University of Padua

Department of Mathematics "Tullio Levi-Civita"

Master degree in Computer Science

Solar irradiance and atmospheric data: an analysis using data mining and machine learning techniques

Master's degree thesis

Supervisor

Professor Annamaria Guolo

Graduand

Massimo Toffoletto

Academic year 2021-2022

Abstract

This thesis describes an analysis of solar irradiance and atmospheric data using data mining and machine learning techniques.
The NSRDB dataset concerning solar irradiance and atmospheric variables in Los Angeles has been studied. Global Horizontal UV Irradiance of wavelength 280-400 nm has been chosen as the variable to analyze.
Data mining and machine learning methods have been implemented to analyze the data making interpretations and predictions. The data mining analysis has considered linear regression, enhanced with polynomials and natural splines, and regularization methods, such as ridge regression, lasso, Elastic Net, and Adaptive lasso. The machine learning analysis has considered the decision tree model, enhanced with the ensemble learning techniques Random forest and Extreme Gradient boosting, the K-Nearest-Neighbour, and the Support Vector Regressor. Moreover, some deepening in the analysis to improve the interpretation of the phenomenon has been performed. Linear regression assuming the Skew-Normal distribution on the response variable, Principal Component Analysis, and predictions with new data have been performed.
Finally, comparisons and discussions about the methods and the results have been described, drawing new scientific conclusions concerning Global Horizontal UV Irradiance of wavelength 280-400 nm.

"Too often we forget that genius, too, depends upon the data within its reach, that even Archimedes could not have devides Edison's inventions."

— Ernest Dimnet

Acknowledgements

First and foremost, I would like to express my special gratitude to professor Annamaria Guolo, who helped me by giving invaluable advice and support throughout the work.

I would like to express my special thanks to all my family for their support and for having always been present during the academic years.

Finally, I would like to thank all my friends for the amazing time we spent together. A special thank goes to Alberto, Lorenzo, and Fiammetta, who have always supported me during the most challenging moments.

Padua, 23 September 2022 Massimo Toffoletto

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This chapter introduces the scientific problem of this thesis with its real applications and some general information about the technologies and tools that have been used, the personal motivations to realize this thesis, and the document structure.

1.1 The scientific problem

The thesis focuses on solar irradiance and its associations with meteorological and atmospheric data. Solar irradiance is defined as the radiant energy received from the Sun in the form of electromagnetic radiation, and its unit of measure is watt per square meter (W/m^2), (see Solar irradiance). Solar irradiance is a general-purpose concept, and there are several measured types. The most relevant are listed below.

- Total Solar Irradiance (TSI): the measure of the solar power over all the wavelengths spectrum per unit area incoming perpendicularly on the atmosphere of the Earth.
- Direct Normal Irradiance (DNI): the measure of the solar radiations coming perpendicular to a surface, like the ground or something parallel to it. The name directed comes from the fact that this type of irradiance belongs to rays coming in a straight line from the sun's direction as its current position in the sky.
- Diffuse Horizontal Irradiance (DHI): the measure of the solar radiations scattered by clouds or other particles in the atmosphere and come equally to a surface from all directions, instead of the ones concerning DNI that come from a direct path.
- Global Horizontal Irradiance (GHI): the measure of the total amount of radiation received by a surface that is parallel, and so horizontal, to the ground.
- Global Tilted Irradiance (GTI): the measure of the total radiation captured on a tilt and azimuth surface.
- Global Normal Irradiance (GNI): the measure of the total irradiance on the surface of the Earth at a specific location and perpendicular to the Sun.

Total Solar Irradiance is a phenomenon that has been studied in great depth in recent decades, and tons of datasets are available on online platforms. GNI is the least

common coefficient since it has few concrete applications. DHI, DNI, and GHI refer to the whole wavelength spectrum (280-4000nm) of solar irradiance and have several applications in scientific fields. They are also linked by the mathematical formula $GHI = DNI * cos(solarZenithAngle) + DHI$, where the Solar Zenith Angle is the angle between the rays of the Sun and the vertical direction at a specific place on the Earth's surface. The coefficient GTI can be computed from DHI, DNI, and GHI (Gueymard, 2009).
There are specific instrumentations to capture all the types of irradiance and also mathematical models to compute them. However, there is a subset of the solar irradiance, called Global Horizontal UV Irradiance (Habte *et al.*, 2018), for which few measuring instrumentations are available. Global Horizontal UV Irradiance wavelength belongs to the range 280-400 nm. Further specifications belonging to more restricted ranges, such as 280-315 nm, 295-385 nm, and 315-400 nm, can be considered according to the phenomenon studied.
Global Horizontal UV Irradiance has many important applications in research fields, as will be described in the next section. Nevertheless, despite being available tons of data about solar irradiance, only a few regard Global Horizontal UV Irradiance. This is due to the lack of instrumentation for measuring that specific range of wavelengths. Mathematical models have been developed to predict Global Horizontal UV Irradiance starting from TSI and GHI coefficients (Habte *et al.*, 2018).
In literature, many factors are influential on the Global Horizontal UV Irradiance (see Seven factors affecting UV irradiance and EPA report). The most relevant are listed below.

- Cloud cover: the more the clouds covering the sky and their layers, the more reduced the UV irradiance reaching the surface of the Earth. However, sometimes reflections of the sun's rays on clouds can increase the Global UV Irradiance unexpectedly.
- Air quality: air pollutants and aerosols can scatter solar UV irradiance having little impact on the UV irradiance. Moreover, a high concentration of ozone can strongly absorb UV irradiance. This is the reason why the atmosphere protects humans from unhealthy levels of UV irradiance.
- Temperature: a high temperature generally means sunny and hot days, which correspond to a high level of UV irradiance.
- Altitude: the higher the altitude, the higher the UV irradiance since the atmosphere is reduced and cannot absorb the incoming UV.
- Ground surface reflexivity: according to the surface type, UV irradiance can be reflected and refracted in different ways causing increments or decrements of it.
- Direct and diffuse UV: the UV irradiance can be either directed or scattered, so according to the place where measurements are taken, scattered rays have to be considered even if they are not visible.
- Time of the day: according to the time of the day, the Sun is closer or farther to the Earth, and the UV irradiance is tilted or perpendicular to the Earth's surface. Hence, the strength of Global Horizontal UV Irradiance varies a lot.

However, in literature, no mathematical models to analyze the associations between those factors and the Global Horizontal UV Irradiance have been implemented yet.

Therefore, this thesis focuses on mathematical models trying to extract information and make predictions from potential associations between meteorological and atmospheric factors, for which tons of data are available, and Global Horizontal UV Irradiance. The analysis will be accomplished with Data Mining methods to find patterns, interpret the data, and make predictions and with Machine Learning methods to implement more complex mathematical models for making predictions according to human non-understandable patterns.
New models can be helpful for researchers to study more in-depth the phenomenon of Global Horizontal UV Irradiance according to the mathematical associations discovered and make accurate predictions when data is missing or alongside existing mathematical models.

1.2 Applications

The thesis investigates the associations between the measures of solar irradiance and the meteorological and atmospheric variables. The problem of solar irradiance has been studied for decades in many research fields, such as photovoltaic panel design or skin cancer. These works often makes use of the National Solar Radiation Database (NSRDB), a large and detailed dataset for solar irradiance and meteorological and atmospheric data created by the National Renewable Energy Laboratory (NREL) in the USA. The same dataset will be used for investigations in this thesis.
The thesis is inspired by some papers in the literature of solar irradiance and related topics.
The first application refers to a paper written in the IEEE Journal of Photovoltaics (Habte *et al.*, 2018). The paper studies the Global Horizontal UV Irradiance with respect to the all solar irradiance spectrum and how it can be estimated from the Global Horizontal Irradiance (GHI) coefficient. In particular, it is focused on the Global Horizontal UV Irradiance of 280-400 nm and 295-385 nm wavelengths, which belong to the medium and long wavelengths in the UV wavelength range.
Figure 1.1 reports a graph of the Global Horizontal UV Irradiance 280-400 nm trend of instrumental versus experimental values under a clear sky condition. The picture highlights the high precision of the model since the difference between the instrumental and the model values is quite low, dealing with the experimental uncertainty.
The second application comes from a report by the company Kipp & Zonen about the monitoring of solar energy and the deterioration of photovoltaic panels. Kipp & Zonen is a leading manufacturer company of sensors for capturing solar radiation. They collaborate closely with NREL providing sensors to catch part of the data stored in the NSRDB and, in turn, making use of them to refine their sensors and research in the field of photovoltaic panels. They found that the direction, which means the Solar Zenith Angle, and the intensity of solar irradiance, which means the values of GHI and related coefficients, affect the panels' performance and durability.
The third application is the damage to human beings' health resulting from incorrect and prolonged exposure to Global Horizontal UV irradiance (see UV effects on humans). More in detail, there are three categories of wavelength UV radiations: short, medium, and long. The short-wavelength UV radiations are filtered by the atmosphere without affecting humans, while the others are not, so their estimation is essential. The medium-wavelength UV radiations are the 5% of the whole that reaches the Earth. They cannot penetrate the superficial skin layers and are responsible for tanning and enhancing skin aging. However, the relatively long-wavelength UV radiations, which

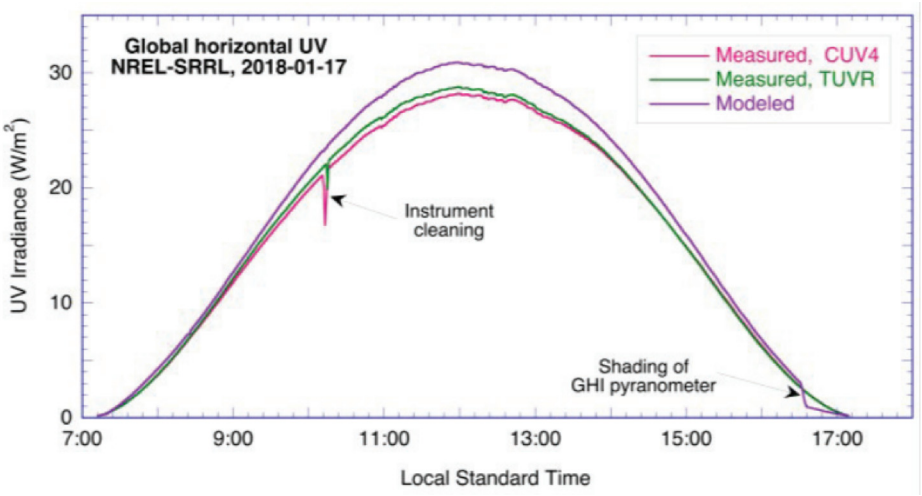

Figure 1.1: Instrumental vs experimental values of the Global Horizontal UV Irradiance 280-400 nm under a clear sky condition.
Source: `https://www.osti.gov/pages/servlets/purl/1484592`.

are the 95% of the whole that reaches the Earth, are the most damaging for humans since they penetrate the deeper layers of the skin and are responsible for skin cancer.

1.3 Technologies and tools used

1.3.1 Technologies

The technologies used in this thesis are R (R Core Team, 2022) and Python (Van Rossum and Drake, 2009). They have allowed the implementation of the preprocessing and the data analysis.

R (R Core Team, 2022) is a programming language for statistical computing and graphical analysis, supported by the R Core Team and the R Foundation for Statistical Computing. It is one of the most used in data mining, bioinformatics, and statistics. The language is multi-paradigm: procedural, object-oriented, functional, and imperative, so it is suitable for many needs. It provides a wide variety of libraries for statistical learning. The ones used in this thesis are described in Appendix A.

Python (Van Rossum and Drake, 2009) is a high-level and general-purpose programming language developed by Python Software Foundation. Its main features are dynamic typing, the use of a garbage collector, multi-paradigm programming, such as structured, object-oriented, and functional, and high-speed execution. It is widely used in scientific disciplines and provides many well-done libraries for machine learning. The ones used in this thesis are described in Appendix B.

1.3.2 Tools

The tools used for this thesis are two environments for implemented R and Python code. They are named RStudio and Google Colaboratory (see Google Colaboratory), respectively.

RStudio is a multi-platform environment for implementing software in R and Python. It was thought for data scientists and statistics. In this thesis, RStudio has been used for data analysis with data mining methods in the R language.

Google Colab (see Google Colaboratory) is an online and shared environment that allows writing notebooks with software in Python and texts in Markup. It is widely spread among data scientists, AI researchers, and computer science students since it provides GPUs and TPUs with limited use without paying money, and it is integrable into the Google suite. In this thesis, it has been used for implementing machine learning methods.

1.4 Personal motivations

During my Master's degree, I attended many courses in artificial intelligence and data analysis. In particular, data analysis has fascinated me. I have looked for some research area to realize something innovative that could have a purpose in the scientific world. I have found solar irradiance to satisfy these requirements. It is a field of research in terms of renewable energy and human health linked to climate change problems. Therefore, I have found the analysis of solar irradiance related to meteorological and atmospheric conditions to be an engaging challenge for the Master's degree thesis.

1.5 Document structure

The current chapter is an introduction of the research work, while the rest of the document is organized according to the following structure.

The first chapter describes the scientific problem, the methods used, and the technologies and tools adopted.

The second chapter describes the dataset, how it has been collected, its structure, and the preprocessing steps to understand the most suitable kind of analysis for it and to get it ready.

The third chapter describes the Data Mining methods that will be used for the data analysis from a theoretical point of view.

The fourth chapter describes the data analysis performed by the Data Mining methods introduced in Chapter 3.

The fifth chapter describes the Machine Learning methods that will be used for the data analysis from a theoretical point of view.

The sixth chapter describes the data analysis performed by the Machine Learning methods introduced in Chapter6.

The seventh chapter describes a deepening in data analysis using Data Mining techniques introduced in the third chapter to extract additional features from the data.

The eighth chapter describes the comparisons and the discussions of the results obtained by the analysis with Data Mining and Machine Learning methods.

The ninth chapter It describes a brief summary of the analyses, new scientific contributions, the limits of the analyses, and interesting extensions to this work.

Chapter 2

The NSRDB

This chapter focuses on the description of the data analyzed in the thesis, with attention to their structure, the content, the infrastructure of sensors built to gain the data, and some real applications using them.

2.1 Introduction

Some decades ago, the need to investigate solar energy was raised, mainly because of the climate change problem. Therefore, a scientific laboratory in the USA, called National Renewable Energy Laboratory (NREL), Figure 2.1, in collaboration with the U.S. Department of Energy and some other companies, including NASA, decided to build an infrastructure of stations, each one equipped with a set of sensors to carry out all the measurements, and to collect the data to increase knowledge about the phenomenon of solar radiations and how it is affected. These data were grouped and became the National Solar Radiation Database (NSRDB) (see NSRDB). Then, the collaboration of companies started processing and analyzing the data pointing out useful information about the phenomenon and it is still going on.

Figure 2.1: Logo of the NREL.
Source: `https://nsrdb.nrel.gov/`.

Nowadays, the NSRDB is a collection of meteorological and atmospheric data and relevant measurements of solar irradiance: DNI, DHI, and GHI that cover all the irradiance spectrum (280-4000nm) and Global Horizontal UV Irradiance of wavelengths in the range of 280-400nm and 295-385nm. Despite lacking of air quality data, NSRDB is one of the most complete dataset in this scientific field.
One of the main roles of solar irradiance is to be the input of solar energy generation systems and it provides knowledge to make decisions about how to manage this renewable energy efficiently and effectively. In particular, in the use-case of solar and photovoltaic panels, the Global Horizontal UV Irradiance is meaningful since its values,

together with the Solar Zenith Angle, have a huge impact on the performance and the reliability.
Furthermore, the Global Horizontal UV Irradiance directly affects human health since a too prolonged exposure to UV rays can cause skin irritation, sunburn, erythema to even skin cancers, thus it is very important to study this phenomenon.
Therefore, it is a case of interest to study and analyze the values of Global Horizontal UV Irradiance during a fixed period and how they are affected by the meteorological-atmospheric variables in the dataset.
The following Figure 2.2 illustrates a map of the international data of the solar irradiance stored in the NSRDB.

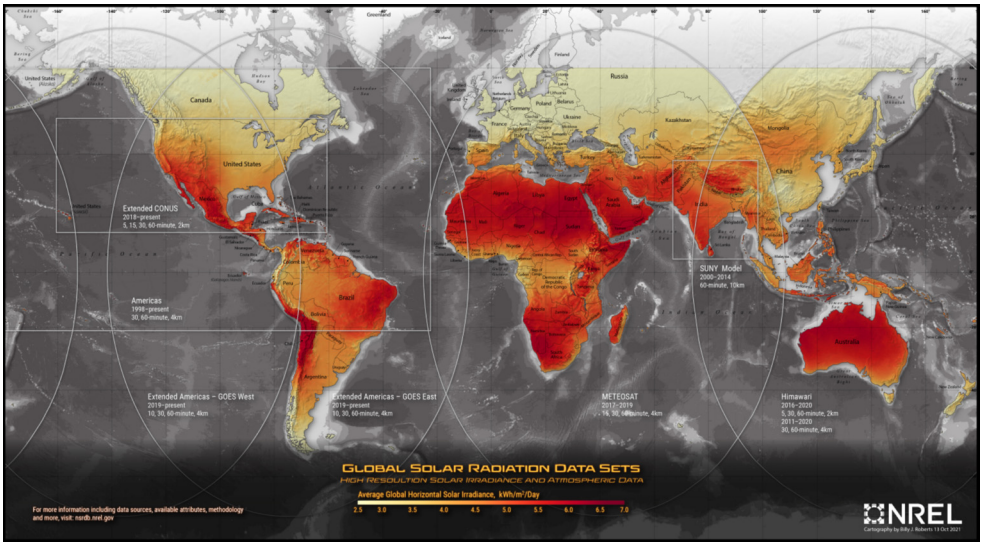

Figure 2.2: The NSRDB International data of solar radiations.
Source: `https://nsrdb.nrel.gov/data-sets/international-data`.

During the years, NREL and its partners have improved the infrastructure of sensors for collecting data both from a quantitative point of view, since the sensor network has expanded from the United States to all continents of the world, and from a qualitative point of view, since more precise sensors have been developed and installed. Nowadays, the state-of-the-art infrastructures are many, depending on the nation in which they are installed, but, in general, the measurements are very accurate and reliable.
In this thesis, I focused on the data collected in Los Angeles. Such a choice is motivated by several reasons. The infrastructure for solar and meteorological data collection used by NREL in the USA is the Physical Solar Model version 3 (PSM V3), one of the most advanced in this field. In addition, the weather conditions are very compliant with the intensive and effective use of solar energy, so the solar irradiance analysis becomes interesting also for applications.
Finally, since there is a huge history of all the data collected over the years, the focus will be only on 2020, the most recent year for which data are available; therefore, the purpose is to understand the behavior of solar irradiance concerning atmospheric conditions during 2020. As a further future side work, it could be done a comparison of the solar irradiance over the years.

2.2 Sensors infrastructure

From the beginning of the research about solar irradiance for creating the NSRDB, the infrastructure of sensors has evolved significantly both from an architectural and a qualitative point of view (see NSRDB versions history and NSRDB processing report). In the first years, especially in the nineties, the solar irradiance measurements were not very precise. In recent years, the refined accuracy of the sensors has allowed obtaining low measurement errors guaranteeing more reliable results.
The major update of the sensors infrastructure was planned in 2003 and released in 2007. The issues to be resolved were about the following fields: the aerosols, water vapor, and ozone measurements, the ground-measured evaluation data set, the solar model selection, the satellite-based modeled irradiance data, and the cloud cover observations. The last update was the most important since, before it, the measurements were inconsistent and discontinuous from station to station because they were made manually by humans, and lots of precision issues were detected according to the altitude of clouds. Therefore, NREL and its partners promoted new sensor networks to improve data quality. In particular, automated cloud observations address the precision issues related to the clouds giving high-quality data when the sky is covered without being affected by the altitude.
The two current main infrastructure models for solar irradiance sensor networks are the State University of New York at Albany (SUNY) used in South Asia and the PSM V3 used in all the other places of the world, especially in Los Angeles.
PSM V3 is a system made by a set of satellites that captures a set of data in collaboration with specific sensors and a set of computation nodes, as illustrated in the scheme of Figure 2.3.

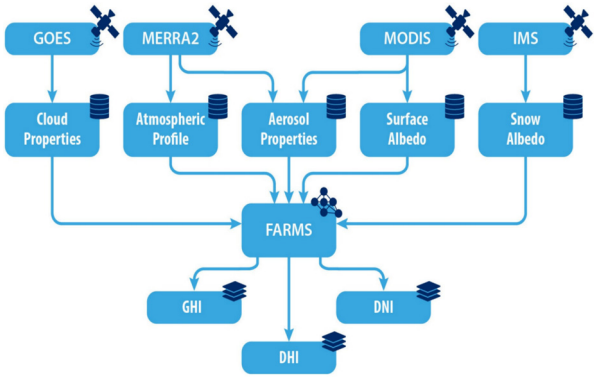

Figure 2.3: Workflow of PSM V3.
Source: `https://www.nist.gov/system/files/documents/2020/01/15/Habte.pdf`.

In particular, the GOES satellite captures data about cloud properties, the MERRA2 satellite captures data about the atmospheric profile and the aerosol properties, the MODIS satellite captures data about surface albedo and aerosol properties, and the IMS satellite captures data about snow albedo. Then, the data are stored in ad hoc databases and are sent to the computation node that calculate the coefficients DNI, DHI, and GHI. In some cases, NREL installed proper sensors to measure DNI, DHI, and GHI instead of computing them. Moreover, the clear sky versions of these

coefficients are calculated by the computational nodes through a proper mathematical model that simulates the absence of clouds.
There are few experimental sensors able to capture the Global Horizontal UV Irradiance value in each wavelength range. Therefore, NREL scientists developed a mathematical model that produces the Global Horizontal UV Irradiance values for each wavelength range according to all the data captured by the sensors in every sky condition. Figure 2.4 illustrates the Global Horizontal UV Irradiance values measured versus its model prediction. It can be seen that the predictions are so accurate to be used for every scientific application.

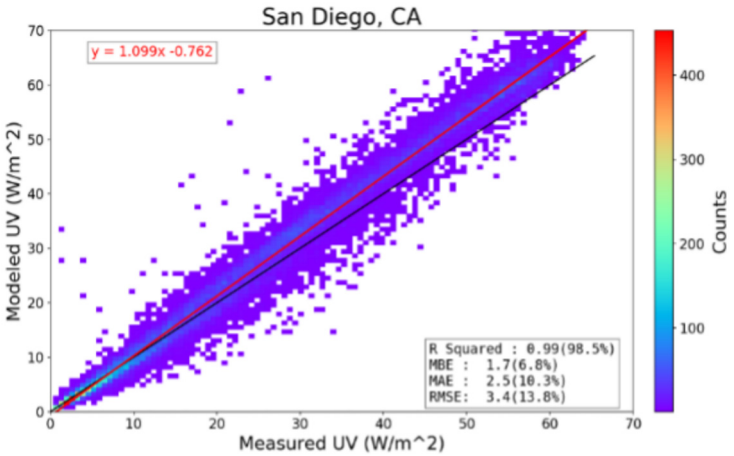

Figure 2.4: Example of modeled versus measured Global Horizontal UV Irradiance (280-400 nm).
Source: `https://www.nrel.gov/docs/fy22osti/82063.pdf`.

PSM V3 runs this model to predict the Global Horizontal UV Irradiance value and fill the dataset.

2.3 Structure of the data

The NSRDB contains all the historical data collected since 1991 and is continuously updated. For the data from 1991 to 2005, a repository has been dedicated (see NSRDB old repository). It is subdivided by station and ordered by class to which they belong. There are three classes: the first represents the most accurate stations that provide completeness of data, while the other two correspond to stations that are less and less accurate, with the incompleteness of data both in variables and in observations. For recent data, there is a graphical interface that allows selecting the area or the station for the data the user is interested in and downloading a portion of them according to filters (see NSRDB new repository). The main drawback is that recent data are not accompanied by exhaustive documentation.
The thesis focuses on a portion of the data collected in Los Angeles by the station located at 128 meters above sea level, corresponding to a file, named 83558_34.05_-118.18_2020, where

1. Location ID: 83558, which corresponds to the ID of the station;

2. Latitude: 34.05, which is the approximate latitude of the station;
3. Longitude: -118.18, which is the approximated longitude of the station;
4. Year: 2020, which is the year the data refer to.

The main content of the file can be represented like a big matrix where each column represents a measured variable and each row represents an observation made at a certain moment.
There are 17568 observations and 24 variables, listed below.

- Year: An integer representing the year in which the observation was collected.
- Month: An integer representing the month in which the observation was collected.
- Day: An integer representing the day of the month in which the observation was collected.
- Minute: The integer 0 or 30 representing the minute in which the observation was collected since the measurements were done every 30 minutes.
- DHI: The DHI numerical coefficient representing the solar radiation that has been scattered by clouds and particles in the atmosphere and comes equally from all directions. Its unit of measurement is W/m^2.
- DNI: The DNI numerical coefficient representing the amount of light that is coming perpendicular to the surface. Its unit of measurement is W/m^2.
- GHI: The GHI numerical coefficient representing the total amount of short-wavelength radiation received from above by a surface that is parallel to the ground. Its unit of measurement is W/m^2.
- Clearsky DHI: An estimated numerical coefficient that represents the DHI without considering any trace of clouds in the sky. Its unit of measurement is W/m^2. It is very useful to understand how the clouds affect the DHI value.
- Clearsky DNI: An estimated numerical coefficient that represents the DNI without considering any trace of clouds in the sky. Its unit of measurement is W/m^2. It is very useful to understand how the clouds affect the DNI value.
- Clearsky GHI: An estimated numerical coefficient that represents the GHI without considering any trace of clouds in the sky. Its unit of measurement is W/m^2. It is very useful to understand how the clouds affect the GHI value.
- Cloud type: An integer value that belongs to the set $\{1, ..., 12\}$ and it represents different kinds of clouds as follows:
  - Cloud type 0: the sky is totally clear;
  - Cloud type 1: the sky is probably clear;
  - Cloud type 2: there is fog in the sky;
  - Cloud type 3: the clouds are full of water;
  - Cloud type 4: the clouds are full of super-cooled water;
  - Cloud type 5: the clouds are of mixed type, no one precisely defined;

  - Cloud type 6: the clouds are opaque ice;
  - Cloud type 7: the clouds like cirrus, high altitude white clouds, thin and transparent, with a fibrous structure;
  - Cloud type 8: the clouds are overlapping;
  - Could type 9: the clouds are overshooting, that is they are passing further in the sky;
  - Could type 10: the clouds are unknown;
  - Could type 11: the clouds are dust;
  - Could type 12: the clouds are smoke;
  - Could type 15: N/A.

- Dew point: Unit of measurement is Celsius degree.
- Solar Zenith Angle: The angle between the sun's rays and the vertical direction. Its unit of measurement is the degrees.
- Fill flag: A positive integer number that identifies some defects in the measurement. The most important are the following:
  - Fill flag 0: N/A;
  - Fill flag 1: missing image;
  - Fill Flag 2: low irradiance;
  - Fill flag 3: exceeds clearsky;
  - Fill flag 4: missing cloud properties;
  - Fill flag 5: Rayleigh violation: it is a phenomenon that happens when the physic law of Rayleigh, which defines a critical statement on the physics of the wave, is violated.
- Surface albedo: A numerical coefficient representing the measure of the reflectivity of the terrestrial surface to solar radiation related to where the station is located.
- Wind speed: Unit of measure is m/s.
- Precipitable water: The amount of potential water contained in the clouds. Its unit of measure is cm.
- Wind direction: Unit of measure is degrees.
- Relative humidity: The percentage of humidity in the atmosphere.
- Temperature: Unit of measure is Celsius degrees.
- Pressure: Unit of measure is $mbar$.
- Global Horizontal UV Irradiance (280-400nm): A numerical coefficient representing the global ultraviolet irradiance parallel to the ground of wavelengths between 280 and 400 nm, a small subset of the whole spectrum of solar radiations. Its unit of measurement is W/m^2.

- Global Horizontal UV Irradiance (295-385nm): A numerical coefficient representing the global ultraviolet irradiance, parallel to the ground, of wavelengths between 295 and 385 nm, a small subset of the whole spectrum of solar radiations. Its unit of measurement is W/m^2.

Unlike what is specified in the infrastructure PSM V3 documentation, no snow albedo values are provided in the dataset since the refraction of snow in Los Angeles is meaningless.

2.4 Preprocessing of the data

The first preprocessing step consists of detecting and fixing abnormal behaviours in the dataset for the preliminary analysis.
The dataset has no missing values since the sensors network is very reliable. Data have been re-arranged to obtain a tabular form, where the first row contains the label of the measured variables in each column and the others contain the observations.
Some variables have been dropped. They include Year as it is a constant equal to 2020, Day as it gives no additional information to Month, Minute that only increases the precision of Hour, GHI, and Clearsky GHI as they could be determined from the remaining data, and Fill flag, whose values do not conform to the documentation.
Therefore, there are 8784 observations on 18 variables.
The variable Solar Zenith Angle has been transformed using the cosine function, as is commonly done in the solar irradiance literature.
The next step was the variables' type detection: numeric or factor. There were no factors, but Cloud Type is supposed to be according to its values and meaning. In addition, Month, Hour, and Surface albedo could be represented as factors, but it will depend on the specific method applied. However, I will treat them as factors to highlight better their potential associations with the response variable.
The chosen response variable is Global Horizontal UV Irradiance of 280-400 nm wavelengths since it has a wider range with respect to that of 295-385 nm wavelengths.
At a first glimpse, we notice that the response includes many zeros. Since the analysis focuses on the relationship between Global Horizontal UV Irradiance of 280-400 nm wavelengths and meteorological and atmospheric conditions, cases of no detection have been suppressed. These correspond to when the sun is not high enough in the sky to provide perceptible solar irradiance.
The dataset ready for the analysis includes 3814 observations on 18 variables.
A preliminary graphical analysis suggests that the distribution of Global Horizontal UV Irradiance of 280-400 nm wavelengths does not fit a Normal, as it can be seen in Figure 2.5.
A logarithmic transformation of the response makes the distribution closer to a Normal and allows the support of the variable to be on the real line, coherently with the support of a normal variable, as illustrated in Figure 2.6.
The choice is relevant to the assumption of the linear regression model and its modification we will use for the analysis. Extensions to more flexible regression models, such as those describing the response using the Skew-Normal distribution, will be discussed.
A graphical inspection of the potential relationship between the covariates and the response has suggested some interesting functions. Nevertheless, some patterns can be hidden by the noise associated with the data.
Figure 2.7 illustrates the associations of DHI and DNI with the response variable.
The response variable increases as DHI increases. Similar behavior is shown for DNI,

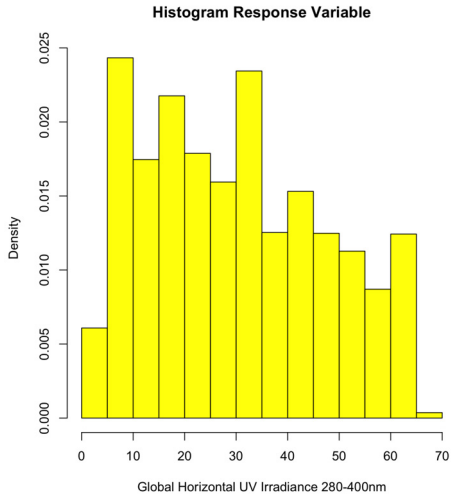

Figure 2.5: Histogram of the Global Horizontal UV Irradiance 280-400 nm distribution without zeroes.

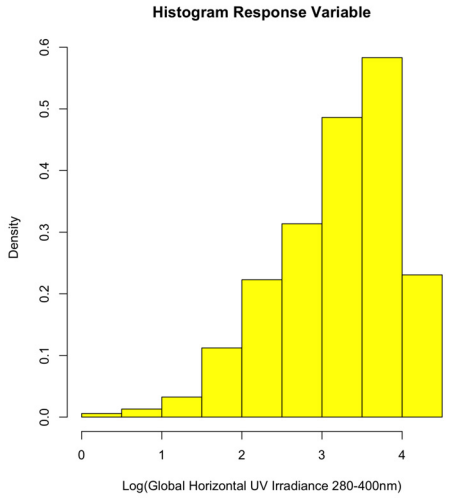

Figure 2.6: Histogram of the natural logarithm of the Global Horizontal UV Irradiance 280-400 nm distribution without zeroes.

