A Methodology for improved, Data-Oriented, Air Quality Forecasting

Ioannis Kyriakidis

PhD

Faculty of Computing, Engineering and Science

University of South Wales

A submission presented in partial fulfilment of the requirements of the
University of South Wales/Prifysgol De Cymru
for the degree of Doctor of Philosophy

[June 2018**]**

Table of Contents

List of Figures

List of Tables

Abbreviations

ACO	Ant Colony Optimization
AI	Artificial Intelligence
AQ	Air Quality
ANNs	Artificial Neural Networks
AOFS	Automated Online Forecasting System
AS	Ant System
BCO	Bee Colony Optimization
CAQI	Common Air Quality Index
CI	Computational Intelligence
CV	Cross Validation
d	Index of Agreement
DOM	Daphne Optimization Methodology
DOP	Daphne Optimization Procedure
DP	Data Pre-processing
DTs	Decision Trees
DUC	Daphne Uniform Crossover
E	Coefficient of Efficiency
EA	Evolutionary Algorithm
EEA	European Environment Agency
EF	Modelling Efficiency
FB	Fractional Bias
FISs	Fuzzy Inference Systems
FNNs	Feed-Forward Neural Networks
FOEX	Factor Of Exceedance
FPI	Forecasting Performance Indices
GA	Genetic Algorithm
GLMs	Generalized Linear Models
GRNNs	Generalized Regression Neural Networks
GTA	Greater Thessaloniki Area
GUI	Graphic User Interface
IAS	Intelligent Automated System

LNNs	Linear Neural Networks
LR	Linear Regression
MAD	Median Absolute Deviation
MAE	Mean Absolute Error
MAPE	Mean Absolute Percent Error
MARS	Multivariate Adaptive Regression Splines
MFB	Mean Fractional Bias
MLE	Maximum Likelihood Estimate
MLR	Multiple Linear Regression
MPE	Mean Percentage Error
MSE	Mean Squared Error
NB	Normalized Bias
NMSE	Normalized Mean Square Error
PA	Percentage of Agreement
PCA	Principal Component Analysis
PCL	Penalty Cancel Level
PGAs	Production Genetic Algorithms
PSO	Particle Swarm Optimization
r	Linear Correlation Coefficient
r^2	Coefficient of Determination
RMSE	Root Mean Squared Error
RMSEs	Systematic RMSE
RMSEu	Unsystematic RMSE
r_s	Spearman’s Rank Correlation Coefficient
SI	Swarm Intelligence
sMAPE	Symmetric Mean Absolute Percentage Error
SOM	Self-Organizing Maps
STD	Standard Deviations
TIC or U	Theil’s Inequality Coefficient
TSP	Travelling Salesman Problem
TSS	Tournament Selection Size
WCF	Windows Communication Foundation

Abstract

Air quality has emerged as an acute environmental problem, especially in densely populated areas, causing amongst other things negative effects on health. Air quality forecasting system can potentially facilitate amelioration of the situation by providing alerts regarding potential high air pollution levels to the public (to enable them to minimise their personal air pollution exposure) and to the authorities (supporting the decision making process and allowing them to take emergency measures).

Creating an Air Quality (AQ) forecasting system with the aid of data-driven models presupposes the availability of historical data, which has to be appropriately pre-processed before it can be used. This pre-processing is required because environmental datasets often include measurement errors, noise, outliers and missing data. Moreover, it is important that the efficacy of any forecasting model be determined. This can be achieved by using appropriate indices to compare the model's forecasts with the actual situational values that transpire.

This thesis documents the research undertaken to:

1) Investigate the process for developing computational intelligence and statistical methods to perform forecasting of environmental parameters;
2) Develop a methodology that identifies the optimum data pre-processing methods and model characteristics that leads to the highest forecasting accuracy (i.e. the best combination of methods);
3) Identify an optimum methodology to evaluate the forecasting performance of the data-driven models on an operational basis.

The models developed to achieve the first aim are able to predict the values of the environmental parameters with a superior forecasting accuracy in comparison to previously published results. Moreover, a semi-automatic procedure was created to perform forecasting via data-driven models, which can be generalized and applied to other locations and is thus expected to be useful in developing and implementing operational air quality management and forecasting systems for environmental parameters.

To address the second aim, the Daphne Optimization Methodology was introduced for optimising the selection of data pre-processing methods and data modelling algorithms in a comparatively shorter time than with the traditional use of the optimization algorithms. The Daphne Optimization Methodology can be applied at each stage of the process; from selecting the input as well as the target dataset (e.g. air quality and meteorological parameters) to the forecasting of the target parameter, taking into account specific performance optimization criteria. Such a holistic optimisation procedure appears in the literature for the first time as a result of this thesis.

In order to achieve the third aim, two new forecasting performance indices were developed, which combine the characteristics of existing indices. The new indices are suitable for use in an automated operational forecasting system. In addition, a methodology to increase the confidence in the estimation of the forecasting performance of different indices by using confidence intervals was introduced, which use relative weights referred to as "penalties". When the new forecasting performance indices are combined with the use of penalties, the confidence for the estimation of the forecasting performance of a model is higher than any studied single measure.

The proposed new forecasting performance indices and the Daphne Optimization Methodology provide the necessary framework to support the creation of an automated online air quality forecasting system.

Declaration

This is to certify that, except where specific reference is made, the work described in this thesis is the result of the candidate’s research. Neither this thesis, nor any part of it, has been presented, or is currently submitted, in candidature for any degree at any other University.

…………………………………………………………………

Ioannis Kyriakidis
University of South Wales, June 2018

…………………………………………………………………

Professor Andrew Ware – Director of Studies
University of South Wales, June 2018

Acknowledgements

This PhD is a result of intensive work for many years. I would never have been able to finish my PhD without the guidance of my supervisors, help from friends, and support from my family and girlfriend.

First and foremost I offer my sincerest gratitude to my three supervisors, Prof. Andrew Ware, Prof. George Papadourakis and Prof. Kostas Karatzas, who have been there for me throughout the entire work. I appreciate their contribution in time, ideas and funding. They have offered me continuous advices and encouragement.

Besides my supervisors, I would like to sincerely thank Prof. Jaakko Kukkonen for his contribution in time, ideas and suggestions in order to improve the quality of the work.

My thanks also go to Prof. Lars Nolle for an interesting discussion at the beginning of my PhD, which helped me organize the priorities of my work and improve its quality.

I would like to thank my friend Pavlidis Theodoros, for his support and for helping me decide on the name "Daphne" for the developed optimization methodology.

Last but not the least, I would like to thank my family and friends for their love, supporting me and encouraging me with their best wishes. Finally, I would like to thank my fiancée, Antonia Mathioudaki, for her support and patient (in the good times and bad) for all years of my PhD work.

Thus, I express my gratitude to all those who encouraged, inspired, supported and helped me in my PhD.

Ioannis Kyriakidis
University of South Wales
June 2018

Chapter 1

Introduction

1.1 Motivation

Pollution is a term commonly used to describe the existence of substances or forms of energy in the environment which are usually not found in its ambient state and have negative effects on humans, the ecosystem and materials. There are several types of pollution with one of the most well-known forms being air pollution. Breathed air is a mixture of gases and aerosol particles, air pollution may thus be defined as the presence of pollutants (i.e. any substance introduced directly or indirectly by humans or natural activities) in the ambient air, in quantities that can have negative effects in the short or long term (EUR-Lex, 1999). There are many contributing factors to the quality of air but increasingly industrial activities and the modern way of life (e.g. traffic loads and urban activities) can have a negative impact (Sydow, 2010).

The motivation for the research articulated in this thesis was to develop an air quality forecasting system “fuelled” solely by air quality monitoring data, in order to provide early warning to the general public and the authorities regarding high air pollution levels.

The exposure to air pollution has been associated with a wide range of health effects including increased respiratory symptoms, hospitalization for heart or lung diseases, and even premature death. Hazardous air pollutants may cause cancer or other serious health effects (such as reproductive effects or birth defects) (Health Canada, 1997) (US EPA, 2017b).

Air pollution has emerged as the deadliest form of pollution and the fourth leading risk factor for premature deaths worldwide. According to The World Bank & Institute for Health Metrics and Evaluation (2016), in 2013, 5.5 million premature deaths worldwide were attributable to air pollution. From the economic perspective, those deaths cost the global economy $225 billion (United States dollars) in lost labour income in 2013 (The World Bank & Institute for Health Metrics and Evaluation, 2016). This perspective is important in order to strengthen the business case for governments to act ambitiously in managing and reducing air pollution.

Most air pollutants are emitted into the atmosphere via anthropogenic activities, including mobile sources (such as cars, buses and planes) and stationary sources (such as factories and power plants). Air pollutants can also have natural sources such as volcanic eruptions and forest fires (Richards, et al., 2006) (US EPA, 2017a). People can be exposed to air pollutants in two distinct ways. Either by directly breathing polluted air or indirectly by other means, such as (US EPA, 2017a):

- Eating contaminated food products including those from animals (meat, milk, or eggs) fed on contaminated plants, and fruits and vegetables grown in contaminated soil (on which air pollutants have been deposited);
- Drinking contaminated water;
- Having a direct (skin) contact with contaminated soil, dust, or water.

In order to manage air pollution and to minimize its impact on humans and the ecosystem, a set of regulations has been established that require AQ monitoring and modelling, for measuring the current status of the environment and forecasting its state one day in advance, thus enabling the necessary information to be provided to decision makers and the general public in order that they may take any appropriate action.

The provision of future AQ forecasts would be valuable to the public (especially for the sensitive groups, such as the elderly and children) in order that they might have time to take measures that would help minimise their personal air pollution exposure. Moreover, the authorities could use this information in order to take emergency measures (such as temporarily shutting off major emission sources, motivating car-pooling and the use of public transportation) to reduce air pollution and to avoid or limit the exposures to unhealthy levels (Zhang, et al., 2012) (Jiang, et al., 2015).

1.2 Research Statement

Air quality modelling is the simulation of the ambient concentration of criteria pollutants (see Section 2.1.2) found within the atmosphere (Collett & Oduyemi, 1997), and air quality forecasting is the application of AQ modelling for foretelling air pollution levels. Different mathematical approaches can be used in air quality modelling, such as data-driven and deterministic models. In this work, the mathematical approach of data-driven models is selected in order to forecast AQ parameters. Additional information regarding the different mathematical approaches and the reasons that the data-driven models were selected is provided in Section 2.1.

Building a forecasting system with the aid of data-driven models (Solomatine, et al., 2009) requires historical data (input data and targeted forecasting parameters), which has to be pre-processed appropriately before it can be used by the data-driven models. The produced results (forecasted values) of the data-driven model are evaluated by comparing them with the actual values (i.e. by computing the forecasting accuracy with the aid of appropriate indices).

In this study the term "data-driven models" is used to describe the use of statistical methods and computational intelligence methods. Data-driven models uncover hidden insights through learning from relationships and patterns in data. It is also important to note that the used data-driven models do not directly employ physical and chemical characteristics of air pollution, but follow robust mathematical rules (different per algorithm employed) which “imitate” the data and thus indirectly depict the basic relationships and mechanisms of dependency and correlation.

Several data-driven algorithms can be used as the basis for AQ forecasts, while algorithm characteristics and parameterisation may vary and greatly affect the modelling outcome. The forecasting accuracy of the data-driven models is critically dependent on the quality of the data. Wrong or poor data pre-processing may lead to incorrect results. The data processing chain in Knowledge Discovery in Data (KDD) (see Figure 1-1) is important for detecting and handling errors, and for correctly pre-processing the data for the selected data-driven models. For example, data-driven models do not always improve their performance by including as many input parameter as available (due to the noise and quality problems that aforementioned data may carry), but in many cases a parameter prioritization and selection step may improve the overall model performance. The KDD steps should be taken into consideration and should be included in any methodology that aims to improve the overall model results by embracing data pre-processing in the optimization loop. In the case of environmental data, it is common to include errors, such as measurement errors (from automatic monitoring, sensors, etc.), noise, outliers and missing data, which result in low-quality data (Gibert, et al., 2008), and thus reinforces the necessity to use the KDD steps.

In this work, the term "forecasting model" is used for the combination of different input data and targeted forecasting parameters, pre-processing methods, data-driven algorithms and parameterization-configurations, in order to perform forecasts. Each forecasting model is evaluated for its forecasting accuracy, on the basis of one or more forecasting verification indices. The forecasting model that achieves the highest forecasting accuracy is characterised as the "best forecasting model". In addition, the term “forecasting method” is used as a general reference to a data-driven model used to perform forecasts.

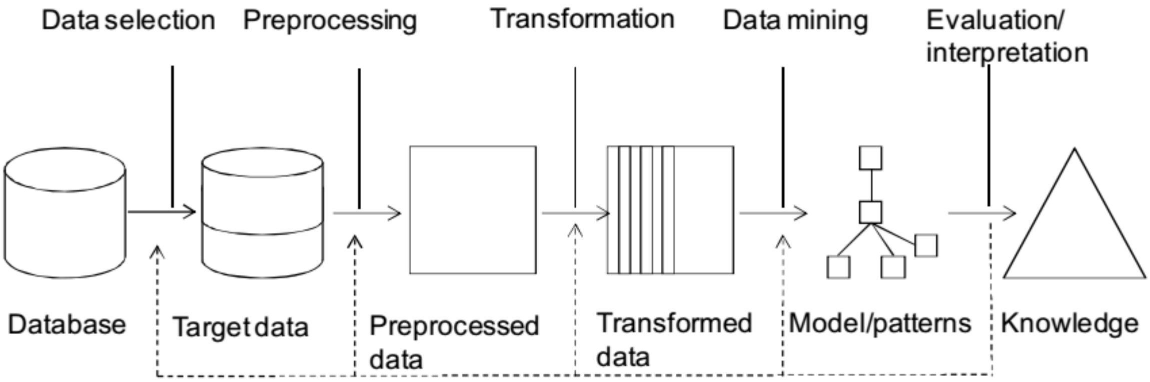


Figure 1-1: The data processing chain in knowledge discovery in databases (Gibert, et al., 2008)

The work documented in this thesis focuses on the optimization of the environmental forecasting workflow (from data pre-processing to algorithm selection and parameterisation) by using data-driven models, where the forecasting accuracy is the optimization criterion used. For that purpose, the research questions of this work were:

1. Given a set of continuously updated time series data (air quality and meteorological data), is it possible to develop data-driven models that will predict the future value(s) of some of these parameter(s) (i.e. future concentration levels)? If yes, are these models better than existing ones (if any), and why?
2. If such models are developed, which are the best ways to evaluate their performance on an operational basis? Are the existing measures sufficient?
3. Is there a methodology available that can allow for the "automatic" selection of data pre-processing methods and data-driven models and their characteristics that lead to "best" forecasting accuracy?

1.3 Summary of Research Work and Contributions

The research plan of this work was organized on the basis of the relevant research questions and produced results contributing to the scientific field of data-driven environmental modelling (see Section 1.2) as follows:

1. Experiments were performed to investigate the effectiveness of data-driven models in order to forecast environmental parameters.
    - Results indicate the importance of data pre-processing, due to its influence in the forecasting performance of the data-driven models.
    - New forecasting models were developed for the selected AQ parameters (Ozone concentrations, Benzene concentrations and Common Air Quality Index) for the cities of Thessaloniki and Athens of Greece. The forecasting performance of the developed data-driven models is at the highest level in comparison to the relevant literature.
    - The described semi-automatic procedure to perform forecasting via data-driven models can be generalized to other locations and is expected to be useful for the implementation of operational forecasting systems for environmental parameters.

2. A literature review was undertaken which summarizes the most frequently used statistical measures of forecasting performance evaluation and their main characteristics and disadvantages. It was evident that existing measures have limitations in their ability to estimate AQ model forecasting performance. On this basis:
    - A new method was introduced in order to increase the confidence in the estimation (in terms of consistency) of the forecasting performance of different measures by using relative weights (referred to as penalties) based on the bounds of the confidence intervals.
    - Two new forecasting performance indices (FPIs) were introduced, which combine the characteristics of existing measures and do not suffer from the same degree of inconsistencies and disadvantages, and can thus be used in an automated operational forecasting system. The confidence in the estimation of the forecasting performance of a model is higher when the new forecasting performance indices were used in comparison to any examined single measure.
3. An optimization methodology was developed for the whole air quality forecasting chain in order to improve data pre-processing and data-driven models, in terms of the performance of various forecasting algorithms:
    - The Daphne Optimization Methodology was introduced which automatically converges towards a "solution" (combination of data pre-processing methods and data-driven models) which is "close to the optimum". In addition, this is achieved in comparatively shorter time than with the traditional use of the optimization algorithms. Such an optimisation methodology has not been described in literature previously.

1.4 Thesis Outline

The rest of this thesis is structured as follows. Chapter 2 provides the necessary knowledge background concerning the research described: It includes an introduction and discussion on air quality forecasting, a comparison of the characteristics of the available forecasting models categories and the criteria considered in order to select the forecasting model category of this work. Chapter 2 also reviews the main existing approaches in the research area referring to the air quality forecasting by using data-driven models and presents some of the developed operational air quality forecasting systems. Moreover, it describes the variable types and model verification, the optimization algorithms, the pre-processing and forecasting methods employed, and the study areas and their characteristics.

Chapter 3 describes the experiments performed in order to investigate the use of data-driven models in order to forecast AQ parameters.

Chapter 4 documents the capabilities and limitations of existing forecasting performance indices. By evaluating the selected indices, latent characteristics were revealed that led to the selection of FPIs to be used in the construction of new statistical forecasting indices. Thereafter, Chapter 4 presents a) an explanation of how the new indices were built, b) the evaluation results, c) the performance of the new indices compared to existing indices, and d) the conclusions reached. The outcome of this effort is tested by evaluating the forecasting performance by using a) the existing evaluation indices and b) the new indices, for a set of different forecasting data-driven models. In order to increase the confidence in the estimation of the forecasting performance, a new methodology was introduced, which is based on the bounds of the confidence intervals.

Chapter 5 describes the development of the data-driven AQ forecasting optimization methodology, and more specifically it details:

- The phases of the optimization methodology;
- The steps within the optimization problem and its division into discrete computational steps (that formulate a general procedure);
- The methods that were used to deal with the problem of each computational step;
- How the optimization methodology is employed by the selected optimization algorithms;
- The evaluated models and the procedure that is performed in order to evaluate them;
- The results of this part of the research and discussion of the findings.

Finally, Chapter 6 reviews how the research questions that motivated the research were addressed, draws conclusions about what has been found, and articulates the contribution that the work has made.

Chapter 2

Background and Methodology

2.1 Air Quality Forecasting

Urban air quality has emerged as an acute environmental problem, especially for densely populated areas, causing negative effects on health, ecosystems and materials (including historic and cultural heritage monuments) (Doytchinov, 2012). Health effects may range from difficulty in breathing to an aggravation of existing cardiac and respiratory conditions. These effects may result in visits to a doctor or an emergency room, increase use of medications, admission to hospital or even premature death (Health Canada, 1997). According to the World Health Organization (WHO) (2005) more than two million premature deaths, each year can be attributed to the effects of air pollution.

Air Quality (AQ) is defined (Collett & Oduyemi, 1997) as a "measure of the degree of ambient atmospheric pollution, relative to the potential to inflict harm on the environment". Moreover, the European Union (EU) has developed an extensive body of legislation which establishes health based standards and objectives for a number of pollutants in ambient air (European Commission, 2017). These standards prescribe, among other things, the limit values of different air pollutants, the permitted exceedances (margin of tolerance) and the threshold levels used to inform and alert authorities. Air pollutants can be in the form of solid particles, liquid droplets, or gases (SEPA, 2017) and can have local, regional, and global impacts. Their limit value is defined in the Clean Air for Europe (CAFE) directive (EUR-Lex, 2008) as a level to be attained within a given period and not to be exceeded once attained, which is based on scientific knowledge, with the aim of avoiding, preventing or reducing harmful effects on human health and/or the environment as a whole. The percentage by which limit values may be exceeded (subject to the conditions established in the (EUR-Lex, 2008)) is referred to as the margin of tolerance. The threshold to alert authorities is a level, which when exceeded there is a risk to human health from brief exposure and at which immediate steps are to be taken.

2.1.1 Processes Affecting Pollution Levels

There are various processes that affect AQ levels (Seinfeld & Pandis, 2006):

The term atmospheric dispersion refers to a series of processes that result in the transport and diffusion of substances in the atmosphere. These processes are influenced by several factors, such as: wind direction and wind speed, the geographical area, and atmospheric heating effects. Thus, air pollution concentration varies spatially and temporarily causing the air pollution pattern to vary according to location and time, due to difference in meteorological and topographical conditions (WHO, 2008).

Pollutants can be carried thousands of kilometres from their original source, sometimes from urban areas to less populated areas (Health Canada, 1997). Advection refers to the transport (usually horizontal movement) of pollutants by means of wind fields, whereas diffusion involves small-scale turbulent mixing of pollutants. By definition, advection transports the pollutants without a significant change in their concentrations, whereas diffusion involves dilution which leads to lowering of pollutant concentrations (Kukkonen, et al., 2012).

As wind speed increases, the volume of moving air in a period of time also increases. Thus, if the emission rate is relatively constant and the wind speed is doubled then the pollutant concentration will be halved (meaning that the concentration is an inverse function of the wind speed). In the case of wind direction, when it is relatively constant, the same area is continuously exposed to high pollutant concentration. If the wind direction is constantly shifting, pollutants are dispersed over a larger area, and thus concentrations are lower over any exposed area (Liu & Liptak, 1999).

Air pollution concentration can also be effected by air temperature inversions, in which vertical air movement is impeded. In air temperature inversions, a layer of warm air is above a layer of relatively cooler air, which prevents the normal vertical mixing (Dye, et al., 2003) (Phalen & Phalen, 2013). Inversions can trap dangerous concentrations of pollutants in the cool air below, sometimes causing dense smog over urban areas. This is because inversions may occur at altitudes less than 100 m (WHO, 2008) (Phalen & Phalen, 2013).

Air pollution is also influenced by chemical and physical processes in the atmosphere:

Physical and chemical processes transform and/or remove chemical compounds emitted into the atmosphere and thus affect the concentration of pollutants (Atkinson, 1988). Such processes include emissions, nucleation, coagulation, condensation, evaporation and dry deposition (Jacobson, 1997).

Figure 2-1 illustrates the different processes involved in the production, growth, and eventual removal of atmospheric aerosol particles (Jacob, 1999). The typical size range of gas molecules is 10^{-4}-10^{-3} μm and when the gases are converted to small liquid droplets (ultrafine particles) their size range is 10^{-3}-10^{-2} μm. This transition phase is called nucleation. These ultrafine aerosols grow rapidly to the 0.01-1 μm fine aerosol size range by condensation of gases (i.e. gases condense into a small solid particle to form a liquid droplet) and by coagulation (i.e. collisions between particles during their random motions). Further growth (beyond 1 μm) is much slower because in this state the particles are too large to grow rapidly by condensation of gases, and because large particles have a slower random motion and thus lower coagulation rate. Aerosol particles originating from condensation of gases tend therefore to accumulate in the 0.01-1 μm size range, often called the accumulation mode. These particles are too small to sediment at a significant rate and are removed from the atmosphere mainly by scavenging by cloud droplets and subsequent rainout (or direct scavenging by raindrops) (Jacob, 1999).

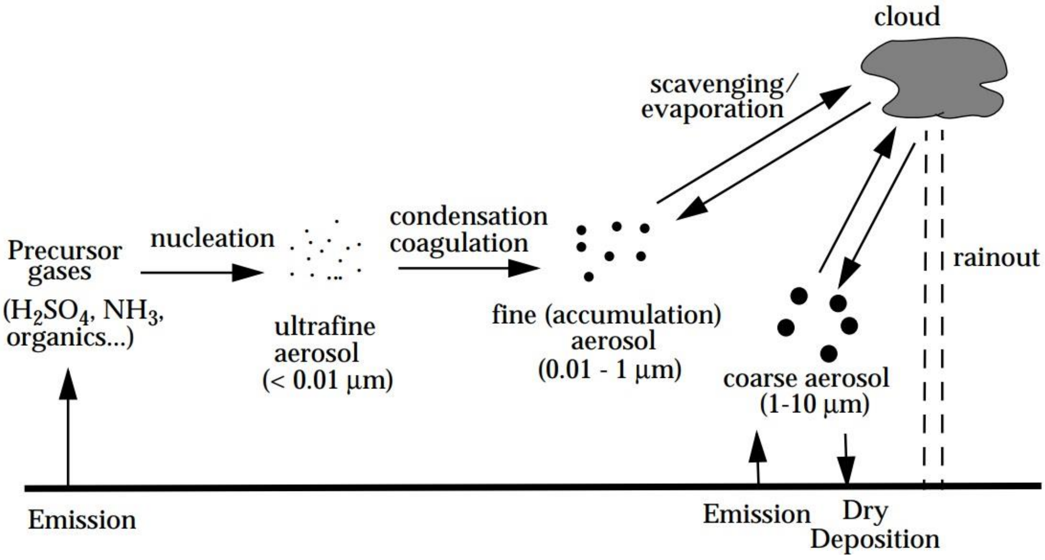


Figure 2-1: Production, growth, and removal of atmospheric aerosols (Jacob, 1999)

2.1.2 Sources of Air Pollutants

Air pollutants can be categorized either as primary (those emitted directly into the atmosphere), or secondary (those forming in the atmosphere as a result of the reactions among other chemicals) (Kumar, et al., 2016) (Health Canada, 1997). In general, there are two types of air pollutant emission sources, point and nonpoint (Kumar, et al., 2016). Point sources are those released at a single point (such as a smokestack of a factory), while nonpoint sources are those not released at discrete points and are not easily attributed to a single source. Nonpoint sources can often be categorised as wide area or line sources; examples of a wide area source are dust blown by the wind or wildfires, while an example of a line source is road traffic pollution.

Air pollutants are emitted into the atmosphere from anthropogenic (human) activities, but also from natural sources, such as windblown dust, wildfires and sea drops. The anthropogenic sources can be categories as: stationary sources (such as factories, power plants, and smelters); area sources (which are smaller sources such as dry cleaners, degreasing operations, housing developments); and mobile sources (such as cars, buses and planes) (Richards, et al., 2006). The stationary sources are also referred to as point sources. The area sources and mobile sources can be considered either as a point or nonpoint sources, depending on the case study.

Concerning the protection of human health, six air pollutants have been selected by the U.S. Environmental Protection Agency (EPA) when setting the National Ambient Air Quality Standards (NAAQSs) (2018). These pollutants are sulphur dioxide (SO_2), nitrogen dioxide (NO_2), carbon monoxide (CO), ozone (O_3), lead (Pb), and particulate matter with mean aerodynamic diameters less than or equal to 2.5 mm ($PM_{2.5}$) and 10 mm (PM_{10}). In the European Union (EU) regulated pollutants include the six pollutants regulated in the U.S. plus benzene (C_6H_6), which is a volatile organic compound (VOC) with known carcinogenic health effects (Richards, et al., 2006) (Zhang, et al., 2012) (McIntosh & Pontius, 2017). These air pollutants are commonly known as criteria pollutants because numerical standards have been set for them to establish safe environmental levels. Table 2-1 presents the main sources of the seven EU regulated pollutants (WHO, 2008) (Richards, et al., 2006) (McIntosh & Pontius, 2017) (Kumar, et al., 2016), (see Appendix IV for additional information regarding some of the major air pollutants).

Table 2-1: The sources of the seven European Union regulated pollutants

Pollutant	Anthropogenic Sources	Natural Sources	Health Effects
Sulphur Dioxide	Industrial sites and their raw materials such as: power plants, burning of fossil fuels, crude oil and coal	Volcanic eruptions	Irritation to the eyes and respiratory system, asthma hospitalization, chronic bronchitis and emphysema
Nitrogen Dioxide	Fossil fuel combustion and atmospheric reactions such as power plants and vehicle tailpipes	Lightning, plants and soil	Irritation to the respiratory system (increase the symptoms in asthmatics, decrease the lung growth)
Carbon Monoxide	Emitted mostly from motor vehicles (when carbon-containing fuel is not burned completely)	Wildfires	Increase the asthma hospitalization, reducing oxygen delivery in the body and causing death at high concentrations
Ozone	A secondary pollutant formed at ground level (also known as bad ozone) by reactions involving nitrogen oxides (vehicle emissions), volatile organic compounds (VOCs) in the presence of heat and sunlight.	Ozone exists in the stratosphere (also known as good ozone), which provides protection from harmful ultraviolet radiation.	Increase the respiratory and asthma hospitalization, decreases in lung function and increased respiratory symptoms such as chest pains and coughing
Lead	Once a major pollutant present in motor vehicles. These days lead comes mostly from ore smelters and metal processing operations	-	Damage the brain and central nervous system to cause coma, convulsions and even death
Particulate Matter	Combustion engines, fuel burning (wood stoves, fireplaces), industry, construction	Agricultural activities (soil and dust), traffic on unpaved roads, sand from deserts, smoke from fires	Penetrate deeply into the lungs (decrease lung growth and function), increase the infant respiratory mortality, increase the symptoms in asthmatics
Benzene	Produced as fuel by-products in a combustion process	-	Carcinogen, involved in the production of ground level ozone, because is the most common VOC

2.1.3 Air Quality Management and Forecasting

Air quality management refers to the activities that a regulatory authority performs in order to protect human health and the environment from the adverse effects of air pollution. Authorities have two options in order to reduce the effects of air pollution: improving air quality and preventing exposure (Honoré, et al., 2008). In Europe, over the past 40 years, a broad range of environmental legislation has been implemented. This has helped to address some of the most serious environmental concerns of citizens and businesses in the EU. Emissions of air pollutants (as well of water and soil) have been reduced significantly over the past decades (EUR-Lex, 2013).

Despite all efforts, there are still exceedances of the limit values, which can cause serious health problems. In order to minimise the exposure to high concentrations of air pollutants, it is important to provide early warning to both authorities and the general public, particularly the most sensitive groups (elderly, children and asthmatics) (Honoré, et al., 2008) (Jiang, et al., 2015). The CAFE directive (EUR-Lex, 2008) in Europe clearly states that information to be provided to the public should include:

- Information on observed exceedance(s)
- Forecast for the following afternoon/day(s)
- Information on the type of population concerned, possible health effects and recommended behaviour

Air quality forecasting is the tool that can help to meet the early warning objectives (Honoré, et al., 2008). The inclusion of AQ forecasting models to an Air Quality Management System cannot by itself solve the described problems, it is only one component of the overall air quality management picture (CENR, 2001) (San José, et al., 2006) (Sunil, et al., 2015). Air quality forecasts are generally classified into the following broad areas on the basis of their goal (Dye, et al., 2003) (Dabberdt, et al., 2004):

- **Health Alerts:** Several cities inform the public when air pollution levels exceed the specified levels. The warnings are directed to inform specific parts of the population (that are particularly sensitive to air pollution) in order to plan and take preventative actions to minimise their personal air pollution exposure during high-pollution episodes (Jiang, et al., 2015). For example, a growing number of states in the U.S. are using summertime ozone forecasts to alert sensitive populations (CENR, 2001).

- **Emergency Control Measures:** Authorities can use air quality forecasts (e.g. the forecasting of next day's air pollution levels) to take efficient emergency control measures (such as temporarily shutting off major emission sources, motivating car-pooling and the use of public transportation) to reduce air pollution and to avoid or limit the exposures to unhealthy levels (Zhang, et al., 2012). For example, many cities in the U.S. offer free access to public transportation to reduce automobile emissions on days with high ozone concentrations (CENR, 2001).
- **Control Strategy:** In air quality, a control strategy is a set of specific techniques and measures, which have been identified and implemented in order to achieve air pollution reduction and formulate an air quality standard or goal. For example, Kumar, et al., (2016) study the formulation of an air quality management framework for industrial air pollution, based on the impact of different control scenarios. These scenarios evaluate the increment of stack height and fuel change (from high emission fuel to low emission fuel). The best scenario provides 79% reduction for SO_2 and 69% reduction for PM.
- **Emergency response:** Smoke from wildfires affects the air quality and the visibility in the area that can cause accidents and increase traffic. An erupting volcano will release ash and gases that can be harmful to health. Reliable forecasts can offer rerouting options, for automobiles and air traffic, to reduce the possibility of accidents.

2.1.4 Air Quality Modelling

According to Collett & Oduyemi, (1997) air quality modelling (also known as air pollution modelling) can be viewed as the attempt to simulate and thus predict the ambient concentration of criteria pollutants found within the atmosphere. In general, AQ forecasting models can be categorised according to the following three mathematical approaches (Collett & Oduyemi, 1997) (Jiang, et al., 2015):

- Data-driven models
- Deterministic models
- Hybrid models

Data-driven modelling is based on the fact that meteorological and air pollutant concentrations are statistically related (Zhang, et al., 2012). Data-driven models usually require a large quantity of (historical) measured data, including a variety of atmospheric and meteorological variables. Generally, data-driven models express complex relations between meteorological conditions, concentrations of air pollutants and potential predictors (criteria pollutants) (Zhang, et al., 2012), and are thus applicable when the primary goal is to forecast specific AQ parameters (Konovalov, 2009). These models are extremely powerful in designing early warning systems, but cannot be used to study what-if scenarios (control strategies).

In data-driven modelling, there is a continuous process of learning, modelling and prediction. When new measured data are available they are fed into the data-driven model in order to facilitate training (learning process) and thus a new refined model is produced (Konovalov, 2009). The goal is to find a model which produces maximum forecasting performance (the best forecasting model) with a test dataset (data that were not used in the training phase) to simulate an operational system. In practice (e.g. in an operational forecasting system), the trained best model is used as an analytical tool to produce forecasts by using other input data. For example, in order to produce tomorrow's forecast, the data observed today may be used an input to the model. Data-driven modelling can be achieved with the aid of computational intelligence (CI) and statistical methods. The most common methods include classification and regression trees (CART), linear regression (LR), artificial neural networks (ANNs) and fuzzy logic (FL) (Zhang, et al., 2012). These models are described in detail in Section 2.5.

Strengths of Data-driven models (Dye, et al., 2003) (Zhang, et al., 2012):

- They are commonly used and easy to operate;
- They generally produce good forecasts;
- Once the model is developed (e.g. an ANN model), the forecaster does not need specific expertise to operate it;
- Generally they are more suitable for the description of complex site-specific relations between concentrations of air pollutants and potential predictors. Some methods (such as ANNs and FL) can detect non-linear relationships between variables.

Limitations of Data-driven models (Zhang, et al., 2012):

- Usually they are limited to the area and conditions for which measurements were collected and cannot be generalized to other regions with different chemical and meteorological conditions;
- Usually requires a large quantity of measurement data;
- They cannot predict concentrations during periods of unusual emissions and/or meteorological conditions that deviate significantly from the historical record;
- The nature of data-driven modelling does not enable better understanding of chemical and physical processes.

Deterministic air quality models (also known as 3-D numerical air quality models) attempt to employ physical laws, chemical reactions and simplifying assumptions to mathematically represent the important processes that affect ambient air quality (Collett & Oduyemi, 1997) (Dye, et al., 2003) (CENR, 2001). Deterministic AQ models are capable of forecasting temporally and spatially resolved, concentrations under both typical and atypical conditions and in areas that are not monitored (Jiang, et al., 2015). These models can be used for operational AQ information provision to decision makers and the general public and in case study analyses to understand air pollution processes and to estimate the effects of emissions changes on pollutant concentrations during episodic conditions (Dye, et al., 2003). Deterministic AQ models consist of other sub-models, in particular by a meteorological model, an emissions model, and a chemical model (San José, et al., 2006) (Dye, et al., 2003) (CENR, 2001).

- **Meteorological models:** simulate the conditions that determine transport and mixing and influence chemistry (solar intensity, temperature, humidity, etc.), emissions (e.g. temperature), and deposition.
- **Emissions models:** simulate the temporal, spatial, and chemical distribution of emissions of a selected group of pollutants, and/or its precursors (in the case of secondary pollutants), from both anthropogenic and natural sources.
- **Chemical models:** use the output (i.e. forecasts) of the meteorological and emissions models as inputs, in order to simulate the transformation by chemical reaction (e.g. the transformation of primary pollutants into secondary pollutants), and removal of air pollution.

Deterministic AQ models are classified as being either Lagrangian or Eulerian depending on the method used to simulate the time-varying distribution of pollution concentrations (Dye, et al., 2003) (CENR, 2001) (National Research Council, 1999).

- Lagrangian (trajectory) models follow individual air parcels over time using the meteorological data to transport and disperse the pollutants. The model's air parcel is allowed to move over the air basin, following a trajectory calculated from the wind fields (i.e. a Lagrangian simulation) (National Research Council, 1999). This approach is computationally efficient when treating a limited number of emission sources. However, it is difficult to properly characterise the interaction of a large number of individual sources when nonlinear chemistry is involved and, additionally, these models have limited usefulness in forecasting secondary pollutants. For these reasons, the Eulerian models are often preferred to forecast air quality (Dye, et al., 2003).
- Eulerian models (also called 3-D Eulerian models or grid models) use a grid of cells (vertical and horizontal) where the chemical transformation equations are solved in each cell and pollutants are exchanged between cells. Eulerian models can produce three-dimensional concentration fields for several pollutants but require significant computational power and storage (disk space), and expertise (Dye, et al., 2003) (CENR, 2001). For that reason, these modelling analyses may take years to set up, evaluate, and complete (Dye, et al., 2003).

  The computational cost is related to the model's resolution (i.e. the grid cell size). Smaller grid cells will result in higher resolution and generally greater model accuracy but at a higher computational cost.

For example, the System for Integrated modeLling of Atmospheric coMposition (SILAM, 2014) has a minimum requirement of an Intel® Xeon® CPU with 16 or 8 dual-core processors (64-bit processor) with 24 GB RAM memory in order to perform forecasting. In addition, the simulations of the CALIOPE Air Quality Forecast System (CALIOPE-AQFS) are run on the MareNostrum supercomputer (Intel Xeon E5-2670, 16 CPUs and 64 GB RAM memory per node) at the Barcelona Supercomputing Center (Pay, et al., 2014). By using parallelization of meteorological and air quality models, MareNostrum uses up to 256 CPUs. As an example of computational costs, the most computationally demanding domain was in Andalusia, Spain, where in order to provide 48-hour forecast at 1 km resolution (666 × 358 cells), 256 CPU max and 5 hours required.

Strengths of Deterministic Models (Zhang, et al., 2012) (Konovalov, 2009) (Dye, et al., 2003):

- They are capable of forecasting temporally and spatially resolved concentrations. The model forecasts can be presented as maps of air quality to show how predicted air quality varies over a region per hour (or per day);
- They provide scientific understanding of pollutant processes that control air pollution in a specific area;
- They can be used for the analysis of hypothetical scenarios (such as for control strategy);
- They can predict air pollution in areas where there are no air quality measurements.

Limitations of Deterministic models (Zhang, et al., 2012) (Konovalov, 2009) (Dye, et al., 2003):

- The insufficient knowledge of pollutant sources, the required approximations and simplifications, and the inaccuracies in description of physicochemical processes can lead to biases in forecasted concentrations;
- The inaccuracies and uncertainties in inputs variables that contribute to pollution;
- The accuracy of these models depends on the accuracy of meteorological predictions and emission estimates;
- They are “expensive” in terms of the investment and time required to acquire all the necessary input data (topography, emission inventory, etc.)
- They are computationally expensive and require a high-speed computer system with a large memory and disk storage.

Hybrid models are used in order to generate forecasts with improved accuracy, either by combining deterministic and data-driven models or by combining two or more methods of the described models. For example, deterministic models can be combined with bias correction techniques (such as statistical adaptation and/or data assimilation). Another common approach is to generate an ensemble of forecasts from different methods (either from deterministic or data-driven models) and used to identify the "mean" or "probable" forecast. It worthies mentioning that the ensemble-based forecasting is also applied in data-driven modelling (Breiman, 1996) and has been implemented in this thesis also (see Section 2.5.4)

2.1.5 Forecasting Aspects Classification

There are various ways described in the literature to classify forecasts, depending on the forecasting techniques, time horizon, methods, and models. Figure 2-2 presents a way to classify the different forecasting aspects.

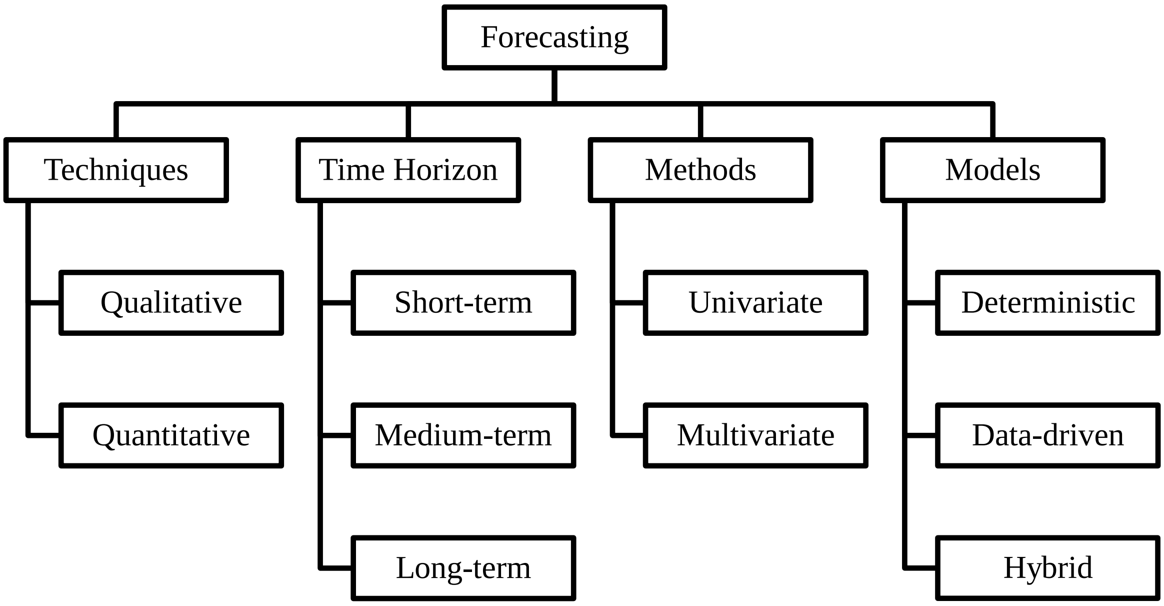


Figure 2-2: A classification scheme of different forecasting aspects

The forecasting techniques are classified into two types, qualitative and quantitative.

1. **Qualitative** forecasting techniques are subjective and require the judgment of experts (by experience). Qualitative forecasts are often used in situations where there is little or no historical data.
2. **Quantitative** forecasting techniques make formal use of historical data and a forecasting model. Quantitative forecasting analyses large amounts of historical data in order to summarize patterns in the data and expresses a statistical relationship between previous and current values of the variable. Thus, the quantitative forecasting relies on the identification of repeated patterns in the data.

A forecasting problem can be classified depending on the forecasting time horizon as short-term, medium-term, and long-term. As Montgomery, et al. (2015) explain, the short-term and medium-term forecasting is typically based on identifying, modelling, and extrapolating the patterns found in historical data. These historical data usually exhibit inertia and do not change dramatically very quickly, and thus, data-driven methods are very useful.

- In short-term forecasting, the forecasts are made for short time periods (such as days, weeks, and months) and used for operating decisions;
- Medium-term forecasts are made for a time period from 1 to 2 years and used for tactical decisions;
- Long-term forecasts are made for a time period greater than two years, and thus, used for strategic decisions.

In terms of the number of features (independent variables) used, modelling methods may be classified into two groups, univariate and multivariate (Kasabov, 1998). In practice, a combination of these methods may be used (Chatfield, 2003).

- In univariate methods, the forecasts are based only on present or past observations of one independent variable;
- In multivariate methods, the forecasts are based on more than one independent variable.

The forecasting models categories have been described in Section 2.1.4, along with their strengths and limitations. The forecasting approach used is also determined from the need to either categorise (classify) AQ levels in terms of categorical values (like “low”, “medium”, “high” and “very high”) or to model (and thus predict) the numerical value of the concentration of a pollutant. The former approach is generally called classification modelling, while the latter is called regression modelling (Ethem, 2010).

2.1.6 Selecting the Forecasting Model

The selection of the forecasting method/model depends primarily on the forecasting requirements, the available resources (in terms of historical data and computational power) and experience (Dye, et al., 2003). The overall cost of a particular model is associated with both development and operation of the model. Table 2-2 presents a summary of the factors that contribute to the selection of a forecasting model.

The selection of the forecasting model takes into consideration the factors articulated in Table 2-2 and the different aspects of forecasting (see Section 2.1.5). Thus, in this thesis the main requirement of forecasting was to be able to model and foretell the numerical values (regression modelling) of air pollutants concentrations in a short-term time horizon by using data-driven models. Classification models were used as part of the investigation experiments performed in Chapter 3. For this reason a combination of univariate and multivariate methods were employed. Data-driven models were selected against deterministic for the following reasons:

1. They were part of the research questions (see Section 1.2)
2. They did not require sound knowledge of pollution sources and processes.
3. They require significantly less development and operational effort. Thus, it would be easier to maintain an operational system with lower cost.
4. They require only a personal computer for the development.
5. Concerning data availability and testing, historical data are available through utilizing the European air quality database (AirBase-V8, 2014) (AQ e-Reporting, 2017).
6. They can directly lead to health alerts and support emergency control measures (which is the aim of this work).

Table 2-2: A summary of the factors that contribute to the selection of a forecasting model

		Data-driven models	**Deterministic models**
Forecasting Application	Health Alerts	✓	✓
	Emergency Control Measures	✓	✓
	Control Strategy-Scenarios	-	✓
Available Resources	Historical Data	A large quantity of measurement data	Can work without air quality measurements
	Hardware	Personal Computer	High-speed computer system with large memory and disk storage
	Development Effort	Moderate to High	Very High
	Operational Effort	Moderate	Moderate to High
Other Factors	Experience	Understanding of statistics and methods such as neural networks, regression, CART etc.	High level understanding of meteorological and air quality relationships, and meteorological, emissions, and chemical models.
	Strengths	Develop non-linear relationships between variables	Provide understanding of pollutant processes
	Limitations	Have to be re-trained to be generalized to other regions with different chemical and meteorological conditions.	The accuracy of air quality predictions depends on the accuracy of meteorological predictions and emission estimates.

2.2 Forecast Verification and Variable Types

Forecast verification is an essential component of a forecasting system since it provides with a “measure” of the quality of forecasts (Murphy & Winkler, 1987) (Doswell III, 1996). In the forecast verification process, forecasts are compared to relevant observations with the aid of various statistical measures, commonly referred to as indices. Those measures depict various aspects of the differences between forecasted and observed values of the parameters of interest. This process is also referred to in the literature as Forecast Evaluation (West, 2006) (Diebold & Lopez, 1996) (Laurent & Violante, 2011) (Clark & McCracken, 2011).

Jolliffe & Stephenson (2012) define forecast quality as a multidimensional concept described by several different scalar attributes, as presented in Table 2-3. All of these attributes provide useful information about the performance of a forecasting system. Thus, no single index is sufficient for forecast evaluation, i.e. for judging and comparing forecast quality (Jolliffe & Stephenson, 2012) (Morse, et al., 2005) (Briggs & Levine, 1997) (Mason & Stephenson, 2008).

Table 2-3: The different scalar attributes that describe the forecast quality.

Scalar Attribute	Description
Overall Bias	The total difference between the expectation of the forecasts and the observations (Jolliffe & Stephenson, 2012).
Reliability (or Calibration)	Refers to the degree of correspondence between a forecaster’s subjective probabilities and the observed relative frequency of event occurrences (Mandel & Barnes, 2014).
Uncertainty	The natural variability of the observable variable. For example, in the Brier Score (Brier, 1950) decomposition, uncertainty is the variance of the binary observable variable. It is an important aspect in the performance of a forecasting system, over which the forecaster has no control (Jolliffe & Stephenson, 2012).
Sharpness (or Refinement)	How much the forecasts deviate from the mean for forecasts, or from the mean probabilities for probabilistic forecasts (Jolliffe & Stephenson, 2012).
Accuracy	The average distance/error between forecasts and observations that depends on bias, resolution and uncertainty attributes (Jolliffe & Stephenson, 2012).
Association	The overall strength of the statistical relationship/dependency between forecasts and observations that is independent of the marginal distributions. Linear association is often measured using the correlation coefficient (Jolliffe & Stephenson, 2012).
Resolution (or Discrimination)	Refers to how well a forecaster separates occurrences from non-occurrences (Mandel & Barnes, 2014).

The forecast verification process and, in consequence, the provided measures can be used in order to compare the forecast quality of different forecasting models and different forecasting parameterizations.

The type of variable being verified is an important factor when choosing verification method. One way of addressing forecasting problems it is depending on its type. Thus, forecasting variables can be categorized as (a) Continuous (numerical) variables (regression problem) and (b) as Categorical variables (classification problem).

In forecasting of continuous variables, the forecast is given as a real value (Gorunescu, 2011), in the units of the variable, for example, the temperature in degrees Celsius. The forecast and the verifying observation may take any value of the variable.

In forecasting of categorical variables, the forecast can be considered as discrete value (Gorunescu, 2011); this means that a pre-determined range of values of the variable is used. Some variables are inherently categorical while others may be redefined as categorical variables by selection of threshold values to delineate the categories.

2.3 Related Work

Data-driven modelling can be achieved with the aid of computational intelligence and statistical methods. Computational Intelligence is the study of adaptive mechanisms to enable or facilitate intelligent behaviour in complex and changing environments. These mechanisms exhibit an ability to learn or adapt to new situations, to generalize, abstract, discover and associate (Engelbrecht, 2007). Statistical methods are mathematical formulas, models, and techniques that are used in statistical analysis of data to extract information and provide different ways to assess the robustness of outputs (NIST/SEMATECH, 2003). The following sections present some of the numerous CI-fuelled AQ forecasting models and operational AQ forecasting systems.

2.3.1 Computational Intelligence AQ Forecasting Models

Kalapanidas & Avouris, (2001) presented a prototype (named NEMO) for the short-term prediction of NO_2 maximum concentration levels in Athens, Greece. NEMO is based on a case-based-reasoning approach combining heuristic and statistical techniques. Results showed that the overall performance of NEMO makes it a good candidate to support air pollution experts in operational conditions.

Slini, et al., (2002) developed a stochastic autoregressive integrated moving average (ARIMA) models to forecast the maximum O_3 concentration in Athens, Greece. Nine years long data (1990-1998) were used by the forecasting models. The results are satisfactory concerning the statistical analysis of the index of agreement (which ranges from 0.83 to 0.97) and the coefficient of determination (which ranges from 0.50 to 0.89).

Chaloulakou, et al., (2003) demonstrate the use of artificial neural networks and linear regression techniques for predicting one day in advance the hourly maximum O_3 levels at several monitoring sites (Liosia, Maroussi, Likovrissi and N.Smirni) in Athens, Greece, by using eight years long data (1992-1999). The proposed artificial neural networks provide a considerable improvement (the index of agreement, d_2 ranges from 0.80 to 0.91, and the coefficient of determination ranges from 0.37 to 0.66) in the forecasting of summertime O_3 concentrations over multiple linear regression models (the index of agreement, d_2 ranges from 0.74 to 0.91, and the coefficient of determination ranges from 0.26 to 0.55) based on the same set of input variables.

Zolghadri, et al., (2004) proposed a new strategy which combines an adaptive nonlinear state space-based prediction mechanism, a gain scheduling strategy and neural network techniques to develop an integrated operational warning system. This was achieved by forecasting the daily maximum O_3 concentrations in Bordeaux, France, by using four years long data (1998-2001). Zolghadri, et al., (2004) states that the advantage of their proposed system is the reliable estimates that can be obtained (with index of agreement of d_1=0.78 and d=0.95), while their implementation of the overall method is simple and the computational effort is rationally low (for instance, in contrast to deterministic models).

Barrero, et al., (2006) presented a multiple linear regression with forward stepwise method for the prediction of the daily maximum O_3 concentrations in Errenteria, Spain. Four years long data (1997-2000) were used by the forecasting models, which are based on the day of the year, the relations between O_3, sunlight and NO_x, and previous day O_3 levels. Results showed that the proposed models have an acceptable performance (the index of agreement ranges from 0.86 to 0.89, and the coefficient of determination ranges from 0.60 to 0.67).

Grivas & Chaloulakou, (2006) evaluated the potential of various developed neural network models to provide reliable predictions of PM_{10} hourly concentrations, in Athens, Greece. In addition, a genetic algorithm optimization procedure was used to evaluate the selection of the input variables. The forecasting models used two years long data (2001–2002). The results of the neural network models were considered satisfactory (the index of agreement ranges from 0.80 to 0.89, and the coefficient of determination ranges from 0.50 to 0.67).

For comparison, the multiple linear regression models were developed but were not as good as the neural network models (their coefficient of determination ranges from 0.29 to 0.35). This shows the superiority of the examined neural network models in this study.

Shiva & Khare, (2006) described a step-by-step procedure to forecast the daily average NO_2 concentrations by using artificial neural networks, in Delhi, India. Three years long data (1997-1999) were used by the forecasting models for training, testing and evaluation purposes. The results show satisfactory performance of the artificial neural network models with an index of agreement (d) ranges from 0.59 to 0.76.

Tzima, et al., (2007) developed a multi-algorithm data-driven AQ forecasting system that was actually used in Thessaloniki, Greece for supporting information provision to citizens in the frame of the www.airthess.gr project (which ceased operation in 2011).

Sousa, et al., (2007) used a new methodology based on feedforward artificial neural networks which use principal components as inputs, to forecast the hourly O_3 concentrations of the next day. The developed model was compared with multiple linear regression models, feedforward artificial neural network models based on the original data and also with principal component regression models. Results showed that the use of principal components as inputs did not improve the models prediction. The performance of the feedforward artificial neural networks based on the original data were the same when principal components were used as inputs (index of agreement d=0.84 and coefficient of determination R^2=0.61). The feedforward artificial neural networks provide better forecasting performance, in comparison to the multiple linear regression model (d=0.81 and R^2=0.49) and to the principal component regression model (d=0.82 and R^2=0.61).

Coman, et al., (2008) attempted to verify the presence of non-linear dynamics in the ozone time series by testing a "dynamic" model, evaluated in comparison to a "static" one, in the context of predicting hourly O_3 concentrations (one-day ahead), in Paris, France. The "dynamic" model uses a recursive structure involving a cascade of 24 multilayer perceptron (MLP) neural networks arranged so that each MLP feeds the next one. The "static" model is a classical single multilayer perceptron neural network. Three years long data (2000-2002) were used by the neural networks. Results indicate a rather good applicability of these models for a short-term prediction of O_3, with an index of agreement (d^2) ranges from 0.83 to 0.98 and a coefficient of determination ranges from 0.53 to 0.92. Coman, et al., (2008) conclude that for their study data, there was no evidence of nonlinear dynamics in the O_3 because the results of the recursive model were similar with those obtained via the "static" model.

Kurt, et al., (2008) developed an online air pollution forecasting system for Istanbul, Turkey. The system predicts three air pollution indicator (SO_2, PM_{10} and CO) levels for three days ahead by using artificial neural networks. The experiments presented by Kurt, et al., (2008) were conducted by using almost one year long data (August 2005 to July 2006). Results show that quite accurate predictions of air pollutant indicator levels are possible with a simple neural network (with relative error ranges from 20% to 5%).

Solaiman, et al., (2008) investigated three CI methods to address the complex non-linear relationships between O_3 and meteorological variables in Hamilton, Canada. Three dynamic neural networks with different structures were investigated: 1) a time-lagged feedforward network, 2) a recurrent neural network, and 3) a Bayesian neural network model. Four years long data (2001-2004) were used by the forecasting methods. Results suggest that these models are effective forecasting tools and outperform the commonly used multilayer perceptron and hence can be applicable for short-term forecasting of O_3 level. Although the overall performances of these models were satisfactory (with a coefficient of determination of 0.85), none of them was able to predict with accuracy the low and extremely high levels of concentrations.

Neto, et al., (2009) applied statistical models based on multiple regression analysis and classification and regression trees analysis, in order to forecast the average daily concentrations of particulate matter and the maximum daily ozone levels. Four years long data (2000-2003) were used by the forecasting models (for training and validation), in Lisbon, Portugal. Results show the statistical models were very successful in forecasting the average daily concentrations of particulate matter (coefficient of determination ranges from 0.93 to 0.97) and the maximum daily O_3 levels (coefficient of determination ranges from 0.97 to 0.99).

Li, (2010) used artificial neural networks to forecast hourly concentrations of O_3, PM_{10} and SO_2 in Yinchuan of Ningxia Hui, China. Four years long data (2005-2008) were used by the forecasting method. The artificial neural networks produced good results (low mean square error) in the middle and long term forecasting of almost all the pollutants. Li, (2010) conclude that the proposed methodology appears to be very useful for local administrations and health and environmental protection institutions, which are usually more interested in catching the future pollutant trend rather than the precise concentration value.

Moustris, et al., (2010) used artificial neural networks in order to forecast the maximum daily value (three days ahead) of the Regional Pollution Index based on the European thresholds of NO_2, CO, SO_2 and O_3. Five years long data (2001–2005) were used by the forecasting method, in Athens, Greece. Results of forecasting three days in advance, show a very good agreement between prediction and observation values, with an index of agreement from 0.59 to 0.94 and a coefficient of determination from 0.19 to 0.83.

Roohollah, et al., (2010) developed proper prediction models by using artificial neural networks and adaptive neuro-fuzzy inference system (ANFIS) models, to forecast daily carbon monoxide (CO) concentrations in Tehran, Iran. Since input selection is a significant step in statistical models, forward selection (FS) and gamma test (GT) methods were used for selecting input variables and developing hybrid models with ANN and ANFIS. A total of six models (ANN, FS-ANN, GT-ANN, ANFIS, FS-ANFIS and GT-ANFIS) were developed by using two years long data (2004-2005). FS-ANN and FS-ANFIS models were selected as the best models considering their coefficient of determination of 0.9 and 0.91, and index of agreement (d) 0.97 and 0.97 respectively. Finally, uncertainty analysis based on Monte-Carlo simulation was carried out for FS-ANN and FS-ANFIS models which show that FS-ANN model has less uncertainty. Thus, FS-ANN was the best model which forecasts satisfactorily the trends in daily CO concentration levels.

Moustris, et al., (2012) examine the one-day forecast of the daily maximum surface O_3 concentration in Athens, Greece, by developing predictive models based on the method of multiple regression analysis (MLR) against artificial neural network (ANN) approach. Five years long data (2001-2005) were used by the forecasting models, which indicated that ANN models forecasting ability (with index of agreement of 0.892 and coefficient of determination of 0.666) does present a limited precedence against MLR models (with index of agreement of 0.887 and coefficient of determination of 0.653). Moustris, et al., (2012) states that in order to improve the predictive ability of the constructed ANN model, additional input parameters could be used, such as the nitrogen oxides concentrations, the intensity of solar radiation, and the sunlight duration.

Russo, et al., (2013) used stochastic data analysis to discover a set of stochastic variables that represent the relevant information on a multivariate stochastic system. These variables were used as input to artificial neural network models to forecast hourly NO_2 concentrations in Lisbon, Portugal. Five years long data (2002-2006) were used by the forecasting method with a coefficient of determination ranging from 0.15 to 0.85. Results show that by using stochastic variables as input data for training the ANN model, it is possible to significantly reduce the number of input variables and preserve the predictive power of the model.

Azid, et al., (2014) used principal component analysis to identify the sources of pollution in Peninsular, Malaysia. The identified sources were used as inputs to artificial neural networks to determine the predictive ability of an air pollutant index set up by the Malaysian Department of Environment to recover and maintain air quality and protect public health. The combination of principal component analysis and artificial neural networks showed better predictive ability in the determination of air pollutant index, when fewer variables were used (following feature selection) with a coefficient of determination of 0.62.

Elangasinghe, et al., (2014) proposed a methodology to extract the key information from routinely available data (meteorological and air pollutant variables) to build a reliable artificial neural network model. The methodology was tested by forecasting NO_2 concentrations using two years long data (2010-2011) in Auckland, New Zealand. Three input optimization techniques were explored, namely a genetic algorithm, forward selection, and backward elimination. The genetic algorithm technique gave predictions resulting in the smallest mean absolute error. The developed artificial neural network model was found to outperform a linear regression model based on the same input parameters. The index of agreement (d_r) of developed artificial neural network models ranges from 0.65 to 0.79 and the coefficient of determination ranges from 0.48 to 0.80. In the case of the linear regression models the range values of the respective measures were 0.53 to 0.70, and 0.17 to 0.60.

