Predictive techniques applied to geothermal power plants data

Oscar Fernando Cideos Nuñez

Faculty of Industrial Engineering,
Mechanical Engineering and
Computer Science
University of Iceland
2015

Predictive techniques applied to geothermal power plants data

Oscar Fernando Cideos Nuñez

60 ECTS thesis submitted in partial fulfillment of a
Magister Scientiarum degree in Mechanical Engineering

Advisor(s)
Magnús Þór Jónsson
Tómas Philip Rúnarsson

Faculty Representative
Halldór Pálsson

Faculty of Industrial Engineering, Mechanical Engineering and Computer Science
School of Engineering and Natural Sciences
University of Iceland
Reykjavik, May 2015

Predictive techniques applied to geothermal power plants data
Predictive techniques applied to geothermal power plants data
60 ECTS thesis submitted in partial fulfillment of a *Magister Scientiarum* degree in Mechanical Engineering

Faculty of Industrial Engineering, Mechanical Engineering and Computer Science
School of Engineering and Natural Sciences
University of Iceland
Dunhagi 5
107 Reykjavík, Reykjavik
Iceland

Telephone: 525 4000

Bibliographic information:
Oscar Fernando Cideos Nuñez, 2015, *Predictive techniques applied to geothermal power plants data*, Master's thesis, Faculty of Industrial Engineering, Mechanical Engineering and Computer Science, University of Iceland, pp. 64.

Printing: Háskólaprent ehf., Falkagata 2, 107 Reykjavík
Reykjavik, Iceland, May 2015

Abstract

An extensive operational database is usually present in any power plant and geothermal power plants are no exception, due to the amount of information that is constantly collected from sensors and measurement parameters during the normal operation. As time goes on power plants start becoming a unique structure due to the different components in the plant and also the added efficiencies that keep changing over the any component lifetime.

Thermodynamic models while always reliable tend to be less accurate over time, this research tries a different approach on predicting a component operation. The idea behind this research is to predict a component output (a turbine in this case) using a series on models based on all the data collected relevant to that particular component.

It is shown that a certain data processing need to be done in order to start the analysis, this data processing is mostly to adapt the algorithms to the data analyzed, otherwise the process becomes straightforward. An event prediction model based on geothermal field reports is also considered to try to determine what causes anomalous operation in the power plant.

For my mom, Fátima and Roberto

Gracias mimosines.

Table of Contents

List of Figures

List of Tables

Acknowledgements

My sincerest gratitude to the UNU-GTP program for giving me the opportunity to come to Iceland to pursue my degree. Special thanks goes to the UNU-GTP staff, the director of the program Ludvik S. Georgsson, Ingimar, Maria, Frida, Tori and Markus.

Special thanks to my supervisors Magnús Þór Jónsson and Tómas Philip Rúnarsson for their support, patience and invaluable guidance, also for all the ideas that made this research possible and their constant support throughout the research.

I want to thank my employer, LaGEO S.A. de C.V. in El Salvador for supporting me in my studies. I am deeply grateful for the opportunity and encouragement to continue with the geothermal development in El Salvador. I would like to thank in particular my boss J. L. Henriquez, and my coworkers L. Aguirre, J. Vides

I would like to thank everyone who in some way or another have contributed to this research, to the people I met here and those who helped me in the distance, I would like to thank you all, Yid, Frida, Mariela, Tecla, Vijay, Gaetan, Stephen, Xavier, Shakiru, Alfredo, Luis, Fr. Gabriel, Fr. Juan Carlos, Fr. Horacio, Martin, Nicolás, Cristina, Blanca, Hna. Cecilia, Hna. Margarita, Hna. Sabiduría, thank you all for your encouragement and friendship.

For those who understand: The thing is…

Two roads diverged in a wood, and
I took the one less traveled by,
And that has made all the difference.
-.Robert Frost

1 Introduction

Today there is an ever increasing trend in collecting data in our daily lives. With the recent advances in digital technology this has become even easier than before. The trend is to gather as much information as possible in the hope that it will eventually become useful. However, collecting data just for the sake of collecting data is not very focused. In general we would like to have a question to be answered before collecting such data. A question in turn may be answered by a model, and the model can only be tested using data. However, once a model has been formulated we know what data to collect. In a geothermal power plant the data collected is generally only analyzed after an unprecedented event or anomalous operation. In addition to the power house, there is data collection in the pipelines, wells, separating stations, etc. making it rich in information, and different from other electric power generation sources. The idea of this research is to try to make use of the data collected over the recent years for a geothermal power-plant in El Salvador and investigate if this data can be used to predict power production of the plant.

Motivation

While a general thermodynamic model can give a very good description of the plant components, there will always be some hidden assumptions made that influence its predictive power. For example, model parameters are difficult to estimate and may be time dependent. The research question of this study is, can a predictive model of a geothermal power plant component be created from operational data?

The data used comes from Berlin geothermal power plant in El Salvador, operated by LaGEO S.A. de C.V. Two different types of reports were used, the operational report and the field report. The operational report stores the data from the sensors in the power house, daily at two hour intervals. The field report stores the data from outside the powerhouse, wells data (temperature, pressure and mass flows) and operational events (scheduled maintenances, shutdowns, etc.).

