

1

Basic Methods and Tools

1.1 General Approach

This first part is devoted to a synthesis of the different methods and concepts used in the various fault diagnosis and failure prognosis techniques (Figure 1.1).

Whatever the approach, these methods require informative data. Therefore we dedicate the first section to the signals that can be extracted from the drive system (Section 1.2). Thus, the different technology of sensors will be presented (Subsection 1.2.1). From these signals are then extracted time, frequency, statistical (Subsection 1.2.2), or model (Subsection 1.2.3) parameters. They could come also from analytical redundancies (Subsection 1.2.4).

However some parameters may be redundant and the problem may be overdetermined. Therefore, a compromise must be found between a minimum number of parameters and sufficient information (Section 1.3). Thus, two main techniques for reducing the size of the pattern vector will be presented: the principal component analysis (Subsection 1.3.1) and the intercorrelation, which measures the dependency of the parameters (Subsection 1.3.2). The quality of the information is measured with the entropy (Subsection 1.3.3). Then, the main techniques for selecting the best parameters in the case of supervised clustering (Subsection 1.3.4) and unsupervised clustering (Subsection 1.3.5) will be detailed.

Another important problem is the determination of the best number of classes (Subsection 1.3.6), especially in the case of unlabeled data. Some criteria are given in the Subsection 1.3.7 to evaluate the quality of the classification.

Diagnosis is mainly based on classifiers (Section 1.4). They allow automatic classification of the unknown states into healthy class or faulty ones. Some generalities are presented in Subsection 1.4.1. Then the supervised classifiers (Subsection 1.4.2) and unsupervised ones (Subsection 1.4.3) are detailed.

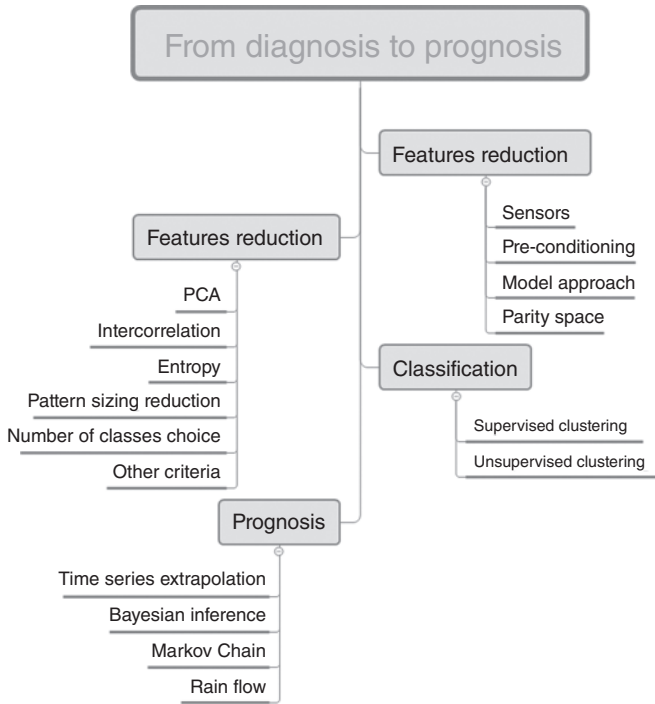


Figure 1.1 General overview of Diagnosis and Prognosis methods

Finally, the methods of failure prognosis are discussed (Section 1.5). First, the general process is described in Subsection 1.5.1. Some techniques are detailed:

- the extrapolation of time series (Subsection 1.5.2) that can be used to forecast the pattern vector,
- the Bayesian inference (Subsection 1.5.2),
- the Markov chain (Subsection 1.5.3) and the hidden Markov chain, which allow the diagnosis to be addressed in terms of stochastic process,
- the rain flow (Subsection 1.5.6) that joins the reliability approaches.

1.2 Feature Extraction: Signal and Preconditioning

1.2.1 Raw Signals: What Kind of Signals and Sensors?

Diagnostic tools need sensors, which are adapted to the faults that are being investigated. The choice of these sensors depends on the nature of the faults, the desired bandwidth, the influence signals but also, of course, the cost and

Sensor	Electrical faults of converter		Electrical faults of actuator	Mechanical faults of actuator
Current and voltage				
Vibration				
Temperature				
Field				
acoustic				



Figure 1.2 Application fields for sensors

the complexity of the interface. The reader can refer to Webster [1998] and Asch [2017] for a more in-depth study of the different types of sensors.

Current sensors and voltage ones are mainly used in motor current signature analysis (MCSA) applications to detect electrical faults in inverter and actuators. They can also give indications of bearing and gear defects.

Vibration and noise sensors are used for the mechanical part of the drive. They also give indications of abnormal torque, eccentricity, and thus the magnetic behavior of the actuator.

Temperature sensors are used in the detection of converter and actuator faults but with a bandwidth of a few hundred Hz maximum.

Axial field sensors are used to detect some defects linked to a magnetic asymmetry (eccentricity, magnetomotive force asymmetry ...).

1.2.1.1 Current Sensors

Current sensors (and voltage sensors) are used for the control, protection, and supervision of electrical drives. Their choice depends on the bandwidth, the influence quantities, the necessity or not of a galvanic isolation, the possibility to measure DC currents and their price. The following tables give the main characteristics of the different technology (Clerc [1989]) and (Asch [2017]).

Table 1.1 gives the characteristics of the shunts (Clerc [1989]).

Table 1.1 Current sensors: shunts 1/8

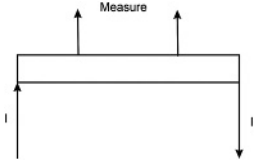
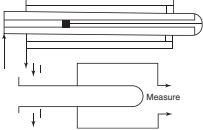
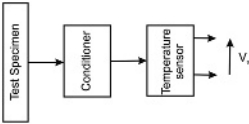
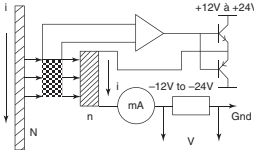
Characteristics	Classical shunt	Coaxial shunt
Schematic diagram		
Transfer function	$U = R_{shunt} \cdot I$	$U = R_{shunt} \cdot I$
Galvanic insulation	No	No
DC measurement	Yes	Yes
Frequency bandwidth	Few hundreds of Hz	Up to 1MHz
Output level	Very low: Few tens of mV	Very low: Few tens of mV
Interface	Differential amplifier with high common mode rejection	Differential amplifier with high common mode rejection
Losses	Joule losses: $R_{shunt} \cdot I^2$	Joule losses: $R_{shunt} \cdot I^2$
Influence quantities	-Frequency: Skin effect and interaction with inductances and capacitances of the structure. -Temperature. -Electromagnetic interference.	-Skin effect for $f > 1\text{MHz}$ -Interaction of inductances and capacities of the structure, -For the U-shaped structure, edge effects increase the influence of the eddy currents -Temperature.
Limitations	-Sensitivity to interferences for low I. -Long-term overloads degrade the structure by overheating.	Intensity is limited by: -the heating -the electrodynamic stresses that can distort too thin walls.
Remarks	- Major drawback: low output level - Advantage: its price. - with inductances and capacitances of the structure. -Temperature: Depends on the material being used. -Electromagnetic interference.	It is more used in proto- types than in industrial series. Major drawback: low output level and its price for high current.

Table 1.2 gives the characteristics of sensors based on temperature and current comparator working at zero flux (Clerc [1989]).

Table 1.2 Current sensors: Temperature-based sensor and current comparator 2/8

Characteristics	Temperature sensor	Current comparator
Schematic diagram		
Transfer function	$I \rightarrow T \xrightarrow[\text{Thermocouple, Quartz, Resistor}]{\text{Temperature Sensor}} f(T) \xrightarrow[\text{Amplifier Bridge}]{\text{Conditioner/Interface}} V$	$N.I = n.i$
Galvanic insulation	Yes	Yes
DC measurement	Yes	Yes
Frequency bandwidth	The response time depends on the test specimen > 1s.	A few tens of kHz Response times < 1μs
Output level	Low but takes advantage of galvanic isolation. Before amplifier: a few mV	Current A few hundred mA
Interface	For thermocouple: amplifier. For resistance: Bridge and amplifier or potentiometer circuit.	Supply: A few W The interface is described on the schematic diagram.
Losses	Depends on the temperature sensor and the test specimen. Very low with thermocouple.	Average
Influence quantities	Ambient temperature. Supply voltage for the conditioner. Humidity for the 8 resistive sensors.	Temperature. Stray fluxes caused by neighboring conductors.
Limitations	Response time. Internal thermal coupling.	Rather poor reliability. Withstands overloads.

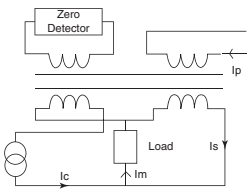
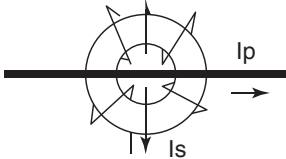
(Continued)

Table 1.2 (Continued)

Characteristics	Temperature sensor	Current comparator
Remarks	Temperature sensor:	Source of errors
	Resistance, Thermocouple,	-Magnetic circuit remanence
	Quartz oscillator, Diode.	-Offset current
	Test Specimen:	-Magnetic interference
	Resistance or semiconductor.	-Temperature drift

Table 1.3 gives the characteristics of a current comparator with a detection coil and of a current transformer.

Table 1.3 Current sensors: Current comparator with detection coil and Current transformer 3/8

Characteristics	Current comparator with detection coil	Current transformer
Schematic diagram		
Transfer function	$N_1 \cdot I_p = N_2 \cdot I_m$	$N_p \cdot I_p - N_s \cdot I_s = I_m \approx 0$
Galvanic insulation	Yes	Yes
DC measurement	No	No
Frequency bandwidth	A few kHz	Limited to few kHz
Output level	Current	Current
Interface	An interface is given in the remarks. The adjustment of G and C can be controlled by the signal of the zero detector. An auxiliary source must provide the compensation current.	TI must have a low input impedance. Current/voltage transformer, must have a high input impedance

(Continued)

Table 1.3 (Continued)

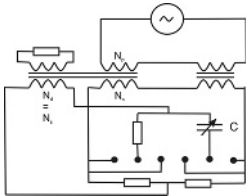
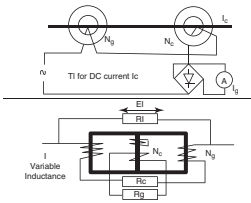
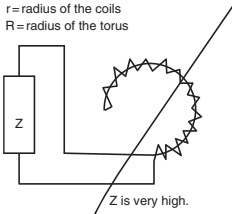
Characteristics	Current comparator with detection coil	Current transformer
Losses	Part of the measurement energy is taken from the main circuit.	Eddy current and Joule losses. Measurement energy pick-up on the primary.
Influence quantities	They have minimal impact: - interference fluxes caused by neighboring conductors. -frequency (parasitic capacitances).	Stray fluxes caused by neighboring conductors. Load: (which modifies the TI characteristics)
Limitations	Parasitic fields, parasitic capacitances between windings, and between windings and conductive screens.	Overload saturation. Self-heating can cause deterioration of the insulation.
Remarks		Sources of errors: -magnetizing current -leakage inductances and winding resistances. Saturation can protect measuring devices by limiting I_S

Table 1.4 gives the characteristics of the transducer and of the Rogowski coil. These later give an image of the current gradient.

Table 1.4 Current sensors: Transducer and Rogowski coil

Characteristics	Transducer	Rogowski Coil
Schematic diagram		

(Continued)

Table 1.4 (Continued)

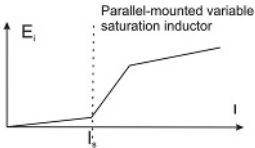
Characteristics	Transducer	Rogowski Coil
Transfer function	 $I_s = I \cdot \frac{N_c}{N_g} \cdot \left(1 + \frac{R_g}{R_l}\right)$ TI for DC current: $N_c \cdot I_c = N_g \cdot I_g$	$E = -\mu_0 \frac{N}{l} \cdot \frac{dI}{dt} \cdot S$ with $\frac{N}{l}$ linear density of turns
Galvanic insulation	Yes	Yes
DC measurement	Yes	No
Frequency bandwidth	Limited to a few kHz	Bandwidth from 0.1Hz to 100kHz. Propagation time: 10μs
Output level	Current	Voltage
Interface	Power Supply: Few tens of W. It is recommended to use a supply signal of a few kHz.	Measurement interface with high input impedance. Differential amplifier
Losses	Important	Very low
Influence quantities	Stray fluxes caused by the neighboring conductors.	- Frequency, - Sensitivity to external magnetic fields very attenuated by the feed-back cable.
Limitations	Saturation on overload	- Minimum measurable voltage. - Breakdown voltage of windings. - Accuracy requirements
Remarks	Drawback: High cost and High consumption.	-Excellent linearity. -Not suitable for low $\frac{dI}{dt}$

Table 1.5 gives the characteristics of the internal inductive flow sensor and of the coreless full shield probe.

Table 1.5 Current sensors: Internal inductive flow sensor and coreless full shield probe 5/8

Characteristics	Internal inductive flow sensor	Coreless full shield probe
Schematic diagram		
Transfer function	$i = -\frac{L}{n.R_0} \cdot \frac{dI}{dt}$ with $\frac{L\omega}{R_0} \ll 1$ and n number of turns	$I_s = \frac{I}{n}$
Galvanic insulation	Yes	Yes
DC measurement	No	No
Frequency bandwidth	Similar to the Rogowski coil.	Similar to the Rogowski coil.
Output level	Current	Current
Interface	Low input impedance measuring device.	Low input impedance measuring device.
Losses	Very low.	Very low.
Influence quantities	-Frequency. -Sensitivity to external magnetic fields very attenuated by the feedback cable. -Temperature	Sensitivity to external magnetic fields very attenuated by the feedback cable.
Limitations	R_0 must be kept independent of -the temperature and -the frequency.	- They depends on -mechanical resistance, -the heating, -the skin effect.
Remarks	Not suitable for low $\frac{dI}{dt}$	

Table 1.6 gives the characteristics of the Hall effect sensor and of the Magnetoresistive probe.

Table 1.6 Current sensors: Hall effect sensor and magnetoresistive probe 6/8

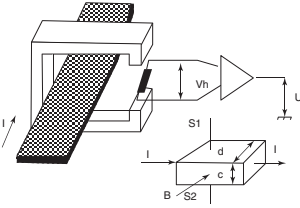
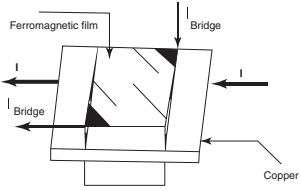
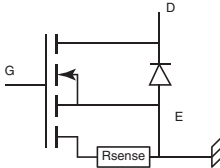
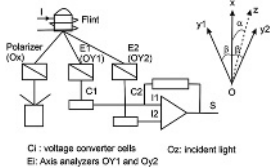
Characteristics	Hall effect sensor	Magnetoresistive probe
Schematic diagram		
Transfer function	$V_h = K.I$ and $B = k.I$ with $K = \frac{R_h.k.i}{d}$ and $R_h = \frac{1}{N.e}$ and $e = 1.610^{-19}C$	R increases with H_s with a good polarization, the relation is linear.
Galvanic insulation	Yes	Yes
DC measurement	Yes	Yes
Frequency bandwidth	Few kHz	Sensor on Fe substrate: 1MHz Sensor on epoxy substrate: 10MHz Sensor with interface : < 10Mhz
Output level	Sensitivity: $\approx 2.5V/T$ Output : few Volts for $I = 1000A$	For $\Delta B = \pm 1T$, $5 < \frac{R}{R_0} < 15$
Interface	Supply: Few mW Interfaces: -differential amplifier -Summators for serial connection of sensors.	Wheatstone bridge and its supply.
Losses	Very low	Very low
Influence quantities	External magnetic field Temperature	Temperature Supply voltage Amplifier Gain External magnetic field
Limitations	Aging Saturation on overload Limited bandwidth	For low fields, noise in $1/f$ Used to measure fields from a few nT to tens of nT.
Remarks	Sources of errors: -offset of the hall effect generator -magnetic circuit hysteresis	We can supply the bridge with an alternative source to improve the behavior.

Table 1.7 gives the characteristics of the current mirror (especially for MOS-FET transistor) and of the optical sensor.

Table 1.7 Current sensors: Current mirror and optical sensor 7/8

Characteristics	Current mirror	Optical sensor
Schematic diagram		
Transfer function	$\frac{I_D}{I_{sense}} = \frac{N \cdot R_{DSOn} + R_{sense}}{R_{DSOn}}$ with R_{DSOn} the on state resistance for all the output cells and N the ratio of the number of output cells to the number of measurement cells	$\alpha = p \cdot I \cdot H$ $i_1 = 2A_1 \cos^2(\beta - \alpha)$ $i_2 = 2A_2 \cos^2(\beta + \alpha)$ For $A_1 = A_2 = A$ $\Delta i = A \sin(2\alpha) \sin(2\beta)$ with A_1, A_2, and p coefficients.
Galvanic insulation	No	Yes
DC measurement	Yes	Yes
Frequency bandwidth	Bandwidth around 1Mz	Around ten kHz
Output level	Current	The output is the polarization angle of rotation. Typically $p = 10^{-5} \text{ rad/A}$ for a silicon fiber
Interface	Interface must have a low input impedance	The polarization angle is converted into current or voltage
Losses	Low	Almost no loss.
Influence quantities	The transfer function depends on the junction temperature, the collector current and the frequency	-Temperature (but a little) -p depends on the wavelength

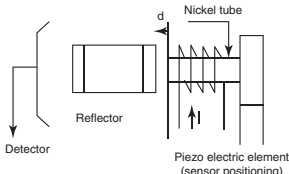
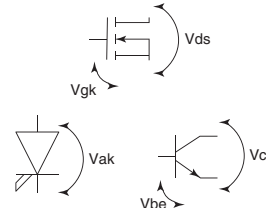
(Continued)

Table 1.7 (Continued)

Characteristics	Current mirror	Optical sensor
Limitations	Nonlinearity in case of overload Low accuracy : $\approx 5\%$	Depends on the sensor. Intensity from mA to 100kA with a resolution of $1.10^{-4} A$ (SAGNAC interferometer)
Remarks	Low price sensor	High voltage applications. Notations: α : rotation of the polarization plan l: length subjected to the field p: Verdet constant

Table 1.8 gives the characteristics of the laser diodes and presents the use of the characteristics of the power devices.

Table 1.8 Current sensors: Laser diodes and device characteristics 8/8

Characteristics	Laser diodes	Device characteristics
Schematic diagram		
Transfer function	$\begin{aligned} &\Delta I \\ &\downarrow \\ &\Delta H \\ &\downarrow \\ &\Delta I \text{ of the nickel tube} \\ &\downarrow \\ &\Delta I \text{ of the reflector} \\ &\downarrow \\ &\Delta P \text{ emitted by} \\ &\text{the laser diode} \end{aligned}$	Transistors: $V_{CE}(I_c)$ and $V_{BE}(I_c)$ MOS: $V_{ds} = R_{on} I_d$ Thyristors: $V_{AK}(I_c)$

(Continued)

Table 1.8 (Continued)

Characteristics	Laser diodes	Device characteristics
Galvanic insulation	Yes	No
	Yes	Yes
Frequency bandwidth	In the range of some kHz	Response time: $\approx 1\mu s$
Output level	Voltage or current	Voltage
Interface	Photodetector output signal processing.	The interfaces must correct the thermal drift.
Losses	Very low	Almost zero
Influence quantities	Laser wavelength	Temperature
	Initial position of the reflector	Drive
	Temperature	Nonlinearity of the transfer function
Limitations	Minimum field strength: $10^{-10}T$. From $10^{-5}A$ to $10A$	Essentially set by the accuracy of the triggering threshold.
Remarks	Improvement of detection conditions: -For variable fields, superimpose a continuous field -Precise positioning of the reflector	Advantages: -Low price -Fast response time Drawback: influence quantities

1.2.1.2 Vibration Measurement and Accelerometers

The diagnosis of mechanical failures or excessive eccentricity of electrical machines often uses vibration measurements (see Table 1.9).

Three type of sensors are used to realize a vibration analysis:

- displacement in mm ,
- speed in mm/s ,
- acceleration in mm/s^2 or in g ($g = 9.81m.s^{-2}$).

In this paragraph, we will focus on one of the most used vibration sensors: the acceleration sensors.

An accelerometer is made of a damped mass (called proof mass) and a spring. The acceleration is given by the proof mass movement. The system is damped in order to decrease the measurement disturbances (see Figure 1.3).

Table 1.9 Vibration sensors

Sensors	Bandwidth	Application
Displacement	low frequency: $f \leq 200Hz$	Balourd, misalignment coupling wear, loosening
Speed	medium frequency: $f \leq 1kHz$	Balourd, misalignment coupling wear, loosening faulty meshing highly degraded bearing
Acceleration	high frequency: $f \leq 20kHz$	Faulty meshing, regular degradation or highly degraded bearing, lack of lubrication

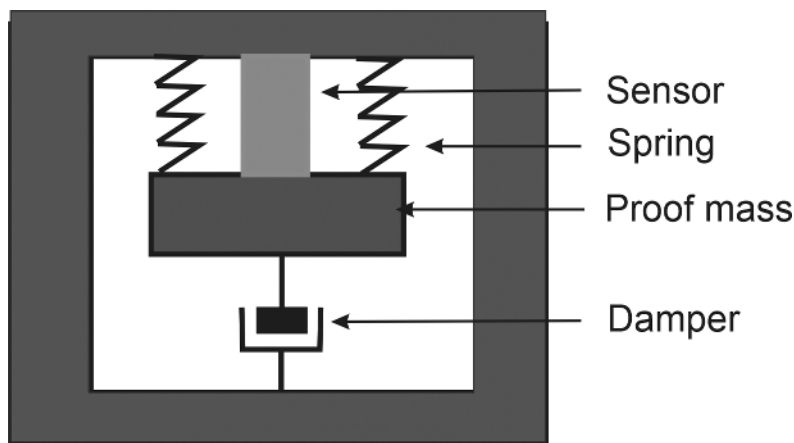


Figure 1.3 Accelerometer

The measurement of the displacement between the proof mass and the frame can be done with piezoelectric, piezoresistive, or capacitive sensors. Piezoelectric accelerometer can measure acceleration up to 100g ($g = 9.81m.s^{-2}$), a shock resistance up to 1,000,000g and a bandwidth greater than 50kHz.

There are single accelerometers that measure in only one direction and multi-axis accelerometers that can measure simultaneously in three directions.

1.2.1.3 Temperature Sensors

Temperature is a key feature for diagnosis and prognosis of electrical drives. But its dynamics, in relation to the thermal constants of the circuit and

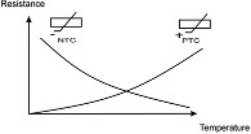
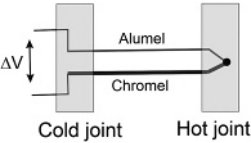
the temperature sensors, is significantly slower than that of the electrical and mechanical quantities. Nevertheless, the temperature can be a component of the pattern vector or an influencing parameter. It is used for both motor (Mohammed et al. [2019]) and converter diagnosis (van der Broeck et al. [2018]).

The main thermal sensors are:

- Thermistor;
- Thermocouple;
- Resistance temperature detector (RTD);
- Fiber optic thermal sensor;
- Thermo sensitive parameter (TSEP). They will be fully described in Subsection 2.6.6.2.

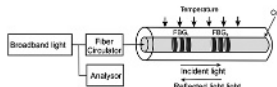
Table 1.10 gives an overview of the temperature sensors.

Table 1.10 Temperature sensors

Thermistor 	Thermistors use the variation of resistance as a function of temperature. CTN (semiconducting material) has a Negative Temperature Coefficient, CTP (barium titanate) has a Positive Temperature Coefficient, Temperature Range: -100 to 350°C Response time: 0.1 to 10s Accuracy: 0.05 to 1.5°C
Thermocouple 	Thermocouples use the Seebeck effect: Two wires of different metals, welded at one of their ends, making a circuit without current flow, when they are submitted to a temperature gradient can exhibit a gradient of voltage at the end of the open loop. Temperature Range: -200 to 2500°C Response time: 0.10 to 10s Accuracy: 0.5 to 5°C
Resistance temperature detector (RTD) (RTD PT100 and PT1000) 	RTD are often made of a length of Platinum wire wrapped around a ceramic or glass core. They are based on the increasing of the resistance with the temperature. Temperature Range: -200 to 650°C Response time: 1 to 10s Accuracy: 0.1 to 1°C

(Continued)

Table 1.10 (Continued)

Fiber optic thermal sensor 	They are mostly based on (Roriz et al. [2020] and Malekzadeh et al. [2014])): - Fiber Bragg Grating (using the selective photo-induced modulation of the optical fiber core refractive index). - Fabry-Pérot interferometer. Temperature Range: Typically -100 to 400°C Response time: 5ms Accuracy: Typically $\pm 0.5^{\circ}\text{C}$
Thermo sensitive parameter	Junction temperature is estimated from some electrical characteristics of the device such as V_{CE}, ... Temperature Range: limited by the junction temperature Response time: depends on the interface - a few ms Accuracy: $> 1^{\circ}\text{C}$. Sensitive to different parameters.

1.2.1.4 Field Sensors

These sensors are mainly made of a measurement coils distributed in the machine body (non intrusive case) or placed inside the machine (intrusive case) (Ewert [2017]).

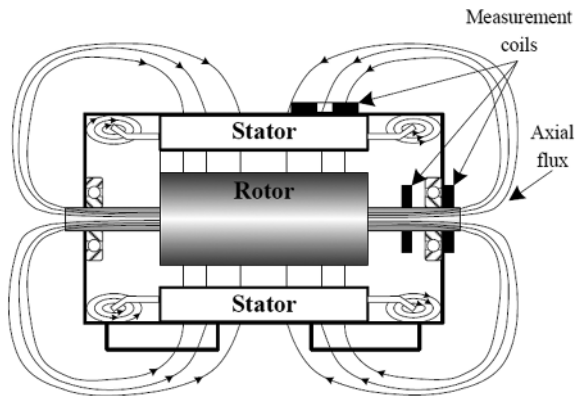


Figure 1.4 Axial flux distribution in an induction motor and possible distribution of measurements coils (Ewert [2017]). Courtesy of IEEE

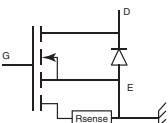
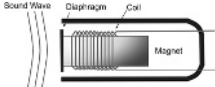
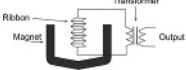
1.2.1.5 Acoustic Sensors

Many machine breakdowns can be detected acoustically since machines consist of moving parts inducing friction and noise.

Pressure and speed can be used to measure an audible signal. But most sensors use sound pressure. Table 1.11 gives the main technologies (Wikipedia contributors [2020a]).

The choice of the acoustic sensors depends on:

Table 1.11 Acoustic sensors

Condenser microphone 	Electrostatic microphone : a thin metallic membrane is placed near a solid plate and acts as a variable capacitor. Bandwidth: few 10kHz
Dynamic microphone 	Moving-coil microphone: The diaphragm drives a coil moving in a magnetic field and induce a voltage. Bandwidth: few 10kHz
Ribbon microphones 	A thin metal ribbon is suspended in a magnetic field, and moves under the effect of a sound wave, creating an induced voltage. Bandwidth: few 10kHz
Piezoelectric microphone 	They use piezoelectric materials which have the ability to produce a voltage when subjected to pressure Bandwidth: few tens kHz
Fiber optic microphones	They are based in a device in which the sound waves modulate the light beam guided in optical fibers. They are use in harsh environment. Bandwidth: few kHz
Micro Electrical-Mechanical System (MEMS) microphones	A pressure-sensitive diaphragm is mounted directly into a silicon wafer using MEMS processing techniques. Bandwidth: up to 1MHz

- the polar patterns: they are the locus of points with the same signal level output from the microphone for a given sound pressure level generated from that point;

- the bandwidth;
- the output impedance.

These sensors can be used, among other things, to detect gear defects (Zurita et al. [2013]), bearing faults (Xue et al. [2017]), stator winding delamination (Zhu et al. [2010]), etc. They have bandwidth up to 1MHz.

1.2.1.6 Other Sensors

Voltage sensors can be used to monitor the state of capacitors and batteries. But, they are also necessary to evaluate the transient power for some diagnosis

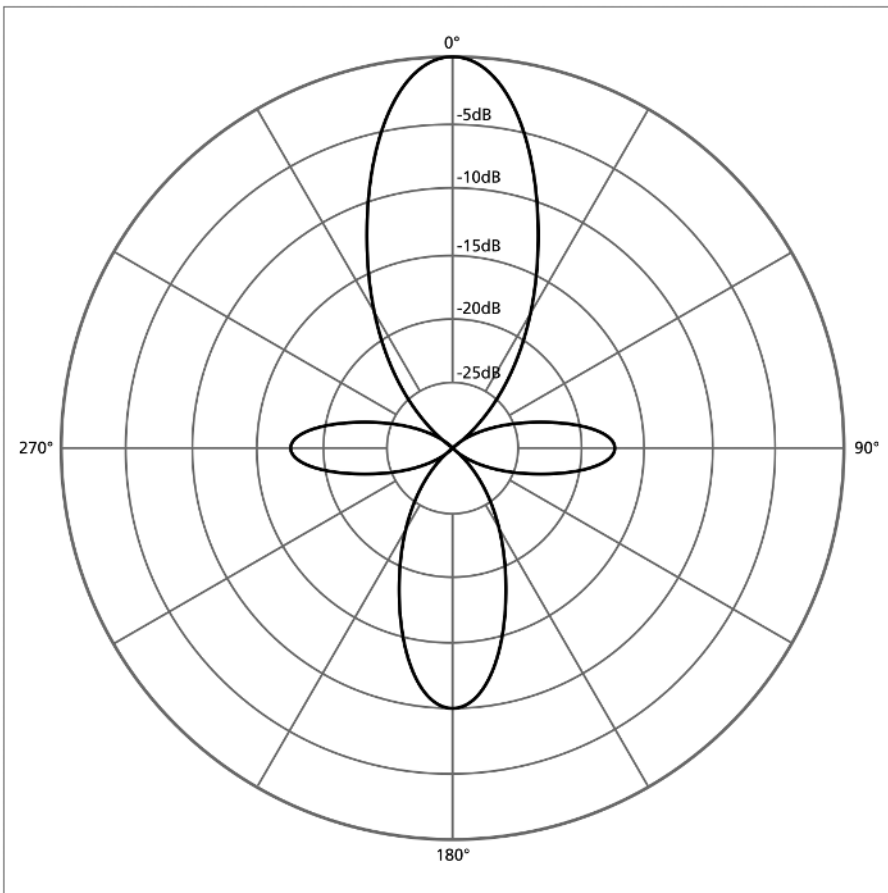


Figure 1.5 Lobar polar pattern

methods of semiconductor devices or actuators. In the latter case, particular attention must then be paid to synchronization with the current channels.

Voltage meters could be (Webster [1998]):

- electromagnetic: they are based on the interaction between currents and magnetic fields ($F = N.B.l.I$ with F the force, l the length of conductor cutting the magnetic field, B and N the number of turns of the coil). Some of them only measure DC voltage (with moving coil or magnet);
- electrodynamic: they are based on the interaction between currents ($F = k.I_1.I_2$);
- electrostatic: they are based on electrostatic interactions between the conductors (for a moving plate displaced by ds , $F = U^2 \frac{dC}{ds}$ with C the capacitance between conductors);
- thermal: they are based indirectly on thermal effects;
- inductive: they are based on magnetic induction (Voltage Transformers) and only measure AC voltage;
- electronic: they are based on digital to analog interaction and signal processing.

Speed sensors are often integrated in the control. They can enter in the evaluation of the mechanical torque from the absorbed power. However, the inertia strongly smoothes the signal, which makes them unsuitable for fast defect detection. Indicators based on speed are more relevant for defects whose vibratory impact appears in low and medium frequencies.

Two main technologies are used:

- **Tachometer generator:** It could be a DC current Tachometer generator. The DC no-load e.m.f. is proportional to the speed (see Figure 1.6).
It could be also a synchronous AC tachometer generator where the AC no-load electromotive force (e.m.f.) is proportional to the speed.
- **Pulse angular speed tachometers:** A train of pulses whose frequency is proportional to the speed is generated. It could be a variable reluctance sensor, a magnet sensor with Hall effect receiver, or an optic sensor.

Position could be used in order to find the orientation of the frame where raw signals are projected (see paragraph Symmetric component, Park Component 1.2.2.2). It is also used in diagnostic methods related to cyclostionarity in particular for bearing defects, etc. (Wu et al. [2009]).

There are two types of position sensors:

- The incremental encoder: at least two square waves are generated in quadrature A and B, whose frequency is proportional to the speed. The direction

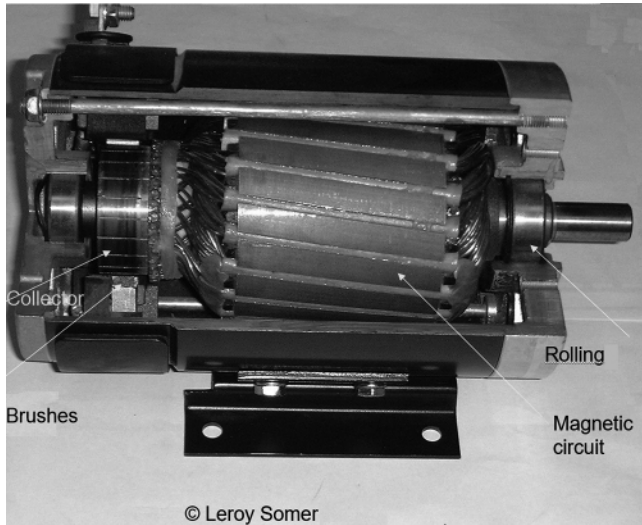


Figure 1.6 Direct current generator @LeroySomer

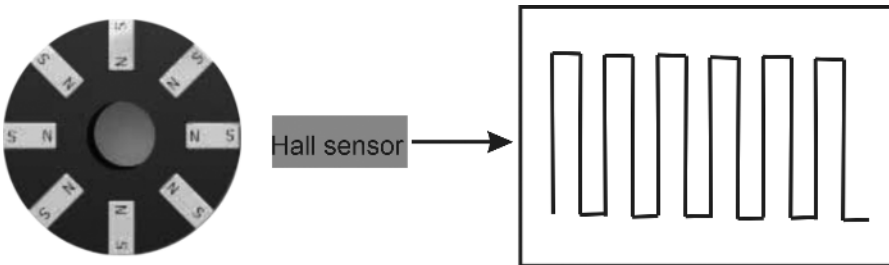


Figure 1.7 Pulse angular speed tachometer

of rotation is indicated by the sign of the phase angle A and B (Figure 1.8). Sometimes, rotary incremental encoders have an index output that provides a pulse at a given angle to synchronize.

- The absolute encoder: They can provide a numeric output (Figure 1.10). They could be mechanic, magnetic (with magneto-resistor or Hall effect sensor), optical, or mechanical sensors. The output signal can be coded in binary or in Grey where two adjacent codes differ by only one bit position, thus reducing coding errors.

It could be also analog (Figure 1.10). A synchro-resolver provides two $\pm 90^\circ$ shifted sinusoidal signals. Speed and position are obtained by demodulation.