Mishra & Goyal, (2015) used a novel approach based on regression models and principal component analysis to find the correlations of different predictor variables between meteorology and air pollutants. The significant variables were used as the input parameters to the artificial neural network models to forecast hourly NO_2 concentrations in Agra, India. In addition, multiple linear regression models were used for comparison. The forecasting models use a one year long data (2013). Results show that the combination of principal component analysis and artificial neural networks provide better forecasting performance (with a coefficient of determination of 0.83) in comparison to the multiple linear regression models (coefficient of determination of 0.48).

Feng, et al., (2015) used a novel hybrid model by combining air mass trajectory analysis and wavelet transformation to improve the artificial neural network forecast accuracy of daily average concentrations of $PM_{2.5}$ (two days in advance). The trajectory based geographic parameter was used as an extra input predictor to the artificial neural network model, in order to recognize distinct corridors for transport of "dirty" and "clean" air to selected stations. The forecasting models used one year long data (September 2013 to October 2014) from Beijing, Tianjin and Hebei, China. It was found that the trajectory based geographic model and wavelet transformation can be effective tools to improve the $PM_{2.5}$ forecasting accuracy (with an index of agreement (d) from 0.90 to 0.98).

Prasad, et al., (2016) developed an adaptive neuro-fuzzy inference system (ANFIS) to forecast daily air pollution concentrations of five air pollutants - sulphur dioxide, nitrogen dioxide, carbon monoxide, ozone and particular matters (PM_{10}) - in Howrah, India. The forecasting models used three years long data (2009-2011). Collinearity tests were conducted to eliminate the redundant input variables. In addition, a forward selection method was used for selecting different subsets of input variables. Results show that forward selection techniques reduce not only the output error but also computational cost due to less number of inputs. In the forecasting models developed for the pollutants CO, SO_2, PM_{10}, and Ozone, the index of agreement ranges from 0.75 to 0.95, indicating a good model, and closer to the ideal value, whereas for NO_2 the index of agreement was around 0.60 indicating a satisfactory forecasting.

Durao, et al., (2016) developed a two-step methodology to forecast hourly ozone concentration levels (1 to 24 hours ahead) for Sines, Portugal. Firstly, classification and regression trees (CART) techniques were used to identify the best O_3 concentration predictors. In the second step, multilayer perceptron models were adopted to forecast O_3 levels by using five years long data (2006-2009). The CART analysis results revealed that O_3 concentrations are mainly dependent on meteorological variables, industrial emissions and air quality variables. The obtained generalisation model performances are very good to classify in advance the expected class of O_3 concentration level. Performance results to forecast O_3 level above a specific threshold vary from 70% of success (to forecast the next 24 hour in advance) up to 99% (to forecast the next hour in advance).

Bai, et al., (2016) developed a wavelet technique and back propagation neural networks (W-BPNN) to forecast daily air pollutants (PM_{10}, SO_2, and NO_2) concentrations in Chongqing, China. The proposed method was compared with the simple back propagation neural networks model. Firstly, stationary wavelet transform (SWT) was applied to decompose historical time series of daily air pollutants concentrations into different scales, of which the information represents wavelet coefficients of air pollutant concentration. Secondly, the wavelet coefficients were used to train a back propagation neural network model at each scale. The input data (one year long, 2011) contain the wavelet coefficients of the air pollutants concentrations 1-day in advance and local meteorological data. Results show that the proposed approach exhibits a better forecasting performance (with a coefficient of determination ranges from 0.86 to 0.95) in comparison to the simple back propagation neural networks (with a coefficient of determination ranges from 0.60 to 0.86).

Biancofiore, et al., (2017) used a multiple linear regression model and a neural network model, with and without recursive architecture, to forecast daily averaged particulate matters ($PM_{2.5}$ and PM_{10}) in Pescara, Italy. Results show that the neural network with recursive architecture has better performances (with a coefficient of determination ranges from 0.69 to 0.83) compared to both the multiple linear regression model (with a coefficient of determination ranges from 0.46 to 0.81) and the neural network model without the recursive architecture (with a coefficient of determination ranges from 0.64 to 0.72). Biancofiore, et al., (2017) concluded that the recursive artificial neural network model could be a powerful operational tool to obtain real time information on PM_{10} and $PM_{2.5}$ and support stakeholders in the development of cost effective control strategies to alert and protect the population.

In the studies quoted, it is clear that researchers use different inputs variables and different combinations of methods (in the KDD steps) to report the best combinations of methods (in terms of the selected measures). In addition, researchers use different measures, such as the coefficient of determination, different versions of the index of agreement, mean square error, etc. This makes it difficult to compare the error of different studies and different forecasting parameters. The main progress of the AQ forecasting by using data-driven (in the recent years) is the use of different or/and new methods of the KDD steps (performed in different locations) and in some cases optimize the parameters of the forecasting methods, but not the optimization of the whole process.

2.3.2 Operational AQ Forecasting Systems

Numerous countries have developed air quality forecasting systems to forecast the concentrations of criteria pollutants (i.e. pollutants with special health concerns, see Section 2.1.2). In many cases, the spatial scale of the forecasting system differs but it is common to start with a mesoscale model implementation before focusing on an urban scale as discussed by Moussiopoulos, et al., (2003). Forecasts can be used to issue early air quality alerts that allow government and people to take precautionary measures (such as temporarily shutting off major emission sources, motivating car-pooling and the use of public transportation) to reduce air pollution and to avoid or limit the exposures to unhealthy levels (Zhang, et al., 2012). The air quality conditions and forecasts can be disseminated to the public via the internet or other media.

The U.S. Environmental Protection Agency (EPA) developed in 1997 the AIRNow platform, in order to communicate real-time air quality conditions and forecasts to the public (https://airnow.gov/) (Zhang, et al., 2012) (CENR, 2001). The AIRNow platform receives real-time O_3 and PM pollution data from more than 115 U.S. and Canadian agencies as well as real-time air quality forecasts (RT-AQFs) from about 400 U.S. cities and represents a centralised, nationwide, governmental repository for real-time data (Zhang, et al., 2012). The U.S. EPA use an Air Quality Index (AQI) to include a simple colour scheme to link air quality concentrations and associated health effects to a simple colour coded index that can be easily and consistently reported to the public (Zhang, et al., 2012). AIRNow has been expanded to include real-time data from other countries such as China (Zhang, et al., 2012).

In 2004 the National Oceanic and Atmospheric Administration (NOAA) and the U.S. EPA developed a national air quality forecasting system that is based on numerical models for meteorology, emissions, and chemistry (Otte, et al., 2005) (CENR, 2001). The air quality forecasting generates gridded model forecasts of ground-level ozone (O_3) that can alert the public of the onset, severity, and duration of poor air quality conditions (https://ready.arl.noaa.gov/) (Otte, et al., 2005). Currently (accessed in March 2018), the website informs the visitors that it is not maintained in an "operational" environment and should not be relied upon for 24/7 access.

The CALIOPE Air Quality Forecast System (CALIOPE-AQFS) provides a 48-hour forecast of NO_2, O_3, SO_2, PM_{10}, $PM_{2.5}$, CO, and benzene (C_6H_6) at a 4 km horizontal resolution over all of Spain, and at a 1 km horizontal resolution over the most populated areas of Spain with complex terrains (Pay, et al., 2014). The CALIOPE system provides forecasts to the public via their website (http://www.bsc.es/caliope) or via apps for mobile devices and tablets.

The air quality agency of the Paris region in France (AIRPARIF), provides real-time forecasts available on its website (https://www.airparif.asso.fr). A near-real-time observation database (named as BASTER) gathers all hourly measurements by the local air quality networks every three hours in France, Germany, Italy, Finland, Austria, and the U.K (Zhang, et al., 2012). The AIRPARIF system uses also the Common Air Quality Index (CAQI), which is suggested by the European Environment Agency (EEA). The CAQI has five categorical levels from very low to very high, in order to be easily interpreted by the public (see Appendix IV).

Prev’Air is the French operational air quality forecasting and monitoring system (Debry & Mallet, 2014) (Honoré, et al., 2008). It became operational in 2003 from an initiative of the French Ministry of Environment. Prev’Air system aims to inform the public and professionals (via the web site http://www2.prevair.org/) about the forecasted pollutant concentrations. On a daily basis, it provides forecasts up to three days ahead for O_3, NO_2 and particulate matter (PM_{10} and $PM_{2.5}$) over France and Europe.

The Office of Environment and Heritage (OEH) is responsible for providing information to local communities on air quality as a priority action in the New South Wales (NSW) Government’s strategic plan NSW 2021 (Jiang, et al., 2015). The OEH currently provides daily air quality forecasts for Sydney, Australia. The forecasts are issued as Air Quality Index (AQI) on their website (http://www.environment.nsw.gov.au/AQMS/aqi.htm). The existing system has limited capability for forecasting pollution events. Hence, OEH plans to progressively advance its capability for forecasting air quality by implementing a multi-level architecture approach (which involve tools and systems of different types) within the Greater Metropolitan Region and key regional areas in NSW (Jiang, et al., 2015).

The Department for Environment, Food and Rural Affairs (DEFRA) in the UK, operate the official air quality forecast system for public information. The UK-AIR (Air Information Resource) web site (https://uk-air.defra.gov.uk/) provides information for the latest pollution levels, pollution forecast information, a data archive, and details of the various monitoring networks. The forecasts use the Daily Air Quality Index (DAQI) that is designed to show complex air quality information on a simple 10 point scale. The DAQI uses a combination of numbers (1-10), words (low, moderate, high and very high) and colours (green/yellow/orange/red/purple) to communicate the levels of pollution expected. Forecasts are produced early in the morning for the current day as well as for the next 4 days.

The National Centre for Atmospheric Science (NCAS) Air Quality Forecast (AQF) system is aimed at the scientific community interested in the prediction of air quality related pollutant species over the UK. The NCAS AQF system is operated by the Centre of Atmospheric and Instrumentation Research, University of Hertfordshire. The AQF system employs an integrated modelling framework for producing operational air quality forecasts based on the Weather Research and Forecasting (WRF) model and the Community Multiscale Air Quality (CMAQ) (Wong, et al., 2012). Forecasts are daily (up to three days) for Europe and UK domains at a grid resolution of 50km and 10km respectively (https://sci.ncas.ac.uk/airquality/).

Athens, Greece, has developed an air quality forecasting system (CAMx-AMWFG) for the Mediterranean Region. The CAMx-AMWFG system produces 48-hour operational forecasts of O_3, NO_2, SO_2, and particulate sulphate (PSO_4) fields for the Mediterranean region and Europe. The 48-hour forecasts (i.e. 16 hourly forecasts, with a 3 hours interval) are produced once per day (http://forecast.uoa.gr/camxindx.php). Additionally, CAMx-AMWFG can provide the concentration and deposition of sodium and chloride (from sea-salt production), sulphate produced on dust (DSO_4), and nitrate produced on dust (DNO_3).

In the past, additional operational systems were available in Greece for the cities of Athens and Thessaloniki. The PASODOBLE (Promote Air Quality Services integrating Observations - Development of Basic Localised Information for Europe) which provided three-day air quality forecasts of O_3, NO_2, NO, CO, SO_2, $PM_{2.5}$ and PM_{10} concentrations (http://lap.physics.auth.gr/pasodoble.asp). The forecasts were performed at two spatial scales: 1) Regional scale (Balkan) and 2) Urban scale (Athens and Thessaloniki). The

AIRTHESS was an operational air quality information and early warning system (until 2011), which was developed for the city of Thessaloniki, Greece. Computational intelligence methods were used for the forecasting of concentration levels. The information was presented, either on a daily basis or on the basis of alerts on forecasted incidents (for example via SMS).

2.3.3 Discussion

After reviewing the academic literature relating to the use of data-driven models for air pollutants forecasting, it is clear that researchers use different inputs variables, different feature selection and transformation methods (e.g. principal component analysis, wavelet transformation, etc.), different data-driven models (e.g. multiple linear regression models, several types of artificial neural networks, classification and regression trees, etc.), in order to achieve the desired forecast. It is important to investigate different data-driven models, because, for example, in some cases simple data-driven models can provide better forecasting performance than more complex models. In the studies quoted, several combinations of methods are used in each and, for that reason, their forecasting performances have been presented as ranges of values and the best combinations of the methods (in terms of the selected measures) are reported. Even though the pre-processing and feature selection and transformation methods can influence the forecasting accuracy of the data-driven models these aspects are not usually reported in the literature.

The most common forecasted pollutants in the studied literature were: O_3, NO_2, SO_2, CO, PM_{10}, $PM_{2.5}$, and some air pollutant index. This was not a surprise since these pollutants are among the regulated major pollutants by the U.S. EPA and by the EU. However, some regulated major pollutants (lead and benzene) were not very common in the literature. This is understandable in the case of lead because following the introduction of unleaded fuels in many countries, the amount of lead in the air has decreased.

Some authors of the studied literature performed statistical analysis by taking into account several forecasting performance measures. The most common reported measures were the coefficient of determination and a version of the index of agreement. In the case of the index of agreement, researchers used one of the three different definitions (d, d_2 and d_r) proposed by Willmott (Willmott, 1981) (Willmott, et al., 1985) (Willmott, et al., 2011). This is a problem because the results of these studies are not comparable. These measures can be used in an automated operational forecasting system, which will evaluate data-driven models to find the models with the highest forecasting performance (best models). But for that purpose, a procedure to combine the measures should be developed. A small number of authors report the forecasting performance by calculating the difference between the observed and forecasted values (i.e. calculating measures such as the mean square error and the mean average error). The problem with these measures is the relative size of the error with the forecasting parameter. Thus, it is difficult to distinguish a big error from a small error and the error is not comparable with the error of a different forecasting parameter. In addition, in the majority of the studies, a resampling method for model validation (such as cross-validation or bootstrapping) was not used. These methods can measure the predictive performance of the data-driven models and thus, are essential to test and evaluate the data-driven models. All of these problems (corresponding to the second research question, in Section 1.2) were addressed by developing two new forecasting performance indices which combine the characteristics of existing measures and do not suffer from the same degree of inconsistencies and disadvantages.

All reviewed studies contribute to the forecasting of their selected pollutants, locations and data. However, if one of those parameters changes, the investigation to find the best model must be repeated. This highlights the need for a methodology to optimise the selection of data pre-processing methods and data-driven models, which find the models with the highest forecasting accuracy (best models), which reflects the third research question (see Section 1.2).

Numerous operational air quality forecasting systems are presented, such as the AIRNow platform, the CALIOPE system, the AIRPARIF system, etc. The presented air quality forecasting systems provide forecasts either for important air pollutants or for an air quality index. The latter is important in order to be easily interpreted by the public. Even though deterministic models are commonly used in the studied operational air quality forecasting systems, it usually requires several years of implementation. Jiang, et al., (2015) proposed a multi-level architecture approach for an air quality forecasting system by involving tools and systems of different types (empirical, data-driven and deterministic). In their proposal, the data-driven models will be developed to provide daily forecasts for an air quality index and/or for pollutants concentration of specific sub-regions or monitoring sites. Data-driven models can be easily implemented, require less implementation time (than deterministic models) and thus are ideal to provide alerts (for the public and authorities) for specific locations, especially to those that do not have an operational air quality forecasting system. For example, in the case of Greece, the CAMx-AMWFG (deterministic) system produces 48-hour operational forecasts of several pollutants, but the information is difficult to be interpreted by the public.

2.4 The Study Areas

The research documented in this thesis makes use of data from two separate geographical areas of Greece, these being Athens and Thessaloniki.

Athens is the capital of Greece, located in a basin of approximately 430 km^2 (Figure 2-3) and surrounded by fairly high mountains (Mt. Parnis, Mt. Pendeli, Mt. Hymettus). The area is open to the sea from the S-SW and affected by sea-land breeze circulations. Industrial activities take place both in the Athens basin and in the neighbouring Mesogia plain, the latter hosting the international airport Eleftherios Velizelos. The Athens basin is characterised by a high concentration of population (about 50% of the Greek population, i.e. approximately 5 million people), accumulation of industry (about 50% of the Greek industrial activities), and high usage of cars: more than two million cars are congested in the roads of Athens each day. Anthropogenic emissions in conjunction with unfavourable topographical and meteorological conditions are responsible for the high air pollution levels in the area (Karatzas, et al., 2008). Figure 2-3 indicates the two air quality monitoring stations in Athens (at Liosia – a suburb area and Patisia – a city centre area) that were used in this research.

Thessaloniki is the second largest city in Greece and the capital of the administrative Region of Central Macedonia. The greater Thessaloniki area (GTA) has a population of over a million people with more than 400,000 vehicles being used daily. The city is characterized by high air pollution levels, especially in terms of PM_{10}, (Voukantsis, et al., 2011). AQ data (pollutant concentrations) as well as meteorological data resulted from the operation of four monitoring stations (Figure 2-4) which are situated in the areas-locations described next.

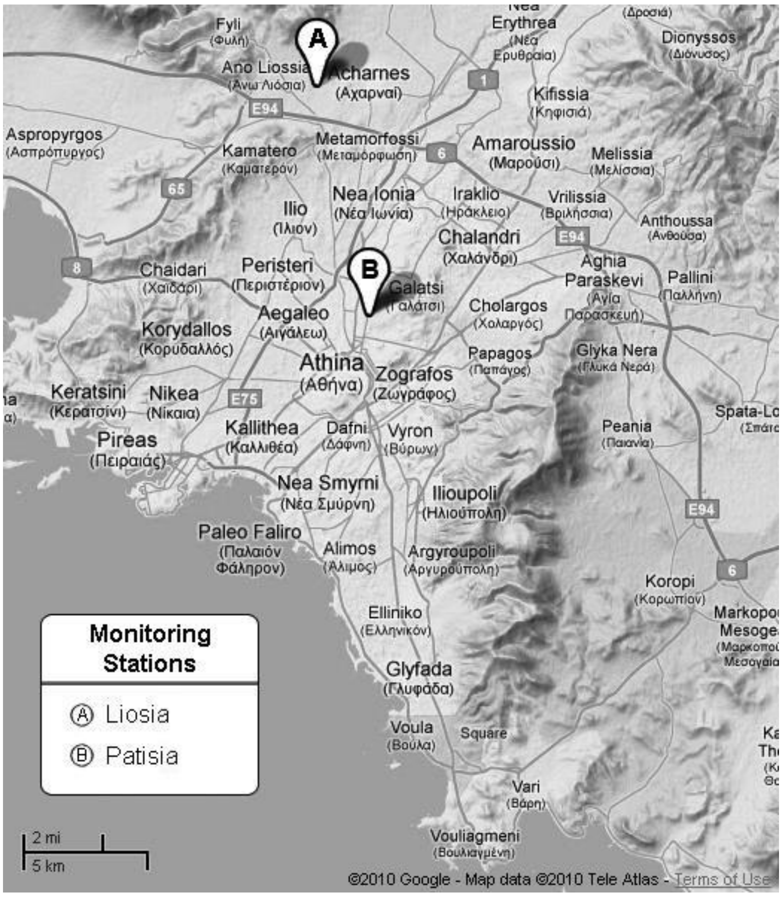


Figure 2-3: The two air quality monitoring stations in Athens at Patisia and Liosia
A) Liosia (a suburb area of the city, mostly used as a residential and industrial area). B) Patisia (city centre area, characterized by traffic and a mixture of household and commercial activities).

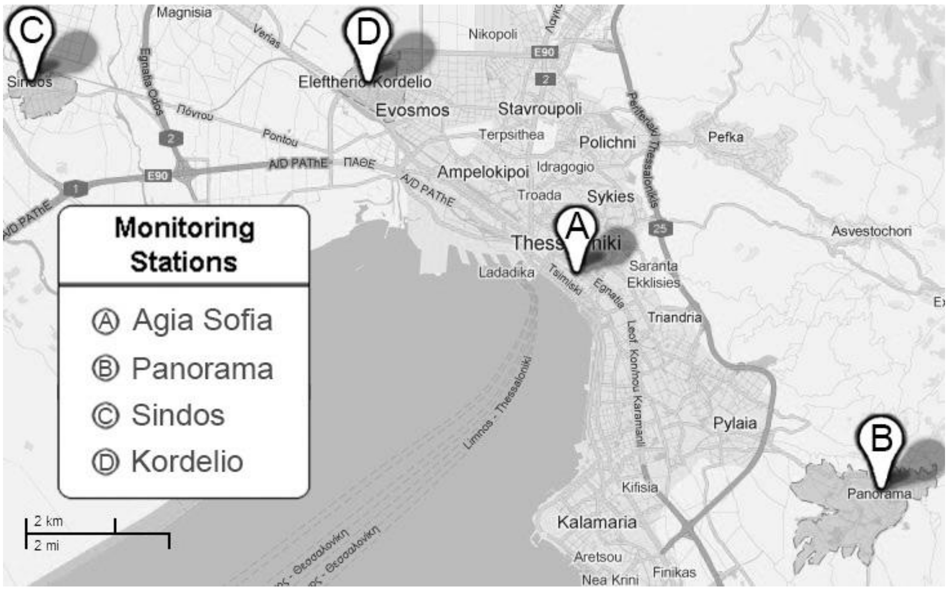


Figure 2-4: The four air quality monitoring stations in greater Thessaloniki area at Agia Sofia, Panorama, Sindos, and Kordelio
A) Agia Sofia (city centre area, characterized by traffic and a mixture of household and commercial activities). B) Panorama (a suburb in the hilly area of the city, mostly used as a residential area). C) Sindos (industrial-suburban area in the west of the city influenced by traffic). D) Kordelio (a densely populated area close to the industrial area of the city, influenced by traffic).

2.5 Employed Methods

The following sections describe the methods employed in the experimental aspects of this research (such as the forecasting methods used in investigation experiments, sensitivity analysis, fuzzy logic, etc.).

2.5.1 Covariance

Covariance is a measure of the strength of how two random variables (X and Y) vary together (i.e. the correlation between these variables) (Weisstein, 2017). A positive covariance would indicate that the variables are moving in the same direction (i.e. a positive linear relationship between the variables), while a negative covariance would indicate that the variables are moving in the opposite direction (i.e. a negative linear relationship between the variables). If there is no correlation between the variables, then covariance would be zero. The covariance of two random variables (X and Y) is computed be using Equation (2-1).

$$Covariance(x,y) = \sum_{i=1}^{N} \frac{(x_i - \bar{x})(y_i - \bar{y})}{N} \tag{2-1}$$

Where:

$\bar{x}$ is the mean of the random variable x

$\bar{y}$ is the mean of the random variable y

2.5.2 Lagrange Interpolating Polynomial

The polynomial interpolation problem is the problem of constructing a polynomial that passes through or interpolates (n+1) data points. The Lagrange interpolating polynomial is the polynomial P(x) of degree <=(n-1) that passes through the n points (Archer & Weisstein, 2017) and is given by Equation (2-2).

$$P(x) = \sum_{j=1}^{n} P_j(x) \tag{2-2}$$

Where

$$P_j(x) = y_j \prod_{\substack{k=1 \\ k \neq j}}^{n} \frac{x - x_k}{x_j - x_k} \tag{2-3}$$

2.5.3 Periodicity Analysis

Urban air quality systems are influenced by natural (usually chaotic) as well as anthropogenic signals (generally related to city activities resulting in emissions), and thus they may include periodic components, that influence their behaviour. For this reason, it is important to investigate if there is any periodicity in the studied system. In order to perform periodicity estimation, the Fast Fourier Transformation (FFT) (Rader & Brenner, 1976) (Karatzas, et al., 2009) was utilised. The discrete Fourier transformation of a stationary discrete time series x(n) is a set of N- discrete harmonics X(k) where:

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N} \tag{2-4}$$

and it represents the frequency response of X(k), where X(k) is complex. The magnitude of X(k) squared (a real value), is called strength, and a diagram of all the strength harmonics is called periodogram. When x(n) is real then the periodogram is symmetric, and only half of the harmonics are needed. To transform the data from non-stationary to stationary, the moving average must be subtracted from the time series.

2.5.4 Forecasting Methods used in Investigation Experiments

The forecasting methods presented in this section are used to investigate the use of computational intelligence and statistical methods to perform forecasting of environmental parameters of interest. These methods have been selected because they have been used previously in related studies and in order to use methods with different characteristics in terms of their algorithms.

Multilayer perceptron ANNs were utilised using the backpropagation training algorithm, due to its successful performance in similar air quality related applications (Voukantsis, et al., 2010). This algorithm was implemented using the MATLAB Neural Network Toolbox. Data to be imported to the ANN were firstly normalised, by applying the hyperbolic tangent sigmoid transfer function (see Appendix V), which was also applied to the hidden layers (receives values between -1 and 1). In the output layer, the linear transfer function (see Appendix V) was used, which also receives values between -1 and 1. This is a common structure for function approximation (or regression) problems (Mathworks, 2008), but has also shown good results in similar studies (Slini, et al., 2003). For the training phase, the Levenberg-Marquardt Backpropagation algorithm (Grivas & Chaloulakou, 2006) was implemented. An additional transfer function exists, the log sigmoid transfer function (Mathworks, 2008) which was not used due to its limited range of values (between 0 and 1). The hyperbolic tangent sigmoid transfer function gives an equal weight to the high and low (normalized) values.

A Decision Tree (DT) is a hierarchical data structure implementing the divide-and-conquer strategy (Ethem, 2010). It is an efficient nonparametric method, which can be used for both classification and regression. DTs are essentially a map of the reasoning process in which a tree-like graph is constructed to explore options and investigate the possible outcomes of choosing the options. The reasoning process starts from a root node, transverses along the branches tagged with decision nodes and terminates in a leaf node on the basis of criteria such as Information Gain (Lim & Jain, 2009). In the investigation experiments an ensemble of DTs were used, by employing the Bootstrap aggregation (bagging) meta-algorithm, in order to improve the performance of the models (Breiman, 1996). An ensemble aims at leveraging the performance of a set of models to achieve better prediction accuracy than that of the individual models. For this purpose, every model (tree) in the ensemble is developed by using an independently drawn bootstrap replica of the input data. This means that from the initial training set (consisting of *n* observations), a number of *m* new training sets (replicas) is created by randomly sampling the initial training set with replacement (this is called "bagging"). As a result, some observations are sampled (and thus included in the replicas) more than once, and others may not be selected at all. The rows that were not included in the replica are called the “out of the bag” data. In the investigation experiments, a DT model is developed on the basis of each one of the replicas, and the overall prediction performance is estimated on the basis of an ensemble of DTs.

For comparison and analysis of the two CI methods above, Linear Regression (LR) models were developed. LR models may take into account the persistence of the air quality system, and have been proven successful especially if their input data includes lagged values, i.e., values determined before the time at which the forecast is made (Voukantsis, et al., 2011).

2.5.5 Validation Procedure

The model validation-evaluation procedure follows the Cross-Validation (CV) approach. This is a popular method applied in order to measure the predictive performance of a data-driven model (Refaeilzadeh, et al., 2009). In CV, the available dataset is divided into two segments, one is used to teach or train a model, and the other is used to validate the performance of the model. There are several types of cross-validation methods like the Holdout method, k-fold cross-validation, and leave-one-out cross-validation (Arlot & Celisse, 2010).

K-fold CV divides the dataset into k (equal) subsets. Each time, one of the k subsets is used as the test set (i.e. the set against which the model performance is estimated), and the other k-1 subsets are used as a training set. The k results from the folds (k times) are averaged for producing a single estimation. In this way, an ensemble of models is actually produced. For example, in the case of ANNs, although the same ANN architecture is employed for each subset, its weights are calculated separately each time. The advantage of this method is that the results are not influenced by the way the data are divided.

In this work the 10-fold cross-validation is used because it is the one most commonly applied (Kohavi, 1995) (Borra & Di Ciaccio, 2010), and proved to provide with better results than other similar techniques like bootstrapping (Kohavi, 1995) (Braga-Neto & Dougherty, 2004) (Borra & Di Ciaccio, 2010).

Figure 2-5 shows how the data was divided and used by the forecasting methods. The 10-fold CV is used in order to:

- Measure the predictive performance of the models being developed, and
- Compute the CIs of the selected measures to be studied.

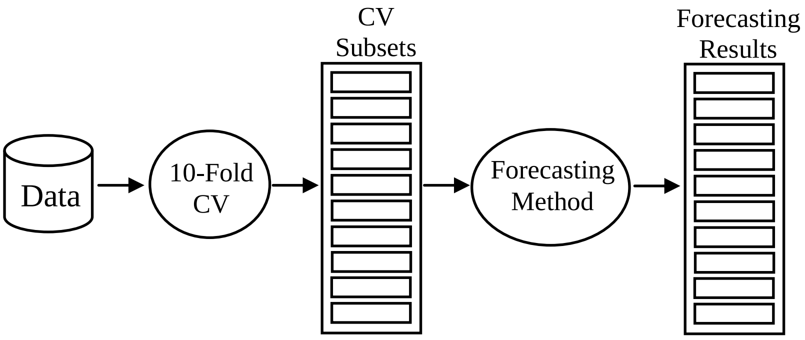


Figure 2-5: How the data was divided and used by the forecasting methods

2.5.6 Sensitivity Analysis

The model predictions can be highly dependent upon model characteristics, as well as uncertainties in the input data. Such model behaviour can be investigated through sensitivity analysis, which seeks to determine the variation in model output as a function of variations in input variables and parameters (i.e., forward sensitivity analysis). Sensitivity analysis could provide information on the input factors that are mainly responsible for the output uncertainties and thus for the model performance (Kukkonen, et al., 2012). There are generally two types of sensitivity analysis: statistical sensitivity analysis and deterministic sensitivity analysis. In statistical sensitivity analysis (Kukkonen, et al., 2012), the model is executed several times, each time with slightly different inputs (or input parameters) and the sensitivity is estimated from the statistical properties of the multiple output variability. In deterministic sensitivity analysis (Kukkonen, et al., 2012), the model output equations are differentiated with respect to its inputs, and the sensitivity is calculated for each different input through an auxiliary set of equations (depending on the uncertainty estimation of the parameters of interest).

The simplest statistical approach of (forward) sensitivity analysis is to vary one input or parameter value in the model by a given amount and examine the impact on the predictions. This is known as one-way (or one-at-a-time) sensitivity analysis since only one parameter is changed at one time. The analysis could be repeated by varying different parameters at different times.

2.5.7 Fuzzy Logic

2.5.7.1 Introduction

According to cognitive scientists humans base their thinking primarily on conceptual patterns and mental images rather than on any numerical quantities (Timothy, 2010). People in their everyday life use concepts such as “many”, “tall”, “much larger than”, “young”, etc. which are accurate only to some degree. These concepts (facts) can be called fuzzy or grey (vague) concepts. They form part of a human’s natural language, thus, human brains work with them (Sivanandam, et al., 2007).

Logic for humans is a way to quantitatively develop a reasoning process that can be replicated and handled with mathematical precepts. The interest in logic is the study of truth in logical propositions; in traditional logic this truth is binary (a proposition is either true or false) (Timothy, 2010) and this is the case for classic algebraic calculations (a result is either equal to x or not equal to x).

The traditional notion of set membership and logic has its origins in ancient Greek philosophy. The precision of mathematics owes its success in large part to Aristotle and the philosophers who preceded him. In their efforts to devise a concise theory of logic, and later mathematics, the so-called “Laws of Thought” were posited. One of these, the “Law of the Excluded Middle”, states that every proposition must either be True of False. Even when Parmenides proposed the first version of his law (around 400 B.C.), there were strong and immediate objections. For example, Heraclitus proposed that things could be simultaneously true and not true. It was Plato who laid the foundation for what would become fuzzy logic, indicating that there was a third region (beyond True and False) where these opposites “tumbled about” (Chennakesava, 2008).

The concept Fuzzy Logic was introduced (in 1965) by Lotfi A. Zadeh (a professor at the University of California at Berkley) as a mathematical tool for dealing with uncertainty. Fuzzy Logic is a way of processing data by allowing partial set membership rather than crisp events (events that either do or do not occur). Thus, Fuzzy Logic provides an inference structure that enables appropriate human reasoning capabilities.

Fuzzy logic is a fascinating area of research that involves a trade-off between significance and precision, something that humans have been managing for a very long time (Figure 2-6). Fuzzy logic is also a convenient way to map an input space to an output space (Chennakesava, 2008).

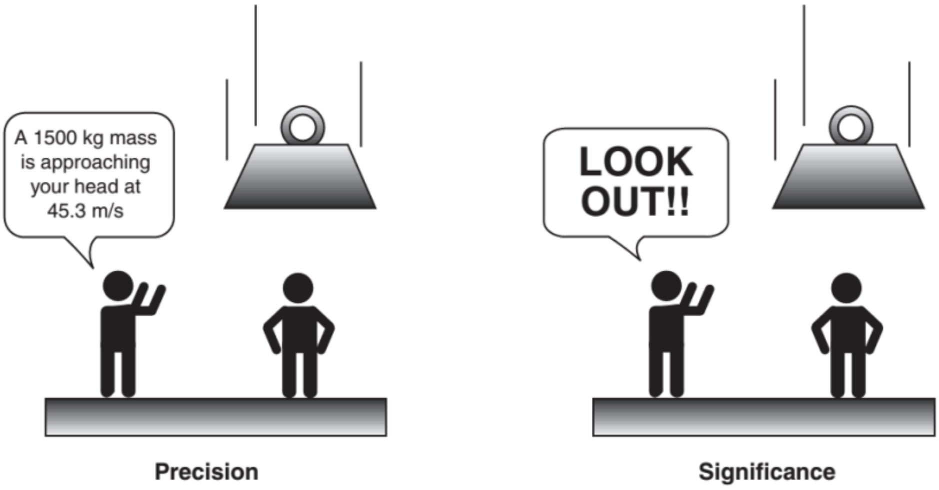


Figure 2-6: Precision and Significance in the real world (Chennakesava, 2008)

In the second part of the research documented in this thesis, Fuzzy logic was used in order to map the different forecasting performance indices (input space) into significance information about the forecasting performance (output space). Fuzzy Logic was selected as an appropriate method to compile the new statistical indices because:

a) It allows rapid prototyping (thus appropriate as in this research, there was no prior knowledge of the new indices is available);
b) The rules applied can be easily modified (thus supporting the study of different approaches to the construction of the new indices);
c) It can encompass great complexity;
d) It relates input to output in linguistic terms (i.e. terms easily understood by humans);
e) It has already been used in the past to describe air pollution (Karatzas, 2003) (Yadav, et al., 2011) which is the application domain of this research's forecasting models.

2.5.7.2 Fuzzy Set

A fuzzy set is a set containing elements that have varying degrees of membership of the set (Sivanandam, et al., 2007). Elements in a fuzzy set can also be members of other fuzzy sets in the same universe because their membership does not have to be complete. Fuzzy sets are denoted by a set symbol with a tilde under strike. Fuzzy sets are mapped to a real numbered value in the interval 0 to 1. If an element of universe, say x, is a member of the fuzzy set $\underset{\sim}{A}$ then the mapping is given by $\mu_{\underset{\sim}{A}}(x) \in [0,1]$.

2.5.7.3 Fuzzy Rules

The Achilles' heel of a fuzzy system is its rules, thus good rules will result in smart systems but unfortunately the number of rules increases exponentially with the dimension of the input space (number of system variables). This rule explosion is called the principle of dimensionality and is a general problem for mathematical models (Sivanandam, et al., 2007).

Fuzzy sets form the building blocks for fuzzy conditional rules which have the general form "IF X is A THEN Y is B," where A and B are fuzzy sets. The "IF" part of an implication is called the antecedent whereas the "THEN" part is a consequent, as can be seen in Equation (2-5). A fuzzy system is a set of fuzzy rules that converts inputs to outputs (Sivanandam, et al., 2007).

$$\text{IF premise (antecedent), THEN conclusion (consequent)} \qquad (2\text{-}5)$$

2.5.7.4 Fuzzy Inference System

Fuzzy inference systems (FISs) are systems that use fuzzy set theory to map inputs (features in the case of fuzzy classification) to outputs (classes in the case of fuzzy classification) (Cook, 2008). The FIS formulates suitable rules and based upon the rules the decision is made. This is mainly based on the concepts of the fuzzy set theory, fuzzy conditional rules, and fuzzy reasoning. FIS uses "IF...THEN..." statements, and the connectors present in the rule statement are "OR" or "AND" to make the necessary decision rules. The basic FIS can take either fuzzy inputs or crisp inputs, but the outputs it produces are almost always fuzzy sets (Sivanandam, et al., 2007). In the research documented in this thesis, the Fuzzy Logic Toolbox of MATLAB was uses in order to build the FISs. The most common types of FISs that have been introduced in the literature and applied to different applications are Mamdani and Sugeno type fuzzy models

(Alshalaa & Issmail, 2013) (Kansal & Kaur, 2013) (Mansi & Gajanan, 2013) (Kaur & Kaur, 2012).

Mamdani's FIS is the most commonly used fuzzy methodology. It was proposed by Mamdani and Assilian (1975) as an attempt to control a steam engine and boiler combination by synthesizing a set of linguistic control rules obtained from experienced human operators. In Mamdani's fuzzy inference method, the fuzzy sets from the consequent of each rule are combined through the aggregation operator and the resulting fuzzy set is defuzzified to yield the output of the system. Aggregation operations on fuzzy sets are operations that can combine several fuzzy sets to produce a single fuzzy set, by using the fuzzy union and intersection (most commonly used). A fuzzy system with two non-interactive inputs x_1 and x_2 (antecedents) and a single output y (consequent) is described by a collection of r linguistic IF–THEN propositions in the Mamdani form (Timothy, 2010), as shown in Equation (2-6). The aggregated output for the r rules will be given from Equation (2-7).

$$\text{IF } x_1 \text{ is } A_1^k \text{ and } x_2 \text{ is } A_2^k \text{ THEN } y^k \text{ is } B^k, \quad \text{for } k = 1, 2, \dots, r \tag{2-6}$$

Where:

A_1^k and A_2^k are the fuzzy sets representing the kth antecedent pairs and B^k is the fuzzy set representing the kth consequent.

$$\mu_{B^k}(y) = \max_k \left[\min\left[\mu_{A_1^k}\big(input(i)\big), \mu_{A_2^k}(input(j))\right]\right], \; k = 1, 2, \dots, r \tag{2-7}$$

Figure 2-7 shows the graphical analysis of two rules, where the symbols A_{11} and A_{12} refer to the first and second fuzzy antecedents of the first rule, and the symbol B_1 refers to the fuzzy consequent of the first rule. The symbols A_{21} and A_{22} refer to the first and second fuzzy antecedents of the second rule, and the symbol B_2 refers to the fuzzy consequent of the second rule.

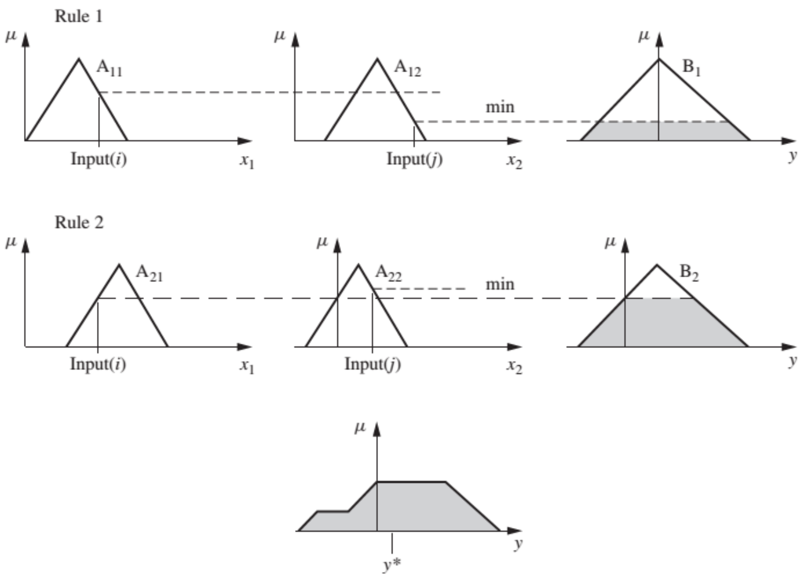


Figure 2-7: Graphical Mamdani (max–min) inference method with crisp inputs (Timothy, 2010)

The Sugeno or Takagi-Sugeno-Kang FIS was introduced by Sugeno (in 1985). The first two parts of the fuzzy inference process, fuzzifying the inputs and applying the fuzzy operator, are exactly the same with the Mamdani method, but it does not involve a defuzzification process. The output membership functions of the Sugeno FIS are either linear or constant. The consequent of each rule is a linear combination of the inputs, and the output is a weighted linear combination of the consequents. A typical rule in a Sugeno model, which has two inputs x and y and output z, has the form of Equation (2-8) (Timothy, 2010).

$$IF\ x\ is\ A\ and\ y\ is\ B, THEN\ z\ is\ z = f(x,y) \tag{2-8}$$

Where:

z = f(x, y) can be a constant or a linear function in the consequent.

In a Sugeno model, each rule has a crisp output, given by a function. Because of this the overall output is obtained via a weighted average defuzzification, as shown in Figure 2-8.

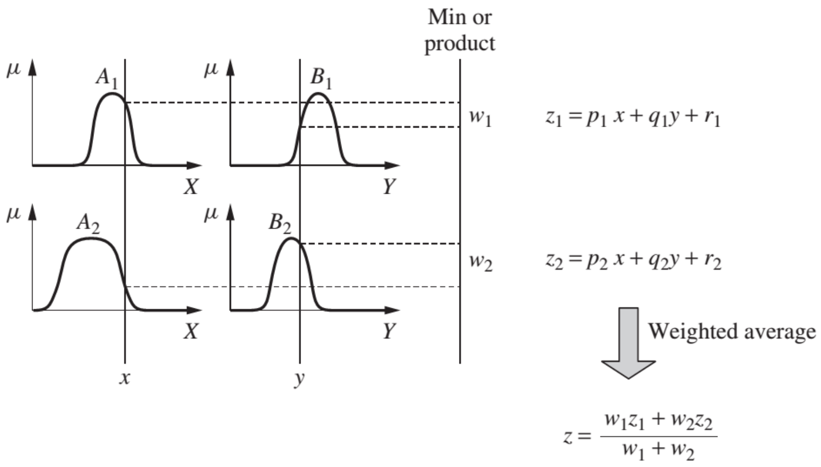

Min or product
μ
A_1
B_1
w_1
$z_1 = p_1 x + q_1 y + r_1$
X
Y
A_2
B_2
w_2
$z_2 = p_2 x + q_2 y + r_2$
Weighted average
x
y
$z = \frac{w_1 z_1 + w_2 z_2}{w_1 + w_2}$

Figure 2-8: A Sugeno fuzzy model (Timothy, 2010)

Below are listed the main characteristics and advantages of each type of fuzzy inference method (Sivanandam, et al., 2007). Whereas Mamdani model uses defuzzification process to yield the output, Sugeno's method uses a weighted linear combination of the consequents to compute the crisp output. For that reason, the Sugeno method is more compact, is computationally efficient and works better with optimization and adaptive techniques than the Mamdani method. Blej & Azizi (2016) show that the performance of Sugeno's method is better than the Mamdani method for the same fuzzy technique, however, this cannot be asserted in general. In the research documented in this thesis, both methods were used for comparison.

Advantages of the Sugeno Method:

- It is computationally efficient;
- It works well with linear techniques (e.g., Proportional-Integral-Derivative controller);
- It works well with optimization and adaptive techniques;
- It has guaranteed continuity of the output surface;
- It is well suited to mathematical analysis.

Advantages of the Mamdani Method:

- It is intuitive;
- It has widespread acceptance;
- It is well suited to human input.

2.5.8 Confidence Intervals

A confidence interval is a standard way of describing the precision of a measurement of some parameter. Confidence intervals provide a range of the observed effect size. This range is constructed in such a way in order to know how likely it is to capture the true, but unknown, effect size (Davies & Crombie, 2009). There are two common forms for stating confidence intervals both of which provide exactly the same information, for example:

- The measurement is 10 ± 3, with 95% confidence.
- The measurement will be between 7 and 13, with 95% confidence.

The confidence level shows the confidence percentage that the true value of a parameter lies between the upper and lower bounds. While confidence intervals computed using a smaller confidence level will be smaller than those computed with a larger confidence level, it is not possible directly to compare two confidence intervals made at differing confidence levels (Kaplan, 1999). The most commonly used confidence levels are 90%, 95% and 99% depending on the application of use. While 95% confidence level is arbitrary, it is traditionally used in applied practice (Rees, 1987) (Altman, et al., 2000). In the research documented in this thesis, a confidence level of 95% was used to calculate the CIs.

A resampling method like bootstrapping or cross-validation can be used to generate the sample used for computing the CIs. With the advent of bootstrapping and computers, it has become possible, and even easy, to compute confidence intervals on all sorts of statistics, even ones that are made for a particular purpose (Kaplan, 1999).

In the current research, cross-validation is used as a resampling method in order to compute CIs for the forecasting performance indices. In order to compute the CIs with 95% confidence level, the bounds (lower and upper) were set to 2.5% and 97.5% percentiles, as described by Kaplan (1999). Depending on the values of the upper and lower bounds of a CI applied in the evaluation procedure, a forecasting model can be characterized as:

1. Having a relatively high effectiveness (i.e. smaller distance between the CI bounds) to detect the variation of a parameter (in this work the value of the FPIs and thus the forecasting performance), or
2. Having a relatively low effectiveness (i.e. larger distance between the CI bounds). In that case, it can be considered as providing less information (in terms of forecasting ability).

2.6 Optimization Algorithms

2.6.1 Introduction to Optimization Problems

In optimization problems, the goal is to find the optimal (or best) solutions from all feasible solutions. The terminology "best" solution implies that there is more than one solution, and the solutions are not of equal value. The definition of best is relative to the problem at hand. Thus, the optimal solution depends on the problem's goal.

Optimization consists of trying variations on an initial concept-idea (input) and using the information gained to improve the output or result. A simple case of optimization problem consists of maximizing or minimizing a real function. A computer is a perfect tool for optimization as long as the idea or variable can be represented in electronic format (Haupt & Haupt, 2004).

The simplest optimization method is the exhaustive search in which all feasible solutions – in what is referred to as the search space - are tested in order to find the optimal solution. The exhaustive search can take an extremely long time to find the optimal solution and thus they are only practical for problems where the search space is not too large.

For larger search spaces, other methods have been developed. These include: genetic algorithm (Holland, 1975), simulated annealing (Kirkpatrick, et al., 1983), and ant colony optimization (Haupt & Haupt, 2004) (Dorigo & Gambardella, 1997). These methods find an acceptably good solution (near the global optimum) in a fixed amount of time, rather than the best possible solution of a problem. A global optimum is a point in the search space whose value exceeds that of any other value (or is exceeded by every other value in case of minimizing). A local optimum is a point whose value exceeds (or is exceeded in case of minimizing) that of any other value in a subset of the search space.

In the current research, genetic algorithms (this is an evolutionary type algorithm) (Schwefel, 1995) and ant colony optimization (this is a particle swarm type optimization), are used as optimization methods. Each optimization method has specific mechanisms and representational components. In genetic algorithms, it is the crossover and mutation of the possible solutions (population). In ant colony optimization, it is the indirect communication (stigmergy) of homogeneous simple agents. It is difficult to directly compare the two optimization methods in a general sense as depending on the problem, one optimization method may perform better than the other. For that reason, both optimization methods are presented (in this chapter), and used and evaluated (in the following chapters).

2.6.2 Evolutionary Algorithms

2.6.2.1 Introduction to Evolutionary Computation

Evolutionary computation is based on the theory of evolution, for this reason, the key points that compose it should be initially studied. The theory of evolution proposed by Charles Darwin (1859) can be summarized in the next four key points:

1. An offspring inherits many of the characteristics of its parents;
2. There are variations in characteristics between individuals that can be passed from one generation to the next;
3. Only a small percentage of the offspring produced survive to adulthood (survival of the fittest);
4. The offspring that will survive depends on their inherited characteristics.

The theory of natural selection is the combination of these four rules. The ability to produce new forms, in essence, to innovate without outside direction other than the imperative to have children that live long enough to have children of their own is the key feature that is desired to reproduce in software (Ashlock, 2005). The ultimate proof of the utility of this approach possibly lies with the demonstrated success of life on Earth (Coley, 1999).

Evolutionary computation emulates the natural selection to solve highly complex optimization and search problems (Cox, 2005). Evolutionary computation operates on a collection (population) of data structures (creatures) to solve a problem. Each of these creatures will have explicitly computed fitness that will be used to decide which of them will be partially or completely copied to the next generation (offspring) (Ashlock, 2005).

2.6.2.2 Introduction to Evolutionary Algorithms

Evolution can be described in terms of an algorithm that can be used to solve difficult engineering optimization problems (Fogel, 1997). An Evolutionary algorithm is any computer program that uses the concept of biological evolution to solve optimization problems (Haupt & Haupt, 2004). Figure 2-9 shows a simple evolutionary algorithm.

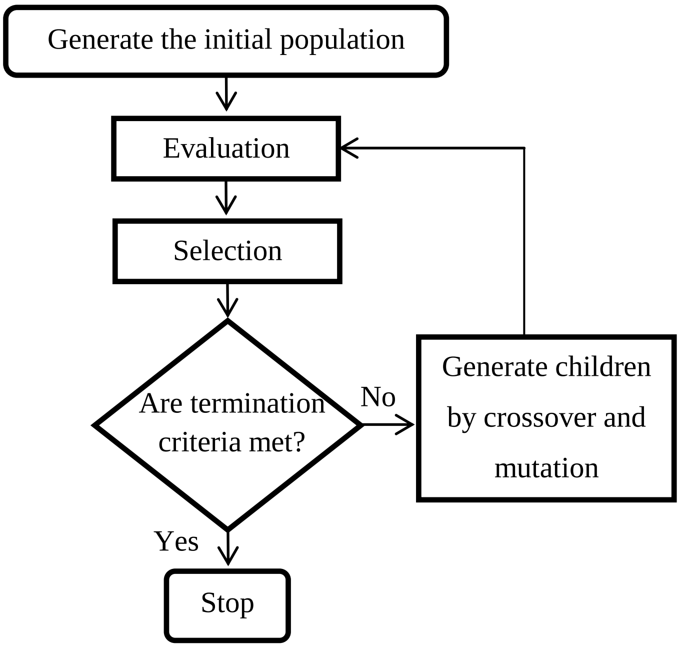

Generate the initial population
Evaluation
Selection
Are termination criteria met?
No
Generate children by crossover and mutation
Yes
Stop

Figure 2-9: A simple evolutionary algorithm

In an evolutionary algorithm, the first step is to create a population of data structures (solutions). These solutions may be filled in at random, designed to some standard, or be the output of some other algorithm. A fitness function is applied to each solution of the population to decide which solutions deserve further attention.

In the main loop of the algorithm, the solutions that are on average fitter (than the other solutions of the population) are selected to be parents (selection process). The least fit solutions are not selected as parents and die (survival of the fittest). The selected parents exchange material between them (sexual reproduction) and generate their children (offspring). This process is called crossover. The resulting children are mutated with a small probability.

In Table 2-4 and in the following sections, the terminologies (which are inspired by biology) that are used in evolutionary algorithms are explained. In addition, the processes that are involved with the evolutionary algorithms are presented in detail.

Table 2-4: The basic terminologies of the biological evolution that are used in evolutionary algorithms

Term	Description
Gene	A gene is a unit of heredity and is a sequence of DNA (deoxyribonucleic acid) that influences a particular characteristic (trait) in an organism. An example of a gene is the eye colour or the ability to metabolize alcohol (Ashlock, 2005).
Genome	A genome is an organism's complete set of DNA, including all of its genes. For bit (or binary) representations, the genome is a series of bits. For a real number representation, the genome is an integer or floating-point number (Cox, 2005).
Allele	An allele is a value of a trait at a particular locus (position) in the genome. For example, the eye colour gene could have a blue allele or a brown allele in different people (Ashlock, 2005) (Eshelman, 1997).
Genotype	The genotype expresses the overall properties of an organism by defining the nature of the chromosome (Cox, 2005).
Phenotype	The phenotype represents an individual expression of the genotype (i.e. The actual values of a genotype) (Cox, 2005).
Chromosome	A chromosome is a collection of genomes that represents a potential solution to the problem (Cox, 2005). The evaluation routine decodes these structures into some phenotypical structure and assigns a fitness value. Typically, but not necessarily, the chromosomes are bit-strings (Eshelman, 1997).
Variation	Genetic variation describes naturally occurring genetic differences among individuals of the same species. It is the process that produces new alleles, and more slowly genes, through DNA mutation, and sexual reproduction. In addition, genetic variation increases diversity and keeps the gene pool healthy (Ashlock, 2005).

2.6.2.3 Fitness Function

The fitness function is the method of assigning a numerical estimate of the quality of a particular chromosome (or the phenotype) in terms of the desired solution (Ashlock, 2005) (Cox, 2005). Fitness evaluation provides feedback to the learning algorithm regarding which individuals should have a higher probability of being allowed to multiply and reproduce and which individuals should have a higher probability of being removed from the population (Banzhaf, et al., 1998). For minimization problems, the fitness function usually approaches zero as the optimal value. For maximization problems, the fitness function usually approaches some upper boundary threshold as its optimal value (Cox, 2005).

2.6.2.4 Representation

In evolutionary algorithms, representation is the way to encode parameters into chromosomes. Traditionally, chromosomes have been encoded as bit-strings, although an increasing number of evolutionary algorithms use real-valued encodings (i.e. base-10), or encodings that have been chosen to mimic in some manner the natural data structure of the problem (Coley, 1999) (Yoon & Kim, 2013) (Mokhade & Kakde, 2014).

2.6.2.5 Selection

Selection attempts to apply pressure to the population in a manner similar to that of natural selection found in biological systems. Depending on the performance of each individual (indicated by the fitness function), the fitter (better) individuals have a greater than average chance to pass their characteristics to the next generation (Coley, 1999).

There are several methodologies for selection, but roulette wheel and tournament selection are standard for most genetic algorithms. In general, it is very difficult to choose which selection methodology works best (Haupt & Haupt, 2004).

Single Tournament Selection

The tournament selection (Goldberg & Deb, 1991) is a popular selection method (Eshelman, 1997). In single tournament selection, the population is divided randomly into small groups (subsets). The two fittest individuals in each subset are chosen to be parents. These parents will be used in the next steps to be crossed over in order to produce two offspring. These offspring will replace the two least fit members of the small group (Ashlock, 2005) (Eshelman, 1997). Selection pressure is adjusted by changing the subset's size (also known as tournament size). If the tournament size is larger, weak individuals have a smaller chance to be selected.

Single tournament selection has two advantages (Ashlock, 2005). First, for small groups of size n, the best n−2 creatures in the group are guaranteed to survive. This ensures that the maximum fitness of a group (with a deterministic fitness function) cannot decline as evolution proceeds. Second, no matter how fit a creature is compared to the rest of the population, it can have at most one child in each generation.

Double Tournament Selection

In double tournament selection (with tournament size n), a subset of n creatures is used from the population and the fittest creature of this subset is selected as a parent, and then this process is repeated with a new subset of n creatures to choose a second parent. Double tournament selection can be used with replacement (the same parent can be picked twice) or without replacement (the same parent cannot be picked twice, i.e., the first parent is excluded during the selection of the second parent) (Ashlock, 2005).

Roulette Wheel Selection

In roulette wheel selection (also called roulette selection, or fitness-proportional selection) parents are chosen in direct proportion to their fitness, i.e., the selection is biased toward chromosomes with best fitness values (Ashlock, 2005) (Cox, 2005) (Coley, 1999). If creature i has fitness f_i, then the probability of being picked as a parent is f_i/F, where F is the sum of the fitness values of the entire population (Ashlock, 2005). To implement roulette wheel selection, the following algorithm can be used (Coley, 1999).

1. Sum the fitness values of the entire population (F);
2. Choose a random number (R) between 0 and F;
3. Add together the fitness values of the population (one at a time), and stopping immediately when the sum is greater than R. The last creature added is selected as a parent;
4. Selection is continued (from Step 2) until N (the population size, assumed to be even) individuals have been selected.

Roulette wheel selection suffers from the problem of high selection pressure. In that case, if an individual is found which is much better than any other, the probability of selecting this individual may become quite high. Thus, there is the danger that many copies of this individual will be placed in the mating process, and this individual (and its similar offspring) will rapidly take over the population (premature convergence) (Eshelman, 1997).

Rank Selection

Rank selection can be used to deal with the problem of high selection pressure of roulette wheel selection (Whitley, 1989). Rank selection works similarly to roulette wheel selection except that the creatures are ordered (descending) by fitness and then selected by their rank instead of their fitness. If creature i has rank f_i, then the probability of being picked as a parent is f_i/F, where F is the sum of the ranks of the entire population. The least fit creature is given a rank of 1 so as to give it the smallest chance of being picked (Ashlock, 2005) (Cox, 2005) (Eshelman, 1997).

2.6.2.6 Elitism

For many applications, the search speed can be greatly improved by not losing the best, or elite, members between generations (Coley, 1999). When in an evolutionary algorithm, the members of a population with the highest fitness are guaranteed to survive (maintain a copy of them) in the next generation, that algorithm is said to exhibit elitism (Ashlock, 2005) (Cox, 2005) (Eshelman, 1997). Those members of the population guaranteed to survive without any change are called the elite.

In many cases, an elitist technique is used in combination with other selection methods to ensure that some of the strongest chromosomes always make it into successive generations, causing their genes to dominate the population (Ashlock, 2005) (Cox, 2005). A good compromise is to have a small number of elite members (Ashlock, 2005) (Rani, et al., 2013).

2.6.2.7 Mutation

In nature, duplicating DNA can sometimes result in errors (e.g. a fault in chromosome transcription during biological reproduction). In addition, DNA is prone to damage in day-to-day existence which also results in errors. These errors or mutations can sometimes result in good features. These features can then come together during reproduction and eventually lead to new species. Therefore, errors in DNA perform a vital role in natural evolution (Cox, 2005) (Banzhaf, et al., 1998) (Green, 1999).

Evolutionary algorithms imitate mutation by generating errors in the offspring, except of the elite member (if elitism is being applied) (Haupt & Haupt, 2004) (Coley, 1999). Mutation enriches the pool of phenotypes in the population, addresses the problem of local minimum and maximum regions, and ensures that new potential solutions, independent of the current set of chromosomes, will emerge in the population (Cox, 2005).

The mutation rate determines the probability that a chromosome will have one of its genes changed through a mutation technique. Mutation operators must be applied carefully and sparingly to the population. Too much mutation and the genome lose its ability to retain any pattern (Cox, 2005). A “good” mutation is one that increases the fitness of a data structure. A “bad” mutation is one that reduces the fitness of a data structure (Ashlock, 2005).

In general, mutation rates should be very low, in order to sustain genetic diversity but not overwhelm the population with too much noise. In the literature the formula (2-9) is proposed as a good default mutation probability rate (Cox, 2005) (Chakrabarti, et al., 2009).

$$m_r = max\left(.01, \frac{1}{N}\right) \tag{2-9}$$

Where:

m_r is the current probability of mutation

N is the population size

Binary (Bit) Inversion

For binary (bit) strings represented genomes, the binary inversion operator simply flips the value of a randomly chosen bit. A 1-bit becomes zero, and a 0-bit becomes one. Due to the nature of binary chromosomes, this is the primary and often principal mutation operator used in classical (binary-represented) genetic algorithms (Cox, 2005).

Uniform Replacement

Uniform mutation replaces a randomly selected gene (locus) with a value chosen from a uniform random distribution between the upper and lower domain bounds for the gene. This is the most frequently used mutation operator because it requires only the range of allowed values for the gene. The uniform replacement is an excellent default mutation type for genomes that are represented in real numbers (Cox, 2005).

Distribution-based Replacement

Instead of a value uniformly drawn from the domain of the gene, this operator updates the gene position with a statistical value drawn from some probability distribution. Normally, a Gaussian distribution is used (and the value is truncated or regenerated if it lies outside the allowable range for the gene) (Cox, 2005).

2.6.2.8 Crossover

Crossover is the equivalent of sexual reproduction in biology, in which two parents (see the selection, in Section 2.6.2.5) exchange genetic material ("mating") to produce one or more offspring. The goal is to combine the best features of each parent to produce children that have genetics with an elevated degree of fitness. Sometimes crossover will combine the worst features from the two parents, in which case these children will not survive for long. But sometimes it will recombine the best features from two good individuals, creating even better individuals (Cox, 2005) (Coley, 1999) (Eshelman, 1997).