There are four turbines currently operating in the Berlin geothermal field, 3 single flash units and 1 binary cycle power plant. This research focuses on Unit 1 and Unit 2, two single flash 28 MW identical turbines. All the models and analysis use only data from Unit 1 and Unit 2, the only exception is in the event analysis were there are events of Unit 1 or Unit 2 that are related to the operation of Unit 3.

During day to day operations in Berlin geothermal power plant, when expansion plans are considered or maintenance schedules are planned, a power plant model is required. When dealing with planning tasks, the thermodynamic model comes short on accuracy of prediction. A failure to predict power production accurately in the power plant, means that there can be an impact on production or project budgeting.

This research aims to be a different approach at the way data is usually processed in geothermal power plants, focusing mainly on Berlin geothermal power plant. With an accurate predictor model of a component in the power plant, the predictor model can be used

for a different set of tasks: temporary prediction in case of sensor failure, future development modeling using different scenarios, etc.

Contribution

The main contributions of this research are:

- A more accurate prediction model for power production than the thermodynamic model currently in use at Berlin geothermal field is created. The predictive model uses the latest logged data on the field. The power prediction model can be used when considering expansion or maintenance projects dealing with power production planning.
- An event analysis showing the distribution of unscheduled (anomalous) events in the power production of the plant is considered. When dealing with a dataset with high variations in power production, an event prediction can be considered.
- The data visualization applied to operational data, for exploring undetected correlations between power plant variables.
- The use of indicators to compare between a thermodynamic model and a regression model.
- To the extent of the publicly available literature, it is the first time that geothermal power plant data is studied systematically for patterns and usability for prediction in the region.

Additional contributions of this research include the in-depth analysis of field data taken from a power-plant. This includes various data visualization techniques such as, correlation plots, principal components analysis. The regression methods are then used to create predictive models. Data visualization, correlation plots and principal components analysis are used to detect any visible patterns within the data and also to check the similarity of the datasets studied.

The regression techniques include: linear regression, regression trees and random forest. These techniques are used for the prediction models of the power plant components. MATLAB was used for data processing, event analysis and also for the general thermodynamic model along with REFPROP library. R and the R data analysis toolbox "rattle" are used for the predictive models and the predictive models comparison.

This research also does an event analysis in the power plant, making a link between the operational report and the field report. This analysis associates the data measurements with any anomalous operation in the power plant (shutdowns or production decreases) by looking up events descriptions in the reports.

Overview

In the second chapter a brief geothermal power plant use in El Salvador is discussed. Geothermal development in El Salvador is described here since its beginnings in the mid-70s were the first geothermal power plant started operations in El Salvador in the state of Ahuachapán. General thermodynamic cycles currently in use in El Salvador are also described in this chapter. A general description of single flash power plants and combined cycle power plants is discussed in the second chapter.

The Berlin geothermal power plant is also discussed in chapter two. The location of the plant and the history of its development are discussed. Information of the production wells supplying steam to Unit 1 and Unit 2 is described.

The data used for this research is defined in chapter three. The origin of the data is explained briefly and the logging interval of the datasets. Two types of reports are used for this research, an operational report and a field report. The operational report is described, its function and what data inside the report is used for the models. The field report, which contains more general data about the geothermal field, is also discussed in chapter four. Data visualization methods are discussed in this chapter, including correlation plots and principal component analysis, as techniques to detect any visible patterns or similarities with the dataset used.

An event analysis is considered in chapter four. Where the un-scheduled power production variations of Unit 1 and Unit 2 are linked with the operational data. Two types of events are discussed in this chapter: shutdowns and decreased production events. The events are categorized and plotted to detect any visible patterns.

The prediction models are discussed in chapter five. The predictive models considered for this research are: a general thermodynamic model, regression trees, random forest, and linear regression models. The models are tested on performance using a different dataset than the one used for creating them.

In chapter six the predictive models are compared with the general thermodynamic model performance. Four different indicators are used as comparison tools for the models: mean absolute error, mean squared error of prediction, coefficient of model determination and modelling efficiency.

A brief discussion and future work is discussed in chapter seven.

2 Geothermal power plants overview

A brief description of the geothermal utilization in El Salvador is described in this chapter. History of geothermal development in El Salvador and its current development state are also described.

Two main types of geothermal power production cycles are also described here. Single flash, and combined cycle power plants are discussed because of their widespread use and also because these are the type being currently in use in El Salvador. Any future plans of geothermal expansion in El Salvador considers these types of geothermal power plants.

2.1 Geothermal power plants in El Salvador

El Salvador is located on the Pacific Ocean coast in Central America. The region lies on top of a tectonic subduction zone between the Caribbean and Cocos plates, also, the Caribbean and North-American plate boundary passes through Guatemala. This strong tectonic activity is one of the reasons of the prominent volcanic chain and abundance of geothermal resources (Molnar and Sykes, 1969).

El Salvador began with the exploration of geothermal resources during the 1960s along with the aid of the United Nations. El Salvador's geothermal developer is LaGeo, a subsidiary of CEL which is the national energy company.