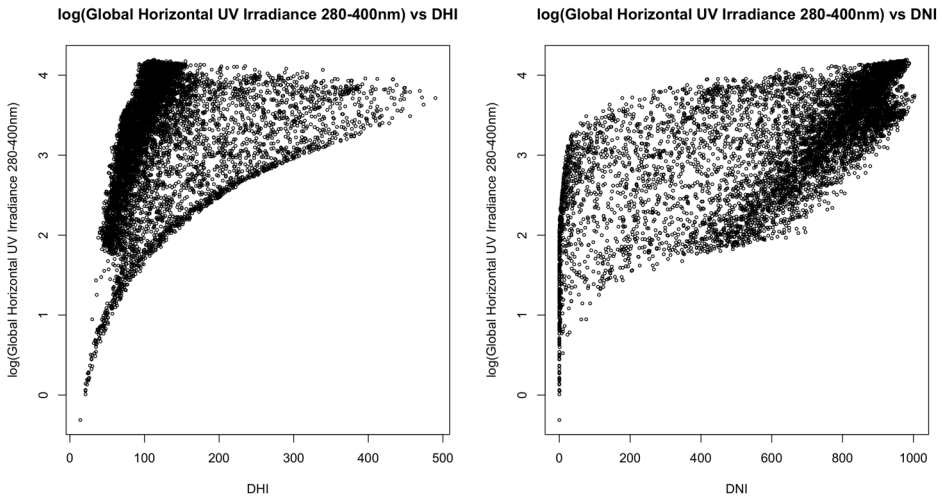

Figure 2.7: Scatter plots of the response variable versus covariates DHI and DNI.

although being less extent. Large variability in the dispersion plot supports that the relationship can be more complex than just linear.
Figure 2.8 illustrates the associations of Clearsky DHI and Clearsky DNI with the response variable. In the left graph, the response variable increases quickly to 150 as Clearsky DHI increases, but then it is stable. Most of them are concentrated in the left region of the graph, while elsewhere, they are sparse. In the right graph, the response variable increases less quickly as Clearsky DNI increases. Most of them are concentrated in the right region of the graph, while elsewhere, they are sparse.
The increasing trend recognized in the previous graphs about the coefficients DNI and DHI is coherent with what we can expect from the literature since the Global Horizontal UV Irradiance of every wavelength is a subset of the whole solar irradiance measured by those coefficients.
Relative humidity and temperature are the only meteorological and atmospheric numeric variables that seem to have a visible association, albeit with a high variance, as Figure 2.9 illustrates. For the others, the points cloud is too messy, and it cannot be gathered useful information.
The response variable slowly decreases as Relative humidity increases, while it increases as Temperature increases. The response variable is expected to be inversely proportional to the humidity and directly proportional to the temperature: in general, when the humidity grows, it is probably raining, so the solar radiations should be weaker, and when the temperature increases, the sun is probably high in the sky, so solar irradiance should be more powerful. However, there are lots of cases. In fact, the variance is even higher than the covariates previously analyzed, so any assumptions are uncertain.
Figure 2.10 illustrates the associations of the factor Month with the response variable. As expected, the response variable values are higher in the warmer months and lower in the colder months because of solar irradiance power. In Summer and Winter, the variance is lower than in Spring and Autumn, since the weather conditions are more stable. For the same reason, the median is almost equidistant between the first and

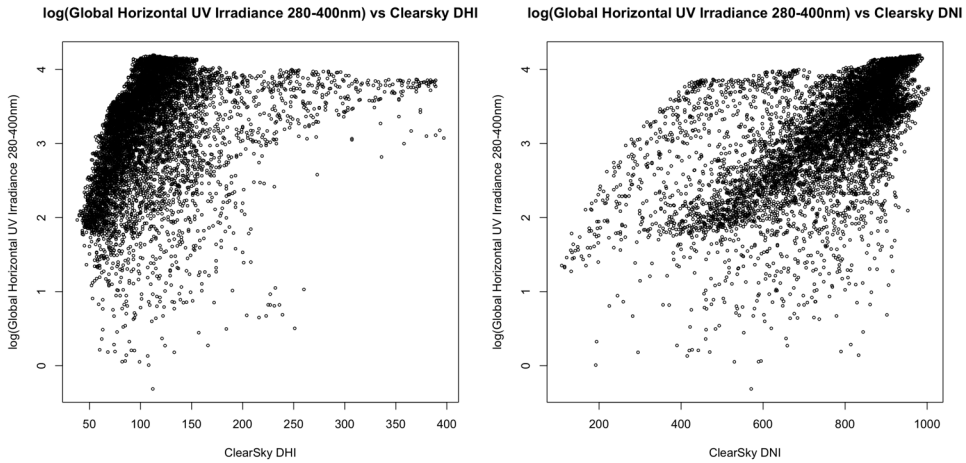

Figure 2.8: Scatter plots of the response variable versus the covariates Clearsky DHI and Clearsky DNI.

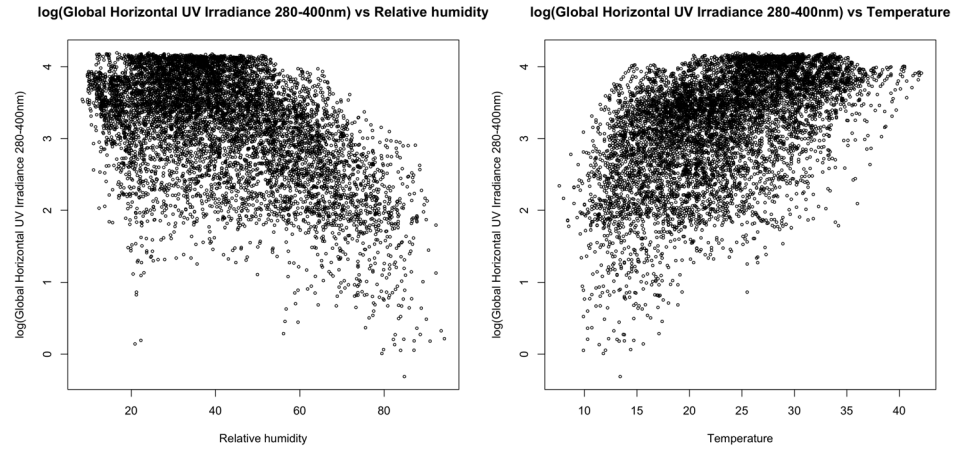

Figure 2.9: Scatter plots of the response variable versus the covariates Relative humidity and Temperature.

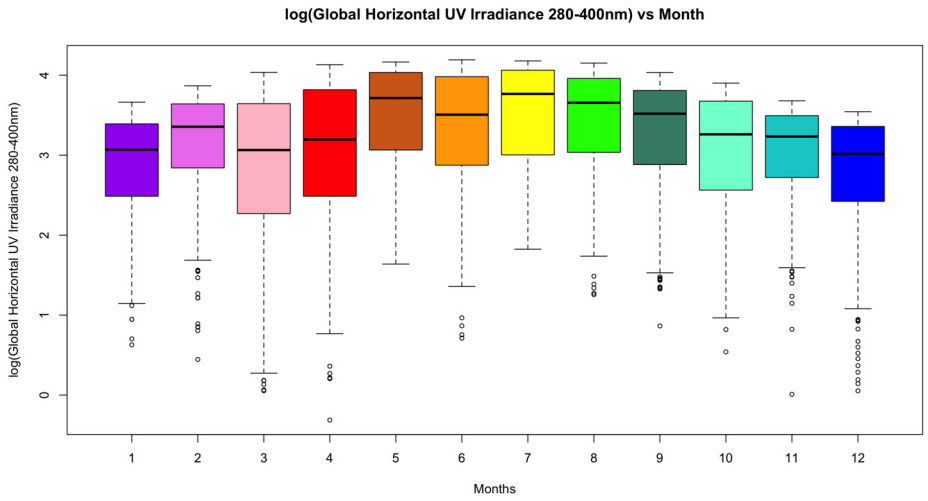

Figure 2.10: Boxblot of the response variable versus the factor covariate Month.

the third quartile in March, April, June, and November, while in the others it is not. The upper whiskers are short, so most of the values are high and similar to each other, while the lower ones are long, even with some outliers, because there are few low values.
Figure 2.11 illustrates the associations of the factor Hour with the response variable. As expected, the response variable values increase from the morning to the afternoon and then decrease. There is high variance early in the morning and late in the afternoon, while in the mid hours there is low, and the median is also more equidistant between the first and the third quartile. The lower whiskers are always longer than the upper. As in all the previous graphs, there are outliers probably due to storms or other meteorological phenomena sometimes happening during the day.
Figure 2.12 illustrates the assessment of the factor Cloud type with the response variable.
There are no observations for levels 1, 5, 10, 11, 12, and 15, so I will not consider them. When the sky is clear (type 0), the response variable values are high, and the boxplot is not symmetric: the bottom whisker is longer than the upper, and there are plenty of outliers. In foggy days (type 2), the values are close to the ones of a clear sky with lower variance. When the sky is full of water (types 3-4-6), the colder the water, the less the value of Global Horizontal UV Irradiance, the less the outliers, but the more the symmetry in the graph. It is probably due to the rain that should let the solar radiations pass easier than the ice, but in every case, the values are more stable during the day. When there are clouds like cirrus (type 7), the situation is similar to the clear sky but with a few smaller values. In cloudy days (types 8-9), the denser the clouds are at low altitudes, the lower the Global Horizontal UV Irradiance values and the lower the variance in the data. Therefore, the presence of clouds results in lower Global Horizontal UV Irradiance values and lower variance.
Figure 2.13 illustrates the assessment of the factor Surface albedo with the response variable.
The higher the response variable values, the higher the Surface albedo value. The median tends to be closer to the first than to the third quartile, and the variance seems

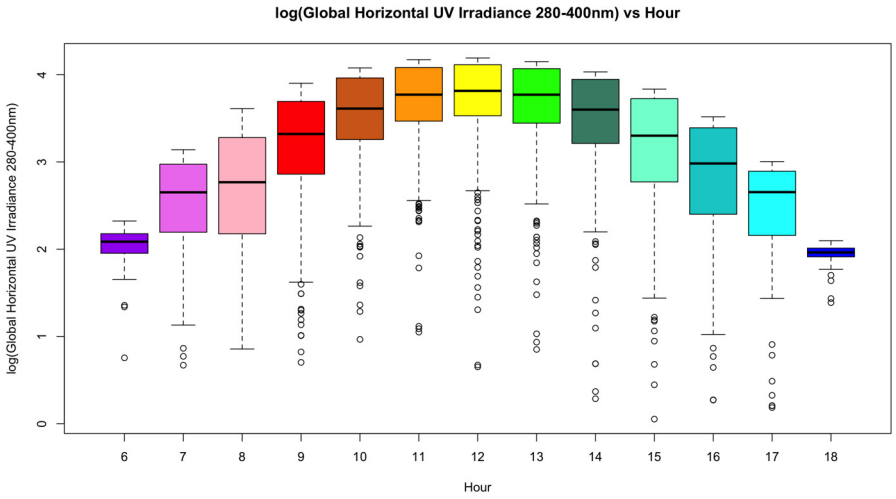

Figure 2.11: Boxblot of the response variable versus the factor covariate Hour.

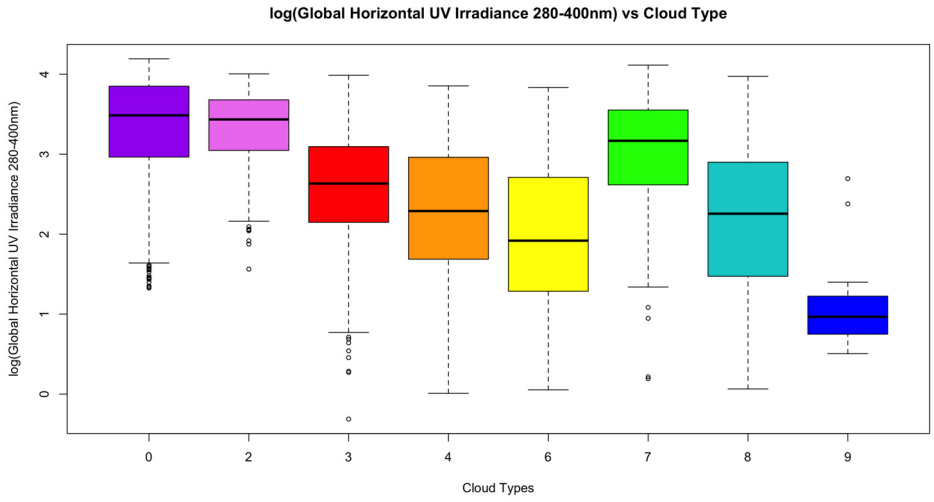

Figure 2.12: Boxblot of the response variable versus the factor covariate Cloud type.

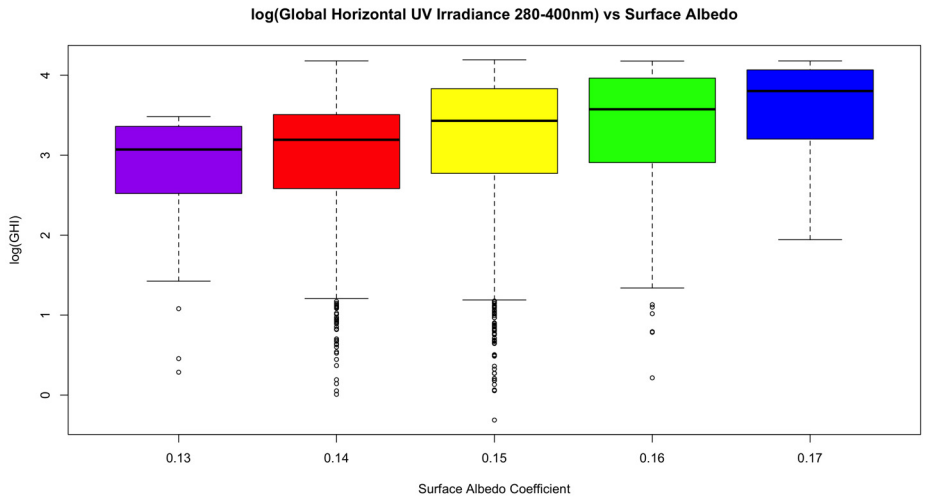

Figure 2.13: Boxblot of the response variable versus the factor covariate Surface Albedo.

lower in the border values, 0.13 and 0.17, than the others that, instead, have some outliers corresponding to the low values of the response variable. Notice that the level corresponding to 0.17 has slightly fewer observations than the others.
To conclude the preprocessing steps, the interactions have been analyzed between the covariates that could affect the response variable. The set of all the possible interactions built with all the covariates has more than 200 variables, so a selection based on the examined literature on solar irradiance has been proposed. Relevant interactions to be included in the modeling process are listed below:

- The interactions of Solar Zenith Angle, with the solar irradiance coefficients DNI and ClearskyDNI. They could be meaningful since they are indicative values to understand the intensity and propagation of solar irradiance, and they belong to the formula to calculate the coefficient GHI.
- The interactions of Cloud Type (Vignola, 2012) and Precipitable water with the solar irradiance coefficients DHI and DNI. They could be meaningful since the clouds and the amount of water they contain should be strictly related to the intensity and propagation of solar irradiance, especially the UV rays which are extremely sensitive.
- The interactions of Surface albedo with the solar irradiance coefficients DHI, DNI, ClearskyDHI, and ClearskyDNI. The combined effect of the irradiance and the reflectivity index of the terrestrial surface could be strictly related to the extremely sensitive UV rays.

Finally, the set of covariates is reported in Figure 2.14 illustrates the assessment of the factor Surface albedo with the response variable.

```
 [1] "Month"                                          "Hour"
 [3] "DHI"                                            "DNI"
 [5] "Clearsky.DHI"                                   "Clearsky.DNI"
 [7] "Cloud.Type"                                     "Dew.Point"
 [9] "Solar.Zenith.Angle"                             "Surface.Albedo"
[11] "Wind.Speed"                                     "Precipitable.Water"
[13] "Wind.Direction"                                 "Relative.Humidity"
[15] "Temperature"                                    "Pressure"
[17] "Global.Horizontal.UV.Irradiance..280.400nm."    "Prec.Water2"
[19] "Pressure2"                                      "DHI2"
[21] "DNI2"                                           "Clear.DHI2"
[23] "Clear.DNI2"                                     "Rel.Humidity2"
[25] "Temperature2"                                   "Prec.Water3"
[27] "Pressure3"                                      "DHI3"
[29] "DNI3"                                           "Clear.DNI3"
[31] "Clear.DHI3"                                     "Rel.Humidity3"
[33] "Temperature3"                                   "DNI-SZA"
[35] "ClearDNI-SZA"                                   "Alb-DNI"
[37] "Alb-Clear.DNI"                                  "Alb-DHI"
[39] "Alb-ClearDHI"                                   "Cloud-DHI"
[41] "Cloud-DNI"                                      "PrecWater-DHI"
[43] "PrecWater-DNI"
```

Figure 2.14: The set of covariates after the preprocessing of the data. Factors are considered as a set of dummy variables: 11 for Month, 12 for Hour, 7 for Cloud type, and 4 for Surface albedo. They are neatly and consecutively distributed.

Chapter 3

Data mining methods

This chapter focuses on the data mining methods used to evaluate the relationship between the covariates, see Figure 2.14 and the natural logarithm of Global Horizontal UV Irradiance of wavelength in the range of 280-400nm, which is the response variable, and to make predictions. The methods based on the normal distribution of the response are Linear Regression, Spline Regression, Ridge Regression, Lasso Regression, Elastic Net, Adaptive Lasso (Zou, 2006), and Principal Component Analysis (Hastie, Tibshirani, and Friedman, 2011, Chapters 3-5) (Jaskes *et al.*, 2013, Chapter 6). Extensions of the Linear Regression based on the Skew-Normal distribution (Azzalini, 1985) and assumption for the response variable will be considered.

3.1 Premises

3.1.1 Normal distribution

The Normal (Gaussian) distribution is the most important in Statistics and is used as a baseline for lots of topics. The density function is

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2}, \tag{3.1}$$

where x is the value of the input variable, and μ, σ are the mean and the standard deviation of the function respectively. A random variable with a Normal distribution can be written as $X \sim N(\mu, \sigma^2)$.
The Normal distribution is symmetrical around the mean, most of the observations cluster around the central peak, and the values further away from the mean taper off equally in both directions. Extreme values in both tails of the distribution are similarly unlikely, also known as outliers.
A special case is the Standard Normal distribution, which is the same as the Normal, but two parameters have a fixed value: $\mu = 0$ and $\sigma^2 = 1$, whose density functions reported in Figure 3.1.

3.1.2 Skew-Normal distribution

The Skew-Normal (SN) distribution (Azzalini, 1985) is an extension of the Normal distribution that allows the presence of skewness. The density function is

$$f(x) = \phi(x)\,\Phi(\alpha\, x), \tag{3.2}$$

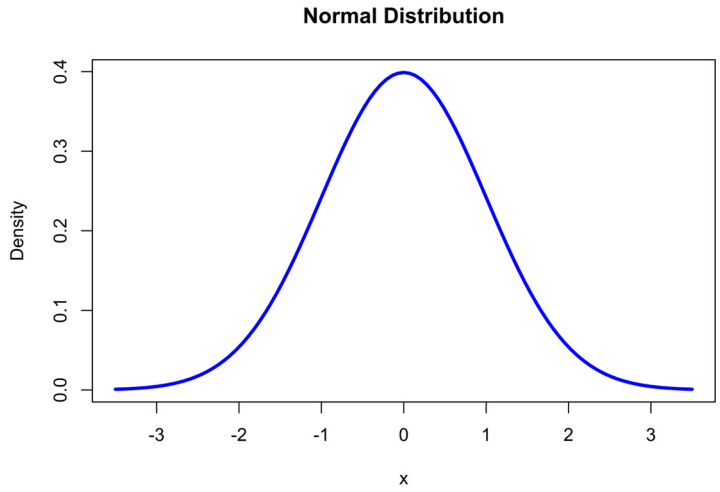

Figure 3.1: Graphical example of a Standard Normal distribution.
Source: `https://bookdown.org/a_shaker/STM1001_Topic_3/4-the-normal-distribution.html`.

where function $\phi(x)$ is the standard Normal distribution density function, $\phi(x) = f(x)$ in (3.1).
Function $\Phi(\alpha,\, x)$ represents the extension of the cumulative distribution function of the standard normal variable to include skewness, namely,

$$\Phi(\alpha,\, x) = \int_{-\infty}^{\alpha x} \phi(t)\, dt\,, \tag{3.3}$$

where α is the skewness parameter.
A random variable with a Skew-Normal distribution can be written as $X \sim SN(\mu,\, \sigma^2,\, \alpha)$.
Three relevant features of the Skew Normal distribution are listed below:

- The Normal distribution is a special case of the Skew Normal when $\alpha = 0$, corresponding to the absence of skewness.
- The skewness of the distribution increases as the absolute value of α increases.
- If the sign of α changes, the density is reflected on the opposite side of the vertical axis.

The density function for a S-N distribution with $\mu = 0$, $\sigma^2 = 1$, and $\alpha = 5$, $\alpha = -5$ are shown in the Figure 3.2.

3.2 Linear regression

The Linear Regression model is a supervised learning method for predicting a quantitative response variable. It assumes that the regression function is linear in the inputs, or covariates are explicative variables or predictors. While the linear regression model is a very good approximation of many relationships, it can fail providing satisfactory predictions when dealing with Big Data. However, it is often used as a baseline for

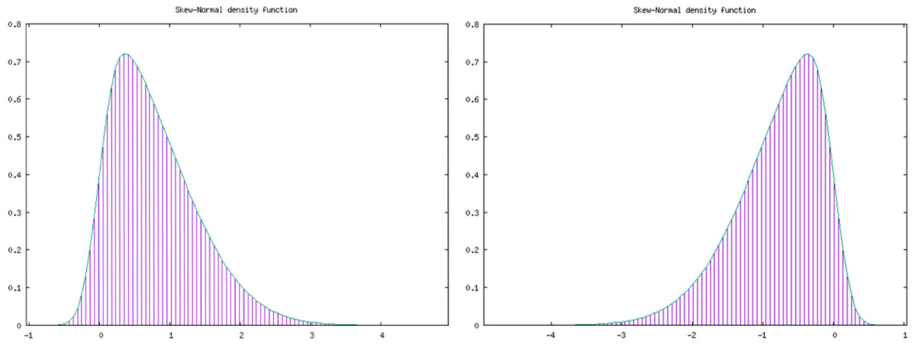

Figure 3.2: Graphical example of two Skew Normal distributions with $\alpha = 5$ and $\alpha = -5$. Source: `http://azzalini.stat.unipd.it/cgi-bin/sn-plot`.

more advanced analysis.
The mathematical formula of the linear regression model is specified as follows,

$$y = \beta_0 + \sum_{J=1}^{p} X_j \, \beta_j + \epsilon, \tag{3.4}$$

where $X^T = (X_1, X_2, ..., X_p)$ is the input vector of the p covariates, β_0 is the intercept, β_j's are the unknown coefficients of the X_j covariates, and ϵ is the error independent from X explaining the information about the phenomenon that the model does not.
The main constraint of this method is the linearity on the model covariates, which can be

- quantitative;
- polynomial with terms of degree greater than one;
- transformation of a quantitative variable applying a function to the whole input, such as the natural logarithm;
- qualitative coding of the levels of each variable with a numeric or dummy values (factors);
- interaction between variables represented by the multiplication of two covariates.

The set of training data is made of couples $(x_1, y_1)...(x_N, y_N)$, each x_i is a vector of the ith features measurement $x_i = (x_{i1}, x_{i2}, ..., x_{ip})^T$, and each y_i is the measure of the response variable.
The coefficients $(\beta_0, \beta_1, ..., \beta_p)^T$ can be estimated via least squares criterion. The idea is finding the coefficients vector $\beta = (\beta_0, \beta_1, ..., \beta_p)^T$ that minimizes the Residual Sum of Squares (RSS):

$$\begin{aligned} RSS(\beta) &= \sum_{i=1}^{N} \{y_i - f(x_i)\}^2 \\ &= \sum_{i=1}^{N} \{y_i - \beta_0 - \sum_{j=1}^{p} (x_{ij}\beta_j)\}^2. \end{aligned} \tag{3.5}$$

This criterion does not make assumptions on the model validity. It is reasonable if the observations are independent random draws taken from their population, but also if the y_i's are conditionally independent given the inputs x_i. The RSS measures the squared average lack of fit, and the Least squares criterion finds the best linear fit to the data minimizing RSS. An example is illustrated in Figure 3.3.

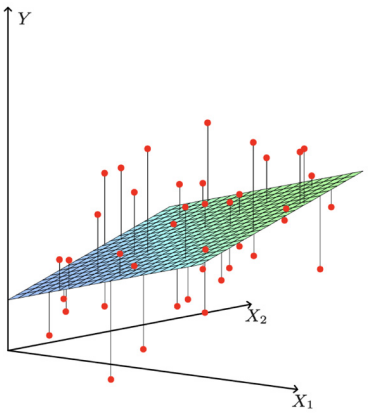

Figure 3.3: Least square criterion graphical example (Hastie, Tibshirani, and Friedman, 2011, Chapter 3).

To minimize the RSS (Hastie, Tibshirani, and Friedman, 2011, Chapter 3), we rewrite it as follows:

$$RSS(\beta) = (\mathbf{y} - \mathbf{X}\beta)^T (\mathbf{y} - \mathbf{X}\beta), \tag{3.6}$$

where $\mathbf{X}$ is the $N \times (p+1)$ matrix with an input vector containing 1 in the first position in each row, and $\mathbf{y}$ is the vector of outputs in the training set.
The minimum will be found differentiating with respect to β and setting the first derivative to zero. The final result gives the fitted values of y as

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\beta} = \mathbf{X}\left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y}. \tag{3.7}$$

The matrix $\mathbf{H} = \mathbf{X}\left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T$ is known as the "hat" matrix (Hastie, Tibshirani, and Friedman, 2011, Chapter 3). A problem rises when the columns of $\mathbf{X}$ are not linearly independent since $\mathbf{X}$ becomes non-full-ranked, and the Least square coefficients $\hat{\beta}$ are not uniquely defined anymore.

3.2.1 Backward-stepwise selection

The estimated linear regression model can include some relevant predictions, useful to explain the variability of y and other less relevant covariates. A selection technique is necessary to include only the relevant components in the model. Subset Selection refers to a set of procedures to automatically select a subset of the most meaningful variables dropping the others.
The Backward-Stepwise selection starts with the complete initial model including all the available covariates and iteratively deletes the predictors affecting the least on the fit one at a time.
The algorithm has to compare the model at the previous iteration with all the new

models generated at the current iteration removing one predictor at each step. Let p be the number of predictors and N be the number of observations of a given dataset. This algorithm scans just $1 + p(p+1)/2$ models in the worst case. It is very efficient, but it can be applied only if $N > p$, and it does not guaranteed to yield the best model containing a subset of the p predictors. However, in practice, it is often infeasible to achieve the best model. Among the many comparison criteria, we consider the following (Hastie, Tibshirani, and Friedman, 2011, Chapter 3).

1. Mallow's C_p

$$C_p = \frac{1}{N}\left(RSS + 2d\hat{\sigma}^2\right), \tag{3.8}$$

   where $\hat{\sigma}$ is an estimate of the variance of the error ϵ associated with each response measurement in (3.1), based on the model containing all the predictors. Therefore, the C_p adds the penalty $2d\hat{\sigma}^2$ to RSS to avoid underestimating the test error. If this criterion is chosen, the algorithm aims to minimize it at each iteration.

2. Akaike information criterion (AIC)

$$AIC = \frac{1}{N\hat{\sigma}^2}\left(RSS + 2p\hat{\sigma}^2\right), \tag{3.9}$$

   where $\hat{\sigma}$ is defined as above. It is a generalization of C_p applicable when the log-likelihood loss function is used if properly adapted. For models based on Least squares, C_p and AIC are proportional to each other.

3. Bayesian information criterion (BIC)

$$BIC = \frac{1}{N\hat{\sigma}^2}\left\{RSS + log(N)\, p\hat{\sigma}^2\right\}, \tag{3.10}$$

   where $\hat{\sigma}$ is defined as above. BIC is similar to C_p, but it applies a heavier penalty due to the logarithm, thus resulting in a selection of a smaller model. If this criterion is chosen, the algorithm aims to minimize it at each iteration.

4. Adjusted R squared (AdjR2)

$$Adjr2 = 1 - \frac{RSS\,/\,(N-p-1)}{TSS\,/\,(N-1)}, \tag{3.11}$$

   where $TSS = \sum_{i=1}^{N}(y_i - \overline{y})^2$ is the total sum of squares of the response, $\overline{y}$ is the true value of y and each y_i is the estimated value. It adds the penalization factor $1\,/\,(N-p-1)$ on the number of variables selected with respect to the R2 statistic to avoid underestimating the test error. If this criterion is chosen, the algorithm aims to maximize it at each iteration.

The main advantage of the Subset Selection techniques is to allow quickly skimming when there are many variables and poor information about the context. However, the disadvantage is that they do not consider the model interpretation made manually by data analysts.

3.3 Regression splines

Polynomial terms with higher degrees can be applied to the covariates to improve the fit of the linear regression model. However, such a choice can lead to a lack of interpretability in the final model and the risk of overfitting, that is, the risk of a model too well defined on the training data to the deterioration of the prediction accuracy in different sets of data. A good alternative is to rely on semiparametric solutions, which allow larger flexibility in modelling. Regression splines are a famous example of semiparametric instruments (Hastie, Tibshirani, and Friedman, 2011, Chapter 5). They are defined as a piecewise polynomial function $g(x)$ by dividing the input domain into contiguous intervals of arbitrary sizes delimited by points, named knots, and representing g by a separated polynomial in each interval. Moreover, two requirements on $g(x)$ are imposed:

- It must be continuous at each knot.
- Let M be the order of $g(x)$, then the derivatives up to order $M-2$ must be continuous, so the final curve results are smooth.

From this definition, we can state that a piecewise polynomial function of order M with continuous derivatives up to order $M-2$ is an order-M spline with knots σ_j, $j = 1, ..., K$. Hence, the cubic spline is the least order having smoothness on the knots. These fixed-knot splines, where least squares estimates are computed in each interval, are called regression splines and belong to semi-parametric regression models.
The natural spline is a more sophisticated solution. In addition to the splines features, it imposes a linearity constraint on the two ends of the range covered by all input in order to reduce the variability of the estimated model. Consequently, the result is analogous to the spline, but more stable estimates on the boundaries are computed.
In general, splines aim to choose a lower degree polynomial at each interval of $g(x)$ instead of choosing one of a high degree. Criteria, such as AIC and BIC, are used to select the best polynomial term for each interval.
The regression splines model allows more flexibility than linear regression, maintaining good interpretability. However, there is a trade-off to take care of: The more the knots, the more the flexibility, but the less the interpretability since the function loses smoothness. Cross-validation should be used to find the best trade-off.

3.4 Shrinkage methods

In the case of large dimensions of the dataset, mainly in terms of the number of predictors, linear regression models can be not satisfactory or even impossible to be estimated. When problems of identifiability and efficiency occur with linear regression, shrinkage techniques are preferable. They aim to shrink the coefficients of the variables by imposing a penalty on the function to minimize depending on the chosen method. In this way, the fitting of the model could be improved o slightly got worse, but the variability associated with the estimates is smaller. There is always a trade-off to take care of: The more the penalty affects the equation, the more the problem of high variance is faced, and the more information about the variables is lost.
Shrinkage methods solutions are not equivariant under scaling of the inputs. Hence, before applying them, it is needed to standardize the inputs.

3.4.1 Ridge regression

The Ridge Regression method (Hoerl and Kennard, 1970) (Hastie, Tibshirani, and Friedman, 2011, Chapter 3) aims to shrink the coefficients of the variables without making them equal to zero.
The ridge regression coefficients are evaluated minimizing the residual sum of squares plus the following L_2 norm:

$$\hat{\beta}^{ridge} = \arg\min_{\beta} \left\{ \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\,\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}. \tag{3.12}$$

The penalization factor is $\lambda \sum_{j=1}^{p} \beta_j^2$, where λ is the tuning parameter controlling the trade-off between fitting and penalization. If λ is too small, the penalty factor does not affect the residual sum of squares, obtaining a result similar to the linear regression. Thus the coefficients have not really shrunk. Note that if $\lambda = 0$, the penalty becomes equal to zero going back to the residual sum of squares, and the ridge regression will produce the least squares estimates. If λ is too big, the weight of the penalty factor is dominant in the equation, losing too much information about the model. Thus, ridge regression coefficient estimates will approach zero.
The more λ increase, the more the distortion since the flexibility decreases, the less the variance. Cross-validation is a good solution to choose the best value for λ to optimize the trade-off.
The method imposes a size constraint on the coefficients. This can be more explicit if it is rewritten as an optimization problem:

$$\begin{aligned} \hat{\beta}^{ridge} = \arg\min_{\beta} & \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\,\beta_j \right)^2 \\ \text{subject to } & \sum_{j=1}^{p} \beta_j^2 \leq t. \end{aligned} \tag{3.13}$$

Therefore, ridge regression alleviates the two main problems of linear regression: the coefficients of correlated variables are poorly determined and the lack of performance for data with high variance.