There are several techniques for combining the genetic material (genomes), such as one-point, two-point, and uniform crossover. Holland (1975) proposed that a single locus has to be chosen at random, and all bits after that point be swapped. This is known as one-point crossover. In two-point crossover, two points are selected at random, and the corresponding segments from the two parents are swapped. Uniform crossover randomly swaps individual bits between the two parents (i.e. exchanges between the parent's values at randomly chosen loci).

Single-Point Crossover

In single-point crossover, a single point is selected at random along the genome. From the selection operation, two parent chromosomes have been selected from the population. The genome segments to the right (or left) of the point are swapped, creating two new chromosomes (the children). Figure 2-10 shows schematically an example of the procedure of single-point crossover.

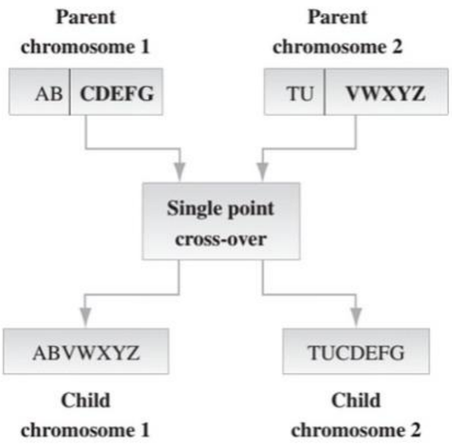


Figure 2-10: An example of single-point crossover (Cox, 2005)

Double-Point Crossover

In double-point crossover, two points are selected at random along the genome. The genome segment to the left of the rightmost point is swapped with the genome to the right of the leftmost point, creating two new children. Figure 2-11 shows schematically an example of the procedure of double-point crossover. The advantage of the double-point crossover is its inherent ability to introduce a higher degree of variability (randomness) into the selection of genetic material (Cox, 2005).

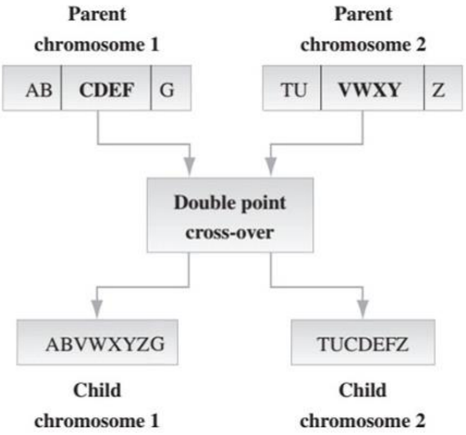


Figure 2-11: An example of double -point crossover (Cox, 2005)

Uniform Crossover

Uniform crossover involves working with individual genes rather than segments. The loci positions of the genes to be exchanged are selected at random. Figure 2-12 shows schematically an example of the procedure of uniform crossover. The probability of selecting a locus for exchange, is called the mixing rate, and can be range from low to high.

As Cox (2005) describes, the uniform crossover has the advantage of precisely controlling the amount of genetic variability in the population, it also has two significant disadvantages. First, because the crossover is done at the individual gene (locus) level rather than with sets or patterns of genes, behaviour that evolves as patterns in the population will not normally be preserved. Second, if the mix rate is too high, too much genetic diversity (that is, noise) emerges in each successive population and the search mechanism cannot find an accurate solution. On the other hand, if the mix rate is too low not enough genetic diversity will emerge and the search mechanism cannot efficiently spread over the solution terrain.

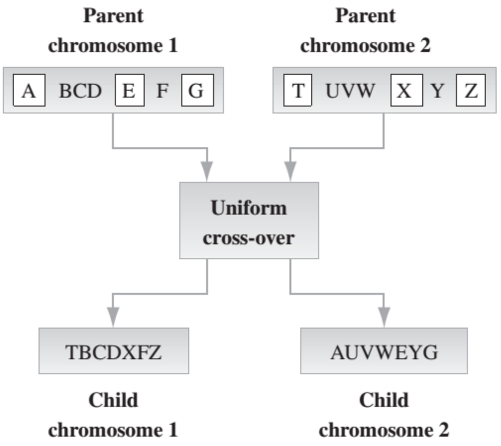


Figure 2-12: An example of uniform crossover (Cox, 2005)

2.6.3 Genetic Algorithms

2.6.3.1 Introduction

One of the best known evolutionary algorithm (EA) is the genetic algorithm (GA), first proposed and analysed by John Holland (Banzhaf, et al., 1998) (Coley, 1999) (Eshelman, 1997). Holland formalized the concepts underlying the genetic algorithms and provided the mathematical foundations for incrementally and formally improving their search techniques (Cox, 2005).

The genetic algorithm is an optimization and search technique inspired by the theory of evolution and an understanding of biology and natural evolution (Banzhaf, et al., 1998) (Green, 1999) (Haupt & Haupt, 2004) (Coley, 1999). GAs uses a heuristic rather than an analytical approach, and thus their solutions are not always exact, and their ability to find a solution is often dependent on a proper specification of the problem representation and the parameters that drive them. The GA starts with a large population of potential (or feasible) solutions and through the application of crossover (also called recombination) and mutation evolves a solution that is better than any previous solution over the lifetime of the genetic analysis (Cox, 2005). GAs have proved useful in a broad range of real-world problems (Banzhaf, et al., 1998) (Coley, 1999) (Blum, et al., 2012).

Much of the terminology used by the GA community is based, via analogy, on that used by biologists (Coley, 1999). In nature, every living organism has a unique set of chromosomes which describes how to build the organism, in GA every chromosome represents a possible solution (as a binary or other string) (Cox, 2005) (Green, 1999)

(Coley, 1999). The values of a chromosome are the parameters of a solution. In order to evaluate the solutions, a fitness function is used.

The traditional GA has two main characteristics: a) operate on fixed length binary strings, and b) uses crossover as the primary method for producing variations (Ashlock, 2005) (Banzhaf, et al., 1998) (Eshelman, 1997). The most common approach to representing individuals is a fixed length string of zeros and ones (binary encoding). Although, recently many researchers have focused on real-number coding and integer encoding, and reported that for some problems real-number coding outperform the binary representation (Arora, 2012) (Yoon & Kim, 2013).

2.6.3.2 Advantages and Disadvantages of GA

Genetic Algorithms have proved themselves capable of solving many large complex problems where other methods have experienced difficulties. Of course, the GA is not the best way to solve every problem. For instance, the traditional methods have been tuned to find quickly the solution of a well-behaved convex analytical function of only a few variables (Coley, 1999). Below are listed some of the advantages and disadvantages of GAs.

Some of the advantages of a GA include that it:

- Optimizes with continuous or discrete variables;
- Does not require derivative information;
- Simultaneously searches from a wide sampling of the search space;
- Deals with a large number of variables;
- Is well suited for parallel computers;
- Optimizes variables with extremely complex search space (they can jump out of a local minimum);
- Provides a list of optimum variables, not just a single solution;
- May encode the variables so that the optimization is done with the encoded variables;
- Works with numerically generated data, experimental data, or analytical functions;
- Has the ability to solve nonlinear, noisy, and discontinuous problems.

Some of the disadvantages of a GA include that it:

- Has a complete dependence on the fitness function. If a fitness function cannot be defined, a genetic algorithm cannot be used to solve the problem;
- Is sensitive to the genetic algorithm parameters. The stability, coherence, and convergence of genetic algorithms depend on the rate of mutation and the crossover frequency;
- Is sensitive to the genome encoding (in traditional GA). Traditional genome coding has been done through a bit-string so that mutations can work in a way similar to random genetic miscoding in biological systems.

2.6.3.3 General GA procedure

The GA works by creating an initial population of N possible solutions in the form of candidate chromosomes (or simply, individuals). Each potential solution must be a feasible solution. Feasibility means that the solution obeys all hard constraints on the solution. In addition, each potential solution must be a unique solution (Cox, 2005).

By using the fitness function (or objective function), the goodness of fit of the individuals is measured. After all individuals in the population have been evaluated for their goodness of fit, the terminating conditions are evaluated.

Until at least one of the terminating conditions is met, a new population is created and evaluated. To create the new population several steps must be taken:

a) Selecting the best individuals and removing other individuals,
b) Merging the parameters of the top best-performing individuals (crossover), and
c) Mutating the parameters in a few of the existing individuals.

2.6.3.4 Convergence

The process of breeding a genetic algorithm’s population over a series of generations to arrive at the chromosome with the best fitness value (calculated by the fitness function) is known as convergence. Thus, the population converges to a solution (Cox, 2005).

2.6.3.5 Population Diversity

Diversity is a measure of the robustness of chromosome representations, i.e. it shows how well the chromosomes (solutions) are distributed over the possible search space (richness of the gene pool).

Population diversity is a critical factor that shows the performance of a genetic algorithm. It helps estimate the chances that continuing the run of the GA would have some prospect of discovering a solution to the problem. High diversity would give an indication that it is profitable to extend the run, but low diversity would indicate the opposite (Banzhaf, et al., 1998) (Morales-Reyes & Erdogan, 2012). Diversity can be assessed in several ways, but there are two common methods that are suggested in the literature (Cox, 2005) (Chakrabarti, et al., 2009).

- By computing the variation in the fitness values over a population, i.e. the standard errors of the fitness function. A large variance indicates a robust degree of variation in the fitness functions, whereas a small variance indicates a more compact and less varied population of fitness functions.

  $$f^{v} = \frac{1}{N}\sum_{i=1}^{N}\sqrt{(f_a - f_i)^2} \quad (2\text{-}10)$$

  Where:

  f^{v} is the average variance of the population chromosomes as measured by the variance of their fitness value.

  N is the total number of chromosomes in the population.

  f_a is the average population fitness.

  f_i is fitness of the ith chromosome.

- By computing the average change of the fitness between successive generations. Diversity in the population can be examined by computing the change in average fitness from one generation to the next. Equation (2-11) shows a simple method of tracking this change. Plotting the change in the average population fitness from one generation to the next is a key indicator of convergence. As new and better solutions are created, the difference between the average fitness in each successive generation should move toward zero.

$$f_{\Delta} = \frac{1}{K}\sum_{n=2}^{K} \bar{f}_n - \bar{f}_{n-1} \tag{2-11}$$

Where:

f_{Δ} is the average change of the fitness from one generation to the next;

K is the total number of generations elapsed in the genetic search;

$\bar{f}_n$ is the average fitness of the nth generation.

2.6.3.6 Population Size

Recommendations in the literature suggest that the size of the initial population is connected to the size of the problem space (Cox, 2005) (Henley, 2015). For example, Cox (2005) suggests that an initial population should be at least as large as five times the number of variables or about half the maximum number of possible states, whichever is smaller. However, the problem to find a good initial population and the optimal population size is a hard problem and a general rule cannot be applied to every type of problem or function to be evaluated (Diaz-Gomez & Hougen, 2007).

2.6.3.7 Termination Conditions

A genetic algorithm must have termination conditions so it can stop creating new generations. These termination conditions are often used in combination. The termination conditions can include:

- When the maximum number of generations has been reached;
- When the amount of computer time has been reached;
- When a particular average fitness has been achieved;
- When the fitness function does not change after a specific number of generations.

The number of generations that evolve depends on whether an acceptable solution is reached or a set number of iterations are exceeded (Haupt & Haupt, 2004). Most GAs keeps track of the population statistics in the form of the population mean and minimum cost (Haupt & Haupt, 2004).

2.6.4 Swarm Intelligence

2.6.4.1 Introduction

Nature not only provides life but also knowledge and inspiration, as Albert Einstein stated "Look deep into nature, and then you will understand everything better". One source of such inspiration attracted a lot of researchers' attention is the way in which gregarious insects and other animals behave when they are in groups. If the group itself is considered as an individual (referred to as a swarm), in some ways, it seems to be more intelligent than any of the individuals (agents) within it (Corne, et al., 2012). A swarm can be viewed as a group of agents cooperating with a certain behavioural pattern to achieve some goal (Lim & Jain, 2009).

Swarm Intelligence (SI), which can be considered as a branch of Artificial Intelligence (AI) techniques, deals with modelling of the collective behaviours of simple agents interacting locally among themselves, and their environment, which leads to the emergence of a coherent functional global pattern (Lim & Jain, 2009) (Chu, et al., 2011) (Venayagamoorthy & Harley, 2007).

The interaction of simple agents within their environment (indirect communication) is a ubiquitous characteristic of swarms. This characteristic is called Stigmergy, which is the communication via signs or cues placed in the environment by one entity, which affect the behaviour of other entities who encounter them (Corne, et al., 2012). The term Stigmergy was introduced by Pierre-Paul Grassé (1959) in his research on the construction of termite nests.

2.6.4.2 Swarm Intelligence Models

Swarm Intelligence models are largely stochastic search algorithms. They are useful for undertaking distributed and multimodal optimization problems. The fundamental principle of swarm intelligence hinges on probabilistic-based search algorithms (Lim & Jain, 2009). All swarm intelligence models exhibit a number of general properties (Corne, et al., 2012) (Bai & Zhao, 2006). The following list shows the properties that make swarm intelligence models easy to be realized and extended, such that a high degree of robustness can be achieved (Lim & Jain, 2009).

- Each entity of the swarm is made of a homogeneous simple agent;
- The individual entities-agents act asynchronously in parallel;
- Communication among agents is generally indirect and short;
- Cooperation among agents is realized in a distributed manner (by some form of stigmergy) without a centralized control mechanism.

Some swarm intelligence algorithms use a forgetting mechanism to explore the solution space in a comprehensive manner. These algorithms are able to avoid convergence to a locally optimal solution and of finding a global optimal solution with a high probability.

Several different swarm intelligence models have been proposed and applied successfully to solve many real-world problems. Besides their applications to conventional optimization problems, SI can be used for controlling robots and unmanned vehicles, predicting social behaviours, enhancing the telecommunication and computer networks (Chu, et al., 2011). Among the most commonly used swarm intelligence models are: Ant Colony Optimization (ACO) (Dorigo & Caro, 1999), Particle Swarm Optimization (PSO) (Kennedy & Eberhart, 1995) and Bee Colony Optimization (BCO) (Karaboga, 2005).

2.6.5 Ant Colony Optimization

2.6.5.1 Biological Inspiration

Deneubourg et al. (1990) (Goss, et al., 1990) designed the biological double bridge experiment to investigate the pheromone trail laying with regard to the behaviour of Argentine ants (Linepithema humile).

In the first scenario of the experiment, a double bridge of different lengths was used to connect the ant's nest with a food source (Figure 2-13). The long bridge was twice as long as the shorter bridge. In most runs of this experiment, it was found that after a few minutes nearly all ants use the shorter bridge. This is interesting because Argentine ants cannot see very well. The explanation of this behaviour has to do with the fact that the ants lay pheromone along their path. It is likely that ants which randomly choose the shorter branch arrive earlier at the food source. When they go back to the nest, they smell some pheromone on the shorter branch and, therefore, prefer this branch. The pheromone on the shorter branch will accumulate faster than on the longer branch so that after some time the concentration of pheromone on the former is much higher, and nearly all ants take the shorter branch (Merkle & Middendorf, 2005).

In the second scenario of the experiment, the bridges have the same length (Figure 2-14). After some minutes, nearly all ants use the same branch, but in several repetitions it is a random process which of the two branches will be chosen. The explanation is that when one branch has got a slightly higher pheromone concentration due to random fluctuations this branch will be preferred by the ants so that the difference in pheromone concentration will increase and after some time all ants take this branch (Dorigo, et al., 2006) (Merkle & Middendorf, 2005).

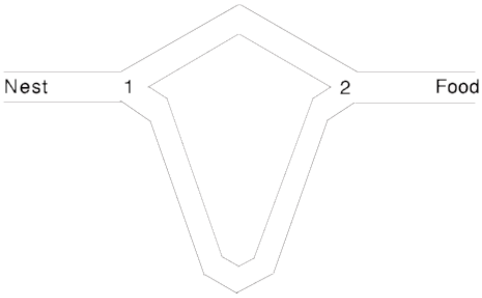


Figure 2-13: Bridges have different lengths (Goss, et al., 1989) (Goss, et al., 1990)

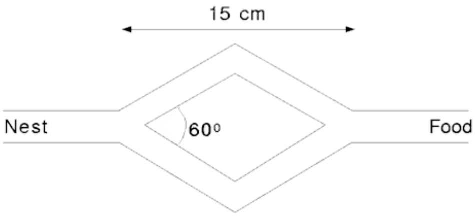


Figure 2-14: Bridges have equal lengths (Deneubourg, et al., 1990) (Goss, et al., 1990)

From those experiments, it is clear that the navigation of ants to food sources depends on the deposition of pheromone by individual ants. Biological ants initially wander randomly around their environment to search for food. When they find a suitable food source, they return to their colony while laying a trail of pheromone along their path. When other ants search for food they will sense the pheromone trail (of their precursors); that will influence the path they choose to follow. In the process, the strength of pheromone will increase along the paths. Given time, the faster or safer path to a food source will have the higher amount of pheromone. This is possible because the pheromone trails evaporate over time. Therefore, the longer time ants take to travel from the food source to the colony, the more pheromone will have evaporated. Eventually, all ants will follow the same path to the food source (Lim & Jain, 2009) (Corne, et al., 2012).

2.6.5.2 Introduction

Dorigo et al. (Dorigo, 1992) (Dorigo, et al., 1996) were inspired by the double bridge experiment to design the first ACO system, with an algorithm called Ant System (AS), which initially applied to the travelling salesman problem. An essential aspect of ACO is the indirect communication of the ants via pheromone (stigmergy). Ants deposit pheromone on the ground (environment) to mark their paths to the food sources. The pheromone traces can be smelled by other ants and lead them to the food source. ACO exploits a similar mechanism for solving optimization problems (Dorigo, et al., 2006) (Merkle & Middendorf, 2005).

ACO proved to be successfully applied to a wide range of different discrete optimization problems. The majority of these problems are NP-hard, which it is unlikely to have an efficient algorithm to compute their exact optimal solution. ACO algorithms can be useful for quickly finding high-quality solutions (Dorigo, et al., 2006). Application areas of ACO based models include: the travelling salesman problem, network routing, graph colouring, quadratic assignment, scheduling problems, vehicle routing, frequency assignment, as well as sequential ordering (Bonabeau, et al., 2000) (Ostfeld, 2011) (Barbosa, 2013) (Câmara, 2015).

2.6.5.3 The Ant Colony Optimization Metaheuristic

ACO has been formalized into a metaheuristic for combinatorial optimization problems (Dorigo & Caro, 1999) (Dorigo & Caro, 1999). A metaheuristic is a set of algorithmic concepts that can be used to define heuristic methods applicable to a wide set of different problems. In other words, a metaheuristic is a general-purpose algorithmic framework that can be applied to various optimization problems with relatively few modifications (Dorigo, et al., 2006).

The idea of ACO metaheuristic is to let artificial ants construct solutions for a given combinatorial optimization problem. A solution can be viewed as a path through a corresponding decision graph (also called construction graph). The aim is to let the artificial ants find paths through the decision graph that correspond to good solutions.

Ants construct feasible solutions based on the current pheromone trail strengths (initially, the pheromone is randomly distributed). To construct a solution, each ant steps through the decision graph choosing among feasible paths. At each point, the ant chooses among available arcs according to a function of the pheromone strength on each arc, and of the local heuristic values of each arc (Corne, et al., 2012). To tune the level at which the algorithm relies on exploration (pheromone trail) and heuristic value, two parameters α (alpha) and β (beta) are used. If α = 0, then the closest the arcs are more likely to be selected. If β = 0, then only pheromone amplification is at work.

Formula (2-12) was used by Dorigo & Caro (in 1999) for the travelling salesman problem (TSP), in order to compute the ant-routing table. When the ants complete their paths, their corresponding solutions are evaluated, and pheromone is laid on each arc travelled in proportion to the overall solution quality. Each ant k deposits a quantity of pheromone $\Delta\tau^k(t) = 1 \,/\, J_\psi^k\,(t)$ on each connection l_{ij} that it has used, where $J_\psi^k\,(t)$ is the length of the tour $\psi^k(t)$ done by ant k in iteration t, as shows in formula (2-15).

This pheromone and the local heuristic value guide the following ants of the next iteration (by using formula (2-12)) so that they search near paths to good solutions. For the TSP, the local heuristic value of formula (2-13) is used, where $J_{c_ic_j}$ is the distance between cities i and j. The shorter the distance between two cities i and j, the higher the local heuristic value. The probability $p_{ij}^k(t)$ with which at the t-th algorithm iteration an ant k located in city i chooses the city $j \in N_i^k$ to move to, is given by formula (2-14).

$$A_{ij} = \frac{\left[\tau_{ij}(t)\right]^{alpha}\left[\eta_{ij}\right]^{beta}}{\sum_{l\in N_i}[\tau_{il}(t)]^{alpha}[\eta_{il}]^{beta}} \qquad (2\text{-}12)$$

$$\eta_{ij} = {}^{1}\!/_{J_{c_ic_j}} \qquad (2\text{-}13)$$

Where:

N_i is the set of all the neighbor nodes of node i

$\tau_{ij}(t)$ is the functional composition of the pheromone trail

η_{ij} is the local heuristic values.

$J_{c_ic_j}$ is the distance between cities i and j

$$p_{ij}^{k}(t) = \frac{a_{ij}(t)}{\sum_{l \in N_i^k} a_{il}(t)} \tag{2-14}$$

Where:

$N_i^k \subseteq N_i$ is the feasible neighborhood of node i for ant k.

In order to avoid convergence to a locally optimal solution, pheromone evaporation is performed. Pheromone evaporation is a useful mechanism allowing the solution space to be explored in a comprehensive manner (Lim & Jain, 2009). In pheromone evaporation, the arc’s pheromone strength is updated as follows: $\tau_{new} = (1 - \rho)\tau_{old}$, where $\rho \in (0, 1]$ controls the speed of pheromone decay (Corne, et al., 2012). The process continues until some stopping criterion is met, for example, a certain number of iterations have been performed or a solution of a given quality has been found (Merkle & Middendorf, 2005). The ACO metaheuristic is shown in Figure 2-15.

$$\tau_{ij}(t) \leftarrow \tau_{ij}(t) + \Delta\tau^{k}(t), \forall\, l_{ij} \in \psi^{k}(t), k = 1, \cdots, m \tag{2-15}$$

Where:

m is the number of ants (for each iteration it remains constant)

```
Set parameters, initialize pheromone trails
while termination condition not met do
        ConstructAntSolutions
        ApplyLocalSearch (optional)
        UpdatePheromones
end while
```

Figure 2-15: The Ant Colony Optimization Metaheuristic (Dorigo, et al., 2006)

2.7 Summary

This chapter provides the necessary knowledge background that forms the bedrock of the research documented in this thesis. It includes an appraisal of previous work that has been undertaken to both measure and predicting air quality and articulates the techniques and rationale for choosing the techniques utilised in this research.

The following chapter presents the performed investigation experiments in air quality forecasting by using the described methods (such as cross-validation, sensitivity analysis, etc.), forecasting methods, and forecasting verification process for the presented study areas.

Chapter 3

Investigation and Forecasting of Environmental Parameters

3.1 Introduction

This chapter presents the experiments performed to facilitate the development of effective computational intelligence (CI) and statistical methods capable of forecasting the selected environmental parameters of interest (Figure 3-1). These environmental parameters include two criteria pollutants (O_3 and Benzene) and the Common Air Quality Index (CAQI) which is based on criteria pollutants (Kumar & Goyal, 2013). Additional information regarding the selected parameters can be found in Appendix IV. The aforementioned environmental parameters were selected to be forecasted for the following reasons:

- O_3 was selected because it is the most commonly forecasted pollutant in the studied literature and in operational forecasting systems, and to investigate the forecasting of a secondary pollutant.

- Benzene was selected to investigate the forecasting of a pollutant that is not commonly forecasted (in the studied literature) but is addressed by existing operational forecasting systems (such as the CALIOPE operational forecasting system) as well as by the relevant EU legal framework. An additional reason for its selection is that it was found to have the strongest association with respiratory diseases when compared to other criteria pollutants, such as PM_{10}, NO_2, SO_2, $O_{3,}$ etc. (Oftedal, et al., 2003).
- The CAQI was selected to investigate the forecasting of an air quality index. In general, there are scarce research results concerning the performance of air quality forecasting models in terms of combined environmental pressure indicators (Kumar & Goyal, 2013).

The following two main steps are considered in this chapter.

Step 1. Construct operational forecasting models for the aforementioned environmental parameters of interest, with the aid of CI methods that include Artificial Neural Networks (ANNs) and Decision Trees (DT), as well as the statistical method of Linear Regression (LR).

Step 2. Employ cross-validation, sensitivity analysis, and an additional forecasting method in an effort to investigate the model's performance, to provide with reliable and accurate results and finally to increase the forecasting performance.

AQ data coming from monitoring stations of the two largest cities of Greece (Athens and Thessaloniki) were studied. In the case of Athens, Benzene and Ozone were the pollutants selected for the study as they are influenced directly or indirectly by traffic (a major environmental pressure in many EU cities). More specifically, the forecasting investigation focused on:

1. The hourly concentration levels of Benzene, in order to investigate the forecasting of detailed variation (as the CALIOPE operational forecasting system that provides with hourly benzene forecasts).
2. The highest daily 8-hour running average (8-HRA) of Ozone concentration levels. The 8-HRA parameter was selected because it is regulated by the EU as a standard to measure exceedances for ozone.

In the case of Thessaloniki, the aim was to investigate an air quality index and not individual pollutants as in the case of Athens. For this purpose, the CAQI was used, which is suggested by the European Environment Agency (EEA), as the parameter of interest that needs to be forecasted (in an hourly and daily horizon) in order to provide health related warnings and information to the general public. For that purpose, the following general scenarios were investigated:

1. The estimation of the CAQI levels by forecasting the CAQI values,
2. The estimation of the CAQI levels by forecasting individual pollutant levels, and
3. The direct forecasting of the CAQI levels.

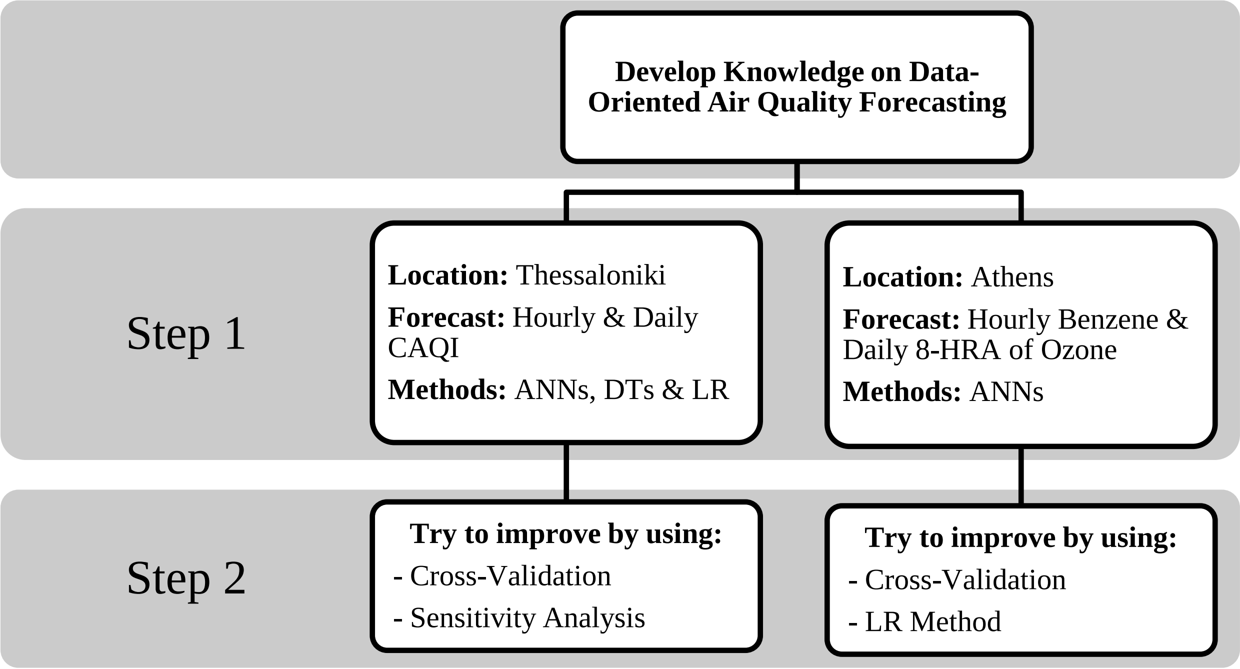


Figure 3-1: The aims in order to develop knowledge on data-oriented air quality forecasting

3.2 Data Presentation

The air quality and meteorological data, came from a total of six monitoring stations, two in Athens (Patisia and Liosia) from a total of fifteen, and four in Thessaloniki (Agia Sofia, Panorama, Sindos, and Kordelio) from a total of seven. In the case of Thessaloniki, the aforementioned four stations were studied (for their CAQI Levels) but the Agia Sofia station was selected for the development of the forecasting models (because it demonstrates the high percentage of “high” and “very high” CAQI levels). These particular monitoring stations were chosen because of the availability of their data and in order to evaluate different location types (see Section 2.4). The measurement technique used by the monitoring station of Patisia for Benzene is the Gas chromatography with a photoionization detector (GC-PID) by using equipment of the Synspec Company (AirBase-V8, 2014). In the case of Ozone both monitoring stations (Patisia and Liosia) use the UV absorption measurement technique by using the "HORIBA APOA 360" equipment (AirBase-V8, 2014).

Table 3-1 presents the parameters under investigation when forecasting Benzene concentrations for the monitoring station of Patisia in Athens. The next three input datasets were used for that purpose.

- Dataset P1: A two-year-long record of hourly air quality and meteorological data, for the years 2004 and 2005,
- Dataset P2: A dataset for the years 2006 and 2007 which was used for evaluation purposes and,
- Dataset P1&2: An overall dataset, for the years 2004-2007.

When forecasting 8-HRA of ozone, six input datasets were used in total, three for each monitoring station (Patisia and Liosia, in Athens). The first three datasets of the following list were used for the monitoring station of Liosia, and the remaining three for the monitoring station of Patisia. Table 3-2 presents the parameters that were included in these datasets. It should be noted that the parameter of wind speed at 9 am was included in the data because relative studies found that, that influences the ozone forecasting in Athens (Ziomas, et al., 1995).

- Dataset L1: A four-year long record of hourly measurements for the time period 2002-2005,
- Dataset L2: A dataset for the year 2007 for evaluation purposes (data for the year 2006 were not available) and,
- Dataset L1&2: An overall dataset that contains the previous datasets L1 and L2.
- Dataset P3: A three-year long dataset for the years 2002, 2003 and 2005 was used together with
- Dataset P4: An additional dataset for evaluation purposes for the year 2007 (again, data for the years 2004 and 2006 were not available) and also,
- Dataset P3&4: A dataset that contains the previous datasets P3 and P4.

Table 3-1: Parameters under investigation when forecasting Benzene concentrations (Datasets P1, P2 and P1&2)

	Parameter	Unit	Abbreviation
1	Benzene	$\mu g/m^3$	BE
2	Carbon Monoxide	mg/m^3	CO
3	Sulphur Dioxide	$\mu g/m^3$	SO_2
4	Relative Humidity	%	RH
5	Air Temperature	C^o	Ta
6	Wind Speed	m/s	WS
7	Wind Direction	deg	WD

Table 3-2: Parameters under investigation when forecasting 8-hour running average ozone values (Datasets L1, L2, L1&2, P3, P4, P3&4)

	Parameter	Unit	Abbreviation
1	Ozone	$\mu g/m^3$	O_3
2	Nitrogen Dioxide	$\mu g/m^3$	NO_2
3	Sulphur Dioxide	$\mu g/m^3$	SO_2
4	Nitric Oxide	$\mu g/m^3$	NO
5	Relative Humidity	%	RH
6	Air Temperature	C^o	Ta
7	Wind Speed	m/s	WS
8	Wind Speed at 9:00am	m/s	WS9
9	Wind Direction	deg	WD

The CAQI has been computed for each of the four locations of Thessaloniki, both using hourly and daily concentration values. The traffic CAQI was applied for Agia Sofia, Kordelio and Sindos while the urban background CAQI for the Panorama station. Table 3-3 presents the number (and the corresponding percentage) of hourly and daily CAQI values, for each CAQI level and station. From Table 3-3 it is evident that Agia Sofia and Kordelio are the stations that demonstrate the highest percentage of values in the CAQI classes “High” and “Very High”. The monitoring station of Agia Sofia was selected for the development of CAQI forecasting models because more data are available on this station. Thus, air quality and meteorological data used for the ANNs and DT models described hereafter, originate from this monitoring station of Thessaloniki.

Table 3-3: The number (and the corresponding percentage) of hourly and daily Common Air Quality Index values, for each Common Air Quality Index level and station

CAQI Level	Agia Sofia (2001-2003)		Sindos (2001)		Kordelio (2001-2002)		Panorama (2001-2002)	
	Hourly	Daily	Hourly	Daily	Hourly	Daily	Hourly	Daily
Very high	3510 (14%)	127 (12%)	591 (7%)	0 (0%)	2538 (17%)	95 (15%)	243 (1%)	1 (0%)
High	3706 (15%)	168 (16%)	618 (8%)	0 (0%)	1958 (13%)	112 (18%)	613 (4%)	12 (2%)
Medium	7414 (30%)	438 (42%)	1917 (24%)	3 (1%)	3866 (26%)	225 (36%)	5569 (33%)	306 (43%)
Low	8264 (34%)	304 (29%)	3558 (45%)	361 (99%)	5056 (34%)	183 (29%)	9427 (56%)	377 (53%)
Very low	1666 (7%)	18 (2%)	1309 (16%)	1 (0%)	1514 (10%)	17 (3%)	885 (5%)	21 (3%)

In order to forecast hourly and daily CAQIs for the Agia Sofia monitoring station of Thessaloniki the following three input datasets were used. Table 3-4 presents the parameters that were included in the data, which correspond to the years 2001-2003. This period was selected based on its low percentage of problematic or missing data.

- Dataset 1 (lagged index values) includes only hourly or daily lagged CAQI values determined during the previous day (day T-1, T-2, … , T-10), or the previous hours (hours T-1, T-2, … , T-10). The CAQI values were thus forecasted with the aid of one up to 10 lagged CAQI values, leading to 10 different model results for hourly and daily values. Dataset 1 consists of 1,042 daily records and 23,671 hourly records.
- Dataset 2 (concentration and meteorological values at time T) includes concentration values of CO, SO_2, and O_3 and six meteorological values, as follows: (1) temperature, (2) dew point, (3) humidity, (4) sea level pressure, (5) wind direction and (6) wind speed. Dataset 2 consists of 1,043 daily records and 23,672 hourly records (plus one record in comparison to Dataset 1, as the latter includes only lagged values).
- Dataset 3 (lagged index values and Dataset 2) includes hourly or daily lagged (previous) CAQI values determined during the previous day (day T-1, T-2, … , T-5), or the previous hours (hours T-1, T-2, … , T-5), separately for the two types of index, and the above-mentioned air quality and meteorological values (Dataset 2). The CAQI values were thus forecasted with the aid of one up to five lagged CAQI values, leading to 5 different model results for hourly and daily values. Dataset 3 consists of 1,042 daily records and 23,671 hourly records.

Table 3-4: Parameters under investigation when forecasting hourly and daily Common Air Quality Index

	Parameter	Unit	Abbreviation
1	Carbon Monoxide	mg/m^3	CO
2	Nitrogen Dioxide	$\mu g/m^3$	NO_2
3	Nitrogen Monoxide	$\mu g/m^3$	NO
4	Ozone	$\mu g/m^3$	O_3
5	Respirable Particles	$\mu g/m^3$	PM_{10}
6	Sulphur Dioxide	$\mu g/m^3$	SO_2
7	Air Temperature	C^o	Ta
8	Dew Point	C^o	DP
9	Relative Humidity	%	RH
10	Sea Level Pressure	hPa	SLP
11	Wind Speed	m/s	WS
12	Wind Direction	deg	WD

3.3 Data Pre-Processing

Every environmental dataset may contain missing values and outliers. Some data-driven forecasting methods (e.g. artificial neural networks) cannot adequately deal with missing values thus, when using these methods, records with missing values must either be excluded from the data or missing values inferred. While records removal solves the missing values problem, valuable information may be lost and the available information for training (the forecasting methods) reduced, which may be a real issue if the data is already limited.

The data available for the investigative experiments were already limited, thus, the Lagrange interpolating polynomial (see Section 2.5.2) and the linear interpolation method (see Appendix V) were used to infer the missing values in the case of Athens and Thessaloniki respectively. It should be noted that no outliers were detected in the investigated data for either location. In addition, the wind direction values were encoded in all datasets by using the formula $WD=1+sin(\theta+\pi/4)$, in order to replace the cyclic nature of this variable with a linear one. Afterwards, a dimensionality reduction (i.e. minimise the parameters to be included in the development of the forecasting models, and identify the most important and influential parameters in the studied datasets) was performed. It is worth mentioning that all the available pollutants and meteorological variables in the data were used in order to perform the selection by only using feature selection methods. This was performed in order to simulate the use of the models as if there is no prior experience in air quality forecasting. The rest of this section presents in detail the aforementioned pre-processing procedure applied to data from each monitoring station and pollutant.

In the datasets of Patisia and Liosia (Athens' monitoring stations) for both Benzene and 8-HRA of ozone forecasting, if missing values corresponded to more than 30% of the hourly data records in one day, the whole daily record was excluded from further analysis. In all other cases, the Lagrange interpolating polynomial (Burden & Faires, 2011) was applied for filling in missing values. For each missing value, a polynomial of 6th degree was constructed, by using three data points before and after the missing value (as shown in Figure 3-2).

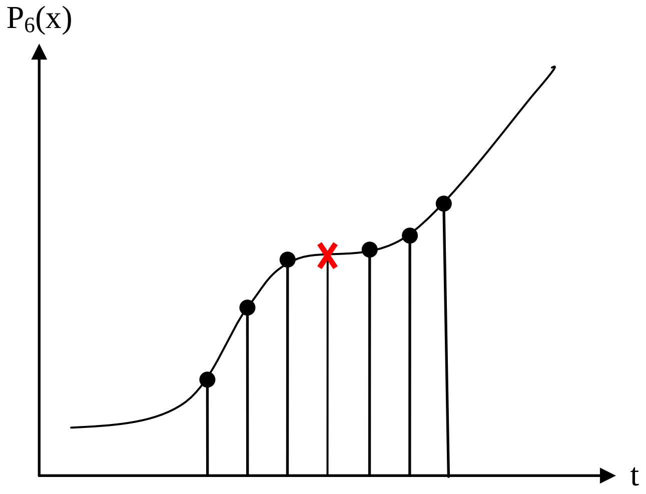


Figure 3-2: An example of using a 6th degree Lagrange interpolating polynomial

In order to perform dimensionality reduction in the datasets used to forecast Benzene (Dataset P1, P2 & P1&2), the Principal Component Analysis (PCA) was used (see Appendix V). Table 3-5 presents the eigenvalues and variances of the computed principal components. One way to select the principal components of interest is the Kaiser rule, which keeps all eigenvalues greater than one (Lance, et al., 2006). Table 3-6 presents the overall covariance of hourly Benzene concentrations with regard to the investigation parameters. It is clear that the highest covariance to Benzene is demonstrated by Sulphur Dioxide (SO_2), Relative Humidity (RH) and Carbon Monoxide (CO). The principal component analysis suggests (by applying the Kaiser rule) to eliminate one parameter. In addition, the results of Table 3-6 indicate that the wind direction has no covariance to Benzene. For that reasons, the PC6 (assuming to be the wind direction) was eliminated from the Benzene datasets. The remaining data (PC1 to PC5) accounted for 99.95% of the variation of both Dataset (P1 and P2), which were used as input to the ANNs.

Table 3-5: Eigenvalues and variances of the computed principal components

Principal Component	Data (04-05)			Data (06-07)		
	Eigenvalues	Variance (%)	Cumulative Variance	Eigenvalues	Variance (%)	Cumulative Variance
PC1	672,62	68,52	68,52	583,68	60,42	60,42
PC2	275,81	28,10	96,61	315,39	32,65	93,06
PC3	27,93	2,85	99,46	43,34	4,49	97,55
PC4	3,35	0,34	99,80	21,82	2,26	99,81
PC5	1,44	0,15	99,95	1,41	0,15	99,95
PC6	0,50	0,05	100,00	0,46	0,05	100,00

Table 3-6: The overall covariance of hourly Benzene concentrations with regard to the other parameters of work

Covariance of Benzene to:	2004 (%)	2005 (%)	2006 (%)	2007 (%)	General (%)
SO_2	59	29	62	58	52.1
RH	20	20	12	14	16.3
CO	11	34	6	7	14.4
WS	5	18	13	16	13.0
Ta	4	0	6	5	4.0
WD	0	0	0	1	0.2

In the case of forecasting ozone in Athens, for each day of the available dataset, the highest 8-hour mean running average (HRA) of ozone was calculated, as the purpose of this work was to forecast this particular parameter. Moreover, the (i) mean value, (ii) the max value and (iii) the highest 8-hour mean values were calculated for all the remaining parameters (1. Nitrogen Dioxide, 2. Sulphur Dioxide, 3. Nitric Oxide, 4. Relative Humidity, 5. Air Temperature, 6. Wind Speed and 7. Wind Direction), with the exception of the wind speed at 9:00 am, resulting in (7*3) + 1 = 22 input parameters per day per station, to be used for the construction of the forecasting models. In addition, PCA was used on the data (when ANNs were applied) in order to change their dimensionality and thus, help the ANNs to find patterns into the data. The modified data (i.e. after the use of PCA) were used as input to the ANNs. It should be noted that none of the features (input parameters) were excluded from the training phase.

In the case of forecasting the CAQI in Thessaloniki, the first step of the pre-processing procedure was to select the records that include all the required input values. Specifically, for Datasets 1 and 3 (of Section 3.2), the records that include CAQI values were selected. In the case of Dataset 2 (of Section 3.2) the records that include all target values (CO, SO_2

and O_3) were selected. The next step was to use the linear interpolation method (see Appendix V) to calculate all missing values.

Moreover, as ANNs require numerical inputs, the CAQI nominal values were converted to CAQI numerical values, as described in Table 3-7. The specific numerical values were selected, because of the hyperbolic tangent sigmoid transfer function (*htstf*), which was used to normalize the input data in the case of ANNs and DTs. This function receives values between -1 and +1, and because the full range of these values must be covered, the mapping of the five nominal values is done by assigning to the lower and higher nominal values the relevant lower and higher values of the *htstf*, and by assigning equidistant numerical values (i.e. -0.5, 0, and 0.5) to the three other nominal values of the CAQI.

Table 3-8 presents the way that the continuous numerical values produced by the forecasting models (output) are mapped back to nominal CAQI values. For this mapping, the numerical values were divided in to five equal in range groups, in order to properly map back the CAQI numerical value to the nominal CAQI value. In addition, PCA was used to the data (when ANNs were applied) in order to change their dimensionality and thus, help the ANNs to find patterns into the data (as in the case of forecasting 8-HRA of ozone). The modified data were used as input to the ANNs without excluding input parameters.

Table 3-7: Conversion table from Common Air Quality Index levels (nominal values) to Common Air Quality Index numerical values

CAQI Level Nominal Value	CAQI Numerical Value
Very Low	-1
Low	-0.5
Medium	0
High	0.5
Very High	1

Table 3-8: Conversion table from Common Air Quality Index numerical values to Common Air Quality Index levels (nominal values)

CAQI Numerical Value (n)	CAQI Level Nominal Value
n<=-0.6	Very Low
n>-0.6 AND n<=-0.2	Low
n>-0.2 AND n<0.2	Medium
n>=0.2 AND n<0.6	High
n>=0.6	Very High

3.4 Performing Periodicity Analysis

Periodicity analysis was performed for the two target features of the Athens monitoring stations (hourly Benzene and daily 8-HRA concentration values), with the aid of MATLAB. The highest values of strength in the periodogram determine the periodicity of the specific parameter within the time series. Table 3-9 presents the maximum strength values and their corresponding periodicities for Benzene at Patisia station for Datasets P1 and P2. Figure 3-3 illustrates the periodogram of Benzene concentrations from Patisia station and indicates the four periodicities with maximum strength. These findings support the hypothesis that the basic source of Benzene is traffic: this is indicated from the periodicity of traffic per week (7-days: highest traffic on working days and less on weekends), accompanied by a periodicity of 8 hours and 24 hours, which closely correlated to the daily traffic patterns. Overall, Benzene hourly concentrations seem to follow the life cycle of the city operation and especially mobility patterns.

Table 3-10 presents the four maximum strength values and their corresponding periodicities for Ozone 8-HRA concentration values, at Patisia and Liosia stations. It is clear that both monitoring stations have a difference between the periodicities of the two datasets. The periodicity of 175-days is very pronounced in the 2007 data for Patisia, indicating a shift in the behaviour of the pollutant between the cold and the warm period of the year, while there are even stronger periodicities, all indicating a periodicity close to monthly, especially for the Liosia area (i.e. an area indirectly influenced by traffic). It is also worth noting that for both locations, there is a strong 24-day periodicity, indicating a monthly cycle of the phenomenon.

Table 3-9: The four maximum strength values and their corresponding periodicities for Benzene concentrations at Patisia station

#	Hourly Benzene Patisia Data (04-05)			Hourly Benzene Patisia Data (06-07)		
	Strength	**Cycles /Hour**	**Periodicity**	**Strength**	**Cycles / Hour**	**Periodicity**
1	$2.36x10^7$	0.0059844	7 days	$1.11x10^7$	0.0061	6.8 days
2	$1.92x10^7$	0.125	8 hours	$9.14x10^6$	0.0066	6.3 days
3	$1.35x10^7$	0.041667	24 hours	$8.74x10^6$	0.0067	6.2 days
4	$8.30x10^6$	0.16667	6 hours	$8.36x10^6$	0.0069	6.1 days

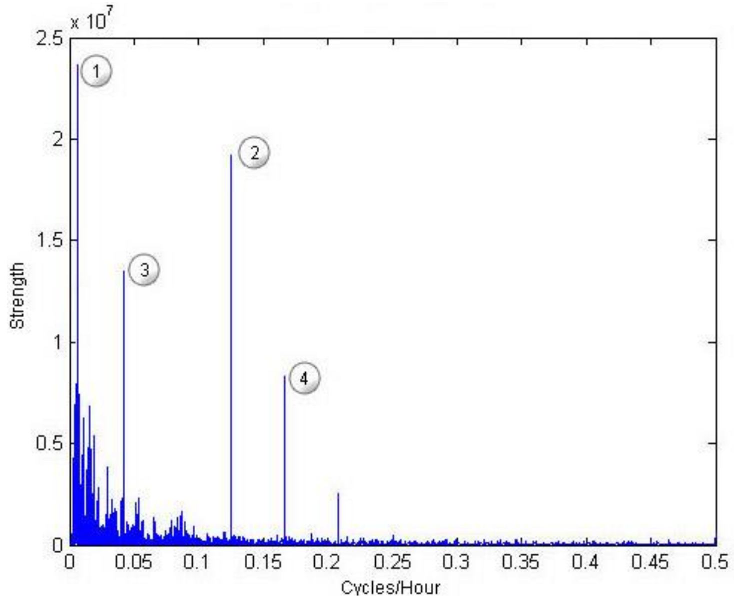


Figure 3-3: The periodogram of Benzene from Patisia station. The numbering indicates the four periodicities with maximum strength

Table 3-10: The four maximum strength values and their corresponding periodicities for daily Ozone 8-hour running average concentration values at Patisia and Liosia monitoring stations.

Station	#	8-HRA Ozone (Datasets P3 and L1)			8-HRA Ozone (Datasets P4 and L2)		
		Strength	Cycles /Day	Periodicity	Strength	Cycles /Day	Periodicity
Patisia	1	$7.71x10^6$	0.0035	286 days	$4.40x10^8$	0.0057	**175** days
	2	$2.92x10^6$	0.0255	39 days	$3.02x10^8$	0.0419	24 days
	3	$1.64x10^6$	0.029	34 days	$2.00x10^8$	0.0421	24 days
	4	$1.27x10^6$	0.1485	1 days	$1.85x10^8$	0.0418	24 days
Liosia	1	$9.47x10^7$	0.0033	303 days	$4.48x10^9$	0.0418	24 days
	2	$3.21x10^7$	0.0041	244 days	$3.08x10^9$	0.042	24 days
	3	$1.70x10^7$	0.0025	400 days	$1.05x10^9$	0.0417	24 days
	4	$1.24x10^7$	0.0074	135 days	$2.38x10^8$	0.013	77 days

3.5 Performing Cross-Validation

Cross-validation is used in the second step of the experiments to investigate how the data-driven methods will generalize to an independent data and thus provide more reliable and accurate results.

For the application of the ANN backpropagation algorithm, the following procedure was used: For each one of the input datasets (of the investigation experiments), the 10-fold Cross-Validation method was employed. Every dataset was thus divided into 10 subsets, each one of the 10-1 subsets was used for training, and the last subset was used for validation and testing (in practice, half was used for the validation and half for the final testing and evaluation). The error on the validation subset was monitored during the training process: when the network begins to overfit the data, the error on the validation subset typically begins to rise (Gencay & Qi, 2001). On this basis, when the validation error increases, training can be stopped, and the weights and biases at the minimum of the validation error are retained. The error of the test subset was not used during the training but was used for computing the performance of the different models.

In the case of the DT models, the 10-fold cross-validation was also used, where the 10-1 subsets were involved in the training phase, and the remaining subset was used for testing. The bootstrap aggregation meta-algorithm was used for each input subset, in order to grow a different number of trees.

In the case of the Linear Regression Models, the 10-fold cross-validation was again used, where the 10-1 subsets were involved in the training phase, and the remaining subset was used for testing.

3.6 Forecasting Hourly Benzene Concentration Values

For the development of the ANN forecasting model for Benzene, twelve network configurations were implemented and tested during the training phase, for all Benzene datasets. In the first step of the investigation experiments (when CV was not used), 50% of the data was used for training, 25% for validation and 25% for testing. In the second step CV was used to split the data into training, testing and validation subsets, as described in Section 3.5. Figure 3-4 illustrates the architecture of the first configuration of the neural network. Table 3-11 presents the performance results by using ANNs for the different model configurations, for the three different datasets, i.e. the dataset for the period 2004-2005 (P1), the dataset for the years 2006 and 2007 (P2) and the dataset for the years 2004-2007 (P1&2) in comparison with the performance results when CV was used, with the aid of the coefficient of determination (R^2) and the index of agreement (d or IA). Additional information regarding these indices can be found in Appendix II. Table 3-12 presents the forecasting performance of hourly Benzene concentrations by using Linear Regression models for the three different datasets. Table 3-13 presents the model results with the aid of all statistical indicators involved, in order to evaluate the best configurations concerning their forecasting performance upon the three datasets. Table 3-14 presents the model performance for two different criteria defining the events of interest (exceedances): 5μg/m^3 and 10μg/m^3 in order to evaluate also the influence of the event definition criteria in the event forecasting ability. In particular, Table 3-14 presents the calculated Critical Success Index (CSI) values and the number of the observed alarms and predicted alarms per dataset and per exceedance value. Information regarding the CSI calculation is available at Appendix III. In addition, it shows the numbers and rates of the alarms that either predicted correctly (a: true positives) or not (c: false negatives). The values of Table 3-14 are presented as real numbers because the 10-fold cross-validation procedure is used (consequently the results are averaged), although they refer to discrete values.

From the results presented in Tables 3-11 and 3-13, it is evident that configurations numbered as 7 and 5 have the best results concerning Benzene forecasting, for the dataset for the years 2004-05. Configuration number 7 was chosen as the best configuration as it performs better and has a simpler architecture (less neuron). From the same tables, it is clear that configuration number 5 produces the best results for the dataset of the years 2006-07 and configuration numbered as 8 produce the best results for the joined dataset for the years 2004-2007.

From Tables 3-11 and 3-12 it is clear that the ANNs have better forecasting performance in comparison with the Linear Regression models, for hourly Benzene concentrations in Patisia monitoring station for the data periods 2004-2005 and 2006-2007, on the basis of the IA criterion. In addition, ANNs provide better results (IA=0.92) with less data (dataset 06-07) in comparison with Linear Regression (IA= 0.89). In contrary, when the complete dataset (for the period 2004-2007) was used, the Linear Regression model provides better forecasting performance (IA=0.91) in comparison with the best ANN model (IA=0.89). The decrease of the forecasting performance of the ANN models by using the complete dataset may have attributed to the different statistical characteristics of data periods 2004-2005 and 2006-2007.

From Table 3-11 it is evident that when CV is used, the forecasting performance was increased. Overall it can be concluded that the proposed methodology provides good forecasting performance for hourly Benzene concentrations for the Patisia monitoring station and has a high forecasting ability.

Moreover, and on the basis of the results presented in Table 3-14, it is clear from the CSI values that for the 5μg/m^3 maximum Benzene limit value the proposed methodology provides a better event forecasting in comparison to the 10μg/m^3 limit value. The configurations numbered as 5 (with Dataset 06-07) and 7 (with Dataset 04-05) result in the highest CSI, highest correct-alarm rate and lower false-alarm rate for the 5μg/m^3 and 10μg/m^3 maximum Benzene limit value respectively.

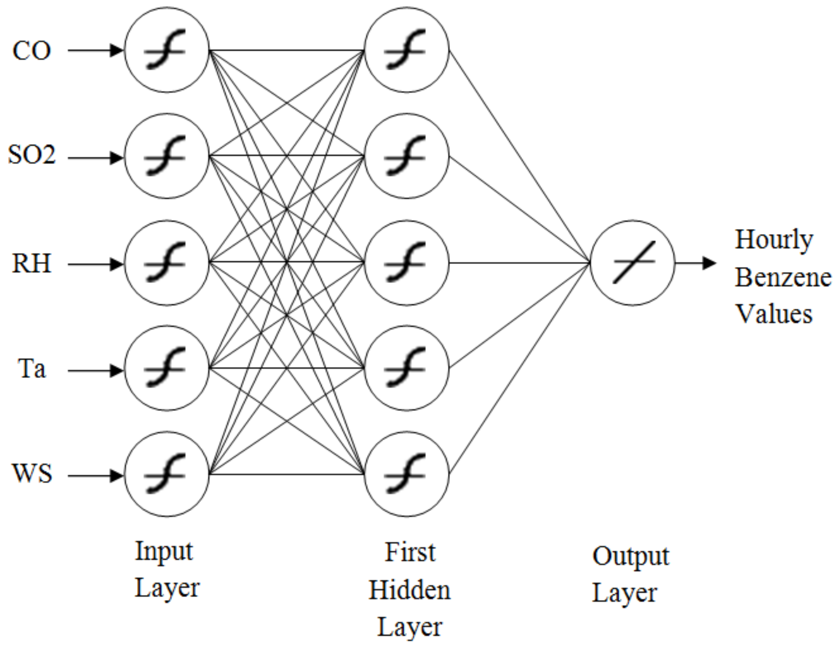


Figure 3-4: An example of the first configuration of the neural network of hourly Benzene concentrations

Table 3-11: Performance results table of hourly Benzene concentrations for three different datasets for the Patisia station by using Artificial Neural Networks

Config. Number	Number of neurons, per Hidden Layer			Data (04-05)		Data (04-05) without CV		Data (06-07)		Data (06-07) without CV		All Data (04-07)	
	1	**2**	**3**	**R^2**	**IA**	**R^2**	**IA**	**R^2**	**IA**	**R^2**	**IA**	**R^2**	**IA**
1	5	-	-	0.68	0.90	0.58	0.76	0.75	0.90	0.61	0.72	0.42	0.70
2	5	5	-	0.70	0.90	0.60	0.77	0.74	0.90	0.64	0.76	0.49	0.75
3	5	5	5	0.59	0.84	0.60	0.77	0.66	0.85	0.63	0.76	0.41	0.68
4	10	-	-	0.70	0.90	0.60	0.77	0.76	0.91	0.63	0.76	0.60	0.83
5	**10**	**10**	**-**	0.71	0.91	**0.61**	**0.78**	**0.78**	**0.92**	0.62	0.75	0.65	0.88
6	10	10	10	0.71	0.91	0.61	0.78	0.77	0.92	0.64	0.76	0.66	0.88
7	**15**	**-**	**-**	**0.72**	**0.91**	0.62	0.78	0.77	0.92	0.61	0.73	0.64	0.87
8	**15**	**15**	**-**	0.71	0.91	0.60	0.76	0.76	0.91	**0.64**	**0.77**	**0.67**	**0.89**
9	15	15	15	0.70	0.90	0.61	0.78	0.77	0.92	0.63	0.76	0.67	0.89
10	20	-	-	0.72	0.91	0.62	0.78	0.77	0.92	0.61	0.72	0.65	0.88
11	20	20	-	0.72	0.91	0.61	0.78	0.78	0.92	0.62	0.75	0.67	0.89
12	20	20	20	0.71	0.91	0.61	0.78	0.77	0.92	0.63	0.77	0.67	0.89

Table 3-12: Performance results table of hourly Benzene concentrations for three different datasets for the Patisia station by using Linear Regression Models

Data (04-05)		Data (06-07)		Data (04-07)	
R^2	IA	R^2	IA	R^2	IA
0.68	0.90	0.69	0.89	0.71	**0.91**

Table 3-13: Results by using statistical indicators to evaluate the best Artificial Neural Network model configurations on the basis of their performance on three datasets for Patisia station and for Benzene.

Patisia Station (Benzene)	Config. 7, Dataset (04-05)	Config. 5, Dataset (06-07)	Config. 8, Dataset (04-07)
Observed mean	6.699	5.442	5.262
Predicted mean	6.799	5.261	5.286
Observed standard deviation	4.597	3.696	3.569
Predicted standard deviation	3.939	2.830	2.779
Correlation of determination (R^2)	0.723	0.781	0.666
Index of agreement (IA)	0.913	**0.919**	0.886
Normalised mean difference (NMD)	0.020	0.033	0.015
Root mean square error (RMSE)	2.423	1.799	2.069
RMSE systematic	0.094	0.068	0.056
RMSE unsystematic	0.174	0.140	0.096
Mean absolute error (MAE)	1.605	1.185	1.417
Mean bias error (MBE)	-0.01	0.180	-0.024

Table 3-14: Artificial Neural Network model forecasting performance for the prediction of two different exceedance-levels events for Patisia.

Patisia Station	Benzene exceedance value: 5 μg/m^3			Benzene exceedance value: 10 μg/m^3		
	Config. 7, Dataset (04-05)	Config. 5, Dataset (06-07)	Config. 8, Dataset (04-07)	Config. 7, Dataset (04-05)	Config. 5, Dataset (06-07)	Config. 8, Dataset (04-07)
Critical Success Index (CSI)	0.78	**0.80**	0.73	**0.60**	0.44	0.36
Number of observed alarms	379.4	333.7	611.5	130	71.5	126.8
Number of predicted alarms	391.5	326.7	625.9	128	46.9	92.5
Number of hours with correct predicted alarms (A)	336.9	293.7	520.5	97	36.2	58.6
Correct Alarm Rate (%)	86%	**90%**	83.2%	76%	**77.7%**	63.5%
Number of hours with false predicted alarms (C)	54.6	33	105.4	31	10.7	33.9
False Alarm Rate (%)	14%	**10%**	16.8%	24%	**22.3%**	36.5%

3.7 Forecasting Daily 8-HRA Concentration Values of Ozone

The initial configuration consisted of 22 input parameters, leading to 22 neurons in the input layer and one neuron in the output layer per ANN model since the objective was to forecast one characteristic (the highest daily 8-HRA of Ozone for Patisia and Liosia stations). A total of five network configurations were implemented and tested during the training phase in order to avoid local minima. Table 3-15 presents the forecasting performance results by using ANNs for the different model configurations, for the different datasets and for the two different monitoring stations in comparison when CV was not used. The number of neurons per hidden layer was not further increased, because of the high delay during the training phase.

Table 3-16 presents the forecasting performance of the Linear Regression Models while Table 3-17 presents the performance results of the best forecasting models for the three different datasets of Patisia and Liosia for daily 8-HRA values of Ozone.

From Table 3-15 it is clear that the configuration with one hidden layer had the best results for all monitoring stations (both locations) and for all datasets, when CV was used for the forecasting of the 8-HRA of Ozone concentrations. From the same table, it is evident that the forecasting performance for daily 8-HRA values of Ozone is slightly lower when using data for only one year (2007, Dataset P4 and L2). This indicates that the applied methodology requires more than one year of data to produce better results. However, it should be noted that the difference between the best forecasting performances of the two datasets of both monitoring stations (Patisia and Liosia) are very low in terms of the IA (approx. 0.09 and 0.08 respectively). This suggests that even with one year of data, such models may be developed and effectively applied.