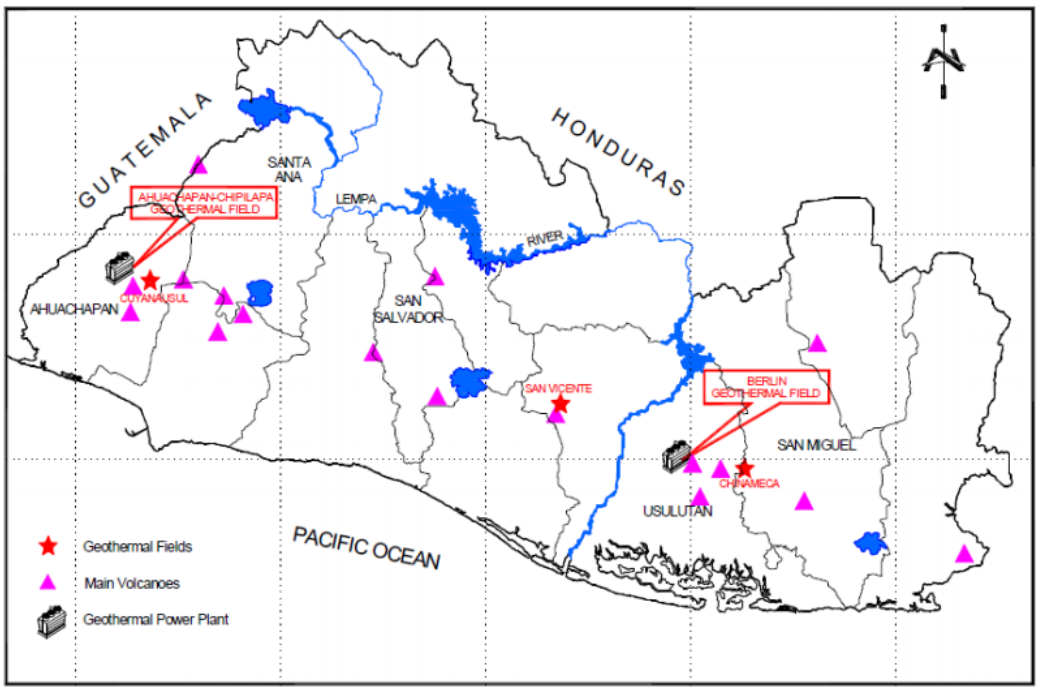

Figure 2.1: Geothermal power plants in El Salvador (JICA,2012)

El Salvador, has an area of approximately 21,000 km2 with a population of 6.3 million (2014) is the smallest country in Central America. It is the first country in the region to develop and utilize geothermal resources for electricity production.

LaGeo holds the exploratory and development rights (concessions) of four of the main geothermal fields in El Salvador, these are: Berlín, Ahuachapán, Chinameca and San Vicente. The first one is the main focus of this research, Chinameca and San Vicente are currently under development which started during the 2010s and Ahuachapán which is the oldest geothermal field on operation in El Salvador. Figure 2.1 shows the map of El Salvador, the geothermal fields being developed, the map also shows the main volcanos in the country.

Geothermal energy in El Salvador is mainly used for electrical power production, there is some potential for direct uses, such as fisheries and greenhouses, but these are not being actively researched from the utility company. Other independent contractors are pursuing direct geothermal uses in lower enthalpy fields. Electrical power production has been steadily growing over the years in El Salvador, from 400 GWh in 1995 to 1444 GWh in 2014 (UT, 2015)

Geothermal power production in El Salvador began in 1975, in the Ahuachapán geothermal field, when the first 30 MW turbine started operating. Currently, the two main geothermal fields in El Salvador, Ahuachapán and Berlín, have an installed capacity of 204 MW, 109 MW in Berlín and 95 MW in Ahuachapán. Figure 2.2 shows the evolution of energy production by source in El Salvador, blue shows hydroelectric generation, gray shows fossil fuel generation, orange is geothermal production and yellow is biomass power production.

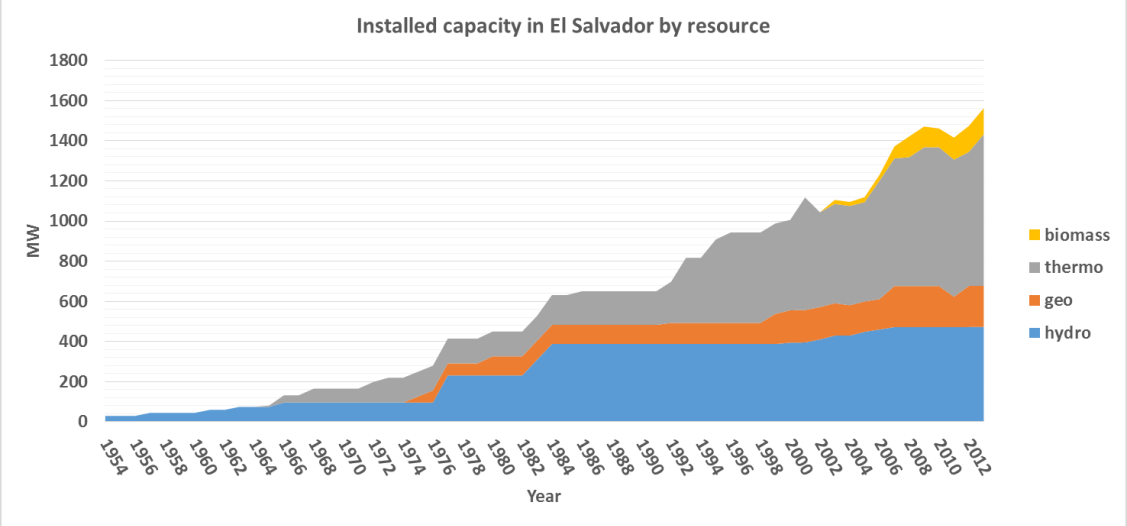

Figure 2.2: Evolution of energy sector production in El Salvador. (SIGET,2014)