3.4.2 Lasso

Lasso (Least Angle Shrinkage and Selection Operator) (Tibshirani, 1996) (Hastie, Tibshirani, and Friedman, 2011, Chapter 3) is an alternative shrinkage method, which provides variable selection. According to lasso, the penalty to be minimized is

$$\hat{\beta}^{lasso} = \arg\min_{\beta} \left\{ \frac{1}{2} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\,\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}, \tag{3.14}$$

where the penalization factor $\lambda \sum_{j=1}^{p} |\beta_j|$, includes a L_1 norm of the coefficients in place of the L_2 norm that characterize ridge regression.
Since the L_1 norm allows the variable coefficients to be equal to zero, lasso can provide model selection and can result in models being easier to interpret if compared to ridge regression. As a price to pay, the solution is nonlinear in the response variable, so it has no closed form expression, nor are the associated variances.

The value of the tuning parameter λ is crucial as it is for ridge regression. Sufficiently large *lapbda* can force some of the coefficient estimates to be zero. But when *lambda* is too large, then the loss of information might be relevant. A good selection of *lambda* is usually determined through cross-validation.
To highlight the different constraints with respect to ridge regression, the lasso approach can be rewritten as an optimization problem:

$$\hat{\beta}^{ridge} = \arg\min_{\beta} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\, \beta_j \right)^2$$
$$\text{subject to } \sum_{j=1}^{p} |\beta_j| \leq t, \tag{3.15}$$

where the constraint is now on the sum of the absolute coefficient values allowing them to be set to zero.

3.4.3 Elastic Net

Ridge regression and lasso are two approaches that apply regularization to the linear regression model by adding a penalty factor to its problem definition. Ridge regression considers a proportional shrinkage since it reduces the coefficients' absolute value without putting it to zero. Lasso carries out soft thresholding since it shrinks the coefficients' absolute value till it reaches zero.
From a graphical point of view, we can interpret the differences between the two methods by considering their optimization problem definition. Figure 3.4 illustrates the case of the model $y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \epsilon_i$. The light blue areas correspond to

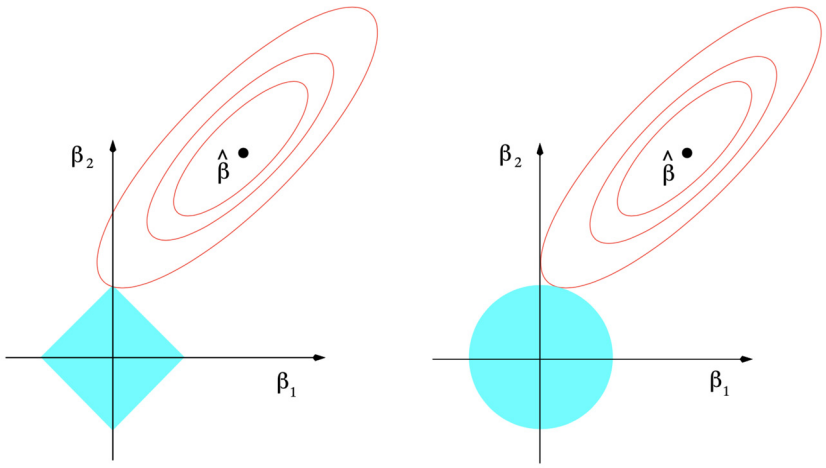

Figure 3.4: Lasso vs ridge regularization in a model with two variables (Hastie, Tibshirani, and Friedman, 2011, Chapter 3).

the couples of values (β_1, β_2) that satisfy the lasso constraint $|\beta_1| + |\beta_2| \leq t$ and the ridge $\beta_1^2 + \beta_2^2 \leq t$ constraint, on the left and right panel, respectively. The red ellipses

represent the least squares error function contours. In the case of ridge regression, the ellipses cannot touch the light blue area at the point where the axes meet because of the area's shape satisfying constraint, so the coefficients β_i cannot be zero. In the case of the lasso, the ellipses can touch the light blue area at that point, so the coefficients β_i can become zero.
Elastic net (Hastie, Tibshirani, and Friedman, 2011, Chapter 3) (Zou and Hastie, 2005) is the best trade-off between these two methods as it exploit their advantages. Elastic net is still based on the residual sum of squares, but the introduced penalty allows variable selection, as the lasso, maintaining the proportional shrinking as the ridge. The penalty is

$$\lambda \sum_{j=1}^{p} \{\alpha\beta_j^2 + (1-\alpha)|\beta_j|\}, \tag{3.16}$$

where λ is the usual training parameter, and α is a new coefficient ranging in $[0, 1]$. Term $\alpha\beta_j^2$ encourages highly correlated features to be averaged, like ridge regression, while term $(1-\alpha)|\beta_j|$ encourages a sparse solution in the coefficients, like lasso. Therefore, α is the coefficient regulating this trade-off and it should be chosen through cross-validation.
A graphical example of the elastic-net penalty result is illustrated in Figure 3.5, where the green line is the contour of the area satisfying ridge regression, the blue line is the contour of the area satisfying lasso, and the red line is the contour of the area satisfying elastic-net.

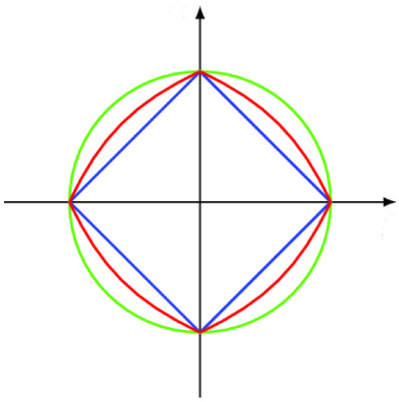

Figure 3.5: A comparison among the lasso, the ridge, and the elastic-net regularization on a model with two variables (Emmert-Streib and Dehmer, 2009). The contours have the same mean as in Figure 3.4, but here the focus is on the comparison. the green line is the contour of the area satisfying ridge regression, the blue line is the contour of the area satisfying lasso, and the red line is the contour of the area satisfying elastic-net.

3.4.4 Adaptive lasso

The Oracle property (Fan and Li, 2001) is important in Statistics and defines a quality requirement for estimators. It states that an oracle estimator must be consistent in

parameter estimation and variable selection.
A disadvantage of lasso is that its shrinkage can produce biased estimates for the large coefficients. In addition, variable selection can become inconsistent in some scenarios. Therefore, lasso does not satisfy the oracle property.
To face this problem, the adaptive lasso method has been proposed (Zou, 2006). As for the lasso, the method shrinks the coefficients and performs variable selection by minimizing the residual sum of squares with a penalty factor similar to L_1. However, it uses adaptive weights for penalizing different coefficients in the penalty.
The adaptive lasso estimates of the coefficients $\hat{\boldsymbol{\beta}}^{*(n)}$ are given by

$$\hat{\boldsymbol{\beta}}^{*(n)} = \arg\min_{\beta} \left\{ \left\| \mathbf{y} - \sum_{j=1}^{p} \mathbf{x}_j \beta_j \right\|^2 + \lambda_n \sum_{j=1}^{p} \hat{w} \, |\beta_j| \right\}, \tag{3.17}$$

where $\lambda_n \sum_{j=1}^{p} \hat{w} \, |\beta_j|$ is the weighted penalty, $\hat{\boldsymbol{\beta}}$ is a root-n-consistent estimator to β^* meaning that its variance is $O(1/n)$, $\hat{\boldsymbol{\beta}} = \frac{1}{|\hat{\boldsymbol{\beta}}|^\gamma}$ is the weight vector, and λ and γ are non-negative parameters controlling the penalty strength and the covariate weights respectively. Both parameters should be chosen through cross-validation.
The constraint of $\hat{\boldsymbol{\beta}}$ to be a root-n-consistent estimator is not strict. A good practice is to use the ridge regression coefficients obtained on the same data as the estimator $\hat{\boldsymbol{\beta}}$.
The adaptive lasso solution can be formulated as a convex optimization problem with the L_1 penalty as a constraint, in order to have a very efficient computation.

3.5 Principal component regression

The Principal Component Regression (PCR) (Hastie, Tibshirani, and Friedman, 2011, Chapter 3) (Jaskes *et al.*, 2013, Chapter 6) is an unsupervised learning technique that produces linear combinations Z_m, $m = 1, ..., M$ of the original standardized input X_j, $j = 1, ..., p$, named principal components, and uses them as the input of a linear regression model. It aims to reduce the problem dimensionality forcing $M < p$, but does not perform variable selection since it uses all the predictors to generate the principal components. Formally, the Z_m are computed as follows:

$$Z_m = \sum_{j=1}^{p} \phi_{jm} \, X_j, \tag{3.18}$$

where $\phi_m = (\phi_{1m}, \phi_{2m}, ..., \phi_{pm})^T$ is the loadings vector, which is made by the m-th principal component coefficients. The purpose of PCR is to build the smallest set of principal components that maximizes the caught variability of the data.
The first principal component is built looking for the loadings vector ϕ_1 that maximizes the variance of the data.

$$\max_{\phi_{11} \ldots \phi_{p1}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=1}^{p} \phi_{j1} \, X_{ij} \right)^2 \right\}$$
$$\text{subject to} \sum_{j=1}^{p} \phi_{j1}^2 = 1, \tag{3.19}$$

where variables $x_i, i = 1, ..., p$ are centred, i.e., with zero mean. The constraint $\sum_{j=1}^{p} \phi_{j1}^2 = 1$ is chosen in order to avoid arbitrarily large loadings, which can result in

arbitrarily large variance. The second principal component is built similarly, with the additional constraint of orthogonality between loadings vectors and ϕ_1. An example of the first two principal components is illustrated in Figure 3.6. Notice that the second principal component caught less variance than the first, by construction.

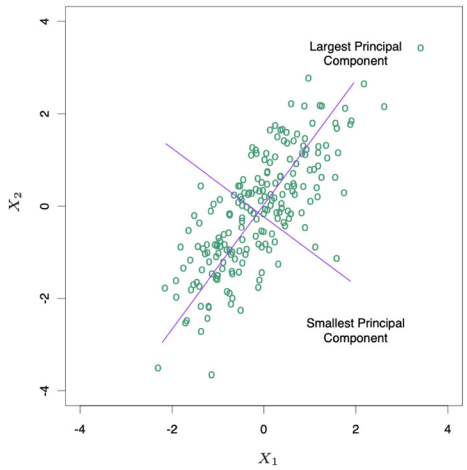

Figure 3.6: A graphical example of the first two principal components (Hastie, Tibshirani, and Friedman, 2011, Chapter 3).

In general, the m-th principal components are constructed in order to be uncorrelated with the previous ones and to catch the maximum of the remaining variability of the data not already caught by the $m-1$ principal component.
Principal Component Regression is a linear regression model where the covariates are substituted by the principal components, and the least squared estimates are still used for the computation:

$$y = \theta_0 + \sum_{i=1}^{M} \theta_i z_i + \epsilon, \tag{3.20}$$

where the response variable is y and the error is ϵ, as for the linear regression previously explained, θ_0 is the intercept, and $\theta_i, i = 1, ..., M$ are the coefficients associated with the principal components $z_i, i = 1, ..., M$ to estimate. The purpose of the analysis with PCR is to use as few principal components as possible, balancing the trade-off between accuracy and dimensionality reduction of the problem. In fact, principal components are generally less than the original covariates, and we can also choose a subset of them for the regression model according to graphical instruments, including graphs of explained variance. As a side effect, PCR helps to reduce overfitting since the fewer principal components, the fewer coefficient to estimate with respect to the original problem. Moreover, PCR is useful for avoiding multicollinearity, that is when high correlated variables are inserted in a model causing biased results because principal components are built through linear combinations, which are uncorrelated by definition. The drawback of this method is the lack of interpretation since we lose the references to the original data building the principal components.

Chapter 4

Data mining analysis

This chapter focuses on the data analysis performed by implementing the data mining methods described in Chapter 3. The analysis evaluates the relationship between the response variable given by the natural logarithm of Global Horizontal UV Irradiance of wavelength in the range of 280-400nm and the covariates reported in Figure 2.14.

4.1 Premises

The interpretative analysis of the models will be accompanied by technical considerations from the literature and graphical representations. The predictive analysis will be conducted by splitting the dataset into train and test sets of 70% and 30%, respectively.
A common baseline for each analysis is built by applying the preprocessing steps described in chapter 2. Further specific preprocessing steps will be implemented according to the features of the methods.

4.2 Linear regression

The analysis starts with linear regression (Hastie, Tibshirani, and Friedman, 2011, Chapter 3). Despite being a simple model, it gives fairly good results. Most of the variables are significant except for Solar Zenith Angle and its interactions, the interaction between Surface Albedo and Clear sky DNI, and the temperature. The residual standard error is 0.1153, the mean squared error is 0.0133, and although the Adjr2 coefficient is 0.9988, which is high, the residuals analysis is not very good, as illustrated in Figure 4.1.
The first graph "Residuals vs Fitted", in the top-left panel of Figure 4.1, shows a kind of deterministic, or not random, pattern of the data, which is in contrast with the expected behaviour of a model with satisfactory accuracy. Despite most of the observations being fitted well by the model, some are not. Such behaviour suggests that polynomial terms with degrees greater than one in the model could be useful.
The "Scale-Location" plot in the bottom-left panel of Figure 4.1 represents the same content as Residuals vs Fitted but standardizes and scales the data to remove position effects. It shows the same trends but accentuated, especially in the points cloud head, confirming the previous suggestions.
The "Normal (Quantile-Quantile)" plot in the top-right panel of Figure 4.1 compares

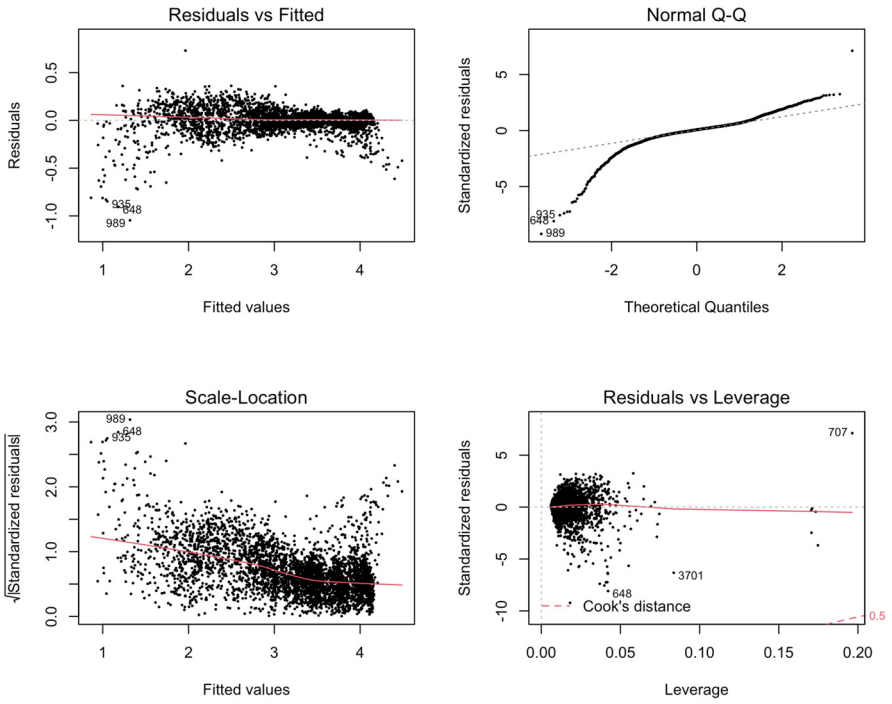

Figure 4.1: Residual analysis graph of the linear regression model with interactions.

the theoretical quantiles of the normal distribution with the empirical ones obtained by the model. The dotted line represents the equality of the quantiles, so a model with a satisfactory fit should have points close to the line. This is not the case for the examined model, suggesting the need to modify it.
The "Residuals vs Leverage" graph in the bottom-right panel of Figure 4.1 represents the standardized residuals according to the observations' weight on the fitted model. The points in the top-right area are called leverage points. The ones located over the Cook's distance leverage curve equal to one are influential because they affect the results and must be treated apart. In this case, there are none of them.
Therefore, from the residual analysis, the estimated linear regression model needs to be substantially improved. Since the residuals show a curved trend, a suggestion could be to add polynomial terms with degrees higher than one.

4.2.1 Polynomial features

The basic linear regression model has been extended by the inclusion of quadratic and cubic terms for a subset of variables that could be relevant: DHI, DNI, Clear sky DHI, Clear sky DNI, Precipitable water, Pressure, Relative humidity, and Temperature. In fact, despite the high variance, the four solar irradiance coefficients, Temperature, and Relative humidity have graphically shown a curved trend when tested against the response in Figures 2.7, 2.8, and 2.9. Moreover, Precipitable water and Pressure should affect the intensity and the direction of short wavelengths of solar rays.
Because of the high number of variables for training the model, the selection of the

relevant covariates in the linear regression model has been carried out using a backward strategy and adopting the selection criteria described in Section 3.2.1, namely Adjr2, Mallow's C_p, and BIC. Graphs 4.2, 4.3, and 4.4 show the leading to the choice of relevant variables for each criterion.
The procedure results are printed starting from the bottom and going up to the first line, where the variables associated to a black rectangle are selected for the best model. According to Adjr2, the best model has been built by selecting 62 variables over 72, as illustrated in Figure 4.2. In this case, ten variables were dropped, so the selection is

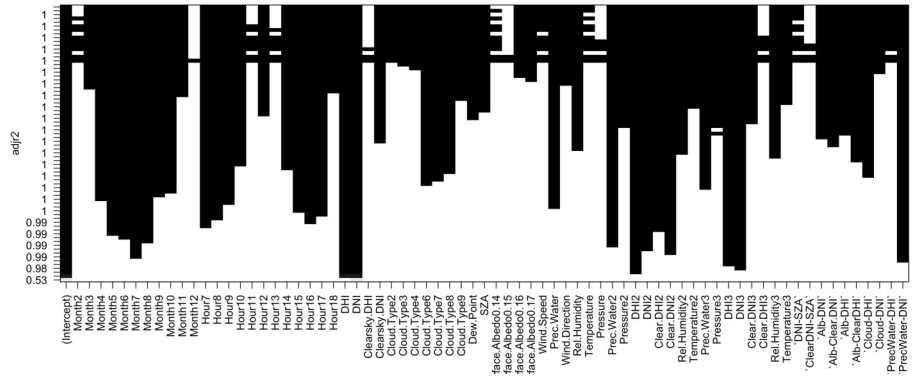

Figure 4.2: Graph of the selected variables using Adjr2.

not so heavy.
According to Cp, the best model results from the selection of 60 variables over 72, as illustrated in Figure 4.3. In this case, the selection is still not heavy.

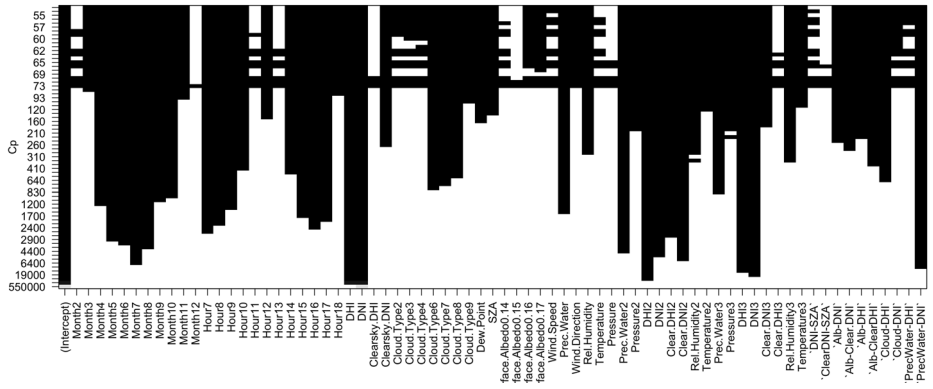

Figure 4.3: Graph of the selected variables using Mallow's C_p.

According to BIC, the best model has been obtained by selecting 52 variables over 72, as illustrated in Figure 4.4. In this case, the selected variables have significantly decreased.
Figure 4.5 shows a comparison among the results of the three criteria in terms of number of selected variables. Looking at the graphs, the choice fell on the best model according to BIC because it allows heavier selection.

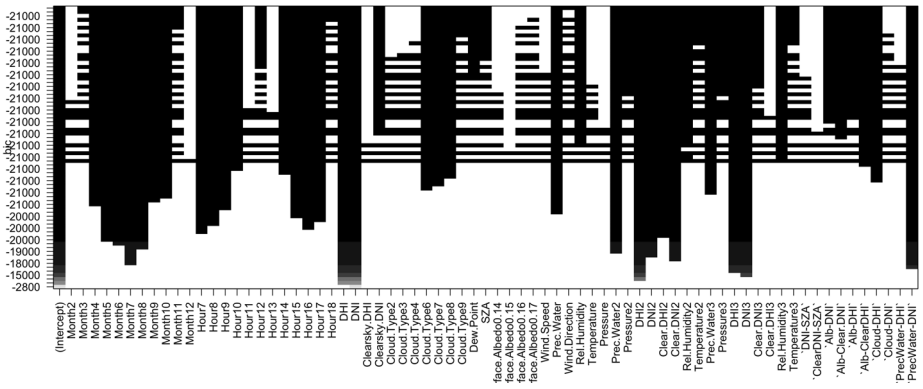

Figure 4.4: Graph of the selected variables using BIC.

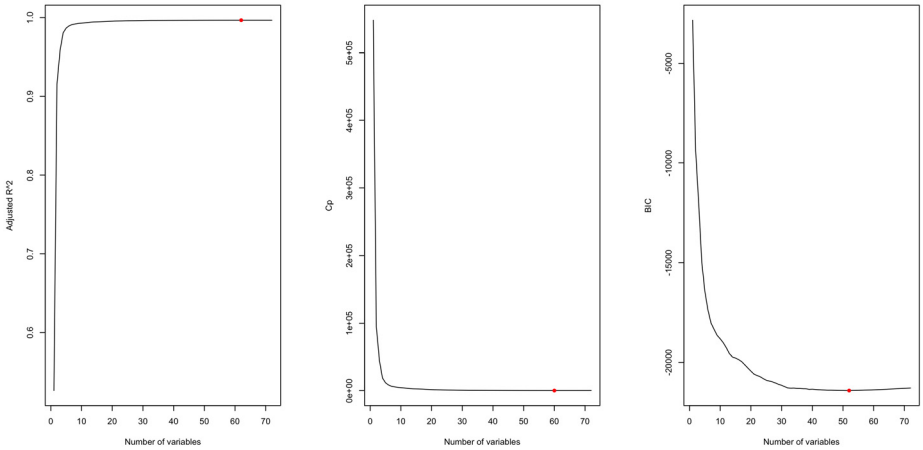

Figure 4.5: Comparison graph among the criteria Adjr2, Cp, and BIC for the variables selection.

The variables of the model selected by BIC are listed in Figure 4.6.

```
 [1] "(Intercept)"                  "Month3"                       "Month4"
 [4] "Month5"                       "Month6"                       "Month7"
 [7] "Month8"                       "Month9"                       "Month10"
[10] "Month11"                      "Hour7"                        "Hour8"
[13] "Hour9"                        "Hour10"                       "Hour12"
[16] "Hour14"                       "Hour15"                       "Hour16"
[19] "Hour17"                       "Hour18"                       "DHI"
[22] "DNI"                          "Clearsky.DNI"                 "Cloud.Type6"
[25] "Cloud.Type7"                  "Cloud.Type8"                  "Cloud.Type9"
[28] "Dew.Point"                    "Solar.Zenith.Angle"           "Precipitable.Water"
[31] "Wind.Direction"               "Relative.Humidity"            "Precipitable.Water2"
[34] "Pressure2"                    "DHI2"                         "DNI2"
[37] "Clearsky.DHI2"                "Clearsky.DNI2"                "Relative.Humidity2"
[40] "Temperature2"                 "Precipitable.Water3"          "Pressure3"
[43] "DHI3"                         "DNI3"                         "Clearsky.DNI3"
[46] "Relative.Humidity3"           "Temperature3"                 "`Surface.Albedo-DNI`"
[49] "`Surface.Albedo-Clearsky.DNI`" "`Surface.Albedo-DHI`"        "`Surface.Albedo-Clearsky.DHI`"
[52] "`Cloud.Type-DHI`"             "`Precipitable.Water-DNI`"
```

Figure 4.6: Selected variables of the best model according to the BIC criterion. The variables at the power of two are written with the suffix 2, the variables at the power of three are written with the suffix 3, and the interactions are written with the symbol - combining the two names of the variables.

The selection seems coherent with the expectations. Month and Hour have been chosen almost at each level. Cloud type has chosen only when the sky is full of clouds without being full of water: it seems the presence of clouds affects the response independently of the water contained. The meteorological and atmospheric predictors have more impact in the model when quadratic or cubic than when they are linear, coherently to their high variability, so polynomial terms with a high degree should fit better. The solar irradiance coefficients are always relevant, but Surface Albedo is never. The refraction index is not meaningful alone because it is a numeric coefficient associated with solar irradiance measures, but it is relevant when combined with DNI and DHI. The combined effect of DHI with Cloud type and Precipitable water with DNI are relevant, coherently with the fact that DHI represents the solar irradiance scattered by clouds, and solar irradiance coming perpendicular is likely affected by the amount of water on the clouds it passes through. The other interaction terms and some variables, such as Wind speed, have not been selected by the procedure.
As expected, the model gives better results in terms of evaluation of the fitting if compared to the basic simplified version previously described. The Adjr2 has risen to 0.9998, while the residual standard error and the mean squared error have dropped to 0.0424 and 0.0018, respectively. The residuals analysis has improved, as illustrated in Figure 4.7.
The "Residual vs Fitted" graph in the top-left panel of Figure 4.7 shows a uniform points cloud, where deterministic patterns do not appear. This is a big improvement with respect to the basic linear regression model, albeit remaining a little deterministic trend on the left of the graph. The "Scale-Location" graph in the bottom-left panel of Figure 4.7 shows the same results despite presenting higher variance and accentuating the little deterministic trend. Both the plots have improved a lot, but they are still not satisfactory. The "Normal (Quantile-Quantile)" plot in the top-right panel of Figure 4.7 has improved a lot since most of the points are now on the dotted line. By the way, some are still too detached, mostly at the beginning. The "Residuals vs Leverage" graph in the bottom-right panel of Figure 4.7 confirms the absence of points whose

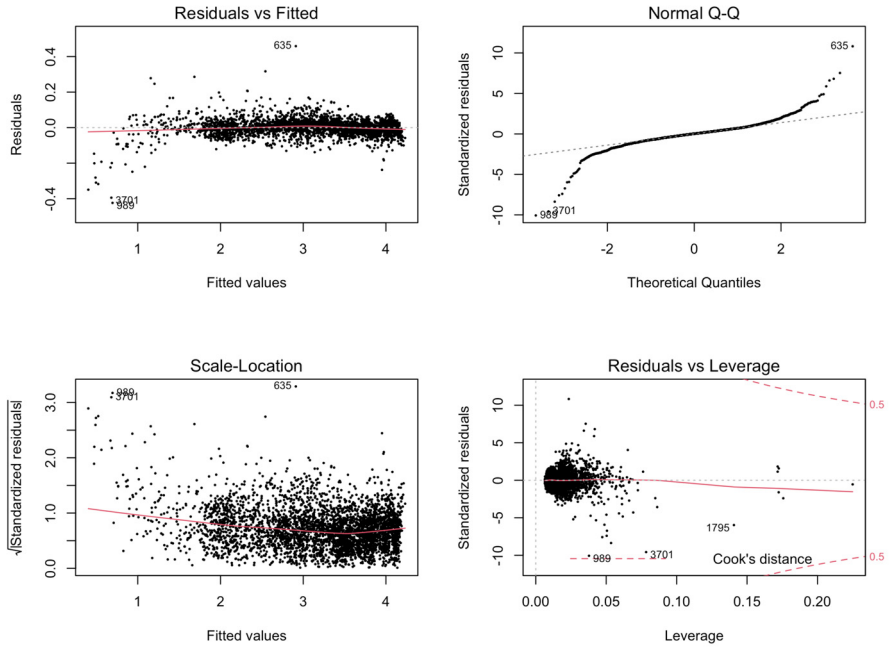

Figure 4.7: Residual analysis graph of the linear regression model with quadratic and cubic terms, and interactions.

Cook's distance is greater or equal than one.
The variables' coefficients different from zero, together with their associated standard error surrounded by round brackets, are reported in Table 4.1. From that, it is possible to state which variables weights the most to interpret the results.

Table 4.1: Estimate of the coefficients for the variables in the linear regression model with polynomial features. Standard error in parentheses.

Month3	Month4	Month5	Month6
-2.47e-04 (6.42e-03)	2.01e-02 (8.41e-03)	4.09e-02 (9.78e-03)	5.43e-02 (1.03e-02)
Month7	Month8	Month9	Month10
8.15e-02 (1.00e-02)	6.31-02 (9.21e-03)	2.44e-02 (7.53e-03)	1.71e-02 (5.90e-03)
Month11	Hour7	Hour8	Hour9
8.62e-03 (4.45e-03)	6.16e-02 (7.67e-03)	4.10e-02 (1.12e-02)	5.22e-03 (1.48e-02)
Hour10	Hour12	Hour14	Hour15
-2.70e-02 (1.76e-02)	-6.75e-02 (2.01e-02)	-2.58e-02 (1.71e-02)	1.31e-02 (1.41e-02)
Hour16	Hour17	Hour18	DHI
3.99e-02 (1.06e-02)	4.69e-02 (7.25e-03)	-3.29e-02 (7.27e-03)	2.11e-02 (3.48e-04)
DNI	Clearsky DNI	Cloud type 6	Cloud type 7
3.12e-03 (9.42e-05)	-1.39e-03 (2.66e-04)	-7.48e-02 (8.82e-03)	-7.09e-02 (6.63e-03)
Cloud type 8	Cloud type 9	Dew point	SZA
-5.63e-02 (9.61e-03)	-8.15e-02 (2.04e-02)	1.02e-02 (2.44e-03)	-8.58e-04 (2.58e-04)
Prec water	Wind direction	Rel humidity	Prec water^2

1.85e-01 (1.45e-02)	3.97e-05 (1.12e-05)	-1.57e-02 (2.9e-03)	-8.21e-02 (6.4e-03)
Pressure^2	DHI^2	DNI^2	Clearsky DHI^2
1.8e-06 (4.58e-07)	-5.39e-05 (6.92e-07)	-2.86e-06 (9.93e-08)	4.61e-06 (2.61e-07)
Clearsky DNI^2	Rel humidity^2	Temperature^2	Prec water^3
1.46e-06 (2.29e-07)	2.33e-04 (3.66e-05)	-3.38e-04 (1.01e-04)	8.87e-03 (9.29e-04)
Pressure^3	DHI^3	DNI^3	Clearsky DNI^3
-1.32e-09 (4e-10)	5.72e-08 (9.84e-10)	2.49e-09 (6.52e-11)	-5.36e-10 (1.3e-10)
Rel humidity^3	Temperature^3	Surf albedo - DNI	Surf albedo - Clearsky DNI
-1.34e-06 (1.88e-07)	4.57e-06 (1.56e-06)	-3.75e-03 (5.73e-04)	5.28e-03 (1.02e-03)
Surf albedo - DHI	Surf albedo - Clearsky DHI	Cloud type - DHI	Prec water - DNI
-1.46e-02 (2.34e-03)	2.17e-03 (8.46e-04)	5.07e-05 (5.23e-06)	1e-04 (5.83e-06)

At first glimpse, all the coefficients' absolute values are at least two orders of magnitude smaller than one, so none of them weighs a lot. The coefficients corresponding to the original variables have the highest but similar values meaning they are all very influential. The higher the polynomial degrees, the smaller the coefficients. This is coherent with the analysis since polynomial terms have been introduced to fit better the observations at both ends of the points cloud, which are a small subset of data. The interactions are spot on since their coefficients are quite high. The coefficients' sign determines if the predictors increase or decrease the response value. Some expected results are confirmed: the response is directly proportional to the solar irradiance coefficients, and the presence of clouds heavily affects the response decreasing its value. Due to their high variance, atmospheric and meteorological variables have discrepancies of signs with different degrees, so no information can be drawn. However, the linear term of each one is the most important and should tell the main behaviour apart from uncommon events. It can be inferred that the higher the temperature, the dew point, the pressure and the precipitable water, the higher the response, while the higher the relative humidity, the lower the response. Two interesting results will be investigated further: the precipitable water is directly proportional to the response, and the combined effect of surface albedo with the solar irradiance coefficients in the clear sky version and not, have discordant signs.
The model has generally improved by adding quadratic and cubic terms, but some other enhancements can be involved. Since the associations with the predictors and the response are very confused, polynomial terms could be added to an even greater degree, but the model would have become too complex, losing interpretability. A better approach could be the use of splines.