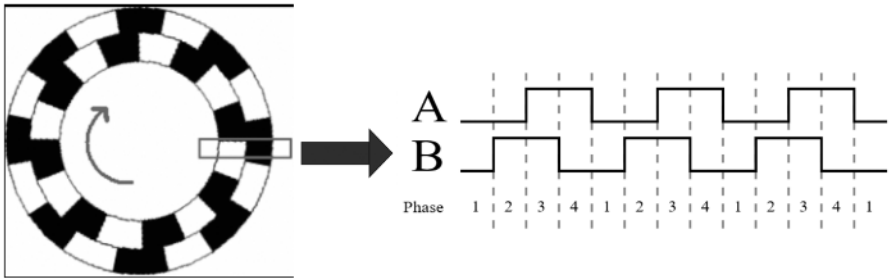


Figure 1.8 Incremental encoder

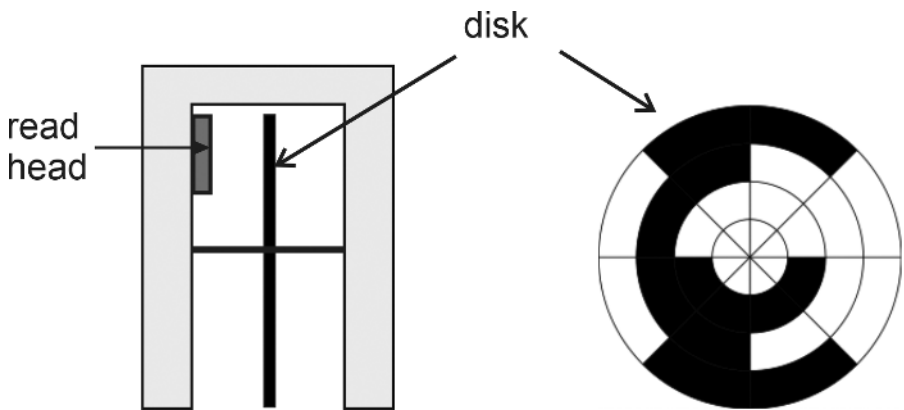


Figure 1.9 Numeric Grey absolute encoder

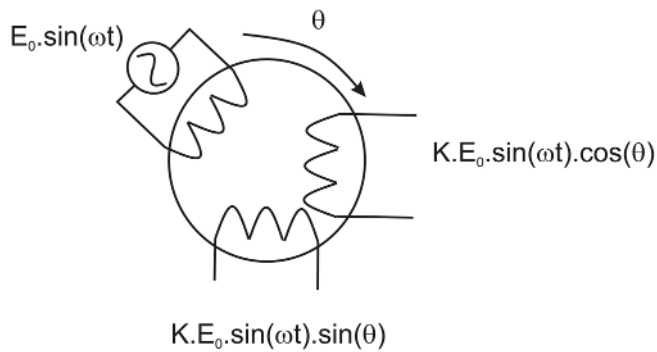


Figure 1.10 Synchro-resolver

Position can be also obtained by virtual sensors such as extended observers (see Subsection 1.2.3.2), or estimated from the e.m.f. of synchronous machines. But in this latter case, the rotor must turn in order to generate an e.m.f. (Zheng et al. [2007] and Comanescu [2012]).

The torque can be used to diagnose electric unbalance, short circuits, and mechanical failures. Torque meters are essentially used for research applications. They can reach bandwidth of a few kHz. However, torque can be computed by extend observer with measured stator currents and estimated flux (Liu et al. [2006] and Wang Chenchen and Li Yongdong [2008]). Torque can be measured from the shear forces experienced on a test body. Their measurement by a stress sensor is transmitted from the rotor to the stator in high frequency by inductive coupling.

1.2.2 Preconditioning

1.2.2.1 Signal Features in the Time Domain

Several techniques are used in order to monitor the state of health of electrical machine, such as the MCSA, which is popular. This technique allows to show up the lines close to the fundamental frequency supplying a three-phase machine. Among all techniques, this one is an online method, which is attractive, because the monitoring of the process doesn't require to interrupt the operating system. To illustrate our meanings, we will present the theoretical components in a spectrum a lot of lines close to the fundamental frequency f_s .

As a consequence, in the case of broken bars for an induction motor, the components frequencies close to f_s are:

$$f_{brb} = \left[\frac{k}{p}(1-s) \pm s \right] f_s \quad (1.1)$$

where $k/p = 1, 5, 7, 11, \dots$

Knowing the fault frequencies in the current spectrum and their evolution in time, which depend on the fault severity, we can easily estimate the state of health of the three-phase induction machine.

1.2.2.2 Symmetric Component, Park Component

Any three-phase voltage or current can be written as the sum of a forward sequence with a backward and a homopolar sequence. So, the forward sequence is written as follows:

$$[V_F(t)] = \begin{cases} v_{a_F}(t) &= V_F \sqrt{2} \cos(\omega_e t) \\ v_{b_F}(t) &= V_F \sqrt{2} \cos(\omega_e t - \frac{2}{3}\pi) \\ v_{c_F}(t) &= V_F \sqrt{2} \cos(\omega_e t - \frac{4}{3}\pi) \end{cases} \quad (1.2)$$

where ω_e is the electrical pulsation, and V_F is the rms value of the voltage for the forward sequence. For the backward sequence, voltages can be written as follows:

$$[V_B(t)] = \begin{cases} v_{a_B}(t) &= V_B \sqrt{2} \cos(\omega_e t) \\ v_{b_B}(t) &= V_B \sqrt{2} \cos(\omega_e t + \frac{2}{3}\pi) \\ v_{c_B}(t) &= V_B \sqrt{2} \cos(\omega_e t + \frac{4}{3}\pi) \end{cases} \quad (1.3)$$

And for the homopolar sequence, one can write as follows:

$$[V_O(t)] = \begin{cases} v_{a_O}(t) &= V_O \sqrt{2} \cos(\omega_e t) \\ v_{b_O}(t) &= V_O \sqrt{2} \cos(\omega_e t) \\ v_{c_O}(t) &= V_O \sqrt{2} \cos(\omega_e t) \end{cases} \quad (1.4)$$

Let us remember that

$$\begin{cases} \cos p \cdot \cos q &= \frac{1}{2} (\cos(p - q) + \cos(p + q)) \\ \sin p \cdot \sin q &= \frac{1}{2} (\cos(p - q) - \cos(p + q)) \\ \sin p \cdot \cos q &= \frac{1}{2} (\sin(p + q) + \sin(p - q)) \\ \sin p + \sin q &= 2 \sin\left(\frac{p+q}{2}\right) \cdot \cos\left(\frac{p-q}{2}\right) \end{cases} \quad (1.5)$$

Let us introduce the not orthonormal Clarke's transformation. The main idea is to transform a three-phase system into a two-phase system. This transformation is based on the fact that a three-phase system is naturally composed of a system shifted by a multiple of $\frac{2}{3}\pi$. Thus, in conclusion, the matrix transformation is written as follows:

$$[C] = \frac{2}{3} \cdot \begin{bmatrix} \cos 0 & \cos \frac{2}{3}\pi & \cos \frac{4}{3}\pi \\ \sin 0 & \sin \frac{2}{3}\pi & \sin \frac{4}{3}\pi \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} = \frac{2}{3} \cdot \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \quad (1.6)$$

If we apply the Clark's transformation to the forward voltage sequence, we obtain:

$$\begin{aligned}
 V_{\alpha\beta O} = [C].V_F(t) &= \frac{2}{3} \cdot \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \cdot \begin{bmatrix} V_F\sqrt{2}\cos(\omega_e t) \\ V_F\sqrt{2}\cos(\omega_e t - \frac{2}{3}\pi) \\ V_F\sqrt{2}\cos(\omega_e t - \frac{4}{3}\pi) \end{bmatrix} \\
 &= \begin{bmatrix} V_F\sqrt{2}\cos(\omega_e t) \\ V_F\sqrt{2}\sin(\omega_e t) \\ 0 \end{bmatrix} \quad (1.7)
 \end{aligned}$$

If we apply now the Clark's transformation to the backward voltage sequence, we obtain:

$$\begin{aligned}
 V_{\alpha\beta O} = [C].V_B(t) &= \frac{2}{3} \cdot \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \cdot \begin{bmatrix} V_B\sqrt{2}\cos(\omega_e t) \\ V_B\sqrt{2}\cos(\omega_e t + \frac{2}{3}\pi) \\ V_B\sqrt{2}\cos(\omega_e t + \frac{4}{3}\pi) \end{bmatrix} \\
 &= \begin{bmatrix} V_B\sqrt{2}\cos(\omega_e t) \\ -V_B\sqrt{2}\sin(\omega_e t) \\ 0 \end{bmatrix} \quad (1.8)
 \end{aligned}$$

In conclusion, any three-phase system or any unbalanced three-phase system can be expressed as the sum a forward sequence with a backward sequence.

1.2.2.3 Symmetric Component, Park Component

As a three-phase system is composed of a two-phase system, one can express the Park's vector. One can notice that it is an abuse of language because the transformation don't use a rotating frame. Consequently, the expression of the Park's vector components (i_α and i_β) using the Concordia's transformation, which is a normalized orthonormal matrix, in a reference frame related to the stator is as follows:

$$i_{\alpha}(t) = \sqrt{\frac{2}{3}} \left(i_A(t) - \frac{1}{2}i_B(t) - \frac{1}{2}i_C(t) \right) \quad (1.9)$$

$$i_{\beta}(t) = \frac{1}{\sqrt{2}} (i_B(t) - i_C(t)), \quad (1.10)$$

where $i_A(t)$, $i_B(t)$, and $i_C(t)$ represent the currents line of the induction motor.

$$[T] = \sqrt{\frac{2}{3}} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \quad (1.11)$$

Under normal conditions, the motor currents are a forward-sequence system. Therefore, the Park's vector can be written as follows:

$$i_{\alpha}(t) = \sqrt{3}I_f \sin \omega_e t \quad (1.12)$$

$$i_{\beta}(t) = \sqrt{3}I_f \cos \omega_e t \quad (1.13)$$

where

I_f is the rms value of the current forward sequence,
 ω_e is the angular supply frequency ($2\pi f_s$).

The Park's current vector can be written in a complex form as follows:

$$\vec{i}_s(t) = i_{\alpha}(t) + j i_{\beta}(t) = \sqrt{3}I_f e^{j\omega_e t} \quad (1.14)$$

where $j^2 = -1$.

Consequently, using the Park's extended vector, the current for the forward sequence becomes:

$$\begin{aligned} \vec{i}_{sf}(t) = & \sqrt{3} \left[I_f e^{j[\omega_e t - \varphi_f]} \right. \\ & + I_{lf} e^{j[(1-2s)\omega_e t - \varphi_{lf}]} \\ & \left. + I_{rf} e^{j[(1+2s)\omega_e t - \varphi_{rf}]} \right] \end{aligned} \quad (1.15)$$

And by the same way, we can deduce the current for the backward sequence:

$$\begin{aligned}\vec{i}_{sb}(t) = & \sqrt{3} [I_b e^{-j[\omega_e t - \varphi_b]} \\ & + I_{lb} e^{-j[(1-2s)\omega_e t - \varphi_{lb}]} \\ & + I_{rb} e^{-j[(1+2s)\omega_e t - \varphi_{rb}]}]\end{aligned}\quad (1.16)$$

The main advantage of the use of Park's extended vector is to analyze and quantify the contribution of a forward and backward power supply on the stator line current. Moreover, the estimation of all line amplitudes in the spectrum allows to better deduce the quality of the power supply and the state of health of the induction machine.

1.2.2.4 Signal Features in the Frequency Domain

The Fourier transform (Poularikas [2000]) $F\{s(t)\}$ for an aperiodic signal $s(t)$ is given by:

$$S(\omega) = F\{s(t)\} = \int_{-\infty}^{\infty} s(t) e^{-j\omega t} dt \quad (1.17)$$

Thus, for a classical signal as a cosinus signal $(\cos \omega_o t)$, the spectral components are:

$$S(\omega) = [\delta(\omega - \omega_o) + \delta(\omega + \omega_o)] \quad (1.18)$$

where δ is the Dirac function. But for the complex signal as the function $e^{j\omega_o t}$, the spectral component is:

$$S(\omega) = [\delta(\omega - \omega_o)] \quad (1.19)$$

Consequently, this approach is recommended for the analysis of the signal in the forward sequence (1.15) as the backward sequence (1.16). All components will appear in the spectral analysis.

(1) Fourier transform and windowing

These methods require the availability of all samples in order to evaluate the spectrum. Moreover, they need a number N of samples equal to a power of two. Consequently, the spectral resolution which depends on $N = 2^r$, is given by the relation $\Delta f = Fe/N$, where Fe is the sampling frequency. The acquisition time of the signal is consequently equal to $T = 1/\Delta f$.

Considering the measured signal $x(p)$ at the sampling time $t = p \cdot Te$, which is characterized by a finite duration sequence of length N and knowing all the previous values $x((p - n)Te)$ ($n = 0, \dots, N - 1$) and starting with the discrete

The current $\overrightarrow{i_s(t)}$: one broken bar & full load

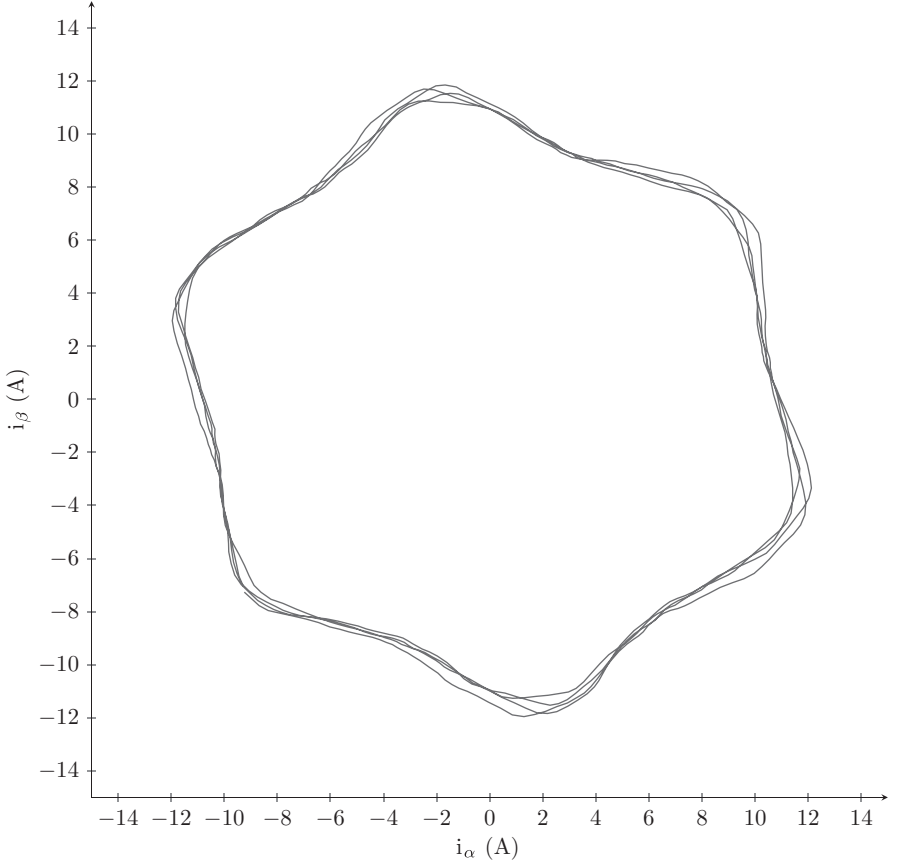


Figure 1.11 The trajectory of the line current: the motor operates under one broken bar and at full load

Fourier transform (DFT) at the time $t = p.T_e$, denoted $F_p[k]$, we can evaluate the spectrum thanks to the relation:

$$F_p[k] = \frac{1}{N} \sum_{n=0}^{N-1} x(p-n)W_N^{nk};$$

where $W_N^{nk} = e^{-j2\pi nk/N}$ with $n, k = 0, 1, 2, \dots, N-1$. If the signal is complex, then:

$$F_p[k] = \frac{1}{N} \sum_{n=0}^{N-1} [\Re\{x(p-n)\} + j \Im\{x(p-n)\}] W_N^{nk}$$

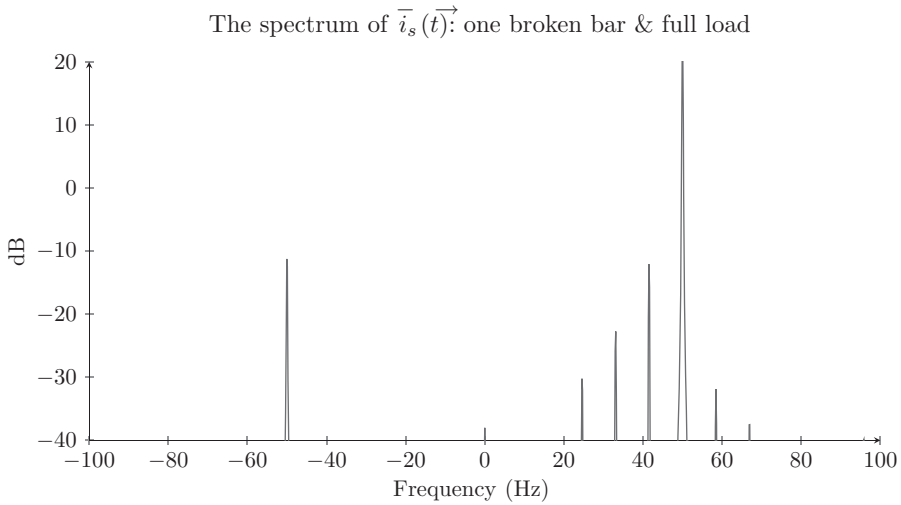


Figure 1.12 The spectrum of the line current: the motor operates under one broken bar and at full load

where $\Re$ and $\Im$ denote the real and the imaginary parts, respectively.

The Hann window: $w(n) = 0.5 \left(1 - \cos\left(2\pi \frac{n}{N}\right) \right)$ with $0 \leq n \leq N$.

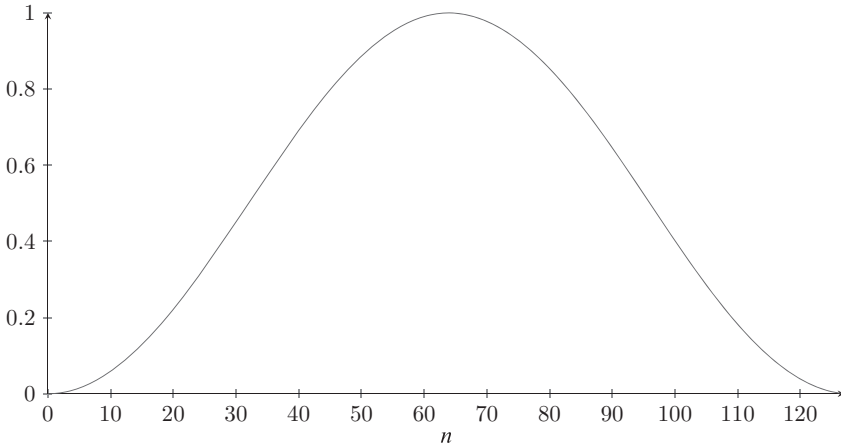


Figure 1.13 The Hann window

The Hamming window: $w(n) = 0.54 - 0.46 \cos\left(2\pi \frac{n}{N}\right)$ with $0 \leq n \leq N$.

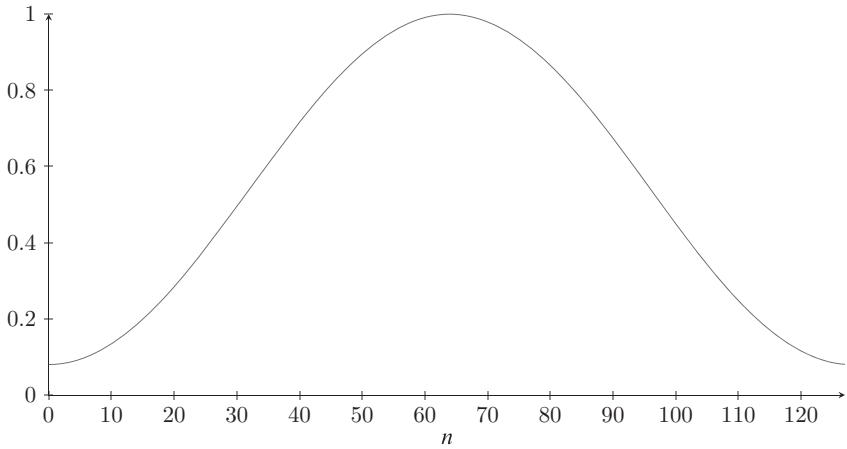


Figure 1.14 The Hamming window

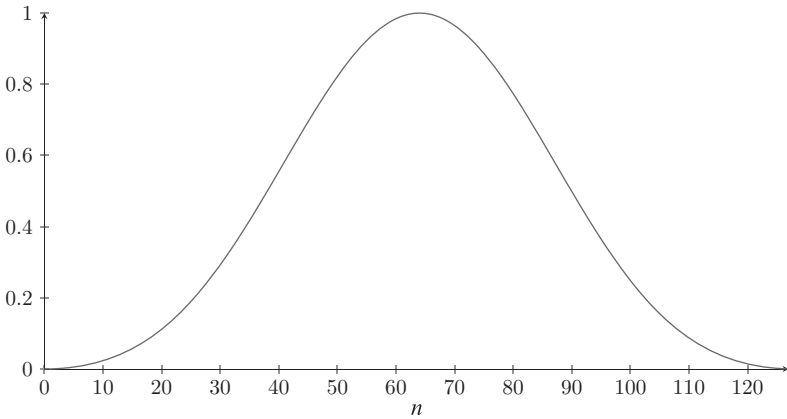


Figure 1.15 The Blackman window

The Blackman window: $w(n) = 0.42 - 0.5 \cos\left(2\pi \frac{n}{N}\right) + 0.08 \cos\left(4\pi \frac{n}{N}\right)$ with $0 \leq n \leq N$.

The impact of windows is very important in the estimation of a spectrum. In fact, their use modifies a lot the estimation of the spectrum. We highlight these meanings by the illustration of a real case of the stator current of an induction motor. To do that, the sampling frequency is 10kHz and the number of samples is 2^{18} . The acquisition time is 26.2144s. We firstly represent the spectrum using:

- a rectangular window;
- a Hamming window;

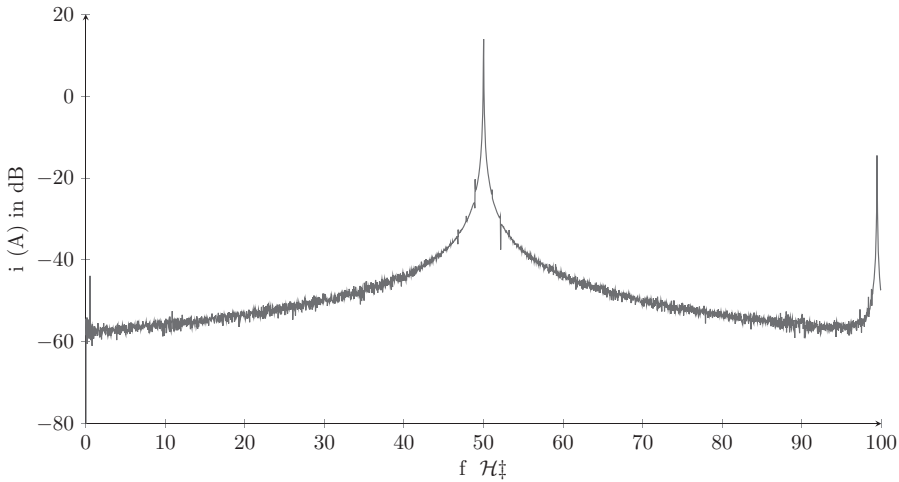


Figure 1.16 Spectrum without window

- a Hann window;
- a Blackman window.

We have to notice that the resolution of the acquisition board is on a 14 bits. So, discretization will produce noise. Moreover, we represent the amplitude of the stator current in A using a logarithmic scale, from $+20\text{dB}$ to -80dB in order to show the impact of the use of a window function on the reduction of the noise in the spectrum.

As a conclusion, we must use a window function to better distinguish lines to each other in the spectrum. To do that, we can see that a window allows to better show the lines but affects the amplitude of each one.

(2) Fast Fourier transform (recursive Fourier transform)

Let's now address the recursivity aspect of the Fourier transform. The classical procedure consists in evaluating the spectral content at the instant pT_e . As a consequence, the spectrum is given by:

$$F_p[k] = \frac{1}{N} \sum_{n=0}^{N-1} x(p-n)W_N^{nk} \quad (1.20)$$

Now, we are going to use an artifact, which consists in adding and subtracting a term that will have the advantage of making an equation with recursivity appearing progressively.

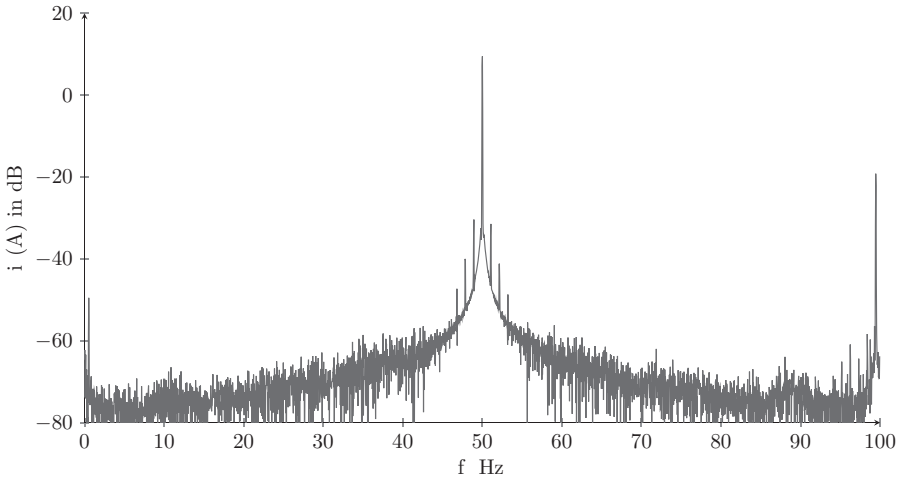


Figure 1.17 Spectrum using the Hamming window

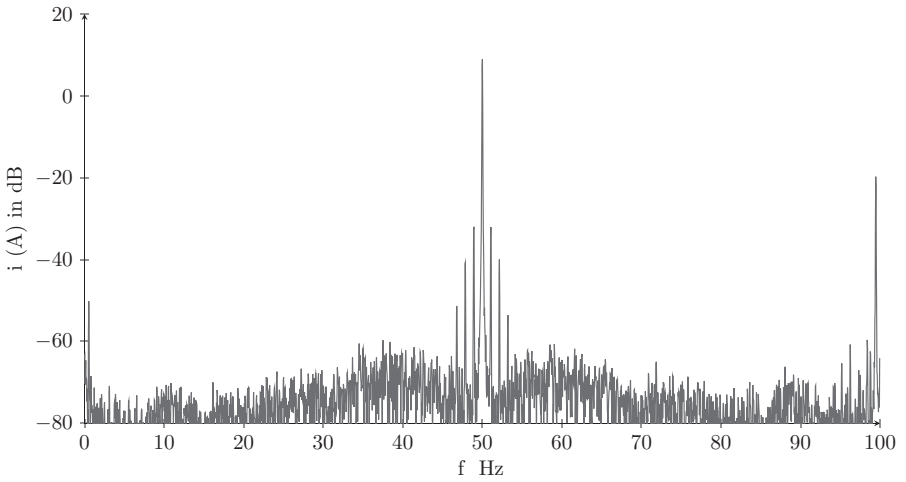


Figure 1.18 Spectrum using the Hann window

$$F_p[k] = \frac{1}{N}x(p-0) + \frac{1}{N} \sum_{n=1}^{N-1} x(p-n)W_N^{nk} \pm \frac{1}{N}x(p-N)W_N^{Nk} \quad (1.21)$$

This can be rewritten as:

$$F_p[k] = \frac{1}{N}x(p) + \frac{1}{N} \sum_{n=1}^N x(p-n)W_N^{nk} - \frac{1}{N}x(p-N) \quad (1.22)$$

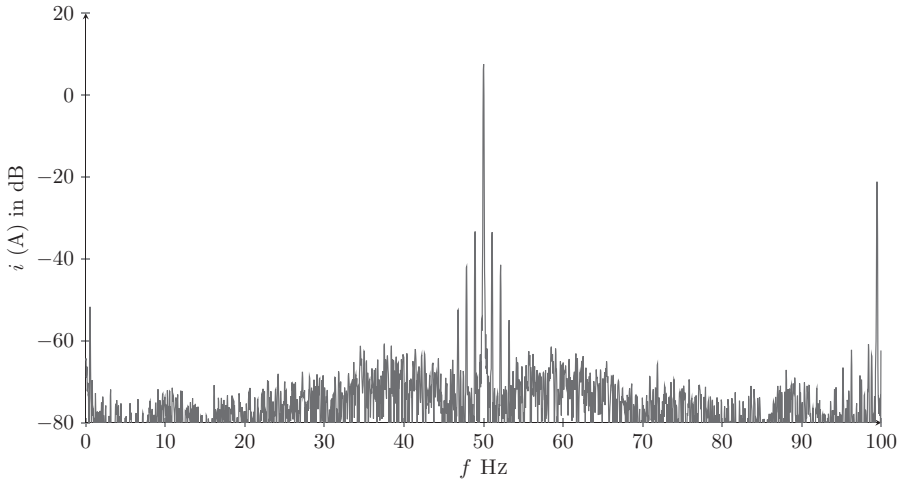


Figure 1.19 Spectrum using the Blackman window

$$F_p[k] = \frac{1}{N}x(p) + W_N^k \cdot \frac{1}{N} \sum_{n=0}^{N-1} x(p-1-n)W_N^{nk} - \frac{1}{N}x(p-N) \quad (1.23)$$

$$F_p[k] = \frac{1}{N}x(p) - \frac{1}{N}x(p-N) + W_N^k \frac{1}{N} \sum_{n=0}^{N-1} x(p-1-n)W_N^{nk} \quad (1.24)$$

As at the instant $(p-1)T_e$, the spectrum is given by:

$$F_{p-1}[k] = \frac{1}{N} \sum_{n=0}^{N-1} x(p-1-n)W_N^{nk} \quad (1.25)$$

The recursive equation at the instant pT_e is:

$$F_p[k] = \left(\frac{x(p) - x(p-N)}{N} \right) + W_N^k F_{p-1}[k] \quad (1.26)$$

A great advantage of the sliding discrete Fourier transform (SDFT) is the possibility to make the spectral estimation only on an interesting bandwidth like around the supply frequency for the monitoring of the induction motor.

$$x(p) = i_\alpha(p) + j i_\beta(p) \quad (1.27)$$

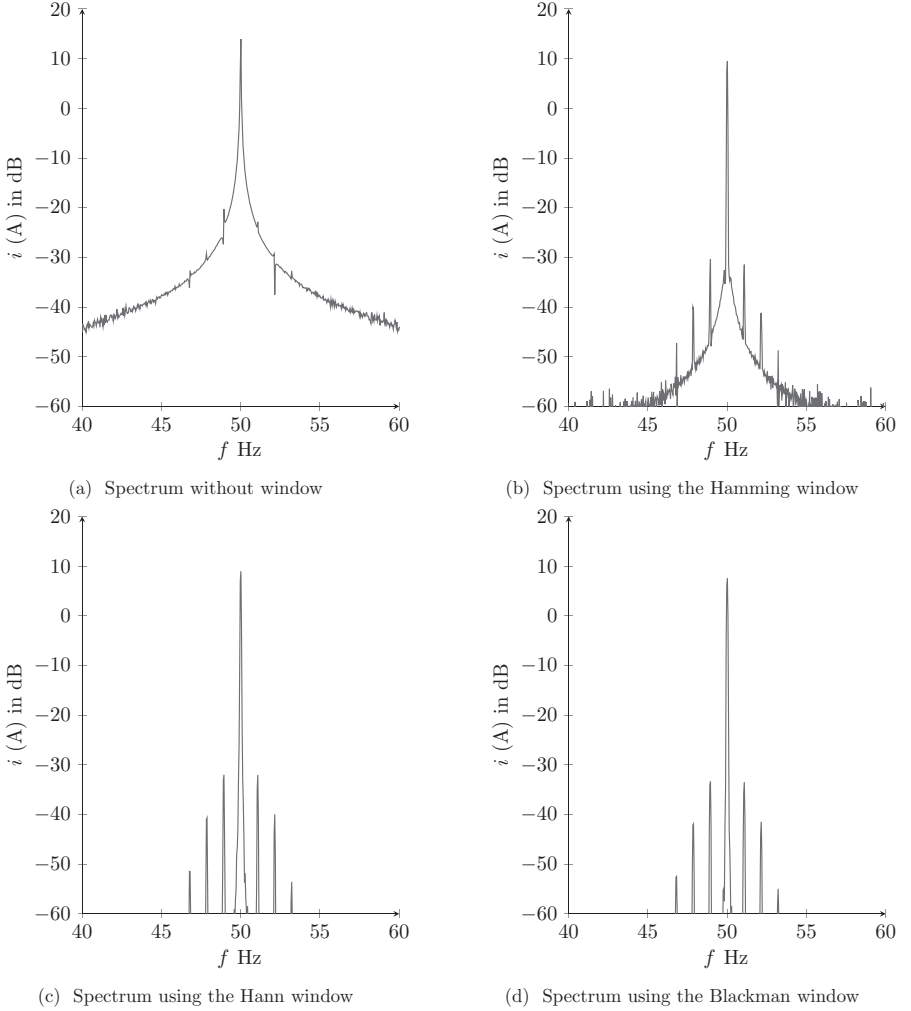


Figure 1.20 Spectra on the range 40–60Hz

The Discrete Fourier transform (DFT) can be written in the form as:

$$F_p[k] = \frac{1}{N} \sum_{n=0}^{N-1} [i_\alpha(p-n) + j i_\beta(p-n)] W_N^{nk} \quad (1.28)$$

This expression is equal to:

$$F_p[k] = \frac{1}{N} \sum_{n=0}^{N-1} i_\alpha(p-n) W_N^{nk} + j \frac{1}{N} \sum_{n=0}^{N-1} i_\beta(p-n) W_N^{nk} \quad (1.29)$$

Consequently, if we denote the DFT of the current i_α by F_{i_α} and by the same way, the current i_β by F_{i_β} , the relation can be rewritten as:

$$F_p[k] = F_{i_\alpha p}[k] + j F_{i_\beta p}[k] \quad (1.30)$$

As the signals i_α and i_β are real, each Fourier transform has the propriety of symmetry. Thus, we can deduce the spectral density of each of them only by using the simple relations:

$$\begin{cases} F_{i_\alpha p}[k] = \frac{1}{2} (F_p[k] + F_p[k]^*) \\ F_{i_\beta p}[k] = \frac{j}{2} (F_p[k]^* - F_p[k]) \end{cases} \quad (1.31)$$

where the asterisk denotes the complex conjugate part.

In conclusion, there is a possibility to estimate in real-time the spectral content of a signal thanks to the recursivity form. The advantage of this approach is that it does not require a lot of numerical calculations but has the disadvantage of not using a weighting window. The spectral estimation is done continuously but doesn't forget that the conditions must be such that the signal is stable; otherwise the estimation will be altered.

