POLITECNICO DI TORINO

Master's Degree Course in Mechanical Engineering

MASTER'S DEGREE THESIS

Development of a Machine Learning code for predicting Soot Tail Pipe in a Compression Ignition Engine

Academic Year 2020/2021

Supervisor

Prof. Daniela Anna Misul

Co-supervisor

Dott. Alessandro Falai

Candidate

Corrado Aurora

Summary

Figure Index

Abstract

One of the most urgent problems facing our planet is the pollution produced in the transport sector. Over the years, increasingly stringent regulations have been imposed on the production of pollutants due to the damage they cause to health and the environment.

In this thesis, the production of Particulate Matter (PM) within Compression Ignition Engines (CI) and the after-treatment systems (ATS) is analyzed. In particular, this work is based on the construction of a virtual sensor based on Machine Learning algorithms for the On-Board Driving (OBD) prediction of the Soot Tail pipe produced in a Diesel Engine. Such a system was completed, building a Predictive Artificial Neural Network (ANN) in python, using calculation models belonging to the Deep Learning branch. The artificial intelligence systems are adequate for the resolution of this type of problem due to their high levels of precision and because they can deal with a large amount of data.

This analysis was possible through the data provided by AVL Italia S.r.l containing some measurements carried out with a diesel engine on a roller bench in stationary and transient conditions. The different datasets contain measurements carried out under different operating or environmental conditions and within them there is the trend of twenty-one features including the Soot Tail Pipe. As a result of this, the predictive algorithm was implemented in supervised learning so that the model can collect input and output data from these sheets and then, through a training phase, it finds a rule which is useful for the generation of a desired output even for input values that it has never seen before. Specifically, attention was focused on two Stationary Datasets with an engine speed variation between 800 rpm and 4500rpm, which differ from each other in EGR conditions. At first, two Fully Connected Feed Forward Neural Networks (FFNN) for the prediction of the Soot Tail Pipe in stationary conditions were constructed. To make the neural networks as efficient as possible, numerous analyzes and tests were carried out, initially to study and understand the dataset's behavior and then to detect the features of greatest relevance for the prediction of the soot through the tuning of the hyperparameters (feature importance). Subsequently, the network parameters were improved with appropriate algorithms for the upgrade of the performances and the minimization of the error between real and predicted values by a numerical evaluation of the Mean Square Error (MSE) and Determination Coefficient (R^2).

After that, attention was focused on the Transient Dataset, containing some measurements made on the WLTC normative guide cycle. Also, in this case a new Feed Forward Neural Network was built carrying out the same optimization processes as in the previous networks for the prediction of Soot Tail Pipe.

Introduction

In recent years, air pollution has been attracting increasing interest from both an environmental and a social point of view due to the serious damage it produces to the environment and to humans. One of the major sources of pollutants is the transport sector, which is required to reduce fuel consumption by up to 30% within the next 5 years. Despite this, the demand for fuel is continuously increasing as well as the number of vehicles; in fact, due to the increase in population, the global car fleet will increase by 80% reaching 2000 million vehicles on the road by 2040.

More than 90% of the vehicles currently on the road have an internal combustion engine (ICE). It represents a simple, compact, and economical solution compared to other systems dedicated to propulsion. It also has a favorable weight-to-power ratio, linked to the high-density energy possessed by liquid fuels, such as diesel and petrol. However, it does have some important drawbacks such as:

- Production of carbon dioxide (greenhouse gas that contributes to global warming)
- Use of fossil fuels
- Production of pollutants resulting from the fact that the combustion process is not ideal and incomplete.

Pollutants are divided into primary and secondary. Primary pollutants are those emitted directly because of combustion, while the secondary pollutants are created as a result of the combination of primary ones.

Primary pollutants are:

- Carbon Monoxide (CO)
- Hydrocarbons (HC)
- Nitrogen Oxides (NOx)
- Particulate Matter (PM)
- Sulphate (SOx)

Diesel Engines produce a large number of suspended particles that seriously harm human health by depositing in the lungs. For these reasons, the regulations set by the various states have become increasingly stringent especially for these types of engine. This has prompted manufacturers to search for new technologies and to focus their attention on more accurate diagnostic tools. It is essential for the protection of the environment and human health to try to constantly monitor the quality of pollutants produced by the various vehicles. Virtual sensors are becoming progressively more useful since they are not subjected to the stress of the surrounding environment and do not create space problems unlike the physical ones.

During my thesis work I have tried to design a virtual sensor for Soot Tail Pipe measurement (downstream of the DOC-DPF system) with the future aim of creating a system that can signal to the user that his car exceeds the preset emission values or if there are some malfunctions within its DPF.

CHAPTER 1 – IC Combustion Engine

To fully understand the following thesis, it is necessary to address the internal combustion model in Compression Ignition Engines (CI). Along with this, it will also be useful to learn the mechanism by which pollutants are formed and the factors that contribute to their variation. After analyzing these phenomena, I will finally discuss the methodologies and instruments to limit their presence at the drain and ATS systems.

1.1 - Fuel Supply System

Inside Compression Ignition Engines, highly reactive fuels with relatively short ignition delays such as diesel are used. In diesel engines, the air is compacted in the cylinder with the production of heat via Joule effect. The latter is composed of a mix of hydrocarbons represented by cetane C16H34. Compared to spark ignition engines, which operate by igniting the spark plug, the fuel cannot be premixed with air because it would lead to immediate combustion reactions.

The injection takes place with the use of injectors for fuel atomization. These devices have inside them a sealing needle along the axis and a sack in the lower part to avoid the dripping phenomenon. Through an appropriate design, the injectors are placed in specific points of the chamber to obtain a combustion process as efficient as possible. The nozzles facing the combustion chamber have electronic controls for the opening. After the command arrives, the needle lifts from its seat and the pressure pump atomizes the fuel particles. Thanks to the high pressure involved, the dimension of the fluid particles is much smaller than the diameter of the nozzle holes.

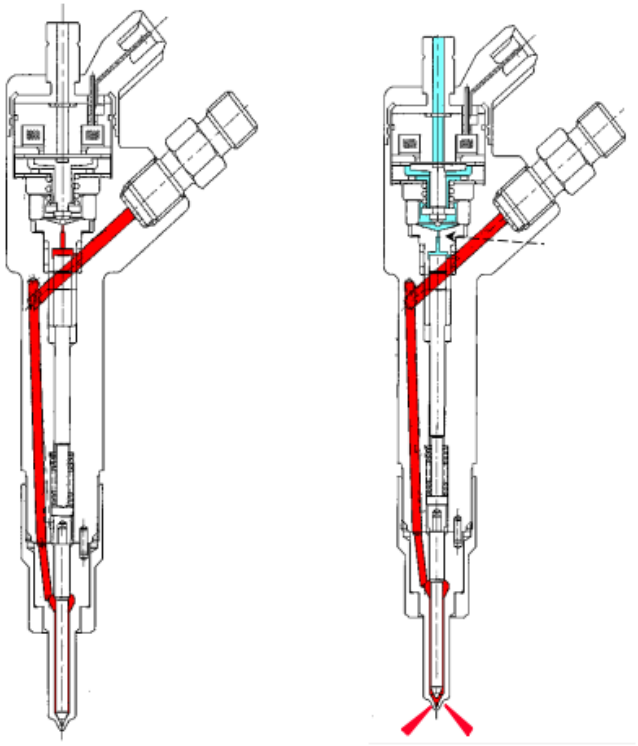

Figure 1 - Injector's structure with nozzle closed and nozzle opened configurations (1)

To control this phenomenon, the fuel must be injected at high pressure (500 – 2200 bar) into the cylinder, almost at the end of the compression phase, near the top dead center (TDC). The jet comes out at a high speed of approx. 100 m/s from the inlet holes (nozzle holes diameter ≈ 0.1mm) and disintegrates in a series of small fuel droplets (d ≈10μm), surrounded by hot compressed air at ≈ 900K at high density (20-30 Kg/m3) which vaporize forming a mixture that ignites spontaneously without the need of an external trigger. The presence of turbulent motions inside the chamber favors the mixing of air and fuel, creating a more homogeneous and efficient combustion. The combustion process starts with an extremely short ignition delay, furthermore this mixture can ignite even with air/fuel ratios far from the stochiometric conditions. This last aspect allows for movement towards poorer environments and experimentation of different strategies for controlling pollutants. (1)

1.2 - Injection Strategies

Nowadays practically all diesel engines are equipped with an electromagnetic injection system, the so called “Common Rail”. This system makes it possible to adopt different strategies for controlling the combustion process. The injectors relate to the “rail” and, overlooking the chamber, they take care of the fuel injection. These devices are coupled with precise injection strategies in order to:

- Contain combustion noise
- Control the production of pollutants
- Release the injection pressure from the rotation speed and the motor load
- Make more injection per cycle

The main parameters which are evaluated in this process are:

- The modulation of the injected flow rate
- Injection time interval
- Pilot stages

The latter application is widely used since the fuel flow introduced during this phase accumulates in the chamber and then burns all together. With appropriate strategies it is possible to make sure that the injected pilot fuel starts to burn when the main injection starts. In this way, the pilot combustion increases the temperature in the chamber and decreases the accumulations which lead to a greater formation of pollutants and inefficient combustion cycles. As already mentioned, one of the greatest advantages of the common rail is the possibility of fragmenting the injection process. The first-generation systems allowed to couple the main injection with a pilot and a post one. On the other hand, those of the latest generation allow the division of the injected quantity of the main phase as shown in figure 2. (1)

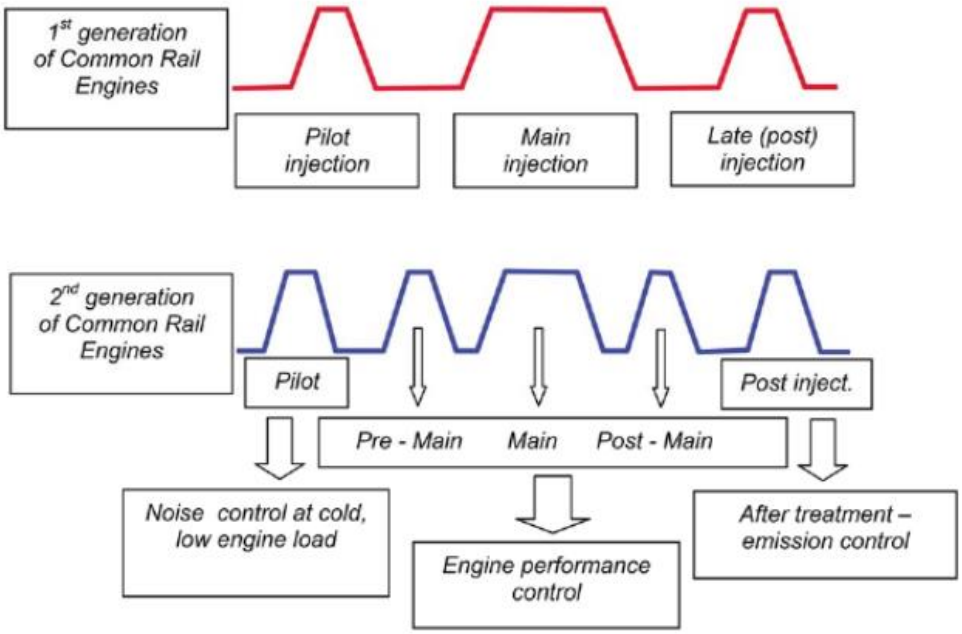

Figure 2 - Common Rail System (1)

Excluding the pilot one, the injections presented are the following:

- Pre-Main: carried out with very low advance values compared to the main injection, it allows to control the rapidity of combustion development, limiting the formation of pollutants (nitrogen dioxide and the carbonaceous portion of the particulate, called soot). Indeed, these injections manage to attenuate the peak temperatures of the combustion process which have a direct correlation with NOx emissions.
- Main: is the main injection and its purpose is the combustion
- Post-Main: is the injection carried out immediately after the main one in order to modulate the final phase of the combustion. Its presence favors soot oxidation, increasing the temperature of the final stage.
- Post: when the piston is around the bottom dead center (BDC) it is possible to carry out a further injection which determines a significant rise in the exhaust temperature allowing the periodic regeneration of the particulate matter. This injection can also produce unburned hydrocarbons (HC), which are necessary to create a reducing environment, essential for the DeNOx catalyst. These are

therefore auxiliary injections for the management of the post treatment system, and it has nothing to do with the combustion process.

1.3 - Combustion Process

The conventional combustion process of a diesel engine can be divided into 4 intervals as shown in figure 3.

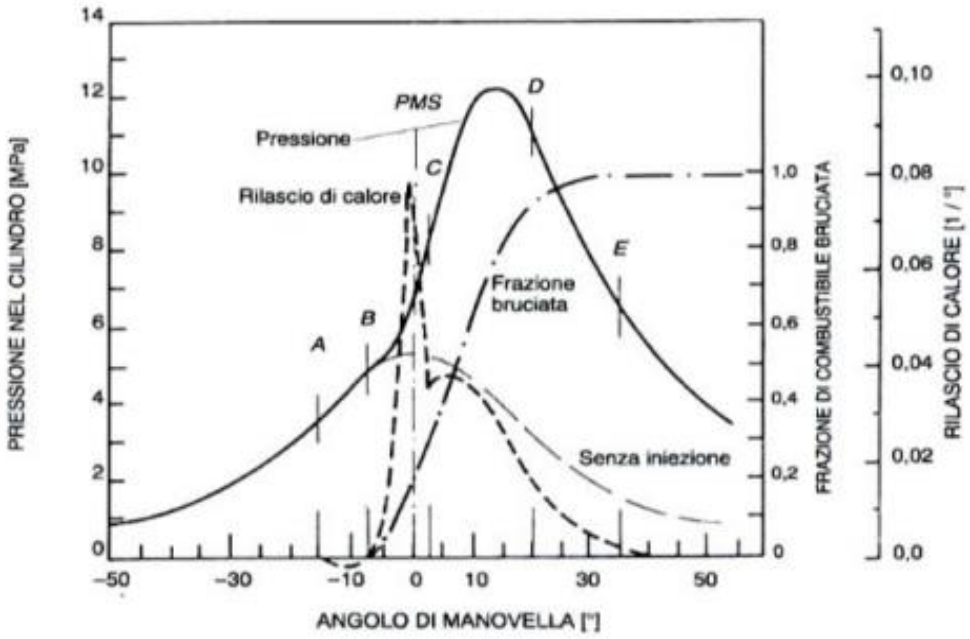

Figure 3- Flow of cylinder pressure as a function of the crank angle in a compression ignition engine. Heat Release Rate (HRR) represented in dotted line. (2)

- Injection delay, corresponding to the stroke (A-B)
- Combustion in the premixed phase, corresponding to the stroke (B-C)
- Diffusive combustion, corresponding to the stroke (C-D)
- Late combustion phase, corresponding to the stroke (D-E)

These phases are visible by evaluating the HRR (heat release rate) and the pressure curve as a function of the crank angle.

A = Start of injection point (SOI)

B= Start of combustion point (SOC)

The **Ignition Delay (A-B)** is the time interval between the start of injection (SOI) and the Start of Combustion (SOC). This time delay can be attributed to both physical and chemical phenomena. The physical delay is determined by the time it takes for the fuel mixing and evaporation with air to form a homogeneous mixture. The chemical delay, on the other hand, is determined by the delays present in the various chemical reactions that precede the self-ignition of the mixture. Because of such delays, it is essential to inject the fuel into the chamber at different crank angles before the Top Dead center (TDC) to control this phenomenon.

$$\tau_{tot} = \tau_{Physical} + \tau_{chemical}$$

Once the self-ignition has been reached, the fuel begins to burn in a phase called **Premixed Combustion (B-C)**. The ignition of the first cores takes place with a consequent increase in temperature and pressure inside the chamber. In premixed combustion, there is a strong release of heat which can cause noise and vibrations that are harmful to the engine. This aspect must be constantly controlled by designers as it appears to be one of the most critical points in the management of the combustion process. A further disadvantage of this aspect is that in the presence of oxygen and high temperatures, nitrogen oxides (NOx) are formed.

Following premixed combustion, **Diffusive Combustion (C-D)** takes place. This process consumes more than 90% of the fuel introduced into the chamber, resulting in a strong release of energy at the maximum pressure of the cycle. The speed with which the energy increases can be controlled by acting on the Injection Rate. Following the consumption of oxygen and the increase in combustion gases, dehydrogenation, condensation, and pyrolysis reactions take place leading to the formation of the first carbonaceous nuclei of soot.

Finally, at the end of the injection process, there is the **Late Combustion Phase (D-E).** At this point the oxidation of several carbonaceous nuclei present in the chamber can occur. In addition to the closure of the injector it may happen the phenomenon of dripping resulting in the production of pollutants in the chamber that is found following the discharge. Part of the unburnt fuel can also be deposited in the different interstitials present inside the chamber or at the level of the piston. For this reason, it is very important that inside the chamber there are turbulent motions and Eddies to create a gas recirculation to make the combustion process more efficient and reduce the amount of pollutants.

1.4 - Particulate Formation Mechanism

The particulates come from the incomplete combustion reaction. The phenomena associated with their formation are particularly complex, despite this there are numerous models that describe their formation.

Before defining such mechanism, it is important to define certain quantities involved:

- Dosage α

$$\alpha = \frac{Air\ mass}{Fuel\ mass}$$

  This ratio defines the relationship between the mass of air and the mass of fuel in the mixture. Stoichiometric ratio $\boldsymbol{\alpha_{st}}$ means the reaction ratio for which all the mixture burns without unburned reagents.

- Relative Air Fuel Ratio $\boldsymbol{\lambda}$

$$\lambda = \frac{\alpha}{\alpha_{st}}$$

- Equivalent Ratio **Φ**

$$\Phi = \frac{1}{\lambda}$$

- Apparent Heat Release Rate **AHRR**

$$AHRR = \frac{\gamma}{\gamma - 1} p\, dV + \frac{1}{\gamma - 1} V\, dp$$

$$\gamma = \frac{cp}{cv}$$

AHRR is a simplified evaluation of the heat release rate (HRR), the difference being that it does not consider the exchanges of heat with the walls of the cylinder.

In the following image it is possible to view in a general example, the evolution of the fuel jet introduced into the chamber as the ASI (After Start of Injection) changes. This representation is particularly useful for evaluating the areas of pollutant formation.

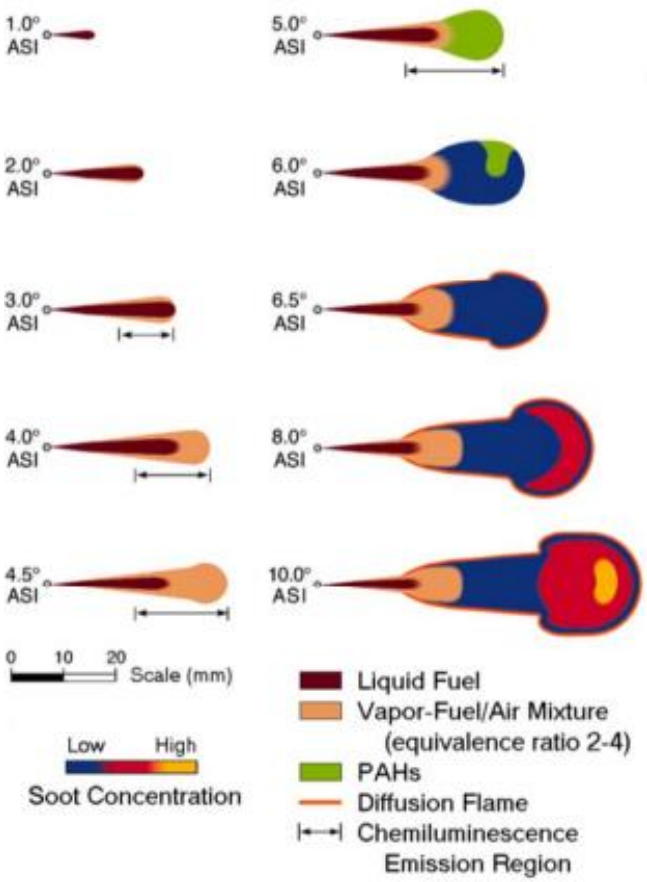

Figure 4 - Jet evolution and pollutants formation (1)

In the early stages, the fuel enters the chamber in liquid form, forming an accumulation zone at the tip. It then evaporates when it meets the air and mixes with it. After 4° ASI it is possible to observe the SOC and the chemiluminescence phenomena due to the formation reactions of the first radicals. In this way, the premixed combustion process begins with the consequent increase in pressure and a strong thermal release (HRR).

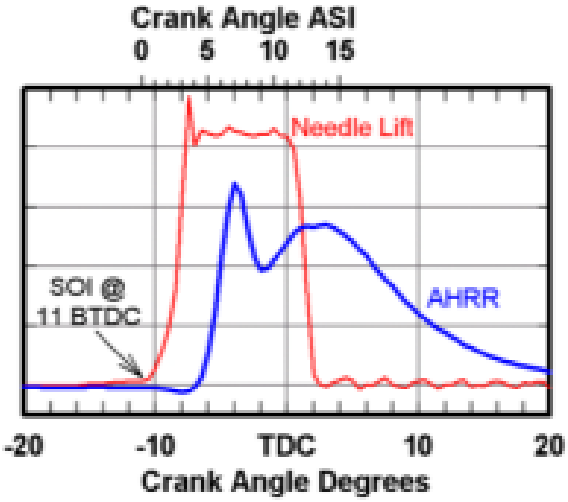

Figure 5 - AHHR curve (1)

This combustion is characterized by very rich local areas with an equivalent ratio $\Phi \sim 4$. The products of the rich combustion are the following:

- CO2
- H2O
- CO
- H2
- Polycyclic aromatic hydrocarbons (PAH)

As also indicated in the figure, the green area present at 5° ASI represents the formation of PAHs. Immediately after 6° ASI it is possible to observe the transformation of the green areas into blue, this represents the soot development from PAH due to some chemical reactions and the growth mechanism of the carbonaceous particles which will form soot.

Subsequently, the oxygen diffuses inside the jet and the diffusive combustion phase begins. Unlike premixed combustion, the latter occurs at equivalent ratios $\Phi \sim 1$, therefore around the stoichiometric ratio and leads to a further increase in temperature. In this phase above 6.5° ACI, the soot produced increases more and more also due to the incomplete oxidation of the various substances present inside the jet. During the diffusive phase, the particulate can be oxidizing if the duration of the diffusive flame and the availability of

oxygen allow it. So, the particulate formation is the result of balance between the mechanisms of formation and those of oxidation.

At the end of the injections, the late combustion leads to a reduction in pollutants and a reduction in engine efficiency, therefore it is important to make an appropriate trade off. Finally, the diffusive flame meets the walls of the chamber and the well and is extinguished due to the temperature gradient and oxygen deficiency. (1)

1.5 - PM evolution into the jet

In the following figure it is possible to analyze the entire development of the jet during all the phases of the combustion and it is representative of all the zones where pollutants are formed.