4.2.2 Spline regression

Spline regression (Hastie, Tibshirani, and Friedman, 2011, Chapter 5) allows to include more flexibility in the relationship with the response variable than linear regression with polynomial terms of higher degrees while maintaining good interpretability. Natural splines are preferred because they can provide more stable estimates, so they are used in the following analysis.
The preprocessing step can consider all variables represented by polynomial features with a degree greater than one in the baseline to be substituted by proper natural splines. The comparison of natural splines of different degrees for each variable is made by AIC. The best degree was often very high. Nevertheless, the improvements were

insignificant beyond degrees four and five, so low degrees have been chosen, maintaining a moderately simple model. The results are reported in Table 4.2.

Table 4.2: Results of the selection procedure of natural splines according to the AIC criterion.

Variable	Spline degree	AIC
DHI	4	-4562.53
DNI	4	-5418.983
Clear sky DHI	5	-3684.977
Clear sky DNI	2	-4666.87
Precipitable water	3	-2241.279
Pressure	3	-2328.924
Relative humidity	4	-3961.853
Temperature	4	-3678.019

As expected, the model with natural splines gives better results than the linear regression with polynomial features in terms of evaluation of the fitting. The Adjr2 is 0.9999, and the residual standard error and the mean squared error go down to 0.0361 and 0.0013, respectively. The residual analysis has improved by following better the curved pattern, as illustrated in Figure 4.8. The "Residual vs Fitted" plot in the

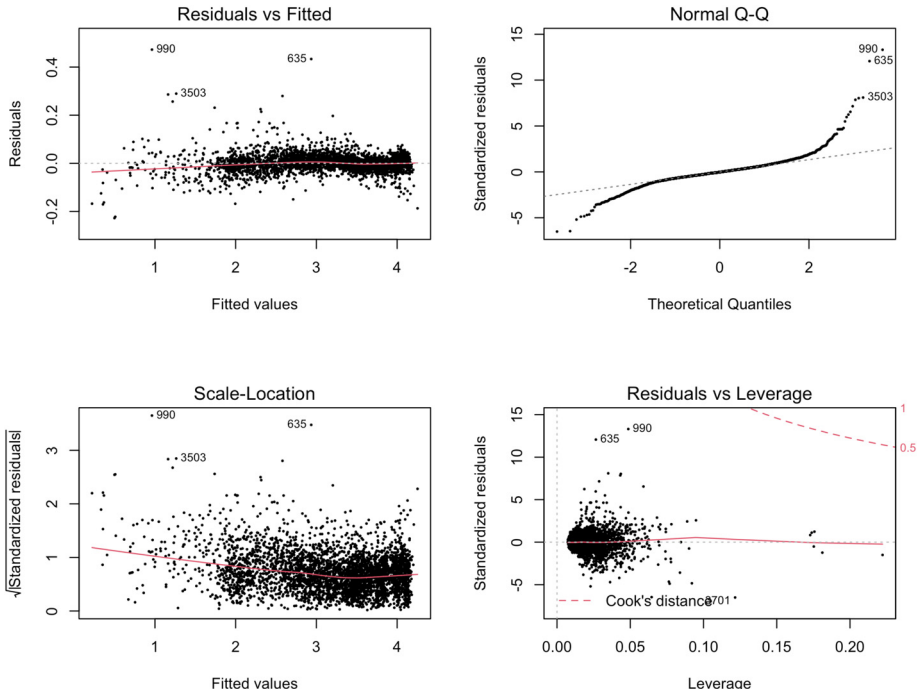

Figure 4.8: Residual analysis graph of the linear regression model with quadratic and cubic terms, and interactions.

top-left panel of Figure 4.8 does not show any noticeable deterministic pattern, even better than the "Residual vs Fitted" plot in Figure 4.7. This improvement is even more emphasized in the "Scale-Location" graph in the bottom-left panel of Figure 4.8. The

other two plots do not improve remarkably, as they were already excellent.
Looking at the variables' coefficients together with the associated standard errors surrounded by round brackets reported in Table 4.3, it is possible to state which variables affect the response variable the most to interpret the results.

Table 4.3: Estimate of the coefficients for the variables in the linear regression model with natural splines. Standard error in parentheses.

Month2	Month3	Month4	Month5
-4.04e-03 (3.73e-03)	1.35e-02 (4.64e-03)	4.05e-02 (5.65e-03)	7.05e-02 (6.15e-03)
Month6	Month7	Month8	Month9
8.55e-02 (6.33e-03)	1.09e-01 (6.4e-03)	8.30e-02 (6.14e-03)	4.24e-02 (5.73e-03)
Month10	Month11	Month12	Hour7
2.13e-02 (4.88e-03)	1.44e-02 (3.8e-03)	7.11e-03 (3.96e-03)	5.76e-02 (6.17e-03)
Hour8	Hour9	Hour10	Hour11
6.88e-02 (8.24e-03)	3.71e-02 (1.04e-02)	1.37e-02 (1.21e-02)	-3.24e-03 (1.30e-02)
Hour12	Hour13	Hour14	Hour15
-1.23e-02 (1.33e-02)	-4.68e-03 (1.29e-02)	1.50e-02 (1.18e-02)	4.11e-02 (1.01e-02)
Hour16	Hour17	Hour18	DHI
5.22e-02 (8.05e-03)	4.58e-02 (6.13e-03)	-1.10e-02 (6.45e-03)	1.68e+00 (3.52e-02)
DHI^2	DHI^3	DHI^4	DNI
2.46e+00 (8.06e-02)	4.61e+00 (1.59e-01)	3.54e+00 (1.82e-01)	1.44e+00 (5.13e-02)
DNI^2	DNI^3	DNI^4	Clearsky DHI
1.75e+00 (5.61e-02)	3.04e+00 (9.68e-02)	2e+00 (5.78e-02)	-1.56e-01 (3.02e-02)
Clearsky DHI^2	Clearsky DHI^3	Cleaersky DHI^4	Clearsky DHI^5
-1.57e-01 (4.02e-02)	-1.24e-01 (8.68e-02)	-2.76e-01 (1.63e-01)	-1.22e-01 (1.9e-01)
Clearsky DNI	Clearsky DNI^2	Cloud type 2	Cloud type 3
3.13e-03 (1.58e-01)	-1.39e-03 (7.25e-02)	-7.48e-02 (5.22e-03)	-7.09e-02 (4.72e-03)
Cloud type 4	Cloud type 6	Cloud type 7	Cloud type 8
3.12e-03 (5.52e-03)	-1.42e-03 (7.62e-03)	-3.66e-02 (5.77e-03)	-7.25e-03 (8.38e-03)
Cloud type 9	Dew point	SZA	Surf albedo 0.14
4.07e-02 (1.75e-02)	-5.61e-03 (9.30e-04)	-1.07e-03 (4.32e-03)	-9.67e-03 (7.96e-03)
Surf albedo 0.15	Surf albedo 0.16	Surf albedo 0.17	Wind speed
-1.33e-02 (1.28e-02)	-1.10e-02 (1.81e-02)	-6.14e-03 (2.4e-02)	1.86e-03 (6.18e-04)
Prec water	Prec $water^2$	Prec $water^3$	Wind direction
1.54e-02 (8.28e-03)	9.43e-03 (1.87e-02)	-1.57e-01 (1.37e-02)	2.87e-05 (9.71e-06)
Rel humidity	Rel $humidity^2$	Rel $humidity^3$	Rel $humidity^4$
1.04e-01 (1.77e-02)	1.42e-01 (1.94e-02)	2.39e-01 (3.74e-02)	1.06e-01 (2.26e-02)
Temperature	$Temperature^2$	$Temperature^3$	$Temperature^4$
9.93e-02 (1.37e-02)	1.44e-01 (1.70e-02)	2.46e-01 (3.11e-02)	1.61e-01 (2.38e-02)
Pressure	$Pressure^2$	$Pressure^3$	DNI - SZA
1.14e-02 (4.28e-03)	1.15e-02 (1.47e-02)	-9.11e-03 (7.14e-03)	-7.34e-06 (3.44e-06)
Clear DNI - SZA	Surf albedo - DNI	Surf albedo - Clearsky DNI	Surf albedo - DHI
1.02e-05 (6.33e-06)	-2.62e-03 (5.08e-04)	4.53e-03 (8.83e-04)	-1.97e-02 (2.67e-03)
Surf albedo - Clearsky DHI	Cloud type - DHI	Prec water - DNI	
6.41e-03 (3.67e-03)	3.5e-05 (4.54e-06)	1.12e-04 (4.97e-06)	

At first glimpse, all the coefficients' absolute values are greater than the linear regression ones with polynomial features, so singularly, they weigh more. The coefficients of solar irradiance indexes DHI, Clear sky DHI, DNI, and Clear sky DNI have the highest absolute value meaning they are all very influential, as for linear regression with polynomial features. As natural splines degrees increase, they do not lose importance differently from the polynomial degrees in the linear regression. This is coherent since the selection procedure of natural splines chooses only the most significant degrees. Looking at the meteorological and atmospheric predictors, we notice that Relative humidity and Temperature have high coefficients, not comparable to the solar irradiance indexes, but they are influential. As expected, Wind speed is almost irrelevant because we know from the literature that it should not affect Global Horizontal Irradiance so much. Surprisingly, the model has not assigned a high weight to the interactions, except for the ones including surface albedo. In fact, interactions with Surface albedo should be essential because they determine how solar irradiance is reflected and refracted. All the other variables have similar coefficient absolute values in the mean between the best and the worst.
DHI, Clear sky DHI, DNI, Clear sky DNI, Relative humidity, and Temperature have positive signs, so they are directly proportional to the response. This satisfies our expectation since the Global Horizontal Irradiance is part of the irradiance. Therefore, the higher the temperature and the relative humidity, typically the hotter and the sunnier the day. Moreover, it can be inferred that the higher the pressure, the higher the response, in agreement with the knowledge that high pressure means sunny days. The relevant interactions tell the same behaviour as for the linear regression model: the combined effect of surface albedo with the solar irradiance coefficients in the clear sky version and not have discordant signs.

4.2.3 Predictions

Predictions with the three different models based on linear regression have been performed. Since the observations are too many, the best representation for predictions is a graph with observed values on the abscissa axis, predicted values on the ordinate axis, and the bisector of the first and third squares, which corresponds to the perfect prediction of the observations.
The predictions provided by linear regression without polynomial terms are illustrated in Figure 4.9. The mean squared error is 0.0178, the root mean squared error is 0.1335, and the AdjR2 is 0.9762. The results are generally not so bad. Most of the points are on the red bisector line or close to it. However, some predictions are incorrect, producing a higher value of Global Horizontal UV Irradiance than the one observed.
The predictions provided by linear regression with polynomial terms are illustrated in Figure 4.10. The mean squared error is 0.0018, the root mean squared error is 0.0429, and the AdjR2 is 0.9968. The results improved a lot. Very few predictions are incorrect and detached from the red bisector line.
The predictions provided by linear regression with natural splines are illustrated in Figure 4.11. The mean squared error is 0.0013, the root mean squared error is 0.0364, and the AdjR2 is 0.9976. The results are slightly better than the previous. Predictions are almost on the red bisector line everywhere except for Global Horizontal UV Irradiance values over 50. However, the prediction error is very restricted.
These results agree with our expectations since linear regression with natural splines has been confirmed to be the best model. In general, if Global Horizontal UV Irradiance values are low, the predictions are precise, but the higher the values, the higher the

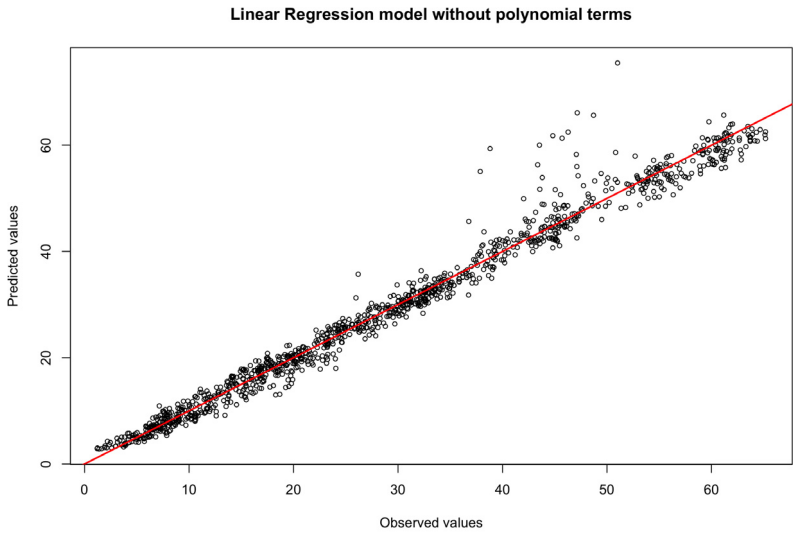

Figure 4.9: Predictions of Global Horizontal UV Irradiance of wavelength in the range of 280-400nm, performed by linear regression without polynomial features.

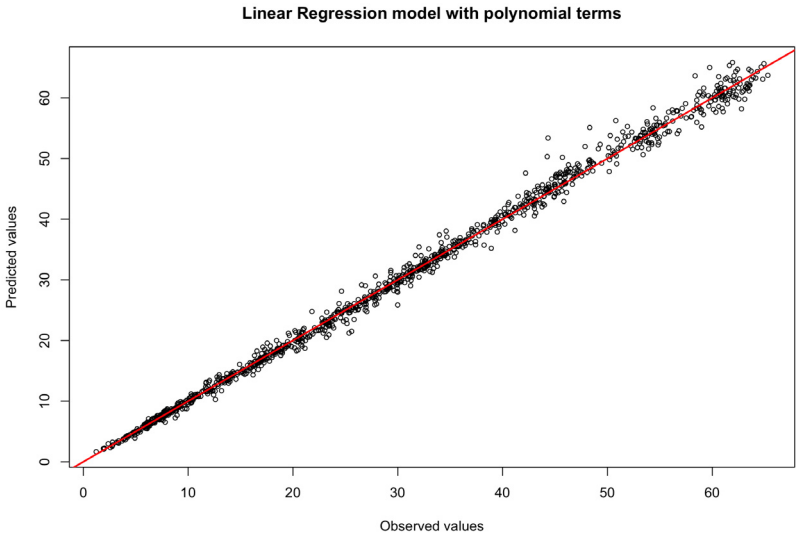

Figure 4.10: Predictions of Global Horizontal UV Irradiance of wavelength in the range of 280-400nm, performed by linear regression with polynomial features.

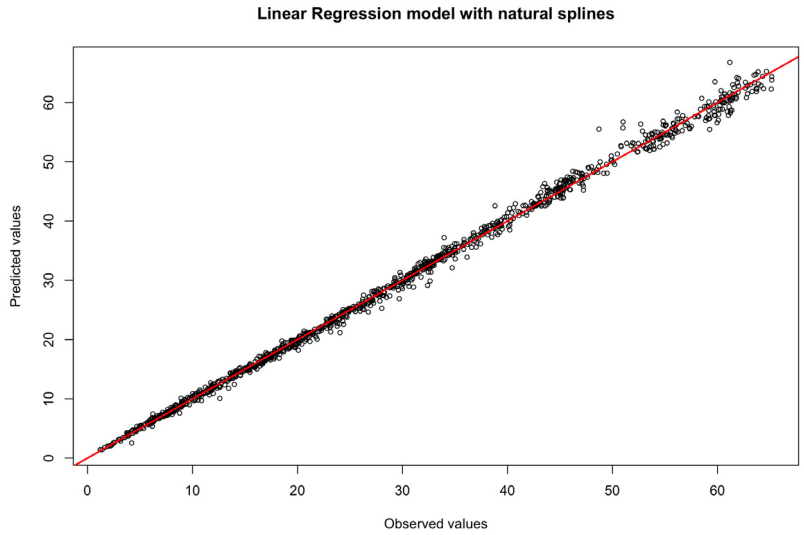

Figure 4.11: Predictions of Global Horizontal UV Irradiance of wavelength in the range of 280-400nm, performed by linear regression with natural splines.

prediction error. More flexible models could guarantee that the predictions remain accurate when the values are low but improve when the values increase.

4.3 Shrinkage methods

The results obtained by the linear regression model including natural splines are satisfactory. However, the natural splines model contains lots of variables, and only a few have been dropped. Furthermore, the data has high variance, as the graphs in Chapter 2 have shown. Therefore, shrinkage methods have been implemented to try to achieve better results against high variance and to focus on fewer variables.
The analysis begins with Ridge regression and Lasso, then continues with two more advanced techniques, Elastic-Net and Adaptive Lasso. These four methods do not require further preprocessing steps starting from the common baseline.
The metric to assess the quality of the results from the following model is the explained deviance, which is the amount of information the model can capture. Since the explained deviance is equivalent to the Adjr coefficient for linear regression, then it will be used for methods comparisons.
Preprocessing steps are executed for every shrinkage method adding quadratic and cubic terms for the same variables as for linear regression with polynomial features: DHI, DNI, Clear sky DHI, Clear sky DNI, Precipitable water, Pressure, Relative humidity, and Temperature. Thus, we leave the shrinkage methods to decide which are significant to explain the response variable.

4.3.1 Ridge regression

Ridge regression (Hoerl and Kennard, 1970) (Hastie, Tibshirani, and Friedman, 2011, Chapter 3) aims to shrink the variables' coefficients without performing variable selection. This method does not fit the purpose of the analysis since is does not drop any

variables, but it is needed to implement Adaptive lasso and useful to better explain the results of Elastic net.
At first, ridge regression has been performed considering multiple values of λ automatically. We can build a graphical representation of the trends of the variables' coefficients as the logarithm of λ increases, illustrated in Figure 4.12. We notice that the number

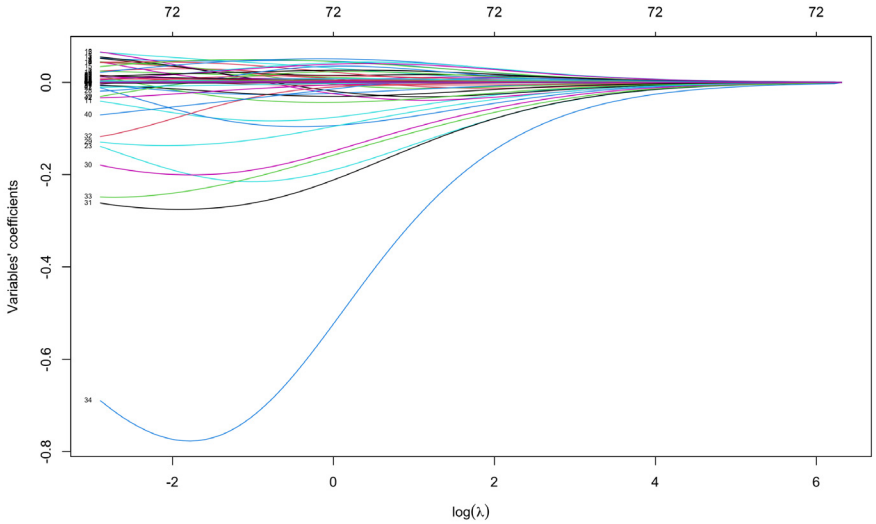

Figure 4.12: Graphical representation of the trend of the variables' coefficients as the value of $log(\lambda)$ increases in the ridge regression model. Each coloured line represents the trend of a coefficient, and the small numbers on the left correspond to their order in the dataset. The number of variables that survived in the model is written at the top of the graph.

of variables always remains the same, meaning that no coefficients are shrunk to zero, as expected. The shrinkage becomes stronger as the value of $log(\lambda)$ increases since the coefficients' values become closer to zero.
Moreover, an analysis of deviance has been performed. It represents the amount of information the model can explain about the response variable: the higher the explained deviance, the more accurate the model. Figure 4.13 shows a graph of its trend according to the value of $log(\lambda)$. From the two previous graphs, we can see that as the value of λ increases, the regularization terms weigh more, so the coefficients are shrunk more, and the model explained deviance drops down.
Cross-validation has been executed to tune the hyperparameter λ. The result of the procedure is shown in Figure 4.14. The mean squared error rises rapidly when $log(\lambda)$ is greater than 0. The two dotted lines represent the values of $log(\lambda)$: on the left, λ minimizes the mean squared error, and on the right, λ is at a distance equal to 1 standard error ($1\,SE$) from the other. The second is suggested by the literature as it empirically gives good results since it allows more regularization at the price of an increase of mean squared error.
In this case, the value of λ minimizing the mean squared error has been chosen. Hence, λ is 0.0552, the mean squared error is 0.011, and the root mean squared error is 0.1049. Finally, the ridge regression coefficients $\hat{\beta}^{ridge}$ have been computed, substituting the proper value of λ in (3.12). The explained deviance is 98.03%, which is a relatively

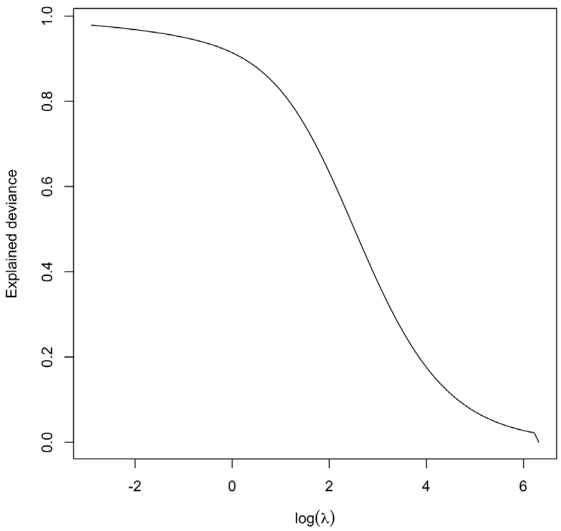

Figure 4.13: Graphical representation of the explained deviance in the ridge regression model according to the increasing value of $log(\lambda)$.

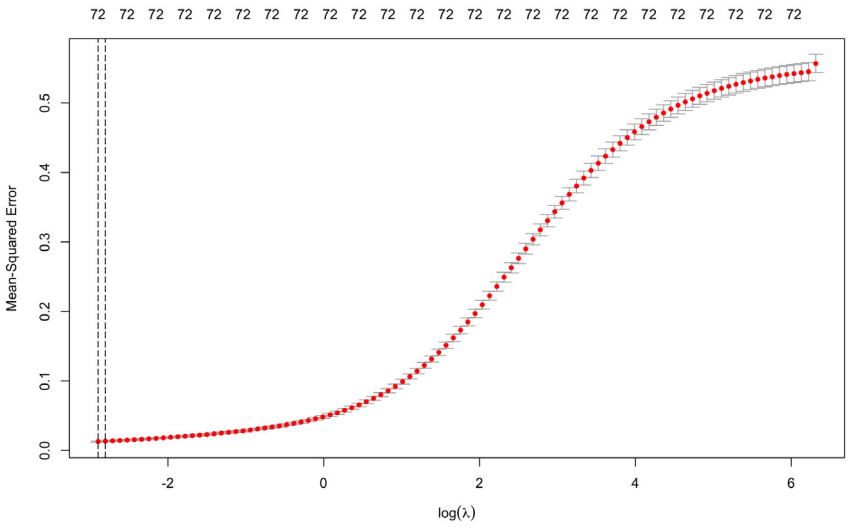

Figure 4.14: Graphical representation of the mean squared error calculated during the cross-validation for ridge regression as the value of $log(\lambda)$ increases.

good value. Nevertheless, it is not satisfactory since no variables are eliminated and, in similar conditions, linear regression performs better.

4.3.2 Lasso

Lasso (Tibshirani, 1996) (Hastie, Tibshirani, and Friedman, 2011, Chapter 3) is one of the most famous shrinkage methods, and refinements of it have been proposed in recent years. Lasso aims to shrink the variables' coefficients making variables selection, so it has been preferred to ridge regression in this thesis.
Cross-validation has been performed to tune the hyperparamenter λ. The purpose is to find the best trade-off allowing variables selection and maintaining a good explained deviance. The cross-validation result is shown in Figure 4.15. The mean squared error

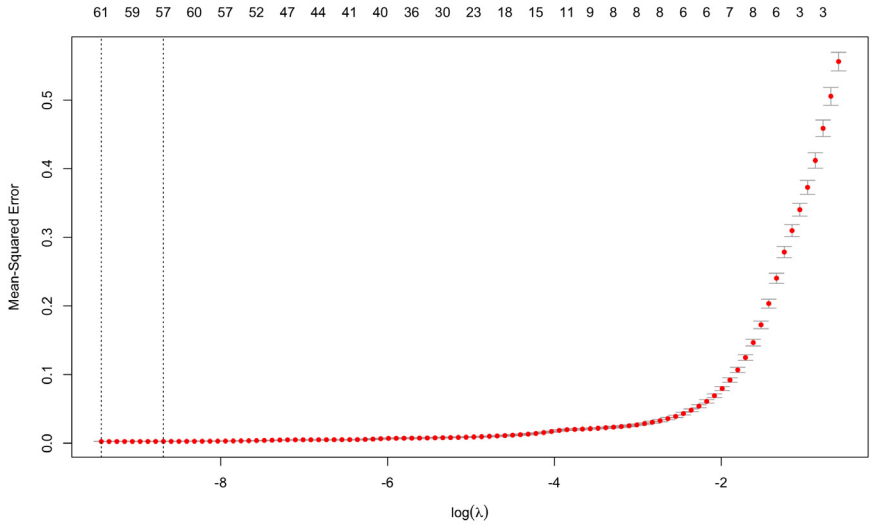

Figure 4.15: Graphical representation of the mean squared error calculated during the cross-validation for lasso as the value of $log(\lambda)$ increases.

is very low when the value of $log(\lambda)$ is low, and at a certain point, it starts rising quickly. The λ value minimizing the mean squared error and at a distance of $1\,SE$ have been compared. The second has been chosen since it allows more variable selection with a minimal increase of the mean squared error. Hence, λ is 0.0002, the mean squared error is 0.0022, and the root mean squared error is 0.0469.
Figure 4.16 summarizes the obtained results showing on the left a graphical representation of the trends of the variables' coefficients as the logarithm of λ increases and on the right the analysis of deviance according to the value of $log(\lambda)$. They both remark the chosen value of $log(\lambda)$ with a dotted line.
The graph of variables' coefficients shows that as the value of $log(\lambda)$ increases, the shrinkage becomes exponentially stronger, and variables go to zero. Moreover, differently from ridge regression, no variables weigh noticeably more than the others, but all the coefficients are small and belong to a short range $[-0.15, 0.10]$.
From the analysis of deviance, we can see that the explained deviance starts with a very high value, close to 1, then it decreases slowly till the value of $log(\lambda)$ is greater than 4, maintaining good values and finally drops down since too many variables

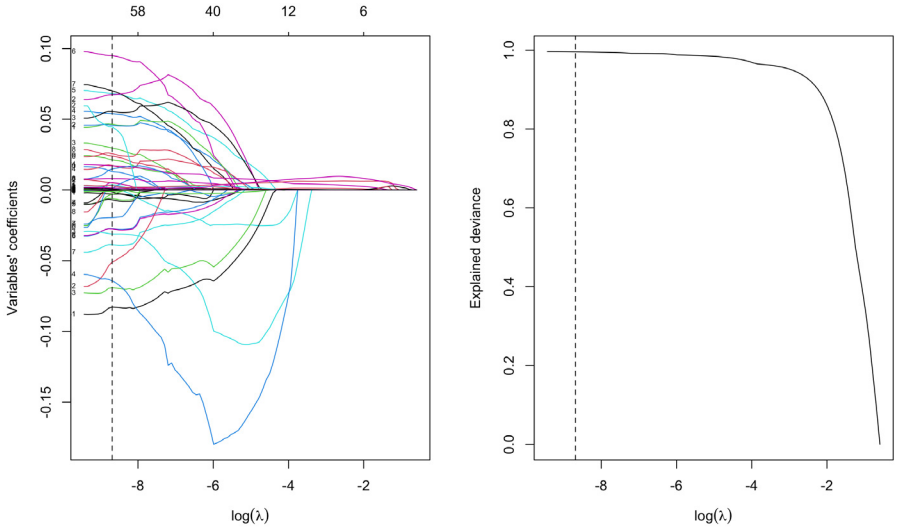

Figure 4.16: Left: trend of the variables' coefficients as $log(\lambda)$ increases in the lasso model. Each coloured line represents the trend of a coefficient, and the small numbers correspond to their order in the dataset. The number of variables that survived in the model is written at the top of the graph.
Right: deviance analysis of lasso according to the increasing value of $log(\lambda)$. Dotted line: the best value of λ.

are eliminated since the model has not enough information to explain the response anymore. However, the value in correspondence with the chosen λ is good.
Finally, the lasso coefficients $\hat{\beta}^{lasso}$ have been evaluated, substituting λ properly in (3.14). The explained deviance is 99.61%, which is remarkably better than ridge regression, considering that lasso produces a simpler model since variable selection has been performed. The selected variables, accompanied by their corresponding coefficients, are shown in Table 4.4.

Table 4.4: Estimate of the coefficients for the variables in the linear regression model using lasso. Standard error in parentheses.

Month2	Month3	Month4	Month5
-2.41e-03	7.797e-03	3.388e-02	5.735e-02
Month6	Month7	Month8	Month9
7.137e-02	9.929e-02	7.202e-02	2.872e-02
Month10	Month11	Month12	Hour7
2.713e-02	1.319e-02	-2.477e-04	6.871e-02
Hour8	Hour9	Hour10	Hour11
5.393e-02	2.198e-02	.	-2.049e-02
Hour12	Hour13	Hour14	Hour15
-3.103e-02	-2.032e-02	-3.971e-04	3.014e-02
Hour16	Hour17	Hour18	DHI
4.905e-02	4.915e-02	-3.104e-02	1.64e-02
DHI^2	DHI^3	DNI	DNI^2
-4.42e-05	4.275e-08	1.956e-03	-1.641e-06

DNI^3	Clearsky DHI	$Clearsky\ DHI^2$	$Clearsky\ DHI^3$
1.741e-09	.	.	4.975e-09
Clearsky DNI	$Clearsky\ DNI^2$	$Clearsky\ DNI^3$	Cloud type 2
.	3.84e-07	9.296e-11	5.063e-03
Cloud type 3	Cloud type 4	Cloud type 6	Cloud type 7
.	.	-8.288e-02	-6.107e-02
Cloud type 8	Cloud type 9	Dew point	SZA
-6.801e-02	-7.875e-02	1.122e-03	6.843e-03
Surf albedo 0.14	Surf albedo 0.15	Surf albedo 0.16	Surf albedo 0.17
.	.	-5.986e-03	1.695e-03
Wind speed	Prec water	$Prec\ water^2$	$Prec\ water^3$
.	4.747e-02	-1.971e-02	.
Wind direction	Rel humidity	$Rel\ humidity^2$	$Rel\ humidity^3$
4.36e-05	3.885e-04	.	-5.793e-08
Temperature	$Temperature^2$	$Temperature^3$	Pressure
1.65e-03	.	.	.
$Pressure^2$	$Pressure^3$	DNI - SZA	Clear DNI - SZA
.	.	-3.687e-06	.
Surf albedo - DNI	Surf albedo - Clearsky DNI	Surf albedo - DHI	Surf albedo - Clearsky DHI
6.123e-04	-7.248e-04	5.961e-03	.
Cloud type - DHI	Cloud type - DNI	Prec water - DHI	Prec water - DNI
4.464e-05	2.904e-06	.	1.139e-04

The coefficients' absolute values are generally in line with those of the linear model with polynomials in terms of the order of magnitude. The trend for polynomial terms is confirmed: the higher the degree, the lower the coefficient's absolute value. Lasso mainly shrinks to zero the polynomial terms with a high degree since they are probably needed to explain only a minority of observations, but some could be discarded. An inverted trend has been noted for Clearsky DHI and Clearsky DNI since they have been dropped when linear and considered when quadratic or cubic. The difference between solar irradiance coefficients and meteorological and atmospheric variables in terms of absolute values is not big, so they are almost equally important for the algorithm. Nevertheless, the weights assigned to months, hours, and cloud types corresponding to think clouds stand out. Moreover, Lasso has assigned less weight to the interactions, discarding three of them. If we look at the signs, there are no anomalies since they are in accordance with the previous analysis. In this case, despite the explained deviance being high, variable selection has not been effective, and only a few variables have been eliminated.