1.2.2.5 Wavelet Analysis

As far as we know, the concept of wavelet took its origin in the beginning of the previous century. Despite this, applications in signal analyses appear close to 1980. Let us be reminded that the Wavelet transform of a function $f(t)$ is a decomposition of it in a set of functions $\psi_{a,\tau}(t)$ named wavelets functions. A Wavelet transform is defined as follows:

$$W_\omega(a, \tau) = \int_{-\infty}^{+\infty} f(t) \psi_{a,\tau}^*(t) dt \quad (1.32)$$

where a is the scale factor and τ is the translation one and with $\psi_{a,\tau}(t)$ is the basic mother wavelet:

$$\psi_{a,\tau}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-\tau}{a}\right) \quad (1.33)$$

The analysis using the orthogonal wavelet allows to obtain a more compact representation of the signal. It permits the reconstitution of the original signal from its transformed one. Nevertheless, the analysis of a signal with a non-orthogonal wavelet is recommended for the continuous amplitude variation of the wavelet coefficients (Torrence and Compo [1998]).

Among all non-orthogonal wavelet families, we have: Gaussian, Morlet, and Mexican Hat. The most common functions used are the Morlet Hat and the

Mexican Hat. The expression of Wavelet Mother of Morlet function is described as follows:

$$\psi(t) = e^{jKt} e^{-\frac{t^2}{2}} \quad (1.34)$$

where the number of oscillation is depending on the coefficient K .

Although the Morlet function is non-orthogonal and complex, it allows to obtain information on the amplitude and the phase of signals. Thus, it is certainly better adapted for oscillatory systems. The Mexican Hat Mother Wavelet is a real function described in Equation (1.35):

$$\psi(t) = (1 - t^2) e^{-\frac{t^2}{2}} \quad (1.35)$$

This real function is particularly well adapted and so recommended for the observation of discontinuities into a signal.

However, the choice of the scale factor is not easy and a lot of work proposes criteria for the determination of this scale factor. It is possible to estimate the values minima and maxima of the scale factor to be used (see Torrence and Compo [1998]), or to determine the optimum value of scale factor (see Saleh and Rahman [2005]).

In our illustration, knowing the number of samples (N) of the signal to analyze, using the Wavelet Mexican Hat, and knowing that N is a power of two, the scale factor to be used is given by:

$$a = \log_2 N \quad (1.36)$$

where the minimal and maximal value are given by:

$$a_{\min} = \frac{1}{2}a \quad \text{and} \quad a_{\max} = \frac{3}{2}a \quad (1.37)$$

1.2.2.6 Instantaneous Amplitude and Frequency

(1) *Instantaneous amplitude*

To calculate the instantaneous amplitude of a real signal $x(t)$, it is possible to use the analytical signal $z(t)$ associated with $z(t)$:

$$z(t) = x(t) + j y(t) = a(t) * e^{j\varphi(t)} \quad (1.38)$$

Where $y(t)$ is the Hilbert transform of $x(t)$, defined by:

$$H[x(t)] = y(t) = \frac{1}{\pi} P \int_{-\infty}^{+\infty} \frac{x'(t')}{t - t'} dt' \quad (1.39)$$

where P is the principal value of Cauchy.

In 1.41, $a(t)$ corresponds to the instantaneous amplitude of $z(t)$:

$$a(t) = \sqrt{x^2(t) + y^2(t)} \quad (1.40)$$

(2) Instantaneous frequency

Instantaneous frequency is an important characteristic of the signal that is present in many areas. It is a concept related to time–frequency analysis, where it can be obtained from a time–frequency distribution (TFD). The frequency of a signal is usually a quantity that is defined independently of time. However, in the case of nonstationary signals, the introduction of the notion of instantaneous frequency can be useful for the analysis. Physically, it is important to note that the analyzed signal must imperatively be a signal with a single spectral component. In the case of multicomponent signals, the concept of instantaneous frequency is no longer valid (Loughlin and Tacer [1997]). Two basic definitions are proposed in the literature for calculating instantaneous frequency. The first one is based on the time–frequency representation of the studied signal, and the second one is defined as the derivative of the phase of the analytical signal associated with the studied signal. The instantaneous frequency $f_i(t)$ of an analytical signal $z(t) = a(t) * e^{j\varphi(t)}$ containing a single component is defined as being the derivative of its phase:

$$f_i(t) = \frac{1}{2\pi} \frac{d\varphi(t)}{dt} \quad (1.41)$$

The analytical signal $z(t)$ is a complex signal containing the same information as the real signal $x(t)$. Unlike the real signal, the complex signal contains only the positive frequencies. It is built from the real signal using the Hilbert transform symbolized by the operator H :

$$z(t) = x(t) + jH[x(t)] \quad (1.42)$$

$$H(t) = \int_{-\infty}^{+\infty} \frac{x(t - \tau)}{\pi\tau} d\tau \quad (1.43)$$

1.2.2.7 Bilinear Time–frequency Distributions or Quadratic Time–frequency Distributions: Cohen’s Class

Time analysis does not allow to represent precisely the harmonic components of the signal, while the frequency analysis does not allow them to be located temporally. Simultaneous representation in the time and frequency domain should achieve a compromise between the two previous approaches (Lebaroud [2007], Chatfield and Xing [2019], and Flandrin [1998]). However,

the Heisenberg uncertainty principle excludes any precise temporal location of a frequency. This ambiguity has led to define a wide variety of time–frequency transformations.

1.2.2.7.a Uncertainty Principle of Heisenberg

For any signal $x(t)$ of finite energy, let us define the centroids in the time and frequency domains (eq. 1.44 and eq. 1.45):

$$\langle t \rangle = \frac{1}{E} \int t |x(t)|^2 dt \quad (1.44)$$

and

$$\langle f \rangle = \frac{1}{E} \int f |X(f)|^2 df \quad (1.45)$$

The time and frequency resolutions can be deduced from these equations (eq. 1.46 and eq. 1.47):

$$\Delta t = \sqrt{\frac{1}{E} \int (t - \langle t \rangle)^2 |x(t)|^2 dt} \quad (1.46)$$

and

$$\Delta f = \sqrt{\frac{1}{E} \int (f - \langle f \rangle)^2 |x(f)|^2 df} \quad (1.47)$$

Thus, we can write the Heisenberg's uncertainty principle as eq. 1.48:

$$\Delta t \cdot \Delta f \geq \frac{1}{4\pi} \quad (1.48)$$

1.2.2.7.b General Representation

The time–frequency representations of Cohen's class (Cohen [1995]) are of the form (eq. 1.49):

$$C_x(t, f) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \varphi_{t-r}(t - \nu, \tau) x(\gamma + \frac{\tau}{2}) x^*(\gamma - \frac{\tau}{2}) e^{-j2\pi f \tau} d\gamma d\tau \quad (1.49)$$

where $\varphi_{t-r}(t - \nu, \tau)$ is the *kernel*.

Let us define the ambiguity function (eq. 1.50):

$$A_x(\tau, \xi) = \int_{-\infty}^{+\infty} x(t + \frac{\tau}{2}) x^*(t - \frac{\tau}{2}) e^{-j2\pi \xi t} dt \quad (1.50)$$

We can find two different forms of the previous equations:

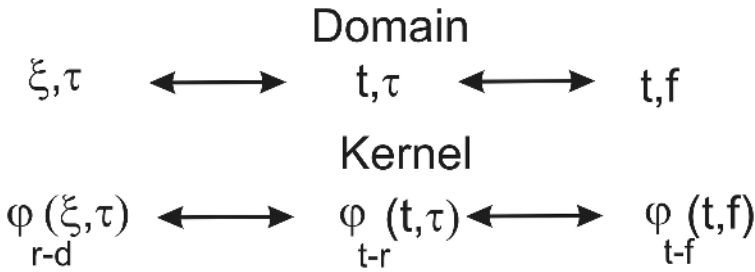


Figure 1.21 Rules of transformation between the time–frequency and Doppler–delay domains encoder

First form:

$$C_x(t, f) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \varphi_{r-d}(\tau, \xi) A_x(\tau, \xi) e^{j2\pi(t\xi - f\tau)} d\xi d\tau \quad (1.51)$$

where $A_x(\tau, \xi) = \int_{-\infty}^{+\infty} x(t + \frac{\tau}{2}) x^*(t - \frac{\tau}{2}) e^{-j2\pi\xi t} dt$ and $\varphi_{r-d}(\tau, \xi) = \int_{-\infty}^{\infty} \varphi_{t,\tau}(t, \tau) e^{-j2\pi\xi t} dt$

The two variables τ and ξ are called delay and Doppler frequency, respectively.

Second form:

$$C_x(t, f) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \varphi_{t-f}(t - v, f - \nu) W_x(v, \nu) dv d\nu \quad (1.52)$$

where $W_x(t, f) = \int_{-\infty}^{+\infty} x(t + \frac{\tau}{2}) x^*(t - \frac{\tau}{2}) e^{-j2\pi f \tau} d\tau$ and $\varphi_{t-f}(t, f) = \int_{-\infty}^{\infty} \varphi_{t,\tau}(t, \tau) e^{-j2\pi f \tau} d\tau$. The function $W_x(t, f)$ is the distribution of Wigner–Ville (Wigner [1932] and Ville [1948]).

The rules of transformation between the time–frequency and Doppler–delay domains are illustrated in Figure 1.21, which explains the type of *Kernel* to be associated with the type of variables. All these *Kernels* are detailed in the Tables 1.16 to 1.19.

1.2.2.7.c Properties

There is a wide variety of time–frequency representations based on the Cohen’s class principle. They are distinguished by their properties. The research work carried out in this field (Hlawatsch and Boudreaux–Bartels [1992] and TA and Mecklenbräuker [1980]) has allowed associate properties and the different *kernels* of the time–frequency representations (TFR).

Let us consider the main properties in Table 1.12.

Table 1.12 First group of properties of the representations of the Cohen's class (Lebaroud [2007], Hlawatsch and Boudreaux-Bartels [1992], and Auger and Doncarli [1992]).

N	Property	Mathematical formulation	Condition
0	Invariance with respect to time and frequency translations	$\forall t_1, f_1, y(t) = x(t - t_1) e^{j2\pi f_1 t}$ $\Rightarrow C_y(t, f) = C_x(t - t_1, f - f_1)$	$\varphi_{r-d}(\tau, \xi)$ whatever is
1	Invariance with respect to changes in scale	$\forall a > 0, y(t) = a^{-1/2} x(t/a)$ $\Rightarrow C_y(t, f) = C_x(t/a, af)$	$\forall \tau, \xi, \forall a > 0,$ $\varphi_{r-d}(\tau, \xi) = \varphi_{r-d}(\tau/a, a\xi)$
2	Invariance with respect to time inversion	$y(t) = x(-t)$ $\Rightarrow C_y(t, f) = C_x(-t, -f)$	$\forall \tau, \xi$ $\varphi_{r-d}(\tau, \xi) = \varphi_{r-d}(-\tau, -\xi)$
3	Invariance with respect to complex conjugation	$y(t) = x^*(t)$ $\Rightarrow C_y(t, f) = C_x(t, -f)$	$\forall \tau, \xi$ $\varphi_{r-d}(\tau, \xi) = \varphi_{r-d}(-\tau, \xi)$
4	Real character of TFR	$\forall t, f,$ $C_x(t, f) = C_x^*(t, f)$	$\forall \tau, \xi$ $\varphi_{r-d}^*(\tau, \xi) = \varphi_{r-d}(-\tau, -\xi)$
5	Causality	$C_x(t, f)$ only depends on $x(u)$ if $u < t$	$\forall t, \tau,$ $t < \tau /2 \Rightarrow \varphi_{t-r}(t, \tau) = 0$

where $\hat{x}(f) = \int_{-\infty}^{+\infty} x(t) e^{-j2\pi f t} dt$ is the Fourier transform of $x(t)$.

1.2.2.7.d Different Representations

Tables 1.16 to 1.19 give the different Cohen's class representations with the properties defined in the paragraph 1.2.2.7.c.

The Wigner-Ville representation is the only distribution that provides perfect localization of linearly frequency-modulated signals. But it has many interference terms, which makes it difficult to interpret.

The pseudo Wigner-Ville representation uses a window $h(\tau)$ that makes an easier computation.

The smoothed pseudo Wigner-Ville representation uses an additional window $g(t)$, in order to perform a time domain smoothing independent of the frequency smoothing ($\varphi_{t,f}(t, f) = g(t) \hat{h}(f)$). This allows to have a good compromise between the localization of the different components and the attenuation of interference terms (Lebaroud [2007]) (Table 1.19).

Rihaczek Distribution provides a perfect localization of sinusoidal signals. The distribution of Margenau-Hill is easier to represent graphically. The Page's distribution is the only Cohen's class that simultaneously verifies the properties of causality (property 5), spectrum conservation (property

Table 1.13 Second group of properties of the representations of the Cohen's class (Lebaroud [2007], Hlawatsch and Boudreaux-Bartels [1992], and Auger and Doncarli [1992])

N	Property	Mathematical formulation	Condition
6	Energy conservation	$\int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} C_x(t, f) dt df = \int_{-\infty}^{+\infty} x(t) ^2 dt$	$\varphi_{r-d}(0, 0) = 1$
7	Instantaneous power conservation	$\forall t, \int_{-\infty}^{\infty} C_x(t, f) df = x(t) ^2$	$\forall \xi, \varphi_{r-d}(0, \xi) = 1$
8	Spectrum conservation	$\forall f, \int_{-\infty}^{\infty} C_x(t, f) dt = \hat{x}(f) ^2$	$\forall \tau, \varphi_{r-d}(\tau, 0) = 1$
9	Instantaneous frequency conservation	$\forall t, \int_{-\infty}^{\infty} f C_x(t, f) df = f_x(t) x(t) ^2$ $f_x(t) = \frac{1}{2\pi} \frac{d}{dt} \arg(x(t))$ [mm]	$\forall \xi, \varphi_{r-d}(0, \xi) = 1$ and $\frac{\partial}{\partial \tau} \varphi_{r-d}(0, \xi) = 0$
10	Group delay conservation	$\forall f, \int_{-\infty}^{\infty} t C_x(t, f) dt = t_x(f) \hat{x}(f) ^2$ $t_x(f) = -\frac{1}{2\pi} \frac{d}{df} \arg(\hat{x}(f))$	$\forall \tau, \varphi_{r-d}(\tau, 0) = 1$ and $\frac{\partial}{\partial \xi} \varphi_{r-d}(\tau, 0) = 0$
11	Positivity	$\forall x(t), \forall t, f, C_x(t, f) \geq 0$	$\varphi_{r-d}(\tau, \xi) = \int_{-\infty}^{\infty} c(\alpha) A_{h_\alpha}(\tau, \xi) d\alpha$ with $\forall \alpha, c(\alpha) \geq 0$

8), Moyal's formula (property 14) and the conservation of the signal domain (property 16) (Lebaroud [2007]) (Table 1.17).

The spectrogram is still widely used today, especially for analyzing signals that are slowly non-stationary or have a large number of components. For short and strongly non-stationary signals, a better compromise can then be obtained by the S representation (Lebaroud [2007]) (Table 1.18).

The use of product *kernels* reduces interference terms (Lebaroud [2007]) (Table 1.19).

We will illustrate the different Cohen's classes with the help of a chirp function (eq 1.53). It is a signal where the frequency increases or decreases with time.

$$y(t) = \Re(a(t)e^{j\Phi(t)}) \quad (1.53)$$

Table 1.14 Third group of properties of the representations of the Cohen's class (Lebaroud [2007], Hlawatsch and Boudreaux–Bartels [1992], and Auger and Doncarli [1992]).

N	Property	Mathematical formulation	Condition
12	Compatibility with linear Filters	$y(t) = \int_{-\infty}^{\infty} h(u) x(t-u) du$ $\Rightarrow$ $C_y(t, f) = \int_{-\infty}^{\infty} C_h(u, f) C_x(t-u, f) du$	$\forall \tau, \tau', \xi$ $\varphi_{r-d}(\tau, \xi) \varphi_{r-d}(\tau', \xi) = \varphi_{r-d}(\tau + \tau', \xi)$
13	Compatibility with modulations	$y(t) = m(t) x(t)$ $\Rightarrow$ $C_y(t, f) = \int_{-\infty}^{\infty} C_m(t, \nu) C_x(t, f - \nu) d\nu$	$\forall \tau, \xi, \xi',$ $\varphi_{r-d}(\tau, \xi) \varphi_{r-d}(\tau, \xi') = \varphi_{r-d}(\tau, \xi + \xi')$
14	Preservation of scalar product	$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} C_x(t, f) C_y^*(t, f) dt df =$ $\left \int_{-\infty}^{\infty} x(t) y^*(t) dt \right ^2$	$\forall \tau, \xi, \varphi_{r-d}(\tau, \xi) ^2 = 1$
15	Reversibility	$x(t)$ can be retrieved, within one phase, from its TFR	$\forall \tau, \xi,$ $\varphi_{r-d}(\tau, \xi) \neq 0$

Table 1.15 Fourth group of properties of the representations of the Cohen's class (Lebaroud [2007], Hlawatsch and Boudreaux–Bartels [1992], and Auger and Doncarli [1992]).

N	Property	Mathematical formulation	Condition
16	Conservation of the signal domain	If $\forall t, t > T \Rightarrow x(t) = 0$ Then $\forall t, t > T \Rightarrow C_x(t, f) = 0$	$\forall t, \tau,$ $ \tau < 2 t \Rightarrow \varphi_{r-d}(t, \tau) = 0$
17	Conservation of the null values of the signal	$x(t_0) = 0 \Rightarrow \forall f, C_x(t_0, f) = 0$	$\varphi_{r-d}(\tau, \xi) =$ $h_1(\tau) e^{-j2\pi\xi\tau/2} + h_2(\tau) e^{j2\pi\xi\tau/2}$
18	Conservation of the null values of the spectrum	$\hat{x}(f_0) = 0 \Rightarrow \forall t, C_x(t, f_0) = 0$	$\varphi_{r-d}(\tau, \xi) =$ $G_1(\xi) e^{-j2\pi\xi\tau/2} + G_2(\xi) e^{j2\pi\xi\tau/2}$

Table 1.16 Cohen’s class representations: Wigner–Ville Distribution and associated forms (Lebaroud [2007], Hlawatsch and Boudreaux–Bartels [1992], and Auger and Doncarli [1992]).

Name	Formula	Properties
Wigner Ville	$\int_{-\infty}^{\infty} x(t + \tau/2) x^*(t - \tau/2) e^{-j2\pi f\tau} d\tau$ Kernel: $\varphi_{r-d}(\tau, \xi) = 1$ $\varphi_{t-r}(t, \tau) = \delta(t)$	0,1,2,3,4,6,7,8,9, 10,12,13,14,15,16
Pseudo Wigner– Ville	$\int_{-\infty}^{\infty} h(\tau) x(t + \tau/2) x^*(t - \tau/2) e^{-j2\pi f\tau} d\tau$ Kernel: $\varphi_{r-d}(\tau, \xi) = h(\tau)$ $\varphi_{t-r}(t, \tau) = \delta(t) h(\tau)$	0,2,3,4,6,7,9,16 if $h(t)$ is real and even, $h(0) = 1$, $h'(0) = 0$
Smoothed Pseudo Wigner– Ville	$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(t - \nu) h(\tau) x(\nu + \tau/2) x^*(\nu - \tau/2) e^{-j2\pi f\tau} d\tau d\nu$ Kernel: $\varphi_{r-d}(\tau, \xi) = h(\tau) \hat{g}(\xi)$ $\varphi_{t-r}(t, \tau) = g(t) h(\tau)$	0,2,3,4,6 if $h(t)$ is real and even, $h(0)=1$ and if $g(t)$ is real and even, $\hat{g}(0) = 1$

Table 1.17 Cohen class representations: Rihaczek Distribution and associated forms (Lebaroud [2007] and Auger and Doncarli [1992]).

Name	Formula	Properties
Rihaczek	$x(t) \hat{x}^*(f) e^{-j2\pi ft}$ Kernel: $\varphi_{r-d}(\tau, \xi) = e^{-j\pi \xi \tau}$ $\varphi_{t-r}(t, \tau) = \delta(t - \tau/2)$	0,1,2,6,7,8,12,13, 14,15,16,17,18
Margenau–Hill	$\int_{-\infty}^{\infty} \frac{1}{2} [x(t + \tau) x^*(t) + x(t) x^*(t - \tau)] e^{-j2\pi f\tau} d\tau =$ $\text{Re}\{x(t) \hat{x}^*(f) e^{-j2\pi ft}\}$ Kernel: $\varphi_{r-d}(\tau, \xi) = \cos(\pi \xi \tau)$ $\varphi_{t-r}(t, \tau) = \frac{1}{2} [\delta(t - \tau/2)] + \delta(t + \tau/2)$	0,1,2,3,4,6,7,8, 9,10,16,17,18

(Continued)

Table 1.17 (Continued)

Name	Formula	Properties
Pseudo Margenau-Hill	$\int_{-\infty}^{\infty} \frac{h(\tau)}{2} [x(t+\tau)x^*(t) + x(t)x^*(t-\tau)] e^{-j2\pi f\tau} d\tau =$ $\text{Re}\left\{x(t) TFCT_x^{h_s}(t, f) e^{-j2\pi f t}\right\} \text{ (h real and even)}$ Kernel: $\varphi_{r-d}(\tau, \xi) = h(\tau) \cos(\pi \xi \tau)$ $\varphi_{t-r}(t, \tau) = \frac{h(\tau)}{2} [\delta(t - \tau/2)] + \delta(t + \tau/2)$	0,2,3,4,6,7,9,1617, if h(t) is real even, $h(0) = 1, h'(0) = 0$
Page	$2 \text{Re} \left\{ x(t) \left(\int_{-\infty}^t x(u) e^{-j2\pi f u} du \right)^* e^{-j2\pi f t} \right\} \text{ Kernel:}$ $\varphi_{r-d}(\tau, \xi) = e^{j\pi \xi \tau }$ $\varphi_{t-r}(t, \tau) = \delta(t - \tau /2)$	0,3,4,5,6,7,16,17

Table 1.18 Cohen class representations: Spectrogram et S representation (Lebaroud [2007] and Auger and Doncarli [1992])

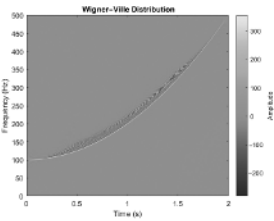
Name	Formula	Properties
Spectrogram	$\left \int_{-\infty}^{\infty} x(u) h^*(u-t) e^{-j2\pi f u} du \right ^2$ Kernel: $\varphi_{r-d}(\tau, \xi) = A_h^*(\tau, \xi)$ $\varphi_{t-r}(t, \tau) = h^*(-t + \tau/2) h(-t - \tau/2)$	0,2,3,4,6,11 if $\int_{-\infty}^{\infty} h(t) ^2 dt = 1$
S- representation	$\int_{-\infty}^{\infty} \hat{S}(\xi) TFCT_x^h(t, f + \xi/2) TFCT_x^{h_s}(t, f - \xi/2) e^{j2\pi \xi t} d\xi$ Kernel: $\varphi_{r-d}(\tau, \xi) = \int_{-\infty}^{\infty} \hat{S}(\nu) A_h^*(\tau, \xi - \nu) d\nu$ $\varphi_{t-r}(t, \tau) = S(t) h^*(-t + \tau/2) h(-t - \tau/2)$	0,2,3,4,6

where the amplitude is slowly varying compared to the variations of Φ . In our example, we will use a quadratic increase. Several distributions are presented in Table 1.20 for the same signal.

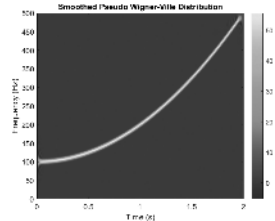
Table 1.19 Cohen class representations: Product-type or cone-shaped core representation (Lebaroud [2007] and Auger and Doncarli [1992]).

Name	Formula	Properties
	$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \frac{\exp\left(-\frac{\nu^2}{4\tau^2/\sigma}\right)}{\sqrt{4\pi\tau^2/\sigma}} x(t-\nu+\tau/2) x^*(t-\nu-\tau/2) e^{-j2\pi f\tau} d\nu d\tau$ Kernel:	
Choi-Williams	$\varphi_{r-d}(\tau, \xi) = \exp(-4\pi^2 \xi^2 \tau^2 / \sigma)$ $\varphi_{t-r}(t, \tau) = \begin{cases} \delta(t) & \text{if } \tau = 0 \\ \frac{\exp\left(-\frac{t^2}{4\tau^2/\sigma}\right)}{\sqrt{4\pi\tau^2/\sigma}} & \text{if } \tau \neq 0 \end{cases}$	0,1,2,3,4,6, 7,8,9,10,15
	$\int_{-\infty}^{\infty} \int_{- \tau /2}^{ \tau /2} \frac{1}{ \tau } h(\tau) x(t-\nu+\tau/2) x^*(t-\nu-\tau/2) e^{-j2\pi f\tau} d\nu d\tau$ Kernel:	
Born-Jordan	$\varphi_{r-d}(\tau, \xi) = \sin c(\pi \xi \tau)$ $\varphi_{t-r}(t, \tau) = \begin{cases} \delta(t) & \text{if } \tau = 0 \\ \frac{1}{ \tau } & \text{if } t < \tau /2 \\ 0 & \text{else} \end{cases}$	0,1,2,3,4,6, 7,8,9,10,16
	$\int_{-\infty}^{\infty} \int_{- \tau /2}^{ \tau /2} h(\tau) x(t-\nu+\tau/2) x^*(t-\nu-\tau/2) e^{-j2\pi f\tau} d\nu d\tau$ Kernel:	
Zhao-Atlas -Marks	$\varphi_{r-d}(\tau, \xi) = h(\tau) \sin c(\pi \xi \tau)$ $\varphi_{t-r}(t, \tau) = \begin{cases} 0 & \text{if } \tau = 0 \\ h(t) & \text{if } t < \tau /2 \\ 0 & \text{else} \end{cases}$	0,2,3,4,16

Table 1.20 TFR examples for a chirp signal



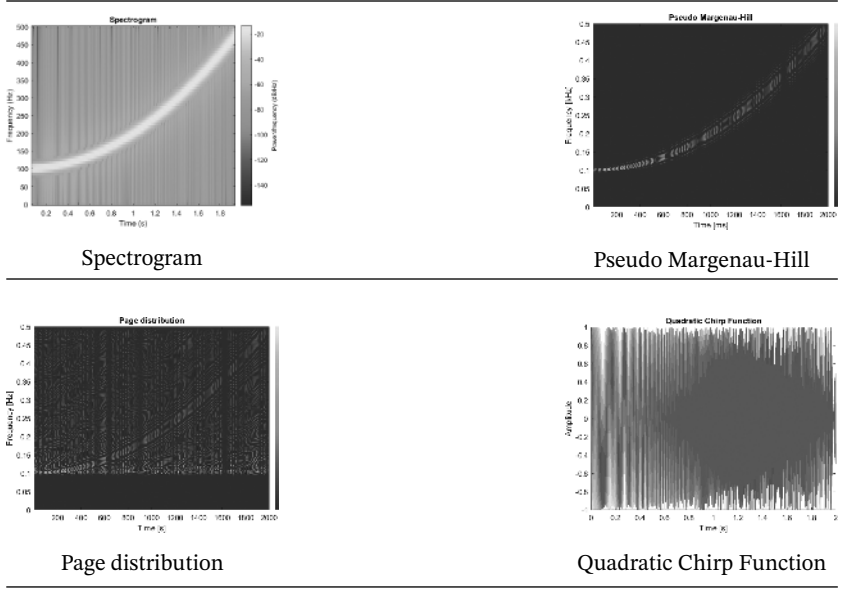
Wigner-Ville Distribution



Smoothed Pseudo Wigner-Ville Distribution

(Continued)

Table 1.20 (Continued)



1.2.2.8 Statistic Features

Eight main statistic features are often used, in particular for vibration studies.

Let us consider a time series x_t of N samples. They are defined as follows:

- The energy

$$E = \sum_{\ell=1}^N x_{\ell}^2 \quad (1.54)$$

- The maximum

$$S_{PA} = \sup_{1 \leq \ell \leq N} |x_{\ell}| \quad (1.55)$$

- The mean value

$$\bar{x} = \frac{1}{N} \sum_{\ell=1}^N x_{\ell} \quad (1.56)$$

- The crest factor

$$F_c = S_{PA}/_{RMS} \quad (1.57)$$

- The mean power

$$P = \frac{1}{N} \sum_{\ell=1}^N x_{\ell}^2 \quad (1.58)$$

- The standard deviation

$$RMS = \sqrt{\frac{1}{N} \sum_{\ell=1}^N (x_{\ell} - \bar{x})^2} \quad (1.59)$$

- The kurtosis

$$s_{kurt} = \frac{\frac{1}{N} \sum_{\ell=1}^N (x_{\ell} - \bar{x})^4}{RMS^4} \quad (1.60)$$

- The factor K

$$F_K = S_{PA} \cdot RMS = F_c \cdot RMS^2 \quad (1.61)$$

RMS value gives a measure of an overall vibration level. It is sensitive to the energy of the signal. The maximum value gives the same kind of information, but it is less adapted than RMS indicator.

Crest factor and kurtosis are suitable for shock measurement. They measure the impulsive nature of a signal. Kurtosis is often better. However, it can have a non-monotonous variation with the severity, in case of several defects.

One must beware of an overly simplistic interpretation of kurtosis (Balanda and Macgillivray [1988]). J. J. A. Moors [1986] explains that two causes involve a high values of the Kurtosis:

- The probability mass is concentrated around the mean and a few scattered values are far from the mean;
- The probability mass is concentrated in the tails of the distribution.

Factor K allows to evaluate the degree of severity of a rolling single defect (Boukar and Hamzaoui [2019]).

1.2.2.9 Cyclostationarity

(1) General description

The concept of cyclostationarity was introduced in the Russian literature by Gudzenko [1959] and Lebedev [1959]. It deals with signals which exhibit some hidden periodicity of their energy flow (see Figure 1.22). In other terms, cyclostationary processes, which result from the interaction of periodic and random phenomena, are characterized by periodic statistics (Antoni [2009]). The utilization of cyclostationarity is especially suitable for diagnosing the

mechanical failures of rotating machines (Bonnardot [2004]), gears (Kidar [2015]), etc.

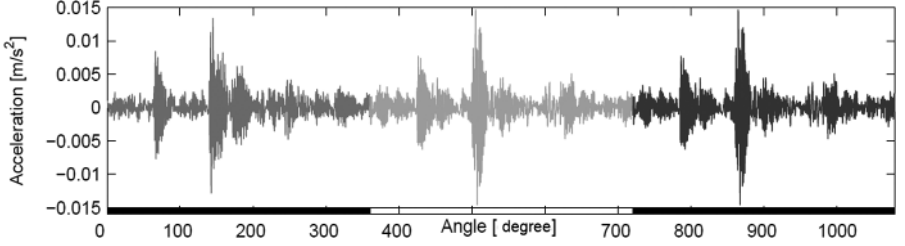


Figure 1.22 Stationary signal example (Bonnardot [2004])

We will now give some definitions of the cyclostationarity for rotating machines.

(2) n^{th} -order strict cyclostationarity

Let us consider a rotating machine signal $X(\theta)$ with a stochastic behavior. For any θ , $X(\theta)$ is a random variable.

The n^{th} -order distribution function is defined by:

$$\begin{aligned} F_{X(\theta + \theta_1) \dots X(\theta + \theta_{N-1}) X(\theta)}(\delta_1 \dots \delta_{N-1} \delta_N) \\ = P(X(\theta + \theta_1) \leq \delta_1 \dots X(\theta + \theta_{N-1}) \leq \delta_{N-1} X(\theta) \leq \delta_N) \end{aligned} \quad (1.62)$$

The process $X(\theta)$ has a strictly n^{th} -order cyclostationary of period θ if and only if its n^{th} -order distribution function is periodic of period θ .

$$\begin{aligned} \forall \left[\delta_1, \dots, \delta_N \right] \in R^N, \forall \left[\theta_1, \dots, \theta_N \right] \in R^N, \forall \theta \in R, \\ F_{X(\theta + \theta_1), \dots, X(\theta + \theta_{N-1}), \theta}(\delta_1, \dots, \delta_{N-1}, \delta_N) \\ = F_{X(\theta + \theta + \theta_1), \dots, X(\theta + \theta + \theta_{N-1}), X(\theta + \theta)}(\delta_1, \dots, \delta_{N-1}, \delta_N) \end{aligned} \quad (1.63)$$

(3) n^{th} -order wide sense cyclostationarity

The process $X(\theta)$ has a wide sense 2nd-order cyclostationary of period θ if and only if it satisfies the following properties:

- Periodicity of the 1st-order moment

$$\forall \theta, E(x(\theta + \Theta)) = E(x(\theta)) \quad (1.64)$$

where E is the expected value.

- Periodicity of the 2nd-order moment

$$\forall \theta, \forall \delta, R_X(\theta, \delta) = R_X(\theta + \Theta, \delta) \quad (1.65)$$

with

$$R_X(\theta, \delta) = E(X(\theta)X(\theta + \delta))$$

As we obtain a cyclic function, and under certain conditions of convergence, we can compute the coefficients of the Fourier series.

$$E(X(\theta)) = \sum_{k=-\infty}^{+\infty} m_{X,\Theta,k} e^{j2\pi\left(\frac{k}{\Theta}\right)\theta} \text{ with } m_{X,\Theta,k} = \frac{1}{\Theta} \int_{-\frac{\Theta}{2}}^{+\frac{\Theta}{2}} E(X(\theta)) e^{-j2\pi\left(\frac{k}{\Theta}\right)\theta} d\theta \quad (1.66)$$

and

$$R_X(\theta, \delta) = \sum_{k=-\infty}^{+\infty} R_{X,\Theta,k}(\delta) e^{j2\pi\left(\frac{k}{\Theta}\right)\theta} \text{ with } R_{X,\Theta,k}(\delta) = \frac{1}{\Theta} \int_{-\frac{\Theta}{2}}^{+\frac{\Theta}{2}} R_X(\theta, \delta) e^{-j2\pi\left(\frac{k}{\Theta}\right)\theta} d\theta \quad (1.67)$$

However, these different parameters and their harmonic decomposition allow the construction of efficient fault signatures in the vibratory domain. These techniques can require angular resampling of signals in order to transform time representation to angular one.

The ergodicity of some cyclostationary signals makes it possible to evaluate these static parameters by splitting the time series into successive blocks of length Θ (cyclic period) and making a synchronous average (Bonnardot [2004]). A signal is ergodic if its statistical characteristics, computed from the values at the same instant of a large number of different realizations of the system, are identical with those computed from the successive values in time of any of these realizations.

