mean and variance of this distribution are the parameters to be obtained. For this, the estimations derived from the available sample are used:

Curious Fact: What is "Normal Type"?

A **normal distribution** (also called Gaussian distribution) is a bell-shaped, symmetric probability distribution that is one of the most important distributions in statistics. It is called "normal" because many natural phenomena approximately follow this distribution.
Key characteristics:

- **Bell-shaped curve**: Symmetric around the mean, with a single peak
- **Completely determined by two parameters**: Mean μ (center) and variance σ^2 (spread)
- **68-95-99.7 rule**: Approximately 68% of data falls within 1 standard deviation, 95% within 2, and 99.7% within 3 standard deviations of the mean
- **Common in nature**: Many measurements (heights, test scores, measurement errors) tend to follow normal distributions due to the Central Limit Theorem

When we say a variable is "of normal type," we mean its probability density function follows the normal distribution formula. This assumption simplifies estimation because we only need to estimate two parameters (μ and σ^2) instead of the entire function shape.

$$\mu_x = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{30}$$

$$\sigma_x^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu_x)^2 \tag{31}$$

Breaking down the equations:
Equation 30 (Sample Mean):

- μ_x: The estimated mean (average) of the data
- n: Total number of samples in the dataset
- $\sum_{i=1}^{n}$: Summation symbol, meaning "add up all values from $i = 1$ to $i = n$"
- x_i: The i-th individual observation/value in the dataset
- $\frac{1}{n}$: Dividing by n to get the average (mean)

In simple terms: Add up all the values and divide by the number of values to get the average.

Equation 31 (Sample Variance):

- σ_x^2: The estimated variance (measure of spread/dispersion)
- n: Total number of samples
- $(x_i - \mu_x)$: The difference between each value and the mean (how far each point is from the center)
- $(x_i - \mu_x)^2$: Squaring the difference (ensures all values are positive and emphasizes larger deviations)
- $\sum_{i=1}^{n}$: Sum all the squared differences
- $\frac{1}{n}$: Average of the squared differences

In simple terms: Calculate how far each value is from the mean, square those distances, then average them. This measures how spread out the data is around the mean.

Once these parameters are estimated, the normal probability density function can be fully specified as:

$$f(x) = \frac{1}{\sigma_x\sqrt{2\pi}}e^{-\frac{(x-\mu_x)^2}{2\sigma_x^2}} \tag{32}$$

This parametric approach is simpler than non-parametric methods but requires the assumption that the data follows a normal distribution, which may not always be valid.

Obviously, the main limitation of this method comes from the initial assumption about the shape of the distribution. The error in the estimation will be more significant, therefore, the more the real distribution of the variable differs from the normal profile initially assumed. If the data is highly skewed, multimodal, or has heavy tails, the normal assumption will lead to poor density estimates and consequently, inaccurate anomaly detection.

Kernel Functions (Parzen Method) This is a hybrid procedure between histogram-based estimation and parametric estimation. In this case, the estimation of the probability density function is given by the superposition of kernel functions centered at each of the initially observed samples x_i. The expression for the estimated function is obtained as follows:

$$\hat{f}(x) = \frac{1}{n} \sum_{i=1}^{n} g(x - x_i, \theta) \tag{33}$$

where:

- $g(x, \theta)$ is the kernel function
- θ represents the set of parameters for this function (e.g., bandwidth h in the Gaussian kernel case)

This formulation is equivalent to the Kernel Density Estimation (KDE) method described earlier, where the kernel function $g(x - x_i, \theta)$ corresponds to $K((x - x_i)/h)$ scaled appropriately. The Parzen window method provides a unified framework that bridges discrete histogram methods and continuous parametric approaches.

Example: Using the same dataset: $\{1.2, 1.5, 1.8, 2.1, 2.3, 2.4, 2.6, 2.7, 2.9, 3.0, 3.1, 3.2, 3.4, 3.5, 3.7, 3.9$ with $n = 20$ samples.

Step 1: Choose a kernel function and its parameters. For this example, we use a Gaussian kernel with bandwidth $h = 0.5$:

$$g(x - x_i, \theta) = g(x - x_i, h) = \frac{1}{h\sqrt{2\pi}} e^{-\frac{(x-x_i)^2}{2h^2}} \tag{34}$$

Step 2: To estimate the density at any point x, sum the kernel functions centered at each data point. For example, at $x = 2.5$:

$$\hat{f}(2.5) = \frac{1}{20} \sum_{i=1}^{20} \frac{1}{0.5\sqrt{2\pi}} e^{-\frac{(2.5-x_i)^2}{2\times 0.5^2}} \tag{35}$$

Each term in the sum represents the contribution of one data point x_i to the density estimate at $x = 2.5$. Points closer to 2.5 contribute more (higher kernel value), while distant points contribute less.