4.3.3 Elastic Net

Elastic-Net (Hastie, Tibshirani, and Friedman, 2011, Chapter 3) (Zou and Hastie, 2005) is a regularization method, which performs variable selection. It aims to find the best trade-off between ridge regression and lasso penalties by mixing both of them. It balances their weights according to a coefficient named α.
Since there are two hyperparameters to tune, α and λ, cross-validation has been executed. The value of the hyperparameters minimizing the mean squared error are α equal to 0.642 and λ equal to 0.0033. The mean squared error is 0.0056, and the root

mean squared error is 0.0748.
Figure 4.17 shows two graphs of the coefficients' variables and the explained deviance according to the λ value, fixing the best α obtained by cross-validation. The graphs

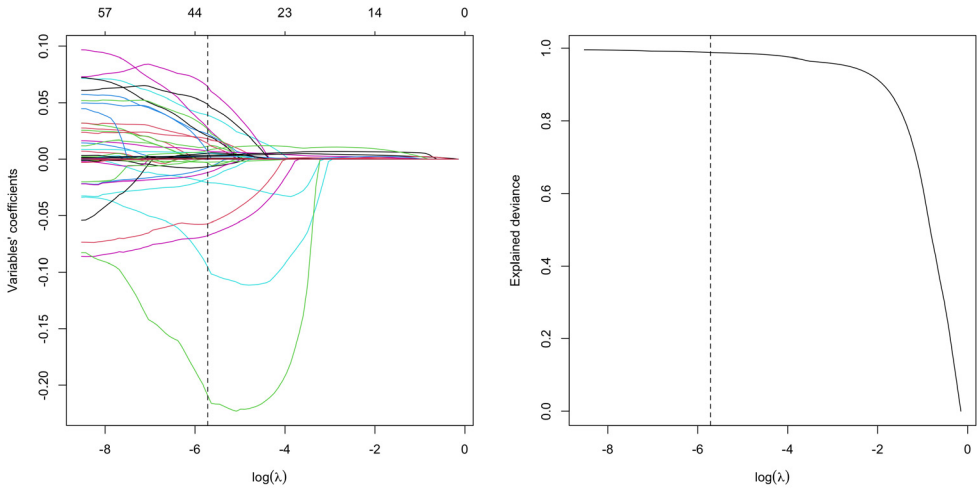

Figure 4.17: Left: trend of the variables' coefficients as $log(\lambda)$ increases in the Elastic Net model. Each coloured line represents the trend of a coefficient, and the small numbers correspond to their order in the dataset. The number of variables that survived in the model is written at the top of the graph. Right: deviance analysis of Elastic Net according to the increasing value of $log(\lambda)$. Dotted line: the best value of λ.

are much more similar to lasso than to ridge regression because Elastic does variables selection. However, the Elastic net coefficients are more spaced and belong to a wider range than lasso, but still closer than ridge regression. Moreover, the dotted line shows that heavy shrinkage has applied and many more variables are eliminated with respect to lasso, at a cost of a slightly lower explained deviance. Nevertheless, the explained deviance remains very good.
Finally, the Elastic net coefficient $\hat{\beta}^{Elastic\,net}$ has been computed according to the penalty factor (3.16), substituting the value of α and λ chosen through cross-validation. The explained deviance is 98.98%. This value is greater than the ridge regression, but a bit less than lasso, as expected. However, since Elastic net has provided a simpler model than lasso, it could be preferable according to the needs. The selected variables, accompanied by their coefficients, are shown in Table 4.5.

Table 4.5: Estimate of the coefficients for the variables in the linear regression model using Elastic Net. Standard error in parentheses.

Month2	Month3	Month4	Month5
-7.155e-03	.	.	2.133e-02
Month6	Month7	Month8	Month9
3.815e-02	2.912e-02	2.184e-02	.
Month10	Month11	Month12	Hour7
1.085e-02	.	-1.945e-02	6.063e-02
Hour8	Hour9	Hour10	Hour11

4.618e-02	1.673e-02	.	-7.088e-03
Hour12	Hour13	Hour14	Hour15
-1.696e-02	-1.146e-02	.	1.302e-02
Hour16	Hour17	Hour18	DHI
2.453e-02	5.177e-03	-9.357e-02	7.446e-03
DHI^2	DHI^3	DNI	DNI^2
-3.331e-06	-1.356e-08	1.014e-03	.
DNI^3	Clearsky DHI	Clearsky DHI^2	Clearsky DHI^3
6.636e-10	3.414e-04	.	.
Clearsky DNI	Clearsky DNI^2	Clearsky DNI^3	Cloud type 2
.	.	4.158e-10	.
Cloud type 3	Cloud type 4	Cloud type 6	Cloud type 7
.	.	-6.702e-02	.
Cloud type 8	Cloud type 9	Dew point	SZA
-5.581e-02	-1.079e-01	2.481e-03	5.254e-03
Surf albedo 0.14	Surf albedo 0.15	Surf albedo 0.16	Surf albedo 0.17
.	4.367e-03	-2.226e-03	-6.458e-03
Wind speed	Prec water	Prec $water^2$	Prec $water^3$
.	.	.	-2.6e-03
Wind direction	Rel humidity	Rel $humidity^2$	Rel $humidity^3$
7.862e-05	.	.	-4.264e-08
Temperature	$Temperature^2$	$Temperature^3$	Pressure
3.402e-03	.	.	.
$Pressure^2$	$Pressure^3$	DNI - SZA	Clear DNI - SZA
-6.715e-07	-1.458e-10	.	.
Surf albedo - DNI	Surf albedo - Clearsky DNI	Surf albedo - DHI	Surf albedo - Clearsky DHI
2.359e-03	.	9.93e-03	.
Cloud type - DHI	Cloud type - DNI	Prec water - DHI	Prec water - DNI
.	3.45e-06	.	1.322e-04

The coefficients' absolute values are similar to lasso but slightly lower. The trend of polynomial terms is also confirmed here: the higher the degree, the lower the coefficient's absolute value. Polynomials with a higher degree have not dropped anymore, differently from lasso. Instead, some linear terms, such as Clearsky DNI, Precipitable water, Relative humidity, and Pressure, are considered only with a degree greater than one. This trend is probably due to the more marked variable selection: if the model had dropped too many polynomial terms, the high variance would have been not considered. The difference between solar irradiance coefficients and meteorological and atmospheric variables in terms of absolute values is not big and heavy weights have been assigned to months, hours, and cloud types, as for lasso. Instead, there is a difference in the interactions: despite half of them being dropped, more weight has been given to the ones concerning DHI, DNI, and Surface albedo than the ones concerning meteorology. Elastic net regularization is effective on this problem since it has focused on much fewer variables than the initial dataset giving more specific information and maintaining high performance. Finally, analysis with Adaptive lasso has been accomplished to try to improve the lasso's variable selection.

4.3.4 Adaptive lasso

Adaptive lasso (Zou, 2006) is a regularization method which refines lasso. Adaptive lasso does not produce biased estimates since it satisfies the Oracle property (Fan and Li, 2001). It uses adaptive weights for penalizing different coefficients, and it is based on two hyperparameters, λ and γ, that must be tuned through cross-validation.
The configuration of the Adaptive lasso model has set α to 1 from the requirements and the weight penalty to $1/(|b|)$, where b represents the variables' coefficients of ridge regression that have been found previously. Ridge regression coefficients have been used as preprocessed weight because it is a best practice taken from the literature (Zou, 2006) that provides empirically good results. Therefore, there is only λ to tune, which controls the strength of the penalty term as for all the other regularization methods.
Figure 4.18 shows the result of cross-validation. The value of λ minimizing the mean

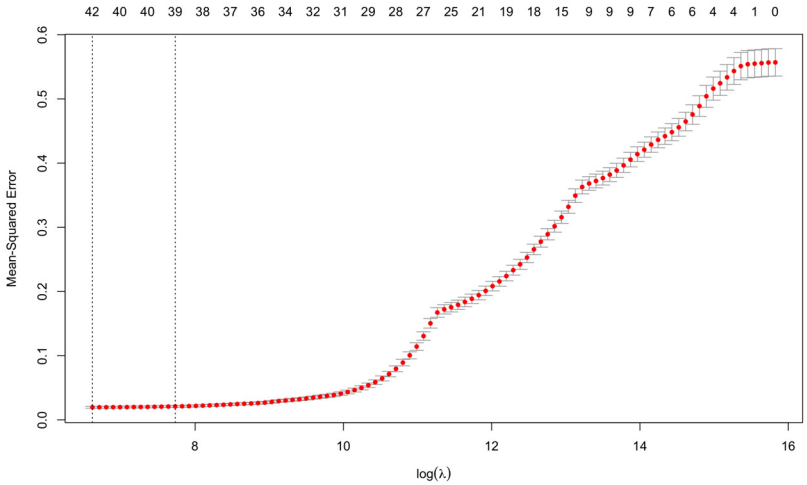

Figure 4.18: Graphical representation of the mean squared error calculated through Adaptive lasso as the value of $log(\lambda)$ increases during the cross-validation.

squared error is preferred since it already performs heavy variables selection. The mean squared error is 0.0196, the root mean squared error is 0.14, and the value of λ achieved is 746.46. This value is much greater than the other regularization methods meaning that the penalty strongly affects the result.
Figure 4.19 shows two graphs of the coefficients' variables and the explained deviance according to the λ value obtained from the cross-validation. Due to the adaptive lasso implementation, the graphs represent only the values of λ from the best on.
The graphs of variables' coefficients show that the values vary a lot as the value of λ increases. However, the range of the coefficients' values belongs to $[-0.5, 0.5]$ in correspondence to the chosen value of λ, so it is small, similarly to the previous methods.
From the analysis of deviance, we notice that the explained deviance in correspondence to the best λ seems relatively high and then drops down quickly.
Finally, the Adaptive lasso coefficients $\hat{\beta}^{Adaptive\,lasso}$ have been evaluated, substituting the proper λ and penalty term in (3.17). The explained deviance is 96.73%, which is not good with respect to the other methods. The model performs the heaviest variable selections among all the shrinkage methods, but the explained deviance is not

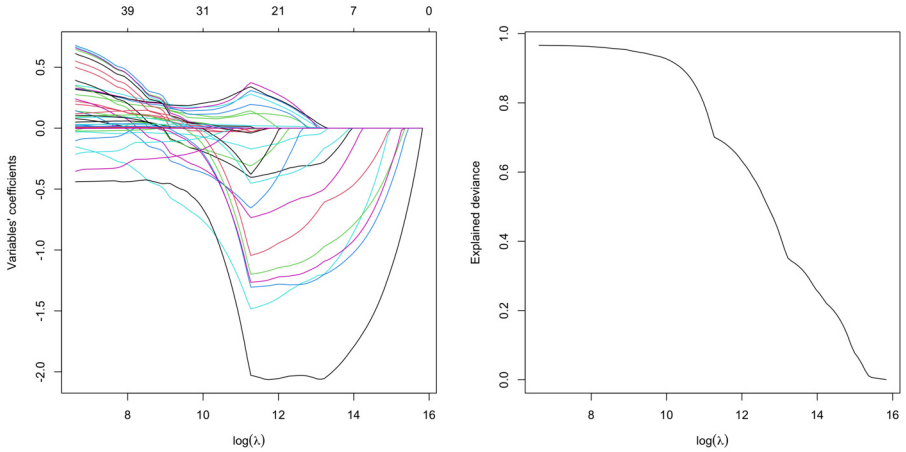

Figure 4.19: Left: trend of the variables' coefficients as $log(\lambda)$ increases in the Adaptive lasso model. Each coloured line represents the trend of a coefficient, and the small numbers correspond to their order in the dataset. The number of variables that survived in the model is written at the top of the graph.
Right: deviance analysis of Adaptive lasso according to the increasing value of $log(\lambda)$.
Dotted line: the best value of λ.

satisfactory, probably because too much selection has been performed. As evidence of this, the selected variables, accompanied by their coefficients, are reported in Table 4.6.

Table 4.6: Estimate of the coefficients for the variables in the linear regression model using Adaptive lasso. Standard error in parentheses.

Month2	Month3	Month4	Month5
0.0749	0.192	0.271	0.317
Month6	Month7	Month8	Month9
0.348	0.322	0.313	0.216
Month10	Month11	Month12	Hour7
0.132	0.026	-0.033	0.239
Hour8	Hour9	Hour10	Hour11
0.392	0.549	0.642	0.676
Hour12	Hour13	Hour14	Hour15
0.665	0.656	0.609	0.498
Hour16	Hour17	Hour18	DHI
0.347	0.143	-0.156	0.002
DHI^2	DHI^3	DNI	DNI^2
.	.	.	.
DNI^3	Clearsky DHI	Clearsky DHI^2	Clearsky DHI^3
.	.	.	.
Clearsky DNI	Clearsky DNI^2	Clearsky DNI^3	Cloud type 2
.	.	.	0.048
Cloud type 3	Cloud type 4	Cloud type 6	Cloud type 7

0.118	0.092	0.023	0.084
Cloud type 8	Cloud type 9	Dew point	SZA
0.007	-0.438	.	0.01
Surf albedo 0.14	Surf albedo 0.15	Surf albedo 0.16	Surf albedo 0.17
-0.012	-0.083	-0.175	-0.307
Wind speed	Prec water	Prec water^2	Prec water^3
.	0.102	-0.023	.
Wind direction	Rel humidity	Rel humidity^2	Rel humidity^3
.	.	-4.745e-06	.
Temperature	Temperature^2	Temperature^3	Pressure
0.005	.	.	.
Pressure^2	Pressure^3	DNI - SZA	Clear DNI - SZA
.	.	.	.
Surf albedo - DNI	Surf albedo - Clearsky DNI	Surf albedo - DHI	Surf albedo - Clearsky DHI
0.0145	.	0.017	-0.003
Cloud type - DHI	Cloud type - DNI	Prec water - DHI	Prec water - DNI
.	.	.	.

The coefficient's absolute values are generally at least an order of magnitude more than every other method analyzed. The selection has dropped too many variables, penalizing solar irradiance coefficients. Therefore, since the performance is worse than lasso and Elastic net, this method could not be preferable to the others.
However, we can identify an interesting fact: without part of the solar irradiance coefficients but maintaining their interactions and the majority of meteorological and atmospheric variables, the result is not dramatic. Therefore, we can say that even if solar irradiance coefficients are essential to explain the phenomenon studied in this thesis, predictors related to meteorology are relevant.

4.3.5 Predictions

Predictions with the four different regularization models have been performed. Since the observations are too many, the best representation for predictions is a graph with observed values on the abscissa axis, predicted values on the ordinate axis, and the bisector of the first and third squares, which corresponds to the perfect prediction of the observations.
The predictions provided by ridge regression are illustrated in Figure 4.20 only for completeness. In fact, the mean squared error is 0.0125, the root mean squared error is 0.1118, and the AdjR2 is 0.9803. The results are not so satisfactory since the method provides a net deterioration in both metrics with respect to linear regression. Some predictions are incorrect, especially with medium-high Global Horizontal UV Irradiance values.
The predictions provided by lasso are illustrated in Figure 4.21. The mean squared error is 0.0025, the root mean squared error is 0.05, and the AdjR2 is 0.9961. The results are very good since the metrics' values are comparable with linear regression with polynomial features or with natural splines.
The predictions provided by Elastic net are illustrated in Figure 4.22. The mean squared error is 0.0059, the root mean squared error is 0.0768, and the AdjR2 is 0.9884. The results are a bit worse than lasso. The predictions are slightly higher than the correct ones for high Global Horizontal UV Irradiance values. Nevertheless, despite

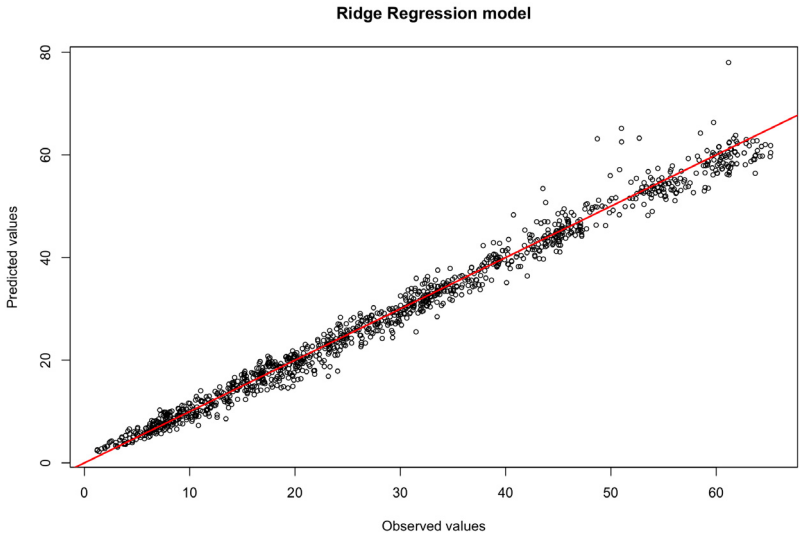

Figure 4.20: Predictions of Global Horizontal UV Irradiance of wavelength in the range of 280-400nm, performed by ridge regression.

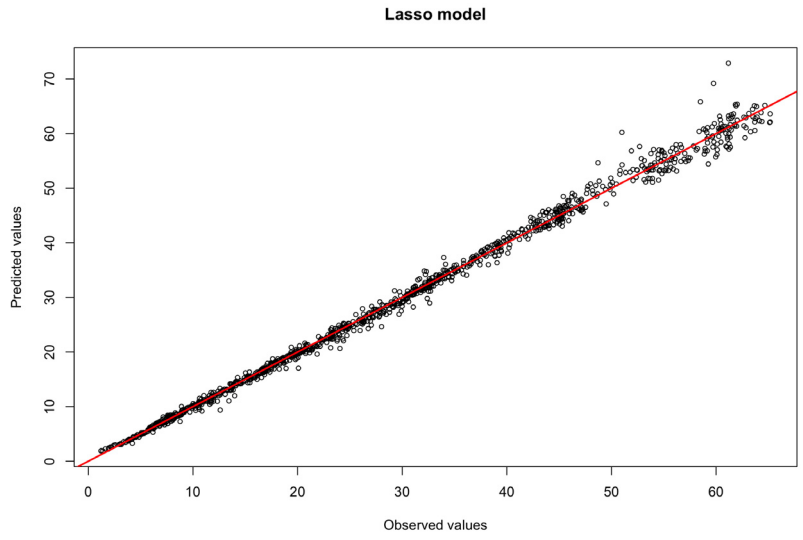

Figure 4.21: Predictions of Global Horizontal UV Irradiance of wavelength in the range of 280-400nm, performed by lasso.

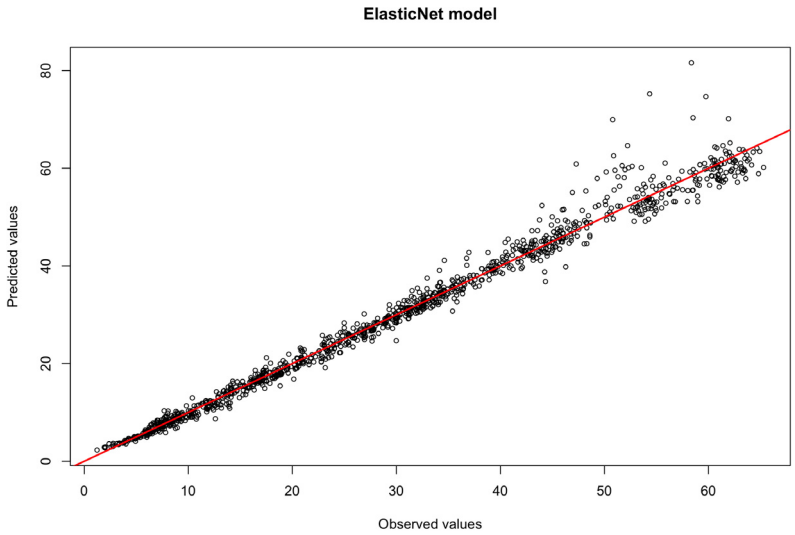

Figure 4.22: Predictions of Global Horizontal UV Irradiance of wavelength in the range of 280-400nm, performed by Elastic net.

heavy variable selection being applied, the prediction metrics are very good.
The predictions provided by Adaptive lasso are illustrated in Figure 4.23. The mean squared error is 0.02, the roor mean squared error is 0.1414 and the AdjR2 is 0.9651. The results are worse than the other regularization methods, and too many predictions are clearly wrong. This corresponds to our expectation since too many variables have been eliminated, even important ones.
The predictions of these four regularization methods do not reserve any surprise with respect to the analysis previously conducted. The trend of the linear regression models is confirmed: the higher the Global Horizontal UV Irradiance values, the worse the predictions.

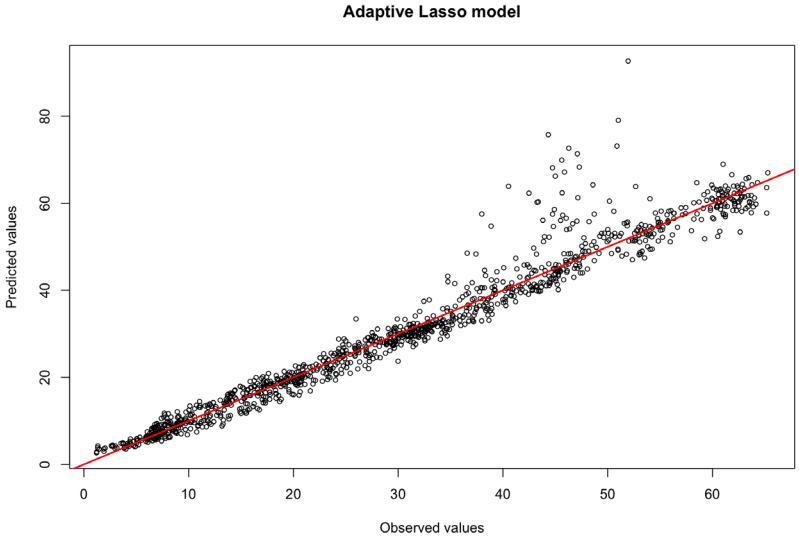

Figure 4.23: Predictions of Global Horizontal UV Irradiance of wavelength in the range of 280-400nm, performed by Adaptive lasso.

Chapter 5

Machine learning methods

This chapter focuses on the machine learning methods used to make predictions on the natural logarithm of the Global Horizontal UV Irradiance of wavelength in the range of 280-400nm and to compare the results with the data mining methods. The input features are reported in Figure 2.14. The machine learning methods considered are Decision tree regression, K-Nearest-Neighbor (KNN) regression, and Support Vector Regressor (SVR) (see SVR). Extensions of decision tree regression exploiting ensemble learning techniques, such as Random Forest and Gradient boosting (see XGBoost), will be considered. Further details are available in the documentation (Mitchell, 1997, Chapters 3-8) (Jaskes *et al.*, 2013, Chapters 13-15) (Hastie, Tibshirani, and Friedman, 2011, Chapter 2).

5.1 Premises

5.1.1 Bagging

Bagging (Breiman, 1996) is an ensemble learning technique aggregating weak learners, which are often homogeneous. It learns them independently through parallel computation, and it combines them according to an averaging algorithm. Its purpose is to build an effective and robust model, especially against high variance.
In practice, getting enough data to apply Bagging is infeasible. Thus, it relies on a technique named Bootstrap to build representative and almost independent samples. Bootstrap consists of generating a set of samples, each one by randomly drawing with replacements M observations from a dataset of size N. If the dataset size is large enough and $M < N$, then the samples are represented with a small correlation. Samples satisfying these two properties are fundamental for building good models to face data with high variance.
A graphical example summarizing the whole process of bagging is illustrated in Figure 5.1.

5.1.2 Boosting

Boosting (Freund and Schapire, 1996) is an ensemble learning technique aggregating weak learners, which are often homogeneous. It learns them sequentially, each one starting from a typically poor model, refining it, and finally combining all the weak learners according to a deterministic strategy. The refinement procedure can be

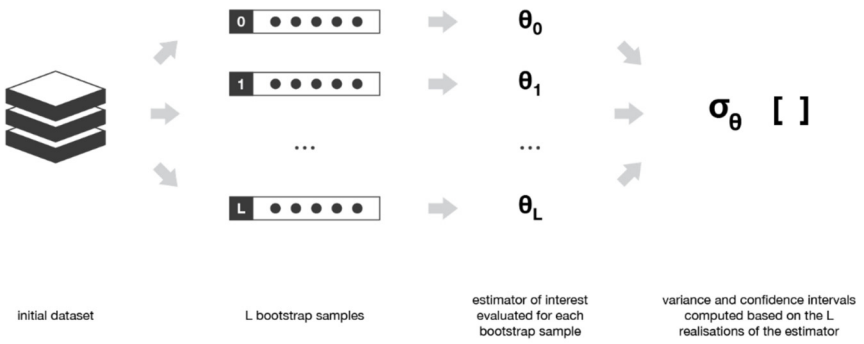

Figure 5.1: Graphical example of the whole process of bagging.
Source: `https://towardsdatascience.com/ensemble-methods-bagging-boosting-and-stacking-c9214a10a205`.
The process starts from the initial data, generates L bootstrap samples, each one of size M, builds an estimator for each bootstrap sample according to the machine learning method chosen, and aggregates the estimators into a unique one.

summarized with the steps listed below.

- Initialize the prediction value to zero and the residual to the true value y.
- Fit the model and make a prediction.
- Update the previous prediction by adding the new one multiplied by a parameter λ that controls the updating quickness and subtracting the same term to the residual.
- Repeat the previous two steps till an exit condition is satisfied.
- Aggregate the models with a deterministic policy, such as the average, and obtain the final output.

A graphical example summarizing the whole process of boosting is illustrated in the Figure 5.2.
The focus of boosting is to build an effective model against high bias instead of high variance. In fact, learning slowly, starting from a poor model and improving it step by step, allows predictors to learn from past mistakes, reducing this bias. Another advantage is the resilience against overfitting.
The disadvantages of boosting are the high sensitivity to outliers and the computational time needed to train the entire model, since each predictor depends on the previous one. This makes the process difficult to parallelize.

5.1.3 Gradient descent

Gradient descent (Goodfellow, Bengio, and Courville, 2016, Chapters 2-5) is a mathematical method for calculating the function's minimum. It is based on the gradient, which tells how to change the parameters in order to improve the function minimum. The positive gradient points uphill, while the negative points downhill, so minimization

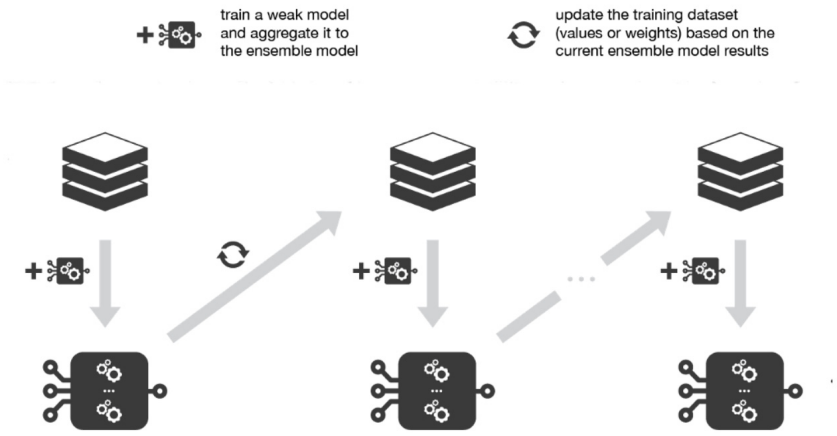

Figure 5.2: Graphical example of the whole process of boosting.
Source: `https://towardsdatascience.com/ensemble-methods-bagging-boosting-and-stacking-c9214a10a205`.
The process starts from an initial weak model and trains and aggregates it to the ensemble model updating the training dataset according to the obtained results. The process is iterated till the model satisfies some requirements identified by metrics.

requires following the negative gradient.
The procedure initially chooses a starting point x belonging to the domain of the function f to minimize. The starting point selection is crucial for the minimization, and the basic version makes this choice randomly. Further improvements of gradient descent consider a stochastic selection of many starting points, running the method for each of them and taking only the best final result. After the starting point has been selected, the algorithm iterates till the minimum has been found. The gradient descent updating rule applied at each iteration for evaluating a new point is

$$x' = x - \epsilon \nabla_x f(x), \tag{5.1}$$

where x is the point given as the input, f is the function to minimize, ∇_x is the gradient of the function f computed with respect to x, and ϵ is the learning rate, a positive coefficient determining the size of a step. In a machine learning context, it should be chosen by cross-validation.
The main drawbacks of gradient descent are listed below.

- When the function derivative is zero, no information about where to move is provided, so the procedure is stuck. Such a point is named a critical or stationary point.
- When the function reaches a local minimum, the procedure could return it as a solution, albeit it is not optimum.
- If the function reaches a saddlepoint that is neither a local minimum nor a local maximum, the procedure could get stuck.

- If the dataset is huge, the method becomes infeasible.

In general, function shapes are complex, and those problems arise frequently. Many techniques were developed to make improvements, such as stochastic gradient descent, which works on a subset drawn uniformly from the dataset.
Further refinements use the second derivatives giving more information to compute the updating step on the function, such as the curvature. However, these methods increase the model complexity.

5.1.4 Minkowski distance

Minkowski distance (see Minkowski distance) is a common measure to evaluate the distance between two points in an n-dimensional space. The Minkowski distance of order p between two points $S = (s_1, ..., s_n)$ and $T = (t_1, ..., t_n)$ is defined as

$$d(S,T)_{Minkowski} = \left(\sum_{i=1}^{n} |s_i - t_i|^p \right)^{1/p}. \tag{5.2}$$

For $p \geq 1$ the Minkowski distance satisfies the triangle inequality, so it is considered as a metric.
There are three special cases according to some values of p:

- if $p = 1$ the formulation is called Manhattan distance;
- if $p = 2$ the formulation is called Euclidean distance;
- if $p \to \infty$ the formulation is called Čebyšëv distance.

The Minkowski distance is also used as a notion of similarity between points in a n-dimensional space, e.g., in the K-Nearest-Neighbour algorithm.

5.1.5 Kernel

In machine learning, a kernel (see Kernel functions) is a function adopted to effectively and efficiently solve non-linear problems. It is applied on every instance of a dataset to map the original non-linear observations into a higher-dimensional space, where they become linearly separable. This operation aims to enlarge the feature space so that a hyperplane can separate the data. This procedure refers to as the kernel trick.
The kernel is defined as

$$k(x,y) = \langle f(x), f(y) \rangle, \tag{5.3}$$

where $k(x,y)$ is the kernel function, x, y are the n-dimensional inputs, f is the function mapping the points from a n-dimensional to a m-dimensional space, with $m > n$, and $\langle \cdot \rangle$ denotes the inner product. Therefore, instead of computing the map function twice and the inner product, a kernel is computed obtaining the same result at a lower computational cost.
The most used kernels are listed below.

- Linear kernel: $k(x,y) = \langle x, y \rangle$.
- Polynomial kernel: $k(x,y) = \left(1 + \sum_{i=1}^{p} x_i y_i \right)^d$.

- Sigmoid kernel: $k(x, y) = \tanh\Big\{\gamma\Big(\sum_{i=1}^{p} x_i y_i\Big) + c\Big\}$.
- Gaussian kernel: $k(x, y) = \exp\Big(-\frac{\|x-y\|^2}{2\sigma^2}\Big)$.
- Gaussian Radial Basis kernel (RBF): $k(x, y) = \exp\big(-\gamma\|x - y\|^2\big)$.