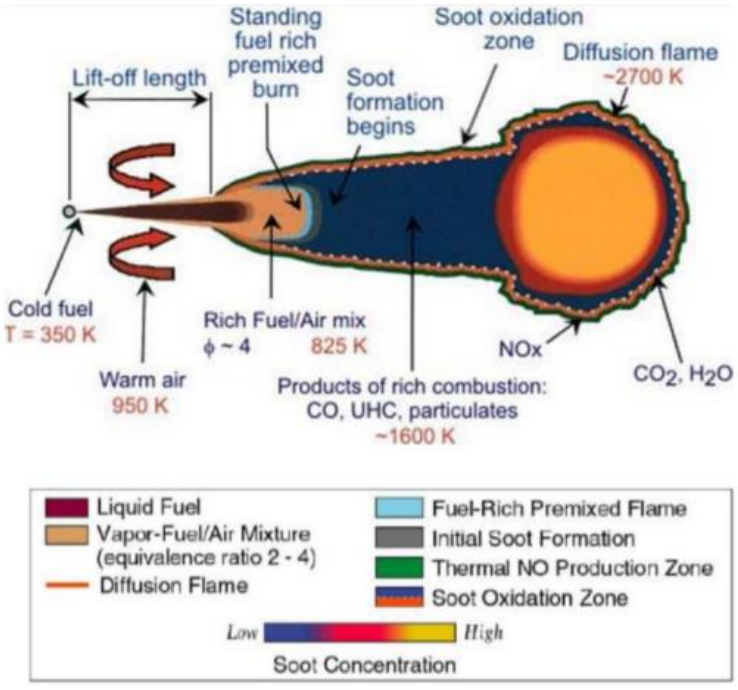

Figure 6- Jet Description (1)

As can be seen, the fuel is fed into the chamber at a temperature of 350 K and then warmed up after its atomization and mixed with air up to 650 K. Then there is a vertical peak

of heat and a strong exothermic reaction, this phenomenon is due to the self-ignition of the particles and indicates the start of premixed combustion. At this point there is a phase in which the temperature remains constant and about equal to 1600K.

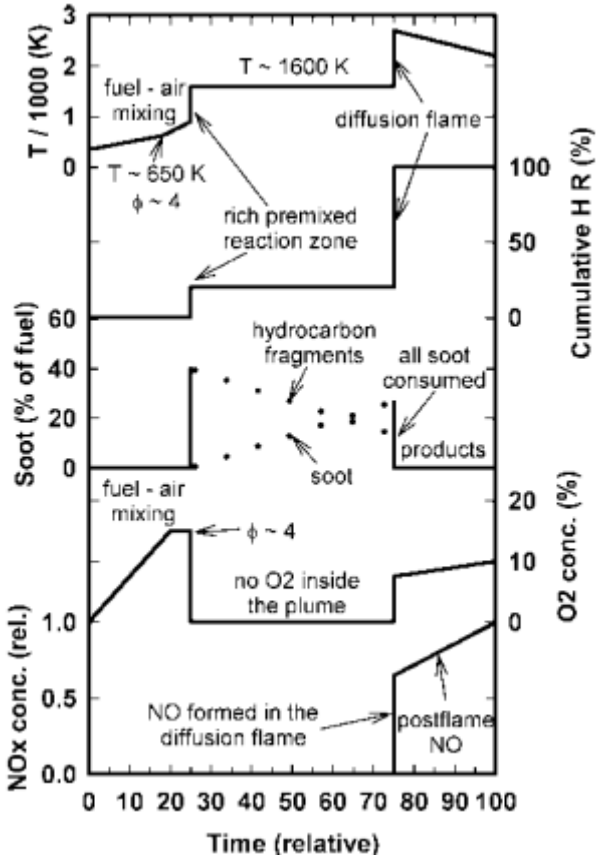

Figure 7 - Properties evolution during combustion (1)

As can be seen in figure 7, during the premixed phase it can be recognize:

- A peak in soot particle release
- After being linearly increased during the mixing phase, oxygen is consumed during combustion
- There is no trace of NOx because it is a rich combustion, deficient in oxygen so less prone to oxidation reduction.

Then, it reaches the state of diffusive combustion with a T ~ 2700 K. Being a stoichiometric combustion, the carbon particles pass through the flame and oxidize. At this point the concentration of O_2 increases as the combustion products expand in the surrounding environment surrounded by oxygen. In this phase it happens: the formation of NOx and the oxidation of soot. As can easily be derived from the latter observation it is important during the combustion design process, to make an appropriate tradeoff between the production of soot and that of NOx.

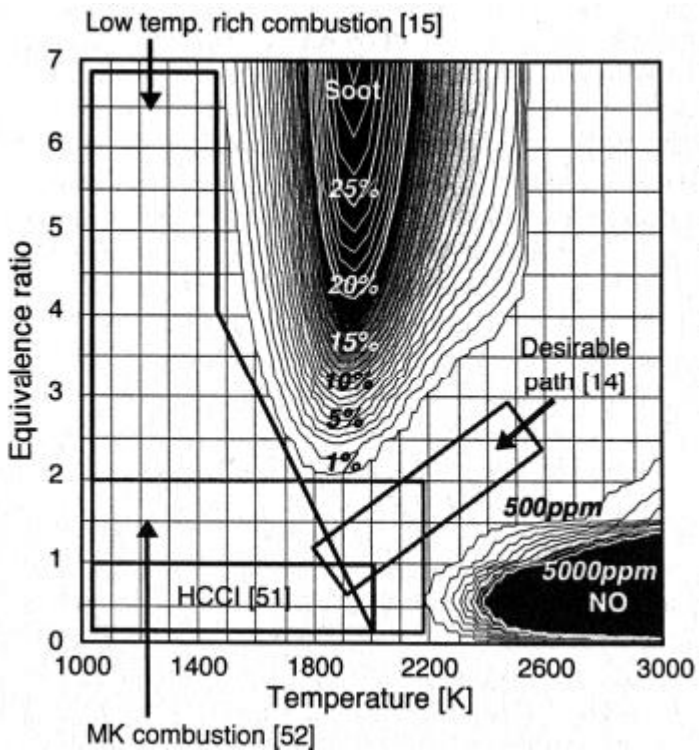

Figure 8 - Kamimoto- Bae Diagram (3) (4) (5) (6)

The diagram of Kamimoto-Bae is particularly efficient to find the right compromise. This representation is a map as a function of temperature and equivalence ratio that explains the different zones of formation. (1)

1.6 - Soot composition

Particulate matter is a nongaseous emission composed by an articulated set of particles.

Most of the particulate matter comes from the carbon found in diesel fuel (H/C ~ 2), the remaining part comes from lubricating oil, ashes, and sulphates. Of all the hydrocarbons present inside the PM, the greatest contribution comes from the aromatics.

PM is mainly composed of 3 parts:

- **Solid Fraction (SOL):** it is composed of elemental carbon and ashes
- **Soluble Organic Fraction (SOF):** It contains the organic material deriving from fuel and lubricating oil
- **Sulphate Particles (SO4):** it is composed of sulfuric acid and water

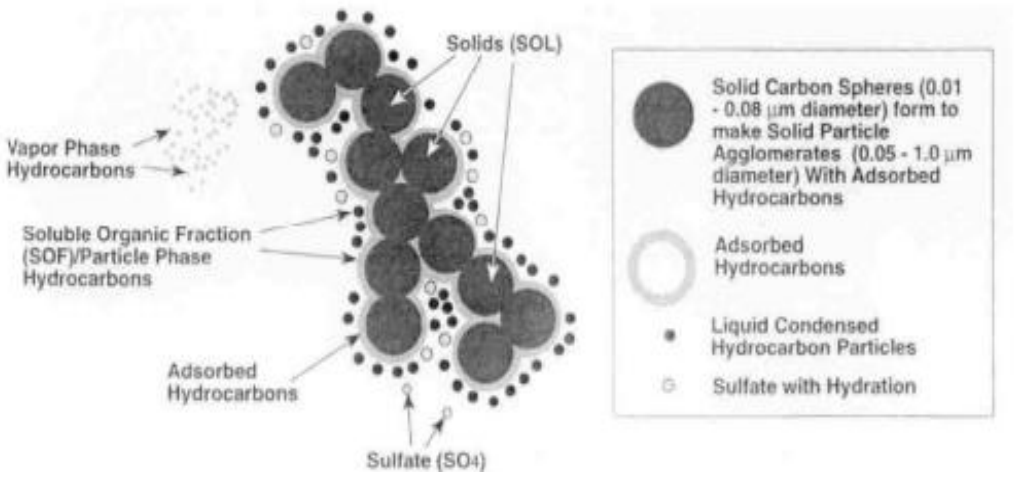

Figure 9 - Particulate structure (1)

It is not possible to define a division between these three fractions because they depend on the operating conditions of the engine. To give an idea of the amount of soot produced depending on engine conditions and the proportion between the different parts, figure 10 shows an example of a map of soot production according to engine speed (rpm) and engine load (%).

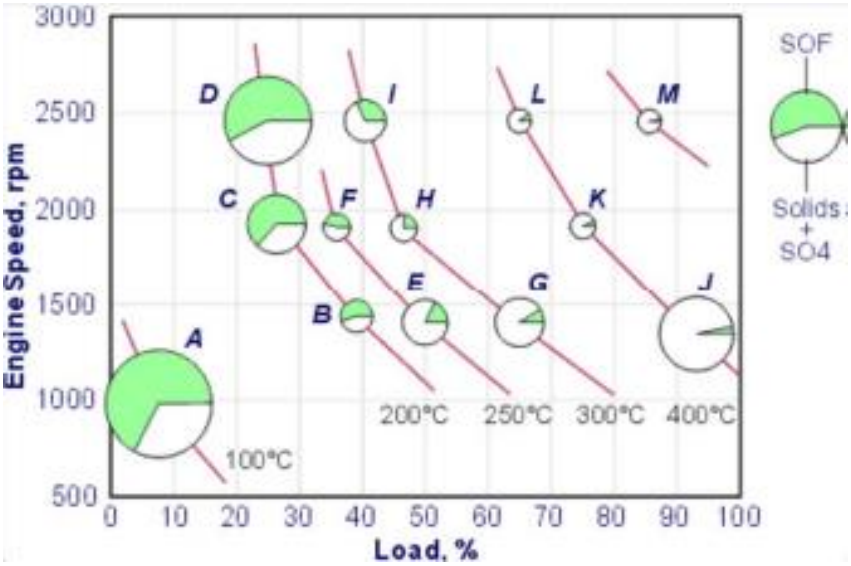

Figure 10 - Graph of the soot variation as a function of Load and Engine Speed (1)

The size of the different balls represents the total amount of particulate matter produced under certain conditions. Within them, the white zone represents the fraction of SOL and SO4, while the green zone represents the fraction of SOF. It is also possible to note that as engine conditions increase, the solid part present in particulate matter becomes more and more dominant.

The part referred to as SOL is formed by unburned carbon particles and derives directly from the premixed combustion process.

The elementary carbon particles deriving from the heterogeneous combustion process have a hexagonal structure. These agglomerates, forming platelets and then synthesize into layers by increasing their size and form crystallite. Finally, the latter continue to randomly agglomerate with different orientation of the planes forming the primary particles (nuclei mode). This mechanism is shown in the following figure 11.

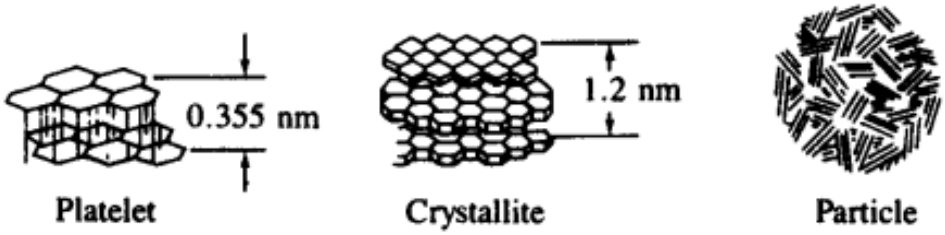

Figure 11 - Evolution of the carbon particle (1)

The ashes instead represent everything that is incombustible and is found in the engine exhaust. This fraction derives from the metals and compounds present in the lubricating oil, in the additives and in the corroded and worn parts of the chamber. In addition, metal additives are also used inside the fuel for the particulate filter (DPF) regeneration, contributing to the ash fraction.

The SOF is composed of organic hydrocarbons, which are absorbable by the solid carbonaceous particles or condense creating liquid particles. This fraction is called soluble because special solvents are used to isolate the SOF from the other particulate parts. The SOF consists of organic compounds belonging to different families of hydrocarbons. It is mostly composed of hydrocarbons with carbon atoms number C between 20 and 36. This composition is very similar to that of the lubricants present in the chamber and less similar to the diesel fuel which has a number of C between 12 and 20. Thus, almost all the PM from the oil is contained in the soluble fraction.

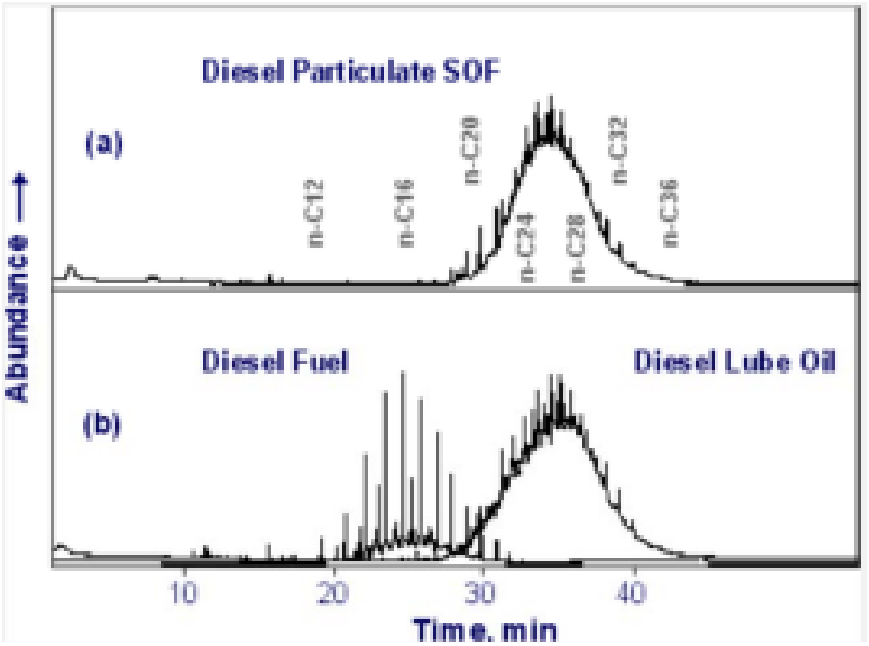

Figure 12 - SOF composition (1)

Sulfate particles derive from the interaction between hydrated sulphuric acid (H2SO4) and water (H2O) in a heteromolecular nucleation process that in saturation conditions leads to the formation of SO4.

Usually, the soot is also called PM10, in this case number ten indicates the diameter of particles that are below 10 μm. The PM is composed of particles of different sizes and they are subdivided in the following way, as visible also in figure 13.

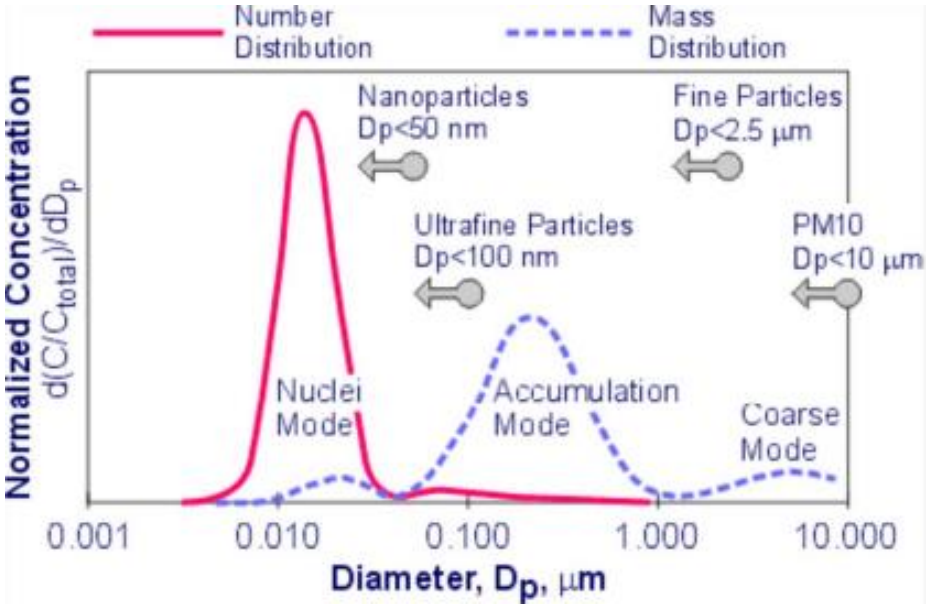

Figure 13 - Particulate mass and particle number distribution (1)

In this figure the mass distribution is represented by a blue dotted curve and the particle number distribution is represented by the red curve, both as a function of the particle aerodynamic diameter.

These curves allow to distinguish 3 different zones:

- **Nuclei mode**: It includes nanoparticles and ultrafine particles with a diameter between 4 nm and 45 nm. This fraction is composed mainly of hydrocarbons and condensed sulphuric acids, in the liquid state, which are formed in the exhaust systems because of the strong decrease in temperature and the mixing with air. The concentration of this part depends heavily on the dilution conditions and although it covers a small percentage by mass, it includes most of the soot particles. Nanoparticles are the most dangerous to human health, so it is very important to try to minimize their concentration.

- **Accumulation mode**: Fine particles with a particle diameter ranging from 45 nm to 1 nm. This fraction consists mainly of carbon particles with absorbed hydrocarbons and condensed vapors. The particles are not really numerous but give a high contribution in mass.
- **Coarse mode**: This part is formed by wear of the exhaust parts and is made up of the larger particles of soot.

1.7 - ATS systems

The volatile (soluble) part can disappear with the evaporation process while to reduce the amount of dry soot cleansed by carbon particles, they need to be oxidized in the presence of O2 and depending on the temperature at which this reaction takes place it will take some time. For this reason, particulate matter can be reduced considerably by heating it in the presence of oxygen, and this is the fundamental principle of ATS systems.

ATS systems for CI engines shall comprise the following parts:

- **DOC (Diesel oxidation Catalyst)**: For controlling CO, HC emissions and organic compounds in soot (SOF).
- **DPF (Diesel particulate filter)**: It is used to capture carbonaceous particles from the soot through filtration mechanisms. Captured particles are removed from the filter continuously or periodically through the thermal regeneration process.
- **SCR (Selective catalytic reduction)**: For the reduction of NOx. These users, after storing the pollutant in question, promote a strong catalytic reduction reaction with ammonia for the decomposition of oxides of nitrogen.

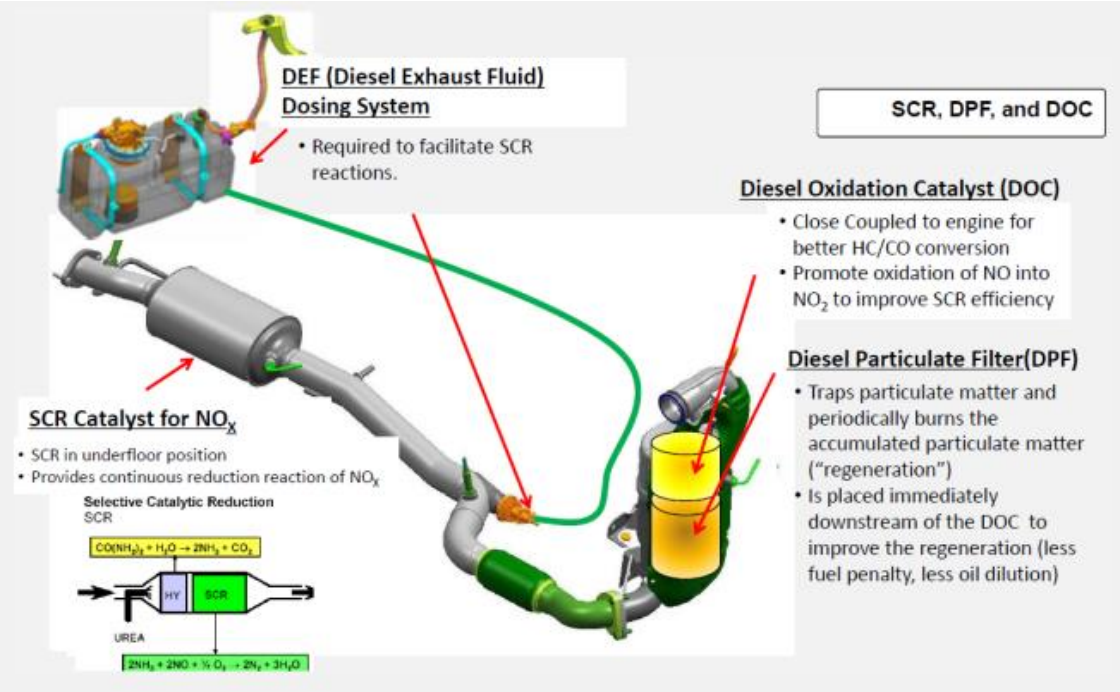

Figure 14 - Euro 6 Diesel ATS configuration (1)

For the following thesis work I focused my attention on the DPF, as the main system for the reduction of particulate matter.

1.8 - DPF Soot Filter

The Diesel Particulate Filter (DPF) is a mechanical filter with the aim of physically capturing soot particles and avoiding their release into the atmosphere. In recent years there have been many models or prototypes that differ fundamentally for:

- The type of material which may be of a metallic or ceramic nature
- The geometric configuration present inside
- Technologies used for regeneration and their control

Figure 15 - DPF (1)

State of the art DPF models are able to achieve filtration efficiencies of 90% accompanied by an excellent level of thermal and mechanical durability. As a mechanical filter, the DPF is not able to reduce the volatile part of the soot but only the solid particles and therefore it is placed in series with the DOC system. The Filter consists of a number of channels, which can have a different type structure:

- **Wall-flow**: alternating blind channels in and out
- **Flow-through**: open channels

Figure 16 - DPF structure Wall-flow and Flow-through (1)

The filtration, therefore, the separation of the carbon particles from the exhaust gases, takes place during the passage of the gas through the channels owing to the porosity of the latter.

Two types of strategies are mainly used:

- **Depth filtration**: The smallest particles pass through the porous matrix and remain trapped by the action of electrostatic forces.
- **Cake filtration**: In this case the particles have larger diameters than the porous matrix.

For this reason, they are deposited on the matrix itself and consequently the successive particles will deposit over the previous ones.

By increasing the accumulation of soot within the DPF, the system achieves an increase in the filtering efficiency, and a better thermal resistance. Nevertheless, as the particulates in the filter increase, it risks to fill itself with a consequent increase in the pressure drop, proportional to the amount of soot accumulated. The latter aspect leads to a lowering of engine performance, so it is important to adopt appropriate filter regeneration strategies to preserve the proper functioning of the engine.

The combustion of the soot by oxygen, requires temperatures above 650 ºC, while the temperature of the exhausts gases is around 200 º C, for this reason an increase in temperature is needed in order to support the combustion.

The regeneration process can be of two types:

- Periodic regeneration: it uses oxygen (O2) as oxidizing agent. Since the temperature of the exhaust gases must be raised above 600 ºC, some catalysts are used to lower them. A widely used strategy is the increase in exhaust gas temperature at the DOC level to make this process as efficient as possible.

- Continuous regeneration: it takes place with the use of NO2 as an oxidizing agent to make the oxidation reactions of particulate matter happen at a temperature of around 250° C. The oxidation process then happens at the same temperature as the exhaust gases.

(1)

CHAPTER 2 – Artificial Intelligence

In this section there are some important notions on Artificial Intelligence and the logic present within the algorithms used for the construction of the virtual sensor. This sector is particularly large and complex, in fact it includes numerous subcategories such as Machine Learning and Deep Learning.

2.1 – Machine Learning

Machine Learning (ML) has been an existing field since 1950 and is currently used in all kinds of industry such as: medical, security and robotics for the construction of predictive algorithms. ML is the science which studies the construction of models capable of learning without being programmed. Such systems work, in a certain sense, just like the human brain, so after extrapolating a series of information from the data, they can form knowledge. After learning from the available data, the system is able to make predictions even from starting data that has never been seen before. For a correct operation of the model, it is necessary to use two sets of data: The **Training Set** with which the algorithms build the predictive structure, and the **Test Set** on which the performances are measured.