From Table 3-17 it is evident that Linear Regression models provide better forecasting ability in comparison to ANNs, in two out of three datasets in both monitoring station locations. This confirms the conclusion of Makridakis and Hibon (2000) that simple methods developed by practicing forecasters do as well or in many cases better than computationally sophisticated models. In addition, the results presented in Table 3-15 suggest that the forecasting performance achieved by using CV is better in comparison (by using the IA) with the results without using CV for both stations and for the Dataset P3 and L1, while the situation is the opposite for the Dataset P4 and L2. According to Table

3-17 at Patisia station the Linear Regression provides the best forecasting performance for 8-HRA of Ozone concentrations (IA=0.94) when all data were used (Dataset P3&4). On the other hand, at Liosia station the configuration numbered as 1 of the ANN models provides the best forecasting performance when all data were used (Dataset L3&4) even when Linear Regression provides better forecasting results in the separate datasets (Dataset L1). From Table 3-17 it is clear that the mean value of the observations (daily 8-HRA of Ozone) reveal the different nature of the two monitoring stations: Patisia is a traffic related station, heavily influenced by nitrogen oxide emissions, and thus with low mean ozone concentration values; Liosia is a station that well represents the "nomadic" nature of ozone, not heavily influenced by traffic, thus posing higher (than in Patisia) mean O_3 concentration values.

Table 3-15: Performance results table of daily 8-hour running average Ozone concentrations for two different monitoring stations and datasets by using Artificial Neural Networks.

Config. per Station		**Hidden Layers**			**Dataset P3 / Dataset L1**		**Dataset P3 / Dataset L1 without CV**		**Dataset P4 / Dataset L2**		**Dataset P4 / Dataset L2 without CV**		**Dataset P3&4 / Dataset L1&2**	
		1	**2**	**3**	**R^2**	**IA**	**R^2**	**IA**	**R^2**	**IA**	**R^2**	**IA**	**R^2**	**IA**
Patisia Station	**1**	**22**	**-**	**-**	**0.77**	**0.93**	0.73	0.84	**0.59**	**0.84**	**0.65**	**0.88**	**0.77**	**0.93**
	2	22	10	-	0.58	0.83	0.73	0.84	0.56	0.77	0.65	0.84	0.43	0.73
	3	22	10	5	0.41	0.74	0.76	0.83	0.29	0.56	0.68	0.86	0.30	0.64
	4	**22**	**22**	**-**	0.52	0.81	**0.72**	**0.88**	0.47	0.79	0.64	0.85	0.48	0.79
	5	22	22	22	0.50	0.82	0.70	0.86	0.39	0.71	0.67	0.85	0.40	0.75
Liosia Station	**1**	**22**	**-**	**-**	**0.69**	**0.90**	0.73	0.87	**0.61**	**0.82**	0.53	0.83	**0.59**	**0.86**
	2	22	10	-	0.52	0.81	0.77	0.88	0.43	0.73	0.59	0.84	0.22	0.61
	3	**22**	**10**	**5**	0.19	0.58	0.74	0.85	0.21	0.55	**0.61**	**0.85**	0.20	0.58
	4	22	22	-	0.51	0.82	0.72	0.85	0.49	0.77	0.57	0.84	0.40	0.75
	5	**22**	**22**	**22**	0.40	0.76	**0.77**	**0.89**	0.32	0.67	0.51	0.81	0.26	0.65

Table 3-16: Performance results table of daily 8-hour running average Ozone concentrations for two different monitoring stations and the available datasets, by using Linear Regression.

Station	**Dataset P3 / Dataset L1**		**Dataset P4 / Dataset L2**		**Dataset P3&4 / Dataset L1&2**	
	R^2	**IA**	**R^2**	**IA**	**R^2**	**IA**
Patisia	0.71	0.91	0.82	0.94	0.78	0.94
Liosia	0.73	0.92	0.55	0.84	0.41	0.79

Table 3-17: Results by using statistical indicators to present the best forecasting models on the basis of their performance on the three datasets for Patisia and Liosia stations.

8-HRA of Ozone	Patisia Station			Liosia Station		
	Config. 1 Dataset P3	LR Dataset P4	LR Dataset P3&4	LR Dataset L1	LR Dataset L2	Config. 1 Dataset L3&4
Observed mean	20.897	35.142	21.828	67.209	71.260	68.123
Predicted mean	20.095	35.819	22.279	68.184	77.980	72.344
Observed standard deviation	18.740	25.865	21.253	36.268	31.057	35.089
Predicted standard deviation	16.498	22.349	20.613	31.998	27.024	30.067
Correlation of determination (R^2)	0.766	0.817	0.785	0.729	0.551	0.581
Index of agreement (IA)	0.927	**0.944**	0.940	**0.919**	0.844	0.857
Normalised mean difference (NMD)	0.050	0.019	0.020	0.014	0.086	0.062
Root mean square error (RMSE)	9.018	10.990	9.966	18.820	21.943	22.968
RMSE systematic	1.375	1.832	0.898	1.711	3.657	2.595
RMSE unsystematic	2.390	4.558	1.735	4.280	6.639	4.832
Mean absolute error (MAE)	6.518	8.777	7.609	13.882	18.488	17.008
Mean bias error (MBE)	0.801	-0.677	-0.450	-0.974	-6.720	-4.220

3.8 Forecasting Hourly and Daily CAQI

3.8.1 Forecasting CAQI Values (Numerical) with the Aid of Regression Models

Linear regression (LR) models were used as the reference method for the development of the CAQI forecasting models. Lagged values of the CAQI (a total of up to 10 values, Dataset 1) were used in order to forecast hourly and daily CAQI numerical values. Results indicate that the hourly CAQI numerical values for Agia Sofia can be forecasted by a LR model when employing only one lagged hourly value (IA = 0.92). The daily CAQI numerical values were not as successfully forecasted (IA = 0.76). The Cohen’s Kappa Index is not applied in this case; as such an index makes sense only in categorical forecasts (nominal values). It should also be noted that the numerical values forecasted here ranged between 0 and up to over 100, according to Figure IV-1.

3.8.2 Calculate CAQI Levels based on Forecasted CAQI Values

Based on the forecasted CAQI values (of the previous section), the CAQI levels (nominal values) were calculated and evaluated. Tables 3-18 and 3-19 shows the observed and predicted frequencies of the hourly and daily CAQI levels for Agia Sofia station, respectively. For example, 74.74% of the low observed hourly CAQI cases (i.e. 6625 low observed cases from a total of 8864) were predicted as low-level cases (correct predictions) while 1.18% were predicted as high-level cases (false predictions). Both hourly and daily CAQI levels are not forecasted in a satisfactory manner with the aid of LR models, one possible reason being the mapping of the CAQI between arithmetic and nominal values. This is evident from Tables 3-18 and 3-19, where the percentages of the correct predictions per level (bold values) are low, especially in the prediction of daily CAQI levels. This can also be seen via their lower IA (reaching 0.5 for daily values) as well as via their low Cohen's Kappa Index (Table 3-20), which calculated on the basis of the numerical values of the contingency Tables 3-18 and 3-19 (as described in Appendix III).

Table 3-18: Observed and predicted frequencies of Hourly Common Air Quality Index levels for Agia Sofia (by using linear regression models)

Observed Hourly CAQI Levels	**Predicted AQI Levels**					
	Very Low	**Low**	**Medium**	**High**	**Very High**	**Total Observed Cases**
Very low	**1186**	579	20	13	18	1816 (100%)
	65,31%	31,88%	1,10%	0,72%	0,99%	
Low	633	**6625**	1462	105	39	8864 (100%)
	7,14%	**74,74%**	16,49%	1,18%	0,44%	
Medium	18	1844	**4906**	987	186	7941 (100%)
	0,23%	23,22%	**61,78%**	12,43%	2,34%	
High	7	166	1354	**1895**	614	4036 (100%)
	0,17%	4,11%	33,55%	**46,95%**	15,21%	
Very High	1	47	244	833	**2496**	3621 (100%)
	0,03%	1,30%	6,74%	23,00%	**68,93%**	

Table 3-19: Observed and predicted frequencies of Daily Common Air Quality Index levels for Agia Sofia (by using linear regression models)

Observed Daily CAQI Levels	Predicted AQI Levels					Total Observed Cases
	Very Low	Low	Medium	High	Very High	
Very low	**4**	6	7	0	0	17 (100%)
	23,53%	35,29%	41,18%	0,00%	0,00%	
Low	18	**197**	74	19	5	313 (100%)
	5,75%	**62,94%**	23,64%	6,07%	1,60%	
Medium	3	155	**244**	45	19	466 (100%)
	0,64%	33,26%	**52,36%**	9,66%	4,08%	
High	0	22	82	**44**	23	171 (100%)
	0,00%	12,87%	47,95%	**25,73%**	13,45%	
Very High	0	5	31	38	**53**	127 (100%)
	0,00%	3,94%	24,41%	29,92%	**41,73%**	

Table 3-20: Results of the nominal hourly and daily Common Air Quality Index level predictions using linear regression

	Index of agreement (IA)	Cohen's Kappa Index
Hourly CAQI levels	0.65	0.53
Daily CAQI levels	0.50	0.28

3.8.3 Calculate Daily CAQI Levels based on Forecasted Numerical 24-Hourly values

The forecasting of the daily CAQI levels on the basis of the forecasted 24 hourly values of the next day was investigated because of the fact that the forecast of the numerical hourly CAQI values reached an IA equal to 0.92. For this purpose, the hourly CAQI values of the previous day were used in the following manner: for the forecasting of the first hourly CAQI value for the next day, the 24 hourly values of CAQI of the previous day (observed) were used. For the forecasting of the second hourly CAQI value of the next day, the value of the previous hour (forecasted), as well as the 23 hourly values of the previous day (observed) were used, and so forth. Via this procedure, the hourly CAQI values were forecasted and then the daily average CAQI value was calculated, based on these values.

Table 3-21 presents the comparison of daily CAQI forecasting performance using LR models at the station of Agia Sofia, by using lagged observed numerical values and by (equally) using 24 hourly forecasted values. The hourly CAQI numerical values for the Agia Sofia station achieved an IA = 0.70, on the basis of LR models. The performance was decreased in comparison to the performance of the model that uses ten observed lagged values, which reached an IA = 0.92. On the other hand, the performance concerning the forecasting of daily CAQI levels was slightly increased when using 24 forecasted lagged values with an IA=0.51 (in comparison to 0.50). This slight difference in the performance may be attributed to the use of a higher percentage of observations for the forecasting of the hourly CAQI values for the first hours of the next day, thus allowing information of the previous day to "penetrate" the target day of the forecast.

Table 3-21: The Index of agreement for the forecasted numerical daily Common Air Quality Index using linear regression

Forecast by using:	Index of agreement (IA)	
	Hourly CAQI Values	Daily CAQI levels
10 observed lagged values	0.92	0.50
24 predicted lagged values	0.70	0.51

3.8.4 Calculate Daily CAQI Levels based on 24-h Forecasted values with Weighting Factors

In order to further investigate the influence of input values to the performance of the forecasting models, additional daily CAQIs were calculated by using weighted factors. These weighted factors dictated that the first forecasted values of each day (values at midnight, 1 a.m., 2 a.m., etc.) will contribute more than the last forecasted values. Table 3-22 presents one of the weighting methods tested, named "Factor4", which provides the best forecasting performance, in comparison to others. When using the hourly forecasted CAQI values to calculate the daily CAQI, the results are slightly better (IA= 0.5105, i.e. an improvement of 1.51%) than those when forecasting directly the daily CAQI.

Table 3-22: "Factor4" weighting of hourly values at Agia Sofia.

Hourly Values	Contribution to daily value
1-6	46.25%
7-12	28.75%
13-18	18.75%
19-24	6.25%

3.8.5 Forecasting CAQI Levels with the aid of ANNs and DTs

In the first step of the investigation experiments (when CV was not used), all ANN models had one hidden layer, and the number of neurons (of the hidden layer) was the same as the number of the input parameters. The basic experiments (indicated as models) and the different datasets that were used for each model are the ones presented in points 1 to 4 listed below. The “S” denotes the step number of the investigation experiment.

1. Hourly and daily CAQI numerical values (which range from 0 to 100+) were forecasted using ANNs, in order to calculate the hourly and daily CAQI levels (according to Figure IV-1). Tables 3-23, 3-26 and 3-27 present the forecasting results for the different datasets for daily and hourly CAQI levels respectively indicated as Model S.3 (when Dataset 1 was used), Model S.4 (when Dataset 2 was used) and Model S.5 (when Dataset 3 was used). The values at columns H.L.x in Tables 3-26 and 3-27 indicate the number of neurons in the hidden layer x.
2. Hourly and daily NO_2 and PM_{10} numerical values were forecasted using ANNs, in order to calculate the hourly and daily CAQI levels by using the formulas of Table IV-1. As input to the ANN, Dataset 2 was used. Tables 3-23, 3-26 and 3-27 present the forecasting results of different datasets for daily and hourly CAQI levels (indicated as Model S.6).
3. Hourly and daily CAQI levels were forecasted using ANNs. In order to match the forecasted numerical values to the CAQI levels (nominal values), the logical expressions of Table 3-8 were used. The output of the ANNs are numerical values which (in this case) range between -1 and +1. Tables 3-23, 3-26 and 3-27 present the forecasting results of different datasets for daily and hourly CAQI levels indicated as Model S.7 (when Dataset 1 was used), Model S.8 (when Dataset 2 was used) and Model S.9 (when Dataset 3 was used).
4. The hourly and daily CAQI levels were forecasted with the aid of DTs. Table 3-28 presents the forecasting results of different datasets for daily and hourly CAQI levels, indicated as Model S.10 (when Dataset 1 was used), Model S.11 when Dataset 2 was used) and Model S.12 (when Dataset 3 was used).

Table 3-23 presents the overall best results of the first step of the investigation experiments. On the basis of these findings, the forecasting performances of each daily and hourly model that were used can be evaluated by using the “Percentage Agreement” (PA) and the Cohen's Kappa Index (κ) (see Appendix III). The best forecasting performance of daily CAQI Levels (PA=61% and κ=0.43), was achieved when Model 1.9 (*Forecast Directly the CAQI Levels with Dataset 3*) was used with the aid of ANNs, with an increase of forecasting performance of 10% PA and 12% κ compared to the best forecasting performance when LR models were used, and with an increase of 4% PA and 7% κ compared to the best forecasting performance when DTs were used. The best forecasting performance of hourly CAQI Levels (PA=65% and κ=0.53), was achieved when LR models were used, with an increase of 5% PA and 8% κ compared to the best forecasting performance when ANNs were used, and with an increase of 2% PA and κ compared to the best forecasting performance when DTs were used.

Table 3-23: Comparison of the best forecasting performances of the forecasting models of the first step of this work for Daily and Hourly Common Air Quality Index Levels

Model	**Sub Model**	**Daily CAQI Levels**		**Hourly CAQI Levels**	
		Percentage Agreement	**Cohen's Kappa**	**Percentage Agreement**	**Cohen's Kappa**
1.1	-	49.54%	0.28	**65.10%**	**0.53**
1.2	Forecasted CAQI - Factor4	51.05%	0.31	-	-
1.3	2 lagged values	45.00%	0.14	57.00%	0.41
	8 lagged values	49.00%	0.22	55.00%	0.39
1.4	-	53.00%	0.30	46.00%	0.25
1.5	1 lagged value	58.00%	0.37	57.00%	0.41
	4 lagged values	55.00%	0.32	60.00%	0.45
1.6	-	53.00%	0.30	44.00%	0.23
1.7	1 lagged value	41.00%	0.11	54.00%	0.37
	10 lagged values	52.00%	0.28	53.00%	0.34
1.8	-	54.00%	0.32	46.00%	0.25
1.9	1 lagged value	60.00%	0.41	55.00%	0.37
	2 lagged values	**61.00%**	**0.43**	54.00%	0.37
1.10	1 lagged value	50.00%	0.23	63.00%	0.51
1.11	-	54,00%	0,32	46,00%	0,26
1.12	4 lagged values	57,00%	0,36	62,00%	0,47
	5 lagged values	57,00%	0,36	62,00%	0,48

Model Details:

1.1. Calculate the CAQI Levels from forecasted Daily/Hourly CAQI values by using regression models
1.2. Calculate the CAQI Levels from forecasted Hourly CAQI values by using regression models
1.3. Calculate the CAQI Levels by forecasting the CAQI values using ANNs, with Dataset 1
1.4. Calculate the CAQI Levels by forecasting the CAQI values using ANNs, with Dataset 2
1.5. Calculate the CAQI Levels by forecasting the CAQI values using ANNs, with Dataset 3
1.6. Calculate the CAQI Levels by forecasting NO2 and PM10 using ANNs, with Dataset 2.
1.7. Forecast Directly the CAQI Levels using ANNs, with Dataset 1
1.8. Forecast Directly the CAQI Levels using ANNs, with Dataset 2
1.9. Forecast Directly the CAQI Levels using ANNs, with Dataset 3
1.10. Forecast Directly the CAQI Levels using Decision Trees, with Dataset 1
1.11. Forecast Directly the CAQI Levels using Decision Trees, with Dataset 2
1.12. Forecast Directly the CAQI Levels by using Decision Trees, with Dataset 3

The investigation experiments of the first step were repeated in order to evaluate the use of a) CV, and b) sensitivity analysis. The 10-fold cross-validation was used in ANN and DT models in order to measure the predictive performance of the models, as described in previous section (3.5). In addition, the one-way sensitivity analysis was used in order to examine different: a) ANN model architectures and, b) numbers of decision trees grown independently.

In order to perform one-way sensitivity analysis on the ANN model architectures, two important topology parameters were examined: the increase of the hidden layers and the increase of the neurons per layer. Two different empirical methods were used in order to generate the different architectures (neurons per layer), having as a “starting point” the number of the input parameters (*N*-Input). The number of the hidden layers was used as one of the parameters of the ANN architecture, and the number of neurons per hidden layer as an additional parameter. Thus, in Method 1 (Equation (3-1)), the number of neurons was the same in all hidden layers. In Method 2 (Equation (3-2)), the number of neurons was different for each hidden layer, and more specifically it increased as the number of hidden layers also increased. Both methods were designed in order to generate multiple architectures, where they will be different in one parameter at a time (in order to perform one-way sensitivity analysis).

$$\text{Method 1:}\quad \text{Number of neurons (same for all hidden layers)} = \text{N-Input} + ((\text{N-Input }/2) \times \text{Neuron Multiplier}) \tag{3-1}$$

$$\text{Method 2:}\quad \text{Number of neurons in every hidden layer} = \text{N-Input} + ((\text{N-Input }/2) \times \text{Hidden Layer Number} \times \text{Neuron Multiplier}) \tag{3-2}$$

The “Neuron Multiplier” was an additional parameter that was used in order to increase the number of neurons and to thus generate different ANN architectures. The “Hidden Layer Number” was used in order to increase the number of neurons per hidden layer. For example, the number of neurons of the third hidden layer when Method 2 is used with five (5) input parameters and with Neuron Multiplier equalled two (2), should be: 5 + ((5/2) × 3 × 2) = 20 neurons. It should be noted that the calculated number of neurons (by using Method 1 and 2) was rounded up.

By combining the results of the two methods, with maximum three hidden layers and with a Neuron Multiplier ranging from 2 to 4 (increased by one), 15 different ANN architectures per input parameter set (N-input) were generated in order to perform the sensitivity analysis. The majority of these models can be characterized as deep neural networks because their architecture includes two or more hidden layers (Nielsen, 2015). Additional hidden layers would increase significantly the requirements in computational time and physical memory of the machine (RAM), and thus, were not used. Deep learning techniques that are based on stochastic gradient descent and backpropagation can use much deeper (e.g. 5 to 10 hidden layers) networks to be trained, but were not examined in this work. (Nielsen, 2015) (Schmidhuber, 2015). Table 3-24 summarizes the construction parameters used for the 15 different architectures, while Table 3-25 presents an example of the different architectures generated on the basis of 5 input parameters, with maximum three hidden layers and with a neuron multiplier ranging from 2 to 4 (increased by one). The hidden layer columns in Table 3-25 report the number of neurons per layer.

In the case of DT models, four different numbers of trees were used (50, 100, 150 and 200) in order to perform one-way sensitivity analysis. The initial number of trees (50) was chosen on the basis of previously published studies (Vens & Costa, 2011). An increase in the number of trees was not investigated, due to the requirements in computational time and physical memory of the machine (RAM).

As a result of the second step of this work, a total 1118 different models (i.e. different scenarios of ANN model architectures and numbers of decision trees) were developed and trained for daily and hourly data (559 for each one respectively). Table 3-29 presents in detail the 559 models that were tested for daily and hourly data (thus leading to a total of 1118 models). It should be noted that the computational (CPU) time required to train the aforementioned 1118 models was approximately 64 days, while the investigation experiments required at least 4GB of RAM in order to be completed (on a machine with Intel(R) i3 Core (2.1 GHz) processor and 4GB of RAM).

Table 3-24: Construction of the 15 different architectures

Parameter	From	To
Number of Hidden Layers	1	3
Neuron Multiplier	2	4
Methods	1	2

Table 3-25: An example of the 15 different architectures with 5-input (parameters)

Architecture ID	N-Input(s)	Neuron Multiplier	Hidden Layer 1	Hidden Layer 2	Hidden Layer 3	Generated by Method
1	5	2	10	-	-	1 and 2
2			10	10	-	1
3			10	15	-	2
4			10	10	10	1
5			10	15	20	2
6		3	13	-	-	1 and 2
7			13	13	-	1
8			13	20	-	2
9			13	13	13	1
10			13	20	28	2
11		4	15	-	-	1 and 2
12			15	15	-	1
13			15	25	-	2
14			15	15	15	1
15			15	25	35	2

The forecasting performance of the daily and hourly NO_2 and PM_{10} numerical values (15 developments of Model 6, Table 3-29), were in average 0.44 R^2 and 0.78 IA for daily numerical values and 0.56 R^2 and 0.84 IA for hourly numerical values.

The daily and hourly CAQI levels were calculated from the forecasted NO_2 and PM_{10} numerical values. The models with the highest forecasting performance (in terms of PA and Cohen's Kappa) of the CAQI levels (daily and hourly) of Model 6 are presented in Tables 3-26 and 3-27 as Model 2.6. The forecasting performances of the numerical values (NO_2 and PM_{10}) of these models were 0.41 R^2 and 0.77 IA for daily values and 0.57 R^2 and 0.85 IA for hourly values. In both cases of the best daily and hourly Model 2.6, the forecasting performance of the numerical values (R^2 and IA) is lower or near the average, which means that models with higher or equal forecasting performance (of the numerical values) exists, but resulting in a lower forecasting performance (PA and Cohen's Kappa) of the CAQI levels. Thus, a good forecasting performance of the NO_2 and PM_{10} values does not necessarily lead to good forecasting performance of the CAQI levels.

Table 3-30 presents the average performance of the ANN forecasting Models for daily and hourly CAQI levels and for each different ANN architecture. Each of these architectures has a unique ID and is characterized by a different parameter distinguishing it from other architectures. The results reported in Table 3-30 support the finding of Section 3.7 that simple ANN architectures (with one hidden layer) can achieve better forecasting performance (in this case for the daily CAQI levels). Contrary to this finding, the results for the hourly CAQI levels suggest that more complex architectures lead to better forecasting performances.

Table 3-31 presents the average performance of the DT Forecasting Models for daily and hourly CAQI levels for each different number of trees. Results indicate that an increment in the number of the trees will not affect considerably the forecasting performance, but it may improve the accuracy and the confidence in the forecasting results. It worthies mentioning that the best forecasting performance (Table 3-28) was achieved with 200 trees (the maximum number of trees being investigated) in the case of the daily CAQI levels and with 100 trees in the case of the hourly CAQI levels.

Table 3-26: Comparison of the best forecasting performance of the Artificial Neural Network forecasting models of the second step of this work for Daily Common Air Quality Index levels

Daily CAQI levels								
Model	**Number of Input parameters**	**Architecture ID**	**H.L.1**	**H.L.2**	**H.L.3**	**Architecture Generated by Method**	**PA**	**Cohen's Kappa**
2.3	1	1	2	-	-	1 and 2	0.50	0.23
2.4	**9**	**6**	**23**	**-**	**-**	**1 and 2**	**0.56**	**0.35**
2.5	10	1	20	-	-	1 and 2	0.48	0.22
2.6	9	5	18	27	36	2	0.54	0.33
2.7	3	8	8	12	-	2	0.50	0.26
2.8	9	1	18	-	-	1 and 2	0.55	0.36
2.9	11	6	28	-	-	1 and 2	0.48	0.23

Table 3-27: Comparison of the best forecasting performance of the Artificial Neural Network forecasting models of the second step of this work for Hourly Common Air Quality Index levels

Hourly CAQI levels								
Basic Model	**Number of Input parameters**	**Architecture ID**	**H.L.1**	**H.L.2**	**H.L.3**	**Architecture Generated by Method**	**PA**	**Cohen's Kappa**
2.3	**2**	**13**	**6**	**10**	**-**	**2**	**0.63**	**0.50**
2.4	9	7	23	23	-	1	0.50	0.31
2.5	12	1	24	-	-	1 and 2	0.62	0.49
2.6	9	4	18	18	18	1	0.50	0.31
2.7	1	10	3	4	6	2	0.62	0.50
2.8	9	2	18	18	-	1	0.50	0.32
2.9	12	1	24	-	-	1 and 2	0.62	0.49

Table 3-28: Comparison of the best forecasting performance of the Decision Trees forecasting models of the second step of this work for daily and hourly Common Air Quality Index levels

Basic Model	Daily CAQI levels				Hourly CAQI levels			
	Number of Trees	Number of Input parameters	PA	Cohen's Kappa	Number of Trees	Number of Inputs	PA	Cohen's Kappa
2.10	150	9	0.56	0.32	150	9	0.64	0.51
2.11	**200**	**9**	**0.69**	**0.51**	100	9	0.60	0.45
2.12	50	14	0.59	0.37	**100**	**12**	**0.65**	**0.53**

Table 3-29: The 559 models that were tested for daily and hourly data in the second part of this work

	Data	Used by	Number of Models Developed and Trained
15 ANN Architectures	Dataset 1 (1 to 10 lagged values)	Model 3	150
	Dataset 2	Model 4	15
	Dataset 3 (1 to 5 lagged values)	Model 5	75
	Dataset 2	Model 6	15
	Dataset 1 (1 to 10 lagged values)	Model 7	150
	Dataset 2	Model 8	15
	Dataset 3 (1 to 5 lagged values)	Model 9	75
4 different number of decision trees	Dataset 1 (1 to 10 lagged values)	Model 10	40
	Dataset 2	Model 11	20
	Dataset 3 (1 to 5 lagged values)	Model 12	4
		Total	**559**

Table 3-30: The average performance of the Artificial Neural Network Forecasting Models for daily and hourly Common Air Quality Index levels for each different ANN architecture

Architecture ID	Architecture Generated by Method:	Daily CAQI levels		Hourly CAQI levels	
		PA Average	Cohen's Kappa Average	PA Average	Cohen's Kappa Average
1	**1 and 2**	**0.46**	**0.20**	0.59	0.45
2	1	0.43	0.16	0.59	0.45
3	2	0.43	0.16	0.59	0.45
4	1	0.43	0.15	0.59	0.44
5	2	0.43	0.16	0.59	0.45
6	**1 and 2**	0.45	0.19	**0.60**	**0.46**
7	1	0.44	0.16	0.60	0.45
8	2	0.42	0.15	0.59	0.45
9	1	0.42	0.15	0.59	0.45
10	2	0.40	0.14	0.59	0.45
11	1 and 2	0.45	0.19	0.59	0.45
12	1	0.43	0.16	0.59	0.45
13	2	0.41	0.15	0.59	0.45
14	1	0.42	0.15	0.59	0.45
15	2	0.38	0.13	0.58	0.44

Method 1: Increase the number of Hidden Layers from the previous topology.
Method 2: Increase the Number of Neurons from the previous topology.
Method 1 and 2: Increase the Number of Neurons from the previous "1 and 2" Topology.

Table 3-31: The average performance of the Decision Trees forecasting models for daily and hourly Common Air Quality Index levels for each different number of trees

Number of Trees	Daily CAQI levels		Hourly CAQI levels	
	Percentage Agreement Average	Cohen's Kappa Average	Percentage Agreement Average	Cohen's Kappa Average
50	0.500	0.255	0.618	0.479
100	0.484	0.230	0.619	0.481
150	0.495	0.256	0.620	0.483
200	0.505	0.249	0.621	0.484

3.8.6 Comparing the Results of the Best CAQI Forecasting Methods

Table 3-32 presents detailed information for the best forecasting methods (of the two steps of the investigation experiments) when forecasting daily and hourly CAQI levels. In basic Model 1.9 (of Table 3-32), the CAQI levels (which are encoded to values that range from -1 to 1) were forecasted directly. In basic models 1.1, 2.3 and 2.4 (of the same Table) the CAQI levels are calculated from the forecasted CAQI values (which ranges from 0 to higher than 100). It is, therefore, evident that when CV was used at the model with the best forecasting performance for daily data (resulting from the first step of this work), the performance decreased from 61% PA and 43% κ (Model 1.9) to 44% PA and 18% κ (Model 1.9^{*1}). On the other hand, when CV was used in the model with the best forecasting performance for hourly data (Model 1.1, that resulted from the first step of this work) the performance did not change. The fact that the performance was decreased for the forecasting of the daily CAQI and not for the hourly one suggests that the results for the daily data of the first step of this work were not as accurate as expected. Thus, CV should be used at all times, in order to increase the credibility and validity of the forecasting models.

From Table 3-32 it can be seen that the models are very different. For example, basic Model 2.4 has an observed mean value of 65.32, because the CAQI levels were calculated from the forecasted CAQI values (which range from 0 to higher than 100 as shown in Figure IV-1 and in Equation IV-8). On the other hand, Model 1.9^{*1} has an Observed Mean value of 0.03, because (a) the CAQI levels (which are encoded to values that range from -1 to 1 as shown in Table 3-7) were directly forecasted, and (b) the tangent sigmoid transfer function was used.

Table 3-32: Detailed information for the best forecasting methods (of the two steps of the investigation experiments) when forecasting daily and hourly Common Air Quality Index levels

CAQI levels	Daily				Hourly			
Computational Intelligence Method	**ANN**	**DT**	**ANN**	**ANN**	**ANN**	**DT**	**LR**	**LR**
Basic Model	**2.4**	**2.11**	**1.9*1**	**1.9**	**2.3**	**2.12**	**1.1*1**	**1.1**
Observed mean	65,32	-	0,03	0,03	65,47	-	66,11	66,10
Predicted mean	65,61	-	0,05	0,00	65,66	-	64,87	63,47
Observed standard deviation	28,28	-	0,48	0,50	39,82	-	39,93	29,64
Predicted standard deviation	23,74	-	0,28	0,28	32,62	-	36,61	14,43
Normalized mean difference (NMD)	0,04	-	1,30	0,97	0,01	-	0,02	0,04
Root mean square error (RMSE)	22,22	-	0,47	0,33	21,93	-	20,45	24,54
RMSE systematic	3,08	-	0,06	0,01	0,64	-	0,40	0,76
RMSE unsystematic	5,24	-	0,03	0,01	1,20	-	0,87	1,14
Mean absolute error (MAE)	15,11	-	0,37	0,26	13,39	-	12,53	16,69
Mean absolute percentage error (MAPE)	0,25	-	Inf.	Inf.	0,23	-	0,20	0,27
Mean bias error (MBE)	-0,28	-	-0,03	0,03	-0,19	-	1,24	2,63
Coefficient of determination (R^2)	0,44	-	0,14	0,61	0,69	-	0,74	0,32
Index of agreement (IA)	0,79	-	0,58	0,81	0,90	-	0,92	0,65
Percentage Agreement	0,56	0,69	0,44	0,61	0,63	0,65	0,65	0,65
Cohen's Kappa	0,35	0,51	0,18	0,43	0,50	0,53	0,53	0,53

*1: *The basic model of the first step of this work, when CV was applied.*

3.9 Results and Discussion

For the AQ investigations in Athens, two regulated major pollutants were used, Benzene and Ozone. The forecasting of the former has to deal with the way that the event of interest (exceedance of a limit value) is defined while the forecasting of the latter deals with an 8-Hour Running Average (HRA) parameter as the basis for the definition of the event of interest. Both pollutants demonstrate differences in nature as well as in the way they should be handled. For this reason, alternative architecture and model setups concerning ANNs as the basis for the development of the forecasting capability as well as classical techniques such as Linear Regression (LR) were investigated. Overall, the methodology applied for the data analysis and for the forecasting of the selected pollutants was proven to be successful, leading to good forecasting results (Hourly Benzene: 0.70 - 0.92 IA and 0.41 – 0.78 R^2, and 8-HRA of Ozone: 0.55 – 0.94 IA and 0.19 – 0.90 R^2).

For the investigations in Thessaloniki, the CAQI was selected as the parameter of interest that needs to be forecasted (in an hourly and daily temporal horizon). On the basis of the results, the Agia Sofia station was chosen as the one for which forecasting models were developed and tested. The performance of a wide variety of ANNs, DTs, and statistical regression models were evaluated and inter-compared on the basis of the following scenarios:

a) The estimation of the CAQI levels by forecasting the CAQI values,
b) The estimation of the CAQI levels by forecasting individual pollutant levels participating in the calculation of the overall CAQI and,
c) The direct forecasting of the CAQI levels.

In order to investigate these scenarios, three different datasets were employed for daily and hourly data. For the various datasets and the different scenarios, 15 ANN architectures and 4 different DT model ensembles were used, indicated as different models. Overall, a total of 1118 different models were developed and trained for daily and hourly data (559 for each one respectively) in order to perform one-way sensitivity analysis.

Results show that when the daily CAQI value was forecasted on the basis of the forecasted hourly CAQI values, the performance was better in comparison to the one achieved via the direct forecast of the daily CAQI values. Moreover, if weighting factors were used in order to calculate the daily CAQI values, the forecasting performance improved further. It was also found that the DT models outperform the ANN models with respect to their forecasting ability of the CAQI. When hourly CAQI levels were forecasted the best performance was achieved by a DT model (Percentage Agreement = 65% and Cohen's Kappa Index = 53%, equal to the performance of a LR model). The presented results suggest that the performance of various ANN and DT models depends both on their internal structure and on the methods used for their training.

The sensitivity analysis results show that in order to predict daily CAQI levels, simple ANN architectures (with one hidden layer) should be used. On the other hand, in order to predict hourly CAQI levels, more complex architectures (with more than one hidden layer) should be employed. It is estimated that the reasons for these differences lie in the aggregated nature of the daily CAQI levels, which smoothes out any short term perturbations associated with the temporal profile of air pollutants, thus allowing for simpler ANN architectures to map any “knowledge” interwoven in the dataset under investigation that is related to the daily CAQI. In the case of DTs, sensitivity analysis results indicate that an increment in the number of the trees (from the initial 50 trees) will not affect considerably the forecasting performance. Even though the best forecasting performance in the case of the daily CAQI levels was achieved with the maximum number of trees being investigated (200 trees) and in the case of the hourly CAQI levels with 100 trees. It worth mentioning that in order to train the aforementioned 1118 models, approximately 64 days of a computational time were needed.

Using model ensembles obtained by a k-fold CV resulted in more accurate forecasts than using individual models. Thus, it became evident that CV should be used in order to increase the credibility and validity of the results. The models developed for the CAQI forecasting are able to predict the values of these parameters with acceptable accuracy (0.92 IA, 0.86 R^2 and 0.53 κ).

3.10 Summary

This chapter presents the experiments performed in an effort to investigate the process to develop CI and statistical methods to forecast several environmental parameters. The experiments consist of the investigation of forecasting environmental parameters for the two largest cities of Greece (Athens and Thessaloniki). For the two AQ monitoring stations of Athens, Benzene and Ozone were the pollutants selected towards episode forecasting, and for the four AQ monitoring stations of Thessaloniki, the CAQI was chosen as the parameter of interest. In addition, cross-validation and sensitivity analysis were used in order to perform a number of investigation experiments and test the performance of different model architectures (a total of 1118 different models), with the aim of evaluating their applicability and reliability concerning the parameters of interest. The sensitivity analysis results show that in order to predict daily CAQI levels, simple ANN architectures (with one hidden layer) should be used. In contrary, in order to predict hourly CAQI levels, more complex architectures (with more than one hidden layer) should be employed. In the case of DTs, sensitivity analysis results indicate that an increment in the number of the trees (from the initial 50 trees) will not affect considerably the forecasting performance. The time required to train the aforementioned 1118 models, was approximately 64 days of computational time. The overall results show that the forecasting accuracy of the developed data-driven models is at the highest level in comparison to the literature, thus, fulfilling the first research question of this work (see Section 1.2). In addition, the described semi-automatic procedure to perform forecasting via data-driven models can be generalized to other locations and is expected to be useful for the implementation of operational forecasting systems for environmental parameters.

This chapter highlights the needs in air quality forecasting by using data-driven models (listed below), which reflects the second and third research questions. The next chapters describe how these needs were addressed in the frame of this work.

a) A more comprehensive forecasting verification method, which combine forecasting performance measures in order to be used in an automated operational forecasting system (Chapter 4).

b) An automated methodology to optimize the whole AQ forecasting chain, in order to find one of best forecasting models more easily (Chapter 5). The investigation to find one of the best forecasting models is depending on the selected pollutants, locations and data, thus it is clear that this process will have to be repeated many times. In addition, it was evident from the investigation experiments that even though sensitivity analysis is important to investigate the performance of different model architectures/parameters, the computational time required was approximately 64 days, thus it becomes difficult to be used many times.

Chapter 4

New Forecasting Performance Indices

The performance of any data-driven forecasting model (like the ones used for air quality) is commonly evaluated with the aid of several forecasting performance measures (statistical indices). On this basis, models can be compared and the best model selected. As there is no formal procedure to follow, this means that for the same model results, different researchers may select different model(s) as the best in terms of forecasting, depending on (a) the employed statistical indices and (b) the criteria used for interpreting the values of these indices. In an effort to overcome this problem, two new forecasting performance indices were developed (denoted as FPI_m and FPI_s for Mamdani-type and Sugeno-type FIS respectively) to provide a combination of several measures (which define the forecast quality by different scalar attributes), that increases the confidence in the estimation of the forecasting performance and standardize the interpretation.

4.1 Statistical Measures Selection

To address the second research question, the best ways of evaluating the forecasting performance of the data-driven models on an operational basis must be found. In addition, the results of the investigation experiments highlighted the need for a more comprehensive forecasting verification method, which combine forecasting performance measures in order to be used in an automated operational forecasting system. For that reasons, this chapter details the research undertaken to improve forecasting verification through the development and application of new indices, based on an analysis of existing indices and their characteristics.

The current work concentrates on continuous variables forecasting and, more specifically, on the forecasting of the concentration levels of atmospheric pollutants in the urban atmosphere. Thus, categorical verification indices were not examined. Twenty-four (24) statistical measures commonly used to evaluate the performance of models that produce forecasts of continuous (numerical) variables were selected. In order to identify the characteristics of each index, a literature review was undertaken, resulting in Table 4-1 which summarizes the most frequently used indices and their main characteristics and disadvantages. Details of each statistical measure can be found in Appendix II.

The next step was to select the suitable indices and create the new indices based on the aforementioned analysis. The "raw material" of this synthesis includes characteristics that cannot be described merely with the aid of straightforward algebraic equations, but require the combination and synthesis of existing index characteristics. For this reason, a fuzzy logic approach was chosen as the most appropriate calculation methodology.

Table 4-1: Summarize description and disadvantage(s) or each measure

Measure (abbreviation)	**Short Description**	**Disadvantage(s)**	**Authors/Reference**
Bias	Indicates the average direction of error	• It provides no measure of the error variance.	Armstrong (1985)
Normalized Bias (NB)	A measure of the over or under-prediction of a variable.	• Possible division by zero. • Cancel each other out.	
Mean Fractional Bias (MBF)	Represents the difference between the forecasted average and the actual average.	• Possible division by zero. • Cancel each other out. • The predicted concentration is found in both the numerator and denominator	Ying, et al. (2007)
Mean Percentage Error (MPE)	Average of percentage errors.	• Cancel each other out. • Possible division by zero.	
Mean Absolute Error (MAE)	Average of the magnitude of the errors.	• Sensitive to outlier errors	
Mean Absolute Percentage Error (MAPE)	Mean Absolute Percent Error	• Possible division by zero. • When the predicted and observed values are the same (having a perfect fit), MAPE is zero.	

Symmetric Mean Absolute Percentage Error (sMAPE)	Symmetric Mean Absolute Percentage Error	• Involve division by a number close to zero	Armstrong (1985), Makridakis & Hibon (2000), Andrawis & Atiya (2009)
Mean Squared Error (MSE)	Average of the squared of the errors.	• It is sensitive to large errors, to large variance of errors and on outlier errors	Jolliffe & Stephenson (2012)
Normalized Mean Squared Error (NMSE)	Estimator of the overall deviations between forecasted and actual values.	• Sensitive to extreme values.	
Root Mean Squared Error (RMSE)	Indicates the magnitude of the error and retains the variable's unit.	• Sensitive to large errors, to large variance of errors and on outlier errors	Willmott (1981)
Linear Correlation Coefficient (r)	Measures the strength and the direction of a linear relationship between two variables.	• Measures the linear relationship. Therefore, a correlation of 0 does not mean zero relationship between two variables.	
Coefficient of Determination (r^2)	Measures the percent of the data that is the closest to the line of best fit.	• Based on the linear fit. • Sensitive to extreme values.	
Spearman's rank correlation coefficient (r_s)	Measure of statistical dependence between two variables.	• Simply places the values in numerical order; it pays no regard to the magnitude of the differences between the values.	Thornes (2006)
Coefficient of Efficiency (E)	Coefficient of Efficiency	• Sensitive to extreme values.	Nash & Sutcliffe, 1970 (1970)
Index of Agreement (d)	Measures the degree to which a model's predictions are error free.	• Large errors are squared, and thus the influence on the sum-of-squared errors is over-weighted.	Willmott (1981)
Index of Agreement (d_1)	Sums of the absolute values of the errors.	• The overall range of d_1 remained somewhat narrow to resolve adequately the great variety of ways that F can differ from A.	Willmott, et al. (1985)
Index of Agreement (d_r)	Sum of the magnitudes of the differences between the model-predicted and observed deviations about the observed mean relative to the sum of the magnitudes of the perfect model and observed deviations about the observed mean.		Willmott, et al. (2011)
Legates and McCabe's (E_1)	Coefficient of efficiency.		Legates & McCabe (1999)
Legates and McCabe's (E'_1)	Coefficient of efficiency (baseline adjusted).	• Can be used for special cases when season or another time period is available to provide a more appropriate baseline	Legates & McCabe (1999)

Berry and Mielke's ($\mathfrak{R}$)	For the analysis of a nominal independent variable with any number or combination of nominal, ordinal, or interval dependent variables.		Berry & Mielke (1992)
Watterson's (M)	Symmetric measure that converges linearly to 1 as errors diminish to zero.	• The upper and lower bounds it is not well defined.	Watterson (1996)
Factor of Exceedance (FOEX)	Measure to indicate if the forecasting results are over or under-predicted.	• Cannot distinguish an under-prediction than a perfect fit	Sokhi, et al. (2006)
Theil's Inequality Coefficient (U_1)	Measure of forecast quality	• It has a little or no value as a forecasting accuracy index.	Theil (1958)
Theil's Inequality Coefficient (U_2)	Measure of forecast quality		Thiel (1966)

The purpose of the new indices is to combine the characteristics of existing indices (without human interaction) in order to facilitate their use in an automated operational forecasting system (with standard interpretation). The problem with the traditional indices which are based on the mean error (such as mean percentage error, mean absolute error, root mean square error) is the relative size of the error with the forecasting parameter. Thus, it is difficult to distinguish a big error from a small error and the error is not comparable with the error of a different forecasting parameter. For that reasons, these measures were not included in the new indices.

The most commonly used forecasting verification indices when forecasting air quality using data-driven models (in the studied literature of Section 2.3.1) are the coefficient of determination and a version of the index of agreement. From Table 4-1 it is clear that one of most common disadvantages of the studied statistical measures (as reported in the literature) is their sensitivity to outliers or large errors. The coefficient of determination has this problem, because it is based on the linear fit. Outliers can have a very large effect on the line of best fit, which can lead to very different conclusions regarding the forecasting performance. For this reason, the indices (including the coefficient of determination) which are sensitive to outliers or large errors were not selected.

The second most common disadvantages of the studied statistical measures (Table 4-1) are a possible division by zero that may occur and cases where negative and the positive errors cancel out each other. Thus, these measures can be very small (indicating good performance) even when there have been large errors. For that reason, these indices were not selected.

In addition, the Spearman's rank correlation coefficient (r_s) and the Theil's Inequality Coefficient (U_1) are not providing a forecasting accuracy index, the Watterson's (M) upper and lower bounds it is not well defined and thus, it is difficult to use in an automated process, and finally, the Legates and McCabe's (E'_1) can be used for special cases, when season or another time period is available. For the aforementioned reasons, these indices were not selected.

Considering these findings, four existing FPIs were selected as the basis for the generation of the new indices. Table 4-2 presents the selected FPIs and the corresponding scalar attributes depending on Table 2-3.

Table 4-2: The selected Forecasting Performance Indices and the corresponding scalar attributes.

	Selected FPI	**Scalar Attribute(s)**
1	Index of Agreement (d_r)	Accuracy
2	Legates and McCabe's (E_1)	Accuracy & Sharpness
3	Theil's Inequality Coefficient (U_2)	Accuracy & Sharpness
4	Berry and Mielke's ($\Re$)	Association

4.2 Penalize Forecasting Performance

The overall forecasting performance for each FPI is the average of the forecasting performances that were calculated for each CV subset (see Section 2.5.5), as can be seen in Figure 4-1. The FPIs are not a set of fixed values but a population of parameters characterizing the population of forecasted values. This is due to the stochastic nature of the data-driven models and on the inherent difference between real and forecasted values, on each model's application. For this reason, it is important to characterize not only the performance of a model but also its effectiveness. The latter may be defined as the range of values that contain the FPIs, thus introducing the notion of Confidence Intervals (CIs) in the approach taken in this current research (see Section 2.5.8).

In order to increase the confidence in the estimation of the forecasting performance, relative weights were assigned for each FPI. These weights aimed at "penalizing" models with relatively low effectiveness, and thus their calculation in based on the bounds of the CIs.

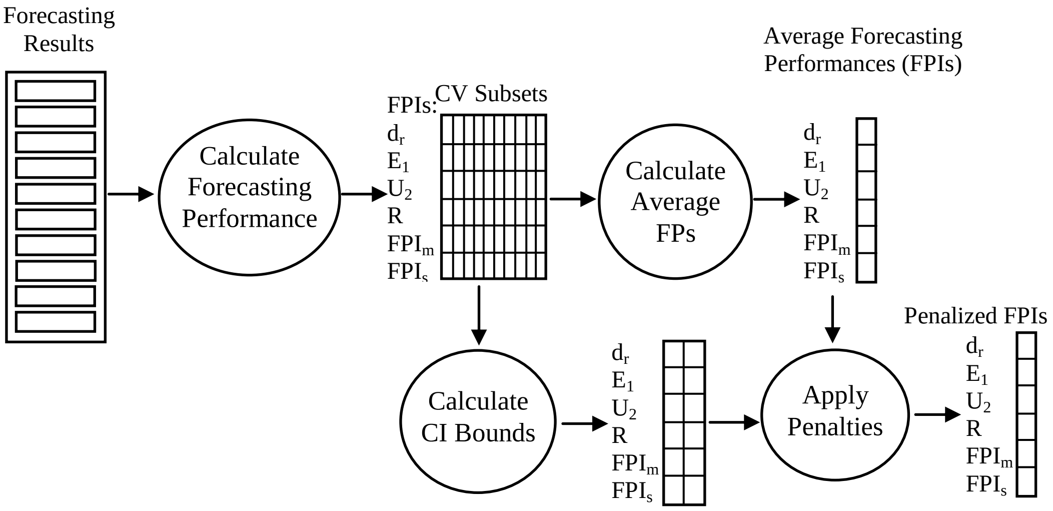


Figure 4-1: Forecasting Performance calculation

Those relative weights were referred to as "penalties" because they are calculated so as to decrease the forecasting performance of a model, but not to affect (reward) the forecasting models with relatively high effectiveness. The penalties are defined in Equation (4-1) where the "Penalty Cancel Level" defines the decrement size of the penalty effect. A small value of Penalty Cancel Level (PCL) will provide a large penalty effect while a large value of the PCL will provide small penalty effect.

$$penalty = 1 - (\text{NormalizedDistance}^{PenaltyCancelLevel}) \tag{4-1}$$

Where,

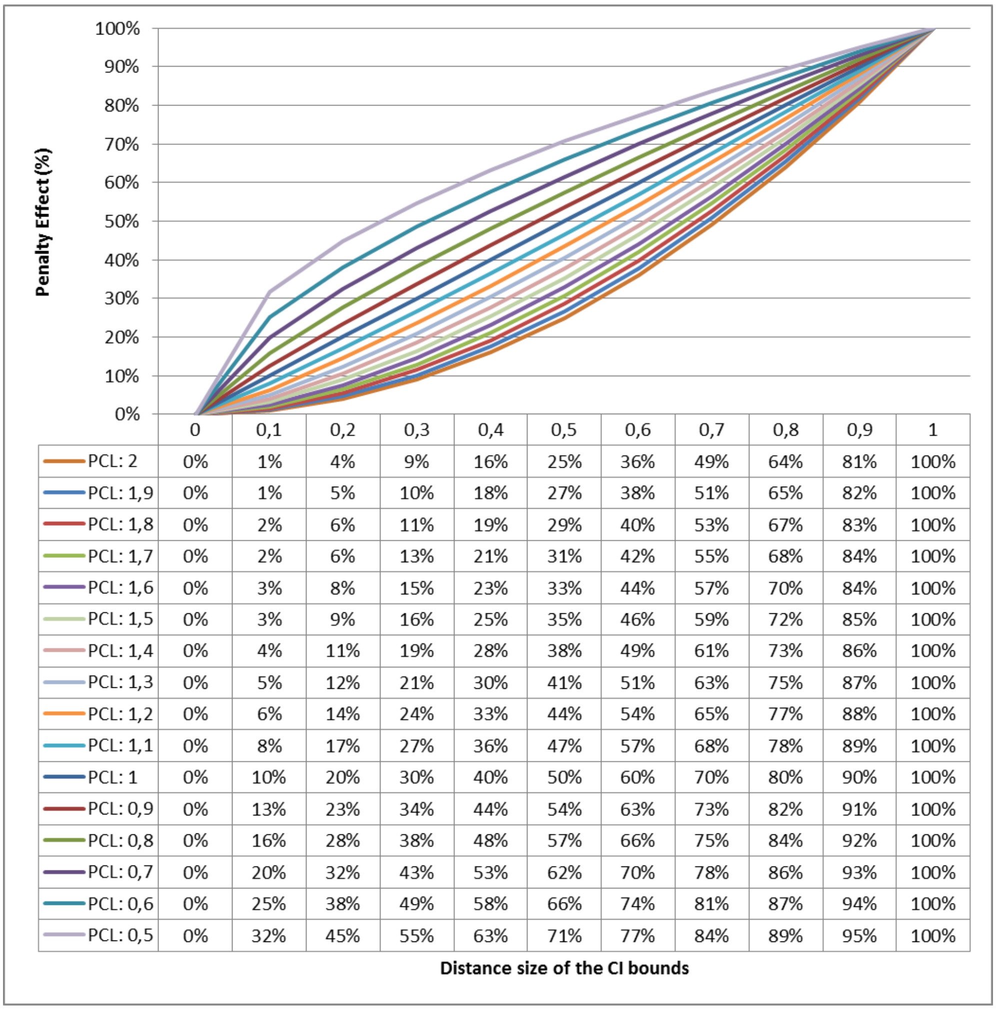
distance = (CI upper bound – CI lower bound)

$$NormalizedDistance = \begin{cases} 1, when\ distance > 1 \\ distance, when\ distance \leq 1 \end{cases}$$

In Equation (4-1) the distance was normalized to a maximum value of +1. The range between 0 and +1 was used for the following two reasons:

a) Some of the selected FPIs converge to infinite values, and
b) This value range is more appropriate for comparing performances and drawing conclusions.

Figure 4-2 presents the influence of the PCL in the penalty effect, depending on the distance size of the CI bounds. From this figure, it is clear that a penalty level of 0.5 or +1 will impose a high penalty depending on the CI distance, for the current work. A penalty level of 1.5 seems to be a preferable value for this kind of work, but this does not prevent the use of a smaller value. For that reason in the current research, three different PCLs were applied (0.5, 1 and 1.5) for comparison.



	0	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1
PCL: 2	0%	1%	4%	9%	16%	25%	36%	49%	64%	81%	100%
PCL: 1,9	0%	1%	5%	10%	18%	27%	38%	51%	65%	82%	100%
PCL: 1,8	0%	2%	6%	11%	19%	29%	40%	53%	67%	83%	100%
PCL: 1,7	0%	2%	6%	13%	21%	31%	42%	55%	68%	84%	100%
PCL: 1,6	0%	3%	8%	15%	23%	33%	44%	57%	70%	84%	100%
PCL: 1,5	0%	3%	9%	16%	25%	35%	46%	59%	72%	85%	100%
PCL: 1,4	0%	4%	11%	19%	28%	38%	49%	61%	73%	86%	100%
PCL: 1,3	0%	5%	12%	21%	30%	41%	51%	63%	75%	87%	100%
PCL: 1,2	0%	6%	14%	24%	33%	44%	54%	65%	77%	88%	100%
PCL: 1,1	0%	8%	17%	27%	36%	47%	57%	68%	78%	89%	100%
PCL: 1	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
PCL: 0,9	0%	13%	23%	34%	44%	54%	63%	73%	82%	91%	100%
PCL: 0,8	0%	16%	28%	38%	48%	57%	66%	75%	84%	92%	100%
PCL: 0,7	0%	20%	32%	43%	53%	62%	70%	78%	86%	93%	100%
PCL: 0,6	0%	25%	38%	49%	58%	66%	74%	81%	87%	94%	100%
PCL: 0,5	0%	32%	45%	55%	63%	71%	77%	84%	89%	95%	100%

Figure 4-2: The influence of the Penalty Cancel Level in the penalty effect, depending on the distance size of the Confidence Intervals bounds

Equations (4-2) and (4-3) show how the penalty was assigned in the forecasting performance depending on the FPI's nature. Equation (4-2) is used in (d_r, E_1 and $\Re$) measures, in which a low value indicates low forecasting performance and a high value indicates high forecasting performance. On the other hand, Equation (4-3) is used in (U_2) measure, in which a low value indicates high forecasting performance and a high value indicates low forecasting performance.

$$PenalizedVariable_1 = \mathrm{var}\,iable * penalty \tag{4-2}$$

$$PenalizedVariable_2 = \mathrm{var}\,iable / penalty \tag{4-3}$$

4.3 Building the Fuzzy Inference System

A basic FIS can be built using the following three steps:

1. Specify the Inputs and Outputs;
2. Determine the Membership function for each input and output;
3. Determine the rules.

4.3.1 Specify the Inputs and Outputs

The four measures selected as inputs (input space) for the FIS are the ones indicated in Section 4.1, i.e. the Index of Agreement (d_r), the Legates and McCabe's (E_1), the Theil's Inequality Coefficient (U_2) and Berry and Mielke's ($\Re$).

The output of the FIS will be the Forecasting Performance of the model scaled in five levels. This scale is selected from an existing benchmarking model. Benchmarking is essential for presenting the results of a study to a wide audience, in addition to providing guidelines to help practitioners with the use of agreement statistics (Kilem, 2012). Three benchmarking models are proposed in the literature, a) Landis and Koch's (1977), b) Fleiss's (1981) and c) Altman's (1991). Although most of these benchmarking models were initially developed to be used with the Kappa index (see Appendix III), they are often used in practice with other agreement coefficients as well (Kilem, 2012). It is worth mentioning that the Kappa index was not used in the computation of the forecasting performance. Table 4-3, Table 4-4 and Table 4-5 describe the benchmark scales of each benchmarking model.

Table 4-3: Landis and Koch Benchmark Scale

Benchmark Range	Strength of Agreement
< 0.0	Poor
0.0 to 0.20	Slight
0.21 to 0.40	Fair
0.41 to 0.60	Moderate
0.61 to 0.80	Substantial
0.81 to 1.00	Almost Perfect

Table 4-4: Fleiss's Benchmark Scale

Benchmark Range	Strength of Agreement
< 0.40	Poor
0.40 to 0.75	Intermediate to Good
More than 0.75	Excellent

Table 4-5: Altman's Benchmark Scale

Benchmark Range	Strength of Agreement
< 0.20	Poor
0.21 to 0.40	Fair
0.41 to 0.60	Moderate
0.61 to 0.80	Good
0.81 to 1.00	Very Good

The only noticeable difference between the Landis-Koch's and the Altman's benchmarking methods is that the first two ranges of values were collapsed by Altman into a single category labelled as "Poor". In the research documented in this thesis, the forecasting performance cannot have values less than zero. Thus, the Altman's benchmarking method was selected for the output of the FIS.

4.3.2 Determine the Membership function for each input and output

As a next step, the membership functions associated with each of the inputs and variables (scale of the output) were defined. Figure 4-3 shows the membership function for the inputs d_r, E_1 and $\Re$, in which the Gaussian curve membership function was used with the five Altman's Kappa levels ranges from 0 (as Poor) to +1 (Very Good). In addition, Figure 4-4 shows the membership function for the input U_2 in which the same membership function was used but with level ranges from 0 (as Very Good) to +1 (Poor). The selected statistical indices can receive values outside the range [0, 1], that were mapped to the minimum (zero) and the maximum (one) values in an appropriate way. Figure 4-5 and Figure 4-6 shows the Mamdani and Sugeno FIS developed during this research.