With the aid of the United Nations Development Programme in 1966, El Salvador identified the first priority area for geothermal development in the region of Ahuachapán in the western part of the country, this region later developed into the Ahuachapán geothermal field. The well AH-1 was drilled 1968, the reservoir was water-dominant, with a depth between 600m

and 1500m, the exploration phase proved that the field was feasible for power production (Rodriguez and Herrera, 2003). With funding from the World Bank the Ahuachapán geothermal started in 1972. In 1975 the first single flash unit of 30 MWe was commissioned and began operation. By 1981 the third unit, a double flash turbine started operations bringing the total capacity of the power plant to 95 MWe.

The Berlín geothermal field was the second to start production with to backpressure turbines of 5 MWe each in 1992. Two single flash units, Unit 1 and Unit 2, 28 MW each, were commissioned in 1999 replacing the backpressure units. Later a third single flash unit and an organic rankine cycle (Ciclo Binario 1) were also commissioned in the geothermal field, bringing the total installed capacity of the Berlín geothermal field up to 109 MWe. (Guidos and Burgos, 2012)

Many types of geothermal power plants and geothermal fields layouts exists in the world, this research focuses on single flash power plants and combined cycle power plants. The descriptions of power plant layouts are influenced by those seen in El Salvador, and those that may be developed in the future. Even though the following descriptions are based on those developed in El Salvador the process in general is the same, and so the description can be used to understand the basic principles of these two geothermal power plant cycles.

2.2 Single flash power plants

In the single flash power plants the geothermal fluid coming from the production wells goes through only one flashing process. In the flashing process, the geothermal fluid coming from the production wells goes through a cyclone separator only once, were it is separated into steam and fluid phase. A more detailed description of the process is shown in Figure 2.3.

The geothermal fluid comes from the production well, usually a two-phase mixture, were it goes through the main well valve. The pressure of the geothermal fluid is lowered at the valve. After the main valve the fluid goes into the cyclone separator were the flashing process occurs, in here a high pressure geothermal fluid separates into a liquid and a vapor mixture, the main assumption of this process is that it occurs at a constant enthalpy. The separation process happens at a constant pressure.

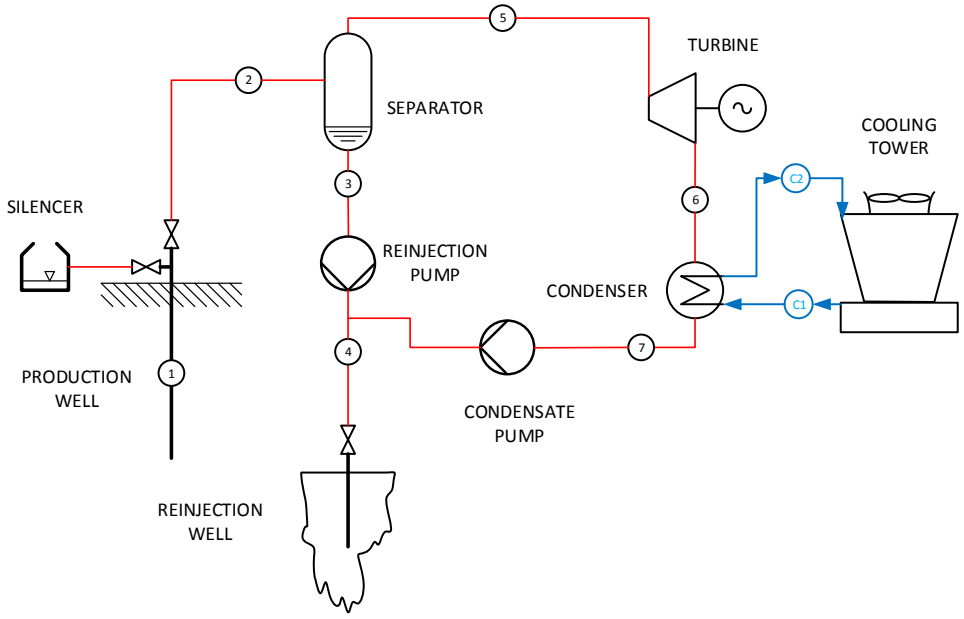

Figure 2.3: Simple single flash power plant schematic

The vapor phase of the geothermal fluid is later used in the turbine and the fluid phase is disposed. In the single flash power plants, the separated geothermal fluid phase goes directly to a reinjection well or sometimes it is disposed in some other way. In other geothermal power plants the separated geothermal fluid is used in different types of utilization process.

In Berlin geothermal field the common practice is to have a separator station in each wellpad, and each wellpad has between 2-4 wells. The geothermal fluid from the production wells is separated as close to the source as possible and then goes to the power plant through the pipelines of the steam gathering system. The separated fluid phase is sent to either the reinjection wells, the reinjection station or the Binary power plant. Figure 2.3 shows a schematic of a simple single flash power plant with a reinjection pump (stage 4).