5.2 Decision tree regression

Decision tree (Mitchell, 1997, Chapter 3) is one of the most used machine learning algorithms. It is based on a tree-like decision model, where a set of rules is applied at each node splitting the prediction space into many regions with a top-down approach. The leaves contain the regression outcomes. However, the algorithm chooses only one leaf after every execution terminates, according to the previous decisions.
A top-down, greedy approach, named recursive binary splitting, is used to build the regression tree. It starts from the root of the tree and iteratively splits the predictor space making the best decision according to the rule assigned to the current node. The greedy choice consists of not looking ahead, assuming that no splitting in the future would have led to a better tree. The procedure has to stop when all the current leaves have fewer than some minimum number of observations that would make any other splits infeasible.
After the tree construction, the cost complexity pruning procedure is run since it would have been too complex to manage. It aims to find a sequence of subtrees indexed by a tuning parameter $\alpha > 0$, and cross-validation is applied to select the one minimizing the test error. Therefore, each value of α corresponds to a subtree. The function to be minimized is

$$\sum_{m=1}^{|T|} \sum_{i:\, x_i \in R_m} (y_i - \hat{y}_{R_m})^2 + \alpha|T|, \tag{5.4}$$

where $|T|$ is the number of leaves in the tree T, R_m is the subset of the predictor space corresponding to the mth leaf, and $\hat{y}_{R_m}$ is the prediction associated with R_m, which is the mean of the training observations in R_m. Parameter α controls the trade-off between the subtree's fit on the training data and the complexity. The more α increases, the more the penalty on having many terminal nodes, so the subtree results smaller.
The prediction is very fast and consists of passing through the tree with a top-down approach and making the decisions according to the observation features.
This model is simple and interpretable when the tree is not too complex, but it is not competitive with the best machine learning approaches. Further improvements, such as sampling on the training set when a split has to be decided, have been developed. Moreover, thanks to ensemble learning techniques combining trees, very effective algorithms were developed, such as Random forest and Gradient boosting.

5.2.1 Random forest

Random forest (Hastie, Tibshirani, and Friedman, 2011, Chapter 15) is an ensemble learning method based on bagging applied to many decision tree learners. It aims to cut down the variance, which is one of the main problems of decision tree. The less the correlation among learners, the more robust the final model to high variance.

It starts drawing a set of samples using bootstrapping on the training set and builds a decision tree simple model on every sample. When it builds the decision trees, each time a split is needed, a further random sample of m predictors is chosen as split candidates instead of the whole set of predictors p. By forcing this strategy, the procedure can decorrelate trees. In fact, if it had not been applied and there had been a strong predictor against the others, it would have been chosen by all the trees, making them highly correlated. However, drawing samples on predictors, makes less probable to find the strong predictor in most of the trees, so they become less correlated.

5.2.2 Extreme Gradient boosting

Gradient boosting (see XGBoost) is an ensemble learning method based on boosting applied to many weak decision trees, but it introduces some improvements. It aims to cut down the bias error learning sequentially and slowly.
Decision trees are fitted separately in each training round, and predictions are computed. A loss function, such as the minimum squared error, is used to calculate the prediction error with respect to the ground truth. Gradient boosting uses the gradient descent algorithm to minimize the loss function, so it finds the direction to change the model parameters reducing the error in the next round. Moreover, a stochastic version of Gradient boosting is considered, in which learners are trained on a randomly drawn subset of the training set to obtain a more generalized model. Extreme Gradient boosting (see XGBoost) is an improvement of the Stochastic Gradient boosting to obtain an even more accurate result. The two main features introduced by this method are listed below.

- The computation of the second-order gradients, i.e., the second partial derivatives of the loss function, similarly to Newton's method, providing more information about the gradient direction to minimize the loss function more effectively.
- The addition of L_1 and L_2 regularizations terms improving model generalization.

5.3 K-Nearest-Neighbour regression

K-Nearest-Neighbour (Mitchell, 1997, Chapter 8) (Hastie, Tibshirani, and Friedman, 2011, Chapter 13) is an instance-based learning method to predict discrete-valued or real-values objective functions. The main feature with respect to other machine learning methods is that the learning phase consists just of storing the training data, while the prediction consists of applying an algorithm.
K-Nearest-Neighbour assumes all instances correspond to points into an n-dimensional space and the training set made by couples of the form $\langle x, f(x)\rangle$.
The regression algorithm steps are listed below.

- Pick a new instance x_q of the test set whose value has to be predicted.
- Let $x_1, ..., x_k$ be the k instances from the training set that are nearest to x_q according to a distance metric.
- Return $\hat{f}(x_q) = \dfrac{\sum_{i=1}^{k} w_i f(x_i)}{\sum_{i=1}^{k} w_i}$,
  where $\hat{f}(x_q)$ is the predicted value and $w_i = \frac{1}{d(x_q, x_i)^2}$ is the weight assigned to

each neighbour computed as the inverse square of the distance function d. The nearer the instance, the more the weight.

Because of the n-dimensional space assumption on the data, Minkowski distance is the most used metric to choose the nearest instances to x_q. The order of the Minkowski distance could be chosen through cross validation.
Another parameter of the algorithm is the number of neighbours k. Since weights are included, the algorithm could consider all the instances to provide better estimates. However, according to the training set size N, it could become infeasible. Cross-validation could be performed to find the best value for k.
K-Nearest-Neighbour is a quite simple learning method. A disadvantage is that the computational cost for predicting new instances could become high due to the absence of training computation since all the effort is postponed at prediction time. Some techniques allow to efficiently index the data stored in the training phase, such as Ball Tree and Kd-tree based on a tree structure.
Ball Tree algorithm implements clustering according to a distance metric where each cluster has a hyperspherical form and corresponds to a tree node. Kd-tree stores the instances at the leaves of a tree, where nearby ones are stored at the same or nearby nodes and uses the non-leaves nodes as a sequence of attributes tests starting at the root to address new instances to their most relevant leaf.

5.4 Support Vector Regressor

Support Vector Regressor (see SVR) is a machine learning method to predict real-values objective functions. The idea behind it is to consider the observations in the dataset as points in a n-dimensional space and to define a hyperplane as a function passing through the points with at most ϵ-deviation from the response variable y. The ϵ-deviation is called soft margin, and it implies that the algorithm does not care about prediction errors as long as they are less than ϵ. As a result, the smaller the ϵ, the smaller the bias, and the larger the variance.
Support Vector Regressor can be stated as a constrained optimization problem. Hence, the same problem can be defined according to its definition, namely primal. Moreover, inverting the problem from min to max (or max to min), we can state another equivalent definition, namely dual. In case of linearly separable data, its primal definition is

$$\min \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{m}(\xi_i + \xi_i^*)$$
$$\text{subject to} \begin{cases} y_i - (\mathbf{w}\cdot\mathbf{x}_i) - b \leq \epsilon + \xi_i \\ (\mathbf{w}\cdot\mathbf{x}_i) + b - y_i \leq \epsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0, i = 1, ..., m \end{cases},$$

where $y_i = \mathbf{w}\mathbf{x}_i + b$ is the regression line formula considering the weights vector $\mathbf{w}$ associated with each feature of the variable x_i and the intercept b, and C is a coefficient controlling the soft margin trade-off, that is how much the errors above or under the line, ξ_i, ξ_i^* respectively, have to weight on the regression line formulation. The more the soft margin, the higher the chance to fit the model, but the lower the accuracy.
In the case of non-linearly separable data, kernel functions are used to enlarge the feature space. To explicit that, the switch to the dual formulation of the optimization

problem is required. The dual formulation is

$$\max \begin{cases} \frac{1}{2} \sum_{i,j=1}^{m} (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)\langle x_i, x_j \rangle \\ -\epsilon \sum_{i=1}^{m} (\alpha_i + \alpha_i^*) + \sum_{i=1}^{m} y_i(\alpha_i - \alpha_i^*) \end{cases}$$
$$\text{subject to} \begin{cases} \sum_{i=1}^{m} (\alpha_i - \alpha_i^*) = 0 \\ 0 \leq \alpha_i, \alpha_i^* \leq C \end{cases},$$

where α_i, α_i^* are the dual variables for each data point. If $\alpha = 0$, then the point is located within the margin; otherwise is located out of it.
A kernel function is substituted to the inner product $\langle x_i, x_j \rangle$ in the formulation above, causing the feature space to be not linear anymore. Thanks to the kernel functions, SVR can fit very complex models with non-linearly separable data efficiently and effectively.

Chapter 6

Machine learning analysis

This chapter focuses on the data analysis performed using the machine learning methods described in Chapter 5, in order to make predictions on the natural logarithm of the Global Horizontal UV Irradiance of wavelength in the range of 280-400nm with the input features reported in Figure 2.14.

6.1 Premises

Machine learning methods do not allow an interpretative analysis as the data mining methods. The focus will be on predictions. The predictive analysis will be conducted by splitting the dataset into train and test sets of 70% and 30%, respectively, as for the data mining methods.
Not all the preprocessing steps described in Chapter 2 are implemented for the machine learning methods. The representation of variables as factors and the addition of the interaction terms are implemented only in specific cases depending on the methods' requirements.
Since the observations are too many, as for the data mining methods, the representation with observed values on the abscissa axis, predicted values on the ordinate axis, and the bisector of the first and the third squares corresponding to the perfect predictions has been chosen.

6.2 Decision tree regression

Decision tree (Mitchell, 1997, Chapter 3) is one of the most used machine learning algorithms. Despite being quite a simple model, it gives fairly good results.
The preprocessing step implemented for building the decision tree model is to make the variable Cloud type a factor since its values correspond to written explanations and not real numbers. The hyperparameters `max_depth`, which is the maximum depth achievable from the tree, `min_sample_split`, which is the minimum number of samples to split an internal code, and `min_sample_leaf`, which is the minimum number of samples to be at a leaf node, must be tuned. Cross-validation has been performed to find the best hyperparameters' values minimizing the mean squared error. In the best configuration, `max_depth` is set to 13, `min_sample_split` is set to 4, and `min_sample_leaf` is set to 5. The evaluated metrics are the mean squared error equal to 0.0068, the root mean squared error equal to 0.0825, and Adjr2 equal to 0.9858.

The result is good but at a price of a high tree depth. Such a tree is infeasible to interpret by looking at the decisions taken along the paths. Trials with shallower trees have been implemented, but poor results have been found.
Predictions have been performed, and good results have been obtained according to the method complexity, as illustrated in Figure 6.1. The mean squared error is 0.0075,

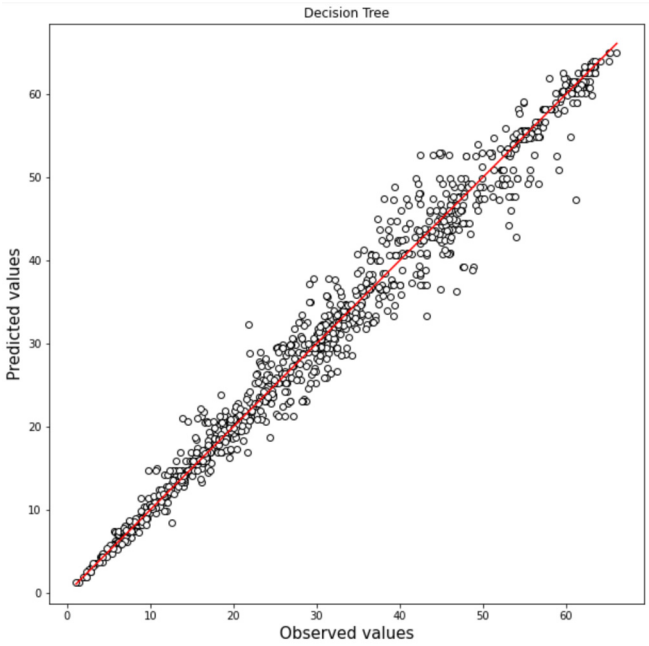

Figure 6.1: Predictions of Global Horizontal UV Irradiance of wavelength in the range 280-400nm, performed by the decision tree model with sampling at the split node.

the root mean squared error is 0.0866, and Adjr2 is 0.9846. Looking at the graph, although low and high values of Global Horizontal UV Irradiance are well predicted, many predictions with values in the middle have visible errors. Since the decision tree is a basic model, we now expect that ensemble of decision trees will perform better.

6.2.1 Random forest

Random forest (Hastie, Tibshirani, and Friedman, 2011, Chapter 15) is an ensemble learning method based on bagging applied to many decision tree learners.
No further preprocessing steps have been implemented for the Random forest model. Some trials have been implemented, but the performances have been worse. The hyperparameters `n_estimators`, which is the number of trees to consider in the forest, `max_depth`, which is the maximum depth achievable from each tree, `min_sample_split`, which is the minimum number of samples to split an internal code for each tree, and `min_sample_leaf`, which is the minimum number of samples to be at a leaf node for each tree, must be tuned. Cross-validation has been performed to find the best hyperparameters' values minimizing the mean squared error. In the best configuration, `max_depth` is set to 15, `min_sample_split` is set to 2, and `min_sample_leaf` is set to 1. The evaluated metrics are the mean squared error equal to 0.0022, the root

mean squared error equal to 0.047, and Adjr2 equal to 0.9952. There is a noticeable improvement in terms of Adjr2 with respect to the basic decision tree model, as expected.
Predictions have been performed, and very good results have been obtained, as illustrated in Figure 6.2. The mean squared error is 0.0025, the root mean squared

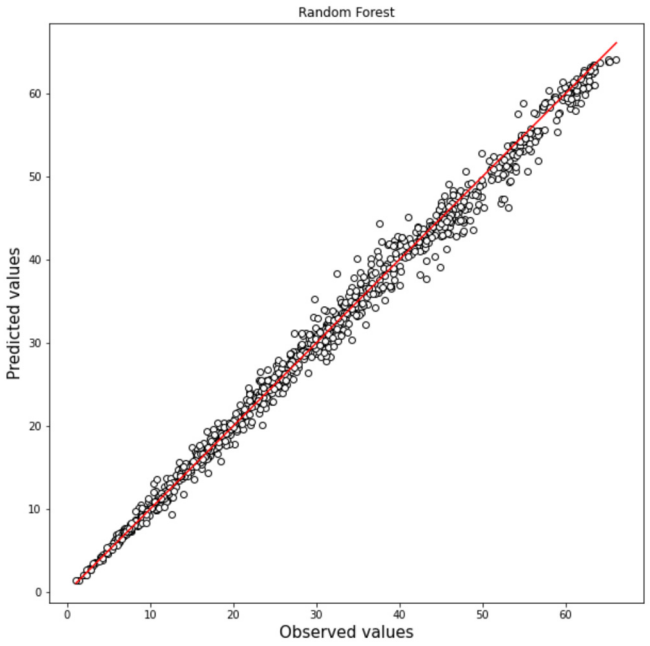

Figure 6.2: Predictions of Global Horizontal UV Irradiance of wavelength in the range 280-400nm, performed by the Random forest model.

error is 0.499, and Adjr2 is 0.995. These values are preferable to those from the decision tree model. Looking at the graph in Figure 6.2, the points are almost all closer to the bisector, indicating a good performance of the method in terms of prediction.

6.2.2 Extreme Gradient boosting

Extreme Gradient boosting (see XGBoost) is an ensemble learning method based on boosting applied to many decision tree learners.
The preprocessing step implemented for building the Extreme Gradient boosting model is to consider cloud type as a factor. It is useful probably for the same reason as the decision tree. The hyperparameters `n_estimators`, which is the number of trees to consider in the ensemble, `eta`, which is the shrinkage applied at each step size during the updating to prevent overfitting, `max_depth`, which is the maximum depth achievable from each tree, and `subsample`, which is the subsample ratio of the training instances occurring once in every boosting iteration to prevent overfitting, must be tuned. Cross-validation has been performed to find the best hyperparameters' values minimizing the mean squared error. In the best configuration, `n_estimators` is set to 1500, `eta` is set to 0.01, `max_depth` is set to 10, and `subsample` is set to 0.7. The evaluated metrics are the mean squared error equal to 0.0015, the root mean squared error equal to 0.0387, and Adjr2 equal to 0.9972. There is a remarkable improvement

in performance with respect to the basic decision tree, as expected.
Predictions have been performed, and excellent results have been obtained, as illustrated in Figure 6.3. The mean squared error is 0.0016, the root mean squared error is 0.0398,

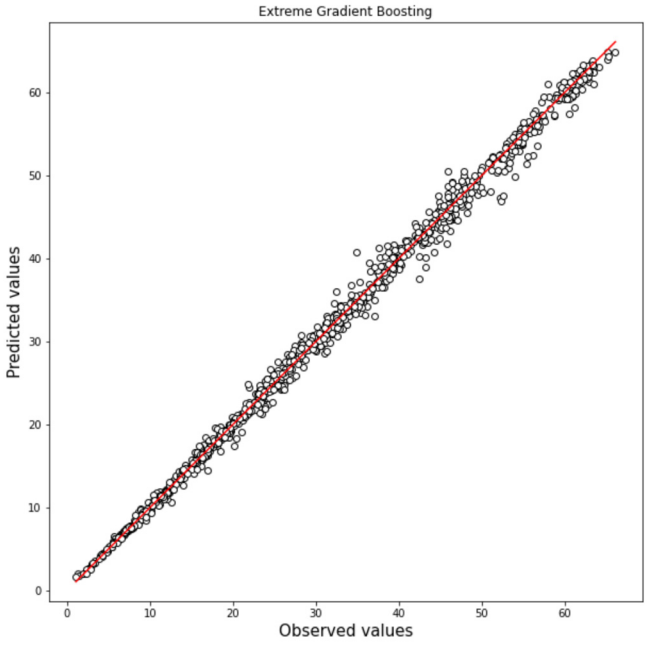

Figure 6.3: Predictions of Global Horizontal UV Irradiance of wavelength in the range 280-400nm, performed by the Extreme Gradient boosting model.

and Adjr2 is 0.997. These values are much better than the those from the decision tree model and the Random forest model. By looking at the graph, only very few predictions corresponding to the middle-high values of Global Horizontal UV Irradiance are detached from the red line presenting a visible error.

6.3 K-Nearest-Neighbour regression

K-Nearest-Neighbour (Mitchell, 1997, Chapter 8) (Hastie, Tibshirani, and Friedman, 2011, Chapter 15) is an instance-based learning method to predict discrete-valued or real-values objective functions. Despite being a simple method, a chance has been given to it since other basic methods, such as the decision tree, have performed well. Two preprocessing steps have been implemented: the addition of the interactions terms described in Section 2.4 since from the trials they have provided significant improvements and the scaling of the data, which is required for K-Nearest-Neighbour. The hyperparameters `p`, which is the power parameter for the Minkowski distance, `n_neighbours`, which is the number of neighbours the algorithm has to consider for each centre, must be tuned. Cross-validation has been performed to find the best hyperparameters' values minimizing the mean squared error. In the best configuration, `p` is 1, and `n_neighbours` is 10. The evaluated metrics are the mean squared error equal to 0.0146, the root mean squared error equal to 0.1208, and Adjr2 equal to 0.9717. The result seems good, but it is not competitive with respect to the previous

machine learning algorithms.
Predictions have been performed, but non satisfactory results have been obtained, as illustrated in Figure 6.4. The mean squared error is 0.0161, the root mean squared

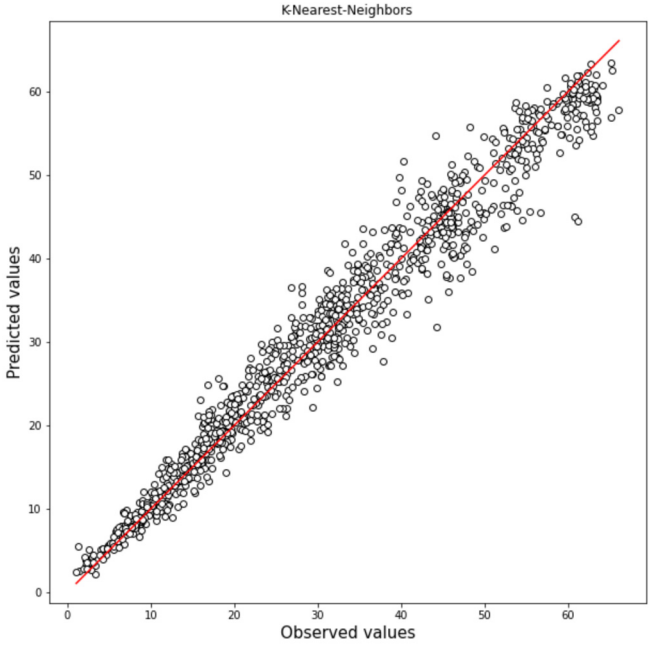

Figure 6.4: Predictions of Global Horizontal UV Irradiance of wavelength in the range 280-400nm, performed by the K-Nearest-Neighbour model.

error is 0.1269, and Adjr2 is 0.967. These values are worse than those determined by the other machine learning methods. The points in Figure 6.4 are too variable around the bisector, so there are noticeable errors throughout almost all the spectrum of Global Horizontal UV Irradiance values.

6.4 Support vector regressor

Support Vector Regressor (SVR) (see SVR) is a machine learning method to predict real-values objective functions.
The preprocessing steps implemented for this method are the same of the K-Nearest-Neighbour method: the addition of the interactions terms described in Section 2.4 and the scaling of the data, which is required also for SVR. The hyperparameters `kernel`, which is the kernel type used in the algorithm, `tol`, which is the tolerance used as a stopping criterion, `C`, which regulates the strength of the L_2 penalty, `epsilon`, which is the ϵ-deviation corresponding to the soft-margin explained in Section 5.4, and `max_iter`, which is the maximum number of iterations reachable from the algorithm, must be tuned. Cross-validation has been performed to find the best hyperparameters' values minimizing the mean squared error. In the best configuration, `kernel` is set to rbf, `tol` is set to 1e-3, `C` is set to 20, `epsilon` is set to 1e-5, and `max_iter` is set to 1000000. The evaluated metrics are the mean squared error equal to 0.022, the root mean squared error equal to 0.0469, and Adjr2 equal to 0.996. The result is very good,

comparable with the ensemble methods of decision tree learners.
The predictions are satisfactory, as well as illustrated in Figure 6.5.

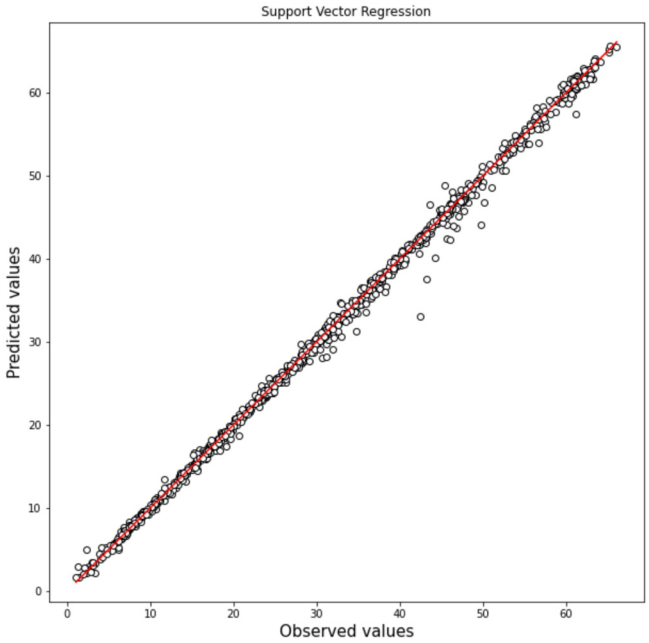

Figure 6.5: Predictions of Global Horizontal UV Irradiance of wavelength in the range 280-400nm, performed by the Support Vector Regressor model.

Chapter 7

Deepening the data analysis

This chapter focuses on deepening data analysis by applying data mining techniques described in Chapter 3 and machine learning methods described in Chapter 5 to identify further details, improve data interpretation, and acquire additional information about the scientific problem.

7.1 Premises

Linear regression assuming Skew-Normal distribution and Principal Component Regression (PCR), and trials with machine learning methods predicting Global Horizontal UV Irradiance of wavelength in the range 280-400nm on new dataset will be implemented as deepening data analysis.
The interpretative analysis will be accompanied by technical considerations from the literature and graphical representations. The predictive analysis will be conducted by splitting the dataset into train and test sets of sizes 70% and 30%, respectively.
A common baseline for the analysis with data mining methods is built by applying the preprocessing steps described in Chapter 2. Further specific preprocessing steps will be described in the next sections according to the features of the methods.

7.2 Linear regression with Skew-Normal distribution

The preliminary graphical analysis of the data, see Figure 2.5, supported that the response variable distribution cannot be assumed to be Normal. As a first approach, the natural logarithm of the response has been computed. The resulting distribution has become much more similar to a normal, as illustrated in Figure 2.6. However, the left tail is much longer than the right, supporting that a distribution accounting for some skewness could be more appropriate.
Linear regression assuming Skew-Normal distribution (Azzalini, 1985) for the response variable has been implemented to try to fit the distribution's trend better than the previous linear regression analysis based on the traditional normality assumption.
The analysis has started with a dataset considering the same polynomial terms introduced for the linear regression based on the normal distribution, namely, DHI, DNI, Clear sky DHI, Clear sky DNI, Precipitable water, Pressure, Relative humidity, and Temperature. Variable selection has been implemented based on the p-value associated with the significance of the variables, with a classical 0.05 threshold. As a result, only

the squared term of Clear sky DHI and the interaction between Surface albedo and DNI have been dropped.
The model evaluation suggests that the skewness component is significant and equal to -1.122. Negative skewness means that the distribution is a bit unbalanced towards the right. Figure 7.1 shows a graphical representation of the shape of the Skew-Normal distribution that has been found, assuming standardization. We notice that the skew-

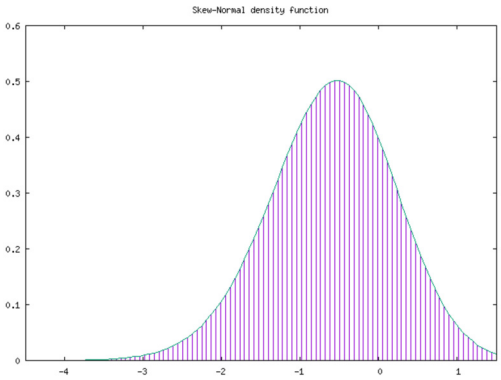

Figure 7.1: Graphical representation of the Skew-Normal distribution found through the analysis of linear regression.

ness is not accentuated, but it is visible.
The residual analysis looks very good, as illustrated in Figure 7.2. The "Residual vs Fitted" graph in the top-left panel of Figure 7.2 shows a uniform cloud without a noticeable deterministic pattern, albeit remaining a little trend on the left of the graph. The result is more satisfactory than linear regression with polynomials assuming a normal distribution, and it is comparable to the regression analysis using natural splines. The "Residual values and fitted error distribution" plot in the top-right panel of Figure 7.2 shows that the residuals are concentrated in the range $[-0.2, 0.1]$ meaning they are very small. The "Quantile-Quantile of scaled squared residuals" plot in the bottom-left panel of Figure 7.2 suggests a satisfactory correspondence between the empirical and theoretical quantiles correspond. Few points deviate from the dotted line, which probably are observations captured during anomalous atmospheric and meteorological conditions (see Los Angeles weather archive). The "Probability-Probability of scaled squared residuals" plot in the bottom-right panel of Figure 7.2 is a modified version of the "Quantile-Quantile of scaled squared residuals" plot focused on the correspondence between empirical and theoretical probabilities. It confirms a satisfactory behaviour of the model. The selected variables, accompanied by their coefficients and the associated standard errors surrounded by round brackets, are shown in Table 7.1.

Table 7.1: Estimate of the coefficients for the variables in the linear regression model assuming the Skew-Normal distribution on the response variable. Standard error in parentheses.

Month2	Month3	Month4	Month5
-9.74e-03 (4.34e-03)	-1.64e-03 (6.07e-03)	1.11e-02 (7.88e-03)	3.02e-02 (9.14e-03)
Month6	Month7	Month8	Month9
3.91e-02 (9.7e-03)	6.59e-02 (9.44e-03)	5.29e-02 (8.63e-03)	1.51e-02 (7.07e-03)

Month10 1.18e-02 (5.6e-03)	Month11 3.83e-03 (4.17e-03)	Month12 -1.01e-03 (4.36e-03)	Hour7 4.92e-02 (7.24e-03)
Hour8 3.19e-02 (1.04e-02)	Hour9 -2.48e-03 (1.37e-02)	Hour10 -3.32e-02 (1.64e-02)	Hour11 -5.96e-02 (1.82e-02)
Hour12 -6.94e-02 (1.88e-02)	Hour13 -5.51e-02 (1.79e-02)	Hour14 -3e-02 (1.59e-02)	Hour15 4.11e-03 (1.31e-02)
Hour16 2.92e-02 (9.96e-03)	Hour17 4.01e-02 (6.85e-03)	Hour18 -2.4e-02 (6.85e-03)	DHI 2.01e-02 (3.9e-04)
DHI^2 -5.21e-05 (6.83e-07)	DHI^3 5.43e-08 (9.69e-10)	DNI 2.57e-03 (4.30e-05)	DNI^2 -3.67e-06 (1.01e-07)
DNI^3 3.13e-09 (7.06e-11)	Clearsky DHI 1.17e-03 (5.92e-04)	Clearsky DHI^3 5.7e-09 (1.37e-10)	Clearsky DNI -1.85e-03 (2.34e-04)
Clearsky DNI^2 1.09e-06 (2.34e-07)	Clearsky DNI^3 -4.15e-10 (1.37e-10)	Cloud type 2 4.72e-03 (7.03e-03)	Cloud type 3 3.91e-03 (7.78e-03)
Cloud type 4 1.41e-02 (8.75e-03)	Cloud type 6 -3.75e-02 (1.09e-02)	Cloud type 7 -5.18e-02 (1.12e-02)	Cloud type 8 -1.54e-02 (1.2e-02)
Cloud type 9 -7.69e-02 (2.03e-02)	Dew point 1.02e-02 (2.33e-03)	SZA -1.73e-02 (2.14e-03)	Surf albedo 0.14 -5.51e-03 (8.82e-03)
Surf albedo 0.15 -7.87e-03 (1.44e-02)	Surf albedo 0.16 -1.65e-03 (2.05e-02)	Surf albedo 0.17 2.8e-03 (2.75e-02)	Prec water 1.88e-01 (1.39e-02)
Prec $water^2$ -8.53e-02 (6.13e-03)	Prec $water^3$ 1e-02 (9.05e-04)	Wind direction 2.71e-05 (1.05e-05)	Rel humidity -1.38e-02 (2.76e-03)
Rel $humidity^2$ 1.87e-04 (3.46e-05)	Rel $humidity^3$ -1e-06 (1.77e-07)	$Temperature^2$ -3.9e-04 (9.76e-05)	$Temperature^3$ 5.91e-06 (1.51e-06)
$Pressure^2$ 1.98e-04 (4.05e-05)	$Pressure^3$ -1.35e-07 (2.75e-08)	DNI - SZA 4.67e-06 (2.82e-07)	Clear DNI - SZA 1.18e-05 (2.10e-06)
Surf albedo - Clearsky DNI 2.91e-03 (7.50e-04)	Surf albedo - DHI -8.53e-03 (2.49e-03)	Surf albedo - Clearsky DHI -1.5e-02 (3.96e-03)	Cloud type - DHI 2.99e-05 (5.11e-06)
Cloud type - DNI 7.84e-06 (2.44e-06)	Prec water - DHI -1.31e-04 (2.01e-05)	Prec water - DNI 1e-04 (6.08e-06)	

The coefficients' absolute values are in line with the ones of the best methods analyzed before, such as Linear regression with splines, lasso, and Elastic net. Polynomial terms have higher absolute values than regularization methods. They are probably proportional to the number of variables in the model since the more the variable, the higher the coefficients' absolute values to have the same relevance.
The weight assigned to solar irradiance and meteorological and atmospheric variables is almost the same, except for the interaction terms, since the ones including Surface albedo dominate the others. The variables associated to hours, months, and cloud types stand out from the others due to their high coefficients.