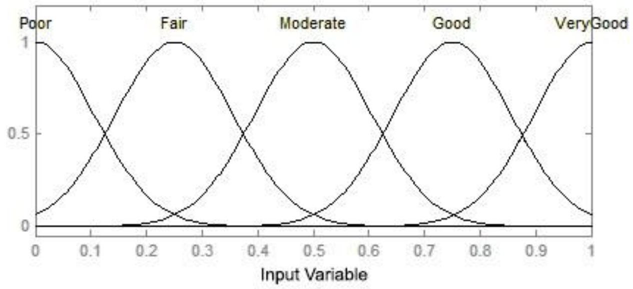


Figure 4-3: The membership function for the inputs (dr, E1 and $\Re$), as also of the output

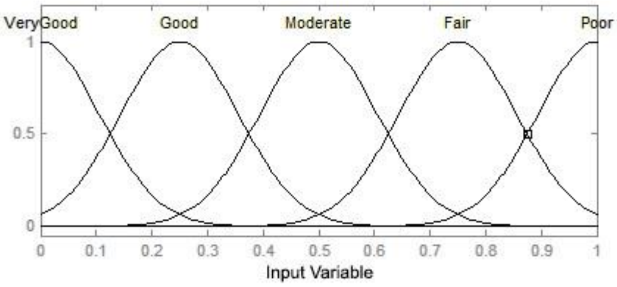


Figure 4-4: The membership function for the Theil's Inequality Coefficient input (U_2)

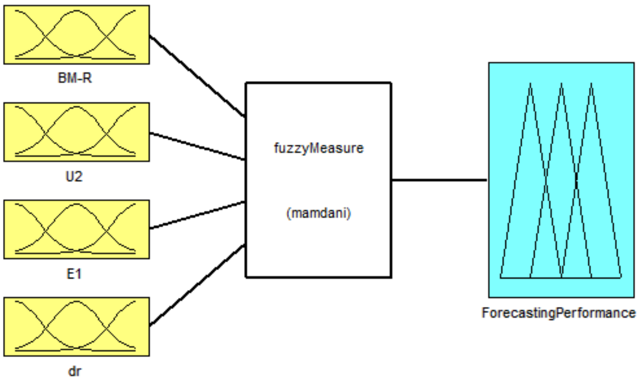


Figure 4-5: The developed Fuzzy Inference System by using the Mamdani method

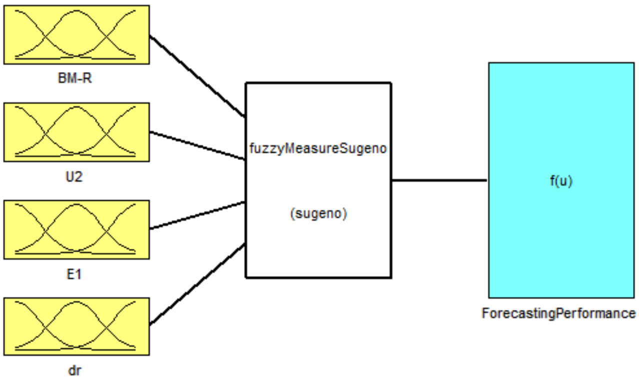


Figure 4-6: The developed Fuzzy Inference System by using the Sugeno method

4.3.3 FIS Rules

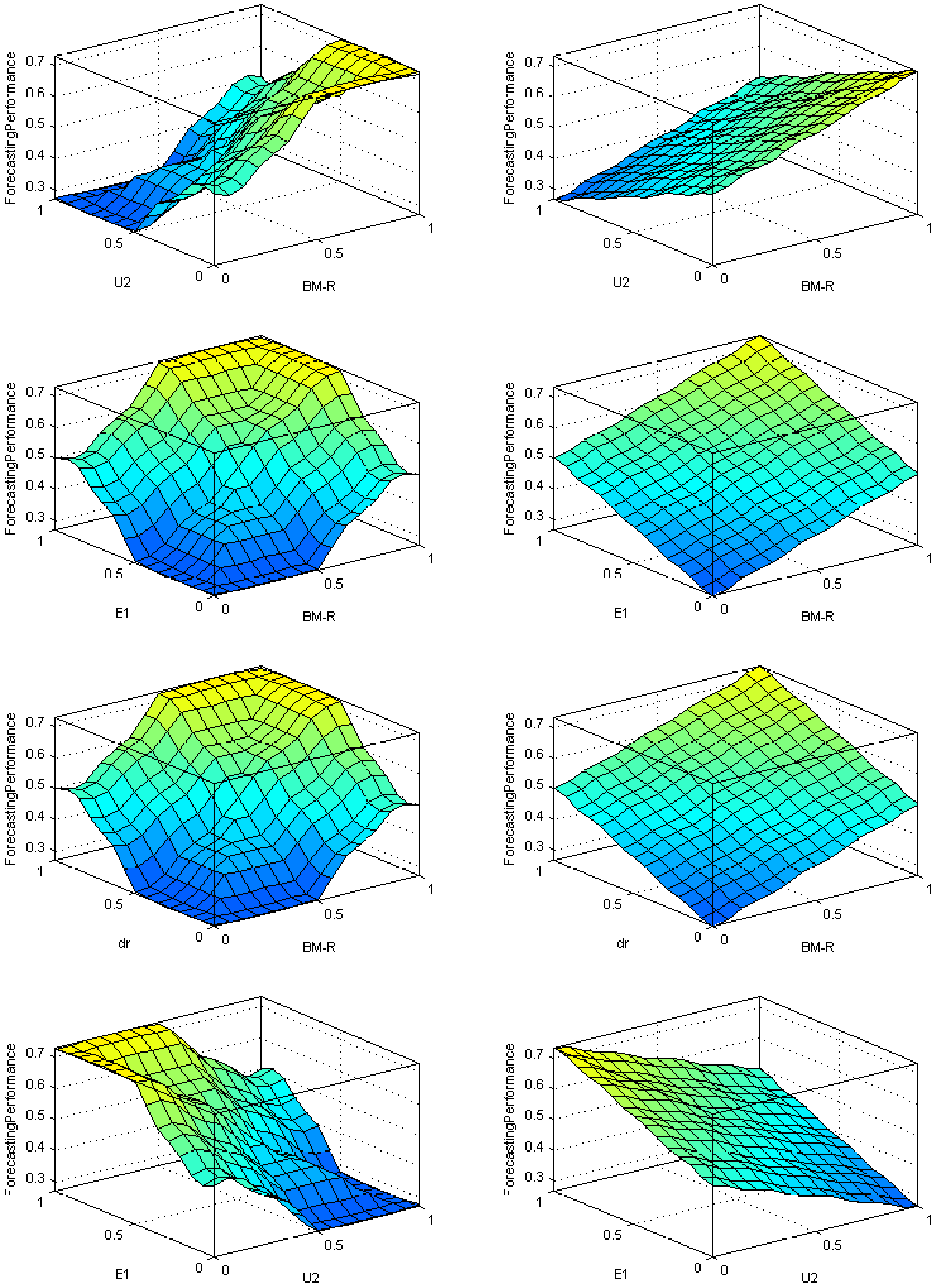
In the last step, the rules of the FIS (presented hereafter) were constructed. The basic idea behind these rules is to support the mapping process between the input and the output space. Thus, the FIS rules were defined to relate each input (forecasting performance index) with the output (forecasting performance). One of the reasons why this solution was chosen was that there are no generally accepted rules to relate the selected measures, a) with each other, and b) with the forecasting performance. An additional reason for choosing the aforementioned solution was that it was not possible to create a rule for each combination of input-output. In that case a very large number of rules would be created (a total of $performance\ levels^{number\ of\ indices} = 5^4 = 625$), and also it would be difficult to select the appropriate output for each input. The FIS rules are the result of the four indices compared with the five levels of performance (a total of 20 rules), as detailed below (Table 4-6).

Table 4-6: The constructed Fuzzy Inference System rules

	Rule	Weight
1	If ($\Re$ is Poor) then (Forecasting Performance is Poor)	1
2	If ($\Re$ is Fair) then (Forecasting Performance is Fair)	1
3	If ($\Re$ is Moderate) then (Forecasting Performance is Moderate)	1
4	If ($\Re$ is Good) then (Forecasting Performance is Good)	1
5	If ($\Re$ is Very Good) then (Forecasting Performance is Very Good)	1
6	If (U_2 is Poor) then (Forecasting Performance is Poor)	1
7	If (U_2 is Fair) then (Forecasting Performance is Fair)	1
8	If (U_2 is Moderate) then (Forecasting Performance is Moderate)	1
9	If (U_2 is Good) then (Forecasting Performance is Good)	1
10	If (U_2 is Very Good) then (Forecasting Performance is Very Good)	1
11	If (E_1 is Poor) then (Forecasting Performance is Poor)	1
12	If (E_1 is Fair) then (Forecasting Performance is Fair)	1
13	If (E_1 is Moderate) then (Forecasting Performance is Moderate)	1
14	If (E_1 is Good) then (Forecasting Performance is Good)	1
15	If (E_1 is Very Good) then (Forecasting Performance is Very Good)	1
16	If (d_r is Poor) then (Forecasting Performance is Poor)	1
17	If (d_r is Fair) then (Forecasting Performance is Fair)	1
18	If (d_r is Moderate) then (Forecasting Performance is Moderate)	1
19	If (d_r is Good) then (Forecasting Performance is Good)	1
20	If (d_r is Very Good) then (Forecasting Performance is Very Good)	1

Every rule has a weight (a number between 0 and +1), which is applied to the number given by the antecedent. In these rules, the weight has a value of one (has no effect at all on the mapping process).

Figure 4-7 shows the surface view of the three-dimensional view of the relationship between the inputs (d_r, E_1, U_2 and $\Re$) and the output (Forecasting Performance) for both Mamdani and Sugeno FIS.



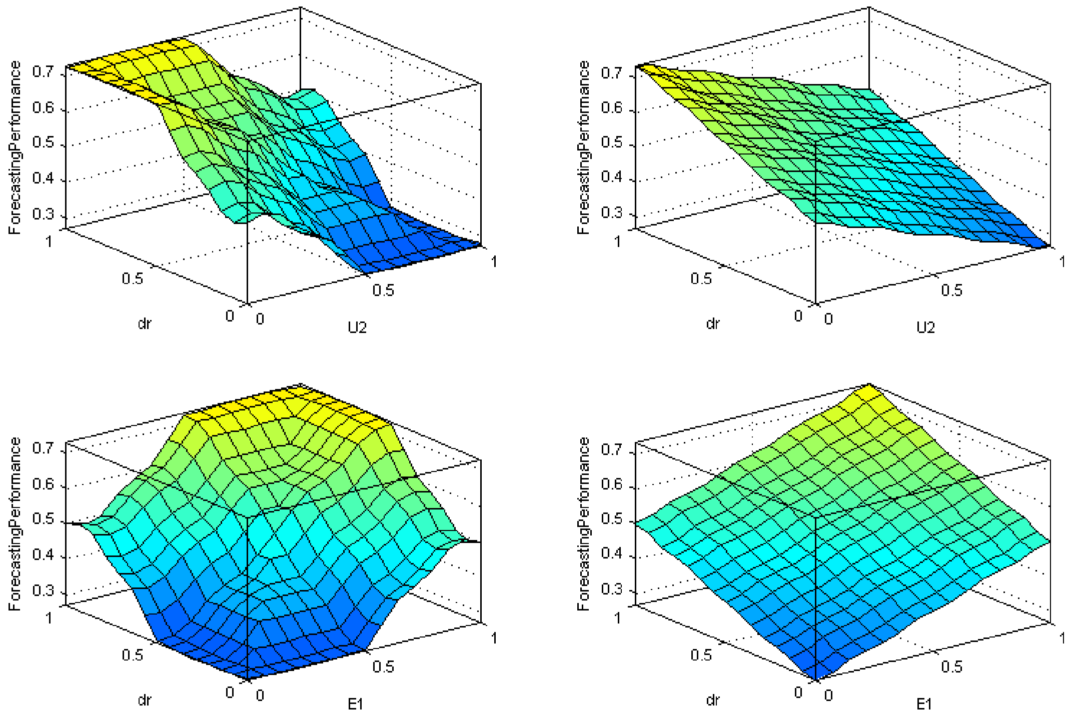


Figure 4-7: The surface view of the three-dimensional view of the relationship between the inputs and the output for both Mamdani (on the left) and Sugeno (on the right) Fuzzy Inference Systems

4.4 New Indices Evaluation

The new FPIs (FPI_m and FPI_s) were used in a population of models with varying forecasting performances, for evaluation purposes. In these models, the aim was to forecast the CAQI numerical values (dependent variable). For the sake of this part of the research, a population of model results was generated by initiating and running eight ANN-based forecasting models 1,000 times each. Subsequently, the output values of the four selected FPIs, as well as the output values of the two new indices, were computed, in order to study the consistency of their behaviours. This computation is repeated for each one of the PCL used (0.5, 1 and 1.5) in order to study its influence on the evaluation of the forecasting performance.

The forecasting performance was calculated based on each one of the indices under study. Subsequently, the percentage where each one of the eight forecasting models has been identified as a) best forecasting model or b) best or second best forecasting model, was calculated. The "best" forecasting model is the one with the highest forecasting performance for each one of the FPIs. The "second best forecasting model" is the one with the next higher forecasting performance for each one of the studied indices.

4.4.1 Area of Interest and Datasets Used

Air quality concentrations as well as meteorological data resulting from the monitoring station of Agia Sofia in Thessaloniki-Greece, for the years 2001–2003, were used. In order to forecast daily CAQIs, the three input datasets (Dataset 1, Dataset 2 and Dataset 3) detailed in Section 3.2 were used.

4.4.2 Population of Forecasting Models

A total of eight different forecasting models were developed which have a different number of inputs (depending on the Dataset) and a different number of neurons in the hidden layer. In all cases, only one hidden layer was used.

The reason to use one hidden layer and a specific number of inputs was that those models had the best forecasting performance (Section 3.9). In the work reported in Section 3.8.5, the models are named as Basic Model 3 (when Dataset 1 was used), Basic Model 4 (when Dataset 2 was used) and Basic Model 5 (when Dataset 3 was used).

In Section 3.8 two different empirical methods were presented in order to generate the different architectures (neurons per layer), having as a "starting point" the number of the input parameters (N-Input). In Method 1, as shown in Equation (3-1), the number of neurons was the same in all hidden layers. In Method 2, as shown in Equation (3-2), the number of neurons was different for each hidden layer, and more specifically, it increased as the number of hidden layers also increased. The "Neuron Multiplier" was an additional parameter used in order to increase the number of neurons and to thus generate different ANN architectures. The "Hidden Layer Number" was used in order to increase the number of neurons per hidden layer.

In the current research, Method 1 was used to generate the different architectures (neurons per layer) because only one hidden layer was used, with "Neuron Multiplier" ranges from 2 to 4. Table 4-7 shows the eight models used in the current research.

Table 4-7: The eight forecasting models used in the current research

Model Number	Dataset Number	Basic Model Number in Section 3.8.5	Number of inputs	Hidden Layer Neurons
1	1	3	1	2
2	1	3	1	3
3	2	4	9	18
4	2	4	9	23
5	2	4	9	27
6	3	5	10	20
7	3	5	10	25
8	3	5	10	30

4.5 Results and Discussion

Figure 4-8, 4-9, 4-10 and 4-11 presents the percentages of the cases in which each model was evaluated as being best, using different PCLs (none, 0.5, 1 and 1.5 respectively). These percentages are different, depending on which FPI (d_r, E_1, U_2, etc.) was used for selecting the best model. Figure 4-12 presents the mean values of each index in the cases where each forecasting model was identified as being the best when no PCLs were applied. In Figure 4-8, 4-11 and 4-12, the models 6-8 have zero values because they were not identified as best in any of the cases. From Figure 4-8 and 4-12 it is clear that d_r and E_1 FPIs lead to the same models being identified as best, even when their mean performance values are different. This does not come as a surprise as, according to Willmott, et al. (2011), these two measures are related. From Figure 4-8 it is evident that Model 3 is the best forecasting model based on all FPIs under study.

From Figure 4-8, 4-9, 4-10 and 4-11 it can be observed that, if no "penalties" are used, Model 3 is identified as the best model by all FPIs, whereas, when penalties are employed (with PCL of 0.5), the U_2 and d_r measures lead to a different model being identified as best. This suggests that CI's distance is influenced not only by the forecasting model but also by the FPI that was selected. Thus, it is crucial to choose a reliable forecasting performance measure. Note that measures E_1, $\Re$ and the new FPIs (FPI_m and FPI_s) identified Model 3 as the best model in all cases (with different PCLs). This demonstrates that those measures are more stable (in terms of consistency in the results when using penalties) in comparison with measures U_2 and d_r.

It is clear that when penalties are employed, the high performance of some models deteriorates because of their CI's in comparison to the other models. This suggests that there is no confidence in the evaluation of the forecasting performance of a model, even when it is accompanied by high FPIs. This indicates the necessity of using CI in order to penalize the forecasting performance measures and increase the confidence in the estimation of the forecasting performance.

Figure 4-8 and 4-9, indicates whether a FPI is stable, if the percentages of the cases, in which each model has been evaluated to be the best (by not using a PCL), are the same as in the case where a PCL is used. In this manner, Equation (4-4) is used to calculate the stable percentage of a FPI.

$$Stable\ Percentage = \frac{\sum_{i=1}^{N}(100 - |m_i(0) - m_i(0.5)|)}{N} \tag{4-4}$$

Where:

$m_i(x)$ is the percentage of the cases in which each model has been evaluated to be the best, by using a FPI (m) for model i, with PCL equal to x.

N is the total number of models (in this work N = 8)

It is useful to identify also the second best model, to recommend an alternative solution. For that purpose, Figure 4-13, 4-14, 4-15 and 4-16 were created in order to present the percentages of the cases in which each model has been evaluated to be the best or the second best, by using different PCLs. The stable percentages of each FPI are:

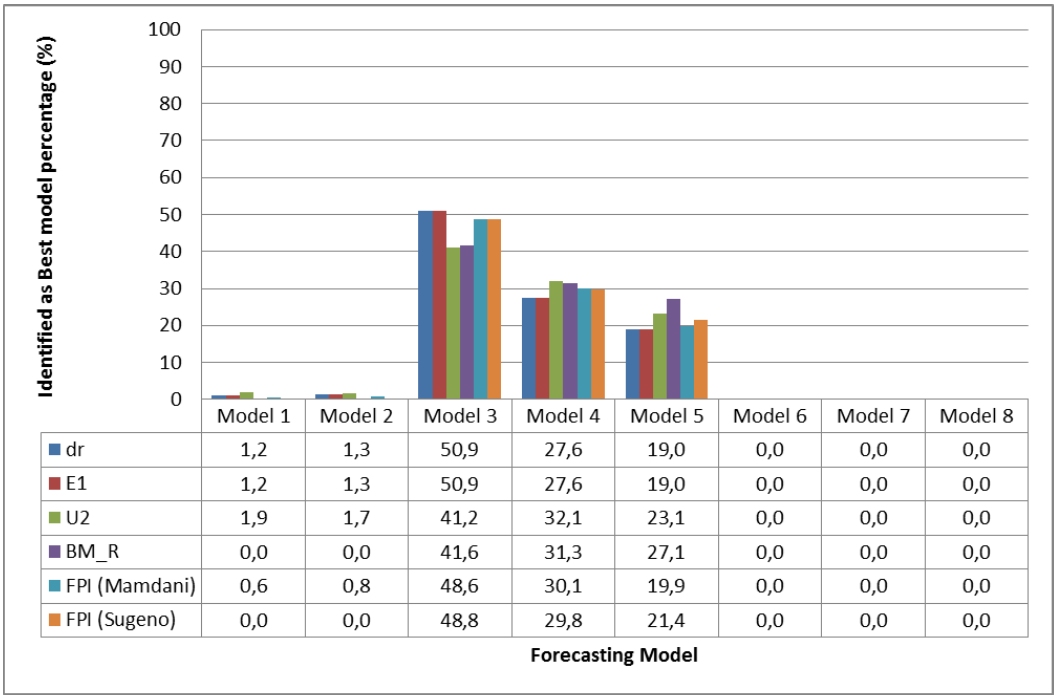
- $\mathfrak{R}$ = 98%
- E_1 = 93.7%
- FPI_s = 91%
- FPI_m = 90%
- d_r = 87.7%
- U_2 = 80.6%

From Figure 4-13, 4-14, 4-15 and 4-16 and from the aforementioned stable percentages, it is clear that measure $\mathfrak{R}$ is the most stable FPI (98%) in terms of varying penalty in comparison to the other FPIs, because its results are not affected by the increase of the penalty in comparison with the other indices.

The new FPIs (both FPI_m and FPI_s) were designed in such a way that they take into account all selected measures in a balanced way (by using the same weight value in the fuzzy rules). These indices are better compared to single measures, in respect of confidence in the estimation of the forecasting performance, because they:

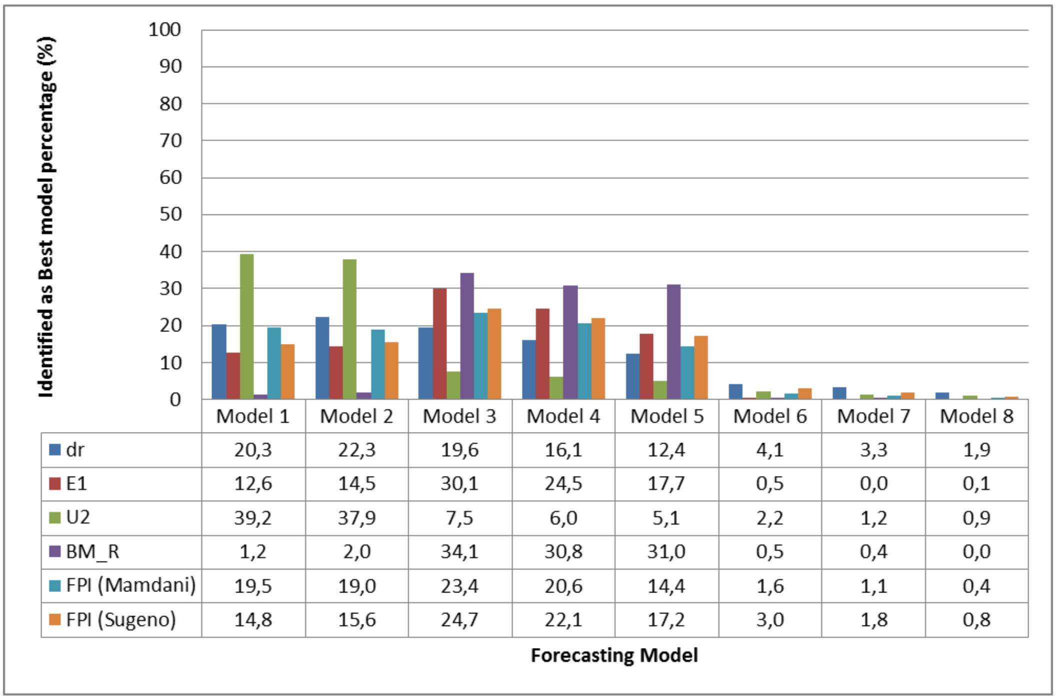
1. Use a combination of measures, which define the forecast quality by different scalar attributes (while it has been shown that there is no confidence in any single measure's estimation);
2. Use the CI in order to penalize the forecasting performance (and thus increase the confidence in the obtained results);
3. Are stable to an acceptable level 90% - 91%, which is better than d_r (87.7%) and U_2 (80.6%), but worse than $\Re$ (98%) and E_1 (93.7%);
4. Can potentially make the evaluation process of forecasting models more straightforward and robust;
5. Can be used in a forecasting system, for automatically selecting and switching to a different operational forecasting model.

By comparing the results obtained for the two new FPIs, Mamdani-type (FPI_m) and Sugeno-type (FPI_s), it is evident that both demonstrate similar behaviour, with Sugeno-type FIS being slightly more stable (by 1%). This is because, Sugeno-type FIS is not affected by the increase of the penalty in comparison with the Mamdani-type FIS.



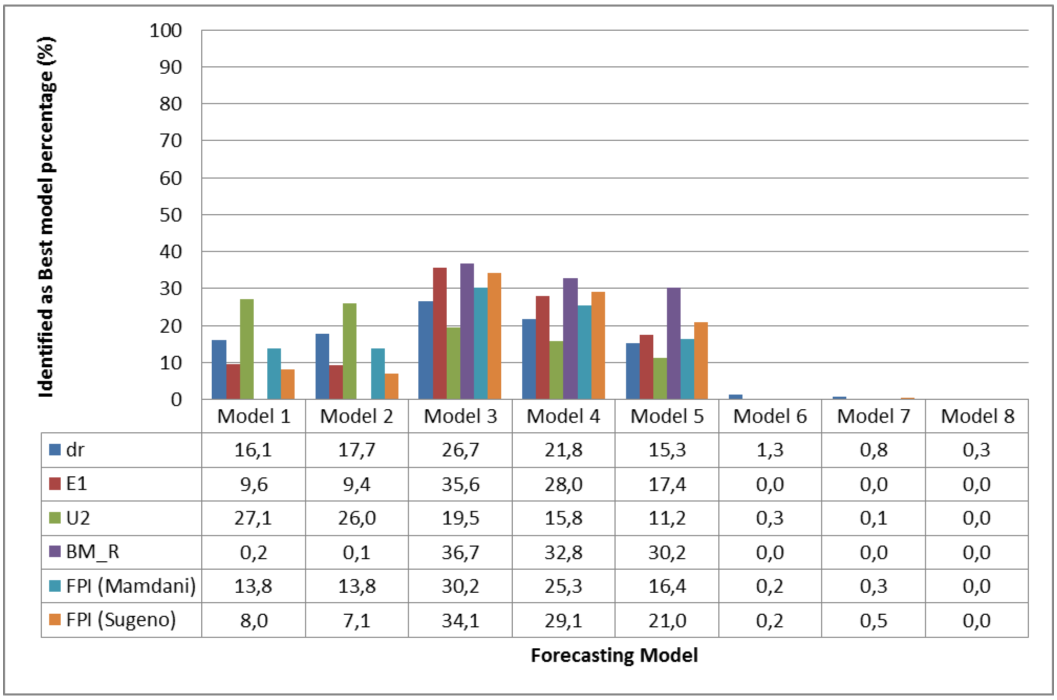
	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7	Model 8
dr	1,2	1,3	50,9	27,6	19,0	0,0	0,0	0,0
E1	1,2	1,3	50,9	27,6	19,0	0,0	0,0	0,0
U2	1,9	1,7	41,2	32,1	23,1	0,0	0,0	0,0
BM_R	0,0	0,0	41,6	31,3	27,1	0,0	0,0	0,0
FPI (Mamdani)	0,6	0,8	48,6	30,1	19,9	0,0	0,0	0,0
FPI (Sugeno)	0,0	0,0	48,8	29,8	21,4	0,0	0,0	0,0

Figure 4-8: The percentages of the cases, in which each model has been identified to be the best, without penalizing the forecasting performance



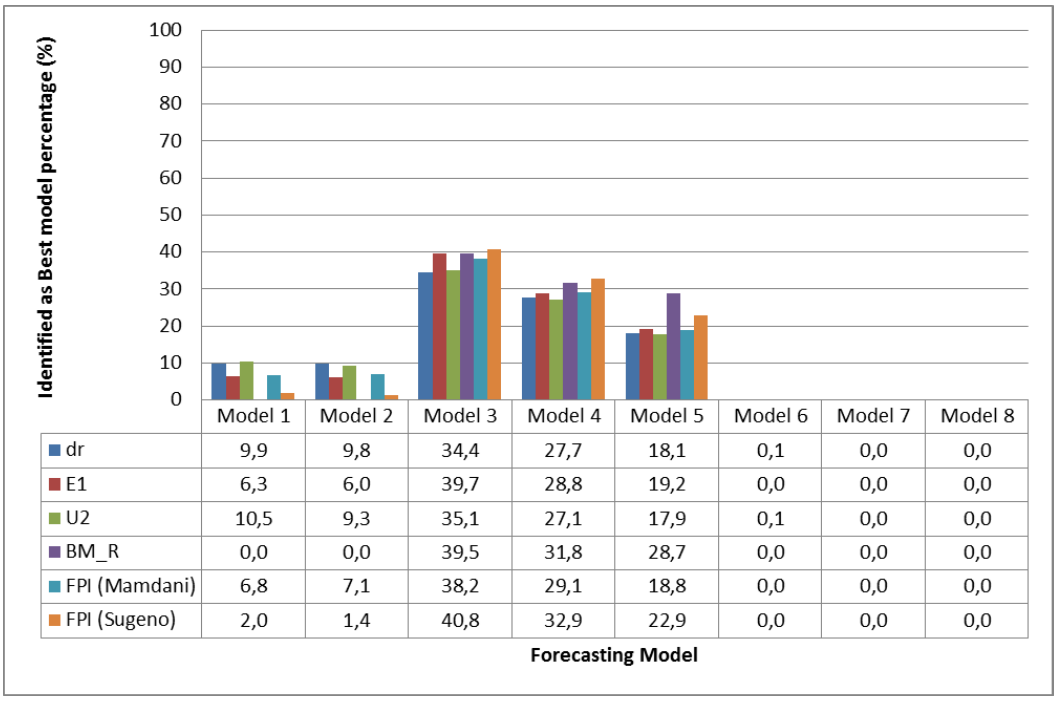
	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7	Model 8
dr	20,3	22,3	19,6	16,1	12,4	4,1	3,3	1,9
E1	12,6	14,5	30,1	24,5	17,7	0,5	0,0	0,1
U2	39,2	37,9	7,5	6,0	5,1	2,2	1,2	0,9
BM_R	1,2	2,0	34,1	30,8	31,0	0,5	0,4	0,0
FPI (Mamdani)	19,5	19,0	23,4	20,6	14,4	1,6	1,1	0,4
FPI (Sugeno)	14,8	15,6	24,7	22,1	17,2	3,0	1,8	0,8

Figure 4-9: The percentages of the cases, in which each model has been identified to be the best, by using a Penalty Cancel Level of 0.5



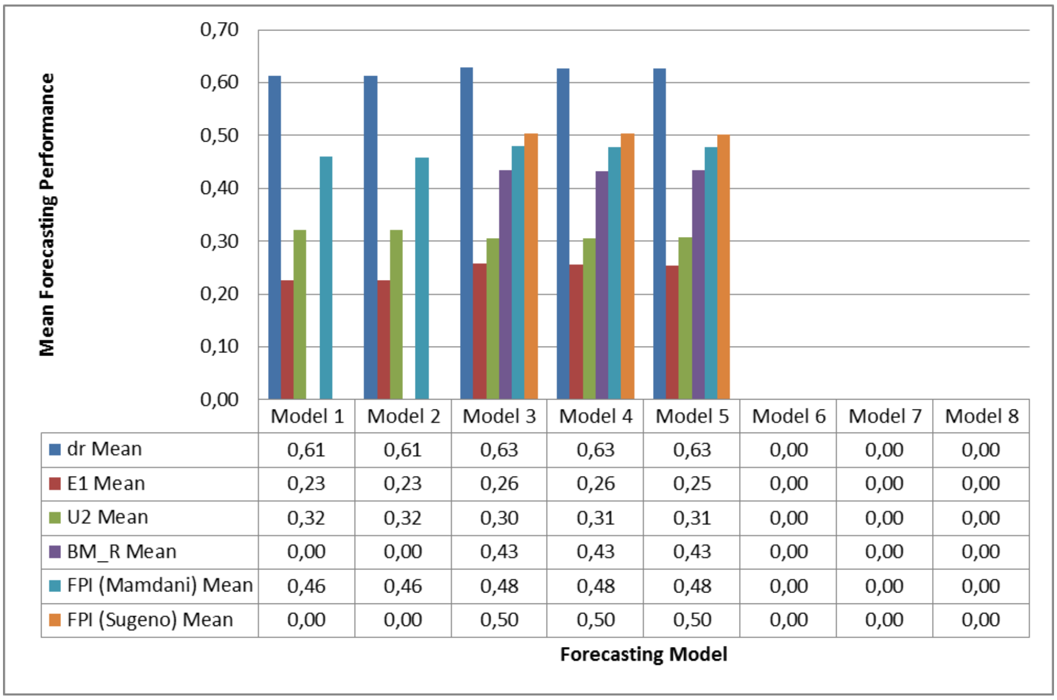
	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7	Model 8
🟦 dr	16,1	17,7	26,7	21,8	15,3	1,3	0,8	0,3
🟥 E1	9,6	9,4	35,6	28,0	17,4	0,0	0,0	0,0
🟩 U2	27,1	26,0	19,5	15,8	11,2	0,3	0,1	0,0
🟪 BM_R	0,2	0,1	36,7	32,8	30,2	0,0	0,0	0,0
🟦 FPI (Mamdani)	13,8	13,8	30,2	25,3	16,4	0,2	0,3	0,0
🟧 FPI (Sugeno)	8,0	7,1	34,1	29,1	21,0	0,2	0,5	0,0

Figure 4-10: The percentages of the cases, in which each model has been identified to be the best, by using a Penalty Cancel Level of 1



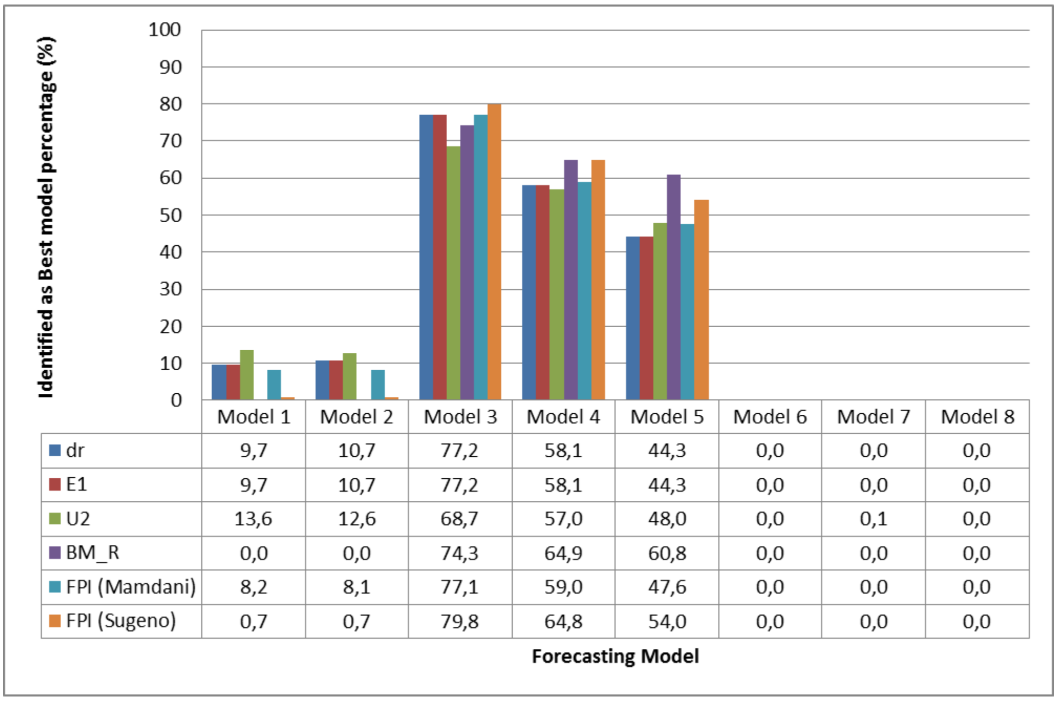
	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7	Model 8
🟦 dr	9,9	9,8	34,4	27,7	18,1	0,1	0,0	0,0
🟥 E1	6,3	6,0	39,7	28,8	19,2	0,0	0,0	0,0
🟩 U2	10,5	9,3	35,1	27,1	17,9	0,1	0,0	0,0
🟪 BM_R	0,0	0,0	39,5	31,8	28,7	0,0	0,0	0,0
🟦 FPI (Mamdani)	6,8	7,1	38,2	29,1	18,8	0,0	0,0	0,0
🟧 FPI (Sugeno)	2,0	1,4	40,8	32,9	22,9	0,0	0,0	0,0

Figure 4-11: The percentages of the cases, in which each model has been identified to be the best, by using a Penalty Cancel Level of 1.5



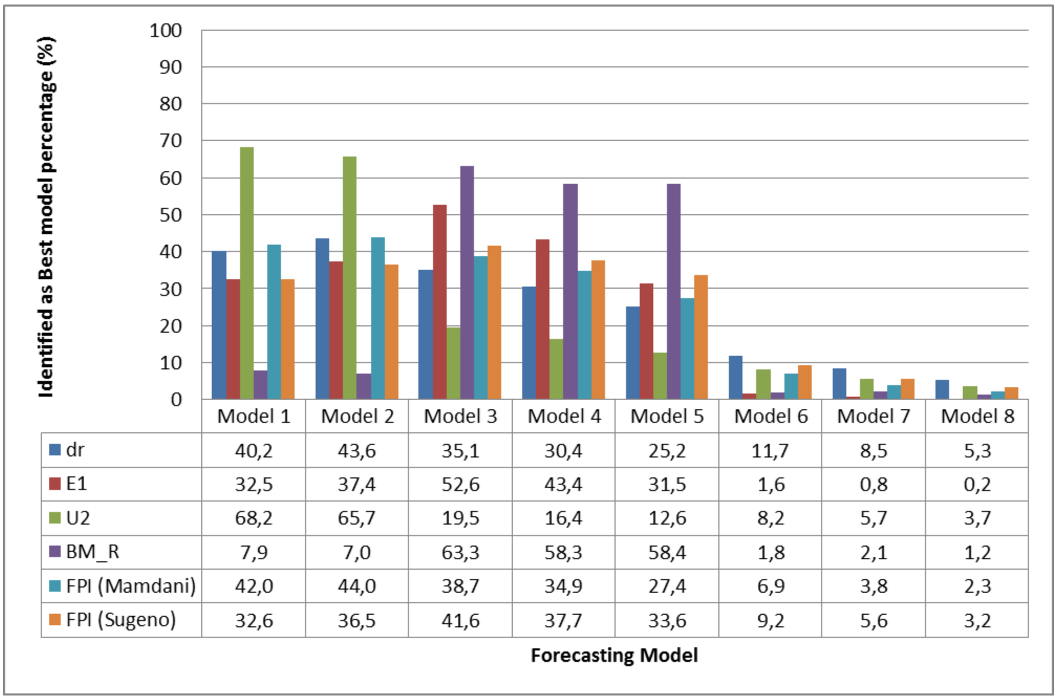
	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7	Model 8
■ dr Mean	0,61	0,61	0,63	0,63	0,63	0,00	0,00	0,00
■ E1 Mean	0,23	0,23	0,26	0,26	0,25	0,00	0,00	0,00
■ U2 Mean	0,32	0,32	0,30	0,31	0,31	0,00	0,00	0,00
■ BM_R Mean	0,00	0,00	0,43	0,43	0,43	0,00	0,00	0,00
■ FPI (Mamdani) Mean	0,46	0,46	0,48	0,48	0,48	0,00	0,00	0,00
■ FPI (Sugeno) Mean	0,00	0,00	0,50	0,50	0,50	0,00	0,00	0,00

Figure 4-12: The mean values of each measure in the cases, for which each forecasting model was identified as best when no penalty cancel levels were applied



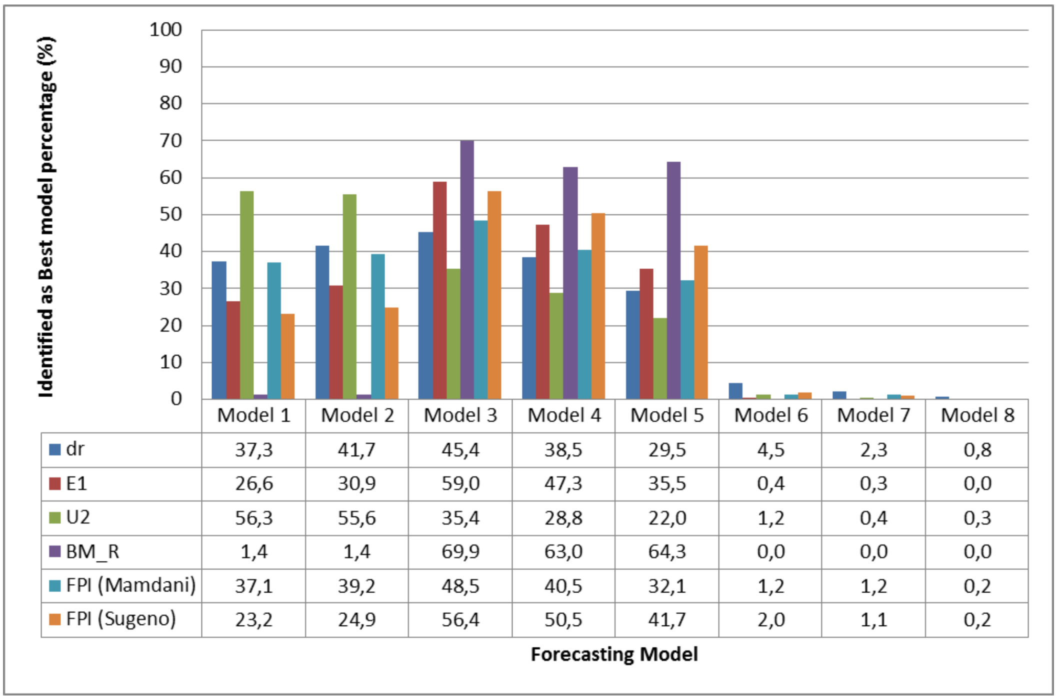
	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7	Model 8
■ dr	9,7	10,7	77,2	58,1	44,3	0,0	0,0	0,0
■ E1	9,7	10,7	77,2	58,1	44,3	0,0	0,0	0,0
■ U2	13,6	12,6	68,7	57,0	48,0	0,0	0,1	0,0
■ BM_R	0,0	0,0	74,3	64,9	60,8	0,0	0,0	0,0
■ FPI (Mamdani)	8,2	8,1	77,1	59,0	47,6	0,0	0,0	0,0
■ FPI (Sugeno)	0,7	0,7	79,8	64,8	54,0	0,0	0,0	0,0

Figure 4-13: The percentages of the cases, in which each model has been identified to be the best or the second best, without penalizing the forecasting performance



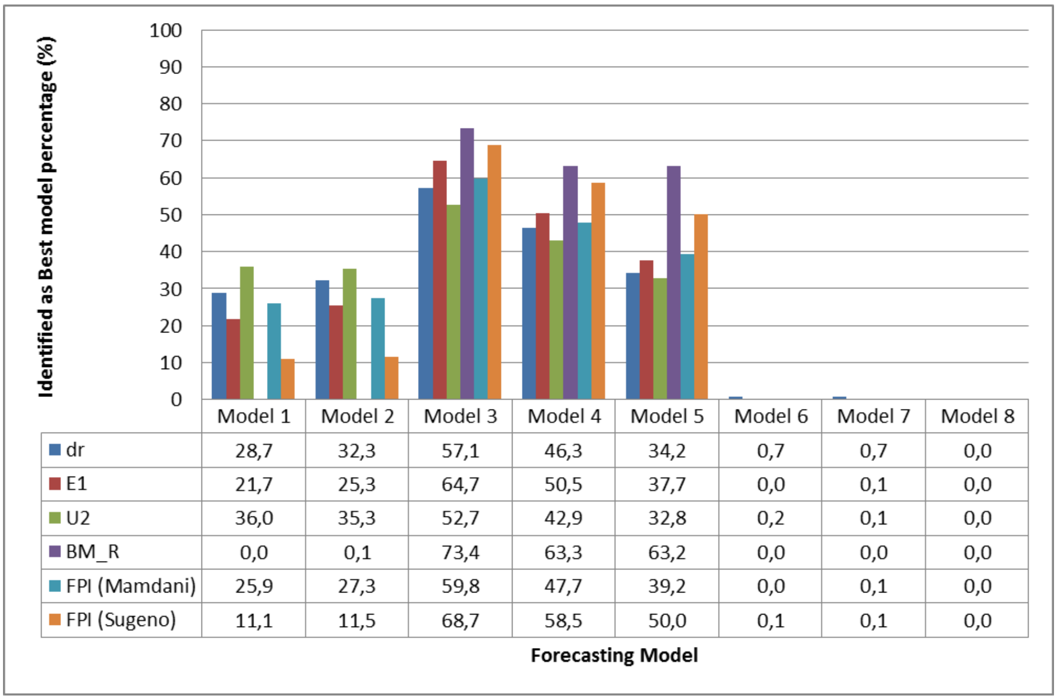
	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7	Model 8
dr	40,2	43,6	35,1	30,4	25,2	11,7	8,5	5,3
E1	32,5	37,4	52,6	43,4	31,5	1,6	0,8	0,2
U2	68,2	65,7	19,5	16,4	12,6	8,2	5,7	3,7
BM_R	7,9	7,0	63,3	58,3	58,4	1,8	2,1	1,2
FPI (Mamdani)	42,0	44,0	38,7	34,9	27,4	6,9	3,8	2,3
FPI (Sugeno)	32,6	36,5	41,6	37,7	33,6	9,2	5,6	3,2

Figure 4-14: The percentages of the cases, in which each model has been identified to be the best or the second best, by using a Penalty Cancel Level of 0.5



	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7	Model 8
dr	37,3	41,7	45,4	38,5	29,5	4,5	2,3	0,8
E1	26,6	30,9	59,0	47,3	35,5	0,4	0,3	0,0
U2	56,3	55,6	35,4	28,8	22,0	1,2	0,4	0,3
BM_R	1,4	1,4	69,9	63,0	64,3	0,0	0,0	0,0
FPI (Mamdani)	37,1	39,2	48,5	40,5	32,1	1,2	1,2	0,2
FPI (Sugeno)	23,2	24,9	56,4	50,5	41,7	2,0	1,1	0,2

Figure 4-15: The percentages of the cases, in which each model has been identified to be the best or the second best, by using a Penalty Cancel Level of 1



	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7	Model 8
■ dr	28,7	32,3	57,1	46,3	34,2	0,7	0,7	0,0
■ E1	21,7	25,3	64,7	50,5	37,7	0,0	0,1	0,0
■ U2	36,0	35,3	52,7	42,9	32,8	0,2	0,1	0,0
■ BM_R	0,0	0,1	73,4	63,3	63,2	0,0	0,0	0,0
■ FPI (Mamdani)	25,9	27,3	59,8	47,7	39,2	0,0	0,1	0,0
■ FPI (Sugeno)	11,1	11,5	68,7	58,5	50,0	0,1	0,1	0,0

Figure 4-16: The percentages of the cases, in which each model has been identified to be the best or the second best, by using a Penalty Cancel Level of 1.5

4.6 Summary

This chapter summarises the literature review carried out to facilitate the selection of suitable indices used as the basis for the generation of new indices. The four existing indices that were selected are the Index of Agreement (d_r), the Legates and McCabe's (E_1), the Theil's Inequality Coefficient (U_2) and Berry and Mielke's ($\Re$). Furthermore, a method developed to increase the confidence in the estimation of the forecasting performance using relative weights (penalties) is detailed.

This chapter also describes the steps performed in order to build the FIS. These steps included a) the specification of the inputs and outputs, b) the selection of the membership functions, and c) the construction of the FIS rules. From this procedure, two new FPIs were developed denoted as FPI_m and FPI_s for Mamdani-type and Sugeno-type FIS respectively. The new FPIs were evaluated using a population of forecasting models with varying forecasting performances by repeating parts of the CAQI forecasting (see Section 3.8). Results show that the new FPIs were better when compared to single indices in respect of confidence in the estimation of the forecasting performance. In addition, the new FPIs define the forecast quality by different scalar attributes (without human interaction), are consistent (stable) in forecasting verification and with a standardized interpretation.

Thus, the new FPIs satisfy the second research question for a more comprehensive forecasting verification (for an automated operational forecasting system).

The next chapter describes how the methodology to optimize the whole AQ forecasting chain of actions was constructed using the new FPI_s.

Chapter 5

Daphne Optimization Methodology

5.1 Introduction

"Daphne" was selected from the Greek mythology as the name for the optimization methodology developed. Pythia (also known as the oracle of Delphi) was the name of any priestess of the temple of Apollo at Delphi, located on the slopes of mount Parnassus. Pythia chew laurel leaves (also named as daphne leaves) to deliver oracles inspired by Apollo. Thus, daphne leaves were one of the means to produce oracles; in the research reported in this thesis, the proposed optimization methodology is a mean to produce forecasts.

The Daphne Optimization Methodology (DOM) was built to improve the pre-processing computational steps using air quality dataset, in terms of the performance of various forecasting algorithms applied to the output of the DOM. The order of the computational steps prior to the application of forecasting models, which define the optimization problem, are described in the Daphne Optimization Procedure (DOP).

DOM was created to address the third research question: Given a specific dataset with time stamped feature records, a set of data pre-processing algorithms and a set of forecasting algorithms, which combination of these algorithms leads to the best performance of the forecasting algorithms?

The DOM was evaluated in comparison to the traditional use of the optimization algorithms, GAs, and ACO. The next four optimization algorithms have been used and evaluated for their optimization performance:

1) Traditional GA,
2) Traditional ACO,
3) Daphne-GA, and
4) Daphne-ACO.

Each optimization algorithm was used to find solutions that lead to the best performances of the forecasting methods when forecasting the concentrations of:

1) Daily mean respirable particles (PM_{10}) in Thessaloniki, and
2) Hourly nitrogen dioxide (NO_2) in Athens.

These optimization algorithms were executed with different predefined execution parameters in order to evaluate their performance. The performance of the optimization algorithms was calculated in terms of best forecasting performance in minimum execution time. In addition, for comparison, an exhaustive search of all solutions has been performed.

Although various approaches to data pre-processing (DP) can be found in the relevant literature, it is generally accepted to include the following computational steps when applied prior to the application of forecasting models (Pyle, 1999) (Han, et al., 2011):

1. Identification and removal of outliers
2. Dealing with the missing value problem
3. Denoising-Smoothing of data
4. Detrending
5. Feature selection
6. Forecasting

It should be noted that the forecasting step mentioned here is actually the testing step that evaluates the effectiveness of all previous pre-processing computational steps.

On this basis, the DOP describes the order in which to use the six aforementioned computational steps and define the optimization problem. Table 5-1 presents the computational steps, while Figure 5-1 demonstrates the structure of the DOP as a graph. In graph theory, a graph is a representation of a set of objects where some pairs of objects are connected by links. The interconnected objects are represented by mathematical abstractions called vertices (also called nodes or points), and the links that connect some pairs of vertices are called edges (also called arcs) (Trudeau, 1993). Typically, a graph is depicted in diagrammatic form as a set of dots (or circles) for the vertices, joined by lines or curves for the edges. The edges may be directed or undirected. A directed graph (or digraph) is a graph, or set of nodes connected by edges, where the edges have a direction associated with them, in contrast with the undirected graph in which there are no directions associated with the edges.

Each computational step in DOP consists of a number of computational methods (presented as vertices in Figure 5-1) that can be applied to the data. At each computational step, only one computational method is applied to the input data, to avoid redundancy. Steps 1, 3, 4 and 5 also include the option not to use any computational method (skip the step). In the current research, the computational methods of the computational steps, prior to the forecasting step, are referred to as pre-processing methods.

There are no algorithms that can select the computational method per step that optimizes the overall DP procedure, apart from the exhaustive search of all combinations of available computational methods. The current optimization problem has the form of a longest path problem (Björklund, et al., 2004), with the following characteristics:

(a) The paths between DP methods are directed and simple (weighted edges), and
(b) Only one method can be used per computational step.

The longest path problem is an NP-complete problem, which finds a simple path of maximum length in a given graph. A path is called simple if it does not have any repeated vertices. The length of a path is measured in terms of the number of edges it has or, in weighted graphs, by the sum of the weights of its edges.

A decision problem L is NP-complete (Papadimitriou, 1994) if it is in the set of NP problems and also in the set of NP-hard problems. The abbreviation NP refers to "nondeterministic polynomial time". Although any given solution to an NP-complete problem can be verified quickly (in polynomial time), there is no known efficient, fast solution. This also applies to the current optimization problem where no technique solves it with significantly better performance than the exhausting search.

The aim of the current optimization problem is to find the path with the best forecasting performance. Nevertheless, in order to accomplish this, the distance (weights of the edges of the graph representing the optimization path) between the methods of the different computational steps, must be computed, an issue addressed in the following section.

Table 5-1: The Daphne Optimization Procedure computational steps

Computational Step	Name	Description
1	Remove Outliers	The outliers will be identified and removed from the data (will be treated as missing values)
2	Replace Missing Data	All missing values will be replaced by an estimated value
3	Smooth Data	A smoothing function will be applied to remove noise
4	Detrend Data	Trends will be identified and removed from data
5	Feature Selection / Extraction	Input data will be reduced, either by selecting a subset of relevant features or by reducing their dimensionality
6	Forecasting	A forecasting method will be used as the criterion to evaluate previous computational steps and thus lead to the final dataset

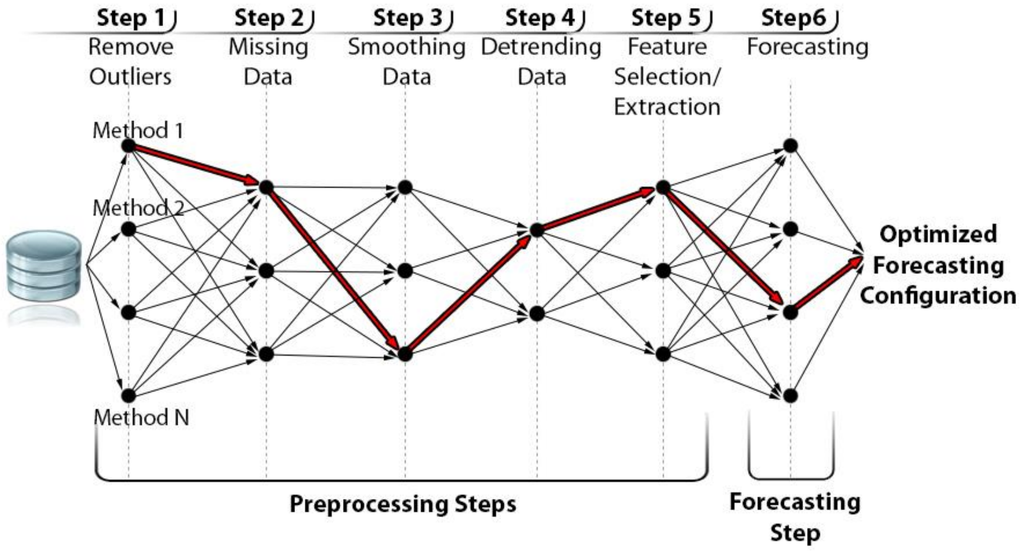


Figure 5-1: A graphical example of the Daphne Optimization Procedure

5.1.1 The Distance between Different Methods

To calculate the distance (D) or edge's weights of the DOP graph (between different pre-processing methods), each pre-processing method should be applied (one method for each computational step) to the input data in the first computational step, and to the output data of the previous pre-processing method in the other computational steps, always accompanied by the calculation of the forecasting performance.

The basic idea of the weights is thus to represent the effect of each method on the forecasting performance. Figure 5-2 presents an example of a distance calculation in the DOP graph. In this example, in order to calculate the distance D (1.2), method 1.2 (means: computational step 1, computational method of aforementioned step= 2) must be applied to the input data.

$$D(m_{a.c}) = P(m_{a.c}) - \min(P(m_a)) \quad (5\text{-}1)$$

Where,

$P(m_{a.c})$, is the forecasting performance, by using data which are pre-processed by method c of step a.

$P(m_a)$, is the vector of forecasting performances for all computational methods of step a.

$$D(m_{a.c}, m_{b.d}) = P(m_{a.c}, m_{b.d}) - P(m_{a.c}) \quad (5\text{-}2)$$

Where,

$P(m_{a.c}, m_{b.d})$, is the forecasting performance, by using data which are pre-processed by method c of step a and then by method d of step b

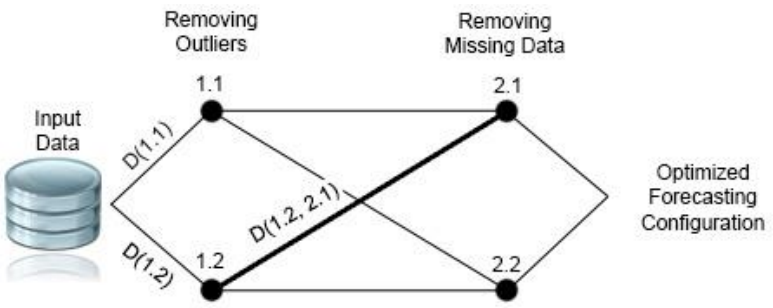


Figure 5-2: Distance calculation in Daphne Optimization Procedure graph

The example presented in Figure 5-3 demonstrates the calculation of the distance of the various computational methods in the DOP graph. In this example, methods 2.1 and 2.2 lead to a better forecasting performance provided that method 1.1 was used in the previous computational step. In addition, method 2.1 has a negative effect on the forecasting performance in the case that method 1.2 was employed in the previous step, while method 2.2 has no effect on the processing forecasting performance when following on from method 1.2.

$$D(1.1) = P(1.1) – \min(P(1)) = 0.90 – 0.90 = 0$$
$$D(1.2) = P(1.2) – \min(P(1)) = 0.91 – 0.90 = 0.1$$
$$D(1.1, 2.1) = P(1.1, 2.1) – P(1.1) = 0.93 – 0.90 = 0.03$$
$$D(1.1, 2.2) = P(1.1, 2.2) – P(1.1) = 0.92 – 0.90 = 0.02$$
$$D(1.2, 2.1) = P(1.2, 2.1) – P(1.2) = 0.85 – 0.91 = -0.06$$
$$D(1.2, 2.2) = P(1.2, 2.2) – P(1.2) = 0.91 – 0.91 = 0$$

For the case where the methods are skipped from the evaluation procedure, their distances have been set to zero (i.e. no effect on the forecasting performance).

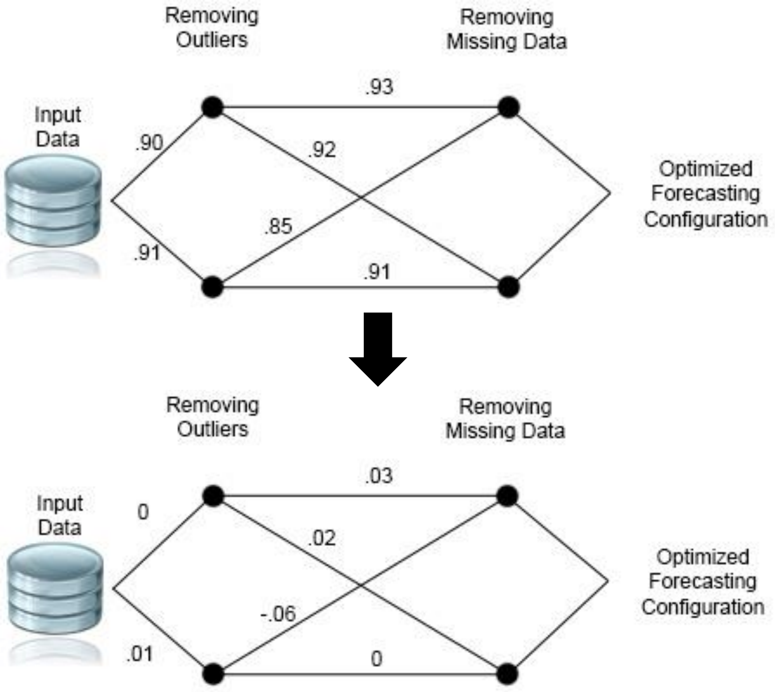


Figure 5-3: Example of a Daphne Optimization Procedure graph, a) with forecasting performance (on the top) and b) the calculated distance (on the bottom)

5.2 DOM Phases

The DOM describes the phases in order to solve the current optimization problem as follows.

Phase 1. The performance of each one of the forecasting methods is calculated without using any data pre-processing method. (In cases where the forecasting methods have special requirements, certain computational steps are followed: for example, ANNs cannot interpret missing values. Thus, a pre-processing method to deal with missing values must be used.)

Phase 2. Calculate the performance of each one of the forecasting methods by using only one data pre-processing method for all computational steps. This is an optional phase, as it can be part of the optimization algorithm that can be used for the final selection of the best possible combination of optimization steps. In the current work, this phase was included in the two optimization algorithms employed, i.e. GA and ACO.

Phase 3. Get the next feasible combination of computational methods to evaluate, by taking into consideration the distance-effect of each method. The formulation of the next feasible combination depends on the optimization algorithm employed. Sections 5.4 and 5.5 describe the modifications necessary for GA and ACO to be applied in the frame of the DOM.

Phase 4. Apply the combination of computational methods to the input data and calculate the distances. For the intermediate computational steps of the DOP (Step 1 to Step 5, of Figure 5-1), the Linear Regression method can be used to calculate the forecasting performance for the distance calculations. Linear Regression can be used for this purpose in order to have a quick estimation of the forecasting performance. The output of each pre-processing method of the combination can be stored temporally (as cache) in order to be reused in different combinations of pre-processing methods. This will reduce the processing time but will increase the necessary disk space and memory. If this mechanism is used, a search to reuse existing cached output data should be carried out and any unsaved pre-processed output data should be saved temporary (cached).

Phase 5. If the terminal conditions are reached the process stops, otherwise Phases 3 to 5 are repeated. The following three terminal condition were used in all optimization algorithms:

1. The maximum number of 20 iterations (in GA named as generations) has been reached;
2. A minimum number of five iterations were guaranteed;
3. The fitness function does not change over successive iterations. For that purpose, the average changes in fitness between successive iterations of formula (2-11) must be calculated. When the three-iterations moving average of $|f_{\Delta}|$ is equal or lower to a pre-set change tolerance threshold, the optimization procedure stops. In the research reported here, the change tolerance threshold has been set to 0.015.

5.3 Arsenal of Methods

The DOP requires an arsenal of methods for each computational step. In this research, a total of 40 methods were available for the DOP. Table 5-2 presents the methods used for each computational step of the DOP. The reasons that lead to the selection of these methods are:

- They are Included in Matlab's libraries or were included in widely used toolboxes for MATLAB. This was an assurance that the implementations of these methods were tested;
- Well-known. In most cases, researchers did not report the methods that they used for data pre-processing (Leys, et al., 2013).

Additional methods could be used for each step but it would increase the search space of the optimization problem significantly. This would make it impossible to use the exhaustive search as part of the evaluation procedure (described in detail in Section 5.6).