After the separation process, all the vapor feeding the turbines travels to the power plant, were it goes to a steam collector system were the pressure of all the feeding pipelines balances and the goes to a dehumidification process.

After the dehumidifier, the vapor goes to the steam turbine (stage 5), were the energy from the vapor is transformed to mechanical work to make the turbine rotate and then produce electricity by being coupled to a generator. After the vapor goes through the turbine stages, it loses pressure and temperature, resulting in a mixture of vapor and water.

After passing through the turbine, the steam goes to the condenser (stage 6), where, as it name implies condensates the steam of the turbine outlet for later disposition. A direct contact condenser is used, in which the cooling water is carried by two pipes going next to the main body of the condenser to a distribution header on top of it, where it is sprayed over by a set of nozzles. The condensate in this case gathered and taken to a reinjection pipe and consequently to a reinjection well, the condensate can also be reused depending on the type of utilization scheme or discarded.

Another element of the geothermal power plant considered for this research are the cooling towers. Cooling towers are closely related to the condenser. The main purpose of the cooling tower is to cool down the water coming from the condenser. In order to accomplish the temperature difference of the cooling process, water coming from the condenser is sprayed from the upper part of the cooling tower (stage c2 and 7), a series of fans on top of the tower induce airflow through the cooling tower, lowering the water temperature due to heat and mass transfer between the air and water.

Figure 2.3 also shows two pumps, the condensate pump and reinjection pump. The purpose of the condensate pump is to pump the condensate water coming from the condenser outlet to the cooling tower. The reinjection pump purpose is, as shown in the figure, to pump the separated water from the flashing process to the reinjection wells.

2.3 Combined cycle power plants

In many cases, due to the diversity of geothermal resources, a single type of power plant is not the only option for a particular location but a combination of the power production techniques to take better advantage of the geothermal resource and increase the efficiency of the resource utilization.

Both geothermal power plants in El Salvador, Berlin and Ahuachapán, are combined cycle power plants. In Ahuachapán geothermal field, three turbines are currently operating, there are two single flash condensing turbines and a double flash (dual pressure) turbine. Berlin geothermal field has three single flash condensing turbines and a binary cycle (ORC) power plant using. Both power plants have used a different approach in the utilization and further optimization of the resource available.

Another example of a combined cycle power plant is the Hellisheidi geothermal power plant in Iceland were a mixture of electricity and hot water for district heating is produced.

Many different types of utilization schemes are being used around the world, not only for power production but for heating purposes, greenhouses, fisheries, balneological, leisure and many more exists, not only being restricted to power production.

2.4 Berlin geothermal power plant

2.4.1 Location and history

The Berlín geothermal field, the second geothermal field used for power production in El Salvador, is located in Usulután, a state east of the capital of the country, San Salvador. The area of the Berlín geothermal field is approximately 8 square kilometers (km^2), the well depth varies around 500 and 3,455 meters. The field is located in the Tecapa Berlín volcanic complex, in its north-northwestern running zone and within a NNW-SSE trending graben structure. (Hernandez Murga, 2012)

During the late 70s and early 80s six deep wells were drilled in the field, TR1, TR2, TR3, TR4, TR5 and TR9 which confirmed the existence of a geothermal reservoir able to produce energy for commercial distribution. In 1992 the energy authority of El Salvador, Comisión Ejecutiva Hidroeléctrica del Río Lempa (CEL), commissioned two backpressure units of 5 MWe, were steam from wells TR2 and TR9 was used to power the turbines, the separated fluid phase was sent for reinjection. (Henriquez Miranda, 1997)

The backpressure units were decommissioned in 1999, when two 28 MW condensing units were installed, a third condensing unit of 44 MW was commissioned in 2007 and finally a binary cycle power plant of 9 MW was commissioned increasing the installed capacity of the power to 109 MWe.

The main focus of this research is Unit 1 and Unit 2, this two condensing units were commissioned in 1999. Unit 1 and Unit 2 are twin turbines, the two units share the same set of sensors.

The Berlín geothermal field is a liquid dominated system with temperatures ranging from 206°C to 300°C measured in the production wells. As of 2012, 38 wells had been drilled in Berlín geothermal field, 16 of those were used for production and 18 for reinjection. Shown in Figure 2.4 is Berlín geothermal field wellpads power plant and main geological features.