The input that is provided to the network is composed of a series of **Features (x)**, which represent the variables useful for learning. The output, instead, is defined as **Label (y)** which can be of different forms depending on the problem being analyzed. Obviously, to make the prediction as accurate as possible, the system needs a series of measurements of the different features, each of these is defined as **Sample**. (7)

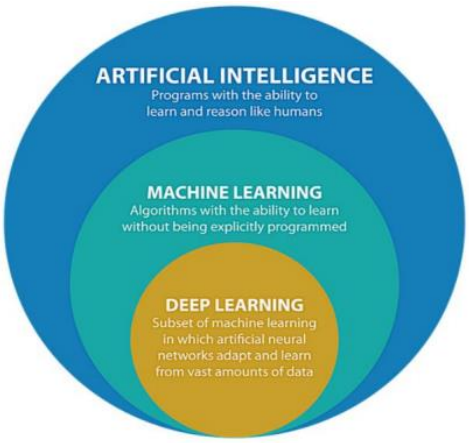

Figure 17- Artificial Intelligence structure (Valvo)

2.2 – Classification of Algorithms

Machine Learning algorithms can be divided into different categories depending on the type of problem or prediction you want to perform or the type of structure of the dataset and the way you want to read it.

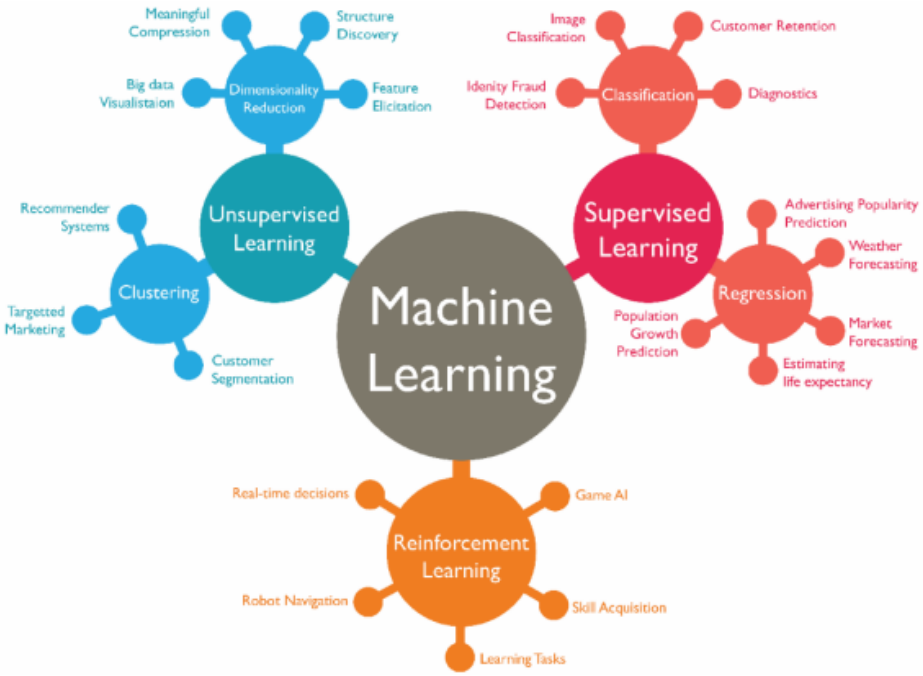

Figure 18 - Classification of Machine Learning algorithms (8)

Machine Learning models are classified as follows:

- Supervised Learning

  In this type of application, the user provides the algorithm with a set of input data (features) and the output value corresponding to each sample (label) on which develop the learning phase. Following the training phase, the program will be able to process a label hypothesis starting from a Sample (feature set) that has never been seen before. In this way, the program finds a rule capable of linking features and labels.

  This model is mainly used for two types of problem:

  - Regression Problem:

    It is an algorithm used to predict a numerical target value after having received the feature set as input. In this way, it can provide a function which approximates the input-output relation. As we can imagine this is not a simple problem because as in most regression problems, we work with a large number of features which must be taken into account. One of the most important cases is Linear Regression.

    We define:

    m= Number of examples

    x= Input variable/ Feature

    y= Output variable/target

    (x, y) = training example

    h= hypothesis linear function which connects x and y

$$h_\theta(x) = \theta_0 + \theta_1 x$$

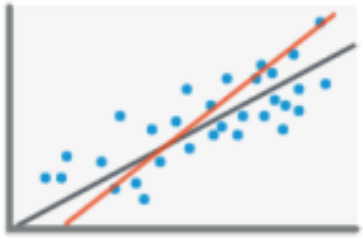

Figure 19 - Regression Model (9)

The main problem is to select the best possible θ parameters to improve the predictive level so that $h_\theta(x)$ becomes as close as possible to y for the training example (x, y). So, we must also minimize the cost function J (Square Error Function), defined by the following equation:

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x_i) - y_i)^2$$

Starting from two defined values of θ_0 and θ_1 we keep changing these values until we find a minimum. This process is carried out with the use of the Gradient Descent Method. The regression type can also be polynomial by increasing the degree of the hypothesis function (h) and thus increasing the number of θ variables to be calibrated for the minimization of the cost function. (7)

- Classification Problem:

  It is used for the recognition of the membership class of the case which is studied, starting from the input data. The output that is returned to us by the program is a discrete value that indicates the belonging class of the sample. Furthermore, we can distinguish two types of classifiers: Binary classification, where the algorithm distinguishes only two classes belonging to the case in question and Multi Class classification, where the output may belong to more than two classes. Usually within

binary classification, the results obtained by the program are 0 or 1 and each of these values represents the belonging class.

Figure 20 - Classification Model (9)

- Unsupervised Learning:
  In this case, in the training phase all the feature values belonging to the different samples are provided to the model but not the label values associated with them. The main algorithm in this category is Clustering, where the machine does not know the classes of the samples supplied in training, but it tries to group the different cases according to the characteristics derived from the data. In this way, the program creates a rule for cluster division, and it associates a specific class to each of them.

Figure 21 - Clustering Model (9)

- **Semi supervised Learning**
  This is a middle way between the 2 previously analyzed cases, in particular only a part of the samples provided to the model contain both the features and the label. In any case, it serves to improve forecasts made on unlabeled data.

- **Reinforcement Learning**

  The algorithm aims at learning optimal behavior, thus improving with experience. The program interacts within an environment and during the study of the problem it performs a series of evaluations, which can obtain recompiles or penalties as feedback. According to the latter, the algorithm continuously improves its experience and optimizes analysis strategies. These systems are very useful for the construction of models that can show the changes in the environment. (10)

In the model developed in this thesis, a Supervised Machine Learning Regressive algorithm has been built since it is an efficient method for predicting continuous values of Soot Tail Pipe. In addition, a neural network will be implemented to allow the algorithm to work more efficiently with a large amount of data.

2.3 – Logic of the System

The behavior of a Machine Learning algorithm is regulated by a set of Parameters that characterize it. Thus, the learning phase of the algorithm is determined by the search of the optimal values of these parameters. Therefore, given a training set we have an objective function (f) that can indicate: the optimal solution to maximize or the error to minimize. Optimization can be carried out through methods that are based on the mathematics of the system or through implicit methods. One of the most widely used mathematical methods, is the calculation of the partial derivatives, in which the function f is derived according to the parameters, placed at 0 and then solved. Most of the algorithms need to define the value of the so called hyperparameters before the learning phase. Such quantities are fundamental for the optimum functioning of the model and can be the degree of polynomial used in a regression, the number of neurons or layers of a neural network or the type of loss function.

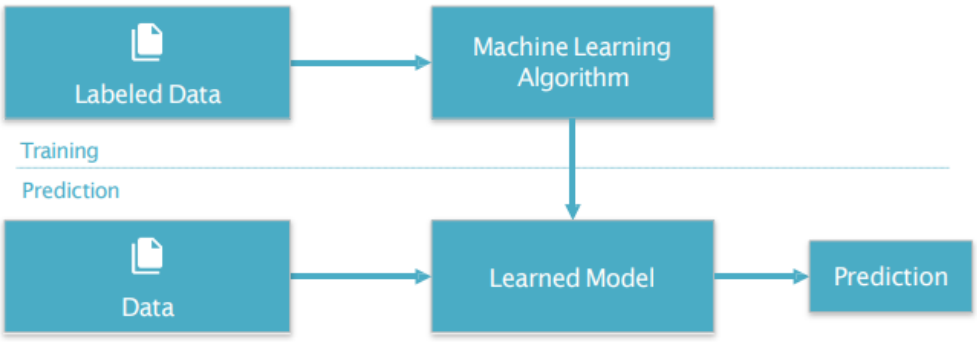

Figure 22 - Machine Learning Logic (9)

By tuning them properly, the algorithm is maximized by increasing its predictive capabilities. Let's analyze how the evaluation of the performance of the algorithm takes place and the definition of the main loss and optimization functions. (11)

2.3.1 – Performance Evaluation

The evaluation of the model is ruled by a set of parameters which characterize it. The learning ability of these algorithms is based on the determination of the optimal values of the parameters and the loss function minimization.

To evaluate model performances in a Regressive Model, a series of coefficients are examined:

- Determination Coefficient (R^2)

  Indicates a portion between the variability of the data and the correctness of the statistical model used. This data is particularly useful as it shows an actual deviation between actual and predicted values.

$$R^2 = \frac{\sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

  Within the formula y_i are the observed data, $\bar{y}$ is the average of the values observed and $\hat{y}_i$ are the data estimated by the model. The Determination

Coefficient varies between 0 and 1, if R^2= 0 the model used is not able to predict the data in analysis, if R^2= 1 the model perfectly explains the data.

- Mean Square Error (MSE)

  Indicates the mean square difference between the real values supplied to the system and those predicted.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2$$

- Root Mean Square Error (RMSE)

  It indicates the square root of the mean quadratic error, present between the real values and those predicted by the system.

$$RMSE = \sqrt{MSE}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2}$$

Within the classification problems, however, the efficiency of the system is evaluated in terms of percentage of accuracy and percentage of the error.

$$Accuracy = \frac{Correctly\ classified\ patterns}{Total\ number\ of\ classified\ pattern}$$

$$Error = 100\% - Accuracy$$

2.3.2 - Training Test and Validation Set

For a correct operation of supervised learning algorithms, the dataset provided to the model must be properly divided into three parts.

- **Training Set**

  It is the set of samples through which the algorithm is trained by finding the optimal value of the hyperparameters. Usually, the dimension of this set in 80% compared to the number of total data and it becomes 60% if we are also including the Validation Set

- **Test Set**

  It is the set of samples on which the algorithm evaluates the final performances after training. The dimension of this set is 20% compared to the number of total data

- **Validation Set**

  Usually, part of the training data is removed from the set to perform the validation. This range contains the patterns on which the model calibrates the hyperparameters without creating a learning from them. The dimension of this set is 20% compared to the number of total data

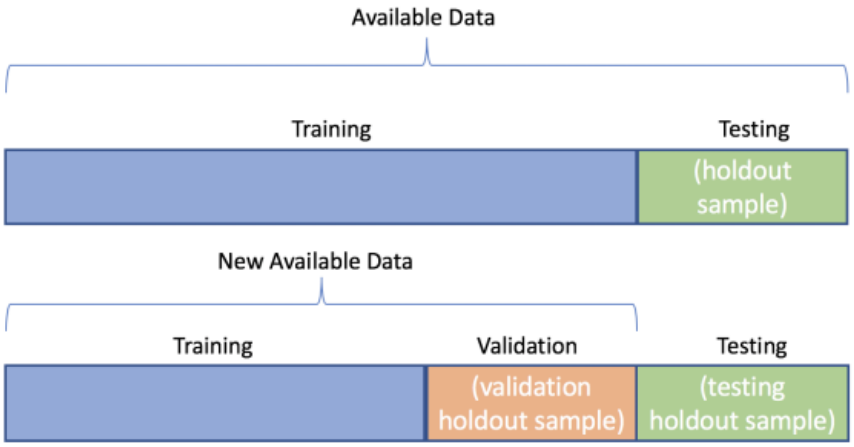

Figure 23 - Representation of the subdivision of a dataset in Train, Test and Validation sets (12)

For the model calibration that I've made for the different dataset it is very important to apply a specific process called Cross-Validation.

2.3.3 Cross – Validation

During this process, the training dataset is divided into complementary subsets, after that the model is trained on a part of these and validated on the remaining part. Such an operation is repeated for different combinations of the subsets. Once best performances and hyperparameters are selected and defined, the final model is trained on the complete training set and then applied to the test set to provide the error. As a result of cross validation, we can have an optimal choice of the hyperparameters.

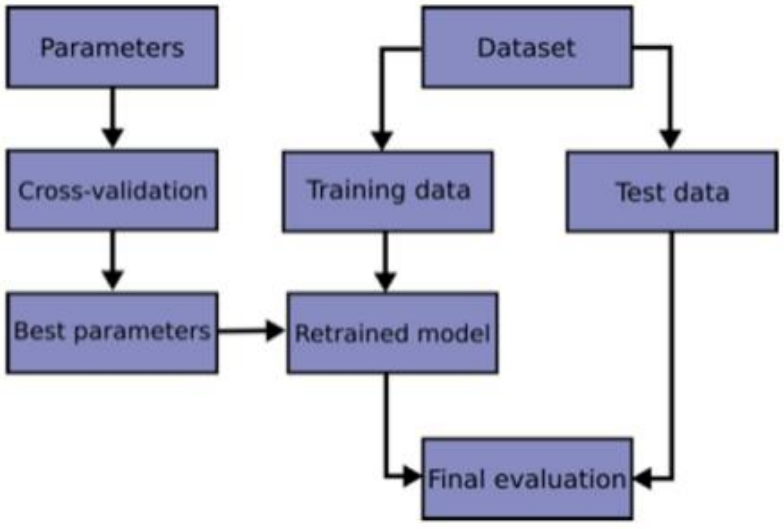

Figure 24 - Flow Diagram with CV (13)

Generally, the training set is divided into 10 parts of which one must be composed of the validation set while the remaining ones make up the training set. Then the training phase starts, repeating the same procedure for all possible combinations, each time selecting a different subset as a validation test.

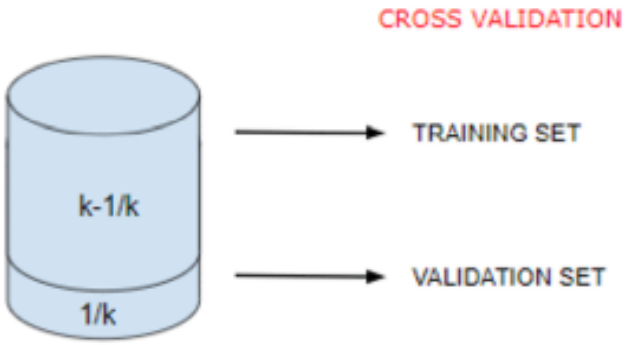

Figure 25 - Cross validation subdivision (13)

During every cycle the validation set is used to carry out the testing phase obtaining two score coefficients vectors, containing the accuracy of the validation set and the training set. They are:

- **CV train score**
- **CV test score**

This test is important for assessing model convergence and assessing loss function during training and validation. The convergence is obtained if the loss function turns out to have a decreasing course regarding the number of iterations that are carried out by the model.

In the algorithm developed within this thesis, the hyperparameter optimization algorithm used is **GridsearchCV**. This function present in the Scikit learn library of python, performs the tuning, thus maximizing the performance of the model. It is an estimator capable of choosing the best combination of hyperparameters from an input list. After testing the model in all possible cases, it returns the values of the hyperparameters for which the loss function is minimized. In addition, the latter is coupled with the K-cross Validation process.

A very useful representation for the evaluation of the model is that of **Learning Curves**.

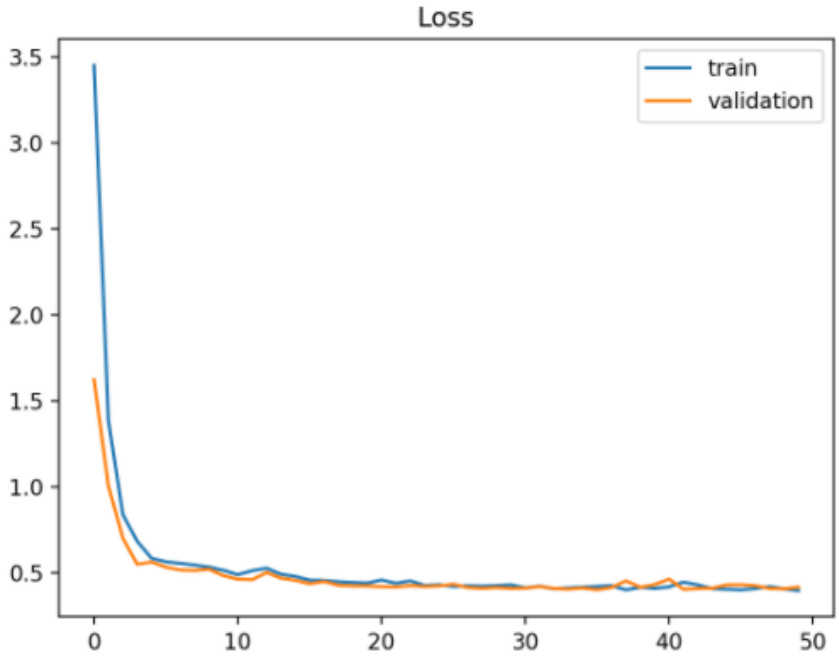

Figure 26 - example of Good Fit of Train and Validation Learning Curves (14)

The diagram represented, introduces the evolution of the model in the time and along the number of iterations on the x axis, while on the y axis the value of the loss function is represented.

The 2 curves in the graph represent:

- **Training Learning Curve:** Learning curve calculated from the training dataset, gives an idea on how the model is learning
- **Validation Learning curve:** Learning curve calculated from the validation dataset, which gives an idea of how well the model is generalizing

A good fit is identified by a decrease of the two curves up to the same point of stability, keeping a small gap between them until convergence is reached.

However, it is not always possible to obtain these results, due mainly to the phenomena of **Overfitting** and **Underfitting**.

Learning curves of model performance can also be used to diagnose whether or not the train and validation datasets are representative of the system.

An unrepresentative Training Set means that it does not provide enough information to learn the problem.

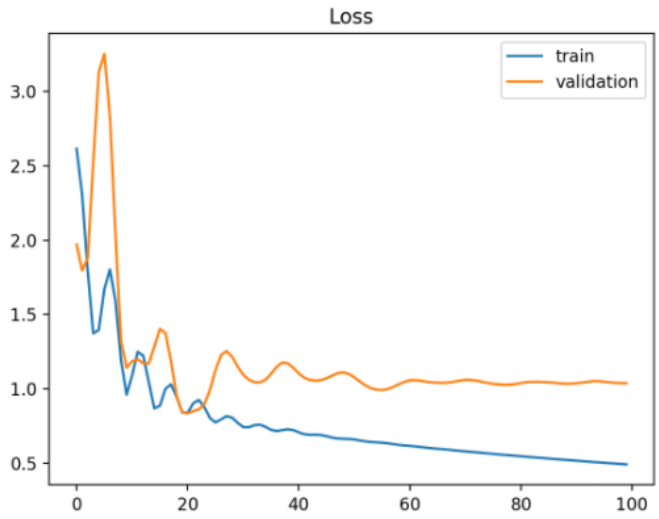

Figure 27 - Learning Curves with unrepresentative Training Set (14)

This phenomenon is identified by a divergence of the two curves that grows over time and by some spikes.

On the other hand, an unrepresentative Validation Set means that it does not provide enough information to generalize the problem.

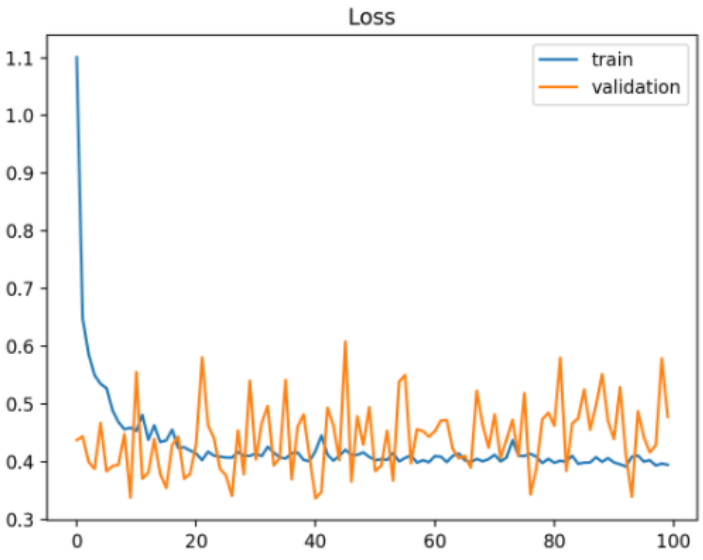

Figure 28 - Learning Curves with unrepresentative Validation Set (14)

This phenomenon is identified by a training learning curve with a good fit and a validation curve that shows many spikes. (14)

2.3.4 - Overfitting and Underfitting

Following the training phase, the algorithm should be able to efficiently predict the cases present within the test set. However, there may be cases in which the training phase is not carried out properly and the hyperparameters are not correct for the resolution of the problem. When the training phase develops due to an excessive number of iterations, the function that connects input and output is incorrect. This phenomenon is called Overfitting.

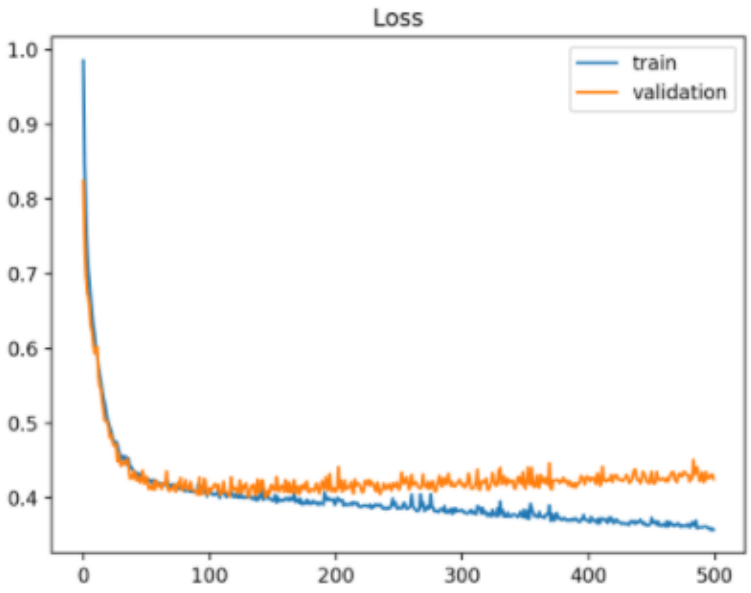

Figure 29 - Overfitting in Learning Curves (14)

On the other hand, if the approximation function has an inaccurate fit or there is a small number of samples, we also obtain bad predictive results. This is called Underfitting.

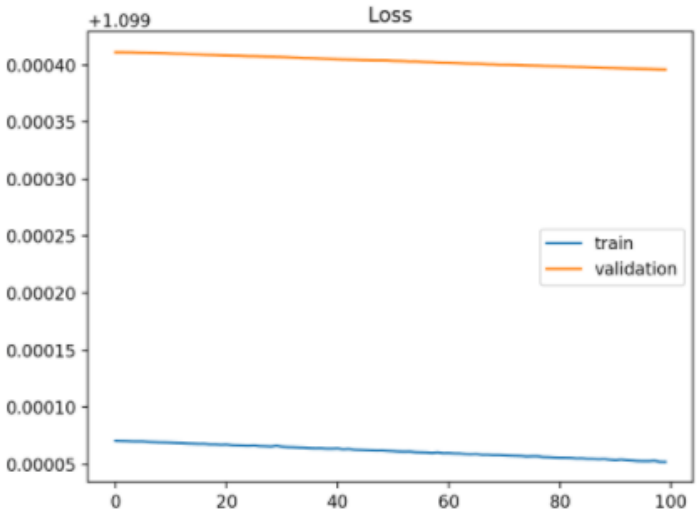

Figure 30 - Underfitting in Learning Curves (14)

A classic example of this phenomena is present in figure 31 below.