1.2.3 Model Approach

Some relevant features for diagnosis come from a model of the system (Clerc and Marques [2011]). They are two different approaches:

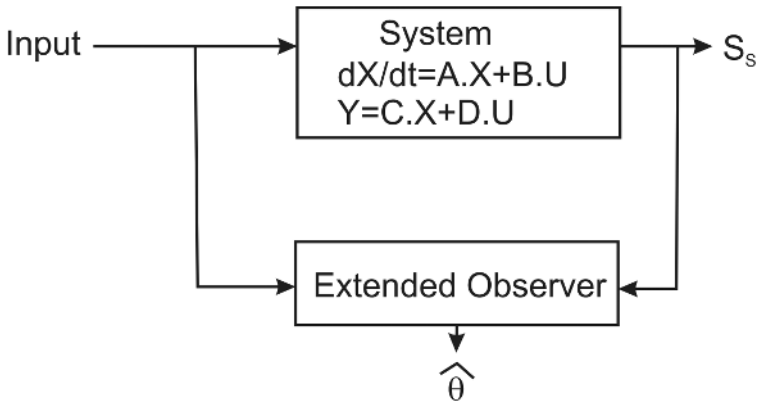


Figure 1.23 Estimation of the parameters

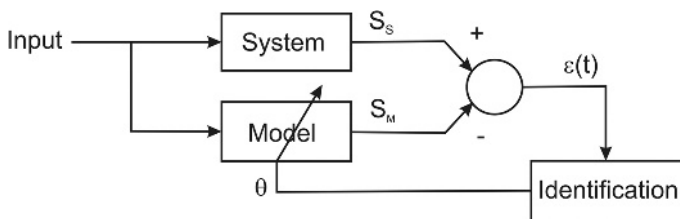


Figure 1.24 Estimation of the parameters

- The first one is based on the estimation of the model parameters.
- The second one is based on an analysis of the residuals between the model and the actual process.

In case of estimated parameters, these later can be identified with off-line algorithms or estimated by extended observers. Table 1.21 gives an overview of the main characteristics of these methods.

We will now describe the main concept of observer. It allows the reconstruction of non-measurable state quantities (rotor flow, motor slip ...) or difficult to measure (torque, speed ...). It associates an estimator and a state correction from the model output error. The estimator is based on the direct solving of the equations associated with the model. It operates in open loop. The observer is an estimator where the estimated quantity is corrected by a term that depends on an error built from measurable quantities.

The observer makes it possible to rebuild the state of an observable system from the measurement of inputs and outputs. It is used in state feedback control when all or part of the state vector cannot be measured. In its extended version,

Table 1.21 Some characteristics of the model approach

	Advantages	Disadvantages
Parameters identification	-Taking into account more complex models that can represent defective behaviors. -Can be used in variable speed drives with converter. -Numerous algorithms that can be adapted to the application.	-Mostly used off-line
Extended observers	-Can be used online. -Can be integrated in controls laws where the state must be rebuilt such as state feedback, etc ... -Can be used in variable speed drives with converter	-Requires simple models. -Requires slowly varying parameters

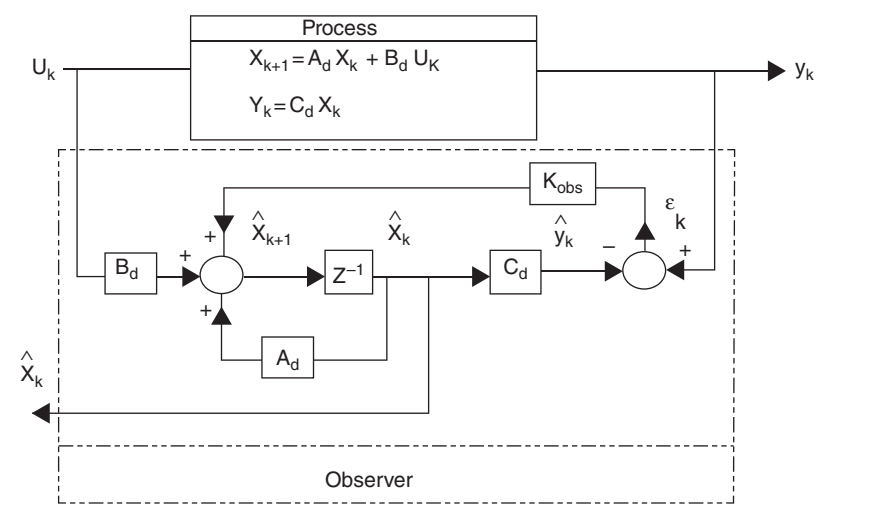


Figure 1.25 Observer

it allows to estimate variable or unknown parameters of a system. It can be represented by the following block diagram (Figure 1.25):

The different types of observers differ on the gain synthesis.

To be able to build an observer, the system must be observable. A discrete linear Single Input Single Output (SISO) system is observable if and only if the initial state of the state vector $X(k.T_{\text{sampling}_0})$ can be reconstructed from the observation of its input and output over a finite number of sampling periods.

A linear SISO described by the following state equations

$$\begin{cases} X_{k+1} = A.X_k + B.U_k & \text{(state equation)} \\ Y_k = C.X_k + D.U_k & \text{(output equation)} \end{cases} \quad (1.68)$$

with the state matrix of dimension n , is observable if and only if the matrix

$$\begin{bmatrix} C \\ CA \\ \dots \\ CA^{n-1} \end{bmatrix} \quad (1.69)$$

is of maximum order (Its determinant is not zero).

In case of nonlinear systems, readers will find an approach to the observability in Gauthier and Kupka [1994] and Hwang and Seinfeld [1972]. We assume the conditions of observability verified in this book, in particular for the extended observers.

1.2.3.1 Kalman Observer

The Kalman filter (Kalman [1960]) gives a realization of the random variable $\hat{X}_k$ representing the state of the system at instant k , knowing the measurement vector $Y = \{y_1, y_2, \dots, y_{k-1}\}$, which minimizes the a priori variance of the estimation error $E(\hat{X}_k^T \hat{X}_k)$ with $\tilde{X}_k = X_k - \hat{X}_k$.

The discretized process is modeled by the following equations:

$$\begin{aligned} X_{k+1} &= A_k.X_k + B_k.U_k + W_k \\ Y_k &= C_k.X_k + V_k \end{aligned} \quad (1.70)$$

W_k and V_k are uncorrelated centered white noises characterized by their covariance matrix.

The noises are centered, so $E(W_i) = 0$ and $E(V_i) = 0$. The covariance matrices are noted:

- $\Sigma_{W_{k_i} W_{k_j}} = [E(W_{k_i} W_{k_j})] = Q_k \delta_{ij}$ where δ_{ij} is the operator defined by $\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ if $i \neq j$ and Q is a positive-definite symmetrical matrix.

- $\Sigma_{V_{k_i} V_{k_j}} = [E(V_{k_i} V_{k_j})] = R_k \delta_{ij}$ where Q is a positive-definite symmetrical matrix.

W_k and V_k are uncorrelated, thus $E(W_{k_i} V_{k_j}) = 0$.

Let us note $\hat{X}_{k|n}$ the instantaneous estimation, at the sample $k.T_e$, minimizing the variance of the estimation error $E((X_k - \hat{X}_k)^T (X_k - \hat{X}_k))$ knowing the measures $y_1, \dots, y_n$

- if $k < n$, $\hat{X}_{k|n}$ is obtained by smoothing;
- if $k = n$, $\hat{X}_{k|n}$ is obtained by filtering;
- if $k > n$, $\hat{X}_{k|n}$ is obtained by a prediction.

The Kalman algorithm minimizing the a priori conditional variance ($n = k - 1$) is decomposed into two steps which are:

1. Prediction or propagation step

$$\begin{aligned}\hat{X}_{k+1|k} &= A_k \cdot X_{k+1|k} + B_k \cdot U_k \\ P_{k+1|k} &= A_k \cdot P_{k+1|k} \cdot A_k^T + Q_k\end{aligned}\tag{1.71}$$

where $P_{k|k}$ is the a priori covariance matrix

2. Correction state

$$\begin{aligned}\hat{X}_{k|k} &= A_k \cdot \hat{X}_{k|k-1} + K_k (Y_k - C_k \hat{X}_{k|k-1}) \\ P_{k|k} &= ([I] - K_k \cdot C_k) P_{k|k-1} \\ K_k &= P_{k|k-1} \cdot C_d^T \cdot (C_d \cdot P_{k|k-1} \cdot C_d^T + R_k)^{-1}\end{aligned}\tag{1.72}$$

where $P_{k|k}$ is the a postiori covariance matrix

This filter has several advantages:

- If the output matrix C_k is constant and the noises are stationary (their stochastic properties are not time-dependent), the computation of the Kalman gain matrix can be done off-line.
- It is recurrent, which reduces the volume of stored data.

1.2.3.2 Extended Observer

(1) Kalman oberver for nonlinear system

Let us consider a nonlinear continuous system

$$\begin{aligned}\frac{dX}{dt} &= f_c(X(t), U(t)) + W(t) \\ Y(t) &= h(X(t)) + V(t)\end{aligned}\tag{1.73}$$

This system is discretized at a sampling step T_e :

$$\hat{X}_{k+1|k} = \hat{X}_{k|k} + \int_{kT_e}^{(k+1)T_e} f(X(t|t_k), u(t)) dt \quad (1.74)$$

The following state equations are obtained:

$$\begin{aligned} X_{k+1} &= f(X_k, U_k) + W_k \\ Y_k &= h(X_k) + V_k \end{aligned} \quad (1.75)$$

The filter linearizes the system at each step around the operating point defined at the previous step. Let us define:

$$\begin{aligned} F_k &= \left(\frac{\partial f}{\partial X} \right)_{X_{k-1|k-1}, U_k} \\ H_k &= \left(\frac{\partial h}{\partial X} \right)_{X_{k-1|k-1}} \end{aligned} \quad (1.76)$$

To calculate $P_{k+1|k}$ the system has been linearized around $\hat{X}_{k|k}$. Thus, the algorithm becomes:

1. Prediction or propagation step

$$\begin{aligned} \hat{X}_{k+1|k} &= f(\hat{X}_{k|k}, U_k) \\ P_{k+1|k} &= F_k P_{k|k} F_k^T + Q_k \end{aligned} \quad (1.77)$$

2. Correction state

$$\begin{aligned} K_{k+1} &= P_{k+1|k} \cdot H_k^T \left[H_k \cdot P_{k+1|k} \cdot H_k^T + R_k \right]^{-1} \\ P_{k+1|k+1} &= ([I] - K_{k+1} H_k) P_{k+1|k} \\ \hat{X}_{k+1|k+1} &= \hat{X}_{k|k+1} + K_{k+1} \cdot (y_k - h(\hat{X}_{k+1|k})) \end{aligned} \quad (1.78)$$

(2) Extended observer

The extended filter (EKF) can be used to identify variable parameters. Two important applications are the monitoring of rotor time constants or the estimation of rotor speed. To achieve this goal, an augmented system is constructed

from the state of the system X_k and the unknown parameters θ_k . A new state vector is defined:

$$\tilde{X}_k = \begin{bmatrix} X_k \\ \theta_k \end{bmatrix} \quad (1.79)$$

By modeling the variation of the parameters by a noise W_{θ_k} and assuming a slow variation of θ_k , the equations of the augmented system are:

$$\begin{bmatrix} X_{k+1} \\ \theta_{k+1} \end{bmatrix} = \begin{bmatrix} A(\theta_k) & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} X_k \\ \theta_k \end{bmatrix} + \begin{bmatrix} B(\theta_k) \\ 0 \end{bmatrix} U_k + \begin{bmatrix} W_{X_k} \\ W_{\theta_k} \end{bmatrix}$$

$$Y_{k+1} = \begin{bmatrix} C(\theta_k) & 0 \end{bmatrix} \begin{bmatrix} X_k \\ \theta_k \end{bmatrix} + V_k \quad (1.80)$$

Then, a Kalman observer can be designed for this nonlinear augmented system with:

$$F(X, u, kT_e) = \begin{bmatrix} A(\theta_k) & \frac{\partial(A(\theta_k)X_k + B(\theta_k)U_k)}{\partial\theta_k} \\ 0 & I \end{bmatrix} \quad (1.81)$$

$$H(X, kT_e) = \begin{bmatrix} C(\theta_k) & \frac{\partial(C(\theta_k)X_k)}{\partial\theta_k} \end{bmatrix}$$

For instance, let us consider the following system (Rayyam et al. [2015]):

$$X_{k+1} = f(X_k, U_k) + W_k$$

$$Y_k = h(X_k) + V_k$$

with

$$X = \begin{bmatrix} i_{sd} & i_{sq} & \varphi_{rd} & \varphi_{rq} & R_r \end{bmatrix} \quad (1.82)$$

$$U = \begin{bmatrix} u_{sd} \\ u_{sq} \end{bmatrix}$$

$$Y = \begin{bmatrix} i_{sd} \\ i_{sq} \end{bmatrix}$$

The state discrete equations are:

$$\begin{aligned}
 f(X_k, U_k) &= \begin{bmatrix} i_{sd}(1 - T_e \left(\frac{R_s}{\sigma L_s} + \frac{1-\sigma}{\sigma \tau_r} \right)) + i_{sq} T_e \omega_s + \varphi_{rd} \frac{T_e L_m}{\sigma L_s L_r \tau_r} + \varphi_{rq} T_e \frac{L_m \omega_{sl}}{\sigma L_s L_r} + u_{sd} \frac{T_e}{\sigma L_s} \\ i_{sd} T_e \omega_s + i_{sq}(1 - T_e \left(\frac{R_s}{\sigma L_s} + \frac{1-\sigma}{\sigma \tau_r} \right)) - \varphi_{rd} T_e \frac{L_m \omega_{sl}}{\sigma L_s L_r} + \varphi_{rq} \frac{T_e L_m}{\sigma L_s L_r \tau_r} + u_{sq} \frac{T_e}{\sigma L_s} \\ i_{sd} T_e \frac{L_m}{\tau_r} + \varphi_{rd} \left(1 - \frac{T_e}{\tau_r} \right) + \varphi_{rq} T_e \omega_m \\ i_{sq} T_e \frac{L_m}{\tau_r} - \varphi_{rd} T_e \omega_m + \varphi_{rq} \left(1 - \frac{T_e}{\tau_r} \right) \\ R_r \end{bmatrix} \\
 h(X_k) &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \\
 \text{with} \\
 \tau_r &= \frac{L_r}{R_r} \\
 \omega_{sl} &= \omega_s - \omega_m
 \end{aligned} \tag{1.83}$$

Thus we obtain the Jacobian matrix F_k :

$$F_k = \begin{bmatrix} (1 - T_e \left(\frac{R_s}{\sigma L_s} + \frac{1-\sigma}{\sigma \tau_r} \right)) & T_e \omega_s & \frac{T_e L_m}{\sigma L_s L_r \tau_r} & T_e \frac{L_m \omega_{sl}}{\sigma L_s L_r} & T_e \frac{L_m \varphi_{rd} - L_m^2 i_{sd}}{L_r^2 L_s - L_m^2 L_r} \\ T_e \omega_s & (1 - T_e \left(\frac{R_s}{\sigma L_s} + \frac{1-\sigma}{\sigma \tau_r} \right)) & -T_e \frac{L_m \omega_{sl}}{\sigma L_s L_r} & \frac{T_e L_m}{\sigma L_s L_r \tau_r} & T_e \frac{L_m \varphi_{rq} - L_m^2 i_{sq}}{L_r^2 L_s - L_m^2 L_r} \\ T_e \frac{L_m}{\tau_r} & 0 & (1 - \frac{T_e}{\tau_r}) & T_e \omega_m & T_e \frac{L_m i_{sd} - \varphi_{rd}}{L_r} \\ 0 & T_e \frac{L_m}{\tau_r} & -T_e \omega_m & (1 - \frac{T_e}{\tau_r}) & T_e \frac{L_m i_{sq} - \varphi_{rq}}{L_r} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{1.84}$$

Then, we obtain the rotor time constant (See Figure 1.26).

1.2.3.3 Unscented Kalman Filter

Unscented Kalman filter (UKF) (Julier and Uhlmann [1997]) deals with online state estimation of a discrete-time nonlinear system. It uses suitable chosen X-state samples (called sigma points), which are passed directly into the state

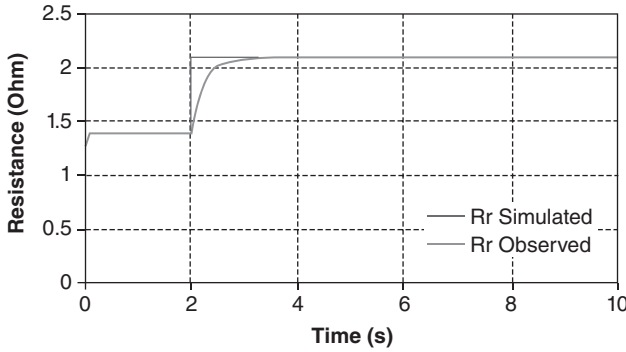


Figure 1.26 Extended rotor resistance observer

function f and whose statistics are extracted from it, rather than linearising it, as in the classical Kalman filter.

Readers could find the algorithm in Wan and Van Der Merwe [2000]. See Figure 1.27.

1.2.4 Parity Space

This method was introduced by Chow and Willsky [1984]. It deals with the generation of residuals based on nonlinear analytical redundancies. In the absence of a failure, the residuals should converge to zero. In the presence of a fault, the residue deviates from zero. The threshold is chosen in order to avoid unsuspected alarms. We will describe it for linear systems (Chow and Willsky [1984]). Let us consider the state equations:

$$\begin{aligned}
 x(k+1) &= A \cdot x_k + \sum_{j=1}^q b_j u_j(k) \\
 y_j(k) &= c_j x(k) \text{ for } j = 1..M
 \end{aligned}
 \tag{1.85}$$

where

- x a state vector of dimension N ,
- A is a constant state matrix of dimension $N \times N$,
- b_j is a constant input vector of dimension $N \times 1$,
- c_j is a constant output vector of dimension $1 \times N$,
- u_j is the known scalar input to the j^{th} actuator
- y_j is the scalar output of the j^{th} sensor.

Initialize with:

$$\begin{aligned}\hat{\mathbf{x}}_0 &= E[\mathbf{x}_0] \\ \mathbf{P}_0 &= E[(\mathbf{x}_0 - \hat{\mathbf{x}}_0)(\mathbf{x}_0 - \hat{\mathbf{x}}_0)^T] \\ \hat{\mathbf{x}}_0^a &= E[\mathbf{x}^a] = [\hat{\mathbf{x}}_0^T \mathbf{0} \mathbf{0}]^T\end{aligned}$$

$$\mathbf{P}_0^a = E[(\mathbf{x}_0^a - \hat{\mathbf{x}}_0^a)(\mathbf{x}_0^a - \hat{\mathbf{x}}_0^a)^T] = \begin{bmatrix} \mathbf{P}_0 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_v & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{P}_n \end{bmatrix}$$

For $k \in \{1, \dots, \infty\}$,

Calculate sigma points:

$$\mathcal{X}_{k-1}^a = [\hat{\mathbf{x}}_{k-1}^a \quad \hat{\mathbf{x}}_{k-1}^a \pm \sqrt{(L + \lambda)\mathbf{P}_{k-1}^a}]$$

Time update:

$$\begin{aligned}\mathcal{X}_{k|k-1}^w &= \mathbf{F}[\mathcal{X}_{k-1}^w, \mathcal{X}_{k-1}^v] \\ \hat{\mathbf{x}}_k^- &= \sum_{i=0}^{2L} W_i^{(m)} \mathcal{X}_{i,k|k-1}^w \\ \mathbf{P}_k^- &= \sum_{i=0}^{2L} W_i^{(c)} [\mathcal{X}_{i,k|k-1}^w - \hat{\mathbf{x}}_k^-][\mathcal{X}_{i,k|k-1}^w - \hat{\mathbf{x}}_k^-]^T \\ \mathcal{Y}_{k|k-1} &= \mathbf{H}[\mathcal{X}_{k|k-1}^w, \mathcal{X}_{k-1}^n] \\ \hat{\mathbf{y}}_k^- &= \sum_{i=0}^{2L} W_i^{(m)} \mathcal{Y}_{i,k|k-1}\end{aligned}$$

Measurement update equations:

$$\begin{aligned}\mathbf{P}_{\mathbf{y}_k \mathbf{y}_k} &= \sum_{i=0}^{2L} W_i^{(c)} [\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{y}}_k^-][\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{y}}_k^-]^T \\ \mathbf{P}_{\mathbf{x}_k \mathbf{y}_k} &= \sum_{i=0}^{2L} W_i^{(c)} [\mathcal{X}_{i,k|k-1}^w - \hat{\mathbf{x}}_k^-][\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{y}}_k^-]^T \\ \mathcal{K} &= \mathbf{P}_{\mathbf{x}_k \mathbf{y}_k} \mathbf{P}_{\mathbf{y}_k \mathbf{y}_k}^{-1} \\ \hat{\mathbf{x}}_k &= \hat{\mathbf{x}}_k^- + \mathcal{K}(\mathbf{y}_k - \hat{\mathbf{y}}_k^-) \\ \mathbf{P}_k &= \mathbf{P}_k^- - \mathcal{K} \mathbf{P}_{\mathbf{y}_k \mathbf{y}_k} \mathcal{K}^T\end{aligned}$$

where, $\mathbf{x}^a = [\mathbf{x}^T \mathbf{v}^T \mathbf{n}^T]^T$, $\mathcal{X}^a = [(\mathcal{X}^w)^T (\mathcal{X}^v)^T (\mathcal{X}^n)^T]^T$, λ =composite scaling parameter, L =dimension of augmented state, $\mathbf{P}_v$ =process noise cov., $\mathbf{P}_n$ =measurement noise cov., W_i =weights as calculated in Eqn. 15.

Figure 1.27 Unscented Kalman Filter Algorithm (Wan and Van Der Merwe [2000]) with permission of IEEE.

Some of the outputs can be linear dependent of the others. They can also depend on some inputs and outputs. These links can provide redundancy relations and, thus, residuals. The general method to generate general redundancy relations is described below.

Let us consider:

$$C_j(k) = \begin{bmatrix} c_j \\ c_j A \\ \dots \\ c_j A^k \end{bmatrix} \text{ for } j = 1..M \quad (1.86)$$

It exist a n_j such as $1 \leq n_j \leq N$ and

$$\text{rank}[C_j(k)] = \begin{cases} k + 1 & \text{for } k < n_j \\ n_j & \text{for } k \geq n_j \end{cases} \quad (1.87)$$

Les us define a nonzero raw vector $\omega = [\omega^1, \dots, \omega^M]$ of dimension $n = \sum_{i=1}^M (n_i + 1)$ such as

$$[\omega^1, \dots, \omega^M] \begin{bmatrix} C_1(n_1) \\ C_2(n_2) \\ \dots \\ C_M(n_M) \end{bmatrix} x(k) = 0 \text{ with } x(k) \in R^N \quad (1.88)$$

Assuming that the system 1.85 is observable, there are only $n-N$ linearly independent ω satisfying 1.88. They define a $(n-N) \times N$ matrix noted Ω . A $n-N$ vector $P(k)$ called generalized parity vector is finally built:

$$P(k) = \Omega \left(\begin{bmatrix} Y_1(k, n_1) \\ \dots \\ Y_M(k, n_M) \end{bmatrix} - B \begin{bmatrix} B_1(n_1) \\ \dots \\ B_M(n_M) \end{bmatrix} U(k, n_0) \right)$$

with

$$Y_j(k, n_j) = \begin{bmatrix} y_j(k) \\ \dots \\ y_j(k + n_j) \end{bmatrix}$$

$$B_j(n_j) = \begin{bmatrix} 0 & 0 & \dots & \dots & \dots & \dots & 0 \\ c_j B & 0 & \dots & \dots & \dots & \dots & 0 \\ \dots & c_j B & 0 & \dots & \dots & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ c_j A^{n_j-1} B & c_j A^{n_j-2} B & \dots & \dots & \dots & c_j B & 0 \end{bmatrix} \quad (1.89)$$

$$B = \begin{bmatrix} b_1 & \dots & b_q \end{bmatrix} \text{ with } q \text{ the number of actuators}$$

$$u(k) = \begin{bmatrix} u_1(k) & \dots & u_q(k) \end{bmatrix}^T$$

$$n_0 = \max(n_1, \dots, n_M)$$

$$U(k, n_0) = \begin{bmatrix} u(k)^T & \dots & u(k + n_0)^T \end{bmatrix}^T$$

$P(k)$ is nonzero if and only if a failure occurs.

But parity space is not limited to linear systems. Zhong et al. [2010] have developed this strategy for linear discrete time-varying systems.

Nguang et al. [2006] described an extension of parity space to nonlinear systems with an LMI approach.

Amrane et al. [2017] applied it on nonlinear systems described in Takagi–Sugeno form by using a transformation by sector nonlinearity.

These methods can be implemented in numerous applications. For instance, they can be used:

- to the fault detection and isolation of an Induction Machine (Amrane et al. [2017]).
- to detect current sensor faults (Berriri et al. [2011]).

- the time-domain synchronous generator diagnosis (Lalami and Wamkeue [2012]).

1.3 Feature Reduction, Principal Component Analysis

The performance of diagnostic and prognostic methods depends very strongly on the choice of pattern vector (fault signature). Its components must be sufficiently informative without being too numerous, at the risk of overloading the useful information and degrading the performance of the classifier. Thus, the selection of the pattern vector components requires:

- selecting informative parameters for diagnosis: Shannon entropy indicator;
- analyzing correlated parameters in order to keep only the relevant ones: Pearson coefficient;
- selecting monotonic parameters for prognostic: Spearman indicator;
- reducing the dimension of the research space for supervised classifier;
- reducing the dimension of the research space for unsupervised classifier: Laplacian score;
- determining the number of classes;
- evaluating the quality of the classification.

First of all, we will present the principal component analysis (PCA), which allows to reduce the representation space. It is used both to visualize the different classes and to perform an unsupervised classification. However, it does not reduce the number of parameters to be calculated.

1.3.1 Principal Component Analysis: A Space Reduction and an Unsupervised Classification

Principal component analysis (PCA) is a multivariate exploratory analysis method introduced by Karl Pearson in 1901 (Pearson [1901]), formalized by Hotelling in 1933 (Hotelling [1933]), and taken up in the work of Rao [1964]. A more detailed history of PCA has been given in Mishra et al. [2017].

The general objective of this method is to summarize the initial variables in the raw data using a reduced number of synthetic variables called principal components. These new variables are a linear combination of the initial variables, allowing the representation of the data to be condensed while maintaining their overall structure (Eke [2018]).

The PCA can then be used:

- to visualize the structure of data;

- to realize an unsupervised classification;
- as a preconditioning step for other classifiers;
- reducing the dimension of the research space for supervised classifier.

The principle of PCA is to reduce the size of the initial data by replacing the initial p variables $[X_1, X_2, \dots, X_p]^t$ by q ones $[X_1^{new}, X_2^{new}, \dots, X_q^{new}]^t$ with $q < p$ with X^t the transposed vector of X .

These later q factors are linear combination of the initial variables.

$$\begin{aligned} X_1^{new} &= \alpha_1^1 X_1 + \alpha_1^2 X_2 + \dots + \alpha_1^p X_p \\ &\dots \\ X_q^{new} &= \alpha_q^1 X_1 + \alpha_q^2 X_2 + \dots + \alpha_q^p X_p \end{aligned} \quad (1.90)$$

They are called factors or principal components.

Equation 1.90 can be rewritten as follows:

$$X^{new} = P_J \cdot X \quad (1.91)$$

with P_J a projection matrix.

The choice of the principal components is made by maximizing the dispersion of individuals according to these factors.

The dispersion of a quantitative variable is generally measured by its variance (or standard deviation). When a cloud of points is available in several dimensions, we speak about inertia, (the sum of the variances of the variables under consideration). Thus, the factors selected must have a maximum variance.

In other words, PCA aims to find the directions of maximum variances in a high p dimensional data and project it into a q lower dimensional subspace without too much loss of information (Figure 1.28).

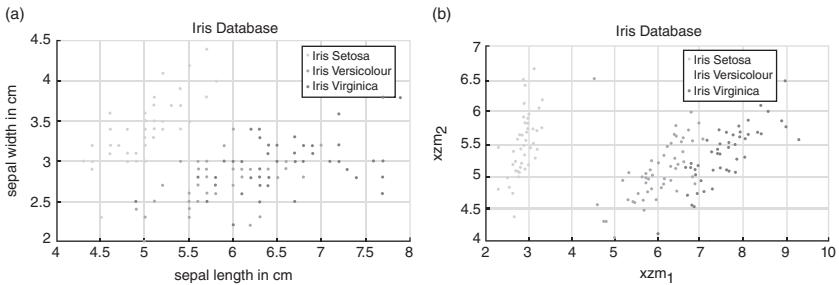


Figure 1.28 Iris data base (a) initial values (b) after PCA.

PCA involves the following steps:

- step 1 - Collect m different data in a matrix X . Each column represents a parameter, each line represents a pattern (data);
- step 2 - Standardize the data: $X_m = \frac{X - \bar{X}}{\sigma_X}$ with $\bar{X}$ the mean of each column of X , and σ_X the variance;
- step 3 - Obtain the covariance matrix of X_m : $\Sigma_X = \frac{1}{p} X_m X_m^T$. Diagonal terms are the variances of each parameter. Non-diagonal terms are the covariances between two parameters. A parameter with a high variance carries information. A high covariance indicates highly redundant parameters;
- step 4 - Calculate Eigen vectors and Eigen values. Diagonalize the matrix in order to maximize the variances and minimize the covariances. Eigen vectors are given with $\det(\Sigma_X - \lambda.I) = 0$. If we note λ_i the Eigen values and V_i the Eigen vectors, then $\Sigma_X.V_i = \lambda_i.V_i$;
- step 5 - Sort the Eigen vectors. Take the eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_p$ and sort them from largest one to smallest one. Then, sort the eigenvectors V_i in the same way in order to build the matrix P_J ;
- step 6 - Compute the new vector: $X^{new} = P_J.X$;
- step 7 - Drop uninformative features from the new set to keep q parameters.

For that, compute the inertia of each axe: $\frac{\lambda_i}{p}$ then the cumulated inertia $\sum_{i=1}^r \frac{\lambda_i}{p}$ where r is the rank of the main component. Finally compute the proportion of variance explained, which is the sum of the eigenvalues of the features you kept until rank r divided by the sum of the eigenvalues of all features $\frac{\sum_{i=1}^r \lambda_i}{\sum_{i=1}^p \lambda_i}$. Keep the q most representative ones.

1.3.2 Intercorrelation

Intercorrelation represents the statistical link between two variables. It doesn't deal with causal interaction. In our case, intercorrelation is used to keep the minimum number of parameters in a set of correlated ones and remove redundant features.

1.3.2.1 Pearson Coefficient "r"

The Pearson coefficient (Pearson and Galton [1895] describes the correlation between two variables by assuming their linear variation. This coefficient always varies between -1 and 1 (Tufféry [2012] and Ben Marzoug [2020]).

Given paired data $\{(X_1, Y_1), \dots, (X_n, Y_n)\}$ with n the sample size, this coefficient can be calculated according to the following equations eq. 1.92 and eq. 1.93.

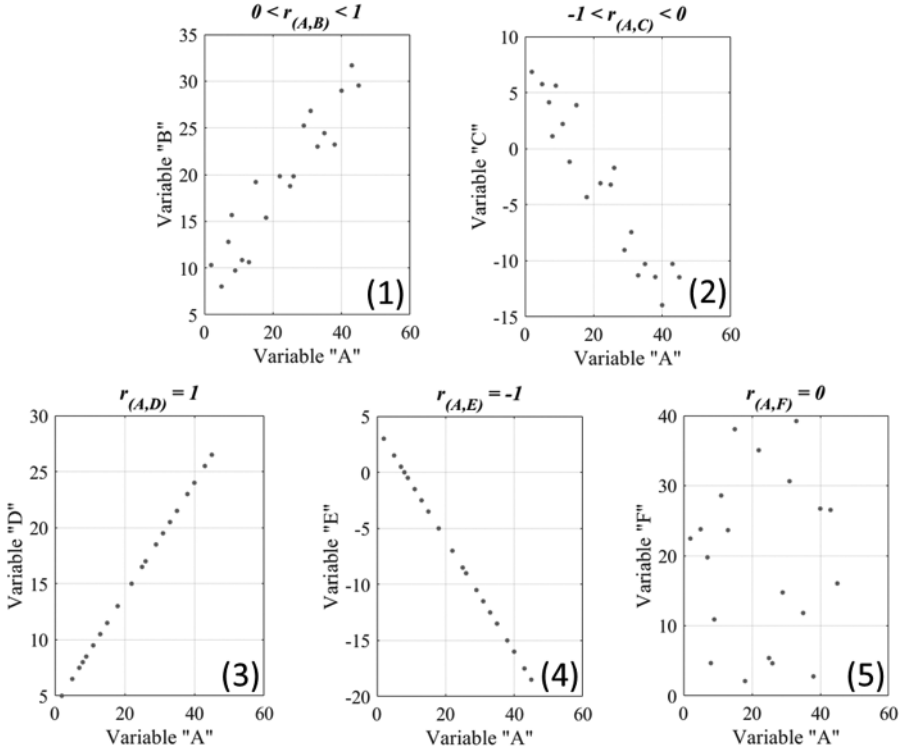


Figure 1.29 Pearson coefficient *Source: Tufféry [2012] and Ben Marzoug [2020]*

$$r = \frac{\sigma_{XY}}{\sigma_X \sigma_Y} = \frac{\frac{1}{n} \sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\frac{\sum_{i=1}^n (X_i - \bar{X})^2}{n}} \sqrt{\frac{\sum_{i=1}^n (Y_i - \bar{Y})^2}{n}}} \quad (1.92)$$

with $\bar{X}$ and $\bar{Y}$, the sample mean of the points X_i and Y_i .

$$r = \frac{1}{n} \sum_{i=1}^n \left(\frac{(X_i - \bar{X})}{\sigma_X} \cdot \frac{(Y_i - \bar{Y})}{\sigma_Y} \right) = \frac{1}{n} \sum_{i=1}^n (XCR_i \cdot YCR_i) \quad (1.93)$$

1.3.2.2 Spearman Coefficient “ ρ ”

The Spearman's index (Spearman [1904]) identifies and quantifies the existence of a monotonic link, linear or not, between the variables under consideration (Gray [2013] and Ben Marzoug [2020]). It deals with the ranks of the variables

under consideration. The following example may explain the procedure for calculating this coefficient. If we consider a variable $X = [7, 10, 26, 20, 23, 2, 5]$, X contains seven observations. The vector $rank_X$ can be obtained as follows: we start by sorting variable X in ascending order. Then, we replace the values taken by variable X by their order number. This gives $rank_X = [3, 4, 7, 5, 6, 1, 2]$. Now, to calculate the Spearman's coefficient for variables X and Y , we can use the (eq. 1.94):

$$\rho = 1 - \frac{6 \sum_{i=1}^n d_i^2}{n^3 - n}, \quad (1.94)$$

with n the number of observations and d_i the difference between $rank_{X_i}$ and $rank_{Y_i}$. Alternatively, the following expression can be used (eq. 1.95):

$$\rho = \frac{cov(rank_X, rank_Y)}{\sigma_{rank_X} \cdot \sigma_{rank_Y}} \quad (1.95)$$

with σ_{rank_X} and σ_{rank_Y} standard deviations of the rank variables.

Figure 1.30 shows the difference between the Pearson's coefficient and the Spearman's index.