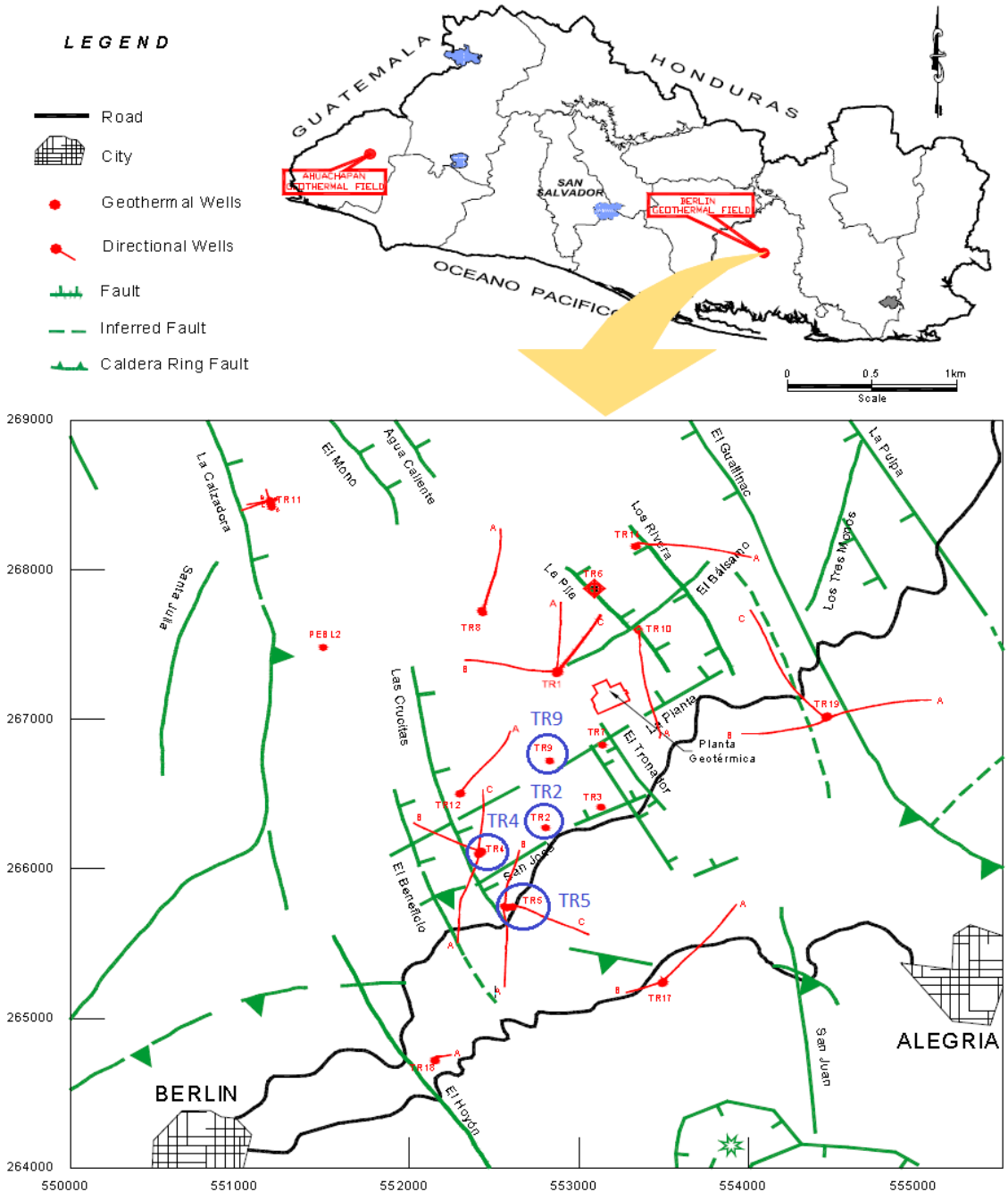

Figure 2.4: Berlin geothermal field (modified from Hernandez Murga, 2012)

2.4.2 Wells

Currently there are 9 production wells feeding units 1 and 2 of the Berlín geothermal field, shown inFigure 2.4. These wells are distributed in four different wellpads in the southern part of the field. The separated water from this process is used in the binary cycle heat exchangers and then take to 11 reinjection wells in the northern part of the field. The well names and details are described in Table 2.1.

Table 2.1 Production wells feeding units 1 and 2

Name	Condition - deviation	Elevation (m.a.s.l.)	Depth (m)
TR-2	VERTICAL	752	1903
TR-9	VERTICAL	649	2298
TR-4B	N-56-W	767	2292
TR-4C	N-05-W	767	2179
TR-4	VERTICAL	767	2379
TR-5A	S-03-W	840	2325
TR-5B	N-17-E	840	2097
TR-5C	S-70-E	840	2343
TR-5	VERTICAL	853	2086

2.5 Summary

While in El Salvador a big part of the energy production is currently relying on fossil fuel production, the current focus of the energy authority is to direct all the efforts into geothermal power. El Salvador, with the years of experience gathered on geothermal development, will increase its installed capacity of single flash and combined cycle geothermal power plants.

This chapter offered a brief description of the geothermal development of El Salvador and the thermodynamic cycles that are currently in operation in Ahuachapan and Berlin gothermal power plants in El Salvador.

The Berlin geothermal field is in constant development, many changes are underway to become a reality (new production and reinjection wells, etc.).

3 Data description

The data provided for this research comes from LaGeo S.A de C.V in El Salvador, specifically from Berlín geothermal power plant. Due to data and information policies in the company, not all the data used to build the models can be published, hence some of the data may be slightly adjusted, and a short clarification will be written anytime this is done.

Two types of reports are used for this research form the Berlín geothermal power plant. The operational daily report and the field (wellpads/wells) daily report.

R and Matlab are used for the analysis of the data and particular code was created on each software for the processing and analysis. Rattle (R package) is used in a different stages of the research for the predictive models part.

3.1 Report structure

3.1.1 Operational report

As stated above, the data used comes from two reports in the Berlín geothermal field. The reports are sent daily to the relevant staff of the company. The reports were procured by the staff of the company through a cloud storage account due to the large size of the data package. Two years of reports are used in this research, the reports from 2011 and 2012.