Table 5-2: The methods used for each computational step of the Daphne Optimization Procedure

Computational Step	Methods	References
Step 1: Remove Outliers (Total: 11 methods)	Standard Deviation Criterion (i.e. removing all values that are more than 2 STD away from the mean)	Lipowsky, et al. (2009)
	Robust Function Bisquare (Tukey's biweight)	Huber & Ronchetti (2009)
	Robust Function Andrews	Huber & Ronchetti (2009)
	Robust Function Cauchy	Huber & Ronchetti (2009)
	Robust Function Fair	Huber & Ronchetti (2009)
	Robust Function Huber	Huber & Ronchetti (2009)
	Robust Function Logistic	Huber & Ronchetti (2009)
	Robust Function Ols (Ordinary least squares)	Huber & Ronchetti (2009)
	Robust Function Talwar	Hinich & Talwar (1975)
	Robust Function Welsch	Huber & Ronchetti (2009), Holland & Welsch (1977)
	Median Absolute Deviation (MAD)	Leys, et al. (2013)
Step 2: Handling of Missing Data (Total: 10 methods)	Interpolation algorithm Linear	
	Interpolation algorithm Linear with extrapolation	
	Interpolation algorithm Piecewise	Boor (2001)
	Interpolation algorithm Piecewise with extrapolation	
	Interpolation algorithm Cubic	Boor (2001)
	Interpolation algorithm Cubic with extrapolation	
	Interpolation algorithm Spline	Boor (2001)
	Interpolation algorithm Spline with extrapolation	
	Interpolation algorithm Nearest	Rukundo & Cao (2012)
	Interpolation algorithm Nearest with extrapolation	
Step 3: Smoothing Data (Total: 6 methods)	Moving Average	
	Local regression 1st degree polynomial model (lowess)	Cleveland (1979), Cleveland & Devlin (1988)
	Local regression 2nd-degree polynomial model (loess)	Cleveland (1981)
	Savitzky-Golay filter	Savitzky & Golay (1964)
	A robust version of 'lowess'	Cleveland (1979)
	A robust version of 'loess'	Cleveland (1979)
Step 4: Detrending Data (Total: 4 methods):	Constant detrending	Kyriakidis, et al. (2009)
	Straight (linear) detrending	Kyriakidis, et al. (2009)
	Fit and remove a 2nd degree of polynomial curve	Weisstein (2015)
	Hodrick-Prescott filter	Hodrick & Prescott (1997)
Step 5: Feature Selection / Extraction (Total: 3 methods):	Factor Analysis	Harman (1976), The MathWorks (2015)
	A simple feature selection based on the Covariance. Which selects the features with the highest covariance (between the target variable)	
	Principal Components Analysis (PCA), with variable explained criteria	Jolliffe (2002)
Step 6: Forecasting (Total: 6 methods):	Linear Regression	
	Artificial Neural Networks, Feed Forward Back Propagation (FFBP)	Graupe (2007)
	Linear Neural Networks	Baldi & Hornik (1995)
	Generalized Regression Neural Networks	Wasserman (1993)
	Generalized linear model regression	Dobson (2002)
	Multivariate adaptive regression splines (MARS)	Gints (2013)

5.4 DOM for Genetic Algorithms

In order to use GAs for the materialization of the DOM, an initial population of solutions must be generated, which, in this case, are possible combinations of methods for the computational steps. In addition, the crossover mechanism must be defined in order to combine GA solutions and produce the (better) solutions of the next generation. For this reason, two parts of the "traditional" GAs have been modified.

In the first modification, a function that generates initial populations with a) feasible and b) unique solutions was created. In this optimization problem, a solution is a combination of methods (one for each DOP step). The initial solutions did not include any pre-processing technique; except for a method to replace the missing values (the method used was selected at random from the ten available methods presented in Section 5.3). This function was used to apply the Phase 1 of the DOM.

In the second modification, the Daphne Uniform Crossover (DUC) method was created in order to use the calculated distance-effect of each method and perform targeted genome crossovers. This method is a modified version of the Uniform crossover which uses the distance between the different methods to perform crossover. The advantage of DUC method is that it swaps the genomes based on a probability depending on the distance-effect of each method, and not at random by using a constant probability (mixing rate).

In the traditional uniform crossover (see Section 2.6.2.8), the loci positions in the genome are selected at random (by using a constant probability referred to as the mixing rate) and the specific genes are exchanged. The DUC selects the loci positions in the genome from a variable probability depending on the distance-effect of each method. In this way, the DUC controls the genetic diversity.

The DUC can be used in many different ways, in order to compute the exchange rate for each gene. In the research reported here, three different ways to compute the exchange rate for each gene from the distance-effect have been presented.

5.4.1 Type-1 crossover: Exchange the "bad" genes

In this type of exchange, the loci positions with "bad" genes (i.e. the genes with lower distance) have a higher probability of being exchanged. The idea is to keep the "good" genes, which will probably be consecutive, and thus, the behaviour that evolves as patterns in the population will be preserved. This is not possible in the traditional uniform crossover. In addition, by exchanging the loci positions of the “bad” genes of both parents, it is expected that the fitness of the offspring will increase. The exchange probability of each loci position varies around a constant mixing rate, as can be seen in formula (5-3). The exchange rate is calculated using formula (5-3) for each parent, and the maximum value is kept for each loci position.

$$ExchangeRate_i = Mixing\ Rate * (1 - (D(loci_i)/Sum(|D|))) \tag{5-3}$$

Where,

i, is the ith loci position

D, in the distance for each loci position in the genome

5.4.2 Type-2 crossover: Pass the “good” genes to the first child

Zajonc formulated (Zajonc, 1975) and applied (Zajonc & Bargh, 1980) the confluence theory to explain the relationships between birth order and intellectual development. He argues that later-born children tend to be less intelligent than earlier-born children. In his work, he shows that the IQ declines with increasing number of children in the family. In an explanation of his results, he states that there are two significant changes assumed to occur in the process of intellectual development of the older child. First, the intellectual environment of the family is diluted by the addition of immature siblings. Second, the older child acquires a teaching function.

In Type-2 crossover, the loci positions are exchanged, when the second child’s distance is greater than the first’s (inspired by Zajonc’s research). The exchange rate in these cases has value 1, and in all others have value 0. In this way, the first child will get all the “good” genes and expected to increase its fitness. It has to be mentioned that in this type a constant mixing rate is not needed.

5.4.3 Type-3 crossover: A combination of Type 1 and Type 2

This type combines the previous two Types in order to exchange the parent’s genes. First, Type-2 is used in order for the first child to get all the “good” genes; in the cases where the second child's distance is greater than the first's. In the cases, in which the first child has already all the good genes (and no exchange will be performed by Type-2), the Type-1 was used. By using Type-1 in these cases, it will prevent premature convergence in the GA.

5.5 DOM for Ant Colony Optimization

The ACO is very close to the DOM because the ants travel through a path towards their destination (food source). The application of the ACO meta-heuristic to the TSP (Section 2.6.5.3) is similar to the current optimization problem. The main differences are a) the current optimization problem finds the longest path (not the minimum), and b) the path in the current optimization problem is not a Hamiltonian circuit.

In order to compute the ant-routing table, formula (5-4) was used, where $D(m_{a.c}, m_{b.d})$ is the distance between methods c (of step a) and d (of step b), as shows in formula (5-5). The distance of each edge (methods c and d) of the path has been calculated with formulas (5-1) and (5-2), with a value range of [-1, 1]. In formula (5-5), the value 1 was added, in order to result in a positive heuristic value in a range of [0, 2]. Thus, the longer the distance between two methods, the higher the local heuristic value.

$$A_{a.c,b.d} = \frac{\left[\tau_{a.c,b.d}\right]^{alpha}\left[\eta_{a.c,b.d}\right]^{beta}}{\sum_{l\in N_i}\left[\tau_{a.c,b.d}\right]^{alpha}\left[\eta_{a.c,b.d}\right]^{beta}} \tag{5-4}$$

Where,

$\tau_{a.c,b.d}$ is the strength of the pheromone,

$\eta_{a.c,b.d}$ is visibility, i.e. a simple heuristic used in deciding which edge is the most attractive to be visited next

$alpha$ is a constant affecting the strength of the pheromone

$beta$: a constant affecting visibility

$$\eta_{a.c,b.d} = D(m_{a.c}, m_{b.d}) + 1 \tag{5-5}$$

Where,

N_i is the set of all the neighbor nodes of node i

Each ant k deposits a quantity of pheromone $\Delta\tau^{k}(t)$ on each connection that it has used. The $\Delta\tau^{k}(t)$ is the forecasting performance of the model, by ant k in iteration t, as shows in formula (5-6).

$$\tau_{a.c,b.d}(t) \leftarrow \tau_{a.c,b.d}(t) + \Delta\tau^{k}(t),\ k = 1, \cdots, m \tag{5-6}$$

Where,

m is the number of ants (is maintained constant in all iterations)

In order to avoid convergence to a locally optimal solution, pheromone evaporation is performed. In pheromone evaporation, the arc's pheromone strength is updated as follows: $\tau_{new} = (1-\rho)\tau_{old}$, where $\rho \in (0, 1]$ controls the speed of pheromone decay (Corne, et al., 2012).

The probability $p^{k}_{a.c,b.d}(t)$ where in the t-th algorithm iteration, an ant k used method c (of step a) chooses to use method $d \in N^{k}_{i}$ (of step b), is given by formula (5-7). Without using the DOM, which calculates the distance between methods, this probability uses only the pheromone deposition (τ) of formula (5-4). In these cases, the β (beta) parameter has been set to zero.

$$p^{k}_{a.c,b.d}(t) = \frac{A_{a.c,b.d}(t)}{\sum_{l \in N^{k}_{i}} A_{a.c,b.d}(t)} \tag{5-7}$$

Where,

$N^{k}_{i} \subseteq N_{i}$ is the feasible neighborhood of node i for ant k.

5.6 DOM Evaluation

The DOM was applied together with two optimization algorithms (GA and ACO). As a result the next four optimization algorithms have been used and evaluated for their optimization performance:

1. Traditional GA,
2. Traditional ACO,
3. Daphne GA, and
4. Daphne ACO.

The evaluation was performed in terms of best forecasting performance in minimum execution time. The term "traditional" is referred to the use of the optimization algorithms in their original form. In the earliest genetic algorithms (traditional), the related research on Holland’s original GA (1975) usually maintains a single population of potential solutions to a problem and uses a single crossover operator and a single mutation operator to produce successive generations (Baluja, 1993) (Hong, et al., 2002) (Hutt & Warwick, 2003). For the case of the traditional ACO the Dorigo & Caro (1999) meta-heuristic was used, which models the behaviour of an ant colony to find rapidly the shortest path between food sources and nest (Chen, et al., 2013).

For comparison, several (existing) methods for each step of the GA were used. Table 5-3 shows a) the methods that have been used for each step of the GA, and b) the design parameters of the GAs. This table includes the important decisions that factor into the design of the GAs (Ashlock, 2005).

In addition, in order to have accurate evaluation results an exhaustive search of all combinations of data pre-processing methods and forecasting methods (solutions) has been performed, for comparison. For the selected number of methods per computational step (Table 5-2), the exhaustive search equates to the execution of 100.800 different solutions (all possible combinations). The number of methods per computational step was selected in such a manner that would result in a limited search space where the exhaustive search could be used. In addition, two dataset sizes are used a) a relatively small dataset (3-years of daily values) was selected for the aforementioned purpose and b) two large datasets (14-years of daily values and 13-years of hourly values) are used to evaluate the optimization algorithms. Additional information regarding these datasets is presented in Section 5.6.2.

A number of trial runs of the optimizations algorithms were performed in order to configure their terminal conditions parameters (number of iterations, change tolerance threshold and minimum iterations) for the current problem (dataset, forecasting parameters, etc.). The change tolerance threshold value 0.015 was selected in order to stop the optimization procedure when a) the change in fitness was very low, and b) further improvement was not possible. The minimum number of 5 iterations was used to prevent early stopping of the optimization algorithm in a case of small changes in fitness on the first iterations. The number of maximum iterations (20) was selected to allow the optimization algorithm search for better solutions and to stop the optimization algorithm in a relatively short time.

Table 5-4 and Table 5-5 show the investigated optimization algorithms models, for GA and ACO respectively. These optimization algorithms models can be used with different predefined execution parameters, which will affect their results (in performance and execution time). In this work, the optimization algorithms models have been executed with the predefined execution parameters of Table 5-6. Each execution, for the combination of optimization algorithms and different parameters, was repeated 10-times and the performance criteria was calculated from the average of those repeated executions. This was performed in order to produce reliable results, because of the stochastic and heuristic nature of the optimization algorithms. This procedure was performed by using the relatively small dataset.

Subsequently, the optimization algorithms models with the execution parameters that lead to the best forecasting performance in minimum execution time (from the aforementioned procedure) are re-executed by using two large datasets. The first large dataset (consists of 14-years of daily values) is used to forecast PM_{10} concentrations in Thessaloniki and the other large dataset (consists of 13-years of hourly values) is used to forecast NO_2 concentrations in Athens.

Table 5-3: The methods that have been used for each step of the Genetic Algorithms and the design parameters

	Method or Parameter
Selection (of parents)	Double Tournament Selection, without replacement (the same parent cannot be selected twice)
	Roulette Wheel Selection
	Rank Selection
Crossover Parents (produce offspring)	Uniform Crossover
	Double Point Crossover
	Daphne Uniform Crossover
Mutation	Uniform Mutation
Elite offspring	One elite offspring
Data structure (representation)	Real values
Fitness Function	Sugeno Forecasting Performance Index (FPI_s)

Table 5-4: The investigated Genetic Algorithms models

#	GA Model Name	Crossover	Selection
1	DDO (Traditional GA)	Double-Point	Double Tournament
2	DRA (Traditional GA)	Double-Point	Rank
3	DRO (Traditional GA)	Double-Point	Roulette Wheel
4	UDO (Traditional GA)	Uniform	Double Tournament
5	URA (Traditional GA)	Uniform	Rank
6	URO (Traditional GA)	Uniform	Roulette Wheel
7	D-UDO (Daphne-GA)	DUC	Double Tournament
8	D-URA (Daphne-GA)	DUC	Rank
9	D-URO (Daphne-GA)	DUC	Roulette Wheel

Table 5-5: The investigated Ant Colony Optimization algorithm models

#	ACO algorithm Model	Description
1	Traditional ACO	Using only the pheromone trails to guide the ants.
2	Daphne ACO	Using a local heuristic value depending on the calculated distance, and the pheromone trails to guide the ants.

Table 5-6: The execution parameters of the optimization algorithms models

Optimization Algorithm	Parameter	Value(s)
GA	Crossover Rate	0.2, 0.3, 0.5
	Mutation Rate	0.05, 0.1, 0.2
	Tournament Selection Size (TSS)	3, 5, 8
	Exchange Type	Type-1, Type-2,Type-3
ACO	Alpha (α)	0.5, 0.75, 1
	Beta (β)	1, 2.5, 5, 7.5, 10
	rho (ρ)	0.25, 0.5, 0.75

5.6.1 The Optimization Performance Score

The criteria to compare the performance of each optimization algorithm’s execution were a) the best forecasting performance, and b) the execution time (speed). The goal of the optimization algorithms was to find a forecasting performance (that is close to the best) in a relatively short time period. In order to compare the performance of an optimization algorithm's execution (e) with the performance of the other optimization algorithms executions, the Performance Score ($PScore$) of formula 5-8 was used. The performance criteria were firstly normalized by using feature scaling to standardize their range. The simplest method of feature scaling is to rescale the range of the variables to the range [0, 1] (Formulas 5-9 and 5-11). The lower the $PScore$ value, the higher the overall performance of the optimization algorithm.

$$PScore(e) = (PerfError_{Norm}(e)/2) + (ExecutionTime_{Norm}(e)/2) \tag{5-8}$$

Where,

e, is an optimization algorithm's execution

$$PerfError_{Norm}(e) = \frac{PerfError(e) - Min(PerfError)}{Max(PerfError) - Min(PerfError)} \tag{5-9}$$

$$PerfError(e) = 1 - BestForecastingPerformance(e) \tag{5-10}$$

$$ExecutionTime_{Norm}(e) = \frac{ExecutionTime(e) - Min(ExecutionTime)}{Max(ExecutionTime) - Min(ExecutionTime)} \tag{5-11}$$

The forecasting performance (the fitness function in GA) was calculated by using the Sugeno Forecasting Performance Index (FPI_s) as proposed in Chapter 4. DOM is not restricted in the use of the FPI_s as a forecasting performance method, but it can be used with other methods such as Index of Agreement (d_r) (Willmott, et al., 1985) (Willmott, et al., 2011) or Coefficient of Determination (r^2).

5.6.2 Data Presentation

The data used in this work consist of air quality observations made at the Agia Sofia monitoring station in Thessaloniki, Greece and at the Patisia monitoring station in Athens, Greece. Thessaloniki is the second largest city in Greece and is characterized by a pronounced problem of air pollution, especially in terms of Particulate Matter (PM) and more specifically of respirable particles (PM_{10}, i.e. particles of aerodynamic diameter smaller than 10μm). For this reason, the selected forecasting parameter of this work was the daily mean concentration of PM_{10}. Athens is the capital of Greece, which is characterised by a high concentration of population and consequently with high usage of cars. For this reason, the selected forecasting parameter of this work was the hourly concentration of NO_2.

Two dataset sizes are used to evaluate the optimization algorithms for the aforementioned locations:

- A relatively small dataset (that consists of 3-years of daily values) and a large datasets (that consists of 14-years of daily values) are used to forecast PM_{10} concentrations in Thessaloniki.
- A large dataset (that consists of 13-years of hourly values) is used to forecast NO_2 concentrations in Athens.

The relatively small dataset (used to forecast daily PM_{10} concentrations) consists of 1010 daily records for the years 2010 to 2012 (3 years) and includes a total of 12 parameters: 8 air pollutant parameters and 4 meteorological parameters. The available air pollutant parameters were daily values of:

1) Mean concentration of Particulate Matter (PM_{10}),
2) Maximum eight-hour running average concentration of Ozone,
3) Mean concentration of Ozone (O_3),
4) Mean concentration of Nitrogen dioxide (NO_2),
5) Mean concentration of Nitrogen oxides (NO_X),
6) Mean concentration of Carbon monoxide (CO),
7) Maximum eight-hour running average concentration of CO, and
8) Mean concentration of Nitrogen monoxide (NO).

The available meteorological parameters were daily mean values of:

1) Relative Humidity (%),
2) Air Temperature (C),
3) Wind Speed (Km/h), and
4) Wind Direction (Degrees).

The large dataset that is used to forecast daily PM_{10} concentrations consists of 2739 daily records for the years 2001 to 2014 (14 years) and includes a total of 16 parameters: 4 air pollutant parameters and 12 meteorological parameters. The available air pollutant parameters were daily values of:

1) Mean concentration of Particulate Matter (PM_{10}),
2) Mean concentration of Ozone (O_3),
3) Mean concentration of Nitrogen dioxide (NO_2), and
4) Mean concentration of Carbon monoxide (CO)

The available meteorological parameters were daily values of:

1) Maximum, minimum and mean Relative Humidity (%),
2) Maximum, minimum and mean Air Temperature (C),
3) Maximum, minimum and mean Dew Point (C),
4) Maximum and mean Wind Speed (Km/h), and
5) Mean Wind Direction (Degrees).

The large dataset that is used to forecast hourly NO_2 concentrations consists of 88188 hourly records for the years 2003 to 2015 (13 years) and includes a total of 8 parameters: 4 air pollutant parameters and 4 meteorological parameters. The available air pollutant parameters were hourly concentration values of:

1) Nitrogen dioxide (NO_2),
2) Sulphur Dioxide (SO_2),
3) Ozone (O_3)
4) Carbon monoxide (CO)

The available meteorological parameters were hourly values of:

1) Relative Humidity (%),
2) Air Temperature (C),
3) Wind Speed (Km/h), and
4) Dew Point (C).

The air pollutants data until the year 2012 were obtained from the European air quality database (AirBase-V8, 2014). The air pollutants data from the year 2013 onwards were obtained from the European “Air Quality e-Reporting” database (AQ e-Reporting, 2017). The meteorological data were obtained from the weather underground website (Weather Underground, 2014).

5.6.3 Data Separation

One of the most important parts of evaluating computational intelligence is to train the models on a training set that is separate and distinct from the test set, for which their modelling accuracy will be evaluated. If this part is not performed, it can result in models that do not generalize to unseen data.

The data in this work were separated into two datasets. The first 70% of the data (approximately, 2 years of daily records) were used in the training set, and the remaining 30% of the data (approximately, 1 year of daily records) were used in the test set. As an exception, in the ANN models, the aforementioned 30% of the data was divided into two parts; the first 15% of the data were used in the test set, and the remaining 15% of the data were used for the validation set. The validation set is employed by the ANNs to monitor the error during the training process, in order to stop the training when the network begins to overfit the data (Caruana, et al., 2000).

In addition the 10-Fold CV was used in order to measure the predictive performance of a data-driven model. Figure 5-4 shows a) how the data was separated and b) how the CV was performed in the data. The test set (and the validation set in the case of ANNs) was not included in the pre-processing steps in order to evaluate their contribution to the forecasting performance and simulate an operational forecasting system. As an exception, the replacement of missing values was performed on the test set (and validation set), for the cases where some forecasting methods cannot interpret them.

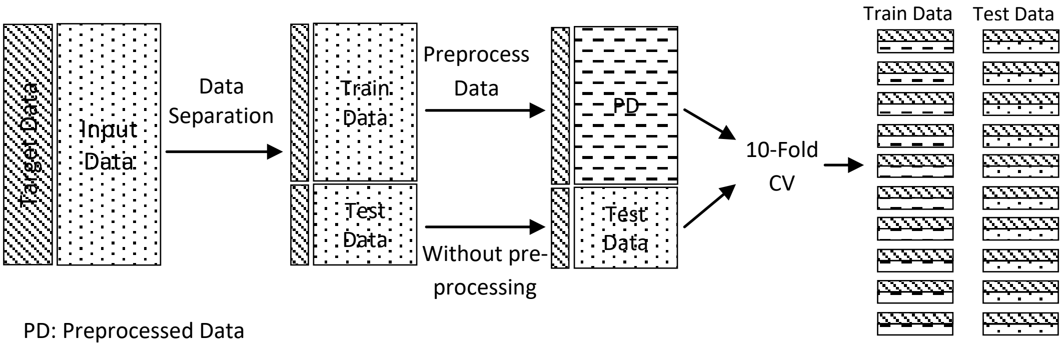


Figure 5-4: Data separation in the evaluation procedure

5.7 Results and Discussion

The exhaustive search equates to the execution of 100.800 different models (all possible combinations). The time needed to execute all possible models was approximately 13 days when using a personal computer, with an Intel(R) Core(TM)2 Quad CPU Q6600 @ 2.40GHz CPU (with Average CPU Mark: 2991 (PassMark Software, 2015)), and 4GByte RAM. The use of the exhaustive search helped, a) to evaluate the results of the optimization accurately, and b) enables to study the search space of the problem.

Figure 5-5 shows the percentage of the models for each forecasting performance group. From Figure 5-5 it is clear that the highest performance (FPI_s) of all models is 0.5. In general, this is not a very good forecasting performance, but it has to be noted that this forecasting performance is penalized and its improvement was not a goal of this work. In addition, 0.3% of these models provide the best possible results (for the studied models), i.e. a performance between 0.45 and 0.5. The optimization algorithms tried to find the models corresponding to this 0.3%.

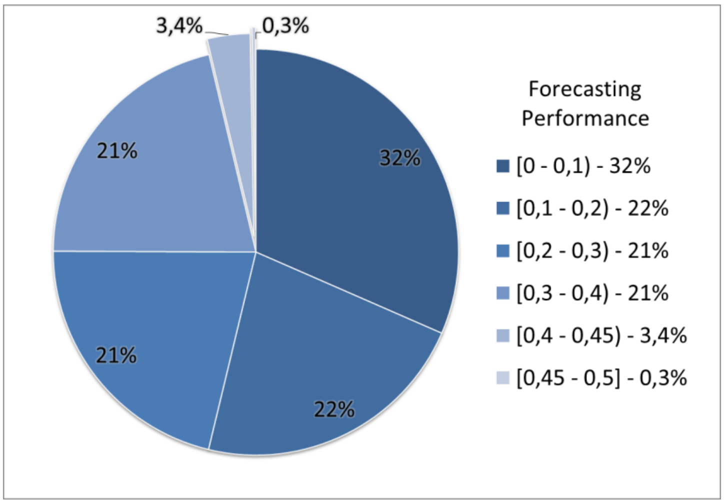


Figure 5-5: The percentage of the models for each forecasting performance group

Table 5-7 shows the ten best GA Models (from a total of 72 models), and Table 5-8 shows the ten best ACO Models (from a total of 48 models), both in terms of $PScore$. From these tables, it is clear that the optimization methods find the best solutions (i.e. solutions with a forecasting performance (FPI_s) greater than 0.45) which correspond to 0.3% of the total search space. In addition, Table 5-9 shows the average time and performance of the ten best models of each optimization type.

- By comparing the average results (Table 5-9) of the Daphne models with the Traditional models, it is clear that the Daphne models converge to solutions with greater forecasting performance. One reason to this is because the Traditional models converge faster to solutions that are not good in comparison to the Daphne models.

- By comparing the average results (Table 5-9) of the Daphne GA and the Daphne ACO, it is clear that the Daphne ACO models converged to solutions with greater forecasting performance but required in average 5 additional minutes.

In addition, it is clear that in the best ten models of each optimization type, the Daphne models have the lowest $PScore$ values, 50% of the ten best models in case of GA (Table 5-7) and 70% of the ten best models in case of ACO (Table 5-8).

In the case of GAs, the 90% of the ten best models used the Double Tournament Selection (DTS) method, with a TSS value 8 (Table 5-7). This shows the superiority of the DTS in comparison to the other selection types in the current work. In addition, 70% of those models use a mutation rate value 0.1. From Table 5-7, it can be seen that the different crossover rates did not affect the Daphne GA models, which is not surprising because the DUC use a variable probability depending on the distance-effect of each method. In the case of the Traditional GA models, the crossover rates of 0.3 and 0.5 provide the best results. Moreover, from Table 5-7 it is evident that the best exchange types of the DUC are the Type-3 and Type-2. This shows that it is a better practice to pass the "good" genes to the first child, than to exchange the “bad” genes.

In the case of ACO, a) 60% of those models use an Alpha (α) value 0.5, and b) 60% use rho (ρ) value 0.5 (Table 5-8). This shows that a balanced configuration between the pheromone’s strength and the pheromone’s decay provide the best results. From Table 5-8 it is clear that the Traditional ACO models have a Beta (β) parameter of 0. This is normal because these models do not use the distance-effect as a heuristic value. In addition, from Table 5-8 it can be seen that the Daphne ACO models with the highest β values, provide the best result in terms of $PScore$. This indicates the importance and the contribution of the heuristic value (i.e. the use of DOM) in the optimization process in order to converge to better solutions.

Table 5-10 presents the execution methods and parameters that lead to the best forecasting performance in minimum execution time (from Table 5-7 and Table 5-9). By using these methods and parameters the optimization algorithms were re-executed by using the two large datasets (described in Section 5.6.2). Table 5-11 presents the average time and performances for each optimization algorithms model and location by using large datasets.

By comparing only the best Daphne GA model and the best Traditional GA model in Figure 5-6 (e.g. the 1st and 4th GA Model of Table 5-7), it is clear that their performance is equal, but the Traditional GA Model has higher $PScore$ by 2%. This is because the latter required approximately 1 more minute to converge to a solution with the same forecasting performance. In addition, by comparing the GA models (Daphne and Traditional) of Table 5-11 it is clear that the Daphne GA models converge to solutions with higher performance (2% FPI_s and 1% d_r for Thessaloniki and 6% FPI_s and 4% d_r for Athens) in comparison to the Traditional GA, but required additional time. This shows that the Daphne GA models can better explore the search space and converge to solutions with higher performance in comparison to the Traditional GA models but may require additional time.

By comparing only the best Daphne ACO model and the best Traditional ACO model in Figure 5-6 (e.g. the 1st and 5th ACO Model of Table 5-8), it is clear that the latter converge to a solution with lower performance. The Daphne model required approximately 5 more minutes to converge to a solution with higher performance (2% FPI_s higher), and still has 6% lower $PScore$ value. In addition, by comparing the ACO models of Table 5-11 it is clear that the Daphne ACO models converge to solutions with higher performance (1% FPI_s for Thessaloniki and 2% FPI_s for Athens) in less execution time (1 minute less for Thessaloniki and 35 minutes less for Athens) in comparison to the Traditional ACO. This shows the limitation of the Traditional ACO to explore the search space, and thus, converge to a local optimum solution. Thus, the Daphne ACO models can better explore the search space and converge to better solutions in comparison to the Traditional ACO models but may require additional time.

From Table 5-11 it is clear that a forecasting performance of 0.48 FPI_s and 0.65 d_r can be achieved by simply importing all the available data (air quality and meteorological parameters) and using the DOM, without concerning about errors in the data and without the scientific knowledge for selecting the appropriate input parameters beforehand.

By comparing the results of the exhaustive search with the outcome of the optimization algorithms (Table 5-9), it is evident the superiority of the optimization algorithms over the exhaustive search. The optimization algorithms needed approximately 0.1% of the total time that was necessary for the exhaustive search, in order to find a near optimal solution (in terms of performance). The termination conditions contribute in order to stop the optimization procedure when there was no prospect of discovering a solution to the problem. By using formula (2-11), the optimization executions stopped on average in 17 iterations, which saved 3 minutes (approximately 20% of the average execution time) in each execution. Figure 5-7 shows an example where the execution of the optimization method converged to a solution with 0.49 FPI_s performance in 18 iterations.

Table 5-7: The ten best Genetic Algorithms Models in terms of the Performance Score

	GA Model	Crossover	Mutation	TSS	Exchange Type	Time (mm:ss)	Performance (FPI_s)	d_r	Best *PScore*
1	D-UDO	0.2	0.1	8	Type-3	14:29	0,47	0.63	0,36
2	D-UDO	0.5	0.05	8	Type-2	08:27	0,45	0.61	0,37
3	D-UDO	0.2	0.1	8	Type-3	15:06	0,47	0.62	0,38
4	UDO	0.3	0.1	8	-	15:27	0,47	0.62	0,38
5	DDO	0.5	0.05	8	-	09:09	0,45	0.61	0,39
6	DDO	0.3	0.1	8	-	12:51	0,46	0.61	0,40
7	D-UDO	0.5	0.1	8	Type-3	16:05	0,47	0.63	0,40
8	DDO	0.5	0.1	8	-	16:08	0,47	0.62	0,40
9	URO	0.5	0.05	8	-	13:21	0,46	0.61	0,41
10	D-UDO	0.5	0.1	8	Type-2	16:41	0,47	0.63	0,42

Table 5-8: The ten best Ant Colony Optimization Models in terms of the Performance Score

	ACO Model	Alpha (α)	Beta (β)	rho (ρ)	Time (mm:ss)	Performance (FPI_s)	d_r	Best *PScore*
1	Daphne ACO	0,5	7,5	0,5	20:23	0,49	0.64	0,41
2	Daphne ACO	0,5	10	0,5	21:20	0,49	0.64	0,43
3	Daphne ACO	0,75	5	0,5	21:26	0,49	0.63	0,43
4	Daphne ACO	1	2,5	0,5	22:46	0,49	0.64	0,47
5	Traditional ACO	0,5	0	0,25	15:12	0,47	0.63	0,47
6	Daphne ACO	0,5	5	0,5	19:21	0,48	0.64	0,48
7	Daphne ACO	0,5	1	0,25	19:42	0,48	0.64	0,49
8	Traditional ACO	0,75	0	0,5	08:29	0,45	0.61	0,50
9	Daphne ACO	0,5	7,5	0,25	20:04	0,48	0.63	0,50
10	Traditional ACO	1	0	0,75	04:50	0,44	0.60	0,50

Table 5-9: The average time and performance of the ten best models of each optimization type.

Models	Average Performance (FPI$_s$)	Average Index of Agreement (d$_r$)	Average Time (mm:ss)
Daphne GA	0.47	0.62	14:10
Traditional GA	0.46	0.61	13:23
Daphne ACO	0.49	0.64	20:43
Traditional ACO	0.45	0.61	09:30

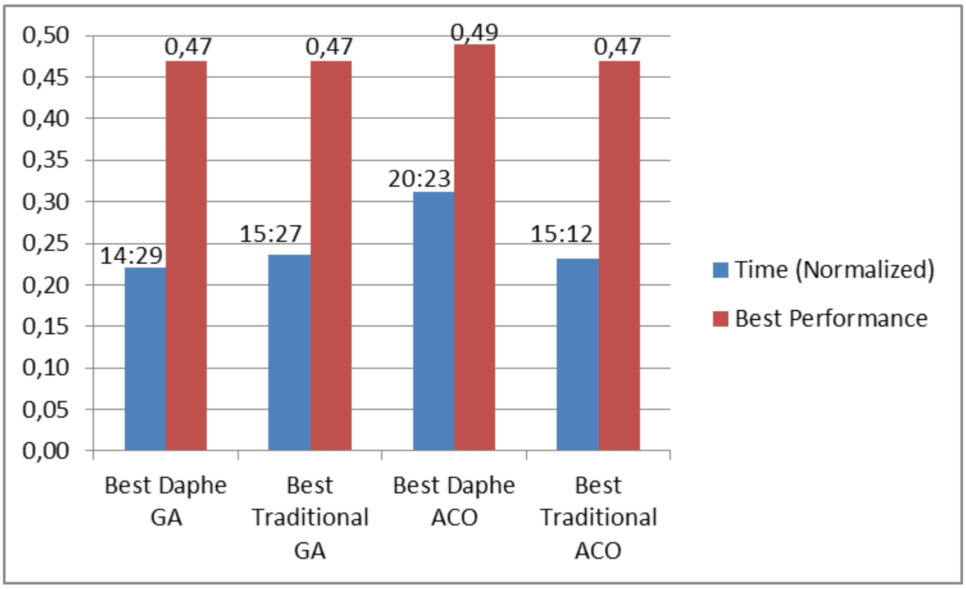


Figure 5-6: The best models of each optimization type

Table 5-10: The execution methods and parameters that lead to the best forecasting performance in minimum execution time

Optimization Algorithm	Method or Parameter	Value
Traditional GA	Crossover Parents	Uniform Crossover
	Selection Type	Double Tournament
	Crossover Rate	0.3
	Mutation Rate	0.1
	Tournament Selection Size (TSS)	8
Daphne GA	Crossover Parents	Daphne Uniform Crossover
	Selection Type	Double Tournament
	Crossover Rate	0.2
	Mutation Rate	0.1
	Tournament Selection Size (TSS)	8
	Exchange Type	Type-3
Traditional ACO	Alpha (α)	0.5
	Beta (β)	0
	rho (ρ)	0.5
Daphne ACO	Alpha (α)	0.5
	Beta (β)	7.5
	rho (ρ)	0.5

Table 5-11: The average time and performance for each optimization algorithms models and locations by using large datasets.

Models	Thessaloniki (PM_{10})			Athens (NO_2)		
	Performance (FPI_s)	Index of Agreement (d_r)	Time (hh:mm:ss)	Performance (FPI_s)	Index of Agreement (d_r)	Time (hh:mm:ss)
Traditional GA	0.46	0.64	00:06:41	0.43	0.58	01:55:57
Daphne GA	0.48	0.65	00:07:21	0.49	0.62	02:23:40
Traditional ACO	0.47	0.65	00:08:01	0.41	0.57	02:57:05
Daphne ACO	0.48	0.65	00:06:54	0.43	0.57	02:22:15

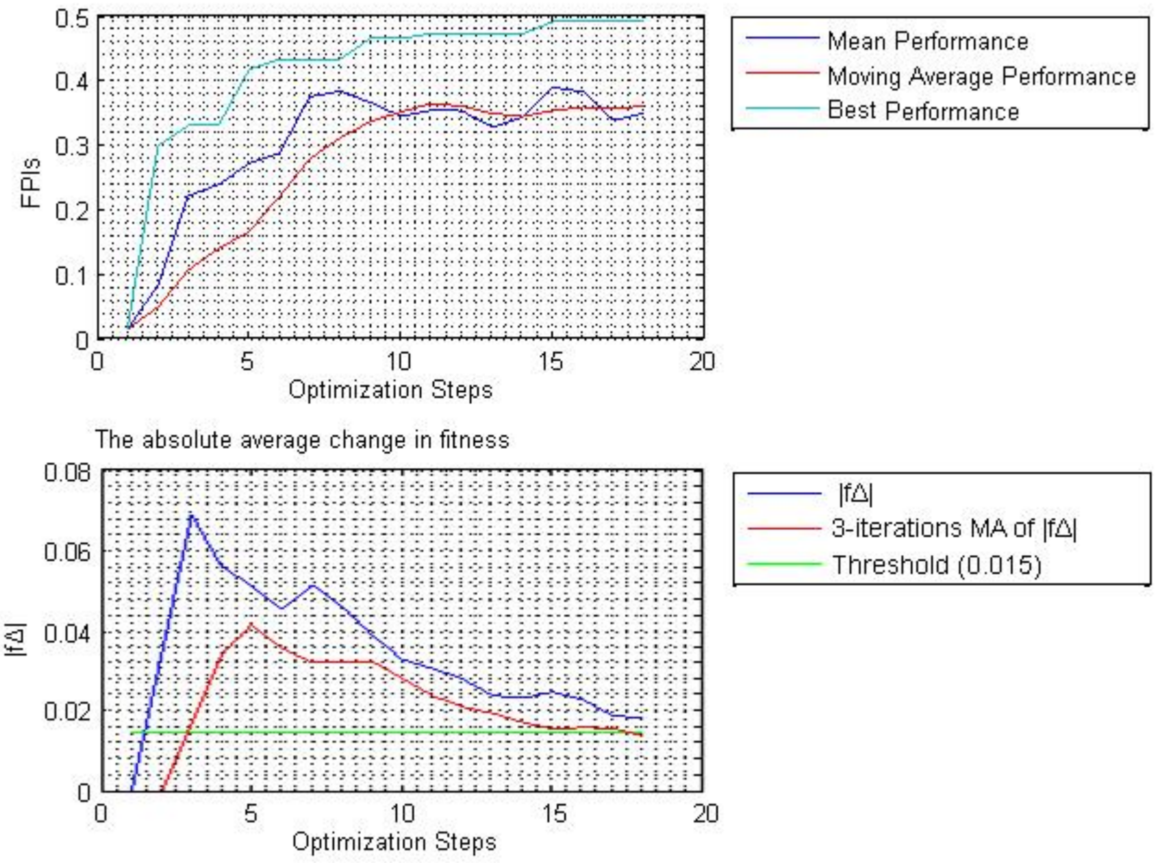


Figure 5-7: Example of the execution of an optimization method

5.8 Summary

This chapter presents the Daphne Optimization Methodology (DOM) built to optimize the whole AQ forecasting chain that includes the data pre-processing and forecasting. In addition, this chapter demonstrated how the DOP was divided into six computational steps that define the optimization problem of this work. The DOM describe in detail the phases to solve the aforementioned optimization problem. Moreover, it is described how the GAs and the ACO was modified to materialize the DOM.

To evaluate the DOM, the traditional optimization algorithms, and the exhaustive search was used for comparison. Two dataset sizes are used to evaluate the optimization algorithms, a relatively small dataset (three-years of daily values) and two large datasets (14-years of daily values and 13-years of hourly values). The criterion for comparing the performance of each optimization algorithm's execution was the Performance Score. Results show that the DOM converge to better solutions faster than the traditional optimization algorithms, which satisfy the third research question for a methodology to optimize the whole AQ forecasting chain.

The next chapter presents the results for each research question of this work, the contribution to knowledge, and the scope of future work.

Chapter 6

Discussion and Conclusions

6.1 Introduction

The motivation for the research articulated in this thesis was the development of an air quality forecasting system "fuelled" solely with air quality and meteorology monitoring data, in order to provide a means of early warning to the public and the authorities regarding high air pollution levels. The mathematical approach of data-driven models was selected in order to perform the air quality forecasting. As explained, Data-driven models require historical data (input data and targeted forecasting parameters) which has to be appropriately pre-processed. There are several data-driven algorithms that can be used as the basis for air quality forecasts, while algorithm characteristics and parameterisation may vary and greatly affect the modelling outcome. The performed research was focused on the optimization of the environmental forecasting workflow (from data pre-processing to algorithm selection and parameterisation) using data-driven models, where the forecasting accuracy is the optimization criterion used.

The work articulated in this thesis was organised by research questions (see Sections 1.2) and the way in which each was addressed is discussed below.

While the research presented in this thesis has achieved the desired goals, it has also highlighted avenues of activity for future development and study. In particular, with regard to the new forecasting performance indices, further work to improve their stability would be useful. More generally, additional work can also be performed for the DOM, as follows:-

- Increase the arsenal of methods for each computational step. This will improve the ability and efficiency of the DOM. For example, additional methods could be added to remove outliers or to produce forecasts (e.g. deep learning techniques);
- Use ensemble forecasting by combining the results (e.g. average) of different forecasting models as part of the optimization procedure;
- Parallel processing could be used in order to increase the performance of the DOM;
- The DOM has the potential to evolve to an Automated Online Forecasting System (AOFS), in which, researchers could:
  a) Upload and analyse their data;
  b) Find the optimal combination of forecasting and data pre-processing methods;
  c) Use the best solutions to provide forecasts to the public.

This AOFS could offer forecasters the opportunity to disseminate their results and thus inform the public about the forecasts. The AOFS could be given the ability to integrate new data via web-services in order to ensure that its models were adjusted along with changing circumstances.

6.2 Research Question 1: Forecasting of Environmental Parameters

Data produced by monitoring stations provide a good basis for the analysis and forecasting of air quality parameters. In this research, AQ monitoring stations from the two largest cities of Greece (Athens and Thessaloniki) were studied. Cross-validation (CV) and sensitivity analysis were used in order to perform a number of investigation experiments to test the performance of different ANN and DTs model architectures, with the aim of evaluating their applicability and reliability concerning the parameters of interest.

For the AQ monitoring stations of Athens, two important parameters of quality of life were selected, Benzene and Ozone pollutants. The forecasting of the former has to deal with the way that the event of interest (exceedance of a limit value) is defined while the forecasting of the latter deals with an 8-Hour Running Average (HRA) parameter as the basis for the definition of the event of interest. Overall, the methodology applied for the data analysis and for the forecasting of parameters that directly affect the quality of life (Benzene and Ozone concentration levels) was proven to be successful, leading to good forecasting results (reaching an index of agreement of 0.94).

For the AQ monitoring stations of Thessaloniki, the CAQI was selected as the parameter of interest that needs to be forecasted (in an hourly and daily horizon), as it is suggested by the European Environment Agency (EEA). For the different datasets and for the different scenarios, 15 ANN architectures and 4 different DT model ensembles were used (indicated as different forecasting models). Overall, a total of 1118 different models were developed and trained for daily and hourly data (559 for each one respectively). The list below summarizes the results and conclusions of the CAQI forecasting work:

- When the forecasted daily CAQI value was calculated on the basis of the forecasted hourly CAQI values, the performance was better in comparison to the one achieved via the direct forecast of the daily CAQI values.
- When weighting factors were used in order to calculate the daily CAQI values, the forecasting performance improved.
- In order to predict daily CAQI levels, simple ANN architectures (with one hidden layer) should be used.
- In order to predict hourly CAQI levels, more complex architectures (with more than one hidden layer) should be employed.
- The performance of various ANN and DT models depends both on their internal structure and on the methods used for their training.
- The DT models outperformed the ANN models with respect to their forecasting ability.
- When hourly CAQI levels were forecasted the best performance was achieved by a DT model (Percentage Agreement = 65% and Cohen's Kappa Index = 53%, equal to the performance of a LR model).

6.3 Research Question 2: Forecasting Performance Indices

This work introduces two new forecasting performance indices, which combine the characteristics of several statistical performance measures. The relative difference between the construction of the new indices (FPI_m and FPI_s) is the type of fuzzy inference system employed for the construction of each index (Mamdani and Sugeno, respectively). In addition, in order to increase the confidence in the estimation of the forecasting performance of each forecasting model, relative weights (referred to as penalties), based on the bounds of the confidence intervals were assigned to the forecasting performance of each model.

In order to evaluate the new forecasting performance indices, numerical simulations have been performed using artificial neural network models for air quality forecasting. Results show that when penalties are employed the high performance of some models deteriorates, suggesting that there is no confidence in the evaluation of the forecasting performance of a model, even when it is accompanied by high FPIs. Thus, it is important to make use of a confidence interval in order to penalize the forecasting performance measures and thus increase the confidence in the estimation of the forecasting performance.

The proposed new forecasting performance indices:

a) Are stable to an acceptable level (90% - 91%),
b) Provide a combination of several measures (which define the forecast quality by different scalar attributes), that increases the confidence in the estimation of the forecasting performance and standardize the interpretation,
c) Are comparable with the results of other studies (because it is a percentage value), and
d) Use a scale of five levels (as an additional representation) so that the results can be easier interpreted.

Thus, by using the proposed new forecasting performance indices (both FPI_m and FPI_s) in combination with the use of confidence intervals for penalizing the forecasting performance, the confidence for the estimation of the forecasting performance of a model is higher than when using any single measure. Consequently, it can be considered that the new forecasting performance indices are appropriate for automated operational forecasting systems.

6.4 Research Question 3: Daphne Optimization Methodology

In order to perform forecasts using air quality data, a combination (chain) of computational methods is employed (including the forecasting method). The selection of these methods is a very difficult but important task because it influences the forecasting performance. In this research, a general optimization methodology (referred to as Daphne) was developed in order to select the appropriate combination of methods to forecast future air quality levels.

The Daphne Optimization Methodology (DOM) is based on the idea of measuring the effect of each method (pre-processing and forecasting) on the forecasting performance and using it as a measure of distance in a directed graph. This graph represents all possible combinations of methods, which are grouped in six steps. Each step deals with a particular problem (e.g. remove outliers) and consists of a number of methods that are potential solutions. In addition, two paradigmatic implementations of DOM together with Genetic Algorithms (GAs) and Ant Colony Optimization (ACO) were described. The DOM could be applied in the selected optimization algorithms (GA and ACO) in a different manner. For example, in the case of GAs, it could be used to create a driven mutation by the distance-effect of each method.

The DOM was applied and evaluated in the two aforementioned optimization algorithms: GAs and ACO. The evaluation procedure included comparisons with the traditional use of the optimization algorithms and comparison with an exhaustive search. The criteria for comparing the performance of each optimization algorithm’s execution were a) the forecasting performance, and b) the execution time (speed), which are formulated in to a single value referred to as Performance Score ($PScore$).

Results show that the DOM converges to better solutions faster than the traditional optimization algorithms. In addition, the DOM has a 2% and 6% better $PScore$, compared to the traditional use of the optimization algorithms, when using GA and ACO respectively. In addition, when large datasets were used, DOM models converge to solutions with (up to 6% FPI$_s$) higher performance and in some cases in (up to 35 minutes) less execution time, in comparison to the traditional models. Moreover, results indicate the importance and the contribution of the DOM in the optimization process in order to converge to better solutions, and also the limitation of the traditional models to explore the search space, and thus, converge to a local optimum solution. Finally, results show that a forecasting performance of 0.48 FPI_s and 0.65 d_r can be achieved by simply importing all the available data (air quality and meteorological parameters) and using the DOM, without concern about errors in the data and without the scientific knowledge for selecting the appropriate input parameters beforehand. The aforementioned forecasting performances could be improved by using additional and more complex forecasting methods in DOM. Thus, the DOM finds a solution (combination of pre-processing and forecasting methods) which is near optimal, in comparatively shorter time than with the traditional use of the optimization algorithms.

6.5 Contribution Summary

The work plan for the research documented in this thesis was organized in to steps in order to address each of the research questions documented at its outset (see Section 1.2). These steps are listed below together with a statement regarding the contribution they have made to the research.

1. Investigate the forecasting performance of various algorithms having as targets various environmental parameters of interest with several experiments.
    - The investigated experiments provide a good understanding of the problem of maximizing the forecasting performance. In addition, the performed studies resulted in new forecasting models for the selected pollution parameters (8-hour running average Ozone concentrations, hourly Benzene concentrations, and hourly and daily Common Air Quality Index) for the cities of Thessaloniki and Athens of Greece.
    - A semi-automatic procedure is developed to perform forecasting via data-driven models, which can be generalized to other geographical locations.

2. Develop new indices for the evaluation of the forecasting performance of data-driven models.
   - Following an analysis of the relative merits of the most frequently used indices, a methodology to increase the confidence in the estimation of the forecasting performance was developed.
   - Due to the stochastic nature of the data-driven models, it is important to characterize not only the performance of a model but also its effectiveness. The effectiveness can be shown from the bounds of the confidence intervals, which were used to calculate relative weights referred to as "penalties", aimed at "penalizing" the forecasting performance of data-driven models with relatively low effectiveness. In addition, the importance of penalizing the forecasting performance measures was shown.
   - Two new forecasting performance indices (FPI_m and FPI_s) were introduced, which combine the characteristics of several statistical performance measures. When the new forecasting performance indices are combined with the use of penalties, the confidence for the estimation of the forecasting performance of a model is higher than when using any single measure.

3. Develop an optimization methodology for the whole air quality forecasting chain in order to improve the pre-processing computational steps over air quality dataset, in terms of the performance of various forecasting algorithms.
   - The Daphne Optimization Methodology (DOM) was developed, which covers the steps from input data selection (e.g. air quality and meteorological parameters) to the forecasting of the target parameter in an optimal way.
   - The DOM finds a “solution” (combination of pre-processing and forecasting methods) which is “close to the optimum”. This is achieved in comparatively shorter time than with the traditional use of the optimization algorithms.
   - The DOM can be generalized to other locations because it is created as an optimization methodology.
   - An alternative methodology that solves this problem has not been described previously in the academic literature.

6.6 Final Note

Using the Daphne Optimization Methodology and the new forecasting performance indices, reliable air quality forecasting can be performed with a forecasting performance which is close to the optimum, with automated selection of the appropriate pre-processing and forecasting methods, without concern about errors in the data or the usual prerequisite scientific knowledge for selecting the appropriate input parameters.

It is hoped that the combination of techniques presented in this thesis will help facilitate an improvement in actual air quality prediction and thus lead to better interventions both by the general public and authorities that will ameliorate poor air quality and its impact.

Index

References

Ababaei, B., Sohrabi, T. & Mirzaeidoi, F., 2012. Assessment of radial basis and generalized regression neural networks in daily reservoir inflow simulation. *Elixir Computer Science and Engineering,* Volume 42, pp. 6074-6077.

Abrahams, C. R. & Zhang, M., 2008. *Fair Lending Compliance: Intelligence and Implications for Credit Risk Management.* 1st ed. New Jersey: Wiley.

Achelis, S. B., 2013. *Technical Analysis from A to Z.* 2nd ed. s.l.:McGraw-Hill Education.

AirBase-V8, 2014. *European Environmental Agency.* [Online]
Available at: http://www.eea.europa.eu/data-and-maps/data/airbase-the-european-air-quality-database-8/
[Accessed 25 October 2014].

Alexandris, S., Stricevic, R. & Petkovic, S., 2008. Comparative analysis of reference evapotranspiration from the surface of rainfed grass in central Serbia, calculated by six empirical methods against the Penman-Monteith formula. *European Water Resources Association,* Issue 21/22, pp. 17-28.

Alshalaa, S. A. & Issmail, E. M., 2013. Comparison of Mamdani and Sugeno Fuzzy Interference Systems for the Breast Cancer Risk. *World Academy of Science, Engineering and Technology: International Journal of Computer Science and Engineering,* 7(10), pp. 840-844.

Altman, D. G., 1991. *Practical Statistics for Medical Research.* s.l.:Chapman and Hall/CRC.

Altman, D., Machin, D., Bryant, T. & Gardner, M., 2000. *Statistics with Confidence: Confidence Intervals and Statistical Guidelines.* 2nd ed. s.l.:BMJ Books.

Andrawis, R. R. & Atiya, A. F., 2009. A New Bayesian Formulation for Holt's Exponential Smoothing. *Journal of Forecasting,* Volume 28, pp. 218-234.

AQ e-Reporting, 2017. *EEA: Air Quality e-Reporting.* [Online]
Available at: https://www.eea.europa.eu/data-and-maps/data/aqereporting-2

Archer, B. & Weisstein, E. W., 2017. *"Lagrange Interpolating Polynomial." From MathWorld.* [Online]
Available at: http://mathworld.wolfram.com/LagrangeInterpolatingPolynomial.html

Arlot, S. & Celisse, A., 2010. A survey of cross-validation procedures for model selection. *Statistics Surveys,* Volume 4, p. 40–79.

Armstrong, S. J., 1985. *Long-Range Forecasting: From Crystal Ball to Computer.* New York: Wiley-Interscience.

Arora, J. S., 2012. Genetic Algorithms for Optimum Design. In: J. S. Arora, ed. *Introduction to Optimum Design (Third Edition).* Boston: Academic Press, pp. 643-655.

Ashlock, D., 2005. *Evolutionary Computation for Modeling and Optimization.* s.l.:Springer, Science+Business Media, Inc..

Atkinson, R., 1988. Atmospheric Transformations of Automotive Emissions. In: A. Y. Watson, R. R. Bates & D. Kennedy, eds. *Air Pollution, the Automobile, and Public Health.* Washington (DC): National Academies Press (US), pp. 99-132.

Azid, A. et al., 2014. Prediction of the Level of Air Pollution Using Principal Component Analysis and Artificial Neural Network Techniques: a Case Study in Malaysia. *Water Air Soil Pollut,* Volume 225:2063, pp. 1-14.

Bai, H. & Zhao, B., 2006. *A Survey on Application of Swarm Intelligence Computation to Electric Power System.* Dalian, Proceedings of the 6th World Congress on Intelligent Control and Automation, p. 7587–7591.

Bai, Y. et al., 2016. Air pollutants concentrations forecasting using back propagation neural network based on wavelet decomposition with meteorological conditions. *Atmospheric Pollution Research,* Volume 7, pp. 557-566.

Baldi, P. F. & Hornik, K., 1995. Learning in linear neural networks: a survey. *Neural Networks, IEEE Transactions,* 6(4), pp. 837-858.

Baluja, S., 1993. *The Evolution of Genetic Algorithms: Towards Massive Parallelism.* Machine Learning: Proceedings of the Tenth International Conference, Morgan Kaufmann Publishers, Inc..

Banzhaf, W., Nordin, P., Keller, R. E. & Francone, F. D., 1998. *Genetic Programming - An Introduction: On the Automatic Evolution of Computer Programs and Its Applications*. s.l.:Morgan Kaufmann Publishers, Inc. and dpunkt—Verlag fur digitale Technologic GmbH.

Barak, P., 1995. Smoothing and Differentiation by an Adaptive-Degree Polynomial Filter. *Analytical Chemistry,* 67(17), pp. 2758-2762 .

Barbosa, H. J., 2013. *Ant Colony Optimization - Techniques and Applications.* s.l.:InTech.

Barrero, M. A., Grimalt, J. O. & Canton, L., 2006. Prediction of daily ozone concentration maxima in the urban atmosphere. *Chemometrics and Intelligent Laboratory Systems,* 80(1), pp. 67-76.

Bazan, J., Skowron, A. & Synak, P., 1994. *Dynamic reducts as a tool for extracting laws from decision tables.* Berlin, Springer, p. 346–355.

Berry, K. J. & Mielke, P. J. W., 1992. A Family of Multivariate Measures of Association for Nominal Independent Variables. *Educational and Psychological Measurement,* 52(1), pp. 41-55.

Biancofiore, F. et al., 2017. Recursive neural network model for analysis and forecast of PM10 and PM2.5. *Atmospheric Pollution Research,* Volume 8, pp. 652-659.

Bigi, A. & Ghermandi, G., 2014. Long-term trend and variability of atmospheric PM10 concentration in the Po Valley. *Atmospheric Chemistry and Physics,* Volume 14, pp. 4895-4907.

Bishop, C. M., 1995. *Neural Networks for Pattern Recognition.* s.l.:Clarendon Press.

Björklund, A., Husfeldt, T. & Khanna, S., 2004. Approximating longest directed paths and cycles. *International Colloquium on Automata, Languages and Programming, Lecture Notes in Computer Science,* Volume 3142, p. 222–233.

Blej, M. & Azizi, M., 2016. Comparison of Mamdani-Type and Sugeno-Type Fuzzy Inference Systems for Fuzzy Real Time Scheduling. *International Journal of Applied Engineering Research,* 11(22), pp. 11071-11075.

Bliemel, F. W., 1973. Theil’s forecast accuracy coefficient: A clarification. *Journal of Marketing Research,* 10(4), pp. 444-446.

Blum, C. et al., 2012. Evolutionary Optimization. In: R. Chiong, T. Weise & Z. Michalewicz, eds. *Variants of Evolutionary Algorithms for Real-World.* Berlin Heidelberg: Springer-Verlag, pp. 1-29.

Bonabeau, E., Dorigo, M. & Theraulaz, G., 2000. Inspiration for Optimization from Social Insect Behavior. *Nature,* 406(6791), p. 39–42.

Boor, C. d., 2001. *A Practical Guide to Splines (Applied Mathematical Sciences).* Revised Edition ed. New York: Springer.

Borra, S. & Di Ciaccio, A., 2010. Measuring the prediction error. A comparison of cross-validation, bootstrap and covariance penalty methods. *Computational Statistics and Data Analysis,* 54(12), p. 2976–2989.

Braga-Neto, U. M. & Dougherty, E. R., 2004. Is cross-validation valid for small-sample microarray classification?. *Bioinformatics,* 20(3), p. 374–380.

Breiman, L., 1996. Bagging predictors. *Machine Learning,* 24(2), p. 123–140.

Brier, G. W., 1950. Verification of Forecasts Expressed in Terms of Probability. *Monthly Weather Review,* Volume 78, pp. 1-3.

Briggs, W. M. & Levine, R. A., 1997. Wavelets and Field Forecast Verification. *American Meteorological Society,* Volume 125, pp. 1329-1341.

Burden, R. L. & Faires, D. J., 2011. *Numerical Analysis.* 9th ed. Boston: Brooks/Cole.

Byung-Gon, K., Min-Hyeok, C. & Chang-Hoi, H., 2009. Weekly periodicities of meteorological variables and their possible association with aerosols in Korea. *Atmospheric Environment,* 43(38), pp. 6058-6065.

Câmara, D., 2015. *Bio-Inspired Networking.* 1st ed. s.l.:Elsevier.

Caruana, R., Lawrence, S. & Giles, L. C., 2000. *Overfitting in Neural Nets: Backpropagation, Conjugate Gradient, and Early Stopping.* Neural Information Processing Systems Conference, MIT Press, pp. 402-408.

Celikoglu, H. B., 2006. Application of radial basis function and generalized regression neural networks in non-linear utility function specification for travel mode choice modelling. *Mathematical and Computer Modelling,* 44(7-8), pp. 640-658.

CENR, A. Q. R. S. o. t. C., 2001. *Air Quality Forecasting: A Review of Federal Programs and Research Needs,* Boulder, Colorado: s.n.

Chakrabarti, S. et al., 2009. *Data Mining: Know It All.* 1st ed. s.l.:Morgan Kaufmann.

Chaloulakou, A., Saisana, M. & Spyrellis, N., 2003. Comparative assessment of neural networks and regression models for forecasting summertime ozone in Athens. *The Science of the Total Environment,* Volume 313, p. 1–13.

Chatfield, C., 2003. *The Analysis of Time Series: An Introduction.* 6th ed. s.l.:Chapman and Hall/CRC.

Chennakesava, A. R., 2008. *Fuzzy Logic and Neural Networks: Basic Concepts and Applications.* 1st ed. s.l.:New Age International Pvt Ltd Publishers.

Chen, W.-J., Jheng, L.-J., Chen, Y.-T. & Chen, D.-F., 2013. Optimization Path Programming Using Improved Multigroup Ant Colony Algorithms. In: J. Juang & Y. Huang, eds. *Intelligent Technologies and Engineering Systems.* New York: Springer, pp. 267-275.

Chu, S.-C., Huang, H.-C., Roddick, J. F. & Pan, J.-S., 2011. *Overview of Algorithms for Swarm Intelligence.* Gdynia, Poland, 3rd International Conference on Computational Collective Intelligence - Technologies and Applications, p. 28–41.

Clark, T. E. & McCracken, M. W., 2011. *Advances in Forecast Evaluation,* s.l.: Research Division: Federal Reserve Bank of St. Louis.

Cleveland, W. S., 1979. Robust Locally Weighted Regression and Smoothing Scatterplots. *Journal of the American Statistical Association,* 74(368), p. 829–836.