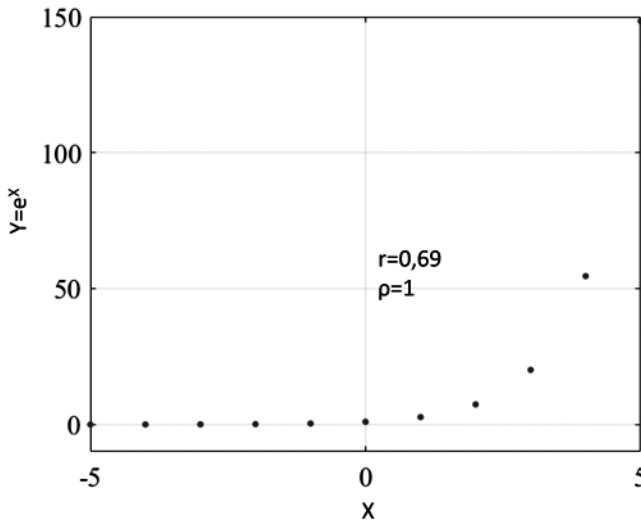


Figure 1.30 Difference between Pearson and Spearman coefficients Source: Gray [2013] and Ben Marzoug [2020]

1.3.3 Information Content: Shannon Entropy

Shannon's entropy (Shannon [1948]) measures the amount of information in a signal. The larger the entropy of a signal is, the more nonredundant information it contains (Gray [2013] and Ben Marzoug [2020]). It is used to remove irrelevant features and keep informative ones.

Given a discrete random variable X , with m possible outcomes $x_1, \dots, x_m$, which occur with probability $P_1, \dots, P_m$, the entropy of X is defined as (eq. 1.96):

$$H(X) = - \sum_{i=1}^m P_i \cdot \log_2(P_i) \quad (1.96)$$

For instance, let us consider X a signal of n individuals, containing m different symbols with $m < n$.

Example: $X = (a, b, a, c, a, b, d, c)$.

In this case, the number of individuals is $n = 8$ and in the vector X , there are four different symbols ("a", "b", "c", and "d") from where $m = 4$.

To measure the quantity of information, we must first calculate the probability of appearance of each of the symbols i with $i \in [1, m]$, we call it P_i . For our example $P = (3/8; 2/8; 1/8)$. As for the calculation of the entropy, we apply equation (eq. 1.96). For our example, we obtain $H(X) = 1.9056$.

1.3.4 Pattern Sizing Reduction for a Supervised Classification

A too large number of parameters involves too many computations and, in some cases, can decrease the performance of the classifier.

To reduce the number of parameters and, thus, the size of the display space, parameter selection methods can be used (Ondel [2006]).

These methods assume that the class structure of the learning set is known, that is it is used in supervised mode.

The objective is to select the d' most representative parameters among the initial d , keeping as well as possible the information on the class structure. Thus, we build a subspace E' from the initial space E .

1.3.4.1 Selection Criteria

The criterion used for the determination of the subspace E' , must represent as well as possible the classes which are supposed to be known. Intuitively, this approach is based on:

- grouping of the points of the same class, that is a minimizing the intra-class variance (notion of compactness);
- separating the different classes, that is a maximizing the inter-class variance (notion of separability);

The intra-class and inter-class variance matrices deal with these two conditions. The intra-class variance matrix involves the samples of each class as well as the centroid (eq. 1.97):

$$\Sigma_w = \frac{1}{N} \sum_{c=1}^M \sum_{v=1}^{N_c} (\underline{X}_{cv} - \underline{m}_c) (\underline{X}_{cv} - \underline{m}_c)^t \quad (1.97)$$

while the inter-class variance matrix characterizes the dispersion of classes among themselves (eq. 1.98).

$$\Sigma_B = \frac{1}{N} \sum_{c=1}^M (\underline{m}_c - \underline{m}) (\underline{m}_c - \underline{m})^t \quad (1.98)$$

with:

- $\underline{m}$: overall center of gravity;
- M : number of clusters;
- $\underline{m}_c$: center of gravity of cluster Ω_c ;
- $\underline{X}_{cv}$: v^{th} vector of the cluster Ω_c ;
- N_c : number of vectors of the cluster Ω_c ;
- N : total number of vectors of the learning base.

We can then define the variance-covariance matrix Σ by (eq. 1.99):

$$\Sigma = \Sigma_w + \Sigma_B \quad (1.99)$$

Two criteria can be defined (eq. 1.100 and eq. 1.101):

$$J_1 = \text{trace}(\Sigma_w^{-1} \cdot \Sigma_B) \quad (1.100)$$

$$J_2 = \frac{\det(\Sigma)}{\det(\Sigma_w)} \quad (1.101)$$

where $\text{trace}(X)$ is the trace of the matrix X (sum of the diagonal elements).

The Fisher ratio can also be applied as a criterion. In the case of a two-class problem, the criterion is expressed, for a parameter (eq. 1.102):

$$J_3(\alpha) = \frac{m_1(\alpha) - m_2(\alpha)}{N_1 \sigma_1^2(\alpha) - N_2 \sigma_2^2(\alpha)} \quad (1.102)$$

where

- $m_c(\alpha)$ with $c = 1, 2$, center of gravity of the class Ω_c , considering only the parameter α

$$m_c(\alpha) = \frac{1}{N_c} \sum_{v=1}^{N_c} \underline{X}_{cv}(\alpha) \quad (1.103)$$

- $\sigma_c^2(\alpha)$: variance of the component α of the vectors of the class Ω_c ;

$$\sigma_c^2(\alpha) = \frac{1}{N_c} \sum_{v=1}^{N_c} [X_{cv}(\alpha) - m_c(\alpha)]^2 \quad (1.104)$$

In the general case of a problem with M classes, $J_3(\alpha)$ is expressed as (eq. 1.105):

$$J_3(\alpha) = \sum_{c=1}^M \sum_{r=1}^{M-1} \frac{m_c(\alpha) - m_r(\alpha)}{N_c \sigma_c^2(\alpha) - N_r \sigma_r^2(\alpha)} \quad (1.105)$$

The greater the criterion is, the more separate the classes are.

1.3.4.2 Sequential Backward Feature Selection and Sequential Forward Feature Selection

The aim of the Feature Selection is to select d parameters from an initial set of m parameters with $d < m$. If the number of features is too great, it is not possible to test all the subset of parameters. Thus, the number of possible combinations is (eq. 1.106)

$$\text{Number of combinations} = \sum_{k=1}^m \frac{m!}{k!(m-k)!} \quad (1.106)$$

For instance, for 10 features, we have 1023 subsets and for 11 features, 2047 subsets.

Thus, it is necessary to use sub-optimal approaches. Two of the main feature selection algorithms come from the work of Efroymson [1960]:

- Sequential Backward Feature Selection (SBS);
- Sequential Forward Feature Selection (SFS).

The Sequential Backward Feature Selection (SBS) starts with the full set of the m features. A selection criterion is computed with all the $m-1$ sets of features extracted from the original one. The combination with the optimal criterion is kept. This removal stage is repeated until a break appear on the criterion. The best combination before this break gives the final set of features. Thus, the features are sequentially removed from the given m features list to reach the list of d features (see Figure 1.31).

In the case of SFS, the algorithm starts with singletons and at each step, all combinations of each remaining feature from the original set with the previous optimal subset are tested (see Figure 1.32).

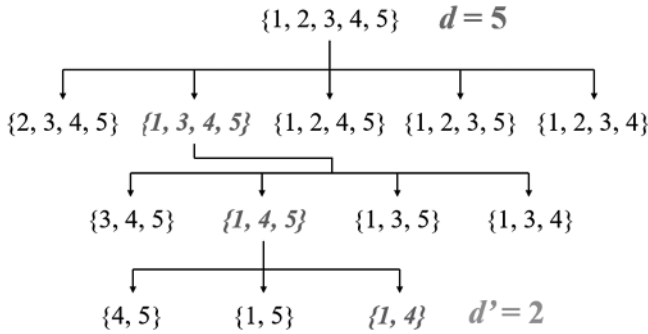


Figure 1.31 Sequential backward feature selection

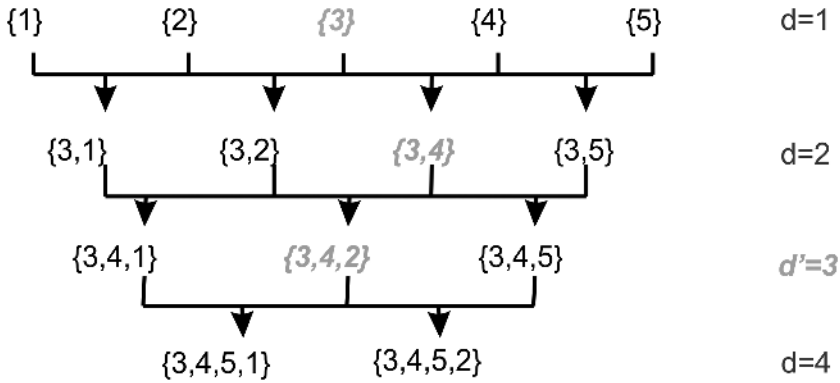


Figure 1.32 Sequential forward feature selection

1.3.5 Pattern Sizing Reduction for an Unsupervised Classification: Laplacian Score

The Laplacian Score (He et al. [2005]) determines the importance of a feature by evaluating its ability to preserve the locality (Eke [2018]).

Let us consider I_r the Laplacian score of the r^{th} variable of the initial pattern vector and f_{ri} the i^{th} value of the r^{th} variable with $i = 1, \dots, n$.

The Laplacian score is built with the following method:

- Construction of the nearest neighbor graph with n nodes. The i^{th} node corresponds to the point x^i . A boundary is built between node i and j if x^i and x^j are close points, that is x^i is among the k closest neighbors to x^j or x^j is among the k closest neighbors to x^i (when label information is available, a boundary can be put between two nodes sharing the same label).

- Construction of the weighting matrix S of the connected nodes. If the nodes i and j are connected, associate $S_{ij} = e^{-\frac{\|x_i - x_j\|^2}{t}}$ where t is a suitable constant and associate $S_{ij} = 0$ in the other case. The matrix S models the local structure of the data representation space. S_{ij} evaluates the similarity between the i^{th} and the j^{th} node.
- For the r^{th} variable, define:

$$\begin{aligned}
 f_r &= [f_{r1}, f_{r2}, \dots, f_{rn}]^t \\
 \mathbf{1} &= [1, 1, \dots, 1]^t \\
 D &= \text{diag}(S\mathbf{1}) \\
 \tilde{f}_r &= f_r - \frac{f_r^t D \mathbf{1}}{\mathbf{1}^t D \mathbf{1}} \mathbf{1} \\
 L &= D - S.
 \end{aligned} \tag{1.107}$$

Then, compute

$$L_r = \frac{\tilde{f}_r^t L \tilde{f}_r}{\tilde{f}_r^t D \tilde{f}_r} \tag{1.108}$$

1.3.6 Choice of the Number of Classes for an Unsupervised Classification

In this case, the label of the different pattern in the learning base and the number of clusters are unknown.

1.3.6.1 Choice of the Number of Classes with a PCA

Two methods are mainly used:

- the elbow criterion;
- the Kaiser's law (Kaiser [1960]).

Both techniques are based on the calculation of inertia (inertia measures the dispersion of the point cloud) explained by the principal component axes. The total inertia of a point cloud is the sum of the squares of the distances of the points from the center of gravity. The average inertia is equal to the total inertia divided by the number of variables.

When the data are projected on the axes of the PCA, the inertia explained by each of the main components can be calculated (Gonzalez [2020]). The variance of a principal component i (variance of the coordinates of the projected points on the axis under consideration) is equal to the inertia λ_i carried by the principal axis i associated with it.

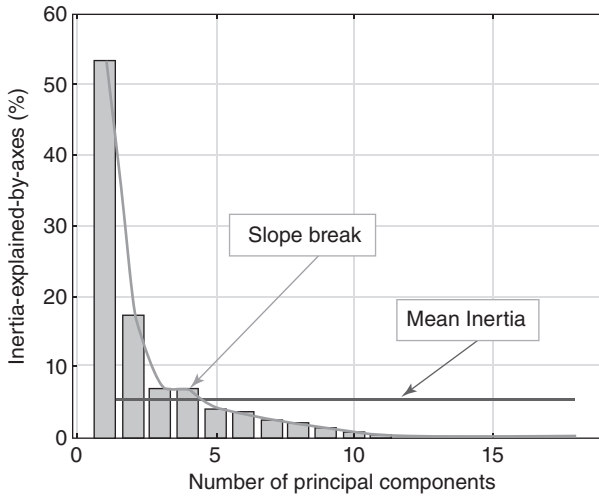


Figure 1.33 Inertia explained by axes

Kaiser's law recommends to retain all axes which have an inertia greater than or equal to the mean inertia (Beavers et al. [2013]).

The elbow criterion is a visual method that proposes to plot the percentage of inertia explained by each component and keep the axes located before the slope break.

The graph of Figure 1.33 shows the inertia explained by each axis of the PCA in relation to the total inertia.

In this example, we have kept four axes, taking into account 80% of the total inertia.

Herrera and Gac [2002] define the quality of the projection of the variables (eq. 1.109):

$$QV_n = \sum_{i=1}^n \text{corr}^2(V, Ax_i) \quad (1.109)$$

where QV_n is the quality index of the projection of the variable V in n axes of the PCA. corr^2 is the square of the Pearson's correlation. A_i is the data vector of the i^{th} axis of the PCA.

1.3.6.2 General Case

One of the most commonly used techniques is to test an increasing number of classes and assess the number of false positives or false negatives on a given test base. One can built for instance a confusion matrix (Figure 1.34).

True Class	setosa	49	1	
	versicolor		36	14
	virginica		15	35
		setosa	versicolor	virginica
		Predicted Class		

Figure 1.34 Confusion matrix after a discriminant analysis of Fisher's iris data

The confusion matrix allows visualization of the performance of an supervised classifier. Each row of the matrix represents the instances in a predicted class, while each column represents the instances in an actual class (or vice versa).

1.3.7 Other Quality Criteria of a Classification

Many indices have been presented in Desgraupes [2017] in order to evaluate the quality of the classification. We will present the main ones:

- R^2 index;
- Calinski–Harabasz index;
- Silhouette index;
- Dunn.

1.3.7.1 R^2 index

R^2 index is based on the calculation of the inertia that is the variance explained by the classes.

According to the Huygens relation and as we have already seen (eq. 1.99), the total inertia can be decomposed into intra-class and inter-class-inertia, Figure 1.35:

$$\sum_{i=1}^n d^2(X, G) = \sum_{j=1}^k \sum_{l=1}^{m_j} d^2(X_l, G_j) + \sum_{j=1}^k m_j d^2(G_j, G) \quad (1.110)$$

$$I_T = I_w + I_B$$

where n is the number of individuals, k is the number of classes, and m_j is the number of individuals in class j . X denotes individuals, G for the global centre of gravity, G_j is the centre of gravity of class j . I_T deals with the total dispersion, I_w represents the within classes dispersion, and I_B denotes the dispersion of the group centers around the global center.

R^2 is given the amount of inertia explained with this partition (1.111):

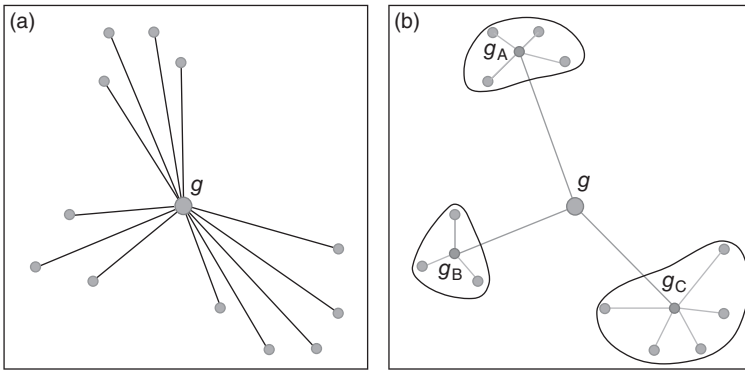
$$R^2(k) = \frac{I_B}{I_T} \quad (1.111)$$

R^2 is always between 0 and 1. R^2 is 0 when we have only one class and 1 when the number of classes equals the number of individuals. To choose the optimal number of classes, R^2 must be as close as possible to 1 without having too many classes and by plotting the curve of R^2 according to the number of classes, we must stop after the last important jump (as in the method described for PCA).

1.3.7.2 Calinski–Harabasz Index

The Calinski–Harabasz criterion (Calinski and Harabasz [1974]) measures the separation between all classes (Desgraupes [2017]). Let us note n is the number of individuals, k is the number of classes. The Calinski–Harabasz, also called pseudo-F is given by (eq. 1.112):

$$PseudoF(k) = \frac{\frac{R^2}{k-1}}{\frac{1-R^2}{n-k}} = \frac{\frac{I_B}{k-1}}{\frac{I_w}{n-k}} \quad (1.112)$$



Total Inertia = Intra-class Inertia + Inter-classes Inertia

Figure 1.35 Illustration of Huygens relation

1.3.7.3 Davies–Bouldin Index

This criterion (Davies and Bouldin [1979]) compares intra-class dispersions with distances between classes. It measures the similarity between classes. Thus, the “best” choice of the number of clusters is the one that minimizes this similarity (Desgraupes [2017]).

Davies–Bouldin index is given by eq. 1.113:

$$DB(k) = \frac{1}{k} \sum_{i=1}^k \max_{a,b \neq a} \left(\frac{S_a + S_b}{d_{ab}} \right) \quad (1.113)$$

1.3.7.4 Silhouette Index

Silhouette index has been by defined by Rousseeuw [1987].

Let us design the data matrix

$$X = \begin{bmatrix} x_{11} & \dots & x_{1p} \\ \dots & \dots & \dots \\ x_{N1} & \dots & x_{Np} \end{bmatrix} \quad (1.114)$$

The set of points belonging to the k_{th} class is defined in eq. 1.115:

$$C_k = \{i \in [1, N] / class(i) = k\} \quad (1.115)$$

According to Wikipédia [2019] and Desgraupes [2017], the silhouette coefficient or score is first defined on a point i , whose group is $k = class(i)$ (see eq. 1.116).

$$a(i) = \frac{1}{|C_k| - 1} \sum_{j \in C_k, j \neq i} d(x_i, x_j) \quad (1.116)$$

where $|C_k|$ is the number of elements of C_k .

The average distance from the point to its neighboring group is given by (eq. 1.117):

$$b(i) = \min_{k' \neq k} \frac{1}{|C_{k'}|} \sum_{i' \in C_{k'}} d(x_i, x_{i'}) \quad (1.117)$$

The silhouette index of point i is provided by (eq. 1.118):

$$s_{silhouette}(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))} \quad (1.118)$$

It can be averaged group by group to compare their homogeneity: those with the highest silhouette coefficients are the most homogeneous. Over the whole

classification, it will be expressed as follows (eq. 1.119):

$$S_{silhouette}(i) = \frac{1}{K} \sum_{k=1}^K \frac{1}{|C_k|} \sum_{i \in C_k} s_{silhouette}(i) \quad (1.119)$$

1.3.7.5 Dunn Index

This index (Dunn [1974]) looks for the minimum distance between two classes in the partition while taking into account the distribution of elements within the classes (eq. 1.120). The greater the distance is, the better the partition. (Desgraupes [2017] and Eke [2018]))

$$ID = \min_{1 \leq i \leq n} \left(\min_{1 \leq j \leq n, j \neq i} \frac{d(w_i w_j)}{\min_{1 \leq k \leq n} d'(w_k)} \right) \quad (1.120)$$

where

- w_i and w_j are two distinct classes;
- $d(w_i w_j) = \min_{1 \leq i \leq n, 1 \leq j \leq n} (d(x, y))$ with $x \in w_i$ and $y \in w_j$;
- $d(w_k) = \max_{x, y \in C_k} (d(x, y))$, maximum distance between two elements of the class.

1.4 Classification Methods

1.4.1 Generalities

In order to diagnose the different states of the electrical drive, clustering techniques can be used. They are used to organize raw data in sets (classes) with similar characteristics. The different classes are organized in order to maximize the intra-class compactness and the inter-class distance (Raschka and Mirjalili [2017], Russell et al. [2010], and Cornuéjols et al. [2018]).

Clustering generally consists of two steps:

- A learning phase where the boundaries of the different classes are defined from a database called the learning database or the training database;
- A classification phase where unknown data are assigned to the different classes.

Rejection rules can be implemented in order not to classify ambiguous data (either too far away from already classified data or too close to two different classes).

This rejected data can also be used to determine the appearance of new classes (Figure 1.36).

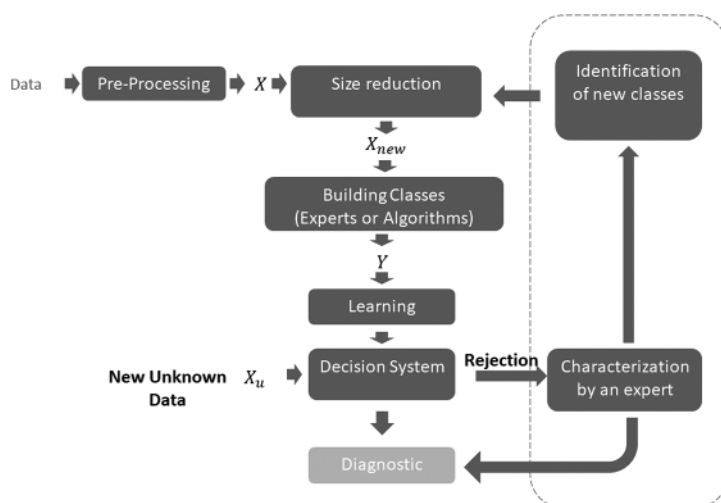


Figure 1.36 General process of classification

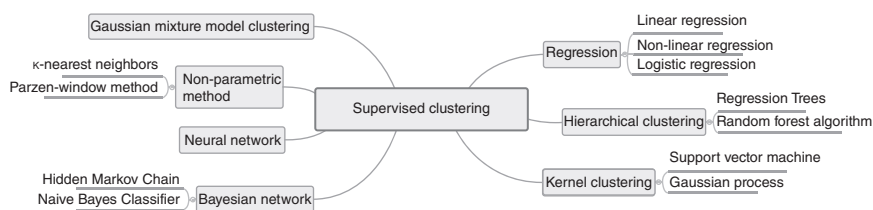


Figure 1.37 Examples of supervised clustering

1.4.1.1 Supervised and Unsupervised Clustering

Supervised clustering uses a learning data based with labeled patterns. It is used to classify unknown raw data in known classes or to build a regression (Bishop [2007], Barber [2011] and Dreyfus et al. [2002]).

Figure 1.37 gives some examples of supervised techniques.

We will present in detail three methods of supervised clustering in Section 1.4.3:

- k nearest neighbors (KNN);
- Support vector machine (SVM);
- Recurrent neural network (RNN).

Unsupervised clustering uses learning data based on unlabeled patterns, but with some diversity. It is used to organize data in sets with similar

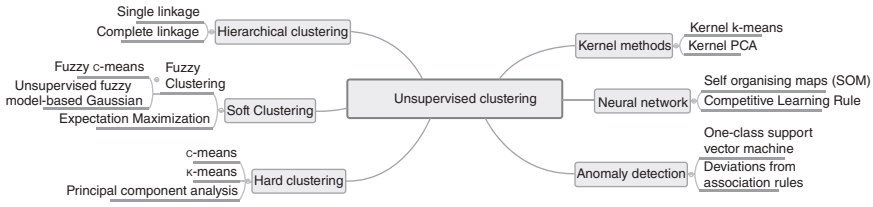


Figure 1.38 Examples of unsupervised clustering

characteristics. The class to which each observation belongs is not known (Yang [2018] and Aldrich and Auret [2013]).

Figure 1.38 gives some examples of unsupervised techniques.

We will present in detail three methods of unsupervised clustering in Section 1.4.3:

- K-means and centroid clustering;
- Hierarchical classification;
- Self-organizing map.

1.4.1.2 Measuring the Similarity: Different Distances

Pattern recognition classifiers are based on similarity measures. These rely on distances between two pattern vectors.

Let us define the points (eq. 1.121) of a set E of objects (elements, patterns):

$$x_i = \begin{bmatrix} x_{i1} \\ x_{i2} \\ \dots \\ x_{id} \end{bmatrix} \quad (1.121)$$

with Σ^{-1} the inverse of the covariance matrix of the descriptive variables.

The distance is a function from $E \times E$ to R^+ , which fulfills the following properties:

$$\left\{ \begin{array}{l} d(x_i, x_j) = d(x_j, x_i) \\ d(x_i, x_j) \geq 0 \\ d(x_i, x_i) = 0 \\ d(x_i, x_j) \leq d(x_i, x_k) + d(x_k, x_j) \end{array} \right. \quad (1.122)$$

The main distances which are used for classifiers are given in Table 1.22:

Table 1.22 Main distances

Euclidian distance	$d(x_i, x_j) = \left(\sum_{k=1}^d (x_{ik} - x_{jk})^2 \right)^{\frac{1}{2}}$
Manhatann distance or city-block	$d(x_i, x_j) = \sum_{k=1}^d (x_{ik} - x_{jk})$
Tchebychev distance d_{∞}	$d(x_i, x_j) = \max_{1 \leq k \leq d} x_{ik} - x_{jk} $
Mahalanobis distance d_{∞}	$d(x_i, x_j) = (x_1 - x_2)' \Sigma^{-1} (x_1 - x_2)$

Figure 1.39 shows some isovalues for Euclidean and Mahalanobis distances:

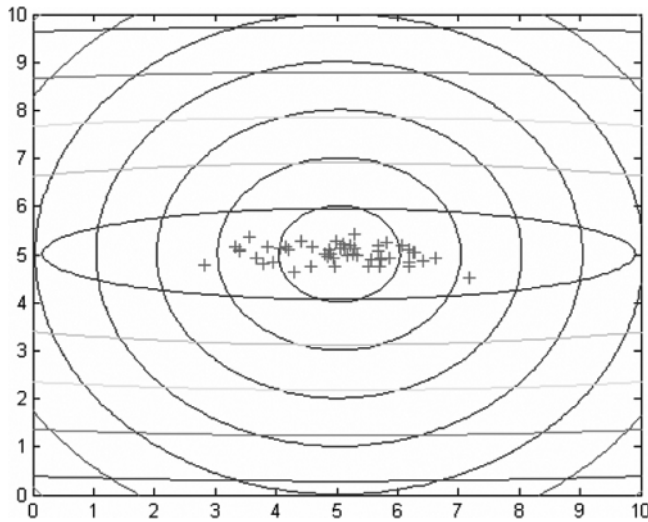


Figure 1.39 Isovalues of Euclidean (circles) and Mahalanobis (ellipse) distances from the center of gravity of the class Ω

As we can see, the Mahalanobis distance takes into account of the variance on each axis of the Ω class, which is not realized by the Euclidean distance.

1.4.2 Supervised Clustering

Wu et al. [2008] have written a synthesis paper dealing with algorithms in data mining in the research community in 2008. They present ten of these algorithms.

1.4.2.1 k Nearest Neighbors

The choice of k is sensitive because larger k would give a reliable estimate, but this choice remains arbitrary. In fact, if k is too small, the result can be sensitive to noise, and if k is too large, the result may include other classes. Consequently, assigning a new observation to a known class can lead to a diagnostic error. It is therefore advisable to choose a rejection class. To do this, the decision rule can include two rejection possibilities which can be based on distance rejection or ambiguous rejection Figure 1.40.

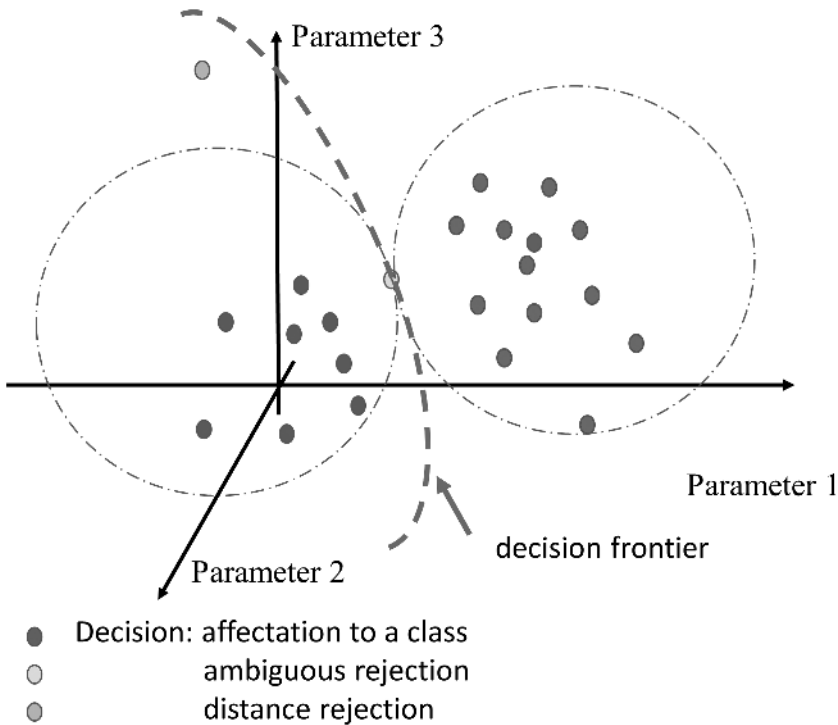


Figure 1.40 kNN rules applied to a new observation

In the case an observation is close to two classes and at the edge of the decision boundary, the decision to reject due to ambiguity is the safest. For this purpose, this new observation can be assigned to a specific and fictitious class of rejection due to ambiguities. However, for a new observation, which is located in an unexplored area, a new specific class can be attributed to it. It is based on distance rejection. Of course, the distance threshold is arbitrarily defined by the

user. However, it can be defined thanks to the distance between the observation and the center of gravity of the assignment class.

There is, however, a weight disadvantage with this approach. This method requires the calculation of distances to all observations in the learning set. This can be cumbersome in terms of computation time, especially if the size of the observations is large. To do this, a mesh-based approach can be efficient. For example, this would involve checking the location of a new observation in a class area. Thus, the calculation of distances will no longer be necessary with vectors. The algorithm is then faster.

When two classes overlap because they are not sufficiently distant from each other, one proposal would be to assign the new observation to the class that is predominantly represented among its closest neighbors Figure 1.41. A nonexclusive decision rule may be interesting in that an observation located at the border of a class or at the border of contiguous or disjoint classes may signify

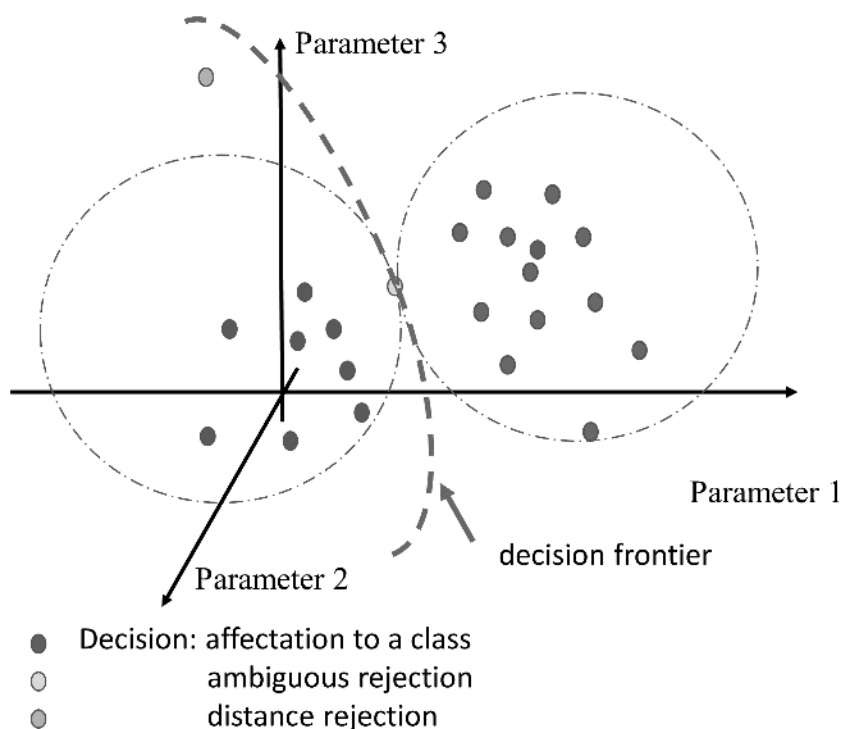


Figure 1.41 Class overlap

a transitional state between classes. The transition from exclusive to nonexclusive assignment must be characterized by a gradation between absolute nonmembership and full membership.

Since the similarity of an object with a reference form is often relative, in general, a membership function characterizing the degree of compatibility between the observation and a class is introduced. There is no general method to construct the membership function associated with a given set. The form of this function is closely related to the context. However, it would be interesting, for a new observation, to take into account not only the distance with the different center of gravity of each class, but also to give an advantage to the classes which contain the greatest number of closest neighbors.

In any case, a new observation located between two classes can correspond to an intermediate state of this mode. By using the exclusive or nonexclusive decision rule with the reject options, the observation can be rejected. It is therefore necessary to take into account the evolution of the different operating modes of the system.

A new observation can be rejected due to ambiguity or due to distance, even though it belongs to a mode of operation defined by its classes, due to poor feedback. The learning set does not exhaustively cover all modes of operation. Also, in order to establish an efficient diagnosis, it will be necessary to take into account the evolution of the operating modes of the system to be diagnosed.

An exclusive classification does not give strong information about the observation except that it belongs to a class. This observation can be at the edge of the class as well as inside the class. In the same way, what about an observation at the border of two contiguous classes or disjoint classes? A way should be implemented to follow the evolution of an observation inside a class and to associate to it a degree of confidence of belonging to this class. The transition from exclusive to nonexclusive assignment should be made by establishing a degree of belonging that could be modulated through the notion of proximity.

Instead of a binary classification, the use of membership functions, as used in fuzzy logic, will allow a gradual notion to be introduced. One possibility is to use an C^∞ function such as a Gaussian function. It would be centered on the center of gravity of the class. A threshold defined in relation to the standard deviation allows to define: the whole membership to a class ($\mu = 1$), the degree of membership to a class ($0 < \mu < 1$), or its non membership ($\mu = 0$).

1.4.2.2 Support Vector Machine

Among classifications techniques, one can find in the literature the support vector machine (SVM), which is characterised by its robustness. It was introduced by Vapnik [1995]. The goal of the SVM is to find a hyperplane separating

two classes of data thanks to support vectors. A new datum coming from observation, is classified by evaluating the distance from this datum to a hyperplane. So, a label is generated, which can have two types of value based on the distance. If the distance is positive, thus the label is $+1$, and if the distance is negative, the label is -1 .

The equation of the hyperplane is:

$$\omega \cdot x + b = 0 \quad (1.123)$$

with ω a normal vector and b a constant.

However, a margin distance d has to be defined as the sum of two distances. The first one is obtained as the distance from the hyperplane and the closest positive datum (observation), the second one as the closest negative datum (observation). The training process of the SVM consist to maximise the margin, which is equivalent to minimise $\|\omega\|$ knowing that all data should satisfy the condition:

$$y_i (\omega \cdot x_i + b) \geq 0 \quad (1.124)$$

Figure 1.42 shows a hyperplane which separates data into classes.