Sala de control			DATOS DE OPERACIÓN DE UNIDAD No 1 (1/2)											
MEDICION	Rotulo Nº	Unidad	01:00	03:00	05:00	07:00	09:00	11:00	13:00	15:00	17:00	19:00	21:00	23:00	PROMEDIO
CARGA	EL-1	MW	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.0
VELOCIDAD DE LA TURBINA	ST-1633	R.P.M.	86	86	86	86	86	86	86	86	86	86	86	86	2.0
FLUJO DE VAPOR DE ENTRADA A LA TURBINA	FI-1005	T/H	3.94	4.19	4.05	4.2	4.1164	4.28	4.2	4.14	4.2	4.1	4.1562	4.06	0.1
PRESION DE FLUJO DE ENTRADA DE VAPOR A LA TURBINA	PT-1010	BARG	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.0
TEMPERATURA DE VAPOR DE ENTRADA A TURBINA	TE-1009	°C	4.3	4.3	4.3	4.3	4.3	4.3	4.3	4.3	4.3	4.3	4.3	4.3	0.1
PRES. DE VAPOR ENTRA A CAMARA DE TURBINA	PT-1011	BARG	0.18	0.19	0.18	0.19	0.18	0.19	0.19	0.19	0.19	0.19	0.19	0.18	0.0
PRES. DE VAPOR DE SELLO PRENSA ESTOPA DE TURBINA	PT-1014	BARG	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.0
PRES. DE ENTRADA DE VAPOR DE SELLO A EYECTOR	PT1019	BARG	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.0
PRESION DE VACIO DEL CONDENSADOR	PT-1012	BARG	0.002	0.002	0.002	0.002	0.002	0.003	0.003	0.003	0.003	0.003	0.003	0.002	0.000
TEMPERATURA DEL CONDENSADOR	TE-1040	°C	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	0.0
NIVEL DEL CONDENSADOR	LT-1037	MM	3.1	1.4	2.8	2.1	2.7	2.0	2.6	1.2	2.7	2.2	2.7	1.3	0.1
POSICIÓN DE VÁLV. DE NIVEL DEL CONDENSADOR	LCV-1036A	%	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
POSICIÓN DE VÁLV. DE NIVEL DEL CONDENSADOR	LCV-1036B	%	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
CORRIENTE DE BOMBA A	AMT13	AMP	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	0.0
CORRIENTE DE BOMBA B	AMT14	AMP	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	0.0
FLUJO DE AGUA DE ENFRIAMIENTO	FE-1008	T/H	141	141	143	141	144	144	144	140	141	143	142	145	3.4
TEMPERATURA DE ENTRADA AGUA DE ENFRIAMIENTO	TE-1008	°C	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.0
TEMPERA DE SALIDA DE AGUA DE ENFRIAMIENTO	TE-1018	°C	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.0
POSICION DE SETEO DE VELOC. DE TURBINA		RPM	89	89	89	89	89	89	89	89	89	89	89	89	2.1
POSOCION DEL LIMITADOR DE CARGA DE LA TURBINA		%	2	2	2	2	2	2	2	2	2	2	2	2	0.1
PRESION DE ACEITE DE CONTROL	PT-1602	BARG	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.0
PRESION DE ACEITE DE LUBRICACION	PT-1613	BARG	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.0
POSICION DE VALVULA DE CONTROL (IZQUIERDA)	CV-1630	%	1	1	1	1	1	1	1	1	1	0.643	1	1	0.0
POSICION DE VALVULA DE CONTROL (DERECHA)	CV-1631	%	1	1	1	1	1	1	1	0.5	1	0.548	1	0	0.0
TEMPERATURA DE ACEITE DE SALIDA DEL ENFRIADOR	TE-1610	°C	1.0	1.1	1.0	1.0	1.0	1.0	1.1	1.1	1.0	1.0	1.0	1.0	0.0

Figure 3.1: Screenshot of a part of an operational report (data in picture altered)

The data is compiled daily at the end of the day in an Excel workbook, each sheet in the workbook contains different type of information, for this research the information comes from Unit 1 and Unit 2, as shown in Figure 3.1.

The information comes from the control room of the power plant. Each parameter is logged into the report every two hours starting from 01:00 and finishing at 23:00. When a parameter is not logged into the report, the cause can be a fault in the sensor or a unit blackout.

Units 1 and 2 share the same logged information and sensor distribution, both units are also logged in the same worksheet of the report, which speeds up the processing of the data into Matlab.

The data is stored with a short description of the logged parameter, the name of the sensor in the power plant which gets the data, the units in which the data is stored, the stored values, and a final column storing the average of the measurement. The average column uses the excel function for calculating averages which includes only the cells with a value. There may be a case were a cell may be left blank, which can be related to the causes stated above for parameters not logged.

3.1.2 Field report

The field report is also distributed daily in Excel format. The report is sent to all relevant personnel of the company. The structure and format of the data reported is significantly different from the one in the operational report.