7.2.1 Predictions

Since the observations are too many, as for the analysis in Chapter 4, the best representation for predictions is a graph with observed values on the abscissa axis, predicted values on the ordinate axis, and the bisector of the first and third squares, which corresponds to the perfect prediction of the observations. Figure 7.3 shows the predictions performed by the linear regression with polynomials assuming the

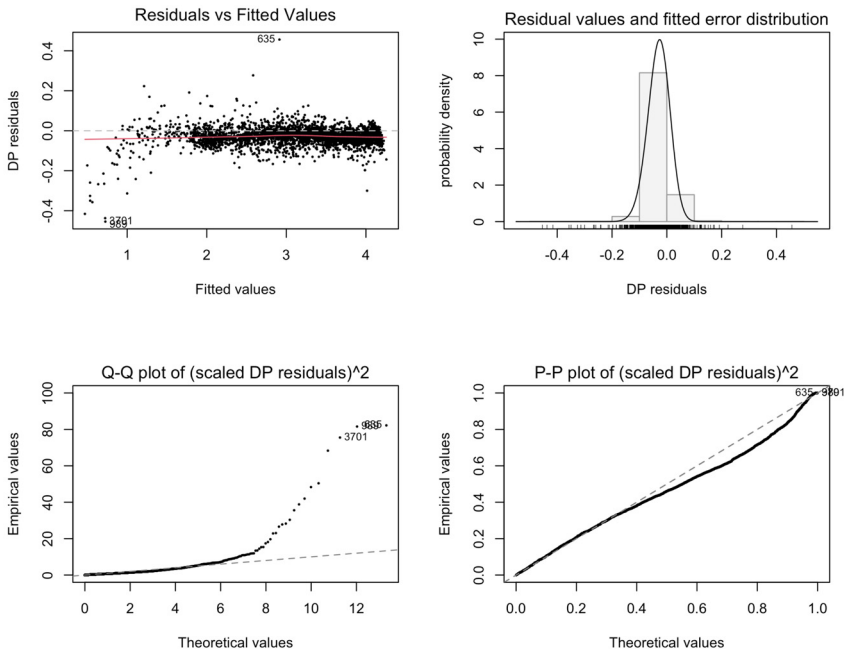

Figure 7.2: Residual analysis graphs of the linear regression model with quadratic and cubic terms, interactions, and assuming Skew-Normal distribution on the response variable.

Skew-Normal distribution on the response. The mean squared error is 0.0016, the root mean squared error is 0.0396, and the AdjR2 is 0.997. The result is generally satisfactory. The model makes almost imperceptible errors when Global Horizontal UV Irradiance is low. However, the predictions tend to slightly overestimate the true values when Global Horizontal UV Irradiance is high.

7.3 Principal component regression

An alternative technique to analyze the data is Principal Component Regression (PCR) (Hastie, Tibshirani, and Friedman, 2011, Chapter 3), which defines principal components as linear combinations to find the covariates affecting the response variable predominantly.
Due to the high data variability, the analysis has started with a dataset considering the same polynomial terms introduced for the linear regression based on the normal distribution: DHI, DNI, Clear sky DHI, Clear sky DNI, Precipitable water, Pressure, Relative humidity, and Temperature. The PCR model has been built using the first 20 principal components. The procedure could have found more principal components, but it would have been useless, as we will see further in the discussion.
At first, the amount of explained variance for each principal component has analyzed. Figure 7.4 shows the percentage of explained variance for each principal component. If we look at the graph, we can see that the first principal component explains 15.7% of the total variance, which is noteworthy. The second explains 12.76% of the whole and the third the 8.02%. From the fourth principal component on, the explained variance

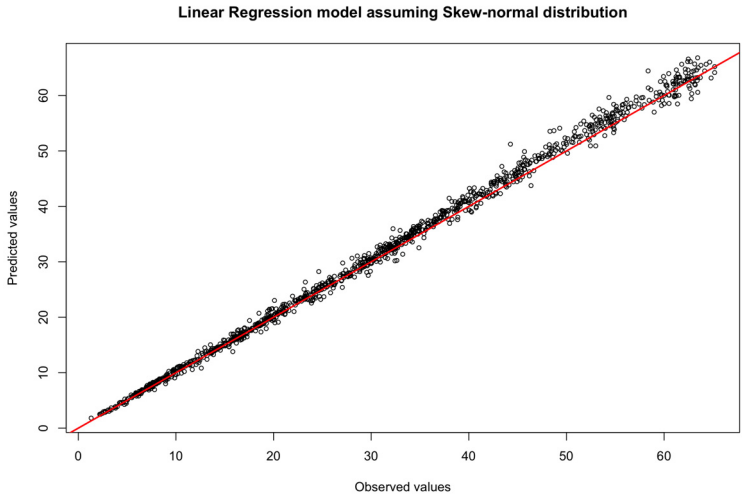

Figure 7.3: Predictions of Global Horizontal UV Irradiance of wavelength in the range of 280-400nm, performed by linear regression assuming Skew-Normal distribution.

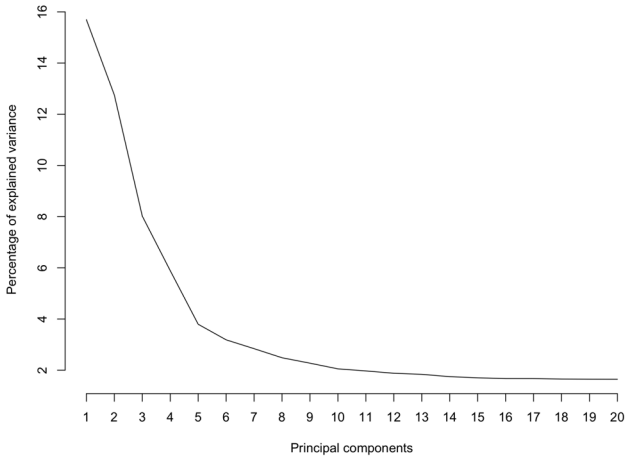

Figure 7.4: Graph of the explained variance of each principal component considered by PCR.

becomes almost negligible since it is about 5% and goes down till the last five, whose explained variance has stabilized at about 1.6%.
Cross-validation with the one-sigma approach for automatic selection of the best number of principal components, which returns the number of principal components such that the cross-validation value is within one standard error from the absolute optimum, has been run and has returned 14. This output is coherent with the graph since the explained variance from the fourteenth component on is low and stable, so there is no advantage from including other components in the model.
The algorithm can explain 76.43% of the total variance using 20 principal components. This result is not so satisfactory since all the other examined approaches have performed better. However, PCR has been implemented mainly to find new attractive features for improving the interpretation of the scientific phenomenon.
Plots in Figure 7.5 and Figure 7.6 illustrate the variation of MSE and R2 of PCR as the number of components considered increases. In Figure 7.5 the MSE drops under

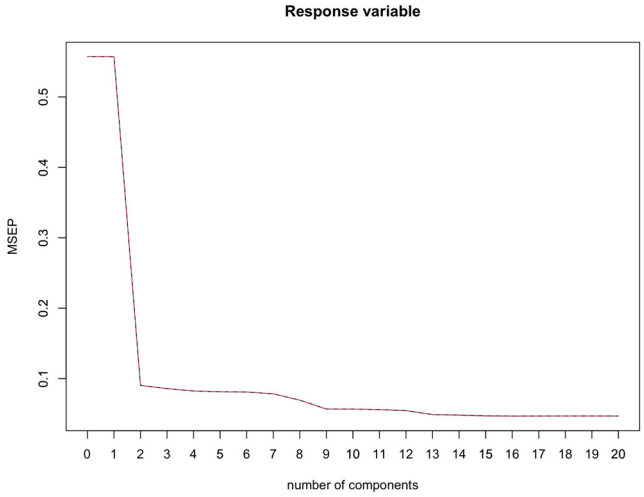

Figure 7.5: Validation plot showing the mean squared error predicted (MSEP) evaluated by PCR as the number of components increases.

0.1 with the first principal component, then it goes down slower till the ninth. The decrease slows down even more till the thirteenth and finally, no improvement is visible.
The graph in Figure 7.6 is specular about R2. The coefficient R2 increases to over 0.8 with the first principal components, and then the increments have the same trend as the decrements of the mean squared error previously described.
From both the graphs, we can see that the first principal component influences the response much more than any other. The second, the seventh, and the eighth are also important since they provide a visible decrease in the mean squared error and increase the R2 index.
A graph showing the value of the coefficients associated to the variables in each principal component is shown in Figure 7.7. It allows more information about how data are affected by principal components. The trends of almost all the principal components overlap. However, as expected, the first and the second stand out, while the other does not. The seventh and the eighth stand out a little, but since the variance they explain

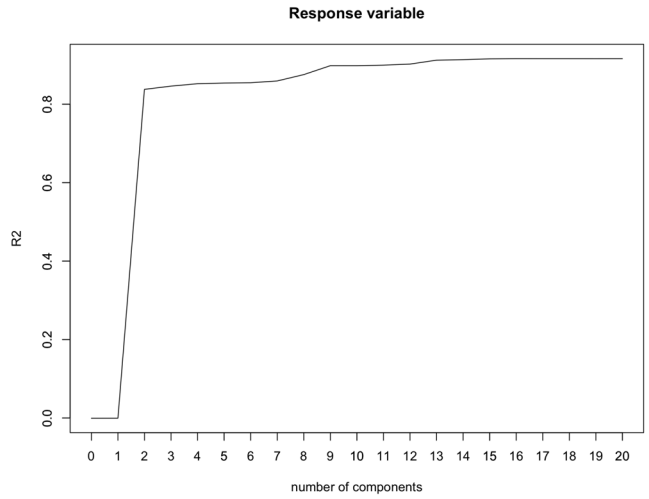

Figure 7.6: Validation plot showing the R squared coefficient evaluated by PCR as the number of components increases.

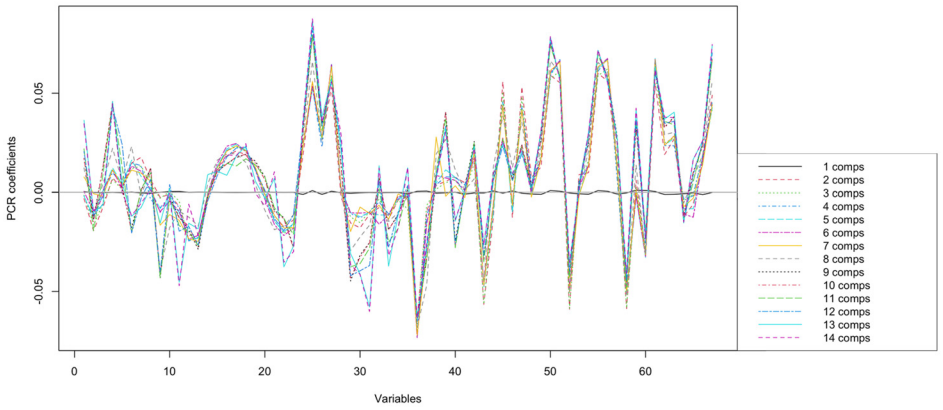

Figure 7.7: Coefficient plot showing the coefficients' variables value for each principal component. The abscissa axis contains the increasing number of variables instead of the variables' names because they would be unreadable since they are too many. The order refers to Figure 4.9.

is a few, we will not consider them.
Specific graphs to analyze the variables' coefficients of these four principal components are provided in Figures 7.8 and 7.9. Figure 7.8 is about the first principal component. The variables' coefficients from this component are very small. However, the algorithm

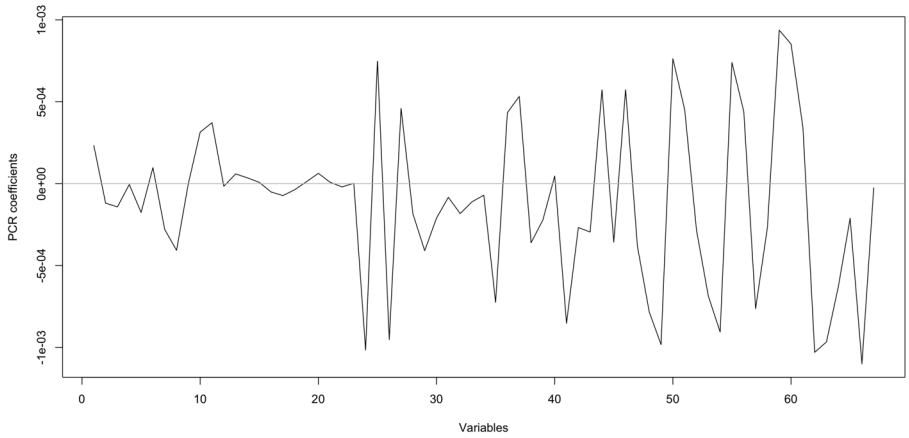

Figure 7.8: Coefficient plot showing the coefficients' variables value for the first principal component. The abscissa axis contains the increasing number of variables instead of the variables' names because they would be unreadable since they are too many. The order refers to Figure 4.9.

has assigned more weight to the solar irradiance coefficients DHI, DNI, Clear sky DHI, and Clear sky DNI, also at the power of two and three, and all the interactions except for between Precipitable water with DNI and Cloud type with DNI. The others have irrisory coefficients. None of the other components overlaps the first, but sometimes they accentuate part of the coefficients it has already addressed. From this result, we can say that solar irradiance coefficients, also combined with Surface albedo and Solar Zenith Angle, and the interactions between Cloud type and Precipitable water with DHI are the most relevant for the phenomenon studied in this thesis since they are the only significantly considered by the first principal components, which explains a large portion of the whole variance.
Figure 7.9 is about the second principal component. The variables' coefficients from this component are at least one order of magnitude greater than the first. Moderately high coefficients have been assigned to the months of Summer, the hours close to midday, and the solar irradiance indexes. Moreover, high coefficients have been assigned to Cloud type 7, 8, and 9, which corresponds to generally thick clouds and generally to atmospheric and meteorological variables, more accentuated when their degrees are two and three. The second principal component emphasizes atmospheric, meteorological, and time variables, which should have importance.
Therefore, from the analysis with PCR, we can say that variables associated to solar irradiance are the most relevant. However, the variables related to atmospheric and meteorological conditions must not be undervalued, especially Cloud type and Precipitable water, which have proved their importance in explaining the phenomenon studied.

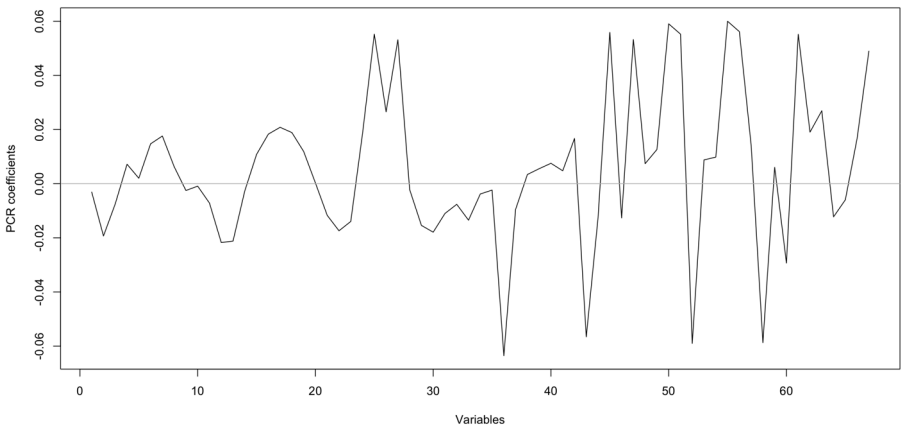

Figure 7.9: Coefficient plot showing the coefficients' variables value for the second principal component. The abscissa axis contains the increasing number of variables instead of the variables' names because they would be unreadable since they are too many. The order refers to Figure 4.9.

7.4 Extended analysis with new datasets

From the analyses conducted in Chapter 4 and Chapter 6, very good results both in the interpretation of the scientific problem and in the prediction of the Global Horizontal UV Irradiance of wavelength in the range 280-400 nm have been found. These analyses refer to the dataset built in 2020 from the data captured by a station in Los Angeles. Therefore, solar irradiance, meteorological and atmospheric variables, and the associations that have been discovered among them do not consider the temporal and spatial variables since they have been set as fixed prior information. The temporal variable could be relevant because of climate change that is affecting more and more our meteorological and atmospheric conditions, shielding us less from the UV sun's rays. The spatial variable could be relevant because, depending on the geographical place on the Earth, there are different kinds of climates, which might correspond to different associations among meteorological and atmospheric variables with solar irradiance.
Therefore, two new datasets from the NSRDB and captured by the same infrastructure of sensors of the dataset used for the analyses have been considered for evaluating the models built in the previous sections and chapters and finding out information about temporal and spatial variables. The dataset to study the temporal variable refers to Los Angeles in 1998, fixing the spatial variable and going back to the past, while the dataset to study the spatial variable refers to New York in 2020, fixing the temporal variable and going where the climate is totally different.
Data have been analyzed via machine learning methods to focus on predictions and avoid preprocessing steps.

7.4.1 Analysis with a new temporal variable

The first analysis will refer to the database in Los Angeles in 1998. The structure of the data refers to Section 2.4 and the preprocessing of the data for each method refers

to Chapter 6. Predictions performed through the five machine learning models trained with the original dataset are reported below.
Predictions with the basic decision tree model have provided generally worse results than the test with the original dataset ones, as illustrated in Figure 7.10. Adjr2 is 0.9655,

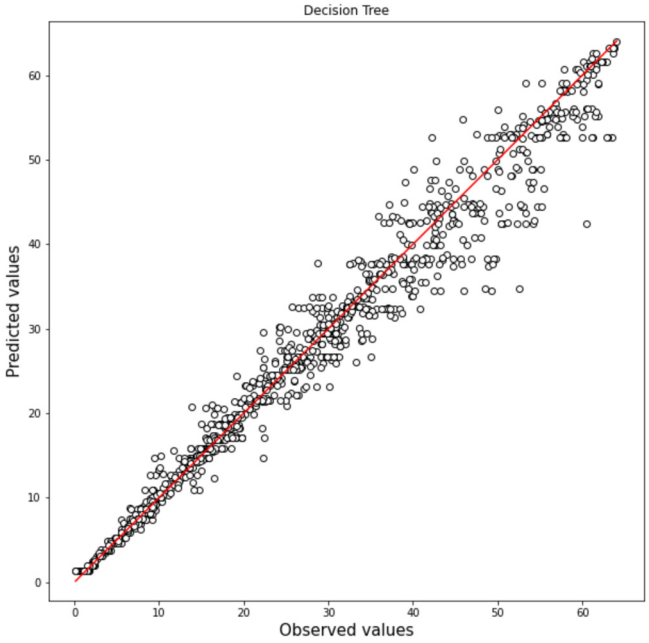

Figure 7.10: Predictions of Global Horizontal UV Irradiance of wavelength in the range 280-400nm, performed by the decision tree model with sampling at the split node using the Los Angeles 1998 dataset.

and the mean squared error is 0.038. The predictions are good when Global Horizontal UV Irradiance values are low. However, they present visible errors underestimating and overestimating the values when Global Horizontal UV Irradiance is high. Despite being a very simple model, it has generalized well, losing not too much performance with respect to the original dataset. This is not the same for the Random forest model, as illustrated in Figure 7.11. The predictions are definitely worse than the test with original dataset ones. It is comparable with the simple model since Adjr2 is 0.9687, and the mean squared error is 0.0345. The predicted values look good when the Global Horizontal UV Irradiance values are low but tend to underestimate the observed values as the Global Horizontal UV Irradiance rises.
The same considerations of the Random forest model can be extended to the Extreme Gradient boosting model, as shown in Figure 7.12. Adjr2 is 0.9581, and the mean squared error is 0.0461. These are even worse performances than the Random forest model. This is probably due to the too high complexity of those two models for the original dataset since simpler models have already given satisfactory results, making them learn too specific features that generalize worse.
The predictions performed with K-Nearest-Neighbour has given very bad results, as illustrated in Figure 7.13. Adjr2 is 0.8438, and the mean squared error is 0.1721. Despite being a simple model, such as the decision tree in its basic formulation, it seems unsuitable for this scientific problem.

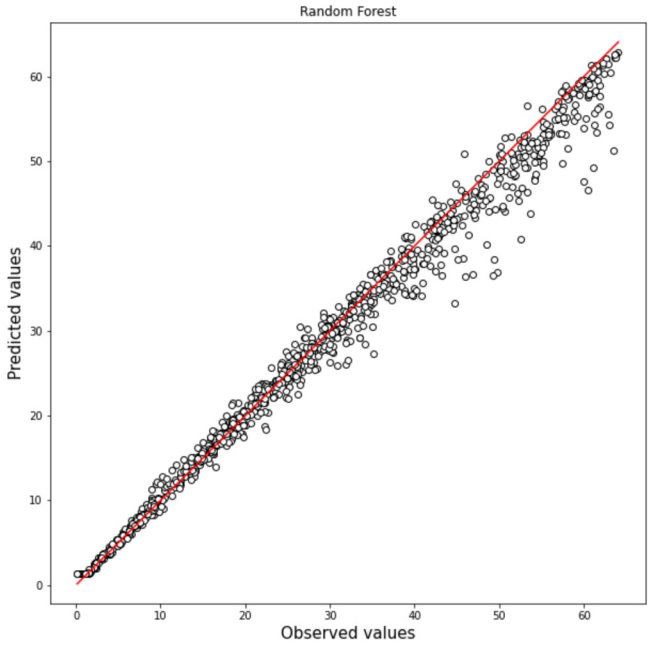

Figure 7.11: Predictions of Global Horizontal UV Irradiance of wavelength in the range 280-400nm, performed by the Random Forest model using the Los Angeles 1998 dataset.

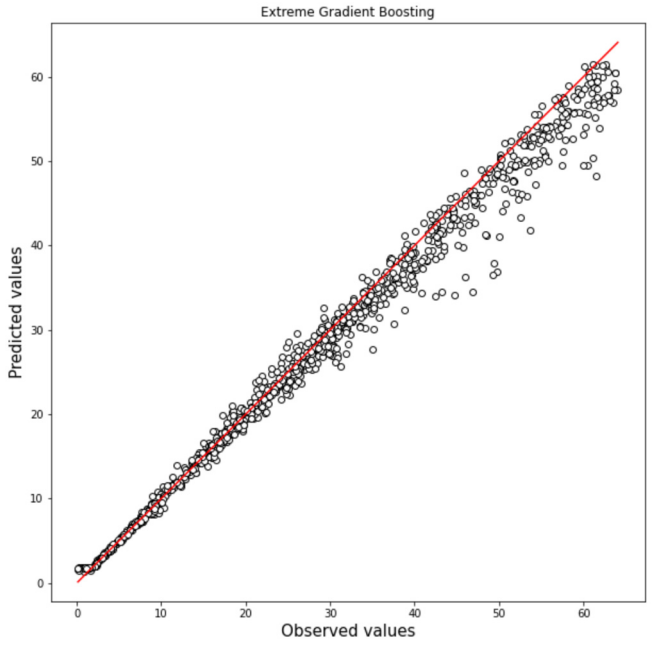

Figure 7.12: Predictions of Global Horizontal UV Irradiance of wavelength in the range 280-400nm, performed by the Extreme Gradient Boosting model using the Los Angeles 1998 dataset.

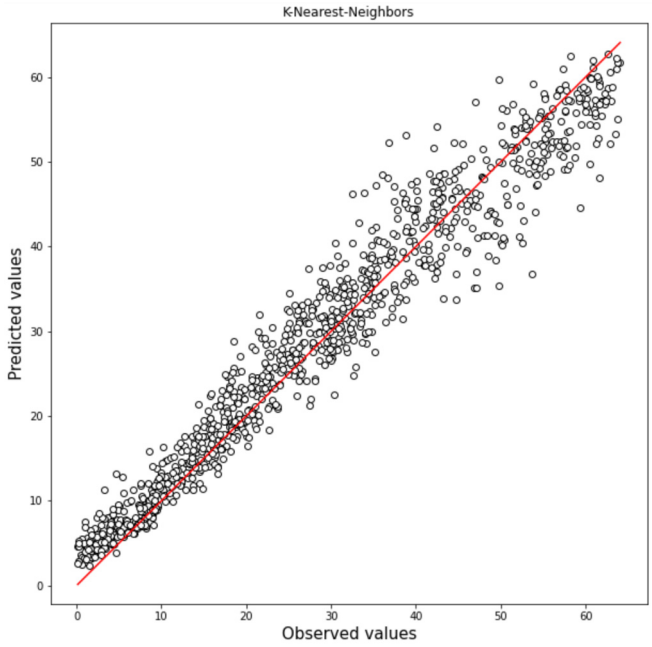

Figure 7.13: Predictions of Global Horizontal UV Irradiance of wavelength in the range 280-400nm, performed by the K-Nearest-Neighbour model using the Los Angeles 1998 dataset.

Support vector regressor has given as bad performance as K-Nearest-Neighbour, as shown in Figure 7.14. Adjr2 is 0.8721, and the mean squared error is 0.1409. The main problem for SVR is the overestimation of many predictions throughout all the spectrum of Global Horizontal UV Irradiance values, in addition to a few remarkable underestimations. This result is quite unexpected since it has been one of the best models in the original dataset evaluation. Probably, it has learned too specific features becoming unable to generalize well anymore.
A summary of the results that have been stated of the prediction analysis on the dataset of Los Angeles in 1998, compared with the predictive analysis on the dataset of Los Angeles in 2020, is reported in Table 7.2.

Table 7.2: Comparison of the performance between machine learning methods applied on the data captured in Los Angeles in 2020 and the same model tested in the dataset captured in Los Angeles in 1998.

Algorithm [a]	Los Angeles 1998			Los Angeles 2020		
	AdjR2 [c]	MSE [c]	RMSE [c]	AdjR2	MSE	RMSE
DT	0.9655	0.038	0.1949	0.9858	0.0068	0.0825
RM	0.9678	0.0345	0.1857	0.9952	0.0022	0.047
XGBoost	0.9581	0.0461	0.2147	0.9972	0.0015	0.0387
KNN	0.8438	0.1721	0.4148	0.9717	0.0146	0.1208
SVR	0.8721	0.1409	0.3753	0.996	0.0022	0.0469

[a] DT = Decision Tree; RM = Random Forest; XGBoost = Extreme Gradient Boosting; KNN = K-nearest-neighbour; SVR = Support Vector Regressor.
[b] Adjr2 = Adjusted R squared; MSE = Mean Squared Error; RMSE = Root Mean Squared Error.

Some further trials have been made with datasets between 1998 and 2020, and the general trend is that the further we go back over the years, the more the simple decision tree model generalizes better than the complex ones.

7.4.2 Analysis with a new spatial variable

The first analysis will refer to the database in New York in 2020. The structure of the data refers to Section 2.4, and the preprocessing of the data for each method refers to Chapter 6. Predictions with the five machine learning models have been performed, but very poor results have been obtained.
See, for example, the results from Random forest model in Figure 7.15, with Adjr2 equal to 0.0782, and the mean squared error equal to 0.8956. Such a poor behaviour is probably due to the different associations between meteorological and atmospheric conditions and solar irradiance, which do not become simpler or more complex but change deeply in their structure according to the climate conditions. Therefore, new proper models should be implemented, depending on the geographical region, so the climate conditions of the place of interest.

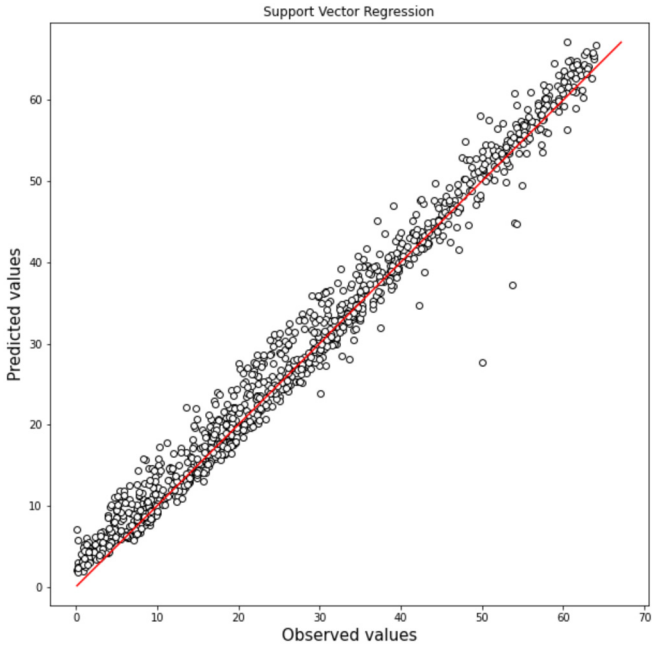

Figure 7.14: Predictions of Global Horizontal UV Irradiance of wavelength in the range 280-400nm, performed by the Support Vector Regressor model using the Los Angeles 1998 dataset.

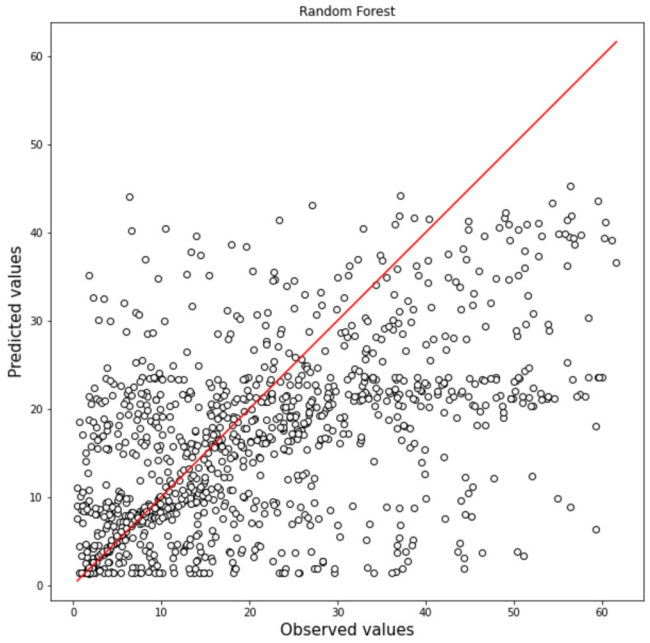

Figure 7.15: Predictions of Global Horizontal UV Irradiance of wavelength in the range 280-400nm, performed by the Random Forest model using the New York 2020 dataset.

Chapter 8

Comparison and discussion of the results

This chapter focuses on discussions and comparisons of the results obtained from the analysis with data mining and machine learning methods. The strengths and weaknesses of these two categories of techniques will be pointed out and assessed according to the study conducted. Finally, considerations about scientific results on the solar irradiance phenomenon will be discussed.

8.1 Considerations on data mining analysis

Linear regression models have generally fitted the data well despite a high variance. The basic formulation of the model with the initial covariates and the interactions has been used as a baseline for further improvements. Polynomials have been essential to fit the model very well. Natural splines have allowed more flexibility, fitting the data even better and maintaining good interpretability. In the models with polynomials and natural splines, the mean squared error is low, and Adjr is close to one, so the performances are good and quite similar. There are some anomalous data coming from extreme climate conditions that the models cannot fit. The more flexibility, the more those anomalous observations are fitted well, but the worse the interpretability. The linear model with natural splines turned out to be the best compromise. As a deepening, the linear model with polynomials assuming Skew-Normal distribution has been implemented to allow more flexibility. The skewness in the response variable distribution has been significant, and the performance has been good, comparable to the model with natural splines.
Because of the high variance in the data and the high number of covariates, several shrinkage methods have been implemented. Lasso and Elastic-Net turned out to be the best solutions. Lasso has dropped fewer variables than Elastic Net but with a better performance.
Finally, looking at the predictions, we can see a common trend for all models: the higher the value of Global Horizontal UV Irradiance, the higher the prediction error. However, the general performances are good, in agreement with the considerations stated above.

8.2 Considerations on machine learning analysis

Machine learning methods have been chosen for predicting Global Horizontal UV Irradiance of wavelength 280-400 nm. The decision tree has been the first method implemented since it is one of the most simple machine learning models and allows interpretability if the final tree model is not too complex. However, this has not been the case. The performance of the decision tree has been good but not satisfying. Hence, two ensemble learning techniques based on decision trees have been chosen: Random forest, which aims to build effective models against the high variance, and Extreme Gradient boosting, which aims to build effective models against high bias. Despite these two models having two different purposes, they have reached excellent and similar performances, clearly better than the basic decision tree model. Random forest performs well probably because of the high variance on the data, while Extreme Gradient boosting performs well probably because it is a refined and optimized algorithm.
KNN is a simple algorithm, and it has been implemented because other simple algorithms have worked well. However, it has provided poor results, probably because it has not been suitable for the data. SVR is a classic machine learning method and has given very good results, comparable to Random Forest and Extreme Gradient Boosting. The predictions performed by Random Forest, Extreme Gradient boosting, and SVR have been excellent. The few prediction errors have mainly concerned the medium-high value of Global Horizontal UV Irradiance, while the rest are great. Therefore, more complex methods, such as neural networks, have not been implemented.