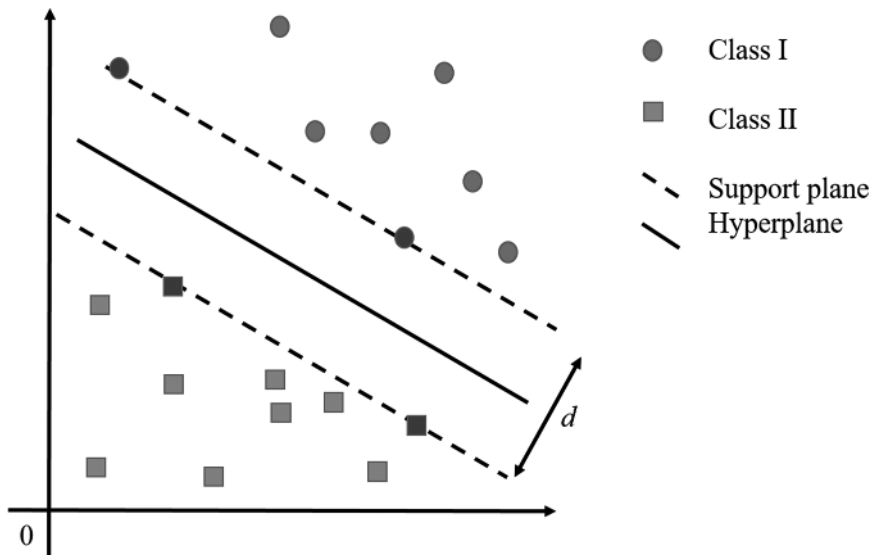


Figure 1.42 Binary SVM

The hyperplane can be obtained by solving a quadratic optimization problem. So, a dual Lagrangian formulation should be used as follows:

$$L_D = \sum_{i=1}^M \alpha_i + \sum_{i,j=0}^M \alpha_i \alpha_j y_i y_j x_i x_j \quad (1.125)$$

knowing that constraints of this problem are:

$$C_M \geq \alpha_i \geq 0$$

and

$$\sum_{i=1}^M \alpha_i y_i = 0$$

where the parameter C deals with the tolerance to errors and the coefficient α_i is the Lagrangian multipliers which is greater than zero if the datum x_i is in the support vector, otherwise α_i is equal to zero.

In order to take into account nonlinear boundaries, a *kernel* can be used. This allows a projection of the research space into a space of higher dimension where the border becomes an hyperplane.

However, in case of nonlinear *kernel*, the expression of the formulation becomes:

$$L_D = \sum_{i=1}^M \alpha_i - \sum_{i,j=0}^M \alpha_i \alpha_j y_i y_j K(x_i x_j) \quad (1.126)$$

The discriminant function for a new datum is based on:

$$f(x_{new}) = \sum_{i=1}^M \alpha_i y_i K(s_i, x_{new}) + b \quad (1.127)$$

where $s_i \in [1 \dots M]$.

Among all *kernel* functions, one can find in the literature these different expressions Breuneval R. and Mansouri [2018]:

- $K(x_i x_j) = x_i \cdot y_j$
- $K(x_i x_j) = (x_i \cdot y_j + 1)^p$
- $K(x_i x_j) = \exp \frac{\|x_i - y_j\|^2}{2\sigma^2}$
- $K(x_i x_j) = \tan(a \cdot x_i \cdot y_j + b)$

1.4.2.3 Recurrent Neural Network

The structure of a neural network is simple. It is based on layers which are composed of perceptrons (mathematical functions). Each output of perceptron is connected to an other perceptron. Figure 1.43 shows a neural network

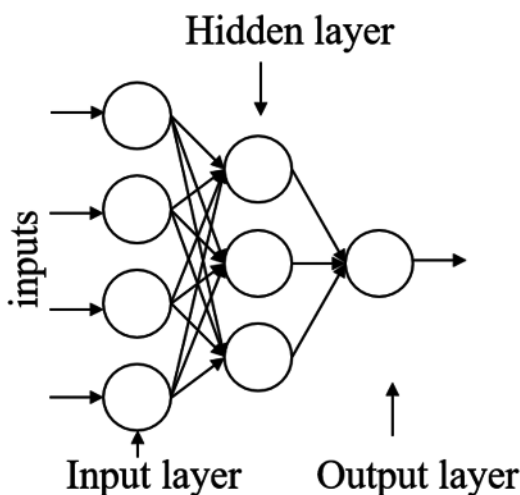


Figure 1.43 A basic neural network with one hidden layer

composed of three layers. The first one is the input layer composed of four perceptrons, the second layer is called hidden layer which is composed of three perceptrons and one output layer composed of one perceptron. The number of layers as the number of perceptrons in each layer depends on the user Figure 1.43.

Hayashi et al. [2002] has used a neural network composed of 240 neurons in the input layer, 30 neurons in the hidden layer and one neuron in the output layer. His neural network was dedicated to fault diagnosis of a machine based on the spectrum of vibrations from it.

The learning of the diagnosis network is performed with some normal and abnormal data coming from the process to monitor. Usually, the learning process is based on a gradient method as described by Rumelhart et al. [1986]. However, there are other optimization algorithms such as the Levenberg-Marquardt, which combines two strategies which are: acceleration of the convergence at the beginning of the optimization then slow convergence when we are close to the optimum, Korbicz [1997]. as presented in the Figure 1.44, an error is evaluated thanks to a comparison of the output of a process with the output of the neural network. Based on this error, the optimization algorithm modifies all coefficients of the neural network in order to minimize the error.

The process can be a multiple input multiple output system. We can use a neural network whose inputs would take into account the past and would then be written in the form:

$$u = [u_0 \ u_1 \ u_2 \ \dots \ u_p]$$

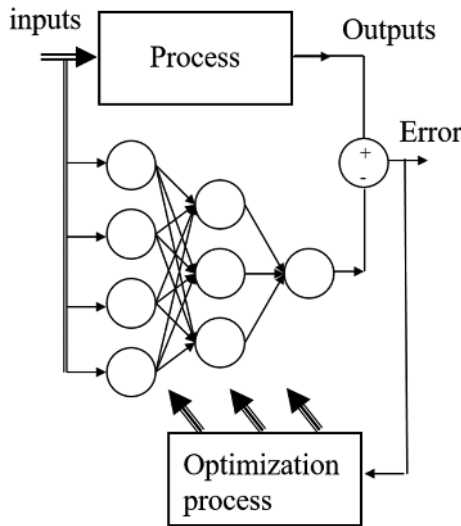


Figure 1.44 Optimization process of a basic neural network

Then, the dynamics of the system can be taken into account by also taking into account the past of the output in the form:

$$y = [y_1 \ y_2 \ y_3 \ \dots \ u_q]$$

Therefore, the neural network should duplicate the dynamic behavior of the process which may be nonlinear. By way of illustration, we represent a neural network duplicating a process using three samples as input u and three samples for the output y Figure 1.45.

Of course, once the learning process is complete, the coefficients (weights) within the neural network will be fixed. So, we do have a recurrent neural network.

Neurons, or perceptrons, are nonlinear functions called “sigmoid.” The equation governing the neuron is simple and is written in the following form:

$$y = f(\Sigma w_i \cdot x_i) \quad (1.128)$$

where w is the connecting weight between two perceptrons of two consecutive layers, and x is the output of the previous perceptron. The sigmoid function can be written as:

$$f(\Sigma w_i \cdot x_i) = \frac{1}{1 + \exp^{-\Sigma w_i \cdot x_i}} \quad (1.129)$$

An application of neural networks can also be the detection of defects in processes. To do this, the inputs will be residuals, for example, residuals formed

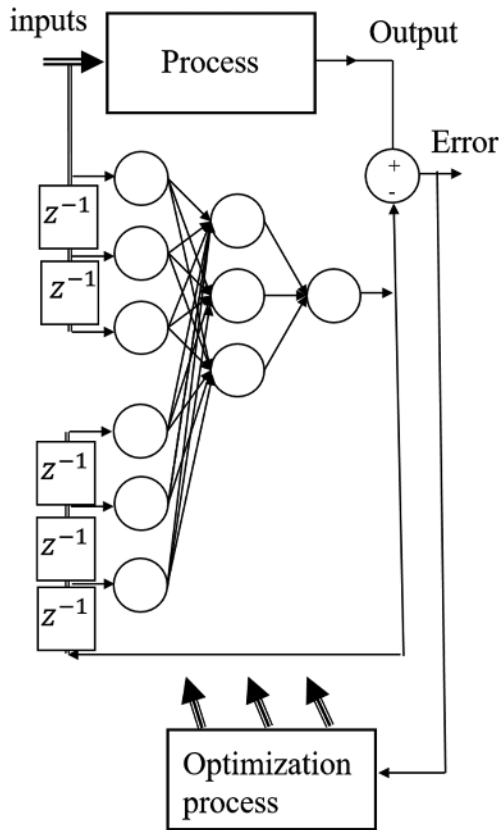


Figure 1.45 A basic recurrent neural network

by the difference in behavior between the output of a process and its model. Consequently, the x variables at the input of the neural network will be the residuals. It is necessary to implement a learning phase in order to train the neural network to detect defects. Therefore, a knowledge and database is mandatory. We represent in Figure 1.46 one structure among many others.

1.4.3 Unsupervised Clustering

There are several methods of unsupervised classification, also known as coalescence methods. If we limit ourselves to exclusive assignment, these methods are divided into two parts (Jain et al. [1999]):

- Hierarchical classification methods (ascending or descending);
- Classification by partition.

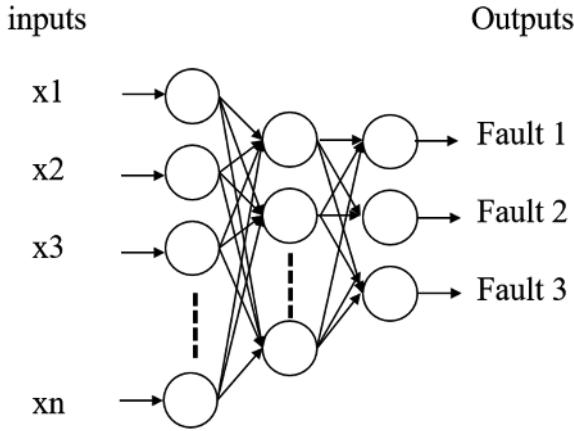


Figure 1.46 A basic neural network for faults detection

1.4.3.1 Hierarchical Classification

Hierarchical classification produces a hierarchical set of classes according to a linkage criterion. The classes, which are nested, are linked recursively either in agglomeration mode as Ascending Hierarchical Classification (AHC) or in division mode as Descending Hierarchical Classification (DHC). In the hierarchical ascending classification, the classes are determined iteratively in four steps:

- Step 1: the classification algorithm begins by considering each observation as a class. The initialization phase is defined by the variable:

$$\Omega_i^\ell = \{x_i\} \quad \forall i \quad \text{and} \quad \ell = 0 \quad (1.130)$$

where the variable Ω_i^ℓ denotes the membership class of observation x_i at the ℓ^{th} iteration, with $\ell=0$ for the initialization phase.

- Step 2: the similarity between two classes is defined by a link criterion. This criterion evaluates two classes using a similarity measure which evaluates the similarity between two classes. The “Sim” similarity between two classes is defined as follows:

$$Sim(\Omega_l^\ell, \Omega_h^\ell) = \beta(\Omega_l^\ell, \Omega_h^\ell) \quad (1.131)$$

where the link criterion is represented by the function β which can be a link criterion by minimum jump, maximum jump, average distance or by the intra-class inertia represented by the Ward criterion. Equation 1.131

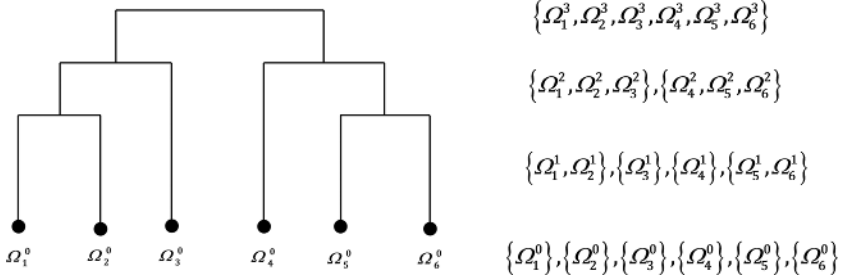


Figure 1.47 Example of a dendrogram

indicates that all observations of class (ℓ) should be compared with all observations of class (h). The link criterion β returns a similarity between the two classes.

- Step 3: select the pairs of classes which have a very great similarity $Sim(\Omega_l^\ell, \Omega_h^\ell)$. Let Ω_{l*}^ℓ and Ω_{h*}^ℓ two classes selected as follows:

$$[\Omega_{l*}^\ell, \Omega_{h*}^\ell] = \arg \max_{l, h \in [1-M]; l \neq h} Sim(\Omega_l^\ell, \Omega_h^\ell) \quad (1.132)$$

- Step 4: merge the two classes with an index $l *$ and $h *$

$$\Omega_{l*}^{\ell+1} = (\Omega_{l*}^\ell \cup \Omega_{h*}^\ell) \quad (1.133)$$

$$\Omega_{h*}^{\ell+1} = \{\} \quad (1.134)$$

Steps 2–4 are repeated until all the classes are merged into one class (see Figure 1.47). In the standard case, the aggregation stops when a stop criterion is reached. For example, the stopping criterion is obtained by a maximum distance between the different classes as well as by a low dispersion of observations in each class. Class fusion sequences can be represented graphically by a tree diagram, called a dendrogram. The dendrogram is a multilevel hierarchy with the merging of two classes at each level (see Figure 1.47).

(1) Linkage criterion

The merging of classes is based on the similarity between them. This similarity is calculated from a linkage criterion. The criterion of connection by minimal jump $\beta_{\text{jump-min}}$ is the first criterion which can be taken into account. It seeks

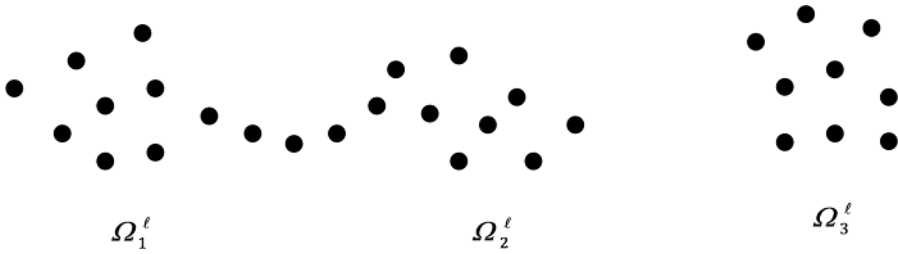


Figure 1.48 Chain effect

the greatest similarity between two observations x_i and x_j which are located in the classes Ω_l^ℓ and Ω_h^ℓ . It is expressed as follows:

$$\beta_{\text{jump-min}}(\Omega_l^\ell, \Omega_h^\ell) = \max \{d^{-1}(x_i, x_j)\} \quad \forall x_i \in \Omega_l^\ell, x_j \in \Omega_h^\ell \text{ and } l \neq h \quad (1.135)$$

The variable x_i denotes the i -th observation and x_j denotes the j -th observation. The similarity distances proposed in the literature can be used for the calculation of the link criterion. However, the linkage criterion by minimal jump is ill-suited in the case where the classes have a stretched shape which has the disadvantage of producing chain effects as presented in Figure 1.48.

- Maximum hop link criterion

The linkage criterion by maximum jump $\beta_{\text{jump-max}}$ calculates a dissimilarity between the classes based on the maximum distance between two observations which are located in the classes Ω_l^ℓ and Ω_h^ℓ . It is expressed as follows:

$$\beta_{\text{jump-max}}(\Omega_l^\ell, \Omega_h^\ell) = \min \{d^{-1}(x_i, x_j)\} \quad \forall x_i \in \Omega_l^\ell, x_j \in \Omega_h^\ell \text{ and } l \neq h \quad (1.136)$$

- Connection criterion by average distance

The average distance linkage criterion $\beta_{\text{jump-avg}}$ calculates the similarity between two classes by finding the average distance between all pairs of observations in classes Ω_l^ℓ and Ω_h^ℓ . It is expressed as follows:

$$\beta_{\text{dist-avg}}(\Omega_l^\ell, \Omega_h^\ell) = \frac{1}{|\Omega_l^\ell|} \frac{1}{|\Omega_h^\ell|} \sum_{x_i \in \Omega_l^\ell} \sum_{x_j \in \Omega_h^\ell} d(x_i, x_j) \quad l \neq h \quad (1.137)$$

$|\Omega_l^\ell|$ and $|\Omega_h^\ell|$ denote the number of observations in the classes Ω_l^ℓ and Ω_h^ℓ , respectively. This criterion can be seen as a compromise between $\beta_{\text{jump-min}}$ and $\beta_{\text{jump-max}}$.

- Ward's criterion

The Ward's criterion β_{ward} corresponds to the intra-class inertia which results from the fusion of Ω_l^ℓ and Ω_h^ℓ . It is expressed as follows:

$$\beta_{\text{ward}}(\Omega_l^\ell, \Omega_h^\ell) = \frac{|\Omega_l^\ell| \cdot |\Omega_h^\ell|}{|\Omega_l^\ell| + |\Omega_h^\ell|} d_{\text{Euc}}^2(g_l, g_h) \quad l \neq h \quad (1.138)$$

where g_l and g_h are the centers of gravity of the classes Ω_l^ℓ and Ω_h^ℓ , respectively; $d_{\text{Euc}}^2(g_l, g_h)$ is the square of the Euclidean distance between the centers g_l and g_h .

1.4.3.2 K-means and Centroid Clustering

Classification by partition subdivides a set of observations into several subsets. Each subset represents a class containing at least one observation. Each class is represented by a function (F) which assigns observations to these classes based on their similarities. Let $\Omega_1^\ell, \Omega_2^\ell, \dots, \Omega_M^\ell$ be a set of M -classes. The function (F) which affects the set of N -observations $x_i \quad i \in (1, \dots, N)$ in M -classes is given as follows:

$$F(x_i) = \arg \min_{1 \leq k \leq M} \{d(x_i, g_k)\} \quad (1.139)$$

where the metric $d(x_i, g_k)$ represents a measure of similarity between the i -th observation x_i and the center of gravity of the class $\Omega_k^\ell \quad k \in (1, \dots, M)$. The variable ℓ designates the ℓ -th iteration.

The K-means algorithm of Forgy [1965] is the most used algorithm in partition classification. It has four stages:

- Step 1: Choose a number of M -observations among the set of N -observations to be partitioned and consider them to be the centers of gravity of the M -classes.
- Step 2: Assign each observation x_i to its closest center of gravity using eq. 1.139.
- Step 3: When all the observations are affected, recalculate the centers of gravity of the M -classes by:

$$g_k = \frac{1}{|\Omega_k^\ell|} \sum_{x_i \in \Omega_k^\ell} x_i, \quad 1 \leq k \leq M \quad (1.140)$$

where $|\Omega_k^\ell|$ denotes the number of observations in the class Ω_k^ℓ .

- Step 4: Repeat steps 2 and 3 until the centers of gravity of the M -classes no longer vary.

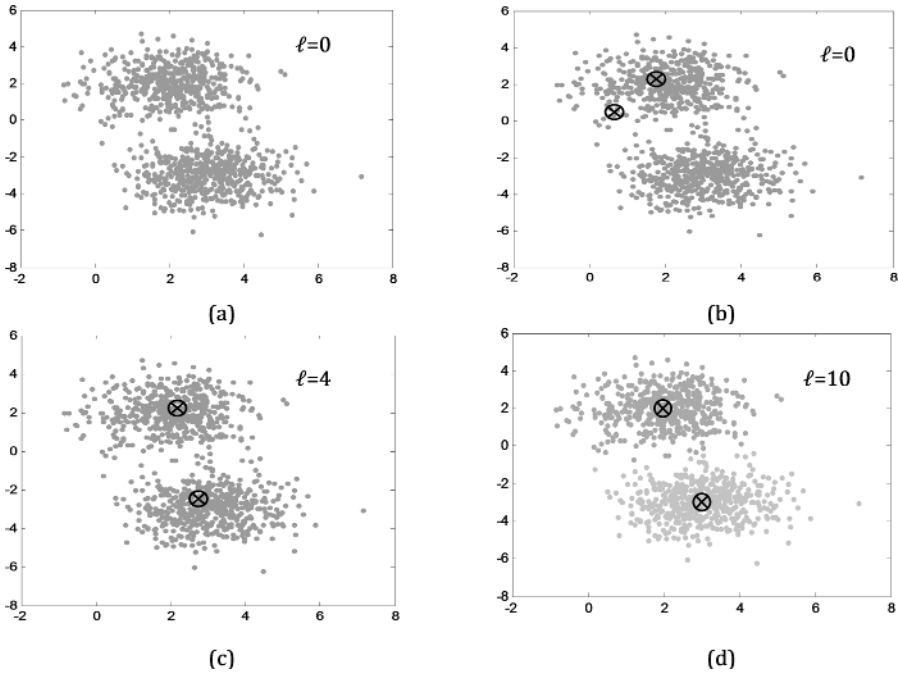


Figure 1.49 K-means algorithm for the partitioning of two classes

An example showing the execution of the K-means algorithm is given in Figure 1.49. At each iteration, the observations are assigned to the closest center of gravity (shown in color, the observations take on the same color as the center of gravity of the class to which they are assigned). The center of gravity of each class is then recalculated as a function of the average of the observations attributed to it. The final partitioning gives us a good separation between the two classes by giving a distinct color for each of them. As we can see in Figure 1.49, observations are represented by points and the centers of gravity of the two classes are represented by crosses. (a) represents the set of observations. (b) represents randomly chosen centers of gravity. (c) and (d) illustrate the distribution of the set of observations after four and ten iterations of the K-means algorithm.

1.4.3.3 Self-organizing Map

Designed by McCulloch and Pitts in 1943, the neuron model derived from the biological model experienced a major boom in the 1980s. This emergence is closely linked to its ability to process incomplete, uncertain or noisy data and

also to its application in various fields ranging from decision support, planning and even to the fields of pattern recognition and classification (Dumolard [1994]). Kohonen self-organizing networks are competitive and dynamic networks, in the sense that they tend to elect a winning neuron and favor it, on the one hand, and create or destroy neurons, on the other hand. The topological map model proposed by Kohonen is a self-organization process that will project the initial data onto a discrete and regular space of low dimension (generally 1, 2, or 3D). The spaces used are regular trellises. Each node is occupied by a neuron. The notion of neighborhood between neurons then follows directly from the structure and defines a topology of the map. Thanks to the self-organization process, the topology which links the initial data is preserved at the level of the responses proposed by the network (IFTENE and Mahi [2003] and Kohonen [1989]).

Kohonen topological maps consist of two layers:

- the input layer is only used for the presentation of the shapes to be classified and the states of all its neurons are forced to the values of the input data;
- the adaptation layer is formed by a network of neurons. The choice of the geometry of the network used is made a priori. The neurons used at this level are linear neurons, each of them plugged into all elements of the input layer. In order to enable the process of self-organization, the weights that link the two layers are adaptively determined.

(1) Structure and operation

In Figure 1.50, we present a network organized in a one-dimensional layer. Each output neuron NE_j $j \in (1, \dots, K)$ is connected to a number x_i $i \in (1, \dots, N)$ of input neurons through N connections ($K * N$) of respective weight $W_{j,i}$. When presenting an input, a neuron on the map is selected based on the minimum distance separating it from that entry.

(2) Description of the algorithm

Here we describe the 5 steps of the algorithm which are composed of:

- Step 0:
 - Initialization of the weights of the output neurons (low random values);
 - Choice of map shape;
 - Initialization of the learning parameters:
 - * Learning rate $\alpha(t)$:

$$\alpha(t) = \frac{\alpha_0}{1 + K_\alpha t}$$

K_α, α_0 : Positive constants.

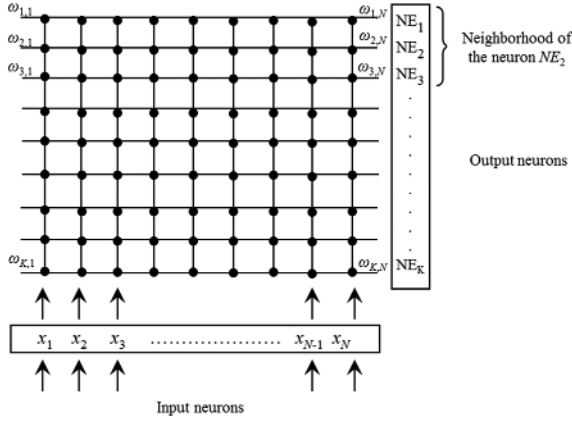


Figure 1.50 Architecture of a self-organizing map (1D network)

* $\nu(s, NE_f, t)$: Neighborhood function.

s : Neuron neighborhood of NE_f $f \in \{1, \dots, K\}$.

The neighborhood function $\nu(s, NE_f, t)$ is a Gaussian, evolving with time, of the form:

$$\nu(s, NE_f, t) = e^{-\frac{d^2(s, NE_f)}{2\sigma^2(t)}}$$

$d(s, NE_f)$: distance between the winning neuron NE_f and its neighborhood identified by the index s defined by:

$$\|\omega_{NE_f} - \omega_s\|$$

$\sigma(t)$ is defined by:

$$\sigma(t) = \sigma_{initial} \left(\frac{\sigma_{final}}{\sigma_{initial}} \right)^{\frac{t}{t_{max}}}$$

where t_{max} is the total number of iterations.

- Step 1: For $t = 1$ to t_{max}
- Step 2: For each input vector x_i $i \in \{1, \dots, N\}$
- Step 3: Find in x_i the most closest output neuron NE_f $f \in \{1, \dots, K\}$ according to the following criterion:

$$NE_f : \|\omega_{f,i} - x_i\| \leq \|\omega_{j,i} - x_i\| \quad \forall j \in \{1, \dots, k\}$$

$\omega_{f,i}$ is the weight associated with the output neuron NE_f .

- Step 4: All neurons in the neighborhood of the winning neuron are adapted:

$$\omega_{j,i}(t+1) = \omega_{j,i}(t) + \alpha(t) \cdot \nu(s, NE_i, t) \cdot (x_i - \omega_{j,i}(t))$$

$\alpha(t)$ is a learning function which decreases slowly and linearly with time ($0 < \alpha(t) < 1$) (see Figure 1.51).

- Step 5: Adjust the learning parameters.

1.5 Prognosis Methods

1.5.1 Prognosis Process

The prediction of the remaining useful life or the forecasting of impending failures allows for maintenance scheduling. This has been the subject of many research works. The theoretical basis will be presented in this section. Many prognosis methods have been developed. Figure 1.52 gives an overview of the different approaches:

They can be classified in three categories:

- model based methods;
- data driven techniques;
- hybrid prognostics approach.

Model-based methods deal with physical representation of the aging process. They do not require historical data but they need an accurate representation of the degradation process. These models can then be exploited by Kalman filters, particle filters, etc. These methods are used, in particular, in mechanics or for electronic components.

Data-driven methods extrapolate the behavior of the system from historical data which come from similar components and working conditions. It can be used to determine the remaining useful life (RUL). They do not require any specific physical knowledge or model. These data, based on the history, are used to extract the parameters which are sensitive to the faults. These later must be able to vary monotonically with aging. The prognosis can be a combination of the extrapolation of these parameters and a diagnosis of the future state from these extrapolated parameters (Figure 1.53).

These methods can be based on belief rule (Zhou et al. [2014]), neural networks (Soualhi et al. [2018], Cui et al. [2017]), Bayesian networks and their derivatives (Bartram and Mahadevan [2013], Medjaher et al. [2012]), on similarity-instance based approach (Khelif et al. [2014]), core methods such as support vector machines (Tao and Zhao [2016], Satishkumar and Sugumaran [2016]).

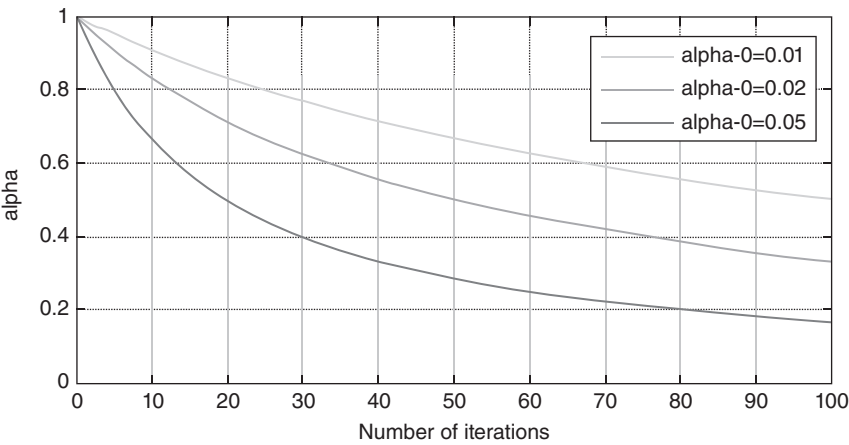


Figure 1.51 Evolution of the learning rate over time

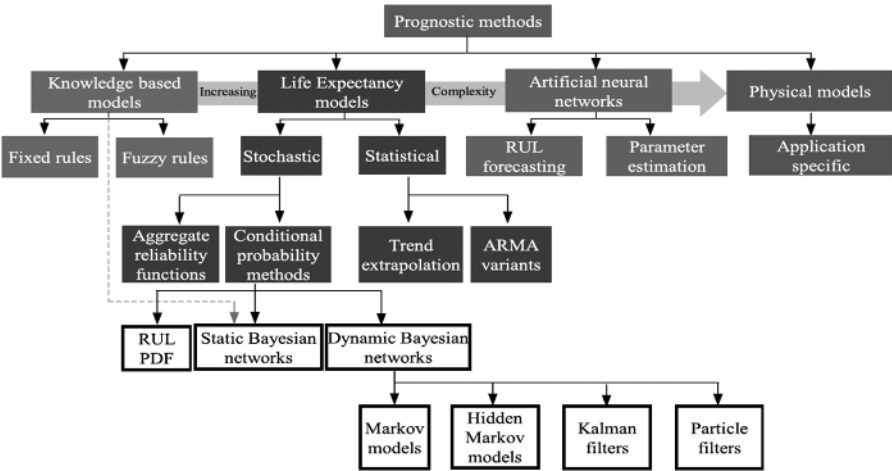


Figure 1.52 Prognosis methods (Soualhi et al. [2020]) with the permission of Elsevier

Hybrid prognostics approach combines model-based and data-driven methods (Figure 1.54).

In the serial hybrid method, the parameters of a physical model are estimated by a data-based method (Mangili [2013]). In the parallel hybrid method, the outputs of a physical model and a data-based method are combined to estimate the RUL (Sachin Kumar et al. [2008]).

We will detail in the next paragraphs some of these techniques:

- the extrapolation of time series (Subsection 1.5.2);

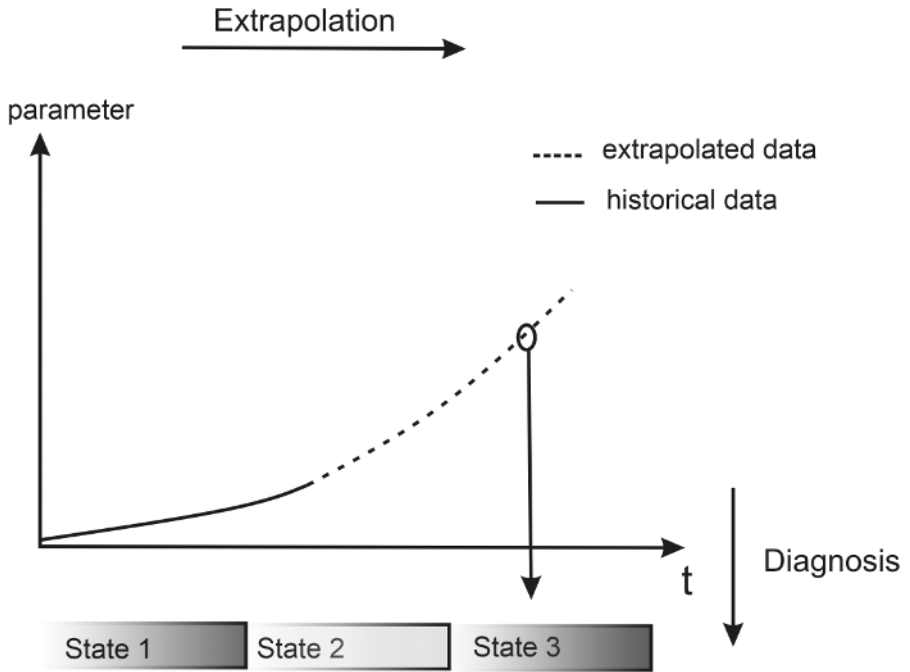


Figure 1.53 Prognosis process

- the Bayesian inference (Subsection 1.5.3);
- the Markov chains (Subsection 1.5.4);
- the rain flow (Subsection 1.5.6).

Diagnostic methods based on supervised classifiers (1.4.2) and Neural Networks (Subsection 1.4.2.3) have been previously described.

1.5.2 Time Series Extrapolation Methods

Time series allows to evaluate the evolution of a variable over the time. The first objective is to find a model that best describes or represents the data. It consists of extracting the signal structure and eliminating the noise. The second objective is to predict the behavior of the process. On the basis of the observations $(X_1, \dots, X_T)$ of the time series at the first T time steps, the value X_{T+h} with $h > 0$ has to be predicted.

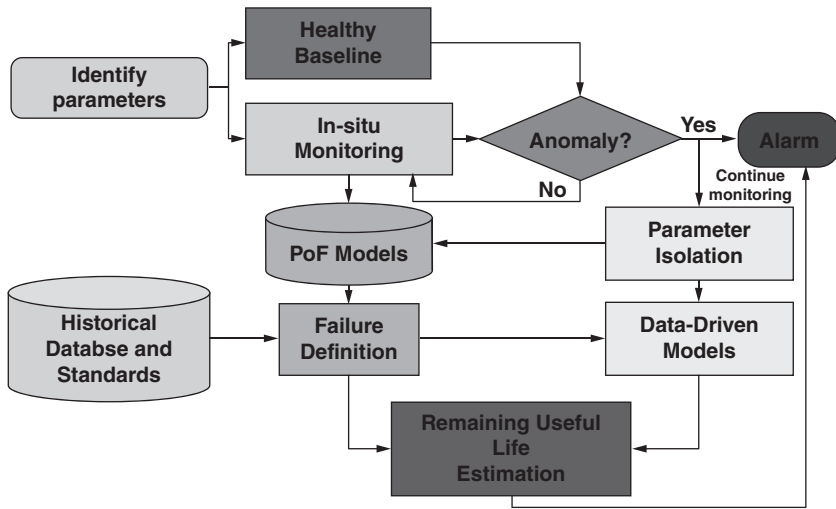


Figure 1.54 Hybrid Prognostics approach. (Pecht and Jaai [2010]) with the permission of Elsevier

(1) Definition

A time series is a series of data points indexed in time order (Wikipedia contributors [2020b]) and Chatfield and Xing [2019]). A time series is a random process such as:

$$X : \Omega \times T \rightarrow \mathbf{R} \quad (1.141)$$

where Ω is a set of events and $T \subset \mathbf{R}$ a discrete set or an interval of $\mathbf{R}$.

Let us note X_t the event at time $t \in T$, a time series (or chronological series). It is a sequence of events in ascending order. The set of measurements of a random variable are the realizations of the random variable X . They are noted:

$$(X_i)_{1 \leq i \leq T} \quad (1.142)$$

Figure 1.55 gives an example of time series from the Time Series Data Library. All the functions are written with MATLAB [2020]. These functions are also available in R Core Team [2020].