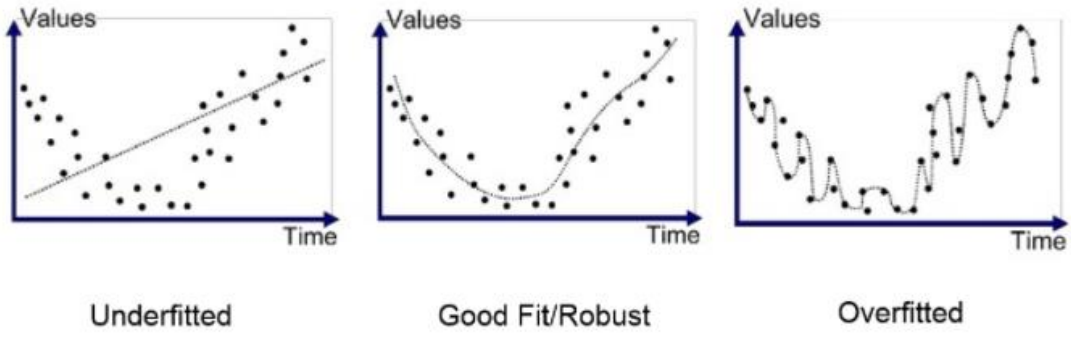

Figure 31 - Representation of different fit curves (15)

For these reasons, it is of great importance to:

- Give a correct degree of the polynomial to a regressive function
- Meticulously check the dataset
- Correctly choose the hyperparameters
- Make a proper feature selection

Compared to underfitting, overfitting is not simple to recognize and, in most cases, it may be solved by adopting a different algorithm or directly acting on its hyperparameters.

2.4 - Feature Selection

The feature selection process is a technique widely used to detect the most significant features within our dataset for label prediction. This allows us to reduce the number of features and to train the model only on the most important ones for our problem. In this way we can reduce the phenomenon of overfitting or redundant data, increase accuracy and decrease the computation times of our model. There are several methods to apply this technique but for the present thesis I decided to use the XGBoost algorithm, based on the Decision Tree method.

2.5 - Decision Tree

The Decision Tree is one of the most common classification and regression techniques used within the ML framework. A decision tree shall consist of the following elements for the decision-making action:

- **Root Node**: Represent the node at the treetop, containing all the sample that will be divided in the decision process
- **Internal Nodes**: Used to make any decision and make multiple branches
- **Branch/ sub tree**: A tree formed splitting the tree
- **Leaf Nodes**: Final output node, containing the predicted value for the target variable (16)

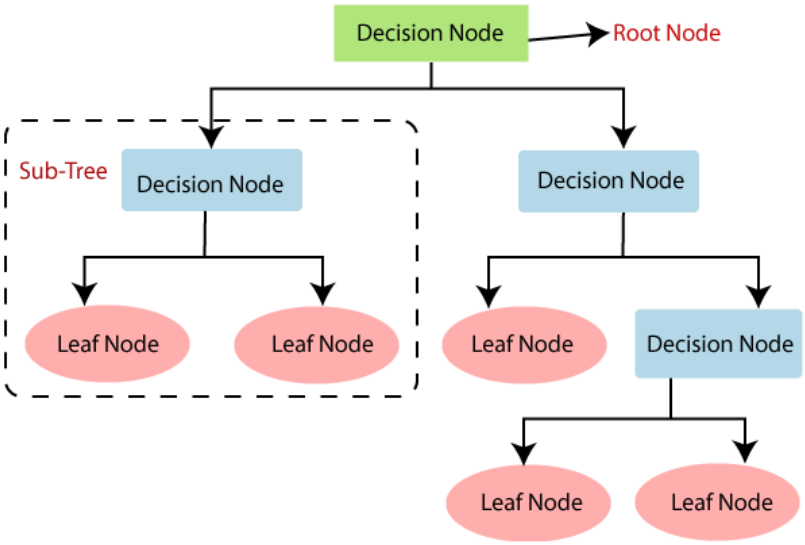

Figure 32 - Decision Tree Structure (16)

The tree operating with several blocks and nodes is able to take a set of data and extract a set of rules in order to understand the problem. For the construction of the tree, we start by providing the algorithm with the dataset containing the different features and the label. The data are divided each time into nodes according to the values assumed by the features. Then the dataset is subdivided into subsets, which are passed through the first branches of the first decision node: if the analyzed data meet the same condition set of the feature, the leaf node is reached, otherwise we proceed to another decision node where they are divided according to the condition of the new node. Each leaf node defines a certain area in which the new target variables will be evaluated.

The main advantages in using this technique are: the easy understanding of the results, the computational speed and the fact that the algorithm takes into account all the possible outcomes of the problem. On the other hand, the decision tree is prone to overfitting, so you must be very careful in its use. To avoid this problem, it is necessary to limit the freedom of the decision tree during training. This process is regulated by the tuning of some hyperparameters. (10).

CHAPTER 3 – Deep Learning

The main algorithms and machine learning models are efficient for solving many problems. However, when dealing with problems with a large number of data or multidimensional ones, it is necessary to use algorithms belonging to the category of Deep Learning. The main foundation of the DL is the use of multilayer architectures and neural networks. Thanks to them, predictive algorithms can operate with high numbers of features, reducing computational cost and achieve excellent results.

3.1 - Artificial Neural Networks (ANN)

Let's analyze the main characteristics of neural networks and models of computation present in them. Artificial neural networks were created for the purpose of reproducing the activity of the human brain. The fundamental unit of our brain is the neuron and they communicate with each other through electrical impulses. This element is therefore able to receive different input signals from Dendrites, rework them and produce in turn new output signals which exit from Axons.

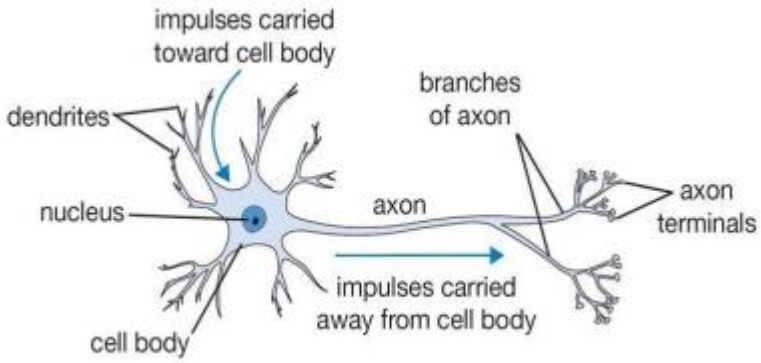

Figure 33 - Neuron Structure (9)

It is in this way that the artificial neural network also operates in which, the activity of the human neuron is replicated by the perceptron as the fundamental unit of the NN. A

series of perceptrons form a single layer of the NN. The first layer of a neural network is called input layer because it contains all the information from the dataset features. Then this information is processed and sent to the different neurons of the next layer, up to the output layer containing the target of our problem. In this similarity, any information from a given perceptron is appropriately weighed with the use of a coefficient; in addition, each layer also contains a bias factor that stabilizes the model.

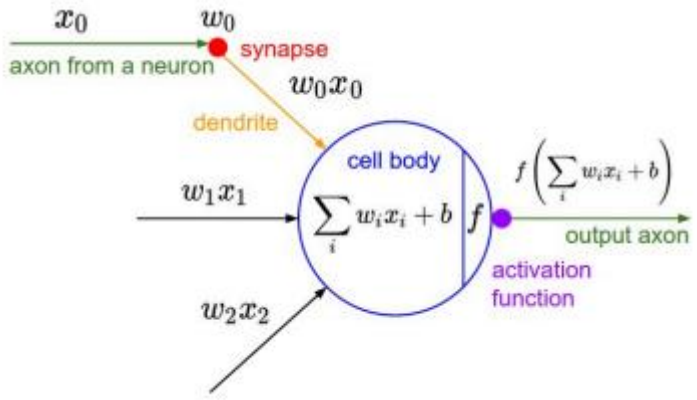

Figure 34 - Perceptron Structure (9)

Let us now analyze the behavior of a single perceptron and the various parts that compose it.

- $\boldsymbol{x_i}$ represents a node and the information which enters in the perceptron
- $\boldsymbol{x_0}$ is the Bias term
- $\boldsymbol{w_i}$ represents the weight that characterizes each connection
- $\boldsymbol{f}$ is the activation function contained in the perceptron
- $\boldsymbol{h_w}$ $(\boldsymbol{x})$ is the output of the perceptron

$$h_w(x) = f(\, x_o + w_1 x_1 + w_2 x_2 + w_3 x_3 + \cdots + w_n x_n \,)$$

$$h_w(x) = f(\, x_0 + \sum_{i=1}^{n} w_i \, x_i \,)$$

The perceptron therefore adds in a weighed way all the contributions in input and the bias term. Then, through the activation function f present in it, it processes its output.

There are also different types of neural networks depending on their conformation and how neurons communicate with each other. In our case we will consider only Fully connected Feedforward neural networks. In this type of network, the information flows from the input layer to the output layer and each neuron of the layer "j -1" is connected to all neurons of the layer "j". (9)

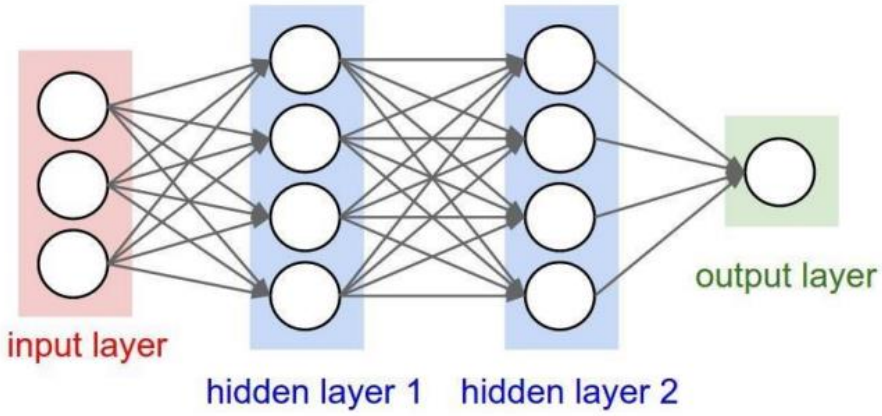

Figure 35 - Feedforward fully connected neural network (9)

3.2 - Code description

As previously described, the purpose of this thesis is to generate a machine learning code capable of predicting the emissions of Soot Tail Pipe coming from a compression ignition (CI) engine. This process is possible using a regressive artificial neural network that works in a supervised environment. The algorithm was implemented in python. The latter is a very useful and powerful programming language composed of several libraries containing pre-scripted algorithms. The libraries are constantly updated and improved by users; they contain functions for solving many problems belonging to different fields of application. In

the field of machine learning this program is particularly used, especially for programming artificial neural networks. The following libraries are used in the code:

- Numpy
- Tensorflow
- Pandas
- Scikit-Learn
- Keras
- Matplotlib

3.2.1 - Pre-Processing

First of all, we should remember that the predictive model works on some excel datasets containing different measurements corresponding to different engine operating points. For each engine point, numerous variables are reported corresponding to the main parameters characterizing the combustion process and therefore the emission. The python code reports the data matrix in the program and then works on the development of the predictive model. To make the algorithm as deductive and simple as possible, it has been divided into several functions and each of them has a specific task. In addition, the use of this technique leads to high time saving during the analysis especially because:

- it allows to modify the dataset considered in a simple way
- during programming it allows you to detect errors in a simpler way
- it makes the code more orderly and simpler in its use

three main codes with separate tasks have been developed. The first code (**Feature_Selection_TP**) is used to perform the feature selection process for the given dataset, then to detect the features of greatest relevance for the prediction of the Soot Tail Pipe label. The second one (**Model_results_TP**) returns the scores and the different plots of the neural network, using the data belonging to the same folder through an appropriate train_test_split as train and test sets. Finally, the third code (**Train_Test**) returns the scores

of the neural network by applying the training and testing phase on two different folders and so, taking two different excel inputs. These three codes recall or use other sub-functions that are important for the correct functioning of the model. The subfunctions used are listed here:

- **Dataset_elimination_TP**

  Function used to eliminate and filter the outliers present inside the dataset. Given that the measurements taken have a certain level of precision, it was decided not to use a numerical filter, but simply to bring to 0 all measurements with negative values of Soot_EO or Soot_TP.

- **Dataset_division_TP**

  Function used for separating dataset into features and label.

- **Normalization_TP**

  Function used to perform dataset normalization using max/min Normalization.

  Normalization is used in preprocessing, making the data more suitable for convergence and ensuring the stability of the model thus avoiding overfitting.

$$Z = \frac{x - \min(x)}{\max(x) - \min(x)}$$

First, an analysis of the dataset is carried out to verify that it does not contain any anomaly or measurement error that could negatively affect the model calculations. After splitting the dataset into features and labels, the data are properly mixed through the shuffle function in the sklearn.utils library.

```
From sklearn.utils import shuffle
dataset = shuffle (dataset)
```

The use of the shuffle is vital to verify the efficiency of the model and not always consider the same data, in fact each analysis has been repeated 10 times to avoid incurring errors creating a variability in results and a more complete evaluation of the process.

After that, the dataset must be divided into train set and test set with an appropriate proportional division. To do this we use the train and test split function in the Sckit-Learn library. It has been used in the following way:

```
From sklearn.model_selection import train_test_split
test_size_TP = 0.2 ( Test set = 20% Dataset )
X_trainTP, X_testTP, y_trainTP, y_testTP = train_test_split(VarEngineATS_TP, scotTP,
test_size=test_size_TP)
```

As shown above, a suitable train test split is 80% of the data for the training phase and 20% of the data for the test set. Nevertheless, to verify the stability of the model and the efficiency of the neural network it is important to obtain high network scores even with lower percentages of the training set.

Finally, we pass to the data normalization to eliminate the variability of the data and the fact that they can be on different orders of magnitude due to different units of measurement. The use of data of different orders can cause anomalies within the network as some features would be privileged compared to others. The min max Normalization is efficient as it reports the values of all the features to a number between 0 and 1 then easily manageable and measurable by the neural network.

Now we move on to the feature selection and FFNN construction phases.

3.2.2 - Feature Selection

As mentioned above this process allows to understand which are the most relevant features for the predictive model and the concerned dataset. The main benefit of this technique is that the algorithm will not have to train on all data but only on those which are important for its operation. In addition, the feature selection allows us to speed up the training time of the network, increase accuracy and decrease the possibility of overfitting as it

eliminates redundant data. The Feature_selection_TP code provides the results from the Feature_Extraction_TP sub-function. The latter is important because through the use of the command window of python, it allows us to start the decision process and choose whether to use the XGBoost algorithm or the Random Forest algorithm for the selection. These algorithms, as explained above, are based on the decision tree and return the percentages of importance that are attributed to the different features. In this thesis XGBoost has been used as feature selection algorithm because it is more suitable for solving the Soot Tail Pipe predictive problem. It stands for Xtreme Gradient Boosting and it exploits a boosting ensemble technique, combining different decision trees in a unique system.

XGB function is managed by the following hyperparameters:

- **Learning Rate**: Indicates the boosting step used to prevent overfitting. Range [0,1].
- **Max_Depth**: Indicates the depth of each decision tree. Range [0,∞].
- **N_estimators**: Number of trees on which the model will operate. It would be the equivalent number of boosting rounds.
- **Min_child_weight**: Minimum number of blocks of each branch. Range [0, ∞].
- **Gamma**: Minimal loss reduction needed to make an additional partition on a node tree leaf. The higher the range, the more conservative the algorithm will be. Range: [0, ∞].
- **Colsample_bytree**: is the ratio of subsampling the columns during the construction of each tree. Subsampling occurs once for each mast built. Range:]0,1].

Within the code, I combined this algorithm with the GridsearchCV process to allow the optimization of the model's hyperparameters. Such process takes place by combining the data entered manually for every hyperparameter, until it finds the best combination. The evaluation of the models is carried out through a cost function or a score. The part of the code used for this process is as follows:

```
if est_selected == 'xgb' :

        params ={'learning_rate':[0.01,0.001],
        'max_depth':[10,6,8],
        'n_estimators':[2500,3000,3500],
        'min_child_weight': [8,10],
        'gamma': [0.05],
        'colsample_bytree':  [1.0]}
        grid_estimator = XGBRegressor()
```

At the end of the k-fold process, we obtain a division of the incoming data in different folders and for each of them we obtain a **cv_score** value, which represents the efficiency of the process. In addition, the use of shuffle allows to achieve higher scores so it is a further proof of the benefits it brings to the model.

3.2.3 - Model Construction

After the first phase of pre-processing, we build the neural network through the sequential model technique. The function used to calculate the network score is Model_results_TP. The latter takes in input the train set and the test set suitably divided and normalized, after that it relies on the Model_Formation function to carry out the processing of the artificial neural network. Using the sequential model, I can define from the input layer all the characteristics of the different layers of the network up to the output one.

```
model = Sequential()

  model.add(Dense(120,
        input_shape = (X_trainTP.shape[1],),
        activation = 'relu',
        kernel_initializer = 'normal'))
```

```
model.add(Dropout(0.1))
.....

model.compile(loss = 'mse',
        optimizer = Adam(lr = 0.001),
        metrics = ['mse', r2_score])

history = model.fit(X_trainTP,
            y_trainTP,
            epochs = 120,
            validation_split = 0.2,
            shuffle = False,
            batch_size = 250)
```

The main parameters that outline the functioning of a neural network are the following:

- **Neurons N°**
- **Hidden Layers N°**
- **Batch_Size**: corresponds to the number of samples used by the algorithm before changing the internal parameters of the model.
- **Epochs**: defines the number of iterations for which the model will need to be trained
- **Optimizer**: Associated with the learning rate used by the network to optimize its functions.
- **Activation Function**: it indicates the function present within each neuron
- **Kernel_initializer**: it defines how the weights for each epoch (iteration) of our neural network are initialized
- **Validation_split**: Indicates the percentage of the train set that is used for the validation process.

- **Dropout**: this is particularly useful to avoid the phenomenon of overfitting as it indicates a certain percentage of neurons belonging to the current layer that are randomly ignored during the flow of information by the model.

Following the network training process, I extrapolate the values of the error functions and continue with the testing phase.

```
y_trainTP_pred = model.predict(X_trainTP)
y_testTP_pred = model.predict(X_testTP)
```

Then we proceed with the denormalization:

```
def denormalize(y, y_max, y_min):
    final_value = y * (y_max - y_min) + y_min
    return final_value
```

Finally, I evaluate the model by analyzing different plots and the network scores in the form of: MSE, RMSE and R^2, returning to the main function Model_results_TP.

```
r2_trainTP = r2_score(y_trainTP, y_trainTP_pred)
r2_testTP = r2_score(y_testTP, y_testTP_pred)

MSE_trainTP = metrics.mean_squared_error(y_trainTP, y_trainTP_pred)
MSE_testTP = metrics.mean_squared_error(y_testTP, y_testTP_pred)

RMSE_trainTP = np.sqrt(MSE_trainTP)
RMSE_testTP = np.sqrt(MSE_testTP)
```

The above procedure can also be performed using two different datasets such as train sets and test sets with the Train_Test function. The latter is very important for the cross-analysis that will be carried out to verify whether the different datasheets can be physically

comparable and whether the logic of the model can work simultaneously with different folders.

3.2.4 - Tuning of the Hyperparameters

Thanks to the combination of two different codes: tune_params and tune_params_results we can achieve the tuning of the neural network hyperparameters and the optimization of the system. Through these two functions, the GridsearchCV process from the Sckit-Learn library is once again recalled combining the different parameters of the neural network in such a way as to minimize the loss function and optimize the system. This code has some limitations as it would be appropriate to indicate the values of the hyperparameters for each layer of the network, while the current model allows only to provide a unique combination of the hyperparameters for the whole network and all the layers which compose it. Reprogramming this code would make it possible to obtain stronger network parameters that would allow further optimization of the latter.

```
param_grid = dict(batch_size = [250,200],
                  epochs = [120,100],
                  activation = ['relu'],
                  loss = ['mean_squared_error'],
                  init_mode = [ 'normal'],
                  dropout_rate = [0.1,0.2],
                  neurons = [200,250],
                  learn_rate = [0.001,0.01],
                  n = [1,2])

  grid = GridSearchCV(estimator = model,
                      param_grid = param_grid,
                      scoring = 'neg_mean_squared_error',
                      n_jobs = -1,
```

```
                refit = True,
                cv = 5,
                verbose = 2)

grid_result = grid.fit(X_trainTP, y_trainTP)
```

CHAPTER 4 – Data Analysis

In this section the observations and the analyses carried out on the data present in the various folders provided by AVL Italia are studied. The several datasets contain many samples coming from some tests carried out on a roller bench to which have been applied many sensors including those for the measurement of Soot both in the engine out position and in the tail pipe one. In this thesis work, two datasets measured in stationary conditions and one on the WLTC regulatory driving cycle are taken into consideration.

4.1 - Variables Pre-Processing

Our aim is to build a virtual sensor capable of predicting the Soot_TP and as I have previously explained this measure is influenced differently by all the variables. For this reason it is very important to observe the performance of the different features and see whether their behavior is physically logical.

The engine parameters measured during the tests describe: the engine operating conditions, the DPF conditions and the environmental conditions. There are 21 of them, including the Soot Engine Out and the Soot Tail Pipe. The following table shows the different features of the system used to predict the Soot_TP.

Time [s]	Vehicle speed [km/h]	Engine speed [RPM]	Engine Coolant Temperature [°C]
Injected quantity [mm^3/hub]	EGR Rate [/]	Environmental Temperature [°C]	DPF Upstream temperature [°C]
Environmental pressure [hPa]	DPF delta pressure [hPa]	Intake manifold pressure [hPa]	DPF Downstream temperature [°C]
Lambda [/]	DPF Soot Mass [%]	TLC_Concentration_EO_from_MSS_INCA [mg/s]	Sensed Intake fresh air [kg/h]
Intake manifold temperature [°C]	Engine Mode [/]	Volume Flow rate across DPF [m^3/s]	Indicated air Mass Flow [kg/h]

The label is indicated as TLC_Concentration_TP from_MSS_INCA (Soot_TP) and it is measured in [mg/s].

Inside the table there are two red features. The measures in question are those of Time and Engine Mode; both are not considered for the prediction of Soot_TP and then they are eliminated. This decision is because: The Engine Mode has a constant trend within each dataset and it does not represent a discriminating feature for the network, while time does not represent a physical variable for the system. As mentioned above, the neural network considered as a feed forward one does not take in account the temporal evolution of the system. On the latter aspect it would be interesting in the future to take this data into account to verify its effectiveness in predicting soot.

It is important to point out that the choice not to eliminate other features entering into the feature selection process is due to the fact that all the remaining features represent a possible factor of relevance for the prediction of a complex phenomenon such as the formation of soot.

The datasets taken into analysis are the following:

- **File 1 ➔ DATASET 1:**
  N°SAMPLES = 134.365
  FREQUENCY RESOLUTION = 10 Hz
  STATIONARY TEST AT DIFFERENT ENGINE SPEED FROM 800 RPM TO 4500 RPM
  BPU 20x20
  EGR SWEEP

- **File 2 ➔ DATASET 2:**
  N°SAMPLES = 36.658
  FREQUENCY RESOLUTION = 10 Hz
  STATIONARY TEST AT DIFFERENT ENGINE SPEED FROM 800 RPM TO 4500 RPM
  BPU 20x20
  NOMINAL EGR

- **File 5 ➔ DATASET WLTC:**
  N°SAMPLES = 18.346
  FREQUENCY RESOLUTION = 10 Hz
  WLTC TEST AT DIFFERENT ENGINE SPEED FROM 0 RPM TO 2500 RPM

Initially the data was analyzed both according to the number of samples and according to the number of revolutions of the engine. In a first approach several filters had been placed to the various features of the system but after numerous meetings with the engineers of AVL

Italia, such filters were eliminated because every measure present in the datasets represents an effective behavior of the engine along the map. In fact, although some points may seem suspicious looking at the trend of the features, they represent the precise behavior of the engine, so the measurements of Soot_TP in these points cannot be ignored.