Cleveland, W. S., 1981. LOWESS: A program for smoothing scatterplots by robust locally weighted regression. *The American Statistician,* 35(1).

Cleveland, W. S. & Devlin, S. J., 1988. Locally-Weighted Regression: An Approach to Regression Analysis by Local Fitting. *Journal of the American Statistical Association,* 83(403), p. 596–610.

Coley, D. A., 1999. *An Introduction to Genetic Algorithms for Scientists and Engineers.* s.l.:World Scientific Publishing Co..

Collett, R. S. & Oduyemi, K., 1997. Air quality modelling: A technical review of mathematical approaches. *Meteorological Applications,* Issue 4, p. 235–246.

Coman, A., Ionescu, A. & Candau, Y., 2008. Hourly ozone prediction for a 24-h horizon using neural networks. *Environmental Modelling & Software,* 23(12), p. 1407–1421.

Cook, P., 2008. *Princeton University, Computer Science Department.* [Online]
Available at: http://www.cs.princeton.edu/courses/archive/fall07/cos436/HIDDEN/Knapp/fuzzy004.htm
[Accessed 16 1 2016].

Corne, D. W., Reynolds, A. & Bonabeau, E., 2012. Swarm Intelligence. In: G. Rozenberg, T. Bäck & J. N. Kok, eds. *Handbook of Natural Computing.* Berlin: Springer-Verlag, pp. 1599-1622.

Cousineau, D. & Chartier, S., 2010. Outliers detection and treatment: A review. *International Journal of Psychological Research,* 3(1), p. 58–67.

Cox, E., 2005. *Fuzzy Modeling and Genetic Algorithms for Data Mining and Exploration.* San Francisco: Elsevier Inc..

Dabberdt, W. F. et al., 2004. Meteorological Research Needs for Improved Air Quality Forecasting: Report of the 11th Prospectus Development Team of the U.S. Weather. *Bulletin of the American Meteorological Society,* pp. 563-586.

Darlington, R. B., 2015. *Psychology Department, Cornell University.* [Online]
Available at: http://www.psych.cornell.edu/Darlington/factor.htm
[Accessed 13 June 2015].

Darwin, C., 1859. *On the origin of species by means of natural selection, or the preservation of favoured races in the struggle for life.* 1st ed. London: John Murray.

Davies, H. T. & Crombie, I. K., 2009. *What are confidence intervals and p-values?,* s.l.: Hayward Medical Communications.

Debry, E. & Mallet, V., 2014. Ensemble forecasting with machine learning algorithms for ozone, nitrogen dioxide and PM10 on the Prev'Air platform. *Atmospheric Environment,* Volume 91, pp. 71-84.

Deneubourg, J. L., Aron, S., Goss, S. & Pasteels, J. M., 1990. The Self-Organizing Exploratory Pattern of the Argentine Ant. *Insect Behavior,* 3(2), pp. 159-168.

Diaz-Gomez, P. A. & Hougen, D., 2007. *Initial Population for Genetic Algorithms: A Metric Approach.* Las Vegas, Nevada, USA, Proceedings of the 2007 International Conference on Genetic and Evolutionary Methods, GEM 2007, June 25-28.

Diebold, F. X. & Lopez, J. A., 1996. Forecast Evaluation and Combination. In: G. S. Maddala & C. R. Rao, eds. *Handbook of Statistics 14: Statistical Methods in Finance.* Amsterdam, North-Holland: Butterworth Heinemann, pp. 241-268.

Dobson, A. J., 2002. *An Introduction to Generalized Linear Models.* 2nd ed. s.l.:Chapman & Hall/CRC.

Dorigo, M., 1992. *Optimization, learning and natural algorithms (in Italian),* Politecnico di Milano, Italy: Ph.D. Thesis, Dipartimento di Elettronica.

Dorigo, M., Birattari, M. & Stutzle, T., 2006. *Ant Colony Optimization: Artificial Ants as a Computational Intelligence Technique,* Institut de Recherches Interdisciplinaires et de Developpements en Intelligence Artificielle: Technical report number TR/IRIDIA/2006-023, IRIDIA.

Dorigo, M. & Caro, G. D., 1999. Ant algorithms for discrete optimization. *Artificial Life,* 5(2), pp. 137-172.

Dorigo, M. & Caro, G. D., 1999. *Ant colony optimization: a new meta-heuristic.* Washington D.C., USA, Proceedings of the 1999 Congress on Evolutionary Computation, pp. 1470-1477.

Dorigo, M. & Gambardella, L., 1997. Ant colony system: a cooperative learning approach to the traveling salesman problem. *Evolutionary Computation, IEEE Transactions,* I(1), pp. 53-66.

Dorigo, M., Maniezzo, V. & Colomi, A., 1996. The ant system: optimization by a colony of cooperating agents. *Systems, Man, and Cybernetics,* 26(1), pp. 29 - 41.

Doswell III, C. A., 1996. *Verification of forecasts of convection: Uses, Abuses, and Requirements.* Avoca Beach, New South Wales, Australia, Proceedings of the 5th Australian Sever Thunderstorm Conference.

Doytchinov, S., 2012. Effects of Air Pollution on Materials, Including Historic and Cultural Heritage Monuments. *EAI Speciale, Knowledge, Diagnostics and Preservation of Cultural Heritage,* Volume II, pp. 95-100.

Durao, R. M., Mendes, M. T. & Pereira, J. M., 2016. Forecasting O3 levels in industrial area surroundings up to 24 h in advance, combining classification trees and MLP models. *Atmospheric Pollution Research,* Volume 7, pp. 961-970.

Dye, T. et al., 2003. *Guidelines for Developing an Air Quality (Ozone and PM2.5) Forecasting Program (EPA-456/R-03-002),* Research Triangle Park, N.C.: U.S. Environmental Protection Agency, Office of Air Quality Planning and Standards, Information Transfer and Program Integration Division, AIRNow Program.

Elangasinghe, M. A., Singhal, N., Dirks, K. N. & Salmond, J. A., 2014. Development of an ANN–based air pollution forecasting system with explicit knowledge through sensitivity analysis. *Atmospheric Pollution Research,* Volume 5, pp. 696-708.

Elshout, S. et al., 2012. *Comparing Urban Air Quality Across Borders.* [Online]
Available at: http://www.airqualitynow.eu/download/CITEAIR-Comparing_Urban_Air_Quality_across_Borders.pdf

Engelbrecht, A. P., 2007. *Computational Intelligence: An Introduction.* 2nd ed. Chichester: John Wiley & Sons.

Eshelman, L. J., 1997. Evolutionary Algorithms and Their Standard Instances - Genetic algorithms. In: T. Back, D. B. Fogel & Z. Michalewicz, eds. *Handbook of Evolutionary Computation.* s.l.:IOP Publishing Ltd and Oxford University Press, pp. B1.2:1-11.

Ethem, A., 2010. *Introduction to machine learning.* 2nd ed. s.l.:Massachusetts Inst. of Tech.

EUR-Lex, 1999. *Directive 1999/30/EC of 22 April 1999 relating to limit values for sulphur dioxide, nitrogen dioxide and oxides of nitrogen, particulate matter and lead in ambient air.* [Online]
Available at: http://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:31999L0030&from=en
[Accessed 25 October 2017].

EUR-Lex, 2008. *Directive 2008/50/EC of the European Parliament and of the Council of 21 May 2008 on ambient air quality and cleaner air for Europe, OJ L 152.* [Online]
Available at: http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:32008L0050:EN:NOT
[Accessed 27 October 2011].

EUR-Lex, 2013. *Decision No 1386/2013/EU, A General Union Environment Action Programme to 2020 "Living well, within the limits of our planet".* [Online]
Available at: http://eur-lex.europa.eu/eli/dec/2013/1386/oj

European Commission, 2017. *Air Quality Standards.* [Online]
Available at: http://ec.europa.eu/environment/air/quality/standards.htm
[Accessed 25 October 2017].

European Environment Agency and World Health Organ, 2008. *Air and Health – Local authorities, health and environment, Taking actions to improve air quality.* [Online]
Available at: http://www.eea.europa.eu/publications/2599XXX/page007.html
[Accessed 1 April 2012].

Feng, X. et al., 2015. Artificial neural networks forecasting of PM2.5 pollution using air mass trajectory based geographic model and wavelet transformation. *Atmospheric Environment,* Volume 107, pp. 118-128.

Fleiss, J. L., 1981. *Statistical methods for rates and proportions.* 2nd ed. New York: John Wiley.

Fogel, D. B., 1997. Why Evolutionary Computation? - Introduction. In: T. Back, D. B. Fogel & Z. Michalewicz, eds. *Handbook of Evolutionary Computation.* s.l.:IOP Publishing Ltd and Oxford University Press, pp. A1.1:1-2.

Fox, J., 2008. *Applied Regression Analysis and Generalized Linear Models.* 2nd ed. s.l.:SAGE Publications, Inc.

Friedman, J. H., 1991. Multivariate Adaptive Regression Splines. *The Annals of Statistics,* 19(1), pp. 1-67.

Fulcher, J., 2006. *Advances in Applied Artificial Intelligence.* s.l.:Idea Group Publishing.

Gallant, S. I., 1990. Perceptron-based learning algorithms. *IEEE Transactions on Neural Networks,* 1(2), p. 179–191.

Gencay, R. & Qi, M., 2001. Pricing and Hedging Derivative Securities with Neural Networks: Bayesian Regularization, Early Stopping, and Bagging. *IEEE Transactions of Neural Networks,* 12(4), p. 726–734.

Gibert, K. et al., 2008. *On the role of pre and post-processing in environmental data mining.* Barcelona, Catalonia, s.n., pp. 1937-1958.

Gints, J., 2013. *Riga Technical University.* [Online]
Available at: http://www.cs.rtu.lv/jekabsons/Files/ARESLab.pdf
[Accessed 5 5 2015].

Goldberg, D. E. & Deb, K., 1991. A comparative analysis of selection schemes used in genetic algorithms. *Foundations of genetic algorithms,* pp. 69-93.

Gorunescu, F., 2011. *Data Mining: Concepts, Models and Techniques (Intelligent Systems Reference Library).* 2011 edition (June 17, 2011) ed. s.l.:Springer.

Goss, S., Aron, S., Deneubourg, J. L. & Pasteels, J. M., 1989. Self-organized shortcuts in the Argentine ant. *Naturwissenschaften,* 76(12), pp. 579-581.

Goss, S. et al., 1990. How Trail Laying and Trail Following Can Solve Foraging Problems for Ant Colonies. In: R. N. Hughes, ed. *Behavioural Mechanisms of Food Selection.* New York : Springer-Verlag in cooperation with NATO Scientific Affairs Division, pp. 661-678.

Grassé, P. P., 1959. La reconstruction du nid et les coordinations interindividuelles chezBellicositermes natalensis etCubitermes sp. la théorie de la stigmergie: Essai d'interprétation du comportement des termites constructeurs. *Insectes Sociaux,* 6(1), pp. 41-80.

Graupe, D., 2007. *Principles of Artificial Neural Networks (Advanced Series in Circuits and Systems Vol. 6).* 2nd ed. Singapore: World Scientific Publishing Co. Pte. Ltd.

Green, C. D., 1999. The Generalisation and Solving of Timetable Scheduling Problems. In: *Practical Handbook of Genetic Algorithms: Complex Coding Systems, Volume III.* s.l.:CRC Press LLC, pp. 39-96.

Green, K. & Tashman, L., 2008. Should we define forecast error as e= F-A or e=A-F?. *Foresight,* Issue 10, pp. 38-40.

Grivas, G. & Chaloulakou, A., 2006. Artificial neural network models for prediction of PM10 hourly concentrations, in the Greater Area of Athens, Greece. *Atmospheric Environment,* 40(7), pp. 1216-1229.

Han, J., Kamber, M. & Pei, J., 2011. *Data Mining: Concepts and Techniques.* 3rd ed. Waltham: Morgan Kaufmann.

Hannan, A. S., 2010. Generalized Regression Neural Network and Radial Basis Function for Heart Disease Diagnosis. *International Journal of Computer Applications,* 7(13), pp. 7-13.

Harman, H. H., 1976. *Modern Factor Analysis.* 3rd ed. Chicago: University of Chicago Press.

Haupt, R. L. & Haupt, S. E., 2004. *Practical Genetic Algorithms.* 2nd ed. New Jersey: John Wiley & Sons Inc..

Health Canada, E. H. D., 1997. Outdoor Air Quality and Human Health. In: *The health and environment handbook for health professionals.* s.l.:ISBN: 0-662-26728-1, pp. 112-124.

Henley, L., 2015. *A Genetic Algorithm for Data Reduction.* s.l., SAS Institute Inc. 2015. Proceedings of the SAS® Global Forum 2015 Conference. Cary, NC: SAS Institute Inc..

Hinich, M. J. & Talwar, P. P., 1975. A Simple Method for Robust Regression. *Journal of the American Statistical Association,* 70(349), pp. 113-119.

Hodrick, R. J. & Prescott, E. C., 1997. Postwar U.S. Business Cycles: An Empirical Investigation. *Journal of Money, Credit and Banking,* 29(1), pp. 1-16.

Holland, J. H., 1975. *Adaptation in Natural and Artificial Systems.* s.l.:University of Michigan Press.

Holland, P. W. & Welsch, P. E., 1977. Robust Regression Using Iteratively Reweighted Least-Squares. *Communications in Statistics: Theory and Methods,* 6(9), p. 813–827.

Hong, T.-P., Wang, H.-S., Lin, W.-Y. & Lee, W.-Y., 2002. Evolution of Appropriate Crossover and Mutation Operators in a Genetic Process. *Applied Intelligence,* 16(1), pp. 7-17.

Honoré, C. et al., 2008. Predictability of European air quality: Assessment of 3 years of operational forecasts and analyses by the PREV'AIR system. *Journal of Geophysical Research,* 113(D4).

Huber, P. J., 1964. Robust Estimation of a Location Parameter. *The Annals of Mathematical Statistics,* 35(1), pp. 73-101.

Huber, P. J., 1981. *Robust statistics.* New York: John Wiley.

Huber, P. J. & Ronchetti, E. M., 2009. *Robust Statistics.* 2nd ed. Hoboken, New Jersey: John Wiley & Sons.

Hutt, B. & Warwick, K., 2003. *Synapsing Variable Length Crossover: Biologically Inspired Crossover for Variable Length Genomes.* Proceedings of the International Conference in Roanne, France, Artificial Neural Nets and Genetic Algorithms, pp. 198-202.

Jacob, D. J., 1999. *Introduction to Atmospheric Chemistry.* s.l.:Princeton University Press.

Jacobson, M. Z., 1997. Development and Application of a New Air Pollution Modeling System - II. Aerosol Module Structure and Design. *Atmospheric Environment,* 31(2), pp. 131-144.

Jekabsons, G., 2015. *ARESLab: Adaptive Regression Splines toolbox for Matlab/Octave.* [Online]
Available at: http://www.cs.rtu.lv/jekabsons/

Jensen, R. & Shen, Q., 2008. *Computational Intelligence And Feature Selection, Rough and Fuzzy Approaches.* 1st ed. s.l.:Wiley-IEEE Press.

Jiang, N. et al., 2015. Enhancing Air Quality Forecast in New South. *Clean Air Society of Australia & New Zealand (CASANZ),* pp. 20-23.

Ji, E. -Y., Moon, Y. -J., Kim, K. -H. & Lee, G. -H., 2010. Statistical comparison of interplanetary conditions causing intense geomagnetic storms (Dst<=-100 nT). *Journal of Geophysical Research,* Volume 115, p. 1–7.

Jolliffe, I., 2002. *Principal Component Analysis.* 2nd ed. New York: Springer.

Jolliffe, I. T. & Stephenson, D. B., 2012. *Forecast Verification: A Practitioner's Guide in Atmospheric Science.* 2nd ed. s.l.:John Wiley & Sons, Ltd.

Kalapanidas, E. & Avouris, N., 2001. Short-term air quality prediction using a case-based classifier. *Environmental Modelling & Software,* 16(3), p. 263–272.

Kansal, V. & Kaur, A., 2013. Comparison of Mamdani-type and Sugenotype FIS for Water Flow Rate Control in a Rawmill. *Scientific & Engineering Research,* 4(6), pp. 2580-2584.

Kaplan, D. T., 1999. *Resampling Stats in MATLAB.* Arlington, Virginia: Resampling Stats, Inc.

Karaboga, D., 2005. *An Idea Based On Honey Bee Swarm for Numerical Optimization, Technical Report-TR06,* s.l.: Erciyes University, Engineering Faculty, Computer Engineering Department.

Karatzas, K., 2003. *A fuzzy logic approach in Urban Air Quality Management and Information Systems.* Prague, Czech Republic, Charles University, pp. 274-276.

Karatzas, K. D., Papadourakis, G. & Kyriakidis, I., 2009. Understanding and Forecasting Air Pollution with the Aid of Artificial Intelligence Methods in Athens, Greece. In: C. Koutsojannis & S. Sirmakessis, eds. *Tools and Applications with Artificial Intelligence.* Berlin Heidelberg: Springer-Verlag, pp. 37-50.

Karatzas, K., Papadourakis, G. & Kyriakidis, I., 2008. *Understanding and forecasting atmospheric quality parameters with the aid of ANNs.* Hong Kong, IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence), pp. 2580 - 2587.

Kasabov, N. K., 1998. *Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering.* 2nd ed. s.l.:Massachusetts Institute of Technology.

Kassomenos, P. et al., 2009. *Characterizing the quality of the atmosphere over an urban complex.* Chania, Crete, Greece, 11th Int. Conf. on Envl. Science and Technology, p. 424–431.

Kaur, A. & Kaur, A., 2012. Comparison of Mamdani-Type and Sugeno-Type Fuzzy Inference Systems for Air Conditioning System. *International Journal of Soft Computing and Engineering (IJSCE),* pp. 323-325.

Kennedy, J. & Eberhart, R., 1995. *Particle Swarm Optimization.* Perth, WA, Proceedings of IEEE International Conference on Neural Networks IV, p. 1942–1948.

Khattree, R. & Naik, D. N., 2000. *Multivariate Data Reduction and Discrimination with SAS Software.* 1st ed. s.l.:Wiley-SAS.

Kilem, G. L., 2012. *Handbook of Inter-Rater Reliability: The Definitive Guide to Measuring the Extent of Agreement Among Multiple Raters.* s.l.:Advanced Analytics, LLC.

Kingston, N. & Clark, A., 2014. *Test Fraud: Statistical Detection and Methodology.* 1st ed. New York: Routledge.

King, T. S., Lengerich, R. & Bai, S., 2017. *STAT 509: Cohen's Kappa Statistic for Measuring Agreement.* [Online]
Available at: https://onlinecourses.science.psu.edu/stat509/node/162

Kirkpatrick, S., Gelatt Jr., C. & Vecchi, M., 1983. Optimization by simulated annealing. *Science,* 220(4598), pp. 671-680.

Kohavi, R., 1995. *A study of cross-validation and bootstrap for accuracy estimation and model selection.* Montreal, Quebec, Canada, Morgan Kaufmann Publishers Inc., pp. 1137-1143.

Kohonen, T., 1997. *Self-Organizing Maps.* 2nd ed. Heidelberg: Springer: Series in Information Sciences.

Konovalov, I., 2009. Combining deterministic and statistical approaches for PM10. *Atmospheric Environment,* Issue 43, p. 6425–6434.

Kukkonen, J. et al., 2012. A review of operational, regional-scale, chemical weather forecasting models in Europe. *Atmospheric Chemistry and Physics ,* Volume 12, p. 1–87.

Kumar, A. & Goyal, P., 2013. Forecasting of Air Quality Index in Delhi Using Neural Network Based on Principal Component Analysis. *Pure and Applied Geophysics,* 170(4), pp. 711-722.

Kumar, A. et al., 2016. Evaluation of control strategies for industrial air pollution sources using American Meteorological Society/Environmental Protection Agency Regulatory Model with simulated meteorology by Weather Research and Forecasting Model. *Journal of Cleaner Production,* Issue 116, pp. 110-117.

Kurt, A., Gulbagci, B., Karaca, F. & Alagha, O., 2008. An online air pollution forecasting system using neural networks. *Environment International,* 34(5), pp. 592-598.

Kyriakidis, I., Karatzas, K. & Papadourakis, G., 2009. *Using Preprocessing Techniques in Air Quality forecasting with Artificial Neural Networks.* Thessaloniki, Proceedings of the Fourth International ICSC Symposium on Information Technologies in Environmental Engineering.

Lance, C., Butts, M. & Michels, L., 2006. The sources of four commonly reported cutoff criteria: What did they really say?. *Organizational Research Methods,* 9(2), p. 202–220.

Landis, R. J. & Koch, G. G., 1977. The measurement of observer agreement for categorical data. *Biometrics,* 33(1), pp. 159-174.

Laurent, S. & Violante, F., 2011. Volatility forecasts evaluation and comparison. *WIREs Computational Statistics,* 4(1), pp. 1-12.

Leblebicioğlu, A., 2009. Financial integration, credit market imperfections and consumption smoothing. *Journal of Economic Dynamics and Control,* 32(2), pp. 377-393.

Legates, D. R. & McCabe, G. J., 1999. Evaluating the use of "goodness-of-fit" Measures in hydrologic and hydroclimatic model validation. *Water Resources Research,* 35(1), pp. 233-241.

Legates, D. R. & McCabe, J. G., 2013. A refined index of model performance: a rejoinder. *International Journal of Climatology,* 33(4), pp. 1053-1056.

Leys, C. et al., 2013. Detecting outliers: Do not use standard deviation around the mean, use absolute deviation around the median. *Experimental Social Psychology,* 49(4), p. 764–766.

Li, F., 2010. *Air Quality Prediction in Yinchuan by Using Neural Networks.* s.l., Springer-Verlag, p. 548–557.

Lim, C. P. & Jain, L. C., 2009. Advances in Swarm Intelligence. In: C. P. Lim, L. C. Jain & S. Dehuri, eds. *Innovations in Swarm Intelligence.* Berlin: Springer-Verlag, pp. 1-8.

Lipowsky, H., Staudacher, S., Bauer, M. & Schmidt, K.-J., 2009. Application of Bayesian Forecasting to Change Detection and Prognosis of Gas Turbine Performance. *Engineering for Gas Turbines and Power,* 132(3), pp. 1-6.

Liu, D. H. & Liptak, B. G., 1999. *Air Pollution.* s.l.:CRC Press.

Lu, P. et al., 2015. Supervised Learning with the Artificial Neural Networks Algorithm for Modeling Immune Cell Differentiation. In: Q. N. Tran & H. Arabnia, eds. *Emerging Trends in Computational Biology, Bioinformatics, and Systems Biology: Algorithms and Software Tools (Emerging Trends in Computer Science and Applied Computing).* Boston: Morgan Kaufmann, pp. 1-18.

Makridakis, S. & Hibon, M., 2000. The M3-Competition: results, conclusions and implications. *International Journal of Forecasting,* 16(4), pp. 451-476.

Mamdani, E. H. & Assilian, S., 1975. An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man-Machine Studies,* 7(1), pp. 1-13.

Mandel, D. R. & Barnes, A., 2014. Accuracy of forecasts in strategic intelligence. *PNAS,* 111(30), pp. 10984-10989.

Mansi, S. & Gajanan, B., 2013. Comparison of Mamdani and Sugeno Inference Systems for Dynamic Spectrum Allocation in Cognitive Radio Networks. *Wireless Personal Communications,* 71(2), pp. 805-819.

Mason, S. J. & Stephenson, D. B., 2008. How Do We Know Whether Seasonal Climate Forecasts Are Any Good?. In: A. Troccoli, M. Harrison, D. Anderson & S. J. Mason, eds. *Seasonal Climate: Forecasting and Managing Risk.* s.l.:Springer Netherlands, pp. 259-289.

Mateo, F. et al., 2015. Forecasting Techniques for Energy Optimization in Buildings. In: K. Mehdi, ed. *Encyclopedia of Information Science and Technology.* s.l.:IGI Global, pp. 967-977.

Mathworks, 2008. *Matlab: Neural Network Toolbox, User's Guide.* [Online]
Available at: http://www.mathworks.com/help/pdf_doc/nnet/nnet_ug.pdf

McIntosh, A. & Pontius, J., 2017. Chapter 3 – Air Quality and Atmospheric Science. Στο: *Science and the Global Environment: Case Studies Integrating Science Global Environment.* s.l.:Elsevier, p. 255–359.

Merkle, D. & Middendorf, M., 2005. Swarm Intelligence. In: E. K. Burke & G. Kendall, eds. *Search Methodologies.* s.l.:Springer US, pp. 401-435.

Michie, D., Spiegelhalter, D. & Taylor, C., 2009. *Machine Learning: Neural and Statistical Classification.* s.l.:Overseas Press.

Mielke, W. P. J. & Berry, K. J., 2007. *Permutation Methods: A Distance Function Approach.* 1st ed. s.l.:Springer.

Miller, J., 1991. Reaction time analysis with outlier exclusion: Bias varies with sample size. *The Quarterly Journal of Experimental Psychology,* 43(4), p. 907–912.

Ministry of Environment, Japan, 1997. *Environmental Quality Standards in Japan.* [Online]
Available at: https://www.env.go.jp/en/air/aq/aq.html
[Accessed 1 April 2012].

Mishra, D. & Goyal, P., 2015. Development of artificial intelligence based NO2 forecasting models at Taj Mahal, Agra. *Atmospheric Pollution Research,* Issue 6, pp. 99-106 .

Mokhade, A. S. & Kakde, O. G., 2014. Overview of Selection Schemes In Real-Coded Genetic Algorithms and Their Applications. *Journal of Industrial and Intelligent Information,* 2(1), pp. 71-77.

Montgomery, D. C., Jennings, C. L. & Kulahci, M., 2015. *Introduction to Time Series Analysis and Forecasting.* 2nd ed. s.l.:Wiley.

Morales-Reyes, A. & Erdogan, A. T., 2012. Internal Lattice Reconfiguration for Diversity Tuning in Cellular Genetic Algorithms. *PLoS ONE,* 7(7), p. e41279.

Morse, A. P. et al., 2005. A forecast quality assessment of an end-to-end probabilistic multi-model seasonal forecast system using a malaria model. *Tellus,* 57(3), pp. 464-475.

Moussiopoulos, N. et al., 2010. Environmental, social and economic information management for the evaluation of sustainability in urban areas: A system of indicators for Thessaloniki, Greece. *Cities,* 27(5), p. 377–384.

Moussiopoulos, N., Schluenzen, H. & Louka, P., 2003. Chapter 6: Modelling Urban Air Pollution. In: N. Moussiopoulos, ed. *Air Quality in Cities.* s.l.:Springer-Verlag, pp. 121-154.

Moustris, K., Nastos, P., Larissi, I. & Paliatsos, A., 2012. Application of Multiple Linear Regression Models and Artificial Neural Networks on the Surface Ozone Forecast in the Greater Athens Area, Greece. *Advances in Meteorology,* Volume 2012, pp. 1-8.

Moustris, K. P., Ziomas, I. C. & Paliatsos, A. G., 2010. 3-Day-Ahead Forecasting of Regional Pollution Index for the Pollutants NO2, CO, SO2, and O3 Using Artificial Neural Networks in Athens, Greece. *Water, Air, & Soil Pollution,* 209(1-4), pp. 29-43.

Mukherjee, A. et al., 2012. Effect of simulated herbivory on growth of the invasive weed Hygrophila polysperma: Experimental and predictive approaches. *Biological Control,* 60(3), pp. 271-279.

Murphy, A. H. & Winkler, R. L., 1987. A general framework for forecast verification. *Monthly Weather Review,* 115(7), pp. 1330-1338.

Muthukrishnan, R. & Radha, M., 2010. M-Estimators in Regression Models. *Journal of Mathematics Research,* 2(4), pp. 23-27.

Nash, J. E. & Sutcliffe, J. V., 1970. River flow forecasting through conceptual models, Part I, A discussion of principles. *Journal of Hydrology,* 10(3), pp. 282-290.

National Research Council, 1999. *Ozone-Forming Potential of Reformulated Gasoline.* Washington, DC: The National Academies Press.

Nelder, J. A. & Wedderburn, R. W., 1972. Generalized Linear Models. *Journal of the Royal Statistical Society,* 135(3), pp. 370-384.

Neto, J., Torres, P. M., Ferreira, F. & Boavida, F., 2009. Lisbon air quality forecast using statistical methods. *International Journal of Environment and Pollution,* 39(3-4), pp. 333-339.

Nielsen, M. A., 2015. *Neural Networks and Deep Learning.* s.l.:Determination Press.

NIST/SEMATECH, 2003. *e-Handbook of Statistical Methods.* [Online]
Available at: https://www.itl.nist.gov/div898/handbook/pmd/section1/pmd14.htm
[Accessed 10 April 2078].

Oftedal, B. et al., 2003. Traffic related air pollution and acute hospital admission for respiratory diseases in Drammen, Norway 1995–2000.. *European Journal of Epidemiology,* 18(7), pp. 671-676.

Ostfeld, A., 2011. *Ant Colony Optimization - Methods and Applications.* Rijeka, Croatia: InTech.

Otte, T. L. et al., 2005. Linking the Eta Model with the Community Multiscale Air Quality (CMAQ) Modeling System to Build a National Air Quality Forecasting System. *Weather and Forecasting,* Volume 20, pp. 367-384.

Papadimitriou, C. H., 1994. *Computational Complexity.* 1st ed. s.l.:Addison Wesley.

PassMark Software, 2015. *CPU Benchmark.* [Online]
Available at: https://cpubenchmark.net/cpu.php?id=1038
[Accessed 19 March 2015].

Pay, M., Martínez, F., Guevara, M. & Baldasano, J., 2014. Air quality forecasts on a kilometer-scale grid over complex Spanish terrains. *Geosci. Model Dev.,* Issue 7, p. 1979–1999.

Phalen, R. F. & Phalen, R. N., 2013. *Introduction to Air Pollution Science: A Public Health Perspective.* 2nd ed. s.l.:Jones & Bartlett Learning.

Poupkou, A., Nastos, P., Melas, D. & Zerefos, C., 2011. Climatology of Discomfort Index and Air Quality Index in a Large Urban Mediterranean Agglomeration. *Water, Air, & Soil Pollution,* 222(1), p. 163–183.

Prasad, K., Gorai, A. K. & Goyal, P., 2016. Development of ANFIS models for air quality forecasting and input optimization for reducing the computational cost and time. *Atmospheric Environment,* Volume 128, pp. 246-262.

Pyle, D., 1999. *Data preparation for data mining.* 1st ed. San Francisco: Morgan Kaufmann.

Rader, C. M. & Brenner, N. M., 1976. A new principle for fast Fourier transformation. *EEE Transactions on Acoustics, Speech, & Signal Processing,* 24(3), p. 264–265.

Rani, D., Jain, S. K., Srivastava, D. K. & Perumal, M., 2013. Genetic Algorithms and Their Applications to Water Resources Systems. In: X. Yang, A. H. Gandomi, S. Talatahari & A. H. Alavi, eds. *Metaheuristics in Water, Geotechnical and Transport Engineering.* Oxford: Elsevier, pp. 43-78.

Rees, D. G., 1987. *Foundations of Statistics.* s.l.:Chapman and Hall/CRC.

Refaeilzadeh, P., Tang, L. & Liu, H., 2009. Cross-Validation. In: L. Liu & O. M. Tamer, eds. *Encyclopedia of Database Systems.* s.l.:Springer, p. 532–538.

Richards, M. et al., 2006. Grid-based analysis of air. *Ecological Modelling,* Issue 194, p. 274–286.

Roohollah, N., Gholamali, H., Khosro, A. & Babak, N. A., 2010. Uncertainty analysis of developed ANN and ANFIS models in prediction of carbon monoxide daily concentration. *Atmospheric Environment,* 44(4), pp. 476-482.

Rousseeuw, P. J. & Croux, C., 1993. Alternatives to the median absolute deviation. *Journal of the American Statistical Association,* 88(424), p. 1273–1283.

Rukundo, O. & Cao, H., 2012. Nearest Neighbor Value Interpolation. *International Journal of Advanced Computer Science and Applications,* 3(4), pp. 25-30.

Russo, A., Raischel, F. & Lind, P. G., 2013. Air quality prediction using optimal neural networks with stochastic variables. *Atmospheric Environment,* Volume 79, pp. 822-830.

Saini, L. M. & Soni, M. K., 2002. Artificial neural network based peak load forecasting using levenberg-marquardt and quasi-Newton methods. *IEE Proceedings of Generation, Transmission and Distribution,* 149(5), p. 578–584.

San José, R. et al., 2006. *AIR QUALITY MODELLING: STATE-OF-THE-ART.* Burlington, Vermont, USA, s.n.

Savitzky, A. & Golay, M. J., 1964. Smoothing and Differentiation of Data by Simplified Least Squares Procedures. *Analytical Chemistry,* 36(8), p. 1627–1639.

Schmidhuber, J., 2015. Deep learning in neural networks: An overview. *Neural Networks,* Volume 61, pp. 85-117.

Schwefel, H. P., 1995. *Evolution and Optimum Seeking.* 1st ed. New York: Wiley-Interscience.

Seinfeld, J. H. & Pandis, S. N., 2006. *Atmospheric Chemistry and Physics, From Air Pollution to Climate Change.* 2nd ed. Hoboken, New Jersey: Wiley & Sons, Inc..

SEPA, S. E. P. A., 2017. *The chemistry of air pollution.* [Online]
Available at: https://www.sepa.org.uk/media/120465/mtc_chem_of_air_pollution.pdf
[Accessed 17 November 2017].

Shiva, S. M. & Khare, M., 2006. Artificial neural network approach for modelling nitrogen dioxide dispersion from vehicular exhaust emissions. *Ecological Modelling,* 190(1-2), p. 99–115.

SILAM, S. f. I. m. o. A. c., 2014. *SILAM User Guide v4.5.* [Online]
Available at: http://silam.fmi.fi/doc/SILAM_v4_5_4_userGuide_general.pdf
[Accessed 19 November 2017].

Sivanandam, S. N., Sumathi, S. & Deepa, S. N., 2007. *Introduction to Fuzzy Logic using MATLAB.* Berlin: Springer.

Slini, T., Karatzas, K. & Mousiopoulos, N., 2003. Correlation of air pollution and meteorological data using neural networks. *Int. J. Environment and Pollution,* 20(1-6), pp. 218-229.

Slini, T., Karatzas, K. & Moussiopoulos, N., 2002. Statistical analysis of environmental data as the basis of forecasting: an air quality application. *The Science of the Total Environment,* Volume 288, pp. 227-237.

Sokhi, R. S. et al., 2006. Prediction of ozone levels in London using the MM5-CMAQ modelling system. *Environmental Modelling and Software* , 21(4), pp. 566-576.

Solaiman, T. A., Coulibaly, P. & Kanaroglou, P., 2008. Ground-level ozone forecasting using data-driven methods. *Air Qual Atmos Health,* 1(4), p. 179–193.

Solomatine, D., See, L. & Abrahart, R., 2009. Data-Driven Modelling: Concepts, Approaches and Experiences. In: R. Abrahart, L. See & D. Solomatine, eds. *Practical Hydroinformatics. Water Science and Technology Library.* Berlin, Heidelberg: Springer.

Sousa, S. I., Martins, F. G., Alvim-Ferraz, M. C. & Pereira, M. C., 2007. Multiple linear regression and artificial neural networks based on principal components to predict ozone concentrations. *Environmental Modelling & Software,* 22(1), pp. 97-103.

Spearman, C., 1904. "General Intelligence," Objectively Determined and Measured. *The American Journal of Psychology,* 15(2), p. 201–292.

Specht, D. F., 1991. A general regression neural network. *IEEE Transactions on Neural Networks,* 2(6), pp. 568-576.

Sugeno, M., 1985. *Industrial applications of fuzzy control.* 1st ed. New York: Elsevier Science Ltd.

Sunil, G., Shiva, N. S., Mukesh, K. & Isha, K., 2015. Urban air quality management–A review. *Atmospheric Pollution Research,* Volume 6, pp. 286-304.

Sydow, A., 2010. *Environmental Systems - Volume III.* Sydow, Achim; ed. Oxford: EOLSS Publications.

The MathWorks, I., 2015. [Online]
Available at: http://www.mathworks.com/help/stats/feature-transformation.html#f72219
[Accessed 5 5 2015].

The World Bank & Institute for Health Metrics and Evaluation, U. o. W. S., 2016. *The Cost of Air Pollution: Strengthening the Economic Case for Action.* Washington: International Bank for Reconstruction and Development/The World Bank.

Theil, H., 1958. *Economic Forecasts and Policy.* s.l.:North Holland Publishing Company.

Theil, H., 1966. *Applied economic forecasting.* Chicago: Rand McNally.

Thiel, H., 1966. *Applied Economic Forecasting.* s.l.:Elsevier Science.

Thornes, N., 2006. *Using Spearman's Rank Correlation Coefficient in Coursework,* s.l.: Geofile Online.

Timothy, R. J., 2010. *Fuzzy Logic with Engineering Applications.* 3rd ed. Singapore: John Wiley & Sons, Ltd..

Trudeau, R. J., 1993. *Introduction to Graph Theory.* New York: Dover Publications Inc..

Tzima, F., Karatzas, K., Mitkas, P. & Karathanasis, S., 2007. *Using data-mining techniques for PM10 forecasting in the metropolitan area of Thessaloniki, Greece.* s.l., s.n., pp. 2752-2757.

US EPA, U. S. E. P. A., 2017a. *Managing Air Quality - Human Health, Environmental and Economic Assessments.* [Online]
Available at: https://www.epa.gov/haps/hazardous-air-pollutants-sources-and-exposure
[Accessed 02 July 2017].

US EPA, U. S. E. P. A., 2017b. *Managing Air Quality - Human Health, Environmental and Economic Assessments.* [Online]
Available at: https://www.epa.gov/air-quality-management-process/managing-air-quality-human-health-environmental-and-economic
[Accessed 25 October 2017].

US EPA, U. S. E. P. A., 2018. *Managing Air Quality - Human Health, Environmental and Economic Assessments.* [Online]
Available at: https://www.epa.gov/criteria-air-pollutants/naaqs-table
[Accessed 21 March 2018].

Varotsos, C., Ondov, J. & Efstathiou, M., 2005. Scaling properties of air pollution in Athens, Greece and Baltimore, Maryland. *Atmospheric Environment,* 39(22), pp. 4041-4047.

Venayagamoorthy, G. K. & Harley, R. G., 2007. *Swarm Intelligence for Transmission System Control.* Tampa, Florida, USA, IEEE Power System Society General Meeting, pp. 1-4.

Vens, C. & Costa, F., 2011. *Random Forest Based Feature Induction.* s.l., IEEE 11th International Conference on Data Mining.

Voukantsis, D. et al., 2011. Intercomparison of air quality data using principal component analysis, and forecasting of PM10 and PM2.5 concentrations using artificial neural networks, in Thessaloniki and Helsinki. *Science of the Total Environment,* 409(7), p. 1266–1276.

Voukantsis, D. et al., 2010. Forecasting daily pollen concentrations using data-driven modeling methods in Thessaloniki, Greece. *Atmospheric Environment,* 44(39), pp. 5101-5111.

Wasserman, P. D., 1993. *Advanced Methods in Neural Computing.* New York: Van Nostrand Reinhold.

Watterson, I. G., 1996. Non-Dimensional Measures of Climate Model Performance. *International Journal of Climatology,* 16(4), pp. 379-391.

Weather Underground, 2014. [Online]
Available at: http://www.wunderground.com/
[Accessed 11 November 2014].

Webb, A. R., 2002. *Statistical Pattern Recognition.* 2nd ed. s.l.:Wiley.

Weisstein, E. W., 2015. *MathWorld--A Wolfram Web Resource.* [Online]
Available at: http://mathworld.wolfram.com/LeastSquaresFittingPolynomial.html
[Accessed 5 5 2015].

Weisstein, E. W., 2015. *MathWorld--A Wolfram Web Resource.* [Online]
Available at: http://mathworld.wolfram.com/Covariance.html
[Accessed 2 April 2015].

Weisstein, E. W., 2017. *"Covariance." From MathWorld.* [Online]
Available at: http://mathworld.wolfram.com/Covariance.html

West, K. D., 2006. Forecast Evaluation. In: E. Graham, G. Clive & A. Timmermann, eds. *Handbook of Economic Forecasting.* s.l.:Elsevier B.V., pp. 99-134.

Whitley, D., 1989. *The GENITOR Algorithm and Selection Pressure: Why RankBased Allocation of Reproductive Trials is Best.* s.l., Proceedings of the Third International Conference on Genetic Algorithms, pp. 116-121.

WHO, 2005. *Air quality guidelines for particulate matter, ozone, nitrogen dioxide and sulfur dioxide.* Geneva, Switzerland: WHO Pess.

WHO, 2008. *Training Package for the Health Sector, Children's Health and the Environment.* [Online]
Available at: http://www.who.int/ceh/capacity/Outdoor_air_pollution.pdf

Willmott, C. J., 1981. On the validation of models. *Physical Geography,* 2(2), pp. 184-194.

Willmott, C. J. et al., 1985. Statistics for the Evaluation and Comparison of Models. *Journal of Geophysical Research,* 90(C5), pp. 8995-9005.

Willmott, C. J., Robeson, S. M. & Matsuura, K., 2011. A refined index of model performance. *International Journal of Climatology,* 32(13), p. 2088–2094.

Wong, D. et al., 2012. WRF-CMAQ two-way coupled system with aerosol feedback: software development and preliminary results. *Geosci. Model Dev.,* 5(2), p. 299–312.

Xiong, L. H., Guo, S. L. & O'Connor, K. M., 2006. Smoothing the seasonal means of rainfall and runoff in the linear perturbation model (LPM) using the kernel estimator. *J Hydrol,* Volume 324, p. 266–282.

Yadav, J. Y., Kharat, V. & Deshpande, A., 2011. *Fuzzy Description of Air Quality: A Case Study.* Banff, Canada, Rough Sets and Knowledge Technology, Lecture Notes in Computer Science, pp. 420-427.

Yang, Z. & Yang, Z., 2014. Artificial Neural Networks. In: A. Brahme, ed. *Comprehensive Biomedical Physics (Reference Module in Biomedical Sciences).* Oxford: Elsevier, pp. 1-17.

Ying, Q. et al., 2007. Verification of a source-oriented externally mixed air quality model during a severe photochemical smog episode. *Atmospheric Environment,* 41(7), pp. 1521-1538.

Yong-Sang, C. & Bo-Ram, K., 2012. PM10 weekly periodicity in Beijing and Tianjin, 2000–2009: Anthropogenic or natural contributions?. *Atmospheric Environment,* Volume 55, pp. 49-55.

Yoon, Y. & Kim, Y.-H., 2013. Geometricity of genetic operators for real-coded representation. *Applied Mathematics and Computation,* 219(23), pp. 10915-10927.

Zadeh, L. A., 1965. Fuzzy Sets. *Information and Control,* 8(3), p. 338–353.

Zajonc, R. B., 1975. Birth Order and Intelligence: Dumber by the Dozen. *Psychology Today,* Volume 8, pp. 37-43.

Zajonc, R. B. & Bargh, J., 1980. The confluence model: Parameter estimation for six divergent data sets on family factors and intelligence. *Intelligence,* 4(4), pp. 349-361.

Zhang, Y. et al., 2012. Real-time air quality forecasting, part I: History, techniques, and current status. *Atmospheric Environment,* Issue 60, pp. 632-655.

Zhang, Z., 1997. Parameter Estimation Techniques: A Tutorial with Application to Conic Fitting. *Image and Vision Computing Journal,* 15(1), pp. 59-76.

Ziomas, I. et al., 1995. Forecasting peak pollutant levels from meteorological variables. *Atmospheric Environment,* Volume 29, p. 3703–3711.

Zolghadri, A. et al., 2004. Development of an operational model-based warning system for tropospheric ozone concentrations in Bordeaux, France. *Environmental Modelling & Software,* 19(4), p. 369–382.

Zwol, R. v., 2006. Multimedia Strategies for B3-SDR, Based on Principal Component Analysis. In: N. Fuhr, M. Lalmas, S. Malik & G. Kazai, eds. *Advances in XML Information Retrieval and Evaluation.* Berlin: Springer-Verlag, pp. 540-553.

Appendix I

Abstracts of Publications

This appendix documents the titles, authors involved, status and abstracts of the publications that were produced during this thesis. Table I-1 summarizes the aforementioned publications.

Table I-1: Publications produced during the thesis

Year	Status	Title
2009	Published (Conference)	Using Preprocessing Techniques in Air Quality forecasting with Artificial Neural Networks
2010	Published (Conference)	Predicting QoL Parameters for the Atmospheric Environment in Athens, Greece
2012a	Published (Conference)	Investigation and Forecasting of the Common Air Quality Index in Thessaloniki, Greece
2012b	Published (Journal)	Using artificial intelligence methods to understand and forecast atmospheric quality parameters
2013	Published (Journal)	Evaluation and analysis of artificial neural networks and decision trees in forecasting of common air quality index in Thessaloniki, Greece
2015	Published (Conference)	New Statistical Indices for Evaluating Model Forecasting Performance
2016	Published (Journal)	A Generic Preprocessing Optimization Methodology when Predicting Time-Series Data

Using Preprocessing Techniques in Air Quality forecasting with Artificial Neural Networks

Ioannis Kyriakidis, Kostas D. Karatzas and George Papadourakis

Abstract: Data quality is one of the fundamental issues influencing the performance of any data investigation algorithm. Poor data quality always leads to poor quality results. In the investigation chain, the data selection phase is followed by the preprocessing phase, which results in increased data quality, while in parallel it demands the highest time resources of the overall data investigation chain. The preprocessing phase includes the handling of missing data, handling of the outliers, data de-trending and data smoothing. The methods that are used in the preprocessing phase are usually not sufficiently reported in the literature of environmental data analysis and knowledge extraction. The current paper investigates the performance of several methods in all phases of the preprocessing chain of environmental data, by emphasizing in the use of ICT (Information & Communication Technology) methods for the materialization of such preprocessing tasks, and by making use of the air quality as the environmental domain paradigm.

Bibliographic Reference:

Kyriakidis, I., Karatzas, K. & Papadourakis, G., 2009. Using Preprocessing Techniques in Air Quality forecasting with Artificial Neural Networks. Thessaloniki, Proceedings of the Fourth International ICSC Symposium on Information Technologies in Environmental Engineering, pp. 357-372.

Predicting QoL Parameters for the Atmospheric Environment in Athens, Greece

Ioannis Kyriakidis, Kostas Karatzas, and George Papadourakis

Abstract: Air quality has a direct impact on the quality of life and on the general environment. Understanding and managing urban air quality is a suitable problem domain for the application of artificial intelligence (AI) methods towards knowledge discovery for the purposes of modelling and forecasting. In the present paper Artificial Neural Networks are supplemented by a set of mathematical tools including statistical analysis and Fast Fourier Transformations for the investigation and forecasting of hourly benzene concentrations and highest daily 8 hour mean of (8-HRA) ozone concentrations for two locations in Athens, Greece. The methodology is tested for its forecasting ability. Results verify the approach that has been applied, and the ability to analyze and model the specific knowledge domain and to forecast key parameters that provide direct input to the environmental decision making process.

Bibliographic Reference:

Kyriakidis, I., Karatzas, K. & Papadourakis, G., 2010. Predicting QoL parameters for the Atmospheric Environment in Athens, Greece. Thessaloniki, Springer Academic Publishers, Lecture Notes in Computer Science Series, pp. 457-463.

Investigation and Forecasting of the Common Air Quality Index in Thessaloniki, Greece

Ioannis Kyriakidis, Kostas Karatzas, George Papadourakis, Andrew Ware, and Jaakko Kukkonen

Abstract: Air pollution can affect health and well-being of people and ecosystems. Due to the health risk posed for sensitive population groups, it is important to provide with hourly and daily forecasts of air pollution. One way to assess air pollution is to make use of the Common Air Quality Index (CAQI) of the European Environment Agency (EEA). In this paper we employ a number of Computational Intelligence algorithms to study the forecasting of the hourly and daily CAQI. These algorithms include artificial neural networks, decision trees and regression models combined with different datasets. The results provide with a satisfactory CAQI forecasting performance that may be the basis of an operational forecasting system.

Bibliographic Reference:

Kyriakidis, I. et al., 2012a. Investigation and Forecasting of the Common Air Quality Index in Thessaloniki, Greece. Halkidiki, Greece, Artificial Intelligence Applications and Innovations, pp. 390-400.

Using Artificial Intelligence Methods to Understand and Forecast Atmospheric Quality Parameters

Ioannis Kyriakidis, Kostas Karatzas, George Papadourakis, J Andrew Ware

Abstract: Quality of life is strongly affected by the quality of the environment. Understanding and managing urban air quality is one of the main concerns of city authorities. For this purpose, it is important to extract knowledge and to be able to model the problem under investigation (air pollution), in order to be able to forecasts parameters of interest (pollutant concentrations). In this paper Artificial Neural Networks and Linear Regression models are used together with a set of mathematical tools that include Principal Component Analysis and Fast Fourier Transformations for the investigation and forecasting of hourly Benzene concentrations and highest daily eight hour mean of ozone concentrations for two locations in Athens, Greece. The methodology is evaluated for its forecasting ability. Results verify the suitability of the computational approach employed and the improvement of results in comparison to previous approaches.

Bibliographic Reference:

Kyriakidis, I., Karatzas, K., Papadourakis, G. & Ware, A., 2012b. Using Artificial Intelligence Methods to Understand and Forecast Atmospheric Quality Parameters. Engineering Intelligent Systems, 20(1/2), pp. 137-149.

Evaluation and Analysis of Artificial Neural Networks and Decision Trees in Forecasting of Common Air Quality Index in Thessaloniki, Greece

Ioannis Kyriakidis, Kostas Karatzas, Jaakko Kukkonen, George Papadourakis and Andrew Ware

Abstract: Air quality management in urban areas requires reliable and accurate computational methods for the forecasting of the concentration levels of pollutants. The Common Air Quality Index (CAQI) has been used by the European Environment Agency for assessing air quality in a harmonized way. We have evaluated the values of this index in Thessaloniki, Greece, in 2001–2003, using a wide range of Computational Intelligence (CI) models. We have applied Artificial Neural Networks (ANN) and Decision Trees (DT) for the forecasting of the CAQI, and we compared the results with those obtained via statistical regression models. An extensive number of computational experiments were performed, in which we evaluated the influences of (i) different model architectures, (ii) various input datasets, and (iii) the training methods of these models. Model sensitivity was analyzed, in terms of various modelling options. In total, the performance of 1118 different modelling options was investigated. The best of the evaluated models performed well in forecasting both the hourly and daily values of the CAQI. The use of model ensembles (based on the same algorithms and structures), obtained by a k-fold cross-validation, resulted in more accurate forecasts than using individual models. However, the performance of various ANN and DT models was found to be dependent both on their internal structure and on the methods used for their training. The presented results are expected to be useful in developing and implementing operational air quality management and forecasting systems.

Bibliographic Reference:

Kyriakidis, I. et al., 2013. Evaluation and analysis of artificial neural networks and decision trees in forecasting of common air quality index in Thessaloniki, Greece. Engineering Intelligent Systems, 21(2/3), pp. 111-124.

New Statistical Indices for Evaluating Model Forecasting Performance

Ioannis Kyriakidis, Jaakko Kukkonen, Kostas Karatzas, George Papadourakis and Andrew Ware

Abstract: A large number of statistical measures have been presented in the literature for the statistical analysis of the agreement of the measured and predicted time-series. The goal of this study was to develop new indices that combine the information contained in several existing measures, making it possible to assess more effectively the quality of the forecasting. The capabilities and limitations of 24 measures that have previously been presented in the literature were studied. The upper and lower bounds of the Confidence Interval were used, in order to include forecasting penalties (relative weights). Results show that by using the proposed new forecasting performance indices we can be more confident in the estimation of the forecasting performance than using a single measure. The proposed new indices would be ideal for a forecasting automated system, because no human interaction is needed to combine the information of other measures.

Bibliographic Reference:

Kyriakidis, I. et al., 2015. New Statistical Indices for Evaluating Model Forecasting Performance. Skiathos Island, Greece, 9th International Conference in New Horizons in Industry, Business and Education, pp. 104-118.

A Generic Preprocessing Optimization Methodology when Predicting Time-Series Data

Ioannis Kyriakidis, Kostas Karatzas, Andrew Ware and George Papadourakis

Abstract: A general Methodology referred to as Daphne is introduced which is used to find optimum combinations of methods to preprocess and forecast for time-series datasets. The Daphne Optimization Methodology (DOM) is based on the idea of quantifying the effect of each method on the forecasting performance, and using this information as a distance in a directed graph. Two optimization algorithms, Genetic Algorithms and Ant Colony Optimization were used for the materialization of the DOM. Results shown that the DOM finds a solution which is near the best possible solution in relatively less time than using the traditional optimization algorithms.

Bibliographic Reference:

Kyriakidis, I. et al., 2016. A Generic Preprocessing Optimization Methodology when Predicting Time-Series Data. International Journal of Computational Intelligence Systems, 9(4), pp. 638-651.

Appendix II

Forecasting Verification Indices

This appendix documents the literature review for the twenty-four selected statistical measures, which have commonly been used to evaluate the performance of models that produce forecasts of continuous (numerical) variables. Table II-1 summarizes the notations that were used in the equations of this chapter.

Table II-1: Summarize description of notations

Notation	Description
A_i	The ith actual (observed) value
F_i	The ith forecasted value
$\overline{A}$	The mean of the actual values
$\overline{F}$	The mean of the forecasted values
s_A	The standard deviation of the actual values
s_F	The standard deviation of the forecasted values

Bias

The Bias indicates the average direction of error. Bias is referred to by several names, for example as Mean Error, Forecast Error, or as Systematic Error. There are two ways to express the error of each actual (A) and forecasted (F) values pair. One way is A minus F

(Equation II-1) or alternatively F minus A (Equation II-2). Green & Tashman (2008) survey shows that researchers (of the International Institute of Forecasters) argue in this manner. The researchers who preferred F - A all reasoned that:

"It was more intuitive that a positive error represented an over-forecast and a negative error an under-forecast. F - A is also more consistent with concepts of bias."

And the researchers who preferred the A - F formulation argued that:

"Statistical convention, ease of statistical calculation, investment in software that adhered to statistical convention and plain pragmatism provided justification. Two advocates of A - F also suggested that this version is intuitive when assessing performance against a budget or plan, because a positive value indicates that a budget has been exceeded or a plan has been surpassed."

In both cases Bias it should not be used by itself because it provides no measure of the error variance (Armstrong, 1985).

$$Bias = \frac{1}{n}\sum_{i=1}^{n}\left(A_i - F_i\right) \qquad \text{Researchers } \textit{who preferred A - F} \qquad \text{(II-1)}$$

$$Bias = \frac{1}{n}\sum_{i=1}^{n}\left(F_i - A_i\right) \qquad \text{Researchers } \textit{who preferred F - A} \qquad \text{(II-2)}$$

Normalized Bias

The Normalized Bias (NB) (Equation (II-3)) is a measure of the over or under-prediction of a variable and is often expressed as a percentage. Positive values indicate over-prediction and negative values indicate under-prediction.

$$NB = \frac{\frac{1}{n}\sum_{i=1}^{n}\left(F_i - A_i\right)}{\max(A) - \min(A)} \qquad \text{(II-3)}$$

Mean Fractional Bias

Fractional bias (FB) (Ying, et al., 2007) is a non-linear measure which is used to represent the difference between the forecasted average and the actual average. FB can be used as a substitute of Normalized Bias because Normalized Bias becomes very large when a minimum threshold is not used. However, its non-linear nature means that the statistics of FB, such as the variance, can only be determined by using sampling methods (e.g. bootstrap). Mean Fractional bias (MFB) varies between +2 (extreme over-prediction) and -2 (extreme under-prediction), where a value of 0 indicates an excellent model in terms of the average values. The MFB is a useful indicator because it has the advantage of equally weighting positive and negative bias estimates. It has also the advantage of not considering observations as the true value. MFB in shown in Equation (II-4), in addition, Table II-2 shows examples of MFB values and their interpretation.

$$MFB = \frac{2}{n}\sum_{i=1}^{n}\frac{F_i - A_i}{F_i + A_i} \tag{II-4}$$

Table II-2: Examples of Mean Fractional Bias values and their interpretation

MFB Value	Interpretation
0.60	Indicates 30% over-prediction (the maximum value is 2)
-0.60	Indicates 30% under-prediction (the minimum value is -2)

Mean Percentage Error

The mean percentage error (MPE) is the computed average of percentage errors by which estimated forecasts differ from actual values of the quantity being forecast. This average allows positive and negative percentage errors to cancel one another. Therefore, it is sometimes used as a measure of bias in the application of a forecasting method. Because the Bias can be defined in two different ways, the MPE can be defined by Equation (II-5) for researchers who preferred F – A, or can be defined by Equation (II-6) for researchers who preferred A - F.

$$MPE = \frac{1}{n}\sum_{i=1}^{n}\frac{F_i - A_i}{A_i} \tag{II-5}$$

$$MPE = \frac{1}{n}\sum_{i=1}^{n}\frac{A_i - F_i}{A_i} \tag{II-6}$$

Mean Absolute Error

The Mean Absolute Error (MAE) as shown in Equation (II-7) indicates the average of the magnitude of the errors. It is the mean of the absolute differences between the forecasts and observations. MAE can be referred as a Linear Measure because each error has the same weight. In contrary to Bias, MAE does not indicate the direction of the error, but only the magnitude. It has to be noticed that MAE is sensitive to outlier errors.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|A_i - F_i\right| \tag{II-7}$$

Mean Absolute Percentage Error

The Mean Absolute Percent Error (MAPE) is a very popular measure, that corrects the "cancelling out" effects of MPE measure and also because it is a percentage error, it can be used to compare the error of time series that differ in level. MAPE has two major disadvantages in practical application, as it can be easily seen in Equation (II-8). The first one is when there are zero observed values (in the denominator) there will be a division by zero and second, when the predicted and observed values are the same (having a perfect fit), MAPE is zero. Furthermore, when the Actual value is not zero, but quite small, the MAPE will often take on extreme values. Thus, MAPE is scale sensitive and should not be used when working with low-volume data.

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{A_i - F_i}{A_i}\right| \tag{II-8}$$

Symmetric Mean Absolute Percentage Error

The Symmetric Mean Absolute Percentage Error (sMAPE) is a variation on the MAPE measure. The sMAPE is defined in different ways in the literature. Two examples of sMAPE can be seen in Equation (II-9) and Equation (II-10).

$$sMAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|A_i - F_i|}{(A_i - F_i)/2}$$ (Armstrong, 1985) (Makridakis & Hibon, 2000) (II-9)

$$sMAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|A_i - F_i|}{(|A_i| - |F_i|)/2}$$ (Andrawis & Atiya, 2009) (II-10)

Symmetric MAPE of Equation (II-9) as defined by Armstrong (1985) and Makridakis & Hibon (2000) solves (a) the problem of large errors when the actual (observed) values are close to zero and (b) the large difference between the absolute percentage errors when the actual (observed) values are greater than forecasted (predicted) and vice versa. However, if the actual value A_i is zero, the forecast F_i is likely to be close to zero. Thus, the measurement will still involve division by a number close to zero.

Mean Squared Error

The Mean Squared Error (MSE) indicates the average of the squared of the errors. It is a widely-used measure of forecast accuracy that depends on bias, resolution, and uncertainty (Jolliffe & Stephenson, 2012). MSE is a quadratic function; therefore, it is sensitive to large errors, to large variance of errors and on outlier errors. MSE can be seen in Equation (II-11).

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(A_i - F_i)^2$$ (II-11)

Normalized Mean Squared Error

The Normalized Mean Square Error (NMSE) is an estimator of the overall deviations between forecasted and actual values. If the NMSE is greater than 1, then the forecasts are worse than the mean of the actual values. If the NMSE is less than 1, then the forecasts are better than the mean of the actual values. Because of the squared differences, NMSE is sensitive to extreme values. NMSE can be seen in Equation (II-12).

$$NMSE = \frac{\sum_{i=1}^{n} \left(A_i - F_i \right)^2}{\sum_{i=1}^{n} \left(A_i - \overline{A} \right)^2} \qquad \text{(II-12)}$$

Root Mean Squared Error

The Root Mean Squared Error (RMSE) indicates the magnitude of the error and retains the variable’s unit. It is the squared root of the MSE. RMSE has similar properties with MSE; therefore, it is sensitive to large errors, to large variance of errors and on outlier errors. RMSE can be seen in Equation (II-13).