A time series can be represented with an additive model and split into three terms:

$$X_t = m_t + S_t + \varepsilon_t \quad (1.143)$$

where:

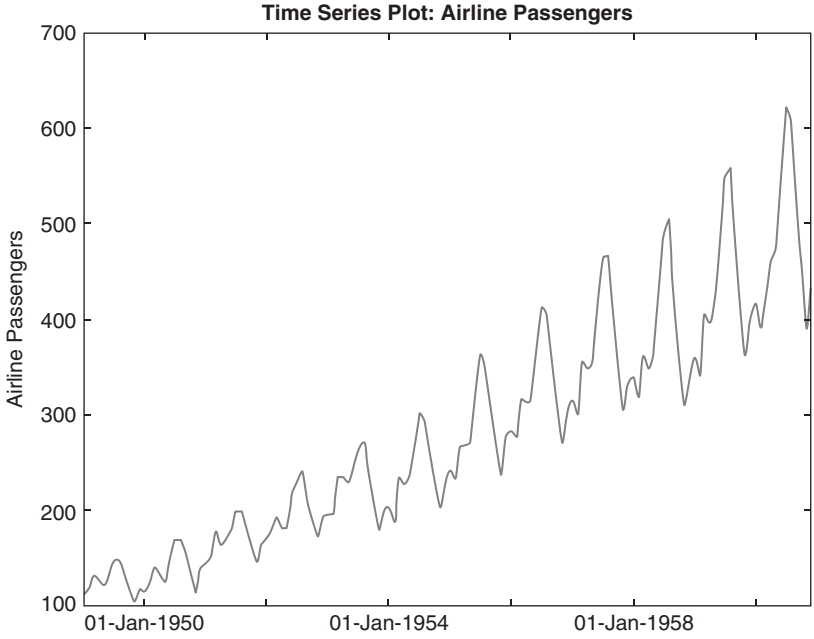


Figure 1.55 Example of time series with monthly counts of airline passengers (Hyndman [2005])

- m_t a deterministic function (called the trend) which describes the “long term” behavior such linear, exponential, logarithmic increase or decrease ... ;
- S_t is a periodic deterministic function (called seasonality) of period m with $\sum_{i=1}^m S_{t+i} = 0, \forall t \in T$;
- ε_t is a random variation in the series and it could be autocorrelated ($cov(\varepsilon_t, \varepsilon_{t+h}) \neq 0$).

Next, Figure 1.56 gives the linear trend plus a seasonal component of the previous time series (Hyndman [2005]).

The time series is stationary if:

$$F_X(X_{t_1}, X_{t_2}, \dots, X_{t_n}) = F_X(X_{t_1+h}, X_{t_2+h}, \dots, X_{t_n+h}), \quad \forall (t_1, \dots, t_n) \in T^n, \forall h \in N \quad (1.144)$$

where $F_X(X_{t_1}, X_{t_2}, \dots, X_{t_n})$ is the cumulative distribution function.

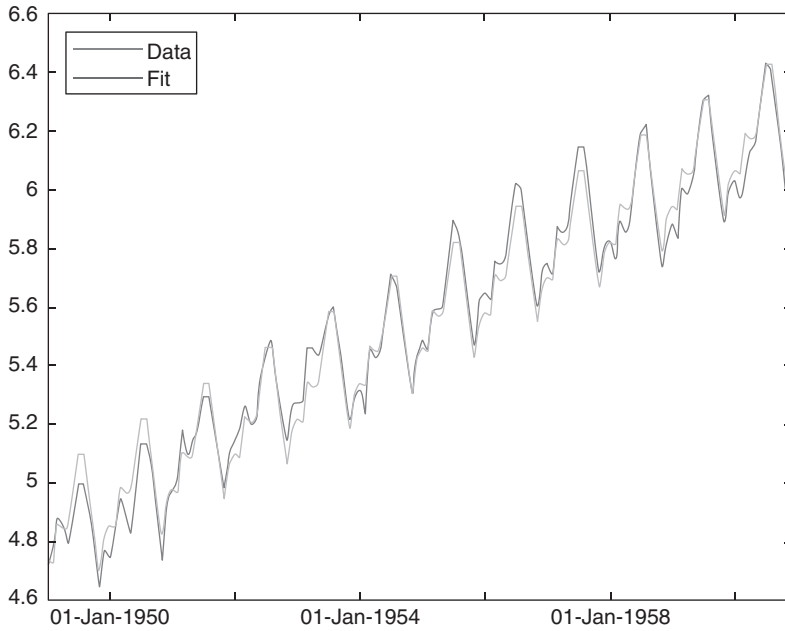


Figure 1.56 Example of the trend and seasonability of a time series (MATLAB [2020])

The time series is weakly stationary if:

$$\begin{aligned} E(X_t) &= \mu \\ \text{cov}(X_t, X_{t+h}) &= \nu_k \end{aligned} \quad (1.145)$$

where μ is constant and ν_k is independent of the time t .

To study a time series, it is necessary to:

1. Write a statistical model (eliminate the trend, periodicity, and test if it is a stationary model);
2. Estimate the parameters of the model;
3. Perform statistical tests to validate/reject the hypotheses;
4. Predict the series of events $(X_{1+T}, X_{2+T}, \dots, X_{n+T})$ from $(X_1, X_2, \dots, X_n)$.

(2) Estimation of the trend (Weber [2001])

(2)1 Time series in the absence of seasonality The objective is to discriminate the trend from the time series. **First approach:** the trend can be evaluated by assuming that it is a linear combination of known functions:

$$m_t = \sum_{i=1}^n \alpha_i m_t^{(i)} \quad (1.146)$$

where α_i are unknown coefficients and the $m_t^{(i)}$ are time functions with a known structure.

1. linear trend: $m_t = \alpha_0 + \alpha_1 t$;
2. polynomial trend: $m_t = \alpha_0 + \alpha_1 t + \dots + \alpha_n t^n$;
3. exponential trend: $m_t = c_0 + c_1 \alpha^t$;
4. Gompertz's trend: $m_t = c_0 \cdot e^{c_1 \cdot e^{c_2 \cdot t}}$;
5. Gompertz's trend: $m_t = \frac{1}{c_0 + c_1 e^{c_2 \cdot t}}$;
6. or a combination of these different functions.

The unknown coefficients can be determined by the method of least squares.

Second approach: the trend can also be estimated by moving average filtering. The moving average is defined by the operator:

$$Y_t = \sum_{i=-m_1}^{m_2} \theta_i Y_{t+i} \text{ with } (m_1, m_2) \in \mathbf{N} \times \mathbf{N} \text{ and } \theta_i \in \mathbf{R} \quad (1.147)$$

It is often given by:

$$Y_t = \frac{1}{2l+1} \sum_{i=-l}^l Y_{t+i} \quad (1.148)$$

The moving average is a non-parametric estimator of trend, that is trend has no a priori structure.

Figure 1.57 gives an example of the output from the moving average of a time series.

Third approach: to eliminate the k_{th} order polynomial trend of a time series, a differentiation of order k can be made. The differentiation operator is defined:

$$\Delta Y_t = Y_t - Y_{t-1} \quad (1.149)$$

and for the k_{th} order differentiation operator is:

$$\Delta^k Y_t = \Delta(\Delta^{k-1} Y_t) \quad (1.150)$$

(2)2 Time series with trend and seasonality The previous methods can be extended in the case of trend with seasonality. In the first approach, the time series can be represented by:

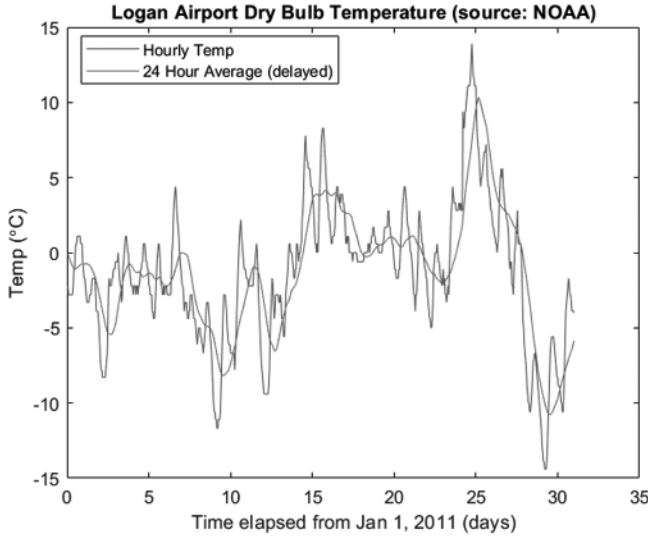


Figure 1.57 Example of a time series moving average filtering (MATLAB [2020])

$$X_t = \sum_{i=1}^n \alpha_i m_t^{(i)} + \sum_{j=1}^p \beta_j s_t^{(j)} + \varepsilon_t \text{ with } E(\varepsilon_t) = 0 \quad (1.151)$$

where $s_t^{(j)}$ are periodic time functions with a known structure.

(2)3 Auto-Regressive Models Many time series can be modeled by auto-regressive process such as:

- Moving average process;
- Auto-Regressive-Integrated-Moving Average process;
- Auto-Regressive-Integrated-Moving Average;
- ...

An auto-regressive model is defined by the equation 1.152.

$$X_t = \sum_{r=1}^q \Phi_r X_{t-r} + \varepsilon_t \quad (1.152)$$

where $\{\varepsilon_t\}$ are uncorrelated random variables following a Normal law ($\varepsilon_t \sim N(0, \sigma^2)$) and Φ_r are constant parameters.

Let us consider some of the auto-regressive processes:

MA : Mean average (eq 1.153).

$$X_t = \sum_{s=0}^q \theta_s \varepsilon_{t-s} \quad (1.153)$$

where $\{\varepsilon_t\}$ are uncorrelated random variables following a Normal law ($\varepsilon_t \sim N(0, \sigma^2)$) and θ_i are constant parameters with $\theta_0 = 1$.

ARMA - Auto-regressive moving average process (1.154).

$$X_t = \sum_{r=1}^p \phi_r X_{t-r} + \sum_{s=0}^q \theta_s \varepsilon_{t-s} \quad (1.154)$$

where $\{\varepsilon_t\}$ is a white noise. This process should be stationary for appropriate θ and ε .

ARIMA - Auto-regressive integrated moving average Average. If X_t is not stationary, we can consider $Y_t = \nabla X_t = X_t - X_{t-1}$ or $Y_t = \nabla^2 X_t = X_t - 2X_{t-1} + X_{t-2}$ or an higher order differentiation.

If the differenced process is a stationary process, X_t is said to be an auto-regressive integrated moving average process.

Figure 1.58 gives an example of an ARIMA model of a scalar time series with linear trend.

(2)4 Other approaches Many other methods can be used to predict the behavior of time series such as Kalman filters (paragraph 1.2.3.1) with state model, neural network (paragraph 1.4.2.3). They also can be associated with clustering techniques (Maharaj et al. [2019]).

1.5.3 Bayesian Inference

Bayesian networks, initiated by Judea Pearl in the 1980s, are among the models of probabilistic analysis. They propose a mathematical formalism and solid theoretical bases for the modeling of complex systems. A Bayesian network is made up of two components:

- A graphic component which consists of a directed graph without circuit. The nodes represent the relevant variables of the domain and the arcs represent the dependency relationships between the variables.
- A numerical component which consists of a set of conditional probability distributions of each node in the context of its parents.

The graph is always directed and acyclic. Directed arcs represent a direct dependency relationship (most of the time this is causation). Thus an arc going from the variable X to the variable Y will express the fact that Y depends

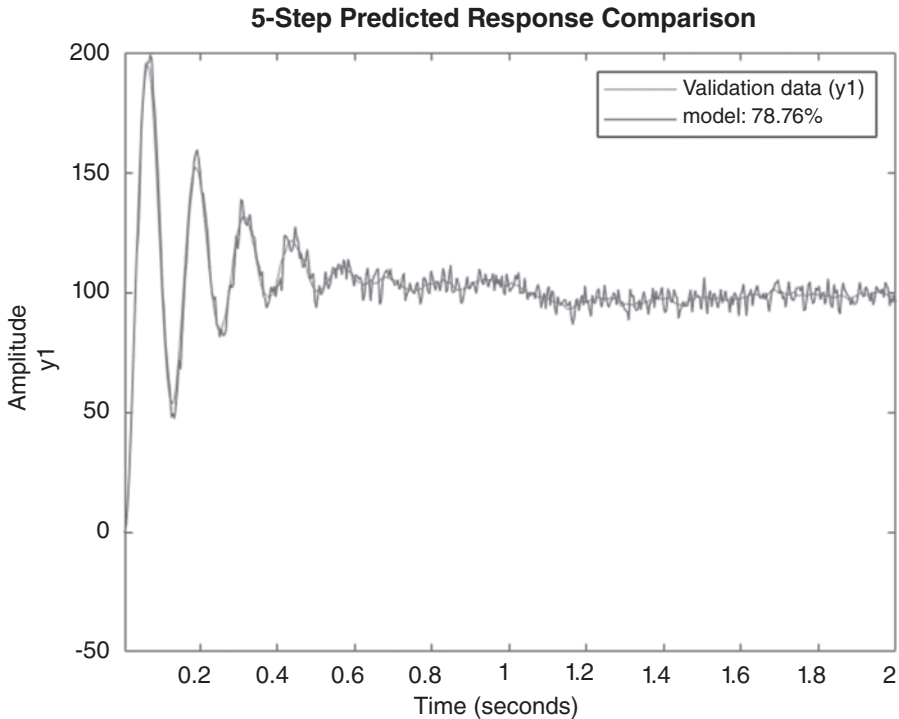


Figure 1.58 Example of ARIMA model (MATLAB [2020])

directly on X . The absence of an arc only informs about the nonexistence of a direct dependence. The parameters express the weights given to these relations and are the conditional probabilities of the variables knowing their parents (example: $P(Y|X)$ or the a priori probabilities if the variable has no parents).

It is possible to achieve efficient classifiers thanks to Bayesian networks (Friedman [1989]; Langley et al. [1992]; Pernkopf [2005]). A Bayesian classifier of a problem with p observations (variables) has the particularity of having $p + 1$ nodes. All Bayesian classifiers model the membership of a class by a discrete node called “class node,” noted Ω , Ω is a multinomial discrete node with k modalities, where k represents the number of classes of the problem ($\Omega_1, \Omega_2, \dots, \Omega_k$). This class node does not have a parent node. The other nodes, p -numbered, represent the observations of the classification problem and are noted O_i . There are two main types of Bayesian classifiers: naive Bayesian network naive Bayesian network augmented by a tree.

The naive Bayesian network has the simplest structure. Also called the Bayes classifier, it is called naive because it makes the very strong assumption that

each observation is, conditionally on the class, independent of the others. The observation independence assumption makes it easy to write the posterior probability of each class as shown in the following equation:

$$P(\Omega_i|O) = P(\Omega_i) \prod_{j=1}^N P(O_j|\Omega_i) \quad (1.155)$$

N is the number of observations.

The structure of a naive Bayesian network can be represented as in Figure 1.59:

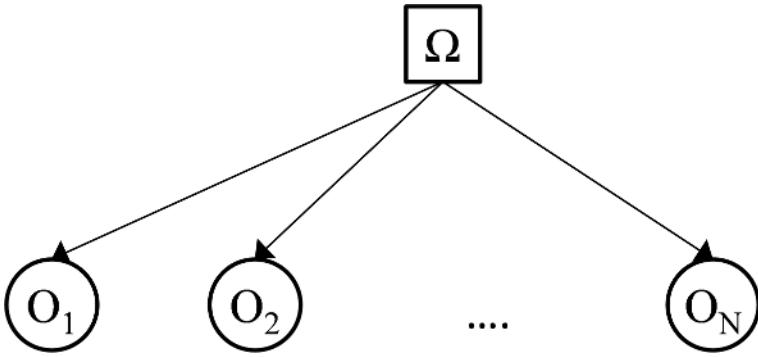


Figure 1.59 Naive Bayesian network

In order to improve the performance of the naive Bayesian network in the case of not completely independent observations, Friedman [1989] suggests adding arcs between the different descriptive observations of the naive classifier. The algorithm starts from a naive Bayesian network and adds an arc between the observations that share the most important mutual information. But, in order to respect the topology of the tree, the algorithm prohibits each node from having more than two parents (i.e. one parent in addition to the class node). Thus, the solution proposed by Friedman [1989] results in a complete structure, that is all nodes are taken into account in the model. The structure of an augmented naive Bayesian network can be represented as in Figure 1.60:

1.5.4 Markov Chain

In this model, a series of random states $X = \{X_t, X_{t-1}, \dots, X_0\}$ in a finite state space E is a Markov chain if the following property is satisfied (Markov

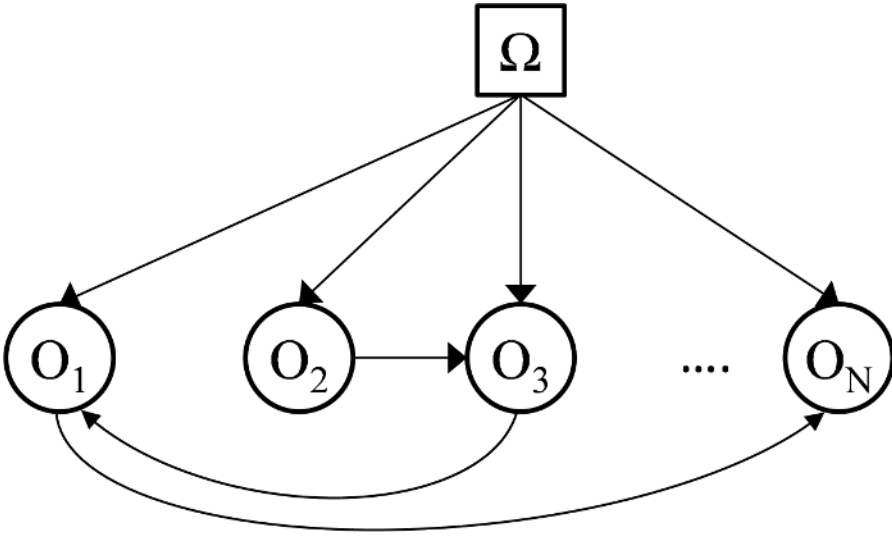


Figure 1.60 Augmented naive Bayesian network

property):

$$P(X_t | X_0, \dots, X_{t-1}) = P(X_t = j | X_{t-1} = i) \quad \text{for all } i, j \in E \quad (1.156)$$

This definition generalizes the notion of a deterministic dynamic system: the probability distribution of the present state X_t depends only on the immediately past state X_{t-1} . It follows that a Markov chain is a sequence of states X which is fully characterized by:

The initial state probability $\pi = (\pi_i)$

$$\pi_i = P(X_0 = i) \quad \text{for all } i \in E \quad (1.157)$$

The transition matrix $A = (A_{ij})$

$$A_{ij} = P(X_t = j | X_{t-1} = i) \quad \text{for all } i, j \in E \quad (1.158)$$

Therefore, the probability distribution of the Markov chain in a sequence of states X , of initial state probability π and a transition matrix A , is given by:

$$\left\{ \begin{aligned} P(X_0, \dots, X_{t-1}, X_t) &= P(X_t | X_0, \dots, X_{t-1}) P(X_0, \dots, X_{t-1}) \\ &= A_{ij}^t P(X_0, \dots, X_{t-1}) \\ &= \pi_i^0 A_{ij}^0, \dots, A_{ij}^{t-1}, A_{ij}^t \end{aligned} \right. \quad (1.159)$$

1.5.5 Hidden Markov Models

In this model, we do not directly observe the sequence X_t , but we have observations o_t with values in a finite observation space O , or in $\mathbb{R}^d$. We suppose that conditionally to the states X_t , observations o_t are mutually independent, and each observation o_t depends only on the state X_t at the same time. This property is expressed as follows:

$$P(o_0, \dots, o_T | X_0, \dots, X_T) = \prod_{t=0}^T P(o_t | X_t) \quad (1.160)$$

It follows that a hidden Markov model for (X_t, o_t) is entirely characterized by:

- The initial state probability: $\pi_i = P(X_0 = i)$ for each $i \in E$
- The transition matrix: $A = (A_{ij})$ with $A_{ij} = P(X_{t+1} = j | X_t = i)$ for each $i, j \in E$
- The observation probability: $b = (b_0, \dots, b_t)$ with $b_t = P(o_t | X_t)$

A local datum is therefore sufficient (the transition probabilities between two successive instants, and the emission probabilities at a given time) to characterize a hidden Markov model globally.

The probability distribution of (X_t, o_t) , with an initial state probability π , a transition matrix A and observation probabilities b is given by:

$$P(X_0 = i, \dots, X_t = j, o_0, \dots, o_t) = \pi_i^0 A_{ij}^0 \dots A_{ij}^t b_0 \dots b_t \quad (1.161)$$

We denote by $\lambda = (\pi, A, b)$ the characteristic parameters of a hidden Markov model. We are interested in the following three problems:

1. Evaluate the model λ : This involves efficiently calculating the probability distribution of the series of observations $(o_0, \dots, o_t)$ (or likelihood function) according to the parameters of the model.
2. Estimate the state of the observation series: Given a series of observations $(o_0, \dots, o_T)$, the objective is the estimation of the present state X_T , or the estimation of an intermediate state X_t for $t = 0, \dots, T$, or to estimate globally the sequence of states $(X_0, \dots, X_T)$, for a given model λ . The answer to this problem is provided by the forward and backward equations of Baum which allow to calculate the conditional probability distribution of state X_T knowing the observations $(o_0, \dots, o_T)$. The answer to the second problem is provided by a dynamic programming algorithm, the Viterbi algorithm, which maximizes the conditional probability distribution of the sequence of states $(X_0, X_1, \dots, X_T)$.

3. Identify the model λ : Given a series of observations $(o_0, \dots, o_T)$, it is a question of calculating the maximum likelihood estimator for the unknown parameters of the model. The answer to this problem is provided by the Baum–Welsh re-estimation formulas, which define an iterative algorithm to maximize the likelihood function.

To justify the introduction of Baum’s forward and backward equations, we start by presenting a first method to calculate the probability distribution of the observations $(o_0, \dots, o_T)$.

The probability distribution of the observations $(o_0, \dots, o_T)$ is given by:

$$P(o_0, \dots, o_T) = \sum_{i,j \in E} \pi_i^0 A_{ij}^0 \dots A_{ij}^T b_0 \dots b_T \quad (1.162)$$

Equation 1.161 is used to calculate the probability distribution of the observations $(o_0, \dots, o_T)$. We obtain:

$$\begin{aligned} P(o_0, \dots, o_T) &= \sum_{\substack{i,j \\ \in E}} P(X_0, \dots, X_T | o_0, \dots, o_T) \\ &= \sum_{i,j \in E} \pi_i^0 A_{ij}^0 \dots A_{ij}^T b_0 \dots b_T \end{aligned} \quad (1.163)$$

This method provides a first expression for the conditional probability distribution of the state’s sequence $(X_0, \dots, X_T)$ knowing the observations $(o_0, \dots, o_T)$:

$$P(X_0, \dots, X_T | o_0, \dots, o_T) = \frac{\pi_i^0 A_{ij}^0 \dots A_{ij}^T b_0 \dots b_T}{\sum_{i,j \in E} \pi_i^0 A_{ij}^0 \dots A_{ij}^T b_0 \dots b_T} \quad (1.164)$$

And for the likelihood function of the model (obtained by using the series of observations $(O = o_0, \dots, o_T)$):

$$P(O/\lambda) = L_T = \sum_{i,j \in E} \pi_i^0 A_{ij}^0 \dots A_{ij}^T b_0 \dots b_T \quad (1.165)$$

Remark:

The number of operations necessary to calculate the probability distribution of the observations $(o_0, \dots, o_T)$ from the given formulas (1.141) is considerable. For each possible trajectory from a state (i) to a state (j) of the Markov chain, we must perform the product of $2(T+1)$ terms and there are $|E|^{T+1}$ different possible trajectories. The total number of elementary operations (additions and

multiplications) to be performed is therefore of the order of: $2(T+1) |E|^{T+1}$. This number grows exponentially with the number T of observations.

• **Forward equation:** We define the forward variable $\alpha_t(i)$ by:

$$\alpha_t(i) = P(o_0, \dots, o_t, X_t = i | \lambda) = P(X_t = i | o_0, \dots, o_t) L_t \text{ for each } i \in E \quad (1.166)$$

The sequence α_t satisfies the following recurrent equation:

$$\alpha_{t+1}(j) = b_j^{o_{t+1}} \sum_{i \in E} A_{ij} \alpha_t(i) \quad (1.167)$$

with the initial condition: $\alpha_0(i) = \pi_i b_0$ for each $i \in E$

• **Backward equation** Conditionally to $(X_t = i)$, the sequence $X_{t+1}, X_{t+2}, \dots$, is a Markov chain, with an initial probability state π_i and transition matrix A . We then deduce that the probability distribution of the observations $(o_{t+1}, \dots, o_T)$ given $(X_t = i)$ is expressed as follows:

$$P(o_{t+1}, \dots, o_T, X_t = i | \lambda) = \sum_{i, j \in E} A_{ij}^{t+1} \dots A_{ij}^T b_{t+1} \dots b_T \quad (1.168)$$

And we define the backward variable as the likelihood function of the model (obtained by using the series of observations $(o_{t+1}, \dots, o_T)$).

$$\beta_t(i) = \sum_{i, j \in E} A_{ij}^{t+1} \dots A_{ij}^T b_{t+1} \dots b_T \quad (1.169)$$

And especially:

$$\beta_{T-1}(i) = \sum_{j \in E} A_{ij}^T b_T \quad (1.170)$$

$$\beta_t(i) = \sum_{j \in E} A_{ij}^t b_{t+1} \beta_{t+1}(j) \quad (1.171)$$

with the initial condition: $\beta_T(i) = 1$ for each $i \in E$.

• **Viterbi algorithm (dynamic programming)**

It results that the forward and backward variables, studied previously, make it possible to calculate the conditional probability distribution of the present state X_T , or of the state X_t at an intermediate instant, knowing the observations

$(o_0, \dots, o_T)$, defined by:

$$P(X_T = i | o_0, \dots, o_T) = \frac{1}{L_T} \alpha_T(i) \quad (1.172)$$

and

$$P(X_T = i | o_0, \dots, o_T) = \frac{1}{L_T} q_t(i) \quad (1.173)$$

with

$$L_T = \sum_{i \in E} \alpha_T(i) = \sum_{i \in E} \alpha_t(i) \beta_t(i) = \sum_{i \in E} q_t(i) \quad (1.174)$$

We define the posterior maximum estimator, which minimizes the probability of the estimation error given the observations $(o_0, \dots, o_T)$ of the present state by:

$$X_T^{LMAP} = \arg \max_{i \in E} P(X_T = i | o_0, \dots, o_T) = \arg \max_{i \in E} \alpha_T(i) \quad (1.175)$$

and for the state at an intermediate instant by:

$$X_t^{LMAP} = \arg \max_{i \in E} P(X_t = i | o_0, \dots, o_t) = \arg \max_{i \in E} q_t(i) \quad (1.176)$$

However, it can happen that the sequence $(X_0^{LMAP}, \dots, X_T^{LMAP})$ thus generated is inconsistent with the model, in the following way: it can happen that one obtains $X_t^{LMAP} = i$ and $X_{t+1}^{LMAP} = j$ for two successive times, while $A_{ij} = 0$ for this same pair (i, j) . This means that the transition from state i to state j is impossible for the model. For this reason, we instead use another estimator defined by:

$$(X_0^{LMAP}, \dots, X_T^{LMAP}) = \arg \max_{i_0, \dots, i_T \in E} P(X_0, \dots, X_T | o_0, \dots, o_T), \quad (1.177)$$

which minimizes the probability of the estimation error of the sequence of hidden states knowing the observations $(o_0, \dots, o_T)$. The efficient calculation of this estimator is provided by a dynamic programming algorithm, called the Viterbi algorithm.

If the following sequence, $(i_0^*, \dots, i_t^*)$ reaches the maximum of the function:

$$(i_0^*, \dots, i_t^*) \rightarrow P(X_0 = i_0, \dots, X_t = i_t | o_0, \dots, o_t) \quad (1.178)$$

Then necessarily i_0^* reaches the maximum of the function:

$$i \rightarrow \max_{i_0, \dots, i_{t-1}} P(X_0 = i_0, \dots, X_{t-1} = i_{t-1}, X_t = i | o_0, \dots, o_t) \quad (1.179)$$

and we therefore define:

$$V_t^i \rightarrow \max_{i_0, \dots, i_{t-1}} P(X_0 = i_0, \dots, X_{t-1} = i_{t-1}, X_t = i | o_0, \dots, o_t) L_t \quad (1.180)$$

The sequence V_t checks the following recurrence:

$$V_{t+1}^j = b_j^{o_{t+1}} \max_{i \in E} [A_{ij} V_t^i] \quad (1.181)$$

with the initial condition: $V_0^i = \pi_i b_i^{o_0}$

We deduce that the optimal trajectory ends up in the state : $X_T^{LMAP} = \arg \max_{i \in E} V_T^i$.

• Baum–Welch re-estimation formulas

It follows from eq. 1.165 that the likelihood function of the model $\lambda = (\pi, A, b)$ admits the following expression: $L_T = \sum_{i,j \in E} \pi_i^0 A_{ij}^0 \dots A_{ij}^T b_0 \dots b_T$ and we propose to study an iterative algorithm to maximize the likelihood function L_T with respect to the parameters $\lambda = (\pi, A, b)$ of the model.

Let $\lambda' = (\pi', A', b')$ be another model, for which we have already evaluated the likelihood function:

$$L'_T = \sum_{i,j \in E} \pi_i'^0 A_{ij}'^0 \dots A_{ij}'^T b'_0 \dots b'_T \quad (1.182)$$

The likelihood ratio between the model λ and the model λ' can be written as:

$$\begin{aligned} \frac{L_T}{L'_T} &= \frac{\sum_{i,j \in E} \pi_i^0 A_{ij}^0 \dots A_{ij}^T b_0 \dots b_T}{\sum_{i,j \in E} \pi_i'^0 A_{ij}'^0 \dots A_{ij}'^T b'_0 \dots b'_T} \\ &= \frac{\sum_{i,j \in E} \pi_i'^0 A_{ij}'^0 \dots A_{ij}'^T b'_0 \dots b'_T \frac{\pi_i^0 A_{ij}^0 \dots A_{ij}^T b_0 \dots b_T}{\pi_i'^0 A_{ij}'^0 \dots A_{ij}'^T b'_0 \dots b'_T}}{\sum_{i,j \in E} \pi_i'^0 A_{ij}'^0 \dots A_{ij}'^T b'_0 \dots b'_T} \end{aligned} \quad (1.183)$$

the log likelihood checks

$$\begin{aligned} Q_T &= \log \frac{L_T}{L'_T} = \log \frac{\sum_{i,j \in E} \pi_i^0 A_{ij}^0 \dots A_{ij}^T b_0 \dots b_T}{\sum_{i,j \in E} \pi_i'^0 A_{ij}'^0 \dots A_{ij}'^T b'_0 \dots b'_T} \\ &\geq \frac{\sum_{i,j \in E} \pi_i'^0 A_{ij}'^0 \dots A_{ij}'^T b'_0 \dots b'_T \log \left[\frac{\pi_i^0 A_{ij}^0 \dots A_{ij}^T b_0 \dots b_T}{\pi_i'^0 A_{ij}'^0 \dots A_{ij}'^T b'_0 \dots b'_T} \right]}{\sum_{i,j \in E} \pi_i'^0 A_{ij}'^0 \dots A_{ij}'^T b'_0 \dots b'_T} \end{aligned} \quad (1.184)$$

where the quantity Q_T vanishes when the model λ coincides with the model λ' .

Maximizing Q_T with respect to the parameters (π, A, b) of the model λ therefore guarantees that the likelihood L'_T of the model λ' where the maximum of Q_T is reached will be greater than the likelihood L'_T of the model λ' . The Baum–Welch re-estimation formulas make it possible to explicitly find the parameters (π', A', b') of the model λ' as a function of the parameters (π', A', b') of the

model λ' . By repeating this procedure, we construct a series of increasing likelihood models, and ideally this series converges to a model which reaches the maximum of the likelihood function.

The iterative algorithm for the maximum likelihood estimation of the model parameters in view of the observations $(o_0, \dots, o_T)$, is given by the explicit re-estimation formulas:

$$\pi'_i = \frac{\alpha'_0(i) \beta'_0(i)}{L'_T} \quad (1.185)$$

$$A'_{i,j} = A_{i,j} \frac{\sum_{t=1}^T \alpha'_{t-1}(i) b'_t \beta'_{t-1}(j)}{\sum_{t=1}^T \alpha'_{t-1}(i) \beta'_{t-1}(i)} \quad (1.186)$$

$$b_t = \frac{\sum_{i=0 \cap y_i = \ell}^T \alpha'_t(i) \beta'_t(i)}{\sum_{i=0}^T \alpha'_t(i) \beta'_t(j)} \quad (1.187)$$

For each $i, j \in E$ where the two sequences α'_t and β'_t are the respective solutions of the forward and backward equations for the values (π', A', b') of the parameters.

1.5.6 Rainflow

The rainflow algorithm (also called Pagoda roof method) was introduced in T. Endo and M. Matsuishi in 1968 (Matsuichi and Endo [1968]). This method is often used to study material fatigue in association with the Miner's rule (Miner [1945]).

This algorithm extracts, from a loadtime profile/stress, the number of reversals. These last are the local minima and maxima where the load changes sign. Thus, it builds half-cycles. The rainflow algorithm relies to the flow of water down a series of pagoda roofs.

Let us consider a stress profile with peaks and valleys. The algorithm is detailed in Ariduru [2004]:

1. Extract a representative profile with peaks and valleys. This one begins with the point of the largest value and ends with a point with the same stress.
2. Consider this profile as rigid sheet. Turn the Figure clockwise 90° with the origin of time at the top. Thus we have a vertical time axis.

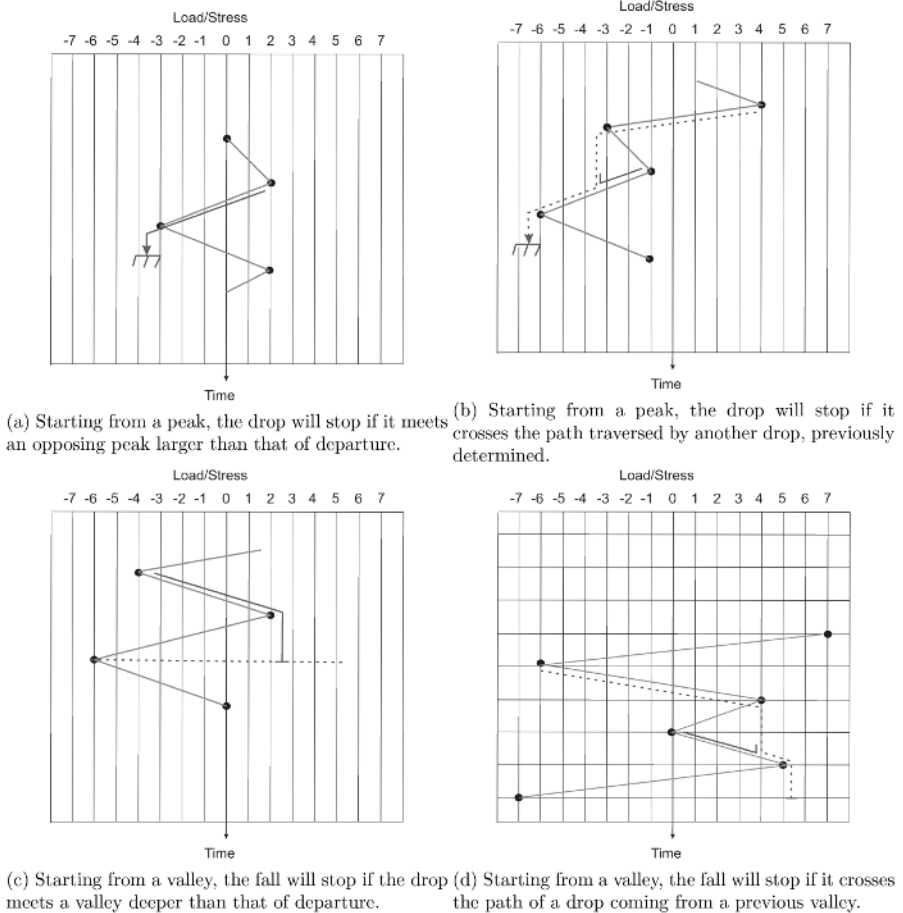


Figure 1.61 Different cases for the rainflow algorithm

3. Let us imagine a flow of rain starting at each successive extremum point.
4. If we start from a peak
 - (a) the drop will stop if it meets an opposing peak larger than that of departure (Figure 1.61a).
 - (b) the drop will stop if it crosses the path traversed by another drop, previously determined (Figure 1.61b).
 - (c) the drop can fall on another roof and continue to slip according to rules 4.a and 4.b.
5. if we start from a valley
 - (a) the fall will stop if the drop meets a valley deeper than that of departure (Figure 1.61c).