Nevertheless, the only important filter we have decided to maintain is the one applied to Soot_EO and Soot_TP variables. Within all five folders there are some negative values of these features. For this reason, I reported these values to 0 without eliminating the sample in analysis. This decision is outlined by the fact that removing these points would mean again not considering some samples that represent the operation of the compression ignition engine. In the following image there is an example of some negative values in the Soot_TP measurements present in the Dataset 1.

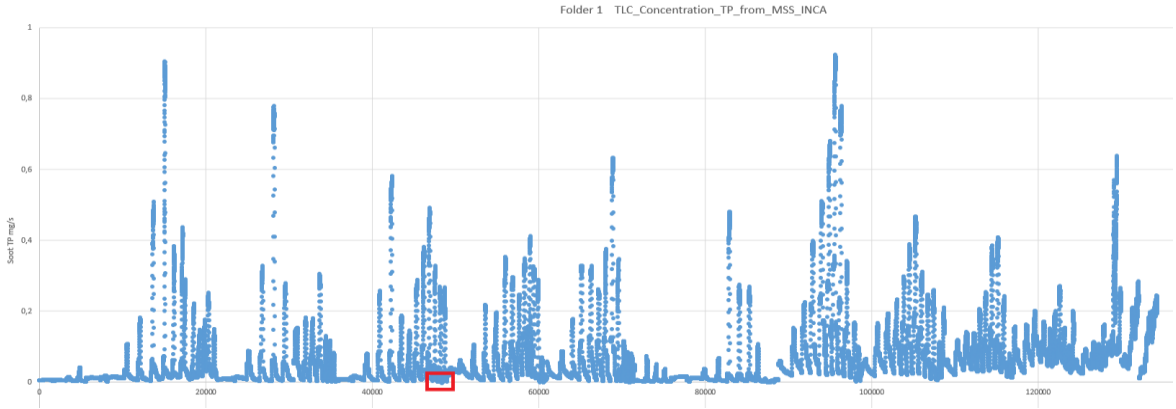

Figure 36 - File 1 Soot_TP plot

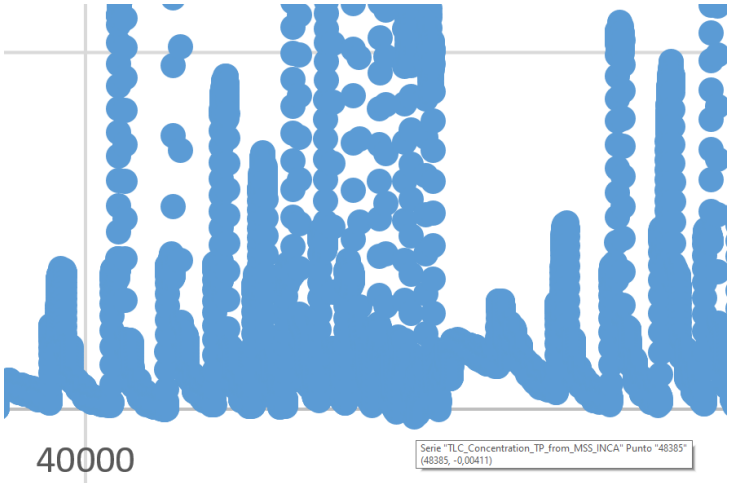

Figure 37 - File 1 Soot_TP Zoom

It is again important to point out that the measurements that are present in steady conditions, and therefore carried out on the first two datasets, are all carried out on the same engine on a roller bench. These also have the same size as BPU 20X20. This dimension indicates the attack size of the DPF, and it is relevant for the comparison of the two folders even if the phenomenon of regeneration is not treated in stationary.

As we will see, regeneration is only present in samples made on the WLTC driving cycle. For this reason, the measurements of soot, present in File 5, will be several orders of magnitude smaller than the values presented in the stationary spreadsheets.

4.2 - Dataset 1

File 1 is the largest dataset among those analyzed and represents a series of stationary measurements made in the presence of EGR sweeps. This means that during fuel injection, the EGR is not kept at constant levels but follows ramps.

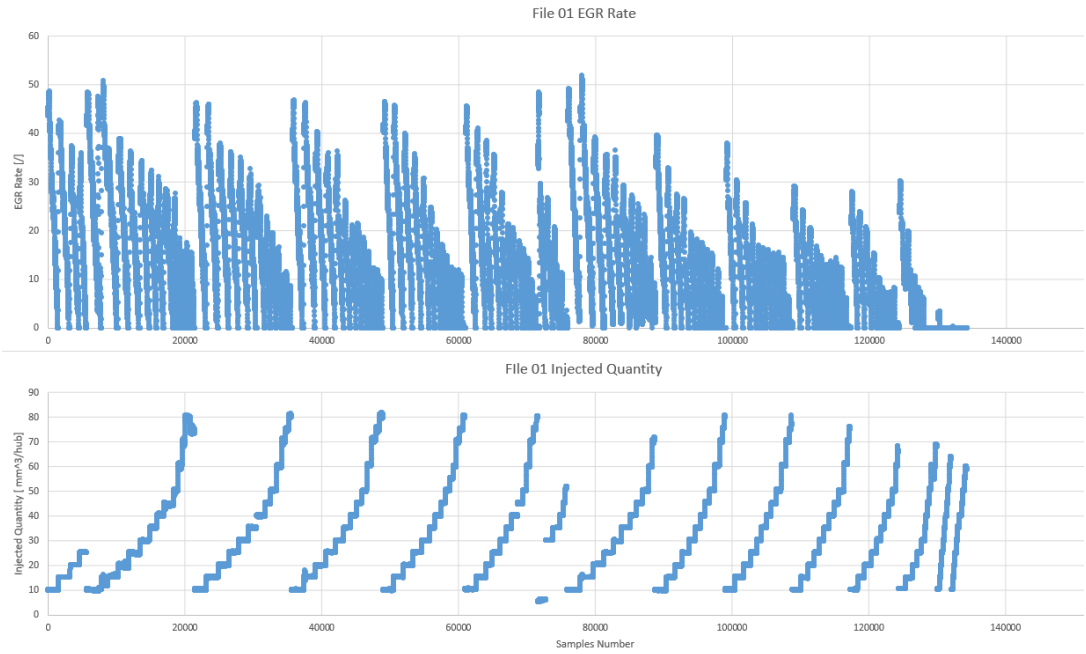

Figure 38 - File 1 EGR Rate and Injected Quantity

First, a check of the different features was made to control that there is no anomaly between them. The engine operates at an environmental temperature of around 25°C with an engine speed following a steps course between 800 and 4500 rpm.

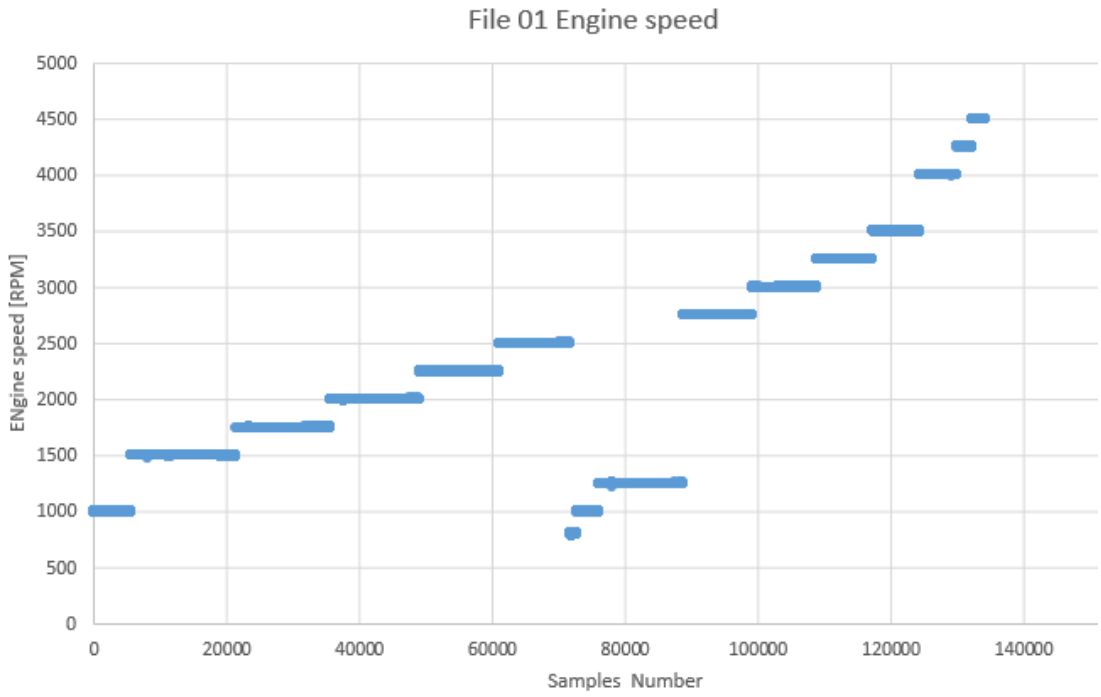

Figure 39 - File 1 Engine Speed

The inlet and outlet temperature of the DPF varies never exceeding 700 °C.

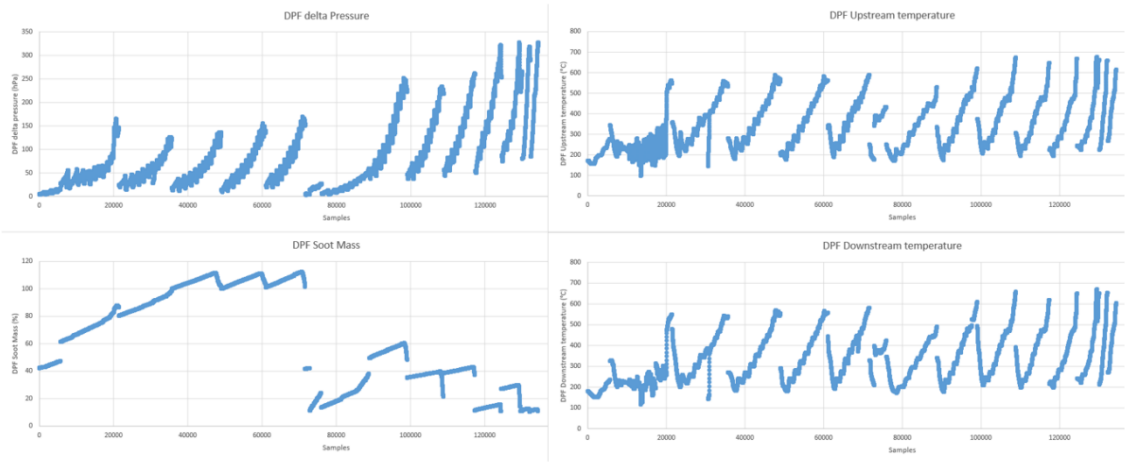

Figure 40 - File 1 DPF Features

The measurements of the DPF Soot mass have values higher than 100%, which is physically senseless. Despite this, discussing with the engineers of AVL we concluded that this feature follows a logical trend in the operation of the engine so it means that with the use of normalization, such features can be maintained and taken in consideration for the analysis of File 1.

Figure 41 - File 1 DPF Soot Mass

Analyzing the performance of the 3 features: Volume Flow rate across DPF, Sensed Intake fresh Air and Indicated air Mass Flow, it is possible to notice a redundancy in the performance of the features that must therefore be kept under control.

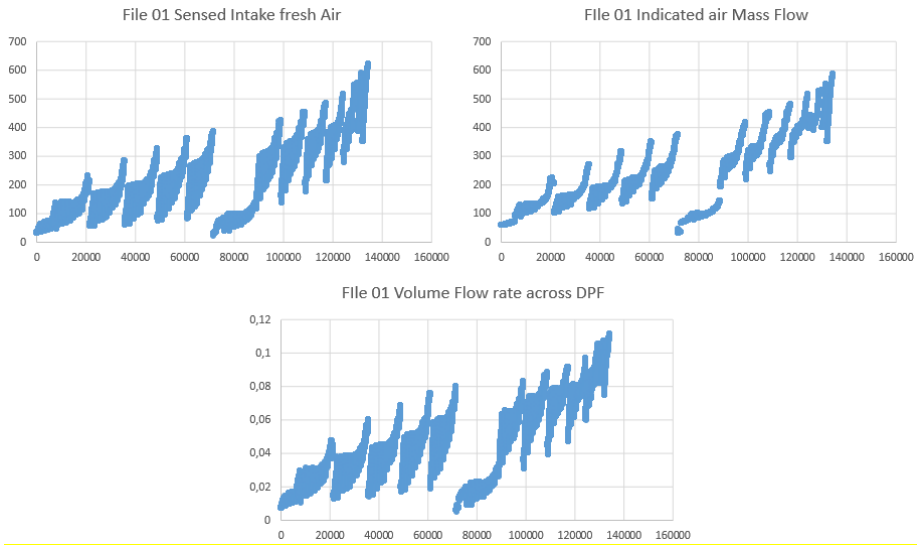

Figure 42 - File 1 Air Flow features

Finally, the trend of Soot_EO is very similar to the trend of Soot_TP even though with different orders of magnitude, so we expect it to fall into the most important features of the predictive system.

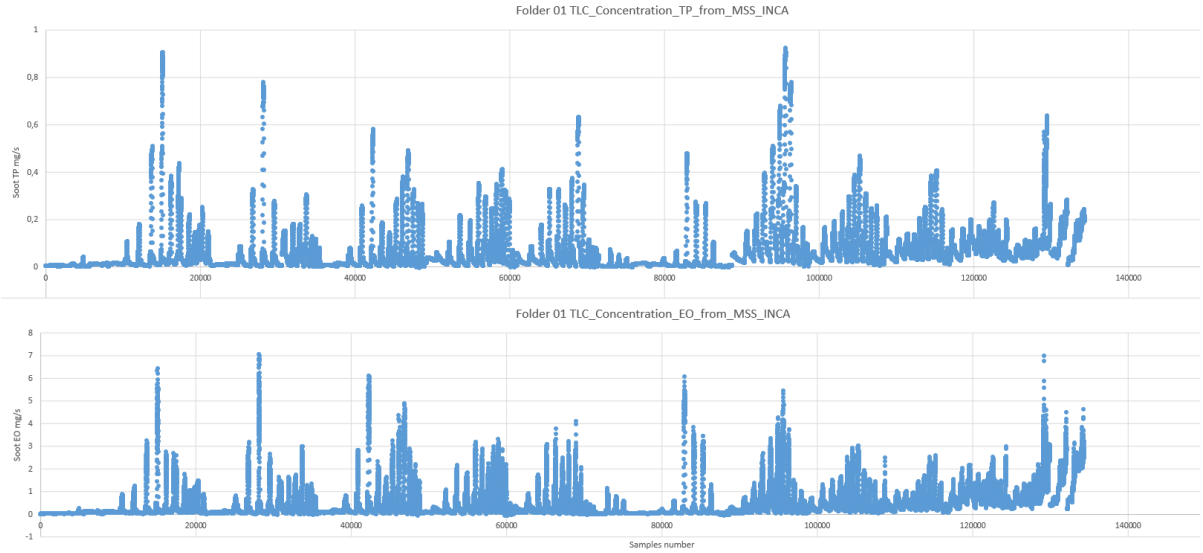

Figure 43 - File 1 Soot_TP and Soot_EO

Now, the Feature Selection process is carried out using the algorithms described above. In this case, thanks to XGBoost we assign a percentage of importance for the prediction of Soot_TP to each feature. At this point we will only consider the most important features up to reach 90% as the sum of the different percentages of importance. It must be remembered that upstream of this process the shuffle was performed and a train_test_split of 0,2 was imposed. This means that the train set will have a size equal to 80% of the total amount of data. The following are the results of this process applied to File 1.

Main Features	Importances [%]
TLC_Concentration_EO_from	54,187
DPF Soot Mass	11,957
Indicated air Mass flow	7,436
DPF delta Pressure	3,675
DPF upstream Temperature	2,472
DPF downstream temperature	2,411
Intake manifold temperature	2,351
Environmental temperature	2,342
lambda	2,067
Engine coolant temperature	1,908

Features Excluded	Importances [%]
Volume flow rate across DPF	1,498
Injected quantity	1,366
Engine speed	1,245
EGR rate	1,206
Vehicle speed	1,197
Intake manifold pressure	0,945
Environmental pressure	0,87
Sensed Intake fresh air	0,65

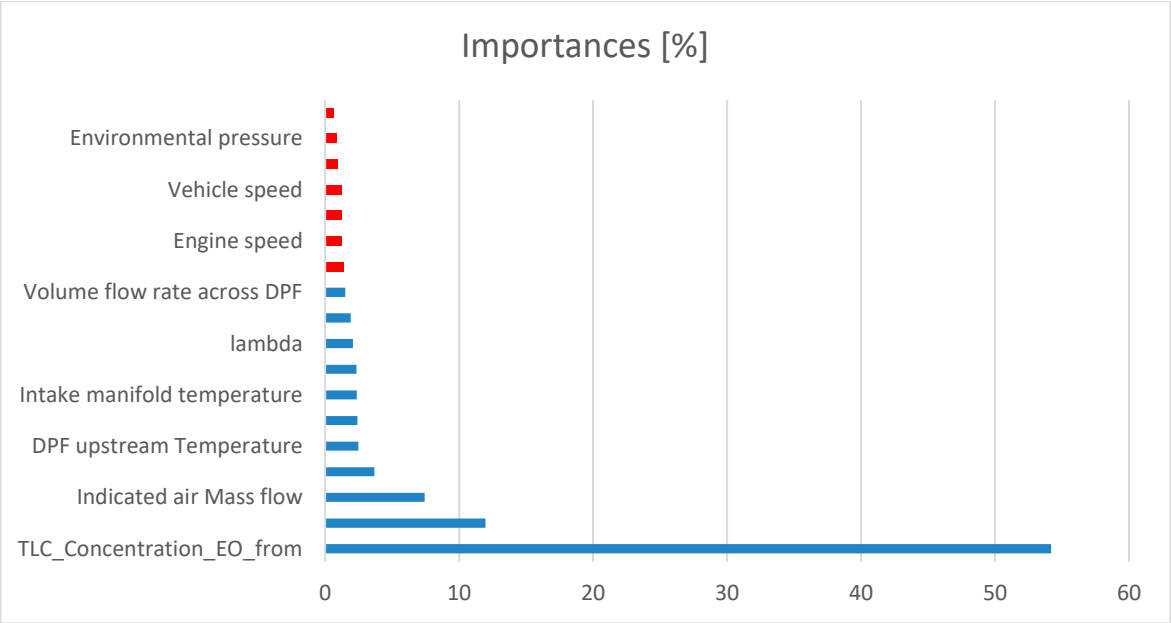

Figure 44 - File 1 Feature Importance Graph

As was expected, among the most important features we can find the Soot_EO and the DPF Soot mass. This behavior was desirable as these variables have a direct connection with the Soot present at the engine discharge. In addition, all the features regarding the DPF are also present in the analysis. As a third feature of greater importance, we can observe the indicated air Mass flow, which among the three redundant features that indicate the air flow, is the only one considered by the system. Here the different parameters at the output of the GridsearchCV process are reported, containing the coefficients of determination (cv_score) obtained during the analysis of the values of the XGBoost parameters linked to them.

Hyperparameter	DATASET 1
Learning rate	0,01
Max_depth	5
N_estimetors	1500
Min_child_weight	6
gamma	0,05
Colsample_bytree	1

CV_SCORE	*R2*
0	0,983
1	0,979
2	0,981
3	0,978
4	0,979
5	0,979
6	0,981
7	0,979
8	0,981
9	0,98
MEAN VALUE	0,98

4.3 - Dataset 2

File 2 contains a series of measurements carried out on the same vehicle in stationary conditions but without the EGR sweeps mentioned in the previous dataset analysis. The performance of these features is maintained at a nominal value with respect to fuel injection. Even within this dataset, the measurements are carried out at an environmental temperature of 20° C in the same Engine Speed range.

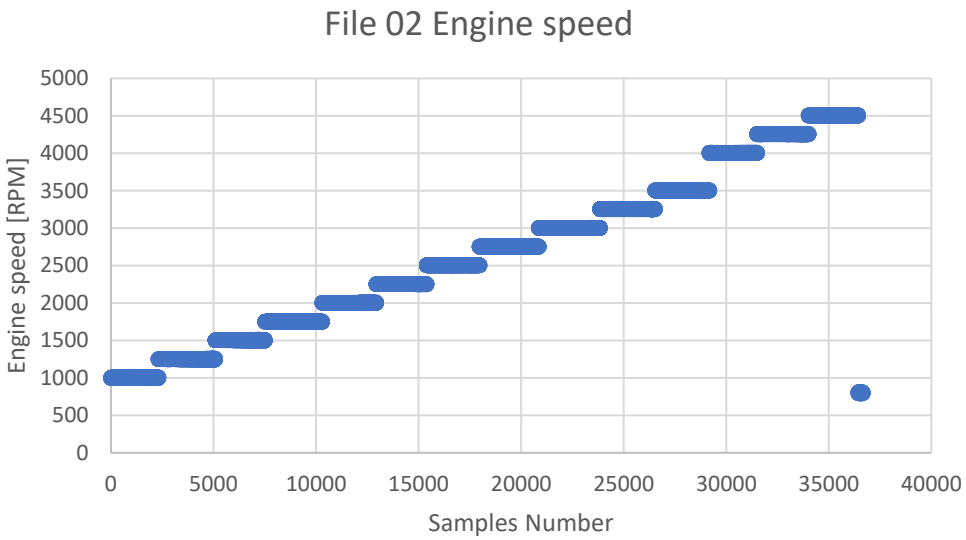

Figure 45 - File 2 Engine Speed

Also, all the features of the DPF respect the standards of specification and as can be seen the DPF Downstream Temperature follows the Upstream one in a proper way.

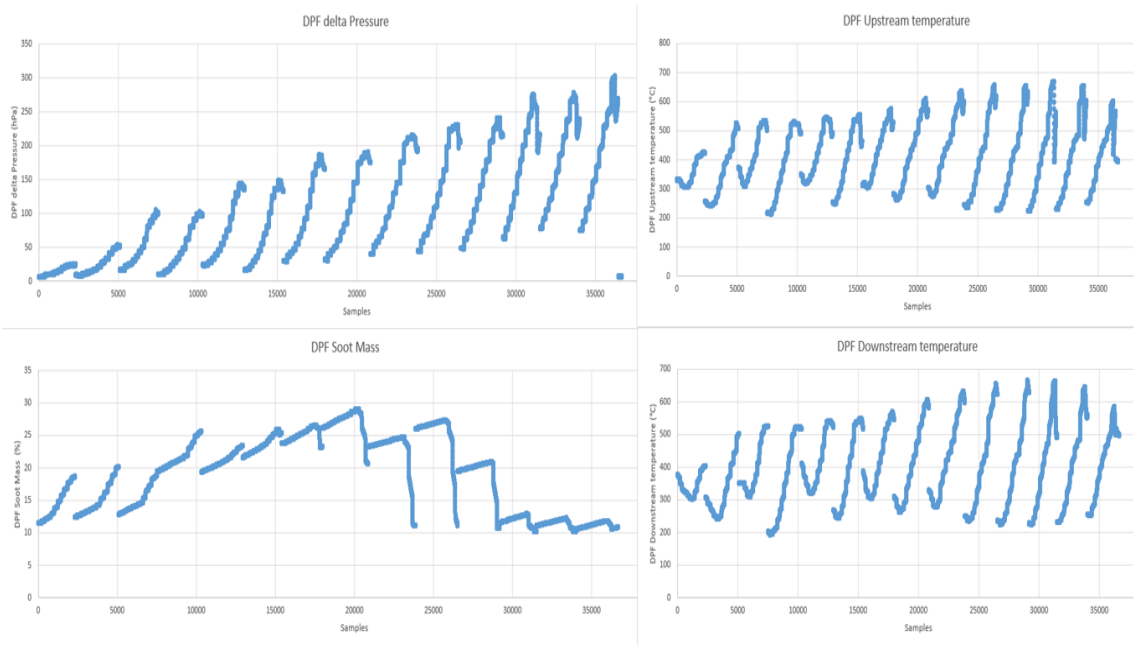

Figure 46 - File 2 DPF Features

The trend of Soot_EO reflects that of Soot_TP and we have the same redundancy case for the three features: Indicated air mass flow, Volume flow rate across DPF and Sensed intake fresh air as in the previous case.