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} \left(A_i - F_i \right)^2} \qquad \text{(II-13)}$$

RMSE, Systematic and Unsystematic

Willmott (1981) suggests the decomposition of the RMSE in the systematic RMSE component, which is due to the model performance and the predictors included, and unsystematic RMSE component that is due to residuals, factors that cannot be controlled.

The Systematic RMSE (RMSE$_s$) as shown in Equation (II-14) measures the linear (or systematic) bias of prediction, which is the difference between the regression line of the observed and predicted observations. This error can usually be corrected by parameter adjustment.

The Unsystematic RMSE (RMSE$_u$) as shown in Equation (II-15) measures the model's unsystematic bias, which is the random error about the regression line of the predicted observations.

A "good" model is considered to have low all RMSE values and a higher unsystematic RMSE than the systematic RMSE.

$$RMSE_s = \left(\frac{1}{n} \sum_{i=1}^{n} \left(\hat{F}_i - A_i \right)^2 \right)^{1/2} \qquad \text{(II-14)}$$

$$RMSE_u = \left(\frac{1}{n} \sum_{i=1}^{n} \left(\hat{F}_i - F_i \right)^2 \right)^{1/2} \tag{II-15}$$

Where:

$\hat{F}_i = a + bA_i$,where "a" and "b" are parameters according to the least squares linear regression between A and F.

Equation (II-16) shows the relationship between RMSE and its components.

$$RMSE^2 = RMSE_s^2 + RMSE_u^2 \tag{II-16}$$

Linear Correlation Coefficient

The Linear Correlation Coefficient (r) measures the strength and the direction of a linear relationship between two variables, and receives a value between -1 and 1. The linear correlation coefficient only measures linear relationships. Therefore, a correlation of 0 does not mean zero relationship between two variables, but it means zero linear relationship. The greater the absolute value of linear correlation coefficient, the stronger the linear relationship. The linear correlation coefficient is sometimes denoted as ρ or R, also sometimes referred to as Pearson’s r, in honour of its developer Karl Pearson. The r can be seen in Equation (II-17).

$$r = \frac{\frac{1}{n} \sum_{i=1}^{n} \left(A_i - \overline{A} \right) \left(F_i - \overline{F} \right)}{\sqrt{\frac{1}{n} \sum_{i=1}^{n} \left(A_i - \overline{A} \right)^2 \frac{1}{n} \sum_{i=1}^{n} \left(F_i - \overline{F} \right)^2}} \tag{II-17}$$

Coefficient of Determination

The Coefficient of Determination (r^2) measures the percent of the data that is the closest to the line of best fit and receives a value between 0 and 1. Therefore, a Coefficient of Determination $r^2 = 0.80$ means that 80% of the total variation in A (actual data) can be explained by the linear relationship between A and F.

Spearman’s Rank Correlation Coefficient

Spearman’s rank correlation coefficient or Spearman’s rho was proposed by Charles Spearman (in 1904) as a measure of statistical dependence between two variables, as presented in Equation (II-18). Spearman’s rank correlation coefficient is denoted as r_s or by the Greek letter ρ (rho). r_s indicates how well the relationship between two variables can be described using a monotonic function. Spearman correlation of +1 (positive correlation) or -1 (negative correlation) occurs when each of the variables is a perfectly described by a monotone function of the other.

$$r_s = 1 - \left(\frac{6\sum_{i=1}^{n}\left(rankF_i - rankA_i\right)^2}{n(n^2-1)} \right) \quad \text{(II-18)}$$

Where:

The rank values (for each $rankF$ and $rankA$) are calculated by giving a rank of 1 in the lowest value (e.g. the lowest value of the forecasted values, for $rankF$), a rank of 2 in the next lowest value, and so on. When two or more values are the same, the ranks of those values are changed by calculating the mean of the existing ranks (as if those values have been ranked normally). These are called tied ranks.

The main limitation of Spearman’s Rank Correlation Coefficient is due to the ranking of the two data sets (Thornes, 2006): it simply places the values in numerical order; it pays no regard to the magnitude of the differences between the values.

Coefficient of Efficiency

The Coefficient of Efficiency (E) defined by Nash & Sutcliffe (1970) which ranges from minus infinity to 1.0, the higher values indicating better agreement. An efficiency of zero (E = 0) indicates that the model predictions are as accurate as the mean of the observed data, whereas an efficiency less than zero (E < 0) occurs when the observed mean is a better predictor than the model. This measure is also referred to in the literature as Modelling Efficiency (EF) (Mukherjee, et al., 2012) (Alexandris, et al., 2008), and can be seen in Equation (II-19).

The coefficient of efficiency represents an improvement over the coefficient of determination (r^2) for model evaluation purposes, in that it is sensitive to differences in the observed and predicted means and variances. However, because of the squared differences, E is overly sensitive to extreme values, as is r^2 (Legates & McCabe, 1999).

$$E = 1 - \frac{\sum_{i=1}^{n}\left(A_i - F_i\right)^2}{\sum_{i=1}^{n}\left(A_i - \overline{A}\right)^2} = 1 - NMSE \qquad \text{(II-19)}$$

Index of Agreement

The Index of Agreement (d) developed by Willmott (1981) measures the degree to which a model's predictions are error free and not the correlation or association in the formal sense. The d of Equation (II-20) varies between 0 and 1, where a value of 1 indicates a perfect match, and 0 indicates no agreement at all. Willmott (1981) mentions that d is a standardized measure in order that (1) it may be easily interpreted and (2) cross-comparisons of its magnitudes for a variety of models, regardless of units, can readily be made. The relationships described by d tend to complement the information contained in RMSE, $RMSE_s$, and $RMSE_u$, because of its dimensionless nature.

$$d = 1 - \frac{\sum_{i=1}^{n}\left|F_i - A_i\right|^2}{\sum_{i=1}^{n}\left(\left|F_i - \overline{A}\right| + \left|A_i - \overline{A}\right|\right)^2} \qquad \text{(II-20)}$$

Willmott, et al. (1985) suspect that squaring the errors prior to summing them was a problem. This is because when larger errors are squared, their influence on the sum-of-squared errors is very strong. Legates & McCabe (1999) also mention that d is overly sensitive to extreme values due to the squared differences. Willmott, et al. (1985) develop a version of d that was based upon sums of the absolute values of the errors (d_1), and he preferred to use d_1 than d. Nevertheless, the overall range of d_1 remained somewhat narrow to resolve adequately the great variety of ways that F can differ from A (Willmott, et al., 2011), as can be seen in Equation (II-21).

$$d_1 = 1 - \frac{\sum_{i=1}^{n} |F_i - A_i|}{\sum_{i=1}^{n} \left(|F_i - \bar{A}| + |A_i - \bar{A}| \right)} \qquad \text{(II-21)}$$

Willmott, et al. (2011) recently develop a refined index of agreement (d_r) that receives a value between -1 and 1, which can be seen in Equation (II-22). This new index is double the range of d or d_1. The refined index of agreement (d_r) indicates the sum of the magnitudes of the differences between the model-predicted and observed deviations about the observed mean relative to the sum of the magnitudes of the perfect model and observed deviations about the observed mean. Table II-3 shows examples of d_r values and their interpretation.

Table II-3: Examples of d_r values and their interpretation

d_r Value	Interpretation
0.5	Indicates that the sum of the error-magnitudes is one-half of the sum of the perfect-model-deviation and observed-deviation magnitudes.
0.0	Signifies that the sum of the magnitudes of the errors and the sum of the perfect-model-deviation and observed-deviation magnitudes are equivalent.
-0.5	Indicates that the sum of the error-magnitudes is twice the sum of the perfect-model-deviation and observed-deviation magnitudes.

Willmott, et al. (2011) mention that Legates and McCabe's measure (E_1) is monotonically and functionally related to the d_r index, and also when E_1 is positive it is equivalent to d_r with c = 1. Willmott, et al. (2011) suggests that c = 2 is a better scaling, because it balances the number of deviations evaluated within the numerator and within the denominator of the fractional part. Legates & McCabe (2013) argues that the refined index of agreement, d_r, proposed by Willmott, et al. (2011) exhibits several distinct flaws that make its utility less favourable.

$$d_r = \begin{cases} 1-\dfrac{\sum_{i=1}^{n}\left|F_i - A_i\right|}{c\sum_{i=1}^{n}\left|A_i-\overline{A}\right|}, when \\ \sum_{i=1}^{n}\left|F_i - A_i\right| \le c\sum_{i=1}^{n}\left|A_i-\overline{A}\right| \\ \dfrac{c\sum_{i=1}^{n}\left|A_i-\overline{A}\right|}{\sum_{i=1}^{n}\left|F_i - A_i\right|}-1, when \\ \sum_{i=1}^{n}\left|F_i - A_i\right| > c\sum_{i=1}^{n}\left|A_i-\overline{A}\right| \end{cases} \tag{II-22}$$

Legates and McCabe

Legates & McCabe (1999) present a generic form of the coefficient of efficiency, E_j (Equation (II-23)). The original equation developed by Nash and Sutcliffe (1970) (Equation (II-19)) with j = 2. Legates and McCabe suggested that j = 1 was a better scaling, because the absolute values are preferable to squared terms since they do not give undue weight to outliers. All the different E_j measures share same characteristics when their value is equal or lower than zero. When the E_j value is zero ($E_j = 0$), it means that such a model has no more ability to predict the observed values than does the observed mean. E_j has no lower bound; the negative values indicate that the model is less effective than the observed mean in predicting the variation in the observations. When E_j values are greater than zero, the interpretation of the results is different for each j. Table II-4 shows examples of E_j values and their interpretation.

Table II-4: Examples of E_j values and their interpretation

E_j Value	**Interpretation**
$E_2 = 0.75$	Indicates that the model can explain three-quarters of the variance in the observed values.
$E_1 = 0.75$	Indicates that the model can explain three-quarters of the absolute-valued differences between the observations and model predictions.

$$E_j = 1-\frac{\sum_{i=1}^{n}\left|A_i - F_i\right|^j}{\sum_{i=1}^{n}\left|A_i-\overline{A}\right|^j} \tag{II-23}$$

Legates and McCabe (1999) mention that better methods exist than the observed mean, in order to define the baseline against which a model should be compared. For example, for most hydrological or hydro-climatological studies, persistence or averages that vary by season or another time period may provide a more appropriate baseline than simply the average of the entire time series (Legates & McCabe, 1999). Thus, E_1 can be rewritten in a "baseline adjusted" form (E'$_1$) (Equation (II-24)).

$$E_1^{'} = 1 - \frac{\sum_{i=1}^{n} |A_i - F_i|}{\sum_{i=1}^{n} |A_i - \overline{A}^{'}|} \qquad \text{(II-24)}$$

Where:

$\overline{A}^{'}$ is the baseline value of the time series against which the model is to be compared. It is usually a function of time and, in some applications, may be a function of other variables as well.

Berry and Mielke

The generalized measure of agreement ($\Re$) of Berry & Mielke (1992), provides for the analysis of a nominal independent variable with any number or combination of nominal, ordinal, or interval dependent variables (Equation (II-25)). The distribution of $\Re$ is usually asymmetric, and the upper and lower bounds depend on both the nature of the data and the structure of δ (described in previous section as MAE). The values of $\Re$ range from negative values to a maximum of 1 for the extreme case when all object measurements within each classified group are identical (i.e., $\delta = 0$). When $\Re = 1$ implies perfect agreement, $\Re > 0$ implies agreement, and $\Re < 0$ implies no agreement. Thus, the closer $\Re$ is to +1, the higher the agreement. As Mielke & Berry (2007) describe, $\Re$ has advantages over the Pearson product-moment correlation coefficient (r), since (a) is a measure of agreement rather than a measure of linearity and (b) is far more stable (i.e., robust) than r since it is based on ordinary Euclidean distances rather than squared Euclidean distances.

$$\Re = 1 - \frac{\delta}{\mu} \qquad \text{(II-25)}$$

Where:

$$\delta = MAE = n^{-1} \sum_{i=i}^{n} \left| F_i - A_i \right| \qquad \text{(II-26)}$$

$$\mu = n^{-2} \sum_{i=1}^{n} \sum_{j=1}^{n} \left| F_j - A_i \right| \qquad \text{(II-27)}$$

Watterson

The Watterson’s (1996) measure of agreement (M) is a symmetric measure that converges linearly to 1 as errors diminish to zero. Thus, positive values indicate "skill", relative to permutations of the data values. Watterson’s argues that other measures based on mean square error; do not have the useful properties of M.

In Equation (II-28), it is clear that if error (MSE) diminishes to zero, the M will not be equal to 1 but it would be equal to 0.7566. Thus, this measure was not used in this work because the upper and lower bounds are not well defined.

$$M = (2/\pi)\sin^{-1}\left\{ 1 - \frac{MSE}{s_F + s_A + (F - \overline{A})^2} \right\} \qquad \text{(II-28)}$$

Factor of Exceedance

The Factor of Exceedance (FOEX) (Sokhi, et al., 2006) is a useful measure to indicate if the forecasting results are over or under-predicted. FOEX varies between -0.5 and 0.5. In order to calculate the percentage of FOEX the Equations (II-29), (II-30) and (II-31) can be used. From Table II-5 it is clear that Equation (II-31) is easier to be interpreted than Equation (II-30). Table II-5 shows examples of FOEX values and their interpretation. It is clear from Equation (II-29) that FOEX cannot be used to evaluate the performance of a forecasting model because it only takes into account the number of over-predictions. Thus, FOEX cannot identify a perfect fit. For example (in the extreme case) when all forecasts have a perfect fit, the $N(F_i > A_i)$ is zero and consequently FOEX=-0.5 and not 0.

$$FOEX = \left[\frac{N(F_i > A_i)}{n} - 0.5 \right]$$

Where: (II-29)

$N(F_i > A_i)$ is the number of over-predictions, i.e. the number of times where $F_i > A_i$

$$FOEX_{P1} = FOEX * 100 \quad \text{(II-30)}$$

$$FOEX_{P2} = FOEX * 200 \quad \text{(II-31)}$$

Table II-5: Examples of Factor of Exceedance values and their interpretation

FOEX Value	FOEX$_{P1}$ Value	FOEX$_{P2}$ Value	Interpretation
-0.5	-50%	-100%	All points lie below the y = x line, i.e. all the forecasted results are under-predicted.
0	0%	0%	The data distribution is optimum, i.e. there are half under- and half over-predictions.
0.5	50%	100%	All points lie above the y = x line, i.e. all the forecasted results are over-predicted.

Theil's Inequality Coefficient

Theil's Inequality Coefficient (TIC) proposed two error measures, at different times and under the same name and symbol "U" which has caused some confusion (Bliemel, 1973). In (1958) Theil proposed Equation (II-32) (denoted as U_1 in this work) and in (1966) Theil proposed the Equation (II-33) (denoted as U_2 in this work) as a measure of forecast quality. The differences in those equations are that a) the F-term is absence in Equation (II-33), and b) their boundaries and their interpretation are different. Table II-6 shows examples of values from the two Theil's equations and their interpretation (Bliemel, 1973). In addition, Table II-6 shows the boundaries of the two Theil's equations, where it is clear that the U_1 varies between 0 and 1, and U_2 varies between 0 and $+\infty$. Theil's U_2 is the RMSE of the proposed forecasting model divided by the RMSE of the "no-change" model (naive). Bliemel (1973) shows that Theil's U_1 has little or no value as a forecasting accuracy index.

$$U_1 = \frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(A_i - F_i)^2}}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}A_i^2} + \sqrt{\frac{1}{n}\sum_{i=1}^{n}F_i^2}} \quad \text{(II-32)}$$

$$U_2 = \frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(A_i - F_i)^2}}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}A_i^{\ 2}}} = \frac{RMSE(\text{mod}el)}{RMSE(\text{"}no-change-\text{mod}el\text{"})} \tag{II-33}$$

Table II-6: Examples of values from the two Theil's equations and their interpretation

U_1 Value	Interpretation of (U_1)	U_2 Value	Interpretation of (U_2)
0	Indicate perfect forecasting.	0	Indicate perfect forecasting.
1	Is the maximum inequality, there is either a negative proportionality or one of the variables (A or F) is zero. For example, it would mean that the forecasted value is always zero.	1	Indicate that the forecasting method is as good as a naive no-change method.
<1	A value lower than 1 (e.g. 0.06) would mean that the standard error of the forecast is 0.06 times as large as the standard value of all forecasts plus the standard value of all outcomes added together.	<1	Indicate the improvement of the forecasting method over a naive no-change method.
-	-	>1	Indicate that the forecasting method is worse than a naive no-change method.

Appendix III

Categorical Verification Scores

Percentage of Agreement

The Percentage of Agreement (PA) is used in order to calculate the percentage of the time that the forecasted values were identical with the observed values, and it is expressed as a numerical value ranging from 0% up to 100%.

Critical Success Index

The Critical Success Index (CSI) also known as Threat Score (TS) verifies how well the events of interest (in this work, Benzene exceedances) are predicted, and it is unaffected by the number of correct forecasts. The CSI values range from 0 (indicates no skill) to 1 which indicates a perfect skill of forecasting the true positive events. Its calculation is based on the Equation (III-1), where *a, b*, *c* and *d* are calculated as shown in Table III-1. The positives or negatives in Table III-1 refer to the occurrence or not of the event of interest (i.e. classification of an occurred episode).The CSI applied in this work does not use the "d" events (true negatives). As a consequence, the CSI is tending to give poorer scores for rare events, but it takes into account both false negative events and false positive events. Therefore, the CSI can be characterized as a more balanced score (Ji, et al., 2010).

$$CSI = \frac{a}{a+b+c} \tag{III-1}$$

Table III-1: The meaning of parameters a, b, c and d of the Critical Success Index formula

Event forecasted	**Event Observed**	
	Yes	**No**
Yes	a: true positives	b: false positives
No	c: false negatives	d: true negatives

Cohen’s Kappa Index

Cohen’s Kappa Index (κ) measures the fraction of correct forecasts after eliminating those forecasts which would be correct due to random chance. For that reason it is generally thought to be a more robust measure than simple percent agreement calculation. The κ score values range from -1 to 1, where 0 indicate no skill and 1 indicate a perfect score.

The κ agreement between categorical variables X and Y can be calculated by Equation (III-2) from the observed and expected frequencies on the diagonal of a square contingency table (King, et al., 2017). Suppose that there are g distinct categorical outcomes for both X and Y from n subjects. In addition, if f_{ij} denote the frequency of the number of subjects with the i^{th} categorical response for variable X and the j^{th} categorical response for variable Y, then the frequencies can be arranged in the g×g Table III-2.

Table III-2: A square (g×g) contingency table

X	**Y**			
	1	**2**	**…**	**g**
1	f11	f12	…	f1g
2	f21	f22	…	f2g
⋮	⋮	⋮	…	⋮
g	fg1	fg2	…	fgg

$$\kappa = \frac{p_0 - p_e}{1 - p_e} \tag{III-2}$$

Where:

The observed proportional agreement between X and Y is defined as:

$$p_0 = \frac{1}{n}\sum_{i=1}^{g} f_{ii} \tag{III-3}$$

The expected agreement by chance is:

$$p_e = \frac{1}{n^2}\sum_{i=1}^{g} f_{i+}f_{+1} \tag{III-4}$$

Where fi+ is the total for the ith row and f$_{+i}$ is the total for the ith column

Appendix IV

Major Air Pollutants and the AQI

Benzene and Ozone

Benzene is a known carcinogen and has been classified as a Group A carcinogen of medium potency by the United States Environmental Protection Agency (USEPA) and a Group 1 carcinogen by the International Agency for Research on Cancer (IARC). It is regulated in Europe on the basis of yearly mean values (not to exceed the levels of 5 μg/m^3) estimated on the basis of hourly measurements. Some parts of the world have however used significantly shorter limit values (for example, the 3 μg/m^3 limit value in Japan (Ministry of Environment, Japan, 1997)). In this work, hourly Benzene concentration levels were addressed, as they are directly related to traffic emissions and thus represent an important parameter for the estimation of the overall environmental burden in a city's atmosphere.

Ozone is a photochemical pollutant with adverse effects on the respiratory and cardiovascular system, with a limit value for the 8-HRA levels of 120μg/m^3 (European Environment Agency and World Health Organ, 2008). Both pollutants are associated with traffic (Benzene being directly emitted and Ozone being a secondary pollutant formulated with the synergy of other car-emitted pollutants) and are found in city areas affected by traffic (Oftedal, et al., 2003).

The Common Air Quality Index

The use of environmental indices comes as a result of scientific initiatives that combine the impacts of various pollutants in order to come up with an aggregated measure, that may be used for the description of the environmental (here atmospheric) quality (Moussiopoulos, et al., 2010). Various air quality indices have been suggested by institutes and authorities in different countries (Kassomenos, et al., 2009) (Poupkou, et al., 2011). The EEA has proposed the Common Air Quality Index (CAQI) to identify the atmospheric quality in respect of public safety and health.

The CAQI is defined in a way that allows for the calculation and the comparison of air quality levels on an hourly or daily basis. The CAQI has five categorical levels, corresponding to a range of values starting from 0 (very low) to >100 (very high), as presented in Figure IV-1. Two types of a CAQI are specified, an urban background index and a traffic (or roadside) related index, in order to better represent areas not directly affected by traffic (the former) as well as areas mostly affected by traffic (the latter). The urban background index takes into account three so-called main pollutants (NO_2, PM_{10} and O_3) and two auxiliary pollutants (SO_2 and CO) while the traffic index takes into account two main pollutants (NO_2 and PM_{10}) and one auxiliary (CO). The use of the auxiliary pollutants in the calculation of the CAQI is decided on the basis of the availability of data. The scientific background of the CAQI as well as the way to calculate it is described in detail (in Elshout, et al., 2012). The calculation method is defined in Table IV-1, where all concentrations are in $\mu g/m^3$, except of CO where the 8-hour moving average value is used.

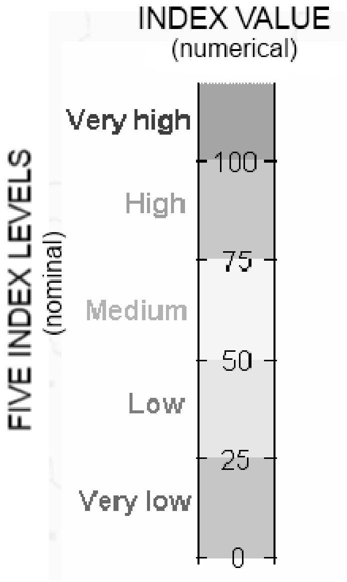


Figure IV-1: The Common Air Quality Index five categorical levels

Table IV-1: The two types of Common Air Quality Index (a background index and a traffic index) proposed by the European Environment Agency (Elshout, et al., 2012)

$SO_2\_Index([SO_2]) = \begin{cases} [SO_2]/2 & 0 < [SO_2] < 100 \\ ([SO_2]/8) + 37.5 & 100 < [SO_2] < 500 \\ [SO_2]/5 & [SO_2] > 500 \end{cases}$	(IV-1)
$NO_2\_Index([NO_2]) = \begin{cases} [NO_2]/2 & 0 < [NO_2] < 100 \\ ([NO_2]/4) + 25 & 100 < [NO_2] < 200 \\ ([NO_2]/8) + 50 & 200 < [NO_2] < 400 \\ [NO_2]/4 & [NO_2] > 400 \end{cases}$	(IV-2)
$\mathrm{O_3\_Index([O_3]) = ([O_3]*5)/12}$	(IV-3)
$\mathrm{PM_{10}\_Index([PM_{10}]) = [PM_{10}]}$	(IV-4)
$CO\_Index([CO]) = \begin{cases} [CO]/200 & 0 < [CO] < 5000 \\ ([CO]/100) - 25 & 5000 < [CO] < 10000 \\ ([CO]/400) + 50 & 10000 < [CO] < 20000 \\ [CO]/200 & [CO] > 20000 \end{cases}$	(IV-5)
Main Background Index = Max(NO_2_Index, PM_{10}_Index, O_3_Index)	(IV-6)
Auxiliary Background Index = Max(NO_2_Index, PM_{10}_Index, O_3_Index, SO_2_Index, CO_Index)	(IV-7)
Main Traffic Index = Max(NO_2_Index, PM_{10}_Index)	(IV-8)
Auxiliary Traffic Index = Max(NO_2_Index, PM_{10}_Index, CO_Index)	(IV-9)

Appendix V

Arsenal of Methods used by the DOM

Methods of Step 1 (Remove Outliers)

The first step in the pre-processing phase is the handling of the outliers. Outliers are data patterns that deviate substantially from the data variation. Outliers result in large errors and consequently large weight updates because of the large deviation from the norm. They may be attributed to measurement error, or they may represent a significant feature within the investigated data. Identifying outliers and deciding what to do with them depends on the understanding of the data, information on their source and knowledge concerning the scientific domain they represent and the range of values that the associated parameters are expected to have.

Detecting the presence of outliers is critical because they have a strong influence on the estimates of the data model parameters that are fitted. This could lead to wrong scientific conclusions and inaccurate predictions. Figure V-1 demonstrates how the (fitted) function is pulled towards the outlier in an attempt to reduce the training error. As a result, the generalization capacity of any model being developed to describe the data deteriorates (Engelbrecht, 2007).

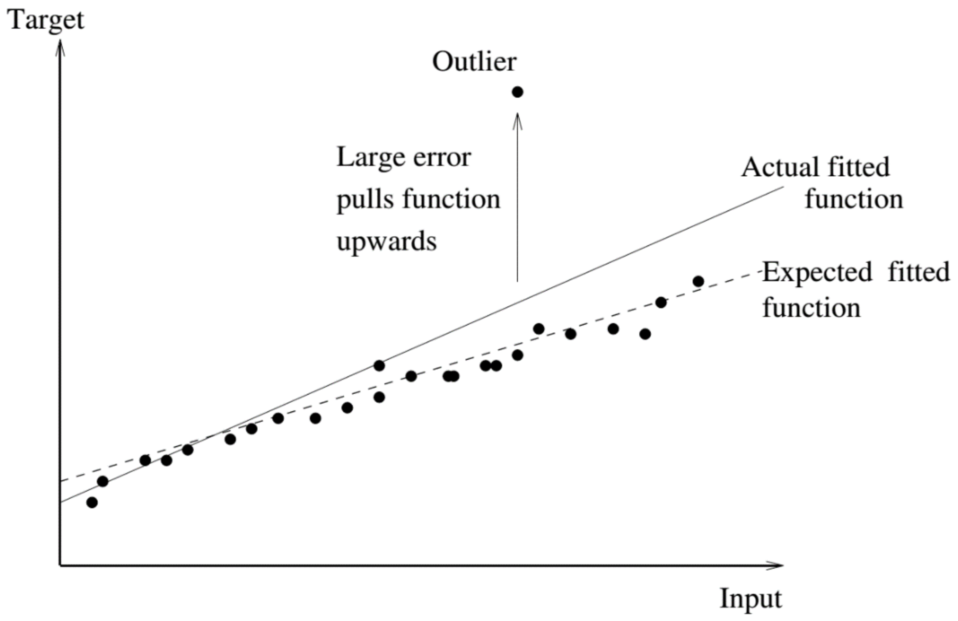


Figure V-1: Illustration of the way that outliers affect the fitted function (Engelbrecht, 2007)

Visual aids (Dot Plots and Box Plots) may be used to identify outliers. Box plots are useful graphical displays for describing the behaviour of the data (Figure V-2), and may help in the qualitative identification of outliers. However, both those methods require human interaction to identify outliers. This is a disadvantage for this work because the DOM it is intended to be used as an automated procedure. In this work, three methods that identify outliers were used, which do not require any human interaction, a) the standard deviation criterion, b) bisquare robust function, and c) the median absolute deviation criterion.

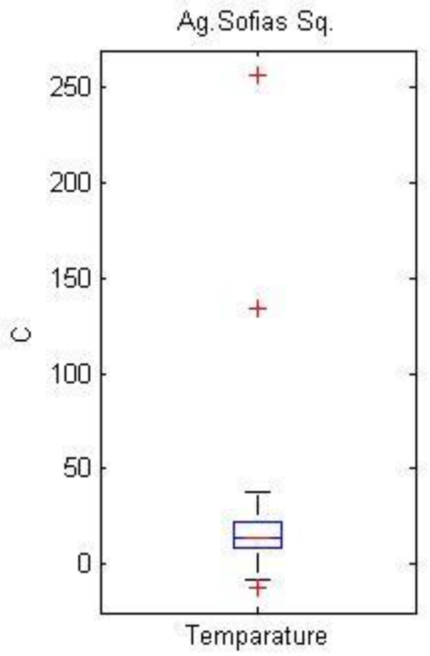


Figure V-2: Example of a Box plot for Temperature (at Ag. Sofia's monitoring station) (Kyriakidis, et al., 2009)

Standard Deviation Criterion

A common method that may be applied in order to quantitatively identifying outliers is to look for values located more than a certain number of standard deviations (STD) away from the mean (for each input variable) (Kyriakidis, et al., 2009). Although there is no general rule, a distance of two up to ten standard deviations above the mean may be applied in most AQ data series, as demonstrated in Figure V-3. The mean and the STD were calculated by the using the Matlab functions, "nanmean" and "nanstd" respectively. These functions were used because they are ignoring the missing values (marked as NaN - Not a Number).

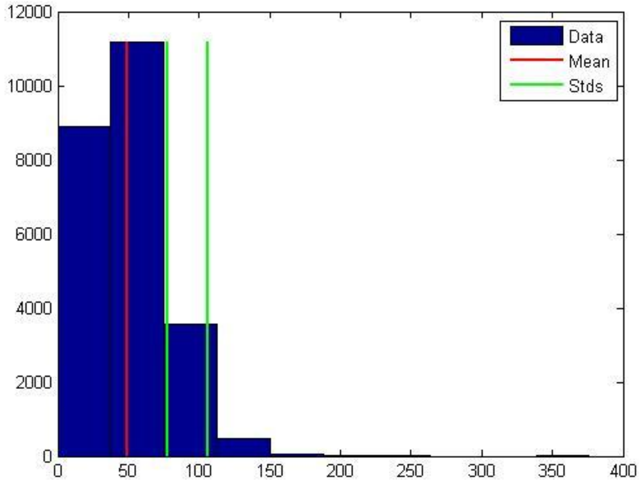


Figure V-3: Example of Temperature standard deviations (at Ag. Sofia's monitoring station) (Kyriakidis, et al., 2009)

Robust Functions

Robust objective functions may be used to identify outliers, as they are not influenced by them. Robust objective functions find the best fitting structure concerning the majority of the data. A robust objective function may also be able to identify the outlier's substructures for further treatment. The most common general method of robust objective function in regression is the "m-Estimator" introduced by Huber (1964) (Muthukrishnan & Radha, 2010).

The standard least-squares method tries to minimize the $\sum_i r_i^2$, which is unstable if there are outliers present in the data. Where r_i is the residual of the i^{th} datum, i.e. the difference between the i^{th} observation and its fitted value. The M-estimetors (Zhang, 1997) try to reduce the effect of outliers by replacing the squared residuals r_i^2 by another function of the residuals, as shown in formula (V-1)

$$min \sum_i \rho(r_i) \qquad \text{(V-1)}$$

Where:

ρ is a symmetric, positive-definite weight function with a unique minimum at zero, and is chosen to be less increasing than square.

In the current work, the "robustfit" function of Matlab was used in order to evaluate nine ρ weight functions (Table V-1). The "tune" parameter of function "robustfit" was not set in order to use the default tuning constant value of Table V-1. Default tuning constants give coefficient estimates that are approximately 95% as statistically efficient as the ordinary least-squares estimates, provided the response has a normal distribution with no outliers. The values that do not contribute to the best robust fit have a zero weight. Those are treated as outliers and removed from the time-series.

Table V-1: The nine robust regression weight functions

Weight Function	Equation	Default Tuning Constant
'andrews'	$w = (abs(r) < pi) .* sin(r) ./ r$	1.339
'bisquare'	$w = (abs(r) < 1) .* (1 - r.^2).^2$	4.685
'cauchy'	$w = 1 ./ (1 + r.^2)$	2.385
'fair'	$w = 1 ./ (1 + abs(r))$	1.400
'huber'	$w = 1 ./ max(1, abs(r))$	1.345
'logistic'	$w = tanh(r) ./ r$	1.205
'ols'	Ordinary least squares (no weighting function)	None
'talwar'	$w = 1 * (abs(r) < 1)$	2.795
'welsch'	$w = exp(-(r.^2))$	2.985

The value r in the weight functions of Table V-1 is

$$r = resid/(tune * s * sqrt(1 - h)) \qquad \text{(V-2)}$$

Where:

$resid$ is the vector of residuals from the previous iteration
h is the vector of leverage values from a least-squares fit
s is an estimate of the standard deviation of the error term given by

$$s = MAD/0.6745 \qquad \text{(V-3)}$$

Where:

MAD is the median absolute deviation of the residuals from their median.
The constant 0.6745 makes the estimate unbiased for the normal distribution.

Median Absolute Deviation

Leys, et al. (2013) states that the mean plus or minus three standard deviations rule is problematic and argues in favour of a robust alternative. In addition, they show how to use the Median Absolute Deviation (MAD) method to deal with the problem of outliers, which was adopted in this work.

There are three problems can be identified when using the mean as the central tendency indicator (Miller, 1991) (Leys, et al., 2013). Firstly, it assumes that the distribution is normal (outliers included). Secondly, the mean and standard deviation are strongly impacted by outliers. Thirdly, as stated by Cousineau & Chartier (2010), this method is very unlikely to detect outliers in small samples.

The median (M) is, like the mean, a measure of central tendency but offers the advantage of being very insensitive to the presence of outliers. To calculate the median, observations have to be sorted in ascending order to identify the mean rank of the statistical series and to determine the value associated with that rank. In this work, in order to calculate the MAD the "mad" function of MATLAB was used. The MAD is defined in formula (V-4) (Huber, 1981).

$$MAD = b\ M_i(|x_i - M_j(x_j)|) \tag{V-4}$$

Where:

b = 1.4826, a constant linked to the assumption of normality of the data, disregarding the abnormality induced by outliers (Rousseeuw & Croux, 1993).

In order to remove the outliers a rejection criterion threshold must be defined. The proposed values for the threshold rejection criterion threshold are 3 (very conservative), 2.5 (moderately conservative), and 2 (poorly conservative) (Leys, et al., 2013). In this work, a conservative threshold value of 3 was used, as shown in formula (V-5). This threshold value was selected because a dataset with daily values was used (i.e. averaged hourly values), and thus, high outlier values were not expected. All values greater than $M + (Threshold * MAD)$ and all values smaller than $M - (Threshold * MAD) < x_i$ can be removed.

$$M - (Threshold * MAD) < x_i < M + (Threshold * MAD) \tag{V-5}$$

Where:

$Threshold$ = 3 (in this work)

Methods of Step 2 (Missing Data)

Interpolation algorithms

The presence of missing values does not affect some algorithms (e.g. Naïve Bayes, CART, Bayes Tree, CN2) (Michie, et al., 2009), whereas others require that the missing values should be replaced or removed. While pattern removal solves the missing values problem, valuable information may be lost, and the available information for training is reduced, which may be a real issue if the data is already of limited volume. Missing or erroneous data complicates the modelling of large data sets, because of the fact that most data processing methods require for complete data metrics before performing any further analysis (Bishop, 1995).

In this work the next five interpolation algorithms were used, 1) Linear, 2) Piecewise, 3) Cubic, 4) Spline, and 5) Nearest. Interpolation algorithms produce estimates between known observations. Thus, the missing values at the start or at the end of the time-series could not be interpolated (i.e. estimated). In these cases, the observations were removed from the time-series.

For this problem, the aforementioned five interpolation algorithms were also used with extrapolation in order to estimate beyond the original observation range (i.e. for the missing at the end of the time-series). Table V-2 shows all the interpolation methods that were used by the "fillts" function of MATLAB.

Table V-2: The used MATLAB Interpolation and Extrapolation methods

Interpolation Method (fill method)	Description
'linear'	Linear interpolation. The interpolated value at a query point is based on linear interpolation of the values at neighbouring grid points in each respective dimension.
'linearExtrap'	
'cubic'	Shape-preserving piecewise cubic interpolation. The interpolated value at a query point is based on a shape-preserving piecewise cubic interpolation of the values at neighbouring grid points.
'cubicExtrap'	
'spline'	Spline interpolation using not-a-knot end conditions. The interpolated value at a query point is based on a cubic interpolation of the values at neighbouring grid points in each respective dimension.
'splineExtrap'	
'nearest'	Nearest neighbour interpolation. The interpolated value at a query point is the value at the nearest sample grid point.
'nearestExtrap'	
'pchip'	Piecewise Cubic Hermite Interpolating Polynomial. Shape-preserving piecewise cubic interpolation. The interpolated value at a query point is based on a shape-preserving piecewise cubic interpolation of the values at neighbouring grid points.
'pchipExtrap'	

Methods of Step 3 (Smoothing Data)

Data smoothing techniques are used to eliminate "noise" and extract real trends and patterns. They provide with a clearer view of the behaviour of the time series studied. Thus, for example, economists use smoothing techniques to reveal economic trends in data (Leblebicioğlu, 2009). Smoothing may also deal with missing values, in cases that they are not of high percentage. In some cases, seasonal variation is so strong that do not allow for any trend or periodicity indications, aspects of great importance for the understanding of the process being studied. Smoothing can remove seasonality and may help in revealing lagged fluctuations within the data set (Xiong, et al., 2006). In this work, six smoothing methods were used in each variable by the “smooth” function of MATLAB. Table V-3 presents the six supported smoothing methods of MATLAB.

Table V-3: The six supported smoothing methods of MATLAB

Smoothing Method	Description
'moving'	Moving average is a lowpass filter with filter coefficients equal to the reciprocal of the span (in this work the span number was 5).
'lowess'	Local regression using weighted linear least squares and a 1st degree polynomial model
'loess'	Local regression using weighted linear least squares and a 2nd degree polynomial model
'sgolay'	Savitzky-Golay filter is a generalized moving average with filter coefficients determined by an unweighted linear least-squares regression and a polynomial model of specified degree (2nd degree is used here).
'rlowess'	A robust version of 'lowess' that assigns lower weight to outliers in the regression. The method assigns zero weight to data outside six mean absolute deviations.
'rloess'	A robust version of 'loess' that assigns lower weight to outliers in the regression. The method assigns zero weight to data outside six mean absolute deviations.

To configure the moving average method’s input parameter (known as, span or length), Equation (V-6 was used. This equation calculates the “ideal moving average length” as described by Achelis (2013). In this work, the periodicity of the forecasting parameter (PM_{10}) was found to be weekly, as in other studies (Byung-Gon, et al., 2009) (Yong-Sang & Bo-Ram, 2012) (Bigi & Ghermandi, 2014). Thus, the moving average length was calculated as $(7/2) + 1 \cong 5$.

$$Ideal\ Moving\ Average\ Length = \frac{Cycle\ Length}{2} + 1 \tag{V-6}$$

Polynomials with low degrees are desirable for maximum smoothing of statistical noise in data (Barak, 1995). Thus, in the current work a 2nd polynomial degree Savitzky-Golay filter was used for data smoothing.

Methods of Step 4 (Detrending Data)

The trend in a time series is a slow, gradual change in some property of the data over the whole time interval under investigation. The trend is sometimes loosely defined as a long-term change in the mean, but can also refer to a change in other statistical properties.

Detrending is the mathematical operation of removing trends from the data under investigation. Detrending is often applied to remove a feature thought to distort or obscure the relationships of interest. Detrending is also used as a pre-processing step to prepare time series for analysis by methods that assume stationarity (Varotsos, et al., 2005). Detrending might improve computational accuracy when the signals vary around a large signal level.

Constant and Straight (linear) detrending

Constant detrending removes the mean of the data to create zero mean data. Straight line detrending finds linear trends (in the least-squared sense) and the removes them. In this work, the “detrend” function of MATLAB was used for each variable.

Fit and remove a 2nd degree of polynomial curve

Measured signals can show overall patterns that are not intrinsic to the data. These trends can sometimes hinder the data analysis and must be removed. To eliminate a nonlinear trend in a signal, the next two steps can be performed, 1) fit a low-order polynomial to the signal, and 2) subtract it from the signal. In this work, the "polyfit" function of MATLAB was used in order to find the coefficients of a 2nd degree of the polynomial that fits each input variable. In addition, the "polyval" function was used to calculate the value of the coefficients of a polynomial of 2nd degree.

Hodrick-Prescott filter

Hodrick & Prescott (1997) proposed a procedure for representing a time series as the sum of a smoothly varying trend component and a cyclical component. Hodrick-Prescott filter is used in this work to remove trends from the time-series. For that purpose, the “hpfilter” function of MATLAB was used for each input variable. This function requires a "smoothing" parameter, which determines how linear the smoothed series will become. As the smoothing parameter increases in value, the smoothed series becomes more linear. Appropriate values of the smoothing parameter depend upon the periodicity of the data. The MATLAB documentation suggests the smoothing values of Table V-4.

Table V-4: Suggested smoothing values per periodicity

Periodicity	Smoothing Value
Yearly	100
Quarterly	1600
Monthly	14400

In this implementation, the periodicity of each input variable was calculated by using the Fast Fourier Transformation (FFT). Formula (V-7) was used in order to map the $SmoothingValue$ from the $MonthlyPeriodicity$. The $MonthlyPeriodicity$ value was calculated from the highest value of strength.

$$SmoothingValue = \frac{14400}{MonthlyPeriodicity^2} \tag{V-7}$$

Methods of Step 5 (Feature Selection / Extraction)

It might be expected that the inclusion of an increasing number of features would increase the likelihood of including enough information to distinguish between classes. Unfortunately, this is not true if the size of the training dataset does also increase rapidly with each additional feature included, resulting in a sparse dataset (Jensen & Shen, 2008). This is the so-called curse of dimensionality (Bazan, et al., 1994).

The main aim of the Feature Selection/Extraction phase is to investigate the features to be selected for the effective modelling and forecasting with the aid of data mining-computational intelligence algorithms. A high dimensional dataset increases the chances that a data mining algorithm will find patterns that are not valid in general. For that purpose, the useful features that represent the data must be found, and the non-relevant features must be removed. This will reduce the time required to perform data mining, and will make the resulting rules more comprehensible and can increase the resulting classification accuracy (Jensen & Shen, 2008) (Webb, 2002) (Fulcher, 2006). Dimensionality reduction can be achieved by two different ways, feature selection, and feature extraction. All systems that deal with datasets with large dimensionality, feature selection, and extraction have found wide applicability. Some of the main areas of application are shown in Figure V-4.

- Feature selection methods select a subset of the original features based on a subset evaluation function. Basically, feature selection methods identify those variables that do not contribute to the classification task.
- Feature extraction (also called Feature transformation) methods transform the underline meaning of the data in order to have a descriptive power that is more easily ordered than the original features. This transformation may be a linear or nonlinear combination of the original variables and may be supervised or unsupervised (Webb, 2002).

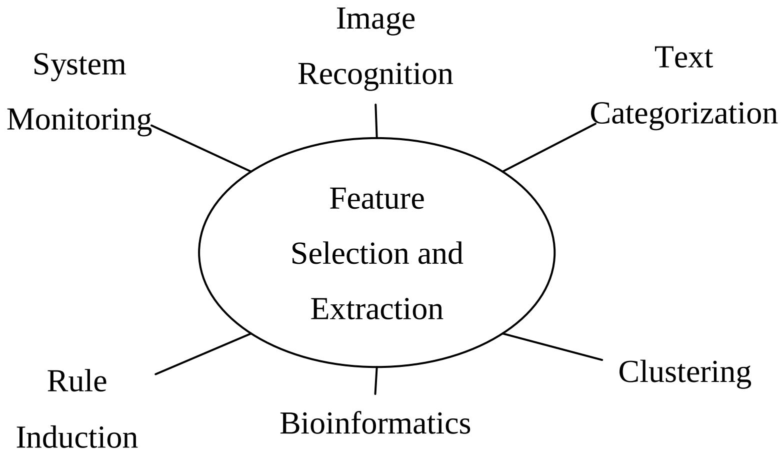


Figure V-4: Typical feature selection and extraction application areas

Factor Analysis

Factor analysis is a statistical method for investigating variable relationships for complex concepts. The purpose of factor analysis is to discover simple patterns in the pattern of relationships among the variables. In particular, it seeks to discover if the observed variables can be explained largely or entirely in terms of a much smaller number of variables called factors, and the coefficients are known as loadings (Darlington, 2015). Specifically, factor analysis assumes that the covariance matrix of the data is of the form described in formula (V-8).

$$cov(x) = \Lambda\Lambda^T + \Psi \tag{V-8}$$

Where:

Λ is the matrix of loadings

$\Psi = cov(e)$ is a d-by-d diagonal matrix of specific variances

In this work, the "factoran" function of MATLAB was used (without rotation) to compute the maximum likelihood estimate (MLE) of the factor loadings matrix Λ in the factor analysis model. The factoran(X, m) function returns among others,

a) The MLEs of the specific variances as a column vector of length d (referred as psi), and
b) A structure referred as stats.

When, X is an n-by-d matrix where each row is an observation of d variables. A psi value of 1 would indicate that there is no common factor component in that variable; while a psi value of 0 would indicate that the variable is entirely determined by common factors. The p-value of the stats structure is the right-tail significance level for the null hypothesis (H_0), that the number of common factors is m.

Because the number of common factors (m) was not known, the execution of the factoran function was repeated with different m values, ranges from 1 to 6. A higher number of common factors were not possible because the maximum number of the input variables was 11. Thus, a higher m value result is an error in the factoran function. When the p-value was greater from a threshold value of 0.8, the execution stops. From the outcome of the last execution, the variables with a psi value lower of equal to a threshold value of 0.4 were selected. These thresholds were chosen by a number of trial runs for the current data. Different thresholds may increase the forecasting performance, but its improvement was not a goal of this work.

Covariance Feature Selection

In probability theory and statistics, covariance is the measure of how much two random variables varies together, as distinct from variance, which measures how much a single variable varies. The covariance between two real-valued random variables X and Y is defined in formula (V-9) (Weisstein, 2015).

$$Cov_{XY} = \frac{\sum(X-\bar{X})(Y-\bar{Y})}{N} \qquad \text{(V-9)}$$

Where:
$\bar{X}$ is the mean of X
$\bar{Y}$ is the mean of Y
N is the number of observations

A simple feature selection method is to calculate the covariance of each input variable with the target variable and keep only the variables with the highest covariance. In this work, the variables with the highest covariance, which correspond in a maximum total covariance of at least 90%, were selected.

Principal Components Analysis

Principal component analysis (PCA) is a computational intelligence method originating from the multivariate statistical analysis that allows for the identification of the major drives within a certain multidimensional data set. Thus, PCA may be applied for data compression, identifying patterns in data, and expressing the data in such a way as to highlight their similarities and differences. Since patterns in data can be hard to find in high dimensional data, where the luxury of the graphical representation is not available, PCA is a powerful tool for such an analysis.

PCA generates a new set of variables, called principal components. Each principal component is a linear combination of the original variables. All the principal components are orthogonal to each other, so there is no redundant information. The principal components as a whole form an orthogonal basis for the space of the data.

In this work, the PCA was used in order to transform the input data and reduce their dimensionality. For that purpose, the "princomp(X)" function of MATLAB was used, which performs PCA on the n-by-p data matrix X, and returns the principal component coefficients, also known as loadings. The rows of X correspond to observations and columns to variables. The princomp function also returns the following information:

a) The principal component scores, which is the representation of X in the principal component space.
b) A vector containing the eigenvalues (variance) of the covariance matrix of X.

In order to keep only the principal components with high variance, a threshold value of 90% was used. Thus, the principal components which account at least 90% of the variation in the data set were selected. A variation threshold of 80% or 90% is used in the literature (Khattree & Naik, 2000) (Zwol, 2006) (Abrahams & Zhang, 2008) and sometimes recommended (Kingston & Clark, 2014) for components selection.

Methods of Step 6 (Forecasting)

Multiple Linear Regression

In a simple linear regression model, a single response measurement Y is related to a single predictor X for each observation, as is shown in formula (V-10). The assumption of the model is that the conditional mean function is linear.

$$Y = \alpha + \beta X \tag{V-10}$$

Multiple linear regression (MLR) attempts to model the relationship between two or more explanatory variables and a response variable by fitting a linear equation to observed data. Every value of the independent variable X is associated with a value of the dependent variable Y. A general equation of MLR is presented in formula (V-11).

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdots + \beta_p X_{ip} + \varepsilon_i \tag{V-11}$$

Where:

β_j are the coefficients that measure the effect of each predictor
ε_i is the error term, which is assumed it have a normal distribution with mean
0 and constant variance σ^2

In order to use the MLR model in this work, the "regress" function of MATLAB was used. The regress(y, X) returns a p-by-1 vector b of coefficient estimates for a multilinear regression of the responses in y on the predictors in X. The X parameter is an n-by-p matrix of p predictors at each of n observations, in which the train input data-set was used. The y parameter is an n-by-1 vector of observed responses, in which the train target data-set was used. To calculate the predicted values, the coefficients b was multiplied with the input test data-set.

Feed-forward Backpropagation Artificial Neural Networks

ANNs are computational data that may be considered as attempts to mimic the biological processes of the human brain and nervous system. ANNs were advanced in the late ’80s, popularizing non-linear regression techniques such as Multi-layer Perceptrons (MLP) (Gallant, 1990) and self-organizing maps (SOM) (Kohonen, 1997). ANNs introduce a new way to handle and analyse highly complex data (Yang & Yang, 2014). Thus, ANNs are widely used for data mining tasks such as classification and pattern recognition. ANNs are especially effective in modelling nonlinear relationships, which makes them ideal candidates for differential processes (Lu, et al., 2015).

Feed-forward neural networks (FNNs) are one of the popular structures among artificial neural networks. In FNNs, the information moves in only forward, from the input nodes, through the hidden nodes and finally to the output nodes, without any cycles or loops in the network.

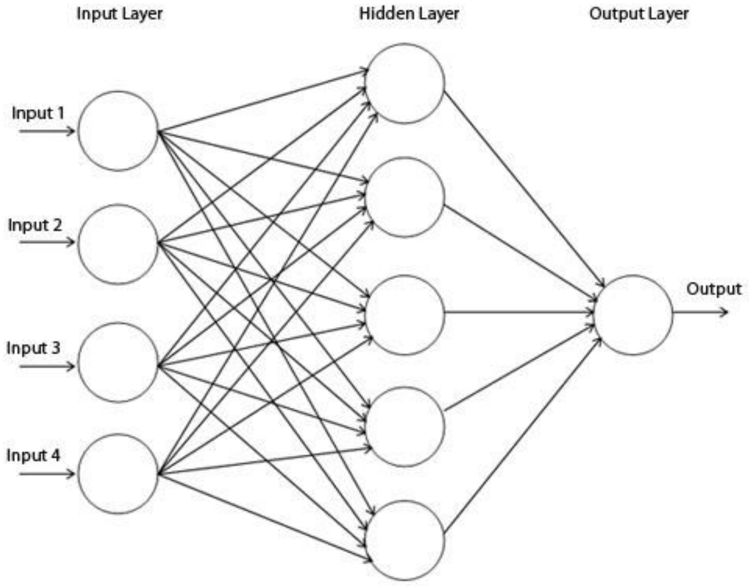


Figure V-5: Feed-forward neural network example

Backpropagation is a supervised learning technique, which means that the desired outputs are already known. The task of the backpropagation network is to learn to generate the desired outputs from the inputs. This is performed by calculating the gradient of a loss function with respects to all the weights in the network. The gradient is fed to the optimization method which is used to update the weights in order to minimize the loss function.

Data to be imported to the ANN were firstly normalized, by applying the hyperbolic tangent sigmoid transfer function (Figure V-6), which was also applied to the hidden layer. In the output layer, the linear transfer function was used (Figure V-7). This is a common structure for function approximation (or regression) problems (Mathworks, 2008), but has also shown good results in similar studies (Slini, et al., 2003).

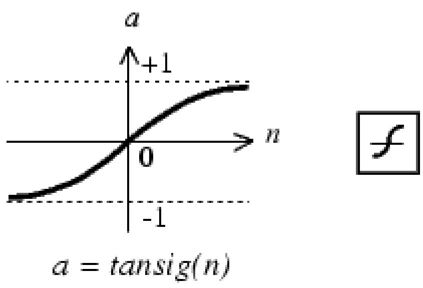


Figure V-6: The hyperbolic tangent sigmoid transfer function

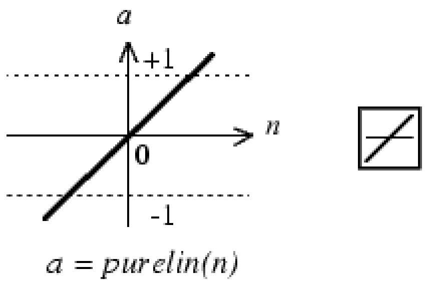


Figure V-7: The linear transfer function

In this work, the Levenberg-Marquardt Backpropagation algorithm (Saini & Soni, 2002) was implemented in the training phase. For that purpose, the "newff" function of MATLAB was used to initialize and configure the ANN. The newff was configured to use only one hidden layer, in which the number of neurons was equal to the number of the input parameters. In addition, it is configured to use two stopping criteria,

a) The maximum number of epochs (iterations), which was set to 100, and
b) The error goal, which was set to 0.01.

These stopping criteria were selected by a number of trial runs for the current data and goal of this work. In a different work, a higher maximum number of epochs or a lower error goal could be used. To train the ANN the "train" function of MATLAB was used, by using the train and validation data sets as arguments. To calculate the predicted values, the "sim" function of MATLAB was used by using the input test data set as an argument.

Linear Neural Networks

Linear Neural Networks (LNNs) are the simplest kind of networks, with a single-layer network whose weights and biases could be trained to produce a correct target vector when presented with the corresponding input vector. The linear neuron uses the "purelin" linear transfer function of MATLAB (Figure V-7). The LNNs is a supervised learning technique, because of the need for target outputs. The LNNs is used to solve linearly separable problems.

In order to use the LNNs in this work, the "newlind" function of MATLAB was used by using the training data sets (input data and targeted forecasting parameters) as arguments. To calculate the predicted values, the "sim" function of MATLAB was used by using the input test data set as an argument.

The "newlind" function calculates weight W and bias B values for a linear layer from inputs P and targets T by solving the linear equation (V-12) in the least squares sense.

$$T = [W \quad B] * \begin{bmatrix} P \\ ones \end{bmatrix} \tag{V-12}$$

Generalized Regression Neural Networks

The generalized regression neural networks (GRNNs) as proposed by Specht (1991) are memory-based networks that provide estimates of continuous variables and converge to the underlying (linear or nonlinear) regression surface. Specht (1991) shown that, even with sparse data in a multidimensional measurement space, the algorithm provides smooth transitions from one observed value to another. The GRNNs are a kind of radial basis network that is often used for function approximation. A GRNN does not require an iterative training procedure as back propagation networks. It approximates any arbitrary function between input and output vectors, drawing the function estimate directly from the training data (Hannan, 2010). The GRNN uses a two-layer network, which has a radial basis layer (first layer) and a special linear layer (second layer).

In the radial basis layer, each neuron's weighted input is the Euclidean distance between the input vector and its weight vector. Each neuron's net input is the product of its weighted input with its bias. The second layer has linear neurons, calculates weighted input, and net inputs are calculated by its weighted inputs (by not use biases).

In order to use the GRNNs in this work, the "newgrnn" function of MATLAB was used by using the training data sets (input data and targeted forecasting parameters) as arguments. The spread argument of the "newgrnn" function is the spread of radial basis functions, which was used by its default value of 1. In the literature, in order to find the best spread parameter for each work, a range of spread values are usually examined. In addition, the examined ranges of the spread values are different in each work. For example, Celikoglu (2006) used a range from 0.1 to 1 (with 0.1 intervals), Mateo, et al. (2015) used a range from 0.1 to 2 (with 0.1 intervals), and Ababaei, et al. (2012) used a range from 0.1 to 2000 (with variable intervals). In this work, additional spread values were not examined in order to increase the forecasting performance, because its improvement was not a goal. As spread becomes larger, the radial basis function's slope becomes smoother (more neurons contribute to the average) and several neurons can respond to an input vector. To calculate the predicted values, the "sim" function of MATLAB was used by using the input test data set as an argument.

Generalized linear model regression

Linear regression models describe a linear relationship between a response and one or more predictive terms. However, many times a nonlinear relationship exists, and in these cases a nonlinear regression can be used to describe general nonlinear models. A special class of nonlinear models called generalized linear models (GLMs) which use linear methods.

The GLMs were formulated by Nelder & Wedderburn (1972), as a technique of iterative weighted linear regression that can be used to obtain maximum likelihood estimates of the parameters with observations distributed according to some exponential family and systematic effects that can be made linear by a suitable transformation. As John Fox (2008) described, a GLM consists of three components:

1. A random component, specifying the conditional distribution of the response variable, Y_i (for the ith of n independently sampled observations), given the values of the explanatory variables in the model. In Nelder & Wedderburn (1972), the distribution of Y_i was a member of an exponential family, such as the Gaussian, binomial, Poisson, gamma, or inverse-Gaussian families of distributions.
2. A linear predictor that is a linear function of regressors: $\eta_i = \alpha + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdot \cdot \cdot + \beta_k X_{ik}$
3. A smooth and invertible linearizing link function $g(\cdot)$, which transforms the expectation of the response variable, $\mu_i = E(Y_i)$, to the linear predictor: $g(\mu_i) = \eta_i = \alpha + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdot \cdot \cdot + \beta_k X_{ik}$.

In order to use the GLM in this work, the "glmfit" function of MATLAB was used. The glmfit(X,y,distr) returns a (p + 1)-by-1 vector b of coefficient estimates for a generalized linear regression of the responses in y on the predictors in X. X is an n-by-p matrix of p predictors at each of n observations. The distr (i.e. distribution) argument can be any of the following strings: 'binomial', 'gamma', 'inverse gaussian', 'normal' (the default), and 'poisson'. In this work, the default 'normal' distribution was used. The other possible distributions were not examined in order to increase the forecasting performance because its improvement was not a goal. Table V-5 shows the commonly employed link functions and their corresponding distribution and link parameter in MATLAB. To calculate the predicted values, the coefficients b (by excluding the first value) was multiplied with the input test data set. This was necessary, because the glmfit function, by default adds a first column of 1s to X, corresponding to a constant term in the model.

Table V-5: The commonly employed link functions and their corresponding distribution and link parameter in MATLAB

Link (glmfit parameter)	**Distribution parameter (distr)**	**Description**
Identity ('identity')	'normal'	$\mu = Xb$
Log ('log')	'poisson'	$log(\mu) = Xb$
Logit ('logit')	'binomial'	$log(\mu/(1-\mu)) = Xb$
Probit ('probit')		$norminv(\mu) = Xb$
Complementary log-log ('comploglog')		$log(-log(1-\mu)) = Xb$
Inverse ('reciprocal')	'gamma'	$1/\mu = Xb$
Log-log ('loglog')		$log(-log(\mu)) = Xb$
Inverse-square (with number -2)	'inverse gaussian'	$\mu^{-2} = Xb$

Multivariate adaptive regression splines (MARS)

Multivariate adaptive regression splines (MARS) is introduced by Jerome H. Friedman (in 1991). The aim of the MARS procedure is to combine recursive partitioning and spline fitting in a way that best retains the positive aspects of both while being less vulnerable to their unfavourable properties. MARS have the ability to model complex and high-dimensional data dependencies. MARS builds models of the form of Equation (V-13). The model takes the form of an expansion in product spline basis functions, where the number of basis functions as well as the parameters associated with each one (product degree and knot locations) are automatically determined by the data through a forward/backward iterative approach (Jekabsons, 2015). Basis functions are known as hinge function, where their general form is shown in Equation (V-14).

$$f(x) = \sum_{i=1}^{k} C_i \times B_i(x) \tag{V-13}$$

Where:

$B_i(x)$ are the basis functions
C_i is constant coefficient
k is the number of total basis functions

$$\max(0, x - const)\ or\ \max(0, const - x) \tag{V-14}$$

In order to use the MARS in this work, the ARESLab toolbox (Jekabsons, 2015) for MATLAB was used. The "aresbuild" function of ARESLab toolbox was used to train the model, by using the training data sets (input data and targeted forecasting parameters) as arguments. To calculate the predicted values, the "arespredict" function of ARESLab toolbox was used by using the input test data set as an argument.