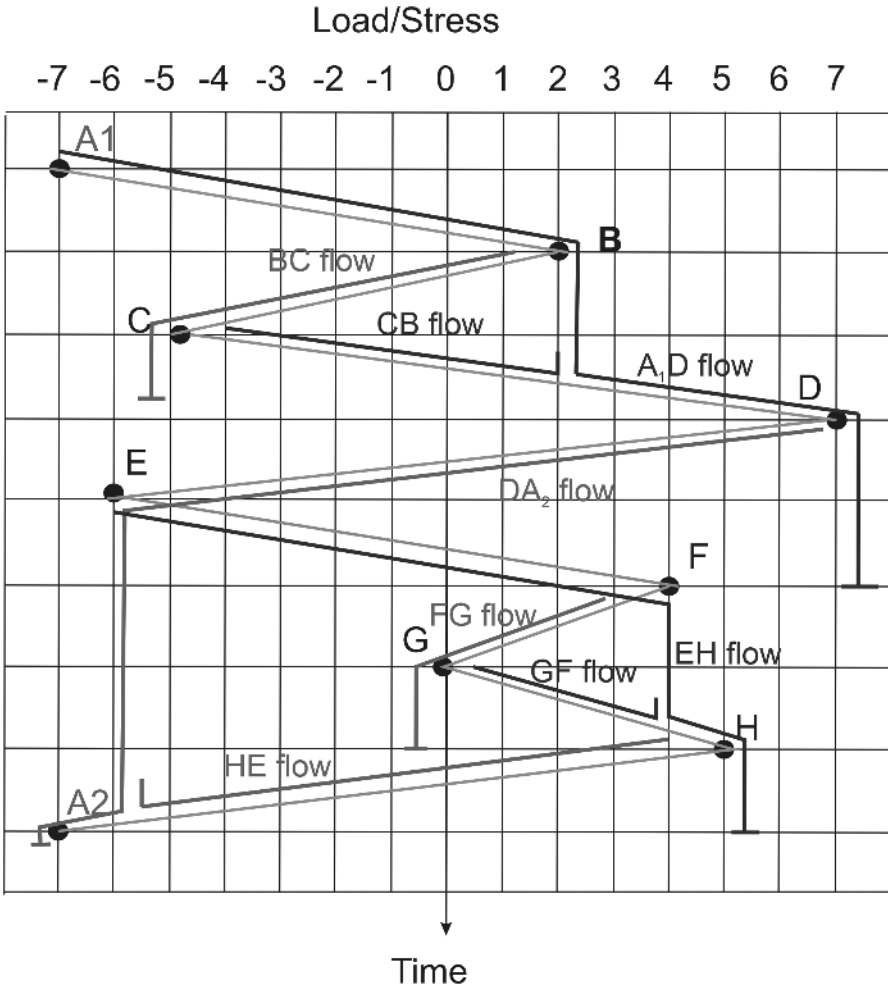


Figure 1.62 Example of the rainflow algorithm (Lee and Tjhung [2012])

- (b) the fall will stop if it crosses the path of a drop coming from a previous valley (Figure 1.61d).
- (c) the drop can fall on another roof and continue according to rules 5.a and 5.b.
- 6. Repeat from 3 until the end of the profile
- 7. Pair up half-cycles of identical magnitude (but opposite sense) to count the number of complete cycles. Typically, there are some residual half-cycles.

Let us consider the following example (Figure 1.62):
The different steps of the algorithm are described below:

1. Take point A1 as the starting point with the greater amplitude (largest peak).
The end point is A2 with the same magnitude.
2. Identify the greater reversal A1 to D. We stop at the smallest valley D.
3. Identify the greater reversal D to A2. It starts from A and finishes at the end of the profile A2 (which is the same point as A1 as we describe a full cycle).
4. **Within the cycle A1–D**, identify the reversal BC that begins at B and ends at C (D is a larger maximum than B – case 4.a).
5. **Within the cycle A1–D**, identify the reversal CB that begins at C and ends at B (The flow meets a previous one at B – case 5.b).
6. **Within the cycle D–A2** (where A1 and A2 are merged), identify the reversal EH that starts at E and ends in the roof H.
7. **Within the cycle D–A2**, identify the reversal HE that begins at H and ends at E (The flow meets a previous one at E – case 5.b).
8. **Within the cycle D–A2**, identify the reversal FG that begins at F and ends at G (H is a larger maximum than F – case 5.a).
9. **Within the cycle D–A2**, identify the reversal GF that begins at G and ends at F (The flow meets a previous one at F – case 4.b).

Thus, we obtain the following reversals (half cycle) on Table 1.23 (Lee and Tjhung [2012]):

Table 1.23 Rainflow reversals

No of reversals	From point	To point	From load	To load	Range	Mean
1	A	D	–7	7	14	0
1	D	A	7	–7	14	0
1	B	C	2	–5	7	–1.5
1	C	B	–5	2	7	–1.5
1	E	H	–6	5	11	–0.5
1	H	E	5	–6	11	–0.5
1	F	G	4	0	4	2
1	G	F	0	4	4	2

Table 1.24 Rainflow cycles

No of reversals	From point	To point	From load	To load	Range	Mean
1	A	D	−7	7	14	0
1	B	C	2	−5	7	−1.5
1	E	H	−6	5	11	−0.5
1	F	G	4	0	4	2

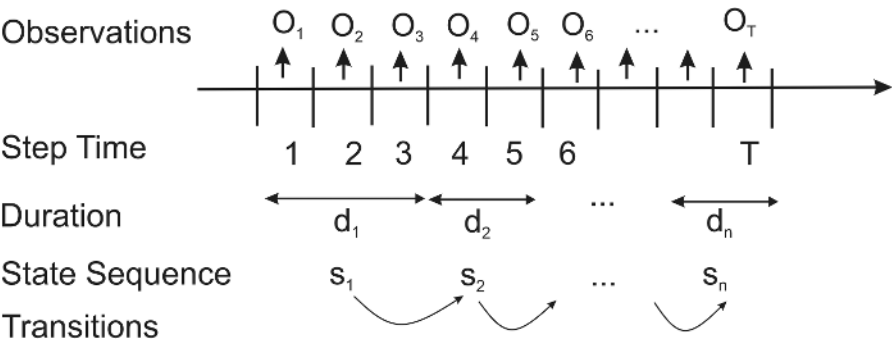


Figure 1.63 Hidden semi-Markov models

After pairing up these reversals, the cycles are given Table 1.24) (Lee and Tjhung [2012]):

1.5.6.1 Hidden Semi-Markov Models

Hidden semi-Markov models (Levy [1954]) are a generalization of hidden Markov model. In this case, each state can emit a sequence of observations. They allow a variable duration for each state and the number of observations emitted in a given state is determined by the duration spent in this state.

The detailed description of Hidden Semi Markov Model is beyond the scope of this book. The reader will find a very good description in Yu [2010].

References

Chris Aldrich and Lidia Auret. *Unsupervised Process Monitoring and Fault Diagnosis with Machine Learning Methods*. Springer Publishing Company, Incorporated, 2013. ISBN 1447151844.

- A. Amrane, A. Larabi, and A. Aitouche. Fault detection and isolation based on nonlinear analytical redundancy applied to an induction machine. In *2017 6th International Conference on Systems and Control (ICSC)*, pages 255–260, 2017.
- Jérôme Antoni. Cyclostationarity by examples. *Mechanical Systems and Signal Processing*, 23(4):987 – 1036, 2009. ISSN 0888-3270. doi: <https://doi.org/10.1016/j.ymssp.2008.10.010>. URL <http://www.sciencedirect.com/science/article/pii/S0888327008002690>.
- Seçyl Ariduru. Fatigue life calculation by rainflow cycling method. Master thesis, Graduate school of natural and applied sciences of middle east technical university, 2004. URL http://www.vibrationdata.com/tutorials_alt/Ariduru_rainflow.pdf.
- Georges Asch. *Les capteurs en instrumentation industrielle (8e édition)*. Technique et Ingénierie. Dunod, dunod edition, 2017. ISBN 2-10-076020-3.
- Francois Auger and C. Doncarli. Quelques commentaires sur des représentations temps-fréquence proposées récemment. *Traitement du Signal*, 9:3–25, 01 1992.
- Kevin Balanda and H. Macgillivray. Kurtosis: A critical review. *The American Statistician*, 42:111–119, 05 1988. doi: 10.1080/00031305.1988.10475539.
- David Barber. *Bayesian Reasoning and Machine Learning*. Cambridge University Press, 2011.
- G. Bartram and S. Mahadevan. Dynamic Bayesian networks for prognosis. *PHM 2013 - Proceedings of the Annual Conference of the Prognostics and Health Management Society 2013*, pages 167–184, January 2013.
- Amy S. Beavers, John W. Lounsbury, Jennifer K. Richards, Schuyler W. Huck, Gary J. Skolits, and Shelley L. Esquivel. Practical considerations for using exploratory factor analysis in educational research. *PARE*, 18(6), 2013. URL <https://scholarworks.umass.edu/pare/vol18/iss1/6/>.
- Mohamed Ben Marzoug. *Apport des cycles d'usage représentatifs synthétiques pour les études de vieillissement de batteries dans les nouveaux usages de mobilité*. PhD thesis, Université Claude Bernard Lyon 1, Villeurbanne, 2020.
- H. Berriri, M. W. Naouar, and I. Slama-Belkhodja. Parity space approach for current sensor fault detection and isolation in electrical systems. In *Eighth International Multi-Conference on Systems, Signals Devices*, pages 1–7, 2011.
- Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, New York, 2007. ISBN 978-0-387-31073-2.
- Frédéric Bonnardot. *Comparaison entre les analyses angulaire et temporelle des signaux vibratoires de machines tournantes. Etude du concept de cyclostationnarité floue*. Theses, Institut National Polytechnique de Grenoble - INPG, December 2004. URL <https://tel.archives-ouvertes.fr/tel-00103067>.
- Abdelhakim Boukar and Nacer Hamzaoui. Evaluation des indicateurs de surveillance par analyse vibratoire : Application aux engrenages et roulements. *International Journal of Innovation and Applied Studies*, 25(2):800–808, 2019.

- ISSN 2028-9324. URL <http://www.ijias.issr-journals.org/abstract.php?article=IJIAS-17-358-09>.
- Babak Nahid-Mobarakeh Breuneval R., Clerc G. and Badr Mansouri.
Classification with automatic detection of unknown classes based on svm and fuzzy mbf: Application to motor diagnosis. *AIMS Electronics and Electrical Engineering*, 2(ElectronEng-02-03-059):59, 2018. ISSN 2578-1588. doi: <http://dx.doi.org/10.3934/ElectrEng.2018.3.59>.
- T. Caliski and J. Harabasz. A dendrite method for cluster analysis. *Communications in Statistics*, 3(1):1–27, 1974. doi: 10.1080/03610927408827101. URL <https://www.tandfonline.com/doi/abs/10.1080/03610927408827101>.
- C. Chatfield and Haipeng Xing. *The analysis of time series: An introduction, sixth edition*. Chapman & Hall/CRC texts in statistical science. CRC Press, 2019. ISBN 978-1-4987-9563-0.
- E. Chow and A. Willsky. Analytical redundancy and the design of robust failure detection systems. *IEEE Transactions on Automatic Control*, 29(7):603–614, 1984.
- G. Clerc and J.-C. Marques. Induction Machine Diagnosis Using Observers. In *Electrical machines diagnosis*, pages 93–129. Wiley edition, September 2011. ISBN 978-1-84821-263-3. URL <https://doi.org/10.1002/9781118601662.ch4>.
- Guy Clerc. *Contribution à l'étude du contacteur statique autoprotégé et de sa stratégie de commande*. PhD thesis, Ecole Centrale de Lyon, Ecully, France, January 1989.
- Leon Cohen. *Time-Frequency Analysis: Theory and Applications*. Prentice-Hall, Inc., USA, 1995. ISBN 0135945321.
- M. Comanescu. A family of sensorless observers with speed estimate for rotor position estimation of im and pmsm drives. In *IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society*, pages 3670–3675, 2012.
- Antoine Cornuéjols, Laurent Miclet, and Vincent BARRA. *Apprentissage artificiel - 3e édition: Deep Learning, Concepts et algorithmes*. Eyrolles, May 2018. URL <https://hal.archives-ouvertes.fr/hal-01829395>.
- Q. Cui, Z. Li, J. Yang, and B. Liang. Rolling bearing fault prognosis using recurrent neural network. In *2017 29th Chinese Control And Decision Conference (CCDC)*, pages 1196–1201, 2017.
- D. L. Davies and D. W. Bouldin. A Cluster Separation Measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-1(2):224–227, 1979.
- Bernard Desgraupes. Clustering indices, 2017. URL <https://cran.r-project.org/web/packages/clusterCrit/vignettes/clusterCrit.pdf>.
- Gérard Dreyfus, J.M. Martinez, M. Samuelides, M. B. Gordon, Fouad Badran, Sylvie Thiria, and L. Herault. *Réseaux de neurones - Méthodologie et applications*. Eyrolles, January 2002. URL <https://hal.archives-ouvertes.fr/hal-01125016>.

- Pierre Dumolard. Les réseaux de neurones. *Hermès*, 23(3):287–288, 1994. ISSN 0046-2497. URL https://www.persee.fr/doc/spgeo_0046-2497_1994_num_23_3_3327.
- J. C. Dunn. Well-separated clusters and optimal fuzzy partitions. *Journal of Cybernetics*, 4(1):95–104, 1974. doi: 10.1080/01969727408546059. URL <https://doi.org/10.1080/01969727408546059>. Citation dunn1974 Publisher: Taylor & Francis tex.eprint: <https://doi.org/10.1080/01969727408546059>.
- M.A. Efroymson. *Multiple Regression Analysis*. Mathematical Methods for Digital Computers. Wiley, ralston, a. and wilf, hs edition, 1960.
- Samuel Eke. *Transformer condition assesment strategy : Outline solutions for aging transformers integrated management*. Theses, Université de Lyon, June 2018. URL <https://tel.archives-ouvertes.fr/tel-02138989>.
- P. Ewert. Use of axial flux in the detection of electrical faults in induction motors. In *2017 International Symposium on Electrical Machines (SME)*, pages 1–6, 2017.
- P. Flandrin. *Time-frequency/time-scale analysis*. ISSN. Elsevier Science, 1998. ISBN 978-0-08-054303-1. URL <https://books.google.fr/books?id=FRwlO2rP5iIC>.
- E. Forgy. Cluster analysis of multivariate data: Efficiency versus interpretability of classification. *Biometrics*, 21(3):768–769, 1965.
- Jerome H. Friedman. Regularized discriminant analysis. *Journal of the American Statistical Association*, 84(405):165–175, 1989.
- J P Gauthier and I A K Kupka. Observability and observer for nonlinear systems. *SIAM J. Control Optim.*, (32):975–994, 1994.
- Pierre-Louis Gonzalez. *L'analyse en composantes principales*, 2020. URL <http://maths.cnam.fr/IMG/pdf/A-C-P-.pdf>.
- Robert Gray, M. *Entropy and Information Theory*. Springer-Verlag, 2013. URL <https://ee.stanford.edu/~gray/it.pdf>.
- L.I. Gudzenko. On periodic nonstationary processes. *Radio Engineering and Electronic Physics*, 4:220–224, 1959.
- S. Hayashi, T. Asakura, and Sheng Zhang. Study of machine fault diagnosis system using neural networks. In *Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02 (Cat. No.02CH37290)*, volume 1, pages 956–961 vol.1, 2002. doi: 10.1109/IJCNN.2002.1005604.
- Xiaofei He, Deng Cai, and Partha Niyogi. Laplacian Score for Feature Selection. In *Proceedings of the 18th International Conference on Neural Information Processing Systems, NIPS'05*, pages 507–514, Cambridge, MA, USA, 2005. MIT Press.
- R.R. Herrera and D.S.L. Gac. *Initiation à l'analyse factorielle des données: Fondements mathématiques et interprétations, cours et exercices corrigés*. Ellipses, 2002. ISBN 9782729811198. URL https://books.google.fr/books?id=X_nYMgEACAAJ.

- F. Hlawatsch and G. F. Boudreaux-Bartels. Linear and quadratic time–frequency signal representations. *IEEE Signal Processing Magazine*, 9(2):21–67, 1992. doi: 10.1109/79.127284.
- H. Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24(6):417–441, 1933. doi: 10.1037/h0071325. URL <https://doi.org/10.1037%2Fh0071325>.
- M. Hwang and J. H. Seinfeld. Observability of nonlinear systems. *Journal of Optimization Theory and Applications*, 10(2):67–77, August 1972. ISSN 1573-2878. doi: 10.1007/BF00934972. URL <https://doi.org/10.1007/BF00934972>.
- Rob Hyndman. Time Series Data Library, 2005. URL <http://www-personal.buseco.monash.edu.au/~hyndman/TSDL/>.
- TAHAR IFTENE and Habib Mahi. Approche de classification par réseaux de kohonen pour Établir des cartes d'occupation du sol de la région d'oran (algérie). *Teledetection*, 3:325–336, 01 2003.
- J. J. A. Moors. The meaning of kurtosis: Darlington reexamined. *The American Statistician*, 40(4):283–284, 1986. doi: 10.1080/00031305.1986.10475415. URL <https://www.tandfonline.com/doi/abs/10.1080/00031305.1986.10475415>.
- A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: A review. *ACM Comput. Surv.*, 31(3):264–323, September 1999. ISSN 0360-0300.
- Simon J. Julier and Jeffrey K. Uhlmann. New extension of the Kalman filter to nonlinear systems. In Ivan Kadar, editor, *Signal Processing, Sensor Fusion, and Target Recognition VI*, volume 3068 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, pages 182–193, July 1997. doi: 10.1117/12.280797.
- Henry F. Kaiser. The application of electronic computers to factor analysis. *Educational and Psychological Measurement*, 20(1):141–151, 1960. doi: 10.1177/001316446002000116. URL <https://doi.org/10.1177/001316446002000116>.
- R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45, March 1960. ISSN 00982202. URL <http://dx.doi.org/10.1115/1.3662552>.
- R. Khelif, S. Malinowski, B. Chebel-Morello, and N. Zerhouni. Rul prediction based on a new similarity-instance based approach. In *2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE)*, pages 2463–2468, 2014.
- Thameur Kidar. *Diagnostic des défauts de fissures d'engrenages par l'analyse cyclostationnaire*. Theses, Université Jean Monnet - Saint-Etienne, March 2015. URL <https://tel.archives-ouvertes.fr/tel-01538338>.
- T. Kohonen. *Self-Organization and Associative Memory: 3rd Edition*. Springer-Verlag, Berlin, Heidelberg, 1989. ISBN 0387513876.

- J. Korbicz. Neural networks and their application in fault detection and diagnosis. In *IFAC Proceedings*, volume 30, pages 367–372, 1997. doi: 10.1016/S1474-6670(17)42429-3.
- A. Lalami and R. Wamkeue. Parity-space approach for time-domain synchronous generator diagnosis. In *2012 25th IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*, pages 1–4, 2012.
- Pat Langley, Wayne Iba, and, and Kevin Thompson. An analysis of bayesian classifiers. In *Proceedings of the Tenth National Conference on Artificial Intelligence*, page 223–228. AAAI Press, 1992. ISBN 0262510634.
- Abdesselam Lebaroud. *Diagnostic des défauts des machines asynchrones : approche signal et système*. PhD thesis, 2007. URL <http://www.theses.fr/2007LYO10288>. Thèse de doctorat dirigée par Clerc, Guy Physique Lyon 1 2007.
- V.L. Lebedev. On random processes having nonstationarity of periodic character. *Nauchnye Doklady Vysshchei Shchekoly. Seria Radiotekhnika i Elektronika*, 2: 32–34, 1959.
- Yung-Li Lee and Tana Tjhung. Chapter 3 - rainflow cycle counting techniques. In Yung-Li Lee, Mark E. Barkey, and Hong-Tae Kang, editors, *Metal Fatigue Analysis Handbook*, pages 89 – 114. Butterworth-Heinemann, Boston, 2012. ISBN 978-0-12-385204-5. doi: <https://doi-org-s.bibliopam.ec-lyon.fr/10.1016/B978-0-12-385204-5.00003-3>. URL <https://www.sciencedirect-com.bibliopam.ec-lyon.fr/science/article/pii/B9780123852045000033>.
- P. Levy. Syst èmes semi-Markoviens à au plus infinité d énombrable d'états possibles. volume 2, page 294, 1954.
- Y. Liu, Z. Q. Zhu, and D. Howe. Instantaneous torque estimation in sensorless direct-torque-controlled brushless dc motors. *IEEE Transactions on Industry Applications*, 42(5):1275–1283, 2006.
- P.J. Loughlin and B. Tacer. Instantaneous frequency and the conditional mean frequency of a signal. *Signal Processing*, 60(2):153 – 162, 1997. ISSN 0165-1684.
- Elizabeth Ann Maharaj, Pierpaolo D'Urso, and Jorge Caído. *Time series clustering and classification*. Computer science and data analysis. CRC Press, Australia, 1st edition, 2019. ISBN 978-1-4987-7321-8. doi: 10.1201/9780429058264.
- Masoud Malekzadeh, Mustafa Gul, Il-Bum Kwon, and Necati Catbas. An integrated approach for structural health monitoring using an in-house built fiber optic system and non-parametric data analysis. *Smart Structures and Systems*, 14:917–942, 11 2014. doi: 10.12989/sss.2014.14.5.917.
- Francesca Mangili. *Development of advanced computational methods for prognostics and health management in energy components and systems*. PhD thesis, Politecnico di milano, Milan (Italie), March 2013. URL <https://www.politesi.polimi.it/handle/10589/74461>.
- MATLAB. R2020a, 2020. URL <https://www.mathworks.com/>.

- M. Matsuichi and T. Endo. Fatigue of metals subjected to varying stress. *Japan Society of Mechanical Engineering*, 1968.
- K. Medjaher, D. A. Tobon-Mejia, and N. Zerhouni. Remaining useful life estimation of critical components with application to bearings. *IEEE Transactions on Reliability*, 61(2):292–302, 2012.
- M.A. Miner. Cumulative Damage in Fatigue. *Journal of Applied Mechanics*, 12(3): 159–164, 1945.
- Sidharth Mishra, Uttam Sarkar, Subhash Taraphder, Sanjoy Datta, Devi Swain, Reshma Saikhom, Sasmita Panda, and Menalsh Laishram. Principal Component Analysis. *International Journal of Livestock Research*, page 1, 2017. doi: 10.5455/ijlr.20170415115235.
- A. Mohammed, J. I. Melecio, and S. Djurovi. Stator winding fault thermal signature monitoring and analysis by in situ fbg sensors. *IEEE Transactions on Industrial Electronics*, 66(10):8082–8092, 2019.
- Sing Kiong Nguang, Ping Zhang, and Steven Ding. Parity relationbased fault estimationfor non linear systems: a lmi approach. *IFAC Proceedings Volumes*, 39(13):366 – 371, 2006. ISSN 1474-6670. doi: <https://doi-org-s.bibliopam.ec-lyon.fr/10.3182/20060829-4-CN-2909.00060>. URL <https://www.sciencedirect-com.bibliopam.ec-lyon.fr/science/article/pii/S1474667015344426>. 6th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes.
- Olivier Ondel. *Diagnosis by Pattern Recognition: application on a set inverter – induction machine*. Theses, Ecole Centrale de Lyon, October 2006. URL <https://tel.archives-ouvertes.fr/tel-00113102>.
- Karl Pearson. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901. doi: 10.1080/14786440109462720. URL <https://doi.org/10.1080/14786440109462720>..
- Karl Pearson and Francis Galton. Note on regression and inheritance in the case of two parents. *Proceedings of the Royal Society of London*, 58:240–242, 1895. doi: <https://doi.org/10.1098/rspl.1895.0041>.
- Michael Pecht and Rubycja Jaai. A prognostics and health management roadmap for information and electronics-rich systems. *Microelectronics Reliability*, 50(3): 317 – 323, 2010. ISSN 0026-2714. doi: <https://doi-org-s.bibliopam.ec-lyon.fr/10.1016/j.microrel.2010.01.006>. URL <https://www.sciencedirect-com.bibliopam.ec-lyon.fr/science/article/pii/S0026271410000181>.
- Franz Pernkopf. Bayesian network classifiers versus selective k-nn classifier. *Pattern Recognition*, 38(1):1 – 10, 2005. ISSN 0031-3203.
- D. Alexander Poularikas. *The Transforms and Applications Handbook*. CRC, 02 2000. ISBN ISBN-10: 0849385954 ISBN-13: 978-0849385957.

- R Core Team. R: A language and environment for statistical computing. manual, Vienna, Austria, 2020. URL <https://www.R-project.org/>. tex.organization: R Foundation for Statistical Computing.
- C. Radhakrishna Rao. The use and interpretation of principal component analysis in applied research. *Sankhy: The Indian Journal of Statistics, Series A* (1961-2002), 26(4):329–358, 1964. ISSN 0581572X. URL <http://www.jstor.org/stable/25049339>. Publisher: Springer.
- S. Raschka and V. Mirjalili. *Python machine learning*. Packt Publishing, 2017. ISBN 978-1-78712-602-2. URL https://books.google.fr/books?id=_pIGDwAAQBAJ.
- M. Rayyam, M. Zazi, and Y. Hajji. Detection of broken bars in induction motor using the extended kalman filter (ekf). In *2015 Third World Conference on Complex Systems (WCCS)*, pages 1–5, 2015.
- Paulo Roriz, Susana Silva, Orlando Frazão, and Susana Novais. Optical Fiber Temperature Sensors and Their Biomedical Applications. *Sensors (Basel, Switzerland)*, 20(7):2113, April 2020. ISSN 1424-8220. doi: 10.3390/s20072113. URL <https://pubmed.ncbi.nlm.nih.gov/32283622>. Publisher: MDPI.
- Peter J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53 – 65, 1987. ISSN 0377-0427. doi: [https://doi.org/10.1016/0377-0427\(87\)90125-7](https://doi.org/10.1016/0377-0427(87)90125-7). URL <http://www.sciencedirect.com/science/article/pii/0377042787901257>.
- Rumelhart, D., Hinton, and R. G., Williams. Learning representations by back-propagating errors. *Nature*, 323:533 – 536, 10 1986. doi: <https://doi.org/10.1038/323533a0>.
- S. Russell, P. Norvig, and F. Popineau. *Intelligence artificielle: Avec plus de 500 exercices*. Pearson education. Pearson, 2010. ISBN 978-2-7440-7455-4. URL <https://books.google.fr/books?id=DWTIFWSGxJMC>.
- Sachin Kumar, M. Torres, Y. C. Chan, and M. Pecht. A hybrid prognostics methodology for electronic products. In *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, pages 3479–3485, 2008.
- S. A. Saleh and M. A. Rahman. Modeling and protection of and three-phase power transformer using wavelet packet transform. *IEEE Transactions on Power Delivery*, 20(2):1273–1282, Apr. 2005.
- R. Satishkumar and V. Sugumaran. Estimation of remaining useful life of bearings based on support vector regression. *Indian journal of science and technology*, 9, 2016.
- Claude Elwood Shannon. A Mathematical Theory of Communication. *The Bell System Technical Journal*, 27:379–423, October 1948.
- A. Soualhi, M. Makdessi, R. German, F. R. Echeverría, H. Razik, A. Sari, P. Venet, and G. Clerc. Heath monitoring of capacitors and supercapacitors using the

- neo-fuzzy neural approach. *IEEE Transactions on Industrial Informatics*, 14(1): 24–34, 2018.
- Abdenour Soualhi, Kamal Medjaher, Guy Celrc, and Hubert Razik. Prediction of bearing failures by the analysis of the time series. *Mechanical Systems and Signal Processing*, 139:106607, 2020. ISSN 0888-3270. doi: <https://doi-org-s.bibliopam.ec-lyon.fr/10.1016/j.ymssp.2019.106607>. URL <https://www-sciencedirect-com.bibliopam.ec-lyon.fr/science/article/pii/S0888327019308283>.
- C. Spearman. The proof and measurement of association between two things. *The American Journal of Psychology*, 15(1):72–101, 1904. doi: 10.2307/1412159. URL <http://www.jstor.org/stable/1412159>.
- Claasen TA and Wolfgang Mecklenbräuker. The wigner distribution - a tool for time–frequency signal analysis. iii. relations with other time–frequency signal transformations. *Philips JI Research*, 35:372–389, 01 1980.
- T. Tao and W. Zhao. A support vector regression-based prognostic method for li-ion batteries working in variable operating states. In *2016 Prognostics and System Health Management Conference (PHM-Chengdu)*, pages 1–5, 2016.
- C. Torrence and G. P. Compo. A practival guide to wavelet analysis. *Bulletin of the American Metereological Society*, 79(1):61–78, Jan. 1998.
- Stéphane Tufféry. *Data mining et statistique décisionnelle*. Technip, Paris, France, 4ème édition edition, 2012. ISBN 2-7108-1180-4.
- C. H. van der Broeck, T. Polom, R. D. Lorenz, and R. W. De Doncker. Thermal monitoring of power electronic modules using device self-sensing. In *2018 IEEE Energy Conversion Congress and Exposition (ECCE)*, pages 4699–4706, 2018.
- Vladimir N. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, Berlin, Heidelberg, 1995. ISBN 0387945598.
- J Ville. Théorie et applications de la notation de signal analytique. cables et transmission. In *Theory and applications of the notion of complex signal, Tech. Rept. T-92, The RAND Corporation*, volume 2A, pages 61–74, 1948.
- E. A. Wan and R. Van Der Merwe. The unscented kalman filter for nonlinear estimation. In *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*, pages 153–158, 2000. doi: 10.1109/ASSPCC.2000.882463.
- Wang Chencheng and Li Yongdong. A novel speed sensorless field-oriented control scheme of im using extended kalman filter with load torque observer. In *2008 Twenty-Third Annual IEEE Applied Power Electronics Conference and Exposition*, pages 1796–1802, 2008.
- Richard Weber. Time series, 2001. URL <http://www.statslab.cam.ac.uk/~rrw1/timeseries/t.pdf>.
- John G. Webster. *Measurement, Instrumentation, and Sensors Handbook*. CRC Press, 1998. ISBN 0-8493-8347-1.

- E. Wigner. On the Quantum Correction For Thermodynamic Equilibrium. *Physical Review*, 40(5):749–759, jun 1932. doi: 10.1103/PhysRev.40.749.
- Wikipedia contributors. Microphone — Wikipedia, the free encyclopedia, 2020a. URL <https://en.wikipedia.org/w/index.php?title=Microphone&oldid=973621749>. [Online; accessed 28-August-2020].
- Wikipedia contributors. Time series — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Time_series&oldid=982786780, 2020b. [Online; accessed 13-October-2020].
- Wikipédia. Silhouette (clustering) — wikipédia, l'encyclopédie libre, 2019. URL [http://fr.wikipedia.org/w/index.php?title=Silhouette_\(clustering\)&oldid=162146683](http://fr.wikipedia.org/w/index.php?title=Silhouette_(clustering)&oldid=162146683). [En ligne; Page disponible le 27-août-2019].
- B. Wu, M. Wang, B. Wu, Y. Luo, and C. Feng. Cyclostationarity analysis and diagnosis method of bearing faults. In *2009 IEEE International Conference on Automation and Logistics*, pages 576–581, 2009.
- X. Wu, J. Kumar, V. and Ross Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, P.S. Yu, Z Zhou, M. Steinbach, D.J. Hand, and D. Steinberg. Top 10 algorithms in data mining. *Knowl Inf Syst*, 14:1–37, 01 2008. doi: <https://doi.org/10.1007/s10115-007-0114-2>.
- K. Xue, Z. Wu, H. Li, and G. Yang. Research on identification method of aero-engine bearing fault using acoustic emission technique based on wavelet packet and rough set. In *2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, pages 1499–1503, 2017.
- Yun Yang. *Temporal data mining via unsupervised ensemble learning*. Elsevier, July 2018. ISBN 978-0-12-811654-8.
- Shun-Zheng Yu. Hidden semi-markov models. *Artificial Intelligence*, 174(2): 215–243, 2010. ISSN 0004-3702. doi: <https://doi-org-s.bibliopam.ec-lyon.fr/10.1016/j.artint.2009.11.011>. URL <https://www.sciencedirect-com-s.bibliopam.ec-lyon.fr/science/article/pii/S0004370209001416>. Special Review Issue. Special Review Issue.
- Z. Zheng, Y. Li, and M. Fadel. Sensorless control of pmsm based on extended kalman filter. In *2007 European Conference on Power Electronics and Applications*, pages 1–8, 2007.
- M. Zhong, S. X. Han, Q. Han, and Q. Ding. Parity space-based fault estimation for linear discrete time-varying systems. *IEEE Transactions on Automatic Control*, 55(7):1726–1731, 2010.
- Zhi-Jie Zhou, Chang-Hua Hu, Xiao-Xia Han, Hua-Feng He, Xiao-Dong Ling, and Bang-Cheng Zhang. A model for online failure prognosis subject to two failure modes based on belief rule base and semi-quantitative information. *Knowledge-Based Systems*, 70:221 – 230, 2014. ISSN 0950-7051. doi: <https://doi.org/10.1016/j.knosys.2014.06.026>. URL <https://www.sciencedirect.com/science/article/pii/S0950705114002470>.

- H. Zhu, D. Kung, M. Cowell, and S. Cherukupalli. Acoustic monitoring of stator winding delaminations during thermal cycling testing. *IEEE Transactions on Dielectrics and Electrical Insulation*, 17(5):1405–1410, 2010.
- D. Zurita, M. Delgado, J. A. Ortega, and L. Romeral. Intelligent sensor based on acoustic emission analysis applied to gear fault diagnosis. In *2013 9th IEEE International Symposium on Diagnostics for Electric Machines, Power Electronics and Drives (SDEMPED)*, pages 169–176, 2013.