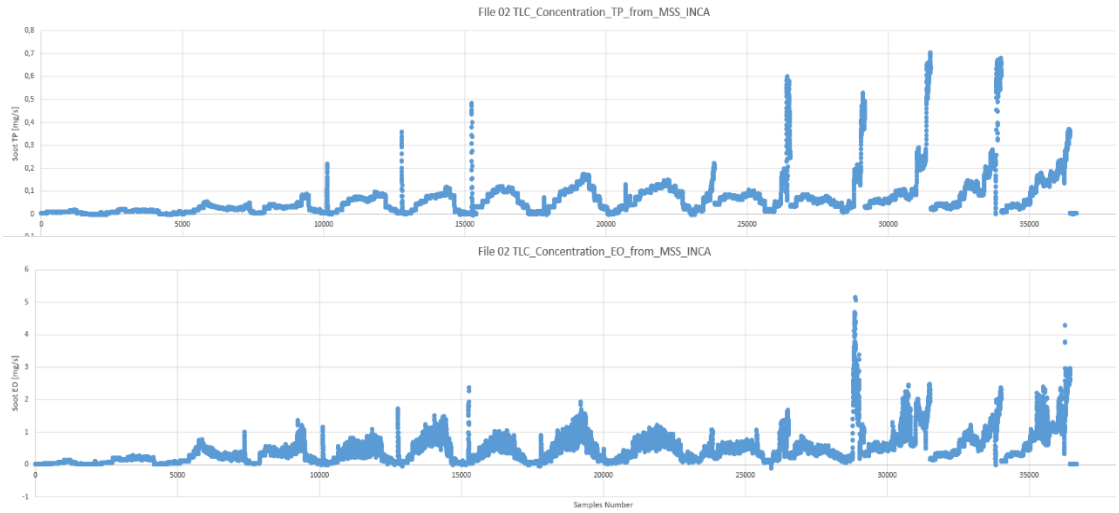

Figure 47 - File 2 Soot_TP and Soot_EO

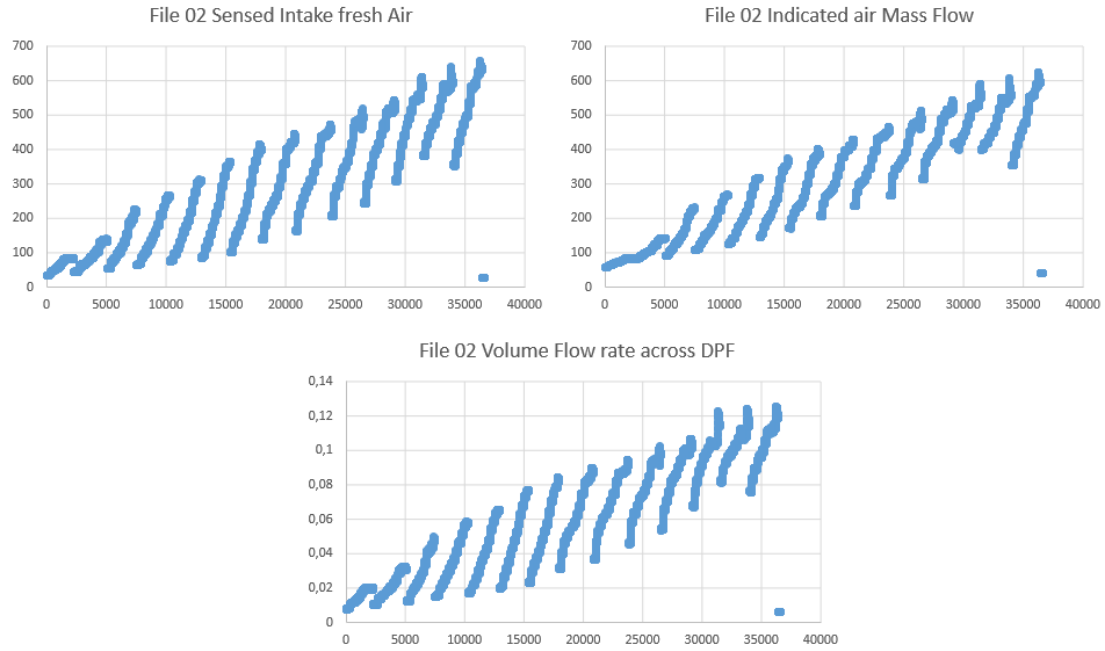

Figure 48 - File 2 Air Flow Features

The following are the results of the feature selection process, applying the same principles as in the previous case: XGBoost and train_test_split = 0,2

Features	Importances [%]
Indicated air Mass flow	49,35
TLC_Concentration_EO_from	16,198
DPF Soot Mass	12,998
Injected Quantity	3,633
Intake manifold pressure	3,565
DPF Downstream temperature	3,093
DPF delta pressure	2,105

Features Excluded	Importances [%]
Lambda	1,778
Environmental pressure	1,383
Engine speed	1,283
Volume flow rate across DPF	0,797
Environmental Temperature	0,795
Sensed Intake fresh Air	0,615
Vehicle Speed	0,612
Engine Coolant temperature	0,569
DPF Upstream temperature	0,483
Intake manifold temperature	0,385
EGR Rate	0,347

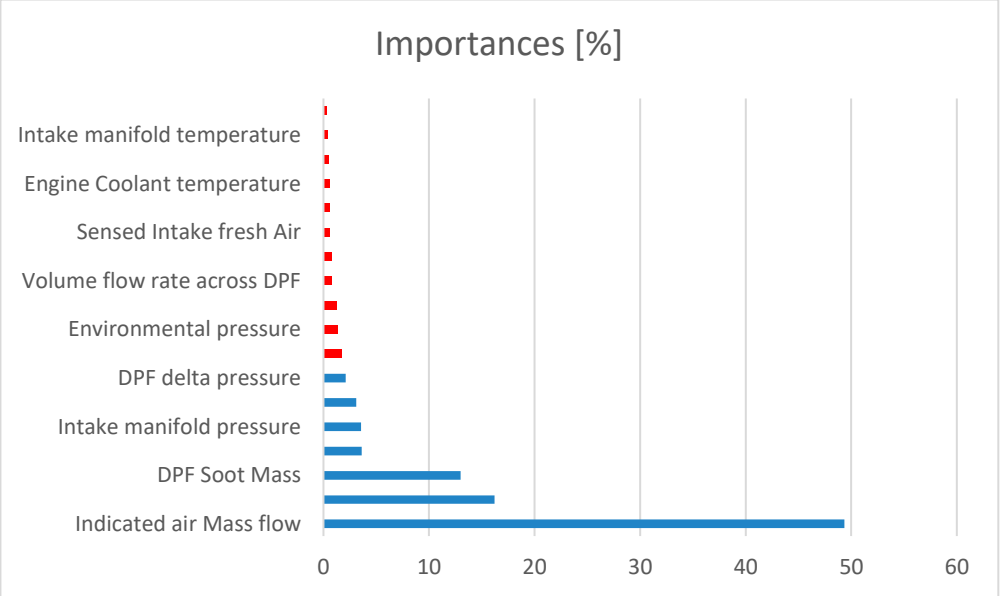

Figure 49 - File 2 Feature importance Graph

Here the different parameters at the output of the GridsearchCV process are reported, containing the coefficients of determination (cv_score) obtained during the analysis of the values of the XGBoost parameters linked to them.

Hyperparameter	DATASET 2
Learning rate	0,01
Max_depth	5
N_estimetors	1600
Min_child_weight	6
gamma	0,05
Colsample_bytree	1

CV_SCORE	$R2$
0	0,979
1	0,98
2	0,98
3	0,98
4	0,978
5	0,98
6	0,98
7	0,981
8	0,979
9	0,98
MEAN VALUE	0,979

As can be seen from the data obtained, the first three features of greater relevance are the same for both datasets. In addition, the EGR rate is excluded during feature selection of both datasets. This leads us to think that although EGR is physically one of the most important parameters for the determination of soot, the predictive process takes little account of this fact. Another important aspect is that the vehicle speed and engine revolutions are not considered for both the analyzed folders. Furthermore, even within the second dataset, among the three redundant variables physically connected to each other regarding the air flow, the predictive system only considers the indicated air mass flow.

These last considerations lead us to think that although the two tests have been carried out under different conditions on the same engine, they can be physically comparable, and that the predictive system can operate with the same logic in both datasets.

4.4 - Dataset 5

After analyzing the data belonging to the two stationary folders on the same vehicle and for an equal value of BPU, I carried out a new analysis on the data belonging to Folder 5 containing the measurements regarding the WLTC driving cycle.

The WLTC (World-wide harmonized Light duty test cycle) guide cycle is derived from real guide data provided by 5 different Regions: USA, India, Korea, Japan, and EU + Switzerland. These data were averaged by considering a large drop in vehicles on different types of roads and under different driving conditions. This pollutant determination cycle replaced the NEDC driving cycle in 2018 inside European legislation. This procedure is particularly important because the use of the NEDC cycle did not give a true view of the pollutants produced by vehicles.

The measurement procedures are applicable to different categories of vehicles that are classified within the legislation according to their power to mass ratio (PMR). This parameter is defined as the ratio between the power expressed in Watt and the curb mass in Kg. (1)

Category	PMR	Speed Phases	Comments
Class 3	PMR > 34	Low, Middle, High, Extra-High	If v_max < 135 km/h, phase 'extra-high' is replaced by a repetition of phase 'low'.
Class 2	34 ≥ PMR > 22	Low, Middle, High	If v_max < 90 km/h, phase 'high' is replaced by a repetition of phase 'low'.
Class 1	PMR ≤ 22	Low, Middle	If v_max ≥ 70 km/h, phase 'low' is repeated after phase 'middle'. If v_max < 70 km/h, phase 'middle' is replaced by a repetition of phase 'low'.

Figure 50 - WLTC Vehicle categories (1)

The third class represented in the table, is representative of the vehicles driven in Europe and Japan.

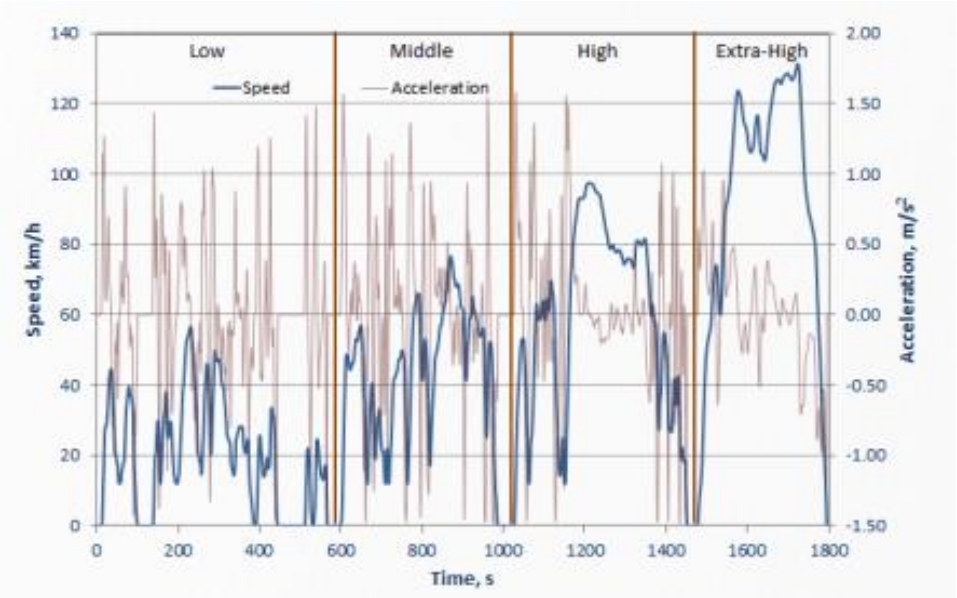

Figure 51 - Vehicle Speed in a WLTC Driving Cycle Example (1)

The cycle is divided into four parts:

- Low
- Middle
- High
- Extra-High

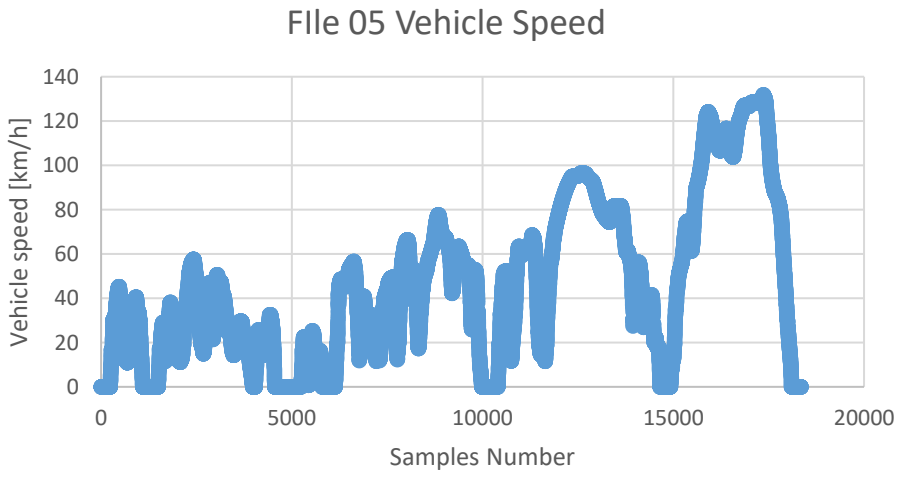

Figure 52 - File 5 Vehicle Speed

Analyzing the diagram of the vehicle speed of Dataset 5, we can notice precisely the four phases of the WLTC cycle. It can also be seen that the maximum speed reached exceeds 120 km/h, this means that the vehicle in analysis belongs to the regulatory class 3b.

Since the test has different engine operating conditions, it is highly variable with sharp variations in speed and acceleration.

Soot_EO and Soot_TP follow a less linear trend than the previous datasets and have orders of magnitude very different from each other since, as said before, there is the phenomenon of DPF regeneration. In addition, the Soot TP has more spikes than the engine out and it is believed that this may bring about some problems in the different analyses. Treating such small Soot_TP values can, despite the normalization process, lead to small unbalances, for this reason it is necessary to be very careful when handling the dataset.

As with previous datasets, no filters were placed on File 5, but only the negative values of Soot_TP and Soot_EO were returned to 0.

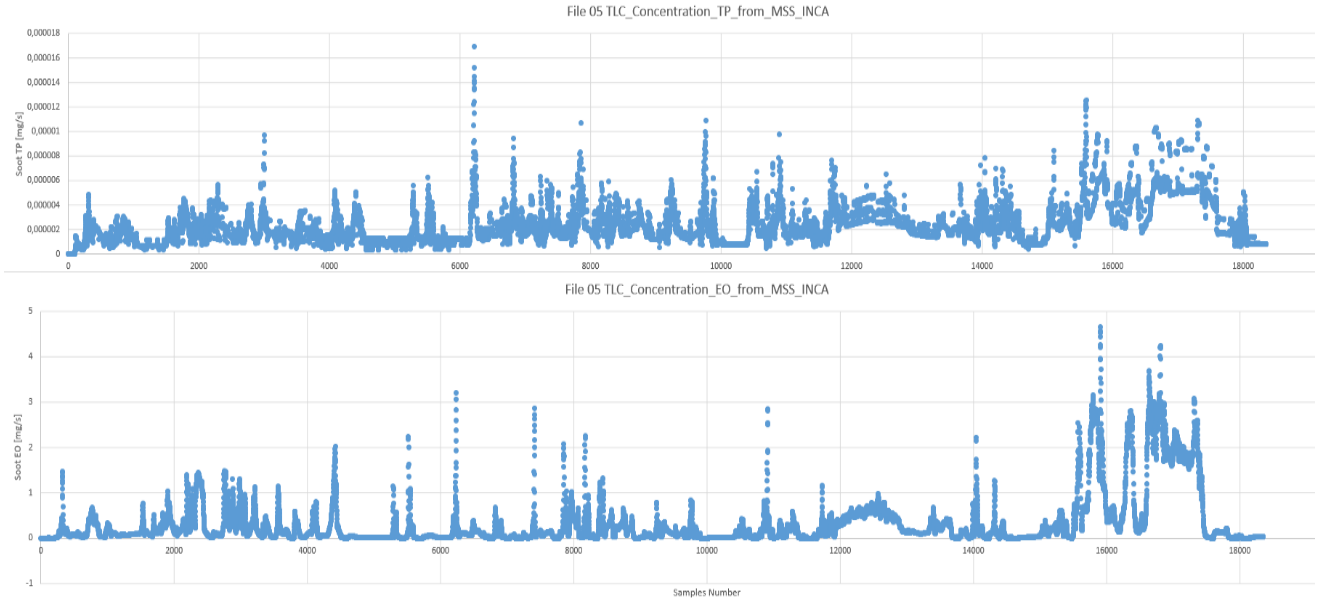

Figure 53 - File 5 Soot_TP and Soot_EO

A check was made on the different features to analyze their performance and again in this case a certain redundancy was found between: Indicated air mass flow, Volume flow rate across DPF and Sensed intake fresh air.

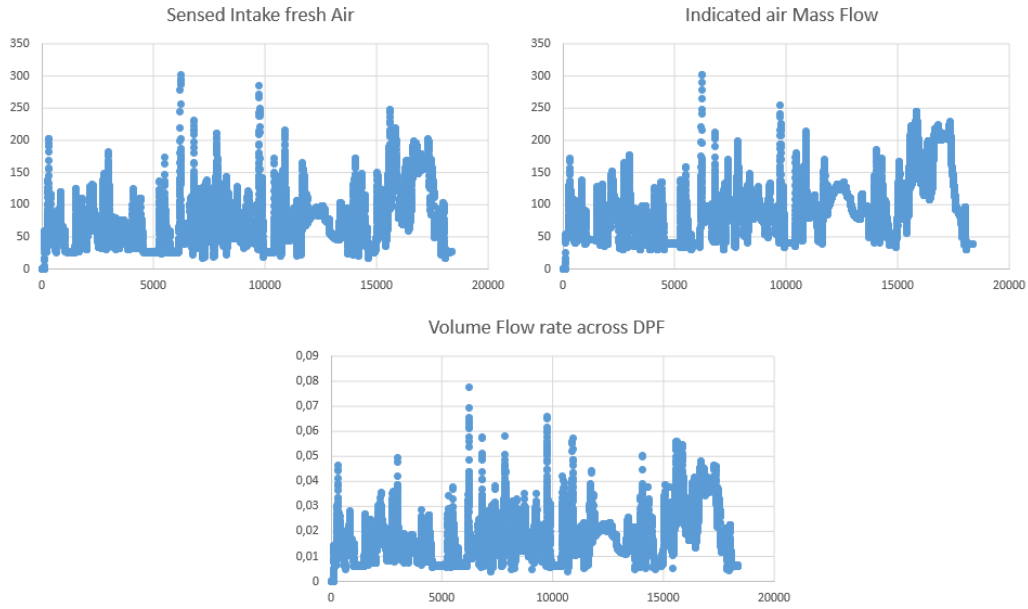

Figure 54 - File 5 Air Flow features

As for the characteristics of the DPF, we can observe that compared to the previous cases the upstream and downstream temperatures of the DPF are reduced. Furthermore, they follow a similar trend but not as close as in the two previous folders. In addition, a lower level of DPF Soot Mass measurements can be noticed.

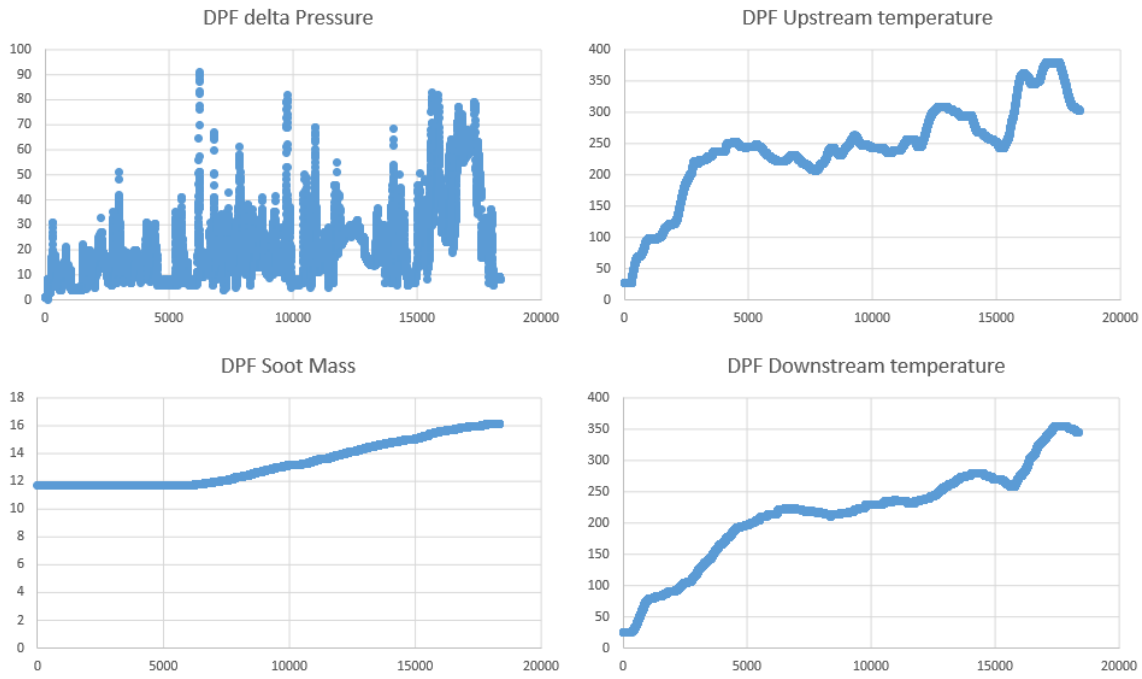

Figure 55 - File 5 DPF Features

In the following tables, we can observe the results obtained from the feature selection process, the different hyperparameters obtained from the GridsearchCV process and the coefficients of determination (R^2) obtained with the parameters of the XGBoost parameters connected to them.