Fecha	09-mar-15		Horas de operación	PRESION DE RESERVORIO		CONDICION POZOS PRODUCTORES				
Datos actualizados al	06-mar-15									
Potencia Bruta: U1	0.01	MW	24.00	Fecha	P TR-3 (bar g)	Pozo	F vapor (kg/s)	F Agua (kg/s)	Presion sep. (bar g)	WHP (bar g)
Potencia Neta: U1	0.59	MW		1/mar/2015		TR-2	0.00	0.00	0.21	Cerrado
Flujo vapor :U1	1.21	kg/s		2/mar/2015		TR-9	0.21	0.87		0.22
Potencia Bruta: U2	0.57	MW	24.00	3/mar/2015		TR-4B	0.28	0.81	0.23	0.27
Potencia Neta: U2	0.53	MW		4/mar/2015		TR-4C	**0.28**	1.06		**0.27**
Flujo vapor :U2	1.08	kg/s		5/mar/2015		TR-4	0.28	0.84	**0.22**	0.25

Figure 3.2: Field report structure (data in picture altered)

The report is divided in several spreadsheets reflecting different type of information. This report also includes a detailed description of routine and non-routine operations in the field and the power plant, during the current month of operation, this feature comes useful later in this research for the "event analysis".

The main difference between this report and the operational report is the interval at which the data is stored. In this report data is stored on a daily average basis, whereas in the operational was every two hours. The different data contained in both reports is used combined for the event analysis part, to correlate anomalous operation with routine or non-routine operations in the plant.

3.2 Data processing

The first step to start processing the information was the acquisition of the data into Matlab, in order to do this, a function was create to open an instance of Excel and convert each spreadsheet in the workbook into a comma separated file (.csv). The conversion of each spreadsheet is done for two reasons, it speeds up the information reading process in a later stage of the research and also allows a timeless compatibility of this information with any computer, which can be a problem in the future with excel files.

The function and the data acquisition process is designed for this particular dataset but can be adapted to another dataset or data structure.

The next step of the function is to store the information into a *cell array* in Matlab, which is basically a single variable that can store many matrices and with any type of information, numeric or strings. This is particularly useful when storing the spreadsheets, as each of them were treated as a single matrix inside the main one(cell array).

Storing all the information in a variable into Matlab allowed to speed up the process later on, when any manipulation is required. During the process of converting the data and storing it, each file and matrix was named according to a previous convention defined by the researcher. The naming of each individual matrix inside the cell array allows to easy identification later on.

Several attempts were done in order to use Matlab's built in functions to get the data from excel, but due to the large amount of information and variability of data available, a code was created to improve the processing time. Table 3.1 shows the pseudo code for the data acquisition Matlab function.

Table 3.1: Pseudo code for the data acquisition algorithm

Function "xls2matlab"

"Initialize general variables and processes"
- Initialize Excel Active server
- Set initial and final dates of the database in number format(d1, d2)
- Create a list of the date length of the database (dates)
- Create a cell array with the size of the dates vector (M3)
- Declare function to locate NA values in a cell array (Fx)
- Declare the location of the folder with the data (myfolder)
- List the contents of the folder (subFolder1)

"Extract the data into the cell array"
- FOR 1 until the size of subFolder1
  - List the contents of the subfolder (files)
  - FOR 1 until the size of files
    - Get the name of the file (currentFile)
    - Get the date of the file (currentDate)
    - Convert the current date to number format (currentDateNo)
    - Locate the date number in the M3 index
    - Get the number of spreadsheets in the current file (Sheets)
    - FOR 1 until the size of Sheets
      - Get the contents of the current sheet (Range)

```
                    Set the content of the current sheet in the cell array
                    IF the current file is the first one read
                        Create list with the spreadsheets in the file (sheetsNames)
                        ELSE
                        IF the current spreadsheet name is new
                            Assign the new name to the list
                            ELSE
                            Store the data in its position in the cell array (M3)
                        ENDIF
                    ENDIF
                ENDFOR
            ENDFOR
        ENDFOR
```

The final output of this function is the M3 cell array, this array contains all the information from the Excel spreadsheets. After storing all the data into Matlab, looking up for it is done quicker than calling it from the source, as the information is readily available in the RAM of the computer and is also indexed by Matlab.

Table 3.2: Pseudo code for sensor data extraction

```
function "getdata"
"Get the function inputs"
    Get data matrix cell array (M3)
    Get the datasheet where the required data is stored (dataSheet)
    Get the name of the sensor from which the data is required (sensorName)
    Get the number of the turbine from which the data is required (Unit)
    Get the starting date for the data to be acquired (startDate)
    Get the last date for the data to be acquired (endDate)
"Initialize general variables"
    Set the variable to remove peaks from data (default is peaks removed)
(clean)
    Get the data position based on variable Unit (X,Y)
    Set list of available sheet names (sheetsNames)
    Create date list in number format from startDate to endDate (dates)
    Declare function to locate NA values in a cell array (Fx)
"Set positions where the requested data is stored"
    Set the initial position for the lookup loop (initialPos)
    Set the last position for the lookup loop (lastPos)
    Compare dataSheet and sheetsNames to set the position of data (ind)
    Initialize the output variable with the size of dates (T)
"Loop to get the requested data"
    FOR initialPos until lastPos
        Read M3 from ind and the current loop
        Get the position of the requested data based on sensor Name,X and
Y
        Convert the data to number format and store it on T
    ENDFOR
```

```
"Remove peaks from data"
    IF clean is equal to 1
        Set the peak treshold
        Change values higher than the treshold in T
    ENDIF
```

The *getdata* function allows to gather data from the M variable, any single parameter, any time frame, and save it into a single variable (T) in Matlab for later processing. This function was