Step 3: For an anomalous point $x = 5.3$, most kernel functions centered at the data points will have very small values because 5.3 is far from most samples. The only significant contribution comes from the kernel centered at $x_i = 5.1$, resulting in $\hat{f}(5.3) \approx 0.02$, indicating an anomaly.

Visualization: The Parzen window method creates a smooth density estimate by placing a "bump" (kernel) at each data point and summing all bumps. The height of the resulting curve at any point x represents the estimated probability density, with low values indicating potential anomalies.

Figure 16 illustrates the Parzen window method, showing individual kernel functions (Gaussian bumps) centered at sample data points and the resulting density estimate obtained by summing all kernels.

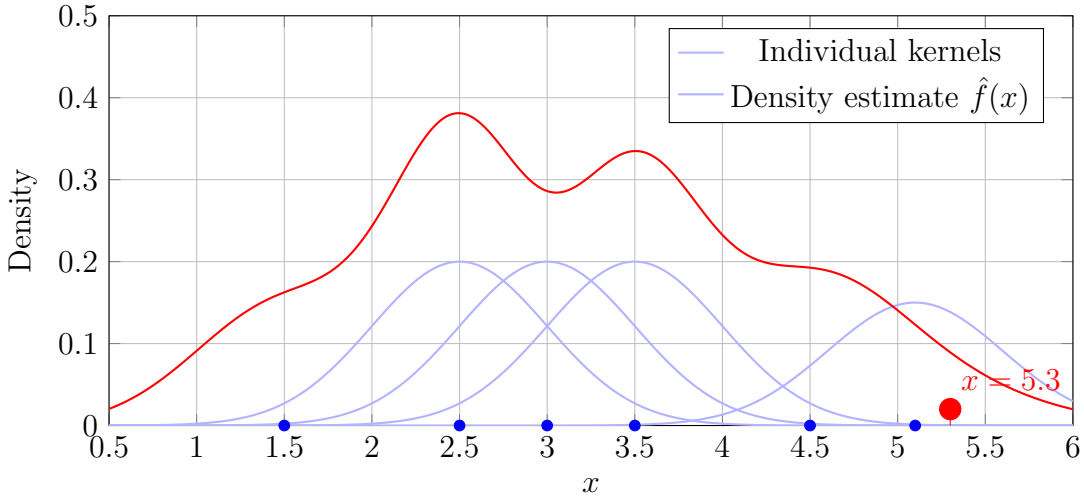

Figure 16: Parzen window method visualization. Individual Gaussian kernels (blue, semi-transparent) are centered at each data point. The resulting density estimate (red curve) is obtained by summing all kernels. The point $x = 5.3$ has very low density, indicating an anomaly.

Commonly, a Gaussian normal is used as the kernel function, so that the parameter set θ is given solely by the variance of the normal, since each kernel function is centered at the corresponding sample. It is common to use the same variance value for all kernel functions, so the estimated probability density function would be obtained as:

$$\hat{f}(x) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(x - x_i)^2}{2\sigma^2} \right] \tag{36}$$

where σ^2 is the variance parameter (bandwidth) that controls the width of each Gaussian kernel. As observed, the effect of the variance of the normal kernel functions is similar to the interval size for histogram calculation. In fact, the histogram can be seen as a particular case of kernel-based estimation, in which these functions would be given by uniform pulses of unit height centered at the midpoint of each interval.

Understanding the Relationship: Histogram vs. Kernel Estimation

Similarity: Both histogram interval size and kernel variance control the **smoothing** of the density estimate:

- **Large histogram intervals** or **large kernel variance** σ^2: Produce smoother, less detailed estimates (may oversimplify)
- **Small histogram intervals** or **small kernel variance** σ^2: Produce more detailed, potentially noisy estimates (may overfit)

Histogram as a special case: A histogram can be viewed as kernel estimation using:

- **Uniform kernels** (rectangular pulses) instead of smooth Gaussian kernels
- Kernels of **unit height** and **width equal to the interval size**
- Kernels **centered at interval midpoints** rather than at individual data points

The key difference is that histograms use **discrete, non-overlapping** uniform kernels, while kernel density estimation uses **continuous, overlapping** smooth kernels (typically Gaussian), resulting in a smoother density estimate.

A common rule for obtaining an adequate value of the variance of kernel functions is to set it to the following value:

$$\sigma = 1.06\sigma_x n^{-1/5} \tag{37}$$

where σ_x is the standard deviation of the sample data and n is the number of samples. This is known as **Silverman's rule of thumb** for bandwidth selection. The factor 1.06 is optimal for Gaussian kernels when the underlying distribution is approximately normal, and the term $n^{-1/5}$ ensures that the bandwidth decreases as the sample size increases, allowing for more detailed estimates with more data.

This rule provides a good starting point for kernel variance selection, balancing between oversmoothing (too large σ) and undersmoothing (too small σ).