Features	Importances [%]
DPF delta Pressure	56,852
Volume flow rate across DPF	27,241
Sensed intake fresh air	2,323
DPF Soot mass	1,549
Intake manifold pressure	1,314
Engine coolant temperature	1,19

Features Excluded	Importances [%]
DPF Upstream temperature	1,039
Injected quantity	1,013
Intake manifold temperature	1,002
Soot_EO	0,947
DPF downstream temperature	0,89
Indicated air mass flow	0,889
Vehicle speed	0,88
lambda	*0,794*
Engine speed	0,752
EGR rate	0,709
Environmental temperature	0,6
Environmental pressure	0

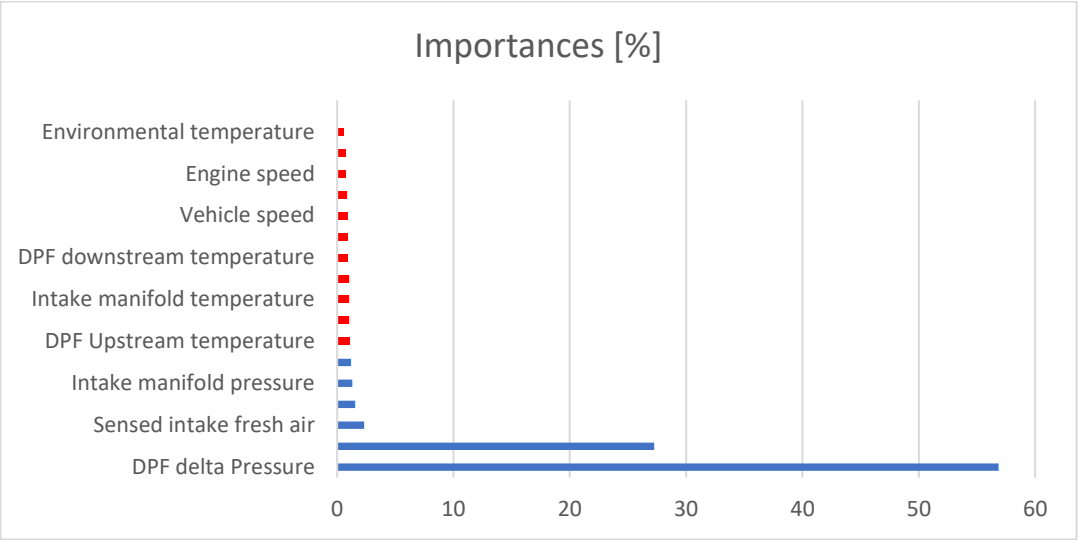

Figure 56 - File 5 Feature Importance graph

Hyperparameter	DATASET WLTC
Learning rate	0,01
Max_depth	10
N_estimetors	3000
Min_child_weight	6
gamma	0,05
Colsample_bytree	1

CV_SCORE	*R2*
0	0,883
1	0,871
2	0,873
3	0,88
4	0,879
5	0,877
6	0,871
7	0,875
8	0,88
9	0,868
MEAN VALUE	0,874

From the feature selection made on File 5, I noticed that the variables concerning engine revolutions and EGR are discarded again. The DPF delta pressure is the most important variable obtained in this process with a very high importance percentage of 56.85%. Among the DPF measurements, temperature variables are excluded, while the algorithm considers the DPF Soot mass as relevant (main variable for all 3 datasets). The main difference between Dataset 5 and the previous cases is the exclusion of the Soot_EO from the main variable of the problem. This may be due to different engine operating conditions

or the different magnitude of the measurements. It is important to point out that this analysis was carried out with several datasets concerning measurements on the WLTC driving cycle, but the results obtained were practically the same. Even using RandomForest as a feature selection algorithm, there were no major variations in the obtained features.

In the following sections I will analyze the results obtained by the artificial neural networks applied to the different datasets and the results obtained by crossing the two stationary spreadsheets.

CHAPTER 5 - Neural Network Results

Let us now pass to the analysis of the results obtained using the neural network and the different plots which are representative of the performance of the system.

Hyperparameters of neural networks have been selected as mentioned above using GridsearchCV. In this way the best combination of parameters has been found to make the system as efficient and accurate as possible. They were then used for the model run in order to achieve the neural network results.

5.1 - Dataset 1

After having treated the dataset with the different functions previously analyzed, starting from the input features obtained by the feature selection process, we can continue with the network analysis.

DATASET 1
Soot_EO
DPF soot mass
Indicated air mass flow
DPF delta pressure
DPF upstream temperature
DPF downstream temperature
Intake manifold temperature
Environmental temperature
Lambda
Engine coolant temperature

Starting from the new dataset, the hyperparameters of the ANN obtained from the GridsearchCV optimization were set.

- N° of Neurnons:
  - Layer 1: 200, Dropout: 0,1
  - Layer 2: 200, Dropout: 0,1
- N° of hidden layer: 2
- Activation Function:
  - ReLu
  - Linear for the output layer
- Loss: Mse
- Batch size: 300
- Epochs: 200
- Learning rate: 0,001
- Kernel initializer: normal
- Train Size: 80%
- Test Size: 20%
- Validation Split: 20%

All pre-processing operations previously carried out are intended to prevent the network from operating incorrectly or incurring inconsistent phenomena such as overfitting and underfitting. As previously mentioned, data are randomly selected from the network by shuffle. To get a more accurate estimations of the model results, the data processing operation was repeated 10 times. Here are the results obtained in training and testing by the network for all analyses in terms of R^2, MSE and RMSE.

TRAINING			
METRIC	$R2$	MSE	RMSE
0	0,987	0,0001	0,0116
1	0,99	0,0001	0,0099
2	0,989	0,0001	0,0108
3	0,99	0,0001	0,01
4	0,992	0,0001	0,0093
5	0,99	0,0001	0,0101
6	0,991	0,0001	0,0101
7	0,991	0,0001	0,0095
8	0,992	0,0001	0,0092
9	0,989	0,0001	0,0108
MEAN VALUE	0,99	0,0001	0,0106

TESTING			
METRIC	$R2$	MSE	RMSE
0	0,986	0,0001	0,0118
1	0,989	0,0001	0,0105
2	0,988	0,0001	0,0114
3	0,99	0,0001	0,0101
4	0,991	0,0001	0,001
5	0,99	0,0001	0,0102
6	0,99	0,0001	0,0101
7	0,99	0,0001	0,01
8	0,991	0,0001	0,01
9	0,987	0,0001	0,0114
MEAN VALUE	0,989	0,0001	0,0108

The tables show very positive predictive results in the order of 99% both in training and in testing. This means that the network is operating correctly and it can deal with this type of dataset without any problem. Despite the positive numerical results, it is also necessary to observe the graphs concerning the learning curves. As mentioned above, this procedure is important to understand whether the network incurs overfitting or underfitting phenomena.

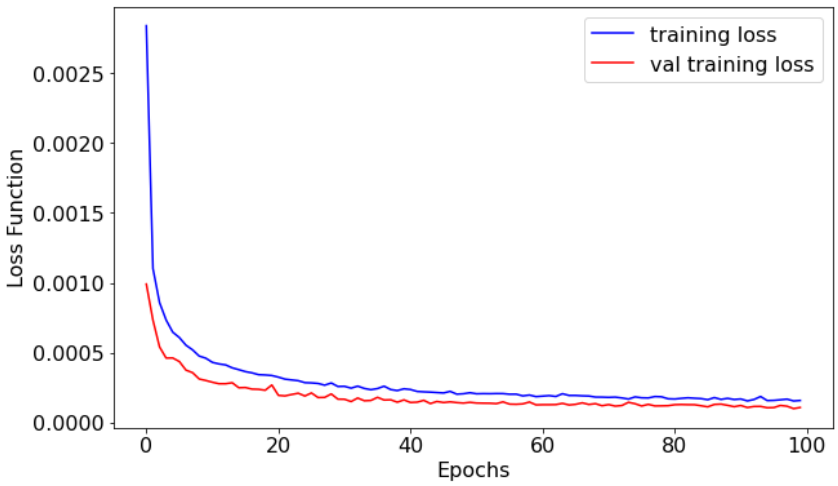

Figure 57 - File 1 Learning Curves

From the diagram it can be observed that we do not incur in any kind of fitting error just because the two curves go perfectly to convergence following the increase in the number of epochs. Network performance can also be analyzed graphically using representations in which the predicted values obtained by the model in the training and testing phases are compared to the real ones present in the dataset. In the following diagrams it is possible to observe such aspects; moreover, the nearer the points are to the bisector, the greater is the precision of the system.

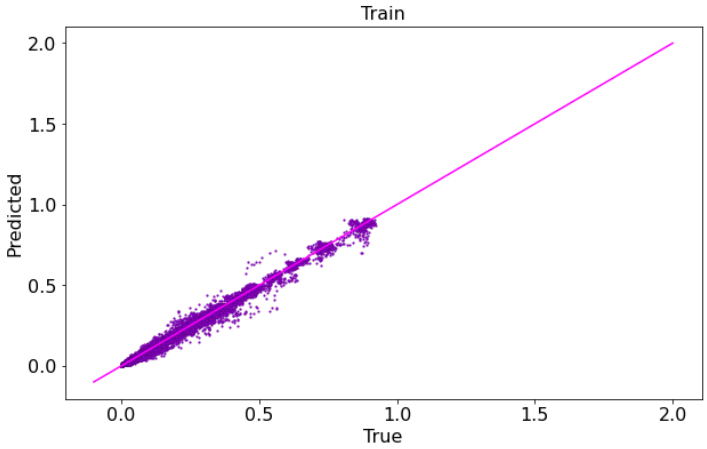

Figure 58 - File 1 Plot Train Real vs Predicted

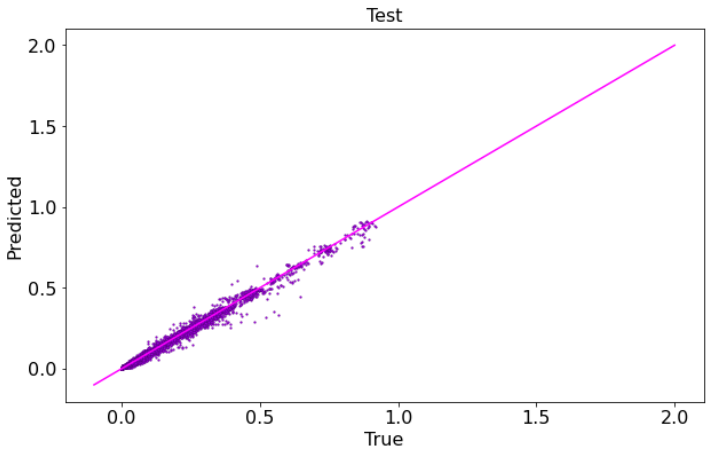

Figure 59 - File 1 Plot Test Real vs Predicted

As we can see the model is working in the proper way and without problems.

The last useful charts to observe were made in Matlab to give a better understand of the obtained results during the testing phase. The first one represents the overlap of real and predicted values of Soot as a function of time. This representation is made for a constant Engine Speed value because considering all the data would be too confusional and it would not give a good idea of the predictive level. Since the Dataset 1 is very large, this graph considers all the points at a constant value of 3250 RPM as engine revolutions.

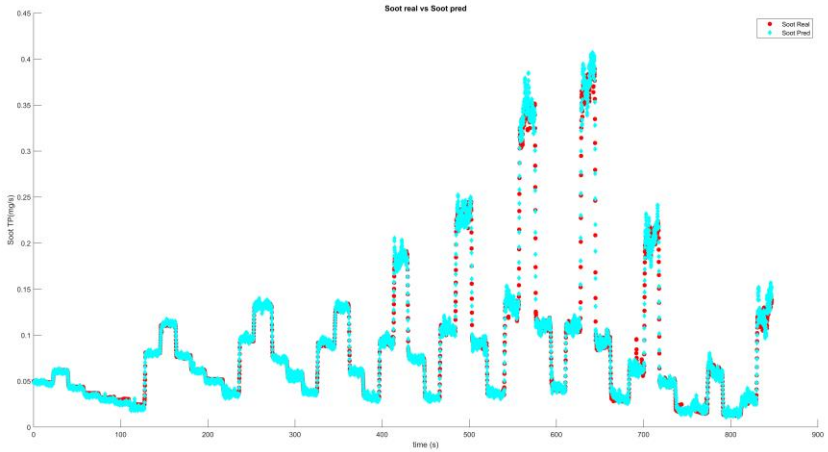

Figure 60 - File 1 Plot Real vs Predicted 3250rpm

The second graph represents the two cumulated curves of real and predicted soot. This representation is made by summing up all the soot values (real and predicted) present in every point. It is useful to observe the total error given by the sum of all the differences computed for the points in analysis.

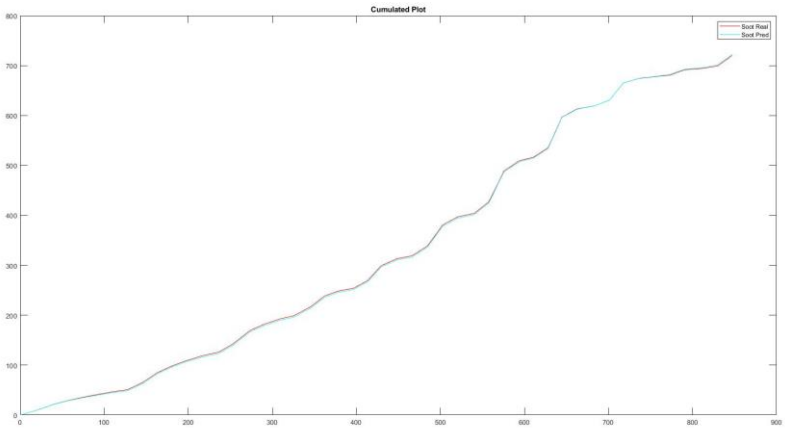

Figure 61 - File 1 Cumulated Plot 3250rpm

Inside the graphs, the red curve and the red points represents the real values of Soot_TP while the blue ones represent the values of Soot_TP predicted by the neural network.

5.2 - Dataset 2

As for the previous case, the results obtained from the use of the File 2 starting from the new features incoming to the system are reported.

DATASET 2
Indicated air mass flow
Soot_EO
DPF soot mass
Injected quantity
Intake manifold pressure
DPF downstream temperature
DPF delta pressure

Hyperparameters of the Feed Forward Artificial Neural Network:

- N° of Neurons:
  - Layer 1: 200, Dropout: 0,1
  - Layer 2: 200, Dropout: 0,1
- N° of hidden layer: 2
- Activation Function:
  - ReLu
  - Linear for the output layer
- Loss: Mse
- Batch size: 250
- Epochs: 120
- Learning rate: 0,001
- Kernel initializer: glorot uniform
- Train Size: 80%
- Test Size: 20%
- Validation Split: 20%

As previously mentioned, the data are randomly selected by the network, to obtain a more and more accurate estimate of the results of our model; it was decided to run and repeat the data processing operation for 10 times.

TRAINING			
METRIC	***R2***	**MSE**	RMSE
0	0,977	0,0002	0,0126
1	0,978	0,0001	0,0122
2	0,978	0,0002	0,0123
3	0,978	0,0001	0,0121
4	0,977	0,0002	0,0127
5	0,979	0,0001	0,0119
6	0,975	0,0002	0,0132
7	0,977	0,0002	0,0126
8	0,974	0,0002	0,0133
9	0,976	0,0002	0,0129
MEAN VALUE	0,975	0,0002	0,0129

TESTING			
METRIC	***R2***	**MSE**	RMSE
0	0,974	0,0002	0,0133
1	0,975	0,0002	0,0129
2	0,973	0,0002	0,0136
3	0,977	0,0002	0,126
4	0,976	0,0002	0,13
5	0,976	0,0002	0,0132
6	0,973	0,0002	0,0131
7	0,974	0,0002	0,0133
8	0,972	0,0002	0,0138
9	0,975	0,0002	0,0131
MEAN VALUE	0,974	0,0002	0,0131

The R^2 values are equal to 97%, this means that the determination coefficients of File 2 are lower compared to the ones of File 1. Such behavior may be due to several causes but the most likely is that Dataset 1 presents a much greater number of samples than Dataset 2. This means that in File 1, the neural network will have the opportunity to learn better and consolidate the system even more efficiently. Despite these observations, in any case the scores obtained in File 2 are very high, this means that the network can efficiently predict the Soot_TP through the data present in Dataset 2. Below there are the different charts as the ones that have been showed in the previous part.

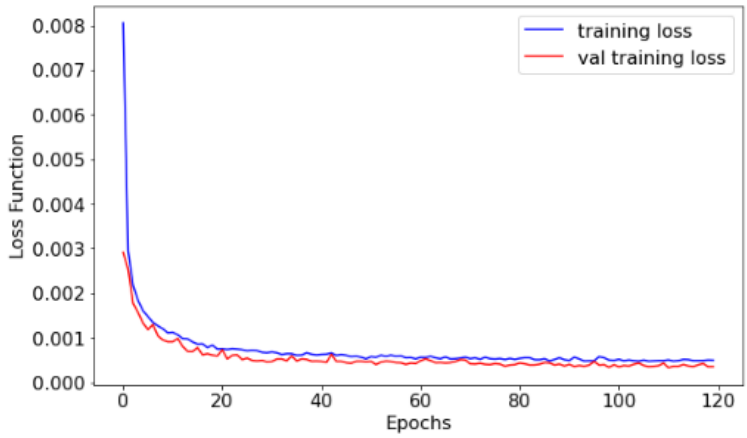

Figure 62 - File 2 Learning Curves

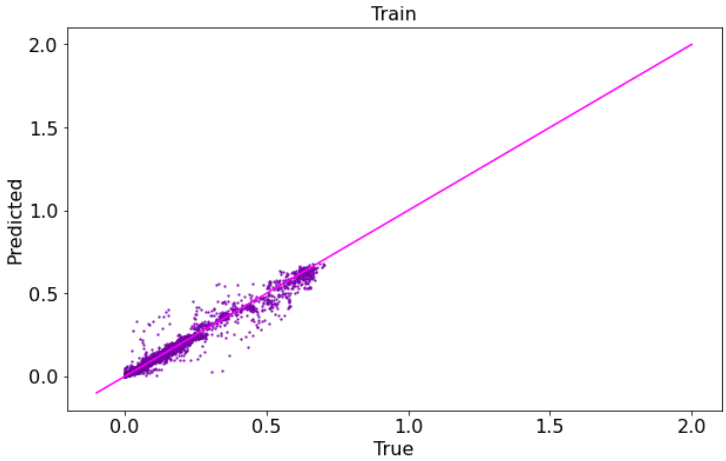

Figure 63 - File 2 Plot Train Real vs Predicted

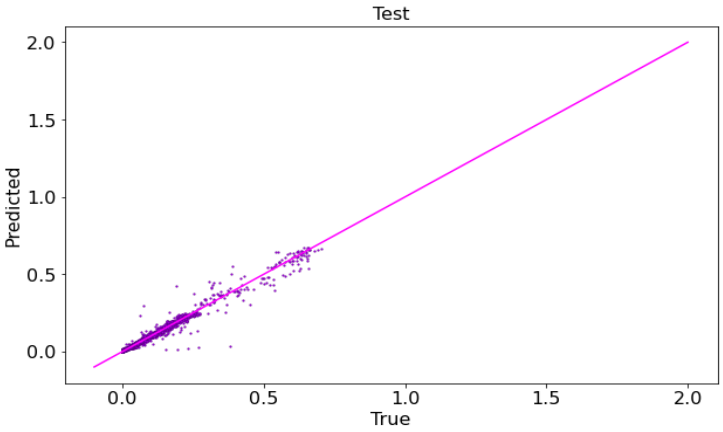

Figure 64 - File 2 Plot Test Real vs Predicted

The following graphs represent part of the measures present inside File 02 at a constant Engine Speed equal to 2500 RPM.

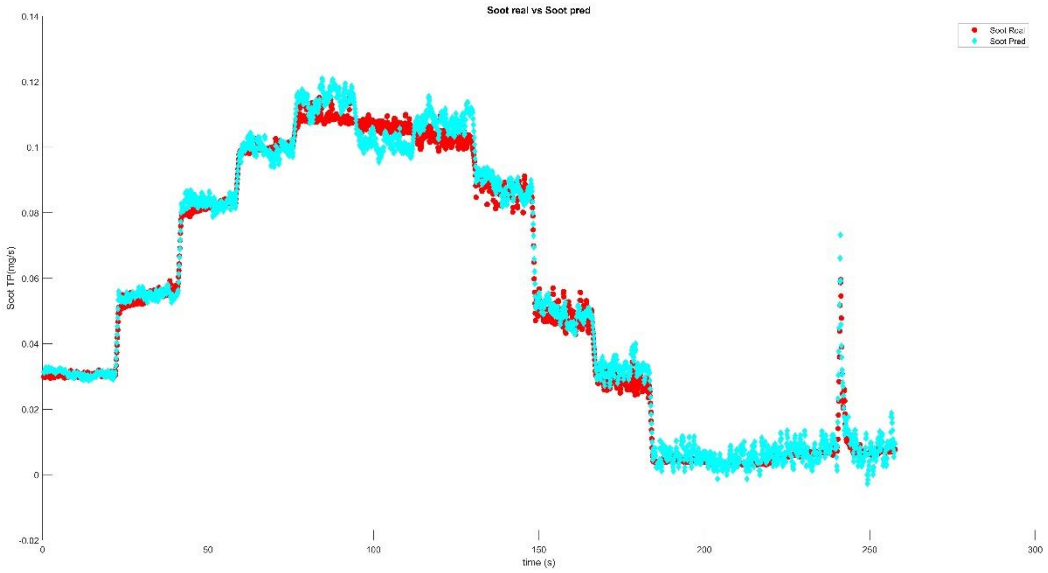

Figure 65 - File 2 Plot Real vs Predicted 2500rpm

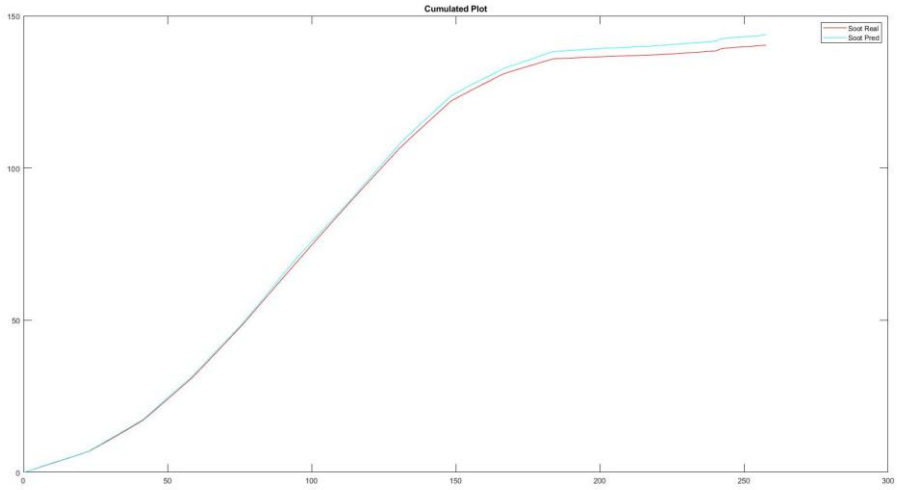

Figure 66 - File 2 Cumulated Plot 2500rpm

We can notice that there is a good fit from the learning curves that converge. Furthermore, looking at the comparison between real and predicted values and the cumulated plot, it is possible to see a small predictive deficit. The obtained results are summarily good for the resolution of the predictive problem.

5.3 - Dataset 5

The same procedure as the stationary files is also followed for Dataset 5 containing data on the WLTC guide cycle. This analysis will be very important to interpret whether or not a neural network with the same structure as that adopted in the stationary cases is able to obtain satisfactory results even operating with transient data.

Starting from the features obtained through the feature extraction process, the optimized network Hyprerparameters using GridsearchCV are reported.

DATASET WLTC
DPF delta Pressure
Volume flow rate across DPF
Sensed Intake fresh air
DPF soot mass
Intake manifold pressure
Engine coolant temperature

- N° of Neurons:
  - Layer 1: 120, Dropout: 0,1
  - Layer 2: 120, Dropout: 0,1
- N° of hidden layer: 2
- Activation Function:
  - ReLu
  - Linear for the output layer
- Loss: Mse
- Batch size: 250
- Epochs: 120
- Learning rate: 0,001
- Kernel initializer: glorot uniform
- Train Size: 80%
- Test Size: 20%
- Validation Split: 20%

As in the previous sections, here we can see R^2, MSE and RMSE values coming from the ANN during training and testing phases.