8.3 Comparison between data mining and machine learning

A comparison of the performance in data prediction between the data mining and the machine learning methods implemented in this thesis is reported in Table 8.1.

Table 8.1: Comparison of the performance of all the implemented algorithms.

Algorithm [a]	Cross-validation (N = 2670) [b]			Prediction (N = 1144) [b]		
	AdjR2 [c]	MSE [c]	RMSE [c]	AdjR2	MSE	RMSE
LM	0.9988	0.0133	0.1153	0.9762	0.0178	0.1335
LM-P	0.9998	0.0018	0.0424	0.9968	0.0018	0.0429
LM-NSP	0.9999	0.0013	0.0361	0.9976	0.0013	0.0364
LM-SN	Hour9	Hour10	Hour11	0.997	0.0016	0.0396
Ridge	0.9803	0.011	0.1049	0.9785	0.0125	0.1118
LASSO	0.9961	0.0022	0.0469	0.9961	0.0025	0.05
EN	0.9898	0.0056	0.0748	0.9884	0.0059	0.0768
AL	0.9673	0.0196	0.14	0.9651	0.02	0.1414
DT	0.9858	0.0068	0.0825	0.9846	0.0075	0.0866
RM	0.9952	0.0022	0.047	0.995	0.0025	0.0499
XGBoost	0.9972	0.0015	0.0387	0.997	0.0016	0.0398
KNN	0.9717	0.0146	0.1208	0.967	0.0161	0.1269

SVR	0.996	0.0022	0.0469	0.9954	0.0024	0.049

[a] LM = Linear Regression; LM-P = Linear Regression with polynomials; LM-NSP = Linear Regression with Natural Splines; LM-SN = Linear Regression assuming the Skew-Normal distribution on the response variable; Ridge = Ridge Regression; EN = Elastic Net; AL = Adaptive Lasso; DT = Decision Tree; RM = Random Forest; XGBoost = Extreme Gradient Boosting; KNN = K-nearest-neighbour; SVR = Support Vector Regressor.
[b] N = number of observations considered for cross-validation and prediction, respectively.
[c] Adjr2 = Adjusted R squared; MSE = Mean Squared Error; RMSE = Root Mean Squared Error. RMSE corresponds to the residual standard error computed by linear regression models.

The data mining methods performing the best are linear regression with natural splines, linear regression with polynomials assuming Skew-Normal distribution on the response variable, and Elastic Net, while the machine learning methods performing the best are Random Forest, Extreme Gradient boosting, and SVR. Since the performance of all of them is satisfactory, intrinsic features related to the algorithms will be considered.
At first, if we look at the model interpretability in order to learn new interesting associations and trends between data, we will discard machine learning methods because they generally focus on too complex features without providing meaningful graphical representations. Among the data mining methods, Elastic Net emphasizes the most important variables dropping the others, while the linear regression models mainly concern the fit of the data maintaining non-crucial variables that make the fit more refined and precise. All the data mining methods considered allow many kinds of graphs and are helpful to reach a global and detailed comprehension of the scientific problem.
Looking at the time computational performances, data mining methods are very fast, while machine learning methods are more intensive. In particular, Extreme Gradient boosting has required a GPU to finish its computation in a reasonable time.
In general, machine learning approaches learn very complex features, typically incomprehensible to humans, so they are suitable for data predictions because those features could often fit the data very well but are useless for interpreting the data. Instead, data mining approaches aim to interpret the relationships among data but are not the best at making predictions. Nevertheless, predictions highly depend on the data structure. In fact, simple models, such as linear regression with polynomials, are suitable for the dataset used in this thesis, so the predictions have been very good also for data mining approaches.
If we look at the deepening, we will notice that the linear regression with polynomials assuming the Skew-Normal distribution on the response variable is a refinement strictly related to the initial dataset, so it should generalize worse than the other linear regression approaches. Therefore, the linear regression with natural splines turns out to give more advantages than the other data mining techniques.

8.4 Scientific results

In this section, scientific considerations summarizing the results of all the analyses will be reported. In support of this, a graph concerning the comparison between the variables' coefficients assigned from lasso and Elastic Net is shown in Figure 8.1, and two other graphs concerning the variables' coefficients assigned from the linear regression with natural splines and the linear regression with polynomials assuming the Skew-Normal distribution on the response are shown in Figure 8.2 and 8.3, respectively.
From the interpretation of the data captured in Los Angeles in 2020, we can confirm that Global Horizontal UV Irradiance of wavelength 280-400 nm is related to the solar

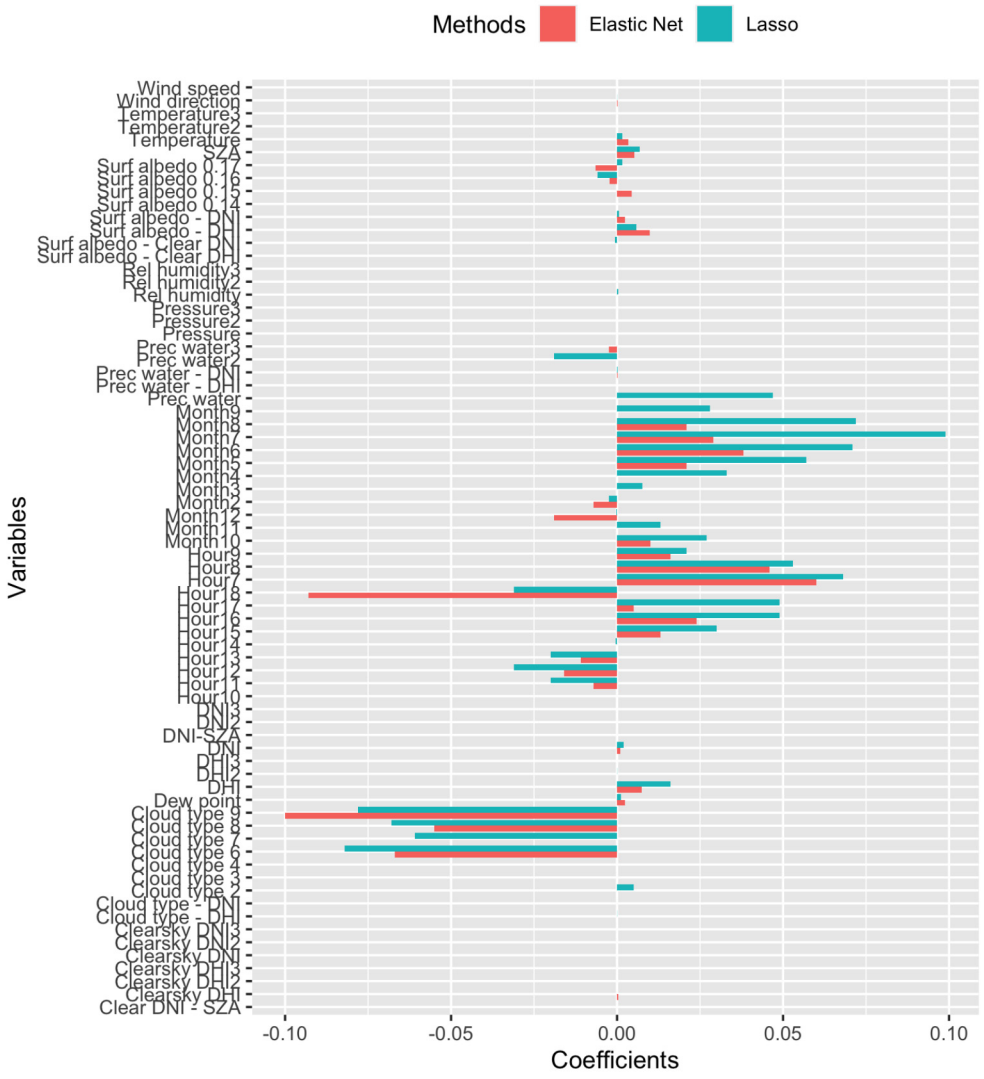

Figure 8.1: Graphical comparison between the variables' coefficients assigned from lasso and Elastic Net.

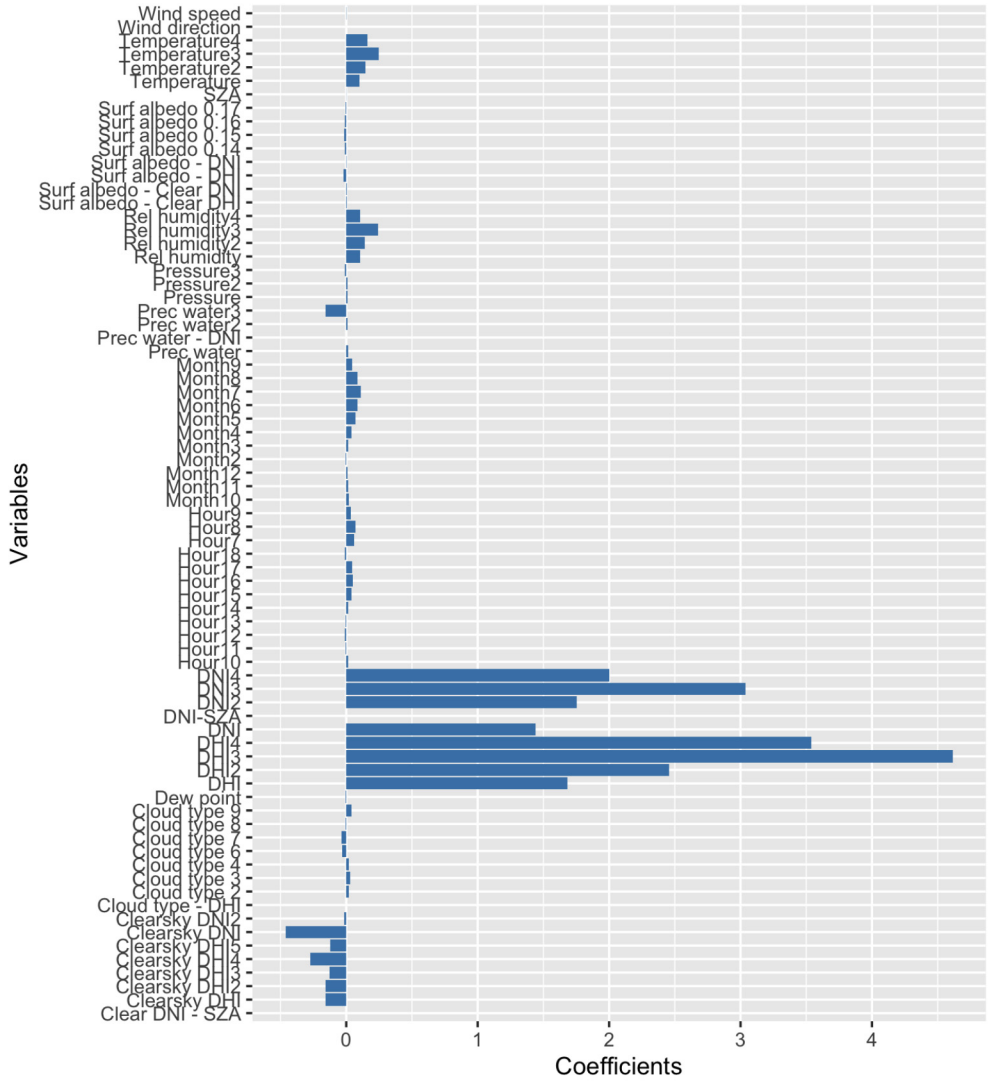

Figure 8.2: Graphical representation of the variables' coefficients assign from the linear regression model with natural splines.

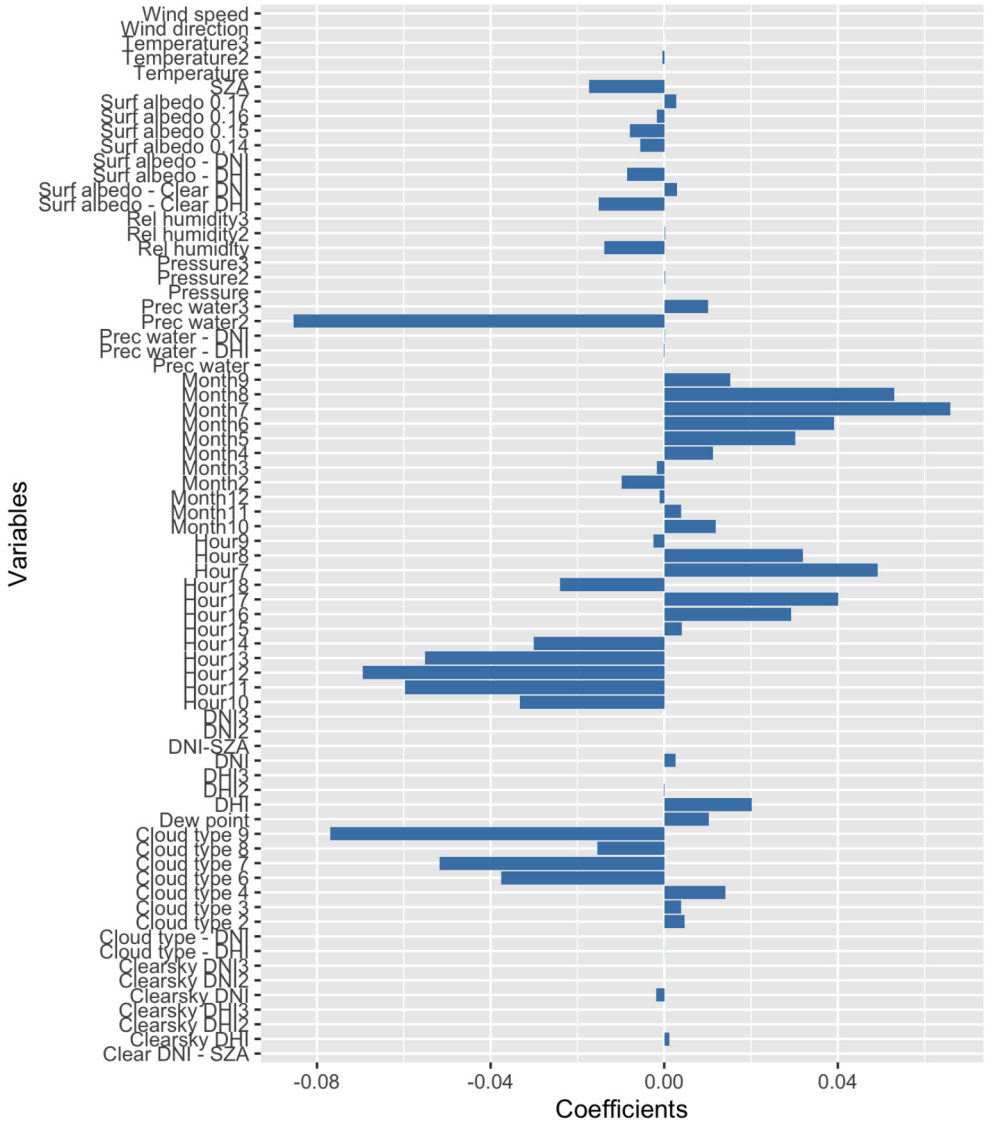

Figure 8.3: Graphical representation of the variables' coefficients assign from the linear regression model with polynomials assuming the Skew-Normal distribution on the response.

irradiance indices. In fact, Global Horizontal UV Irradiance refers to a subset of the whole irradiance spectrum and to the reflectivity coefficient of the surface close to the sensor to which solar rays impact. Therefore, the more intense the solar irradiance, the higher the Global Horizontal UV Irradiance value. Moreover, even if the reflected rays could be underestimated, their intensity has proved to be relevant.
The last variable related to the irradiance is the Solar Zenith Angle, which depends on the position of the Sun with respect to the Earth. It turned out to be meaningful since the closer the cosine of the angle to one, the more powerful the solar irradiance intensity. Looking at the temporal features, Month and Hour are considered crucial from all the methods since SZA varies a lot during the seasons and the days.
Meteorological and atmospheric features are the most interesting. Despite being not considered by mathematical models in the literature, they affect the Global Horizontal UV Irradiance quite a lot. Cloud type affects solar irradiance the most, followed by Precipitable water. Clouds become relevant when they are thick and full of water because they make solar rays scattered. Hence, Global Horizontal UV irradiance is influenced by those conditions because it is very sensitive to impeding, due to the intrinsic structure of UV rays. Temperature and Pressure turned out to be significant. However, their interpretation is more difficult because they follow a general trend: high temperature and high pressure imply high Global Horizontal UV Irradiance, but some variability is associated with this statement. For example, we can have a high temperature on some days of July but a sky full of thick clouds, resulting in a lower value of Global Horizontal UV Irradiance than some days of January with a clear sky. In fact, polynomials or natural splines have been assigned to those variables against the variability, and different signs have been to coefficients of different degrees. Relative humidity is also relevant but suffers more from high variability than Temperature and Pressure.
According to the literature (see Seven factors affecting UV irradiance) (see EPA report), most of the results found through the interpretation of the data analysis are confirmed, such as cloud coverage, features related to the time, surface reflexivity, and temperature. However, some other factors that are not covered in the literature have been tested. Relative humidity and Pressure, have been found to affect the Global Horizontal UV Irradiance, while Dew point, Wind speed, and Wind direction have presented poor associations.
Looking at the deepening concerning the data captured in Los Angeles in 1998 and New York in 2020, we can deduce information about how temporal and spatial features affect Global Horizontal UV Irradiance. Going back to the past and fixing the geographic position does not highlight changes in the relationships between the variables and the response, but it does in the meteorological and atmospheric conditions, probably due to climate change. This can be seen in the analysis conducted in Section 7.4 since predictions have shown higher errors, mainly underestimating the Global Horizontal UV Irradiance, but also in the Los Angeles database of meteorological conditions (see Los Angeles weather archive). Instead, fixing the year but moving to New York, where the meteorological conditions are totally different, makes the associations not valid anymore. This means that the geographical location is very important since it influences the climate and, consequently, the relationships between variables.

Chapter 9

Conclusions

In this thesis, Global Horizontal UV Irradiance of wavelength 280-400 nm and its relationships with solar irradiance coefficients and meteorological and atmospheric variables have been examined. The analysis has started from a dataset downloaded from a repository of NREL (see NSRDB new repository), and many algorithms, coming from data mining and machine learning techniques, have been proposed to fit the data. A summary of their main features is reported in Table 9.1.

Table 9.1: Overview of all the implemented algorithms

Algorithm [a]	Group	Model possible non-linear relationships between response [b] and predictors	Variable selection	Model structure
LM LM-P LM-NSP LM-SN	Linear regression algorithms	No, only a priori transformations can be applied	Yes, except for LM	Showed coefficient estimates and gave high interpretability
Ridge LASSO EN AL	Shrinkage or regularization algorithms	No, only a priori transformations can be applied	Yes, except for Ridge	Showed reliable coefficient estimates against high data variability, paying performance
DT RM XGBoost KNN SVR	Machine learning algorithms	Yes	No	Difficult to interpret, except for DT when the generated tree is short and narrow

[a] LM = Linear Regression; LM-P = Linear Regression with polynomials; LM-NSP = Linear Regression with Natural Splines; LM-SN = Linear Regression assuming the Skew-Normal distribution on the response variable; Ridge = Ridge Regression; EN = Elastic Net; AL = Adaptive Lasso; DT = Decision Tree; RM = Random Forest; XGBoost = Extreme Gradient Boosting; KNN = K-nearest-neighbour; SVR = Support Vector Regressor.

[b] Response = Global Horizontal UV Irradiance of wavelength 280-400 nm.

From the literature, many meteorological and atmospheric factors have been found to affect the Global Horizontal UV Irradiance, but few mathematical models concerning that exist, and they do not consider those factors. Instead, this thesis focuses on the relationships between those factors and Global Horizontal UV Irradiance to acquire knowledge about interpretation and prediction, and new models are proposed. Meteorological and atmospheric variables turned out to be significant, especially clouds, temperature, and humidity.
The deepenings have highlighted some limits of this analysis. The relationships found by the models are strictly related to the temporal and spatial features of the data, so, as at least one of them changes, independently from the model implemented, it should be re-trained. To overcome those limits, extensions of this work should be considered. An extension could be to add further variables in the dataset about air quality, ozone, and altitude because we know from the literature that they should affect Global Horizontal UV Irradiance (see Seven factors affecting UV irradiance). Hence, more precise and generalizable estimates should be found. Another extension could be to merge datasets of multiple geographical locations with different climate conditions into a unique dataset and use more complex models, such as neural networks, to fit the data. In this way, the relationships between the variables and Global Horizontal UV Irradiance could generalize better to new datasets, allowing good predictions at the price of losing interpretability.

Appendix A

R programming language

A.1 Libraries

- **glmnet** (Library glmnet): library for Lasso and Elastic-Net Regularized Generalized Linear models in language R. It offers efficient procedures for fitting Lasso and Elastic Net models for linear regression, logistic and multinomial regression, Poisson regression, Cox model, multiple-response Gaussian, and the grouped multinomial regression. This work uses version 4.1-4, published on the date 2022-04-13.
- **leaps** (Library leaps): library for Regression Subset Selection. It offers an exhaustive search for the best subsets of the covariates to predict the response variable, and it uses an efficient version of the branch-and-bound algorithm. This work uses version 3.1, published on the date 2020-01-16.
- **pls** (Library pls): library for Principal Component Regression and Partial Least Squares Regression. It offers a traditional interface according to R conventions and deals with the existing R generic functions. This work uses version 2.8-1, published on the date 2022-07-16.
- **splines** (Library splines): library to design, create, evaluate, and graphically represent splines. This work uses version 3.6.2, published in July 2021.
- **caret** (Library caret): library for classification and regression training. It needs the ggplot2 library to be installed before being able to run. It offers methods to set up, analyze, and plot training models with lots of configurations and parameters. This work uses version 6.0-92, published on the date 2022-04-19.
- **gam** (Library leaps): library for Generalized Additive Models. It offers methods to create, fit, predict, and plot Generalized Additive Models. This work uses version 1.20.2, published on the date 2020-06-27.
- **sn** (Library sn): library to run a Linear Regression model whose response variable distribution and assumption belong to the Skew-Normal family, such as Skew-t and Skew-Cauchy. It offers methods to build and manipulate those probability distributions. Moreover, only for the Skew-Normal and the Skew-t, statistical methods are provided for data fitting and model diagnostics, both in the univariate and multivariate cases. This work uses version 2.0.2, published on the data 2022-03-07.

- **Metrics** (Library Metrics): library implementing evaluation metrics for regression, time series, classification, and information retrieval algorithms. This work uses version 0.1.4, published on the date 2018-07-09.
- **caTools** (Library caTools): library providing utility functions, such as moving, rolling, and running window statistic functions, read/write for GIF, fast calculation of AUC, and many others. This work uses version 1.18.2, published on the date 2021-03-08.
- **tidyverse** (Tidyverse libraries collection): collection of libraries for data science purposes. Ggplot2, dplyr, and rsample belong to it.
- **ggplot2** (Library ggplot2): library for creating elegant data visualizations declaratively. It is based on the Grammar of Graphics. It is needed to install the caret library. This work uses version 3.3.6, published on the date 2022-05-03.
- **dplyr** (Library dplyr): library dplyr providing a grammar of data manipulation, represented by a consistent set of efficient functions. It is included in the library tidyverse. This work uses version 1.0.9, published on the date 2022-04-28.
- **rsample** (Library rsample): library for creating and summarizing different type of resamples of data and corresponding classes for the analysis. This work uses version 1.1.0, published on the date 2022-08-08.

A.2 Commands

- **summary**: provide summaries results of model fitting functions.
- **as.factor**: transform a column type from numeric to factor (also called categorical).
- **hist**: plot a histogram of the given data values.
- **boxplot**: compute a boxplot of the given data values.
- **plot**: compute scatter plots according to the objects given as a parameter.
- **validationplot**: compute a plot with validation statistics, such as R^2 and mean squared error (MSE), of a model, fitted using principal component regression, as a function of the number of components.
- **coefplot**: plot the coefficients from model objects.
- **biplot**: compute a biplot representing the observations and the variables of a matrix of data.
- **coef**: extract model coefficients from objects of modeling functions.
- **names**: extract the variable names from the result of the function coef.
- **regsubsets**: performs model selection by forward or backward stepwise algorithm, exhaustive search, or sequential replacement.
- **lm**: fit linear models to carry out regression even with splines, single stratum analysis of variance, and analysis of covariance.

- **glmnet**: fit a generalized linear model with ridge regression, lasso, or Elastic Net regularization.
- **ns**: implement natural splines with specifications of knots allowing to centre, and clamp them.
- **selm**: fit a linear regression model with a skew-elliptical error term. It has been used to fit a linear regression model assuming the Skew-Normal distribution on the response.
- **trainControl**: control the train parameters for performing cross-validation. It is useful when many parameters must be considered.
- **train**: fit data mining models over different tuning parameters. It is often used in combination with trainControl.
- **pcr**: fit a transfer function model using principal component regression.

Appendix B

Python programming language

B.1 Libraries

- **Sklearn** (Library sklearn): library for machine learning algorithms concerning classification, regression, clustering, dimensionality reduction, model selection, and preprocessing tasks. It provides simple and efficient tools for making data predictions. It is built on the three main libraries for scientific computing and data visualization, NumPy, SciPy, and Matplotlib, to provide an accessible interface and allow reusable code. This work uses version 1.0.2 published on the date 2021-12-25.
- **XGBoost** (XGBoost): optimized library for the Extreme Gradient boosting algorithm. It is designed for high efficiency, flexibility, and portability. The implementation allows parallel and distributed computation resulting in a very fast training with proper hardware. This work uses the stable release published in 2021.
- **NumPy** (Library numpy): the main library for scientific computing. It provides a multidimensional array object with various derivations, such as arrays and matrices, and many optimized functions for fast mathematical, logical, and manipulation operations on arrays. This work uses version 1.22.0, published on the data 2021-12-31.
- **Matplotlib** (Library matplotlib): library for creating static, animated, and interactive visualizations in Python. This work uses version 3.5.2, published on the date 2022-05-02.
- **Pandas** (Library pandas): library for fast, flexible, and powerful data analysis and manipulation tools. This work uses version 1.4.2, published on the date 2022-04-02.

B.2 Commands

All the algorithms implemented in sklearn provide a common interface. The three main methods are:

- **fit**: trains the model.

- **score**: returns the adjusted R^2 obtained by the model after fit has been executed.
- **predict**: performs model predictions.

Bibliography

References

Azzalini, A. (1985). A class of distributions which includes the normal ones. *Scand J Stat* **12**.

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

Emmert-Streib, F. and M. Dehmer (2009). High-Dimensional LASSO-Based Computational Regression Models: Regularization, Shrinkage, and Selection. *Machine Learning and Knowledge Extraction.*

Fan, J. and R. Li (2001). Variable Selection via Nonconcave Penalized Likelihood and its Oracle Properties. *Journal of the American Statistical Association.*

Freund, Y. and R.E. Schapire (1996). Experiments with a New Boosting Algorithm. *International Conference on Machine Learning*, 123–140.

Goodfellow, Y., Y. Bengio, and A. Courville (2016). *Deep Learning.* The MIT Press.

Gueymard, Christian A. (2009). Direct and indirect uncertainties in the prediction of tilted irradiance for solar engineering applications. *Solar Energy* **83**, 432–444.

Habte, Aron M. *et al.* (2018). Estimating Ultraviolet Radiation From Global Horizontal Irradiance. *IEEE Journal of Photovoltaics* **9**.

Hastie, T., R. Tibshirani, and J. Friedman (2011). *The elements of statistical learning: data mining, inference, and prediction.* Second Edition. Springer.

Hoerl, Arthur E. and Robert W. Kennard (1970). Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics* **12**, 55–67.

Jaskes, G. *et al.* (2013). *An Introduction to Statistical Learning with Applications in R.* First Edition. Springer.

Mitchell, Tom M. (1997). *Machine Learning.* International Edition. McGraw-Hill Education.

R Core Team (2022). R: A Language and Environment for Statistical Computing. *R Foundation for Statistical Computing.* URL: `http://www.R-project.org/`.

Tibshirani, R. (1996). Regression Shrinkage and Selection via the lasso. *Journal of the Royal Statistical Society. Series B (methodological).*

Van Rossum, G. and Fred L. Drake (2009). *Python 3 Reference Manual.* CreateSpace.

Vignola, F. (June 2012). GHI Correlations with DHI and DNI and the Effects of Cloudiness on one-minute Data.

Zou, H. (2006). The Adaptive Lasso and Its Oracle Properties. *Journal of the American Statistical Association* **101**.

Zou, H. and T. Hastie (2005). Regularization and Variable Selection via the Elastic Net. *Journal of the Royal Statistical Society. Series B (methodological)* **67**, 301–320.

Sitography

References

EPA report. United States Environmental Protection Agency (EPA) report on ultraviolet (UV) radiation. URL: https://www.epa.gov/sites/default/files/documents/uvradiation.pdf.

Google Colaboratory. URL: https://colab.research.google.com/notebooks/intro.ipynb.

Kernel functions. URL: https://data-flair.training/blogs/svm-kernel-functions/.

Library caret. Library for classification and regression training in R. URL: https://cran.r-project.org/web/packages/caret/caret.pdf.

Library caTools. Library for utility functions in R. URL: https://www.rdocumentation.org/packages/caTools/versions/1.17.1/topics/caTools-package.

Library dplyr. Library providing a grammar of data manipulation in R. URL: https://dplyr.tidyverse.org/.

Library ggplot2. Library for data visualizations in R. URL: https://cran.r-project.org/web/packages/ggplot2/index.html.

Library glmnet. Library for the lasso and the Elastic-Net regularized generalized linear models in R. URL: https://cran.r-project.org/web/packages/glmnet/glmnet.pdf.

Library leaps. Library for Regression Subset Selection in R. URL: https://cran.r-project.org/web/packages/leaps/leaps.pdf.

Library leaps. Library for Generalized Additive Models in R. URL: https://cran.r-project.org/web/packages/gam/gam.pdf.

Library matplotlib. Library for data visualization in Python. URL: https://matplotlib.org/stable/index.html.

Library Metrics. Library for evaluation metrics in R. URL: https://cran.r-project.org/web/packages/Metrics/Metrics.pdf.

Library numpy. Library for scientific computing with Python. URL: https://numpy.org/.

Library pandas. Library for data analysis and manipulation tools in Python. URL: https://pandas.pydata.org/.

Library pls. Library for Principal Components Regression and Partial Least Squares Regression in R. URL: https://cran.r-project.org/web/packages/pls/index.html.

Library rsample. Library for creating different resampling objects in R. URL: https://rsample.tidymodels.org/.

Library sklearn. Library for machine learning algorithms in Python. URL: https://scikit-learn.org/stable/.

Library sn. Library for probability distribution of the Skew-Normal family in R. URL: https://cran.r-project.org/web/packages/sn/sn.pdf.

Library splines. Library for regression splines function and classes in R. URL: https://www.rdocumentation.org/packages/splines/versions/3.6.2.

Los Angeles weather archive. URL: https://www.ilmeteo.it/portale/archivio-meteo/Los+Angeles.

Minkowski distance. URL: https://en.wikipedia.org/wiki/Minkowski_distance.

NSRDB. NSRDB official site. URL: https://nsrdb.nrel.gov/.

NSRDB new repository. URL: https://maps.nrel.gov/nsrdb-viewer/.

NSRDB old repository. URL: https://www1.ncdc.noaa.gov/pub/data/nsrdb-solar/.

NSRDB presentation. URL: https://www.nist.gov/system/files/documents/2020/01/15/Habte.pdf.

NSRDB processing report. URL: https://www.nrel.gov/docs/fy22osti/82063.pdf.

NSRDB versions history. URL: https://nsrdb.nrel.gov/about/version-history.

Seven factors affecting UV irradiance. URL: https://www.myuv.com.au/.

Solar irradiance. URL: https://en.wikipedia.org/wiki/Solar_irradiance.

SVR. Support Vector Regressor (SVR) documentation. URL: https://cs.adelaide.edu.au/~chhshen/teaching/ML_SVR.pdf.

Tidyverse libraries collection. Collection of libraries for data science in R. URL: https://www.tidyverse.org/.

UV effects on humans. URL: http://files.cie.co.at/cie209_2014.pdf.

XGBoost. Extreme Gradient boosting (XGBoost) documentation. URL: https://xgboost.readthedocs.io/en/stable/.