TRAINING			
METRIC	$R2$	MSE	RMSE
0	0,873	0	0
1	0,862	0	0
2	0,869	0	0
3	0,872	0	0
4	0,865	0	0
5	0,867	0	0
6	0,867	0	0
7	0,866	0	0
8	0,87	0	0
9	0,864	0	0
MEAN VALUE	0,859	0	0

TESTING			
METRIC	$R2$	MSE	RMSE
0	0,857	0	0
1	0,861	0	0
2	0,851	0	0
3	0,865	0	0
4	0,861	0	0
5	0,853	0	0
6	0,848	0	0
7	0,858	0	0
8	0,857	0	0
9	0,86	0	0
MEAN VALUE	0,858	0	0

From the obtained results, we can immediately notice a lowering of 86% of the predictive level compared to the previous cases. This decrease is mainly due to the fact that this dataset operates in completely different conditions than the previous models. In addition, Dataset 5 contains 18,346 samples which are too low, compared to 136,365 of Dataset 1 and 36,658 of Dataset 2. Having a smaller dataset can be translated into a further cause of decreased accuracy because the network has less points to train itself and to perform an optimal prediction. Inside the tables we can also see that all the MSE and RMSE values are reported equal to 0. This aspect does not mean that the error is null but that it is very small and several order of magnitude under the unity. Now let us see the graphs of the analysis.

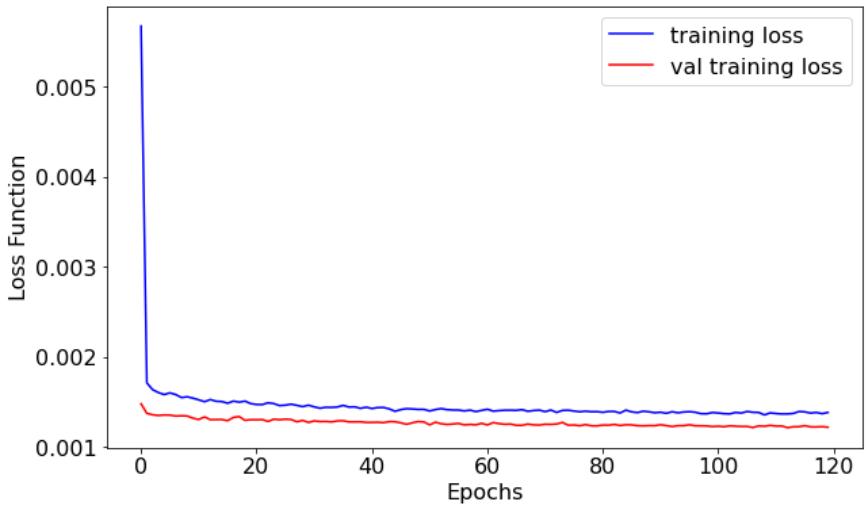

Figure 67 - File 5 Learning Curves

Looking at the learning curves graph, we can notice that the system does not encounter any phenomenon of overfitting or underfitting and the hole system is making a correct training phase without internal errors.

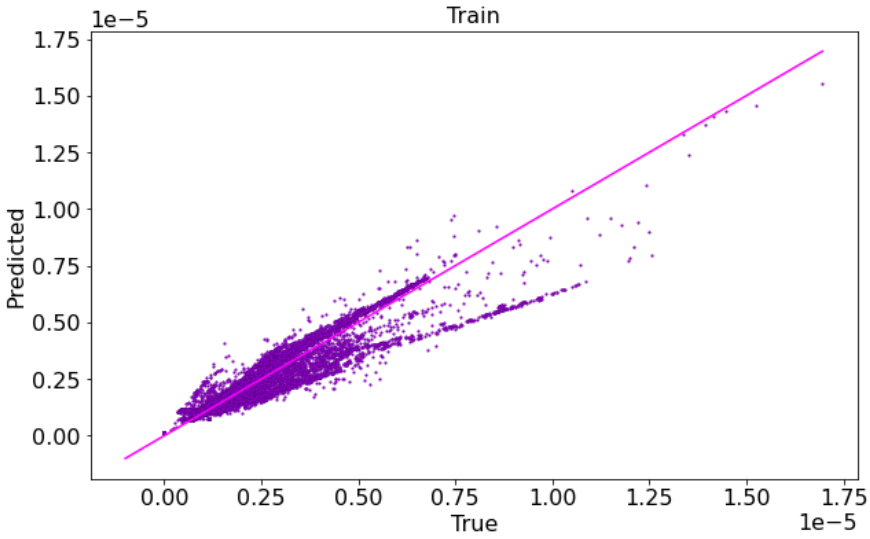

Figure 68 - File 5 Plot Train Real vs Predicted

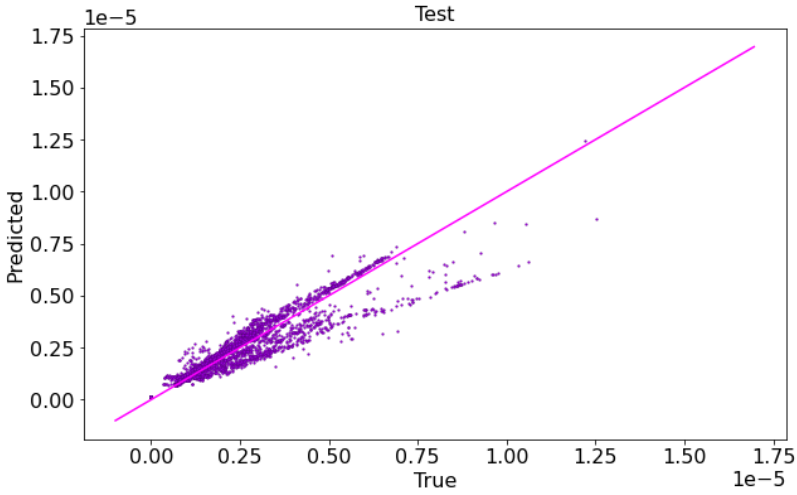

Figure 69 - File 5 Plot Test Real vs Predicted

As can be seen from the different representations of Soot_TP, the measurements are very small and are about 6 orders of magnitude lower than the values present in the other datasets. This has a negative effect on the efficiency of the system but, nevertheless, we get acceptable levels of accuracy. Looking at the true vs predicted value plot it is clear that the

network is incurring in some difficulties to give a proper prediction. This aspect is visible by the fact that many points both in training and testing are not on the bisector but are following a random trend in some points.

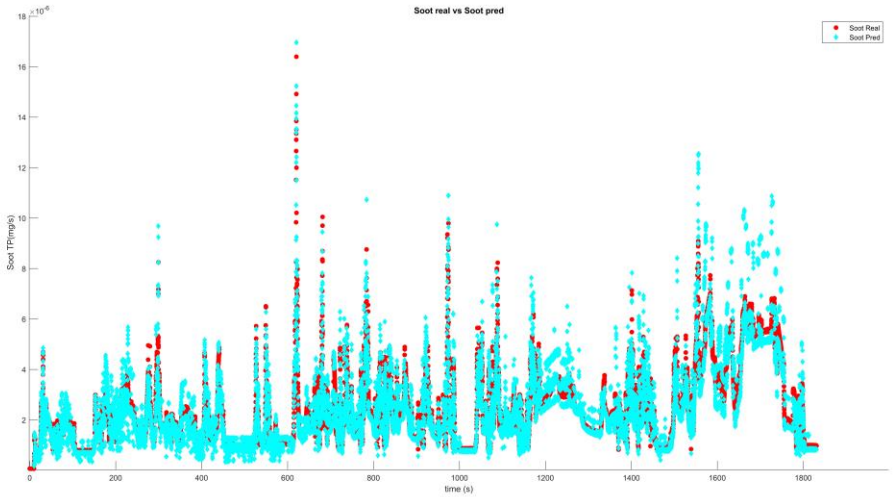

Figure 70 - File 5 Plot Real vs Predicted

For what concerns the plot comparing predicted and real values, it shows the decrease in efficiency compared to the previously analyzed stationary systems.

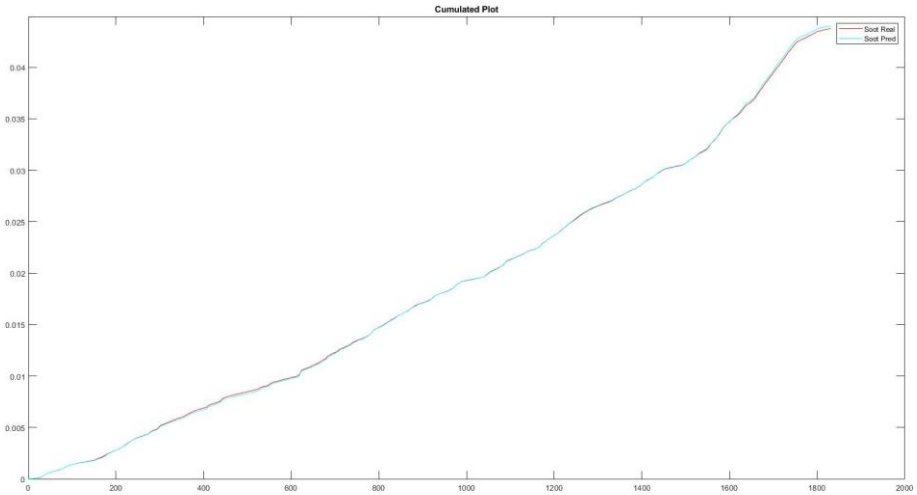

Figure 71 - File 5 Cumulated Plot

Although the cumulate looks quite efficient, this representation is not very useful since the orders of magnitude in question are small.

The results obtained show that the predictive model can also work when dealing with the WLTC cycle. Nevertheless, the predictive levels of the previous cases are not reached. Furthermore, in order to build high precision sensors, scores like 85% are not a good enough solution to our problem. It is important to point out that all the analyses previously shown have been carried out with different train_test_split to demonstrate the stability of the neural network. In all these cases, even by modifying this variable, there are no changes in the use of the artificial neural network. This clearly demonstrates the efficiency of the system used. In the next section we will see if the network is able to achieve good predictive levels even crossing different datasets during training and testing.

5.4 - Cross Analysis: Train set File 1 – Test set File 2

After having analyzed the construction of the artificial neural network and the different yields obtained from the predictive model working on the three files, a cross-analysis was performed to assess whether the model is able to maintain high predictive levels by training on Dataset 1 and trying to predict in testing Dataset 2. Both files contain some measurements carried out in stationary conditions, however, the main difference present between the two, is the presence of EGR sweeps regarding File 1, while the EGR is kept at nominal values within File 2. As can be seen from the features of greater relevance for both folders, they turn out to be very similar. On this subject, there have been several meetings with AVL's engineers to cover whether these datasets are physically comparable with each other. Despite the different characteristics of EGR, the levels of soot obtained in the two folders are comparable, so we can try to perform this type of analysis.

The choice to place Dataset 1 as a train set and File 2 as test set is mainly because Dataset 1 is much bigger than Dataset 2. As already said, File 1 contains 134.365 samples while File 2 contains only 36.658. Summing up all the samples we can see that the training set composed by the first dataset covers 78.5% of the total number of data, while the test set

covers 21.5%. These percentages are close to the previously used train_test_split = 0.2, so the data partition was carried out efficiently.

Since the training phase was carried out on Dataset 1, I have placed as input variables to the neural network, those coming from the feature selection process carried out on the spreadsheet in question. We also use the same network parameters used by File 1 in order to achieve our goal.

DATASET 1	DATASET 2
Soot_EO	Indicated air mass flow
DPF soot mass	Soot_EO
Indicated air mass flow	DPF soot mass
DPF delta pressure	Injected quantity
DPF upstream temperature	Intake manifold pressure
DPF downstream temperature	DPF downstream temperature
Intake manifold temperature	DPF delta pressure
Environmental temperature	
Lambda	
Engine coolant temperature	

- N° of Neurons:
  - Layer 1: 200, Dropout: 0,1
  - Layer 2: 200, Dropout: 0,1
- N° of hidden layer: 2
- Activation Function:
  - ReLu
  - Linear for the output layer
- Loss: Mse
- Batch size: 300
- Epochs: 200

- Learning rate: 0,001
- Kernel initializer: normal
- Validation Split: 20%

These are the results obtained by 10 tests made with the ANN in order to control the randomicity due to the shuffle of data.

TRAINING			
METRIC	$R2$	MSE	RMSE
0	0,099	0,0001	0,0103
1	0,993	0,0001	0,0099
2	0,0993	0,0001	0,0083
3	0,992	0,0001	0,0089
4	0,992	0,0001	0,0089
5	0,992	0,0001	0,0093
6	0,99	0,0001	0,0101
7	0,992	0,0001	0,0093
8	0,993	0,0001	0,0085
9	0,991	0,0001	0,0095
MEAN VALUE	0,991	0,0001	0,0095

TESTING			
METRIC	$R2$	MSE	RMSE
0	0,195	0,0055	0,0742
1	0,042	0,0066	0,081
2	0,162	0,0057	0,0757
3	0,012	0,0068	0,0822
4	0,047	0,0065	0,0808
5	0,02	0,0067	0,0819
6	0,129	0,006	0,0772
7	0,153	0,0058	0,0762
8	0,109	0,0061	0,0781
9	0,095	0,0062	0,0787
MEAN VALUE	0,13	0,006	0,0772

Unlike the previous analyses, in this case we obtain particularly low predictive network scores during the testing phase. Although the training done on Dataset 1 has not changed compared to the previous case, the artificial neural network is not able to obtain good predictive levels for the testing phase of Dataset 2. As we can see, the determination coefficient is near 0.1, this means that the network cannot find a logical connection between the study of the two problems.

Let's see the others plots related to the analysis:

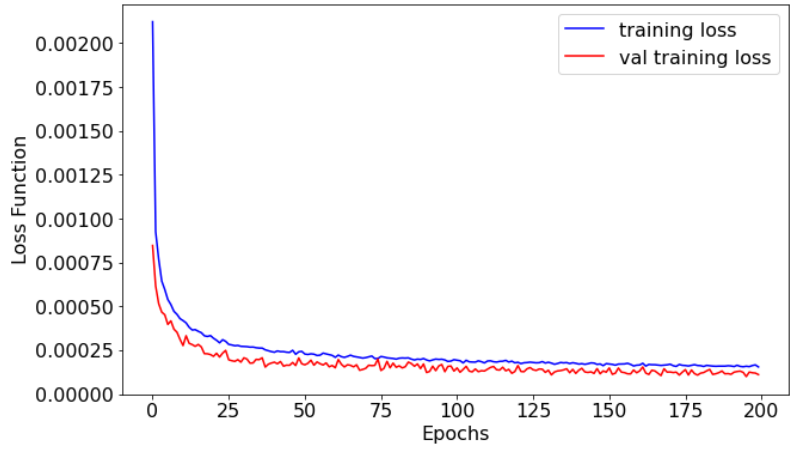

Figure 72 - Figure 67 - Static cross analysis Learning Curves

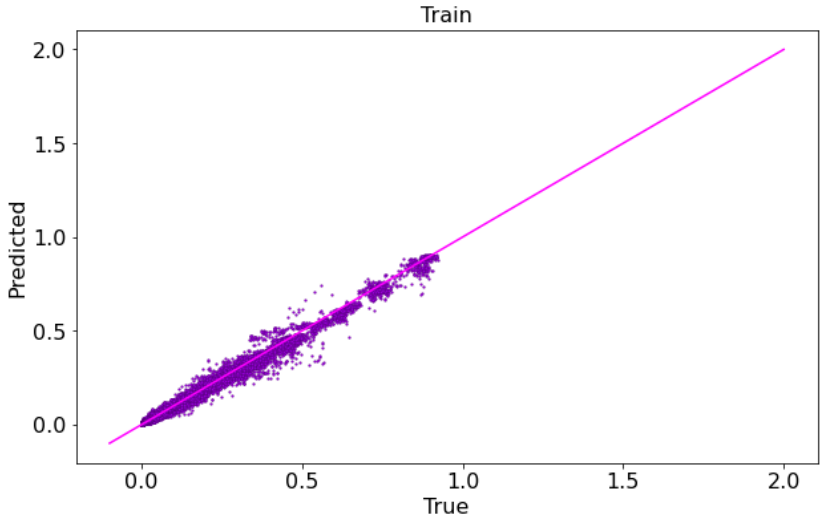

Figure 73 - Train File 1 Plot Real vs Predicted

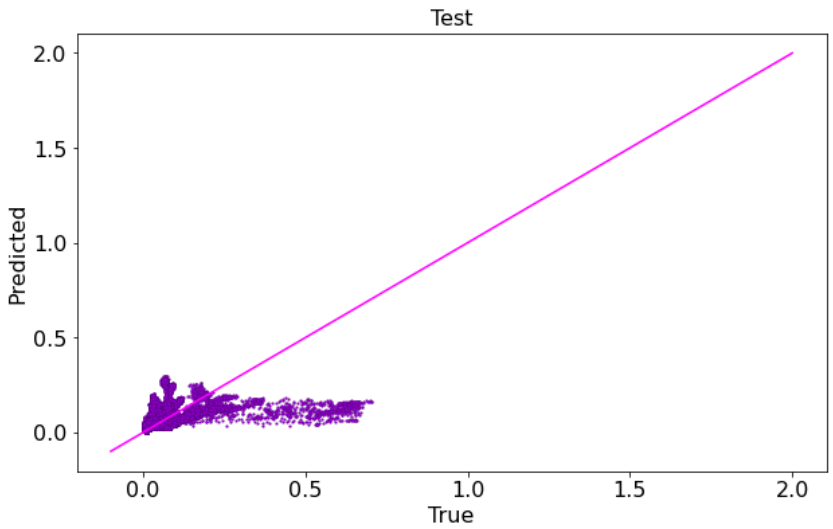

Figure 74 - Test File 2 Plot Real vs Predicted

As was easily predictable, the learning curves have an excellent performance precisely because the system does not encounter any problem during its training. But looking at the plot comparing predicted and real soot values, we can easily notice that the system is working wrongly.

Unfortunately, contrary to what we would have expected from observing the features and behaviors of the different features of the two folders, this analysis has not led to any result.

The artificial neural network is not able to solve this problem and to achieve acceptable predictive levels in the stationary cross analysis.

Conclusions

This thesis aims at the development of a virtual sensor in python for the prediction of the soot levels present at the discharge of a diesel engine (IC) (tail pipe) through the use of Machine learning algorithms, in this case belonging to the branch of deep learning. The use of neural networks for such analyzes represents the state of the art especially for the resolution of problems related to the use of physical sensors such as, for example, encumbrance and maintenance.

The aim is to lay the foundations for the development of new OBD systems that can reliably report the amount of pollutants emitted by the vehicle and the conditions of the particulate filter in order to implement precise regeneration strategies. The analysis presented would not have been possible without the support and datasets provided by AVL Italia. Within the work analysis was carried out on how to build the regressive artificial neural network and the different logics present within it in order to predict the complex phenomenon of soot formation.

The algorithm was first applied to File 1 containing measurements made on a diesel engine on a roller bench in stationary conditions. The neural network has reported excellent levels of accuracy and reliability of around 99%, demonstrating that this system is actually able to work efficiently with these types of datasets. The high predictive accuracy is also due to the high number of samples present within the folder which therefore allow the network to operate in the best way there can be. In the second analysis on File 2, a new tuning of the hyperparameters was made and reuse of the same logic as the previous neural network was tried in order to obtain results.

While Dataset 1 is characterized by some EGR sweeps, Dataset 2 presents a nominal EGR value with respect to the amount of fuel injected. Furthermore, the number of samples present in File 2 turns out to be much lower (about a quarter) than those present in File 1. Despite this, the results obtained from the network are very positive, to the order of 97 %, therefore satisfactory for solving the problem. In the next analysis, the dataset containing the WLTC regulatory guide cycle measurements was taken into consideration. Although this dataset is very small compared to two in stationary, use of the same logic as the neural network used previously to predict the soot values produced in the transient regime was tried.

The first major difference within this dataset is that the soot measurements are approximately six orders of magnitude lower than those present within the other folders. This is due to the fact that within these tests, the regeneration phenomenon of the DPF is also introduced, which implies a substantial reduction in the quantities of soot produced.

Despite having adopted a normalization for the different datasets, this variation negatively affects the accuracy of the system. The analysis of File 5 shows predictive scores to the order of 86% and, although this result seems high, it is not enough for the construction of a high-precision sensor.

In the last analysis presented, comparison of the two datasets in stationary was made and the training phase on Dataset 1 and the testing phase on Dataset 2 were carried out. Based on the results obtained following the features selection process, positive results would have been expected, but the scores obtained by the network are particularly low. It should be noted that during the analysis process many other stationary datasets were used with conditions similar to the two mentioned but in each application, satisfactory results were not obtained.

Finally, cross-analysis with the datasets provided on the WLTC guide cycle was carried out. During this phase, AVL provided 3 other datasets with measurements of the Soot_TP but, despite this, the results obtained did not satisfy the analysis from a predictive point of view, thus a decision was made not to report them since negative determination coefficients were detected. This aspect demonstrates an impossibility on the part of the current neural network to find a physical link that correlates different transient datasets or the same transient data with those in stationary. I think that these unsatisfactory results are not a point of arrival but a departure in order to continue to improve the model and find a solution to the problem. The aim therefore remains to create a universal sensor that has the ability to combine the different spreadsheets and obtain excellent predictive levels in any type of stationary or transient condition.

In order to improve the research, I think that two important foods for thought could be that of:

- improving the GridsearchCV algorithm to obtain more precise optimizations of the network parameters, thus moving to a new algorithm that does not say standardized hyperparameters for the entire length of the network but which

can give precise structural parameters to each layer of the network. In this way it could be possible to improve the different network scores.

- programming and trying to use a new regressive neural network that can take into account the temporal evolution of the system, especially for the study of datasets in a transient regime.

Finally, it would be interesting to carry on this work towards the design of a classifier capable of periodically monitoring the operating status of the DPF and the soot in order to then be able to coordinate precise regeneration strategies so as to reduce the quantity of pollutants released into the atmosphere.

I firmly believe that by continuing to experiment with new models and new ideas, the solution for the construction of the universal virtual sensor for the prediction of Soot_TP can be reached.

Bibliography

1. **E., Spessa.** *"Controllo delle Emissioni di Inquinanti".* Politecnico di Torino : s.n., 2020\2021.

2. **Millo, F.** *Corso di ingegneria meccanica "Propulsione dei veicoli terrestri".* 2018.

3. **Akihama.** *SAE 2001-01-0655.*

4. **Kimura.** *SAE 2001-01-0200.*

5. **Kamimoto.** *SAE 880423.*

6. **Ryan.** *SAE 961160.*

7. **Ng.A.** *Online Course: Machine Learning.* Stanford University : s.n.

8. **Giron, Arèlien.** *Hands On Machine Learning with Sckit-Learn & TensorFlow.*

9. **Cirrincione, Giansalvo EXIN.** *Deep Learning .*

10. **Falai, Alessandro.** *Tesi di laurea magistrale in ingegneria meccanica: Applicazione e validazione di un modello Random Forest per la stima della massa di particolato in motori diesel".* 2018.

11. **D.Maltoni.** *Fondamenti di Machine Learning.*

12. **G.Valvo.** *Tesi di Laurea Magistrale "Sviluppo di un sensore virtuale per la stima della concentrazione si soot allo scarico di motori diesel in ottica OBD.".* Politecnico di Torino : s.n.

13. **Nobile, Giuseppe.** *Sviluppo di un codice Machine learning per la stima di ossidi di azoto e particolato in motori Diesel.*

14. **Mastery, Machine Learning.** *How to use Learning Curves to DIagnose Machine Learning Model Performance .*

15. **https://medium.com/greyatom/what-is-underfitting-and-overfitting-in-machine-learning-and-how-to-deal-with-it-6803a989c76.**

16. **point, java T.** *https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm.*

17. **Aggarwall, Charu C.** *"Neural Network and Deep Learning".* Yorktown Heights, NY, USA : Springer, 2018.

18. **Mahesh, Batta.** *"Machine Learning Algorithms- A Review".* s.l. : International journal of science and research (IJSR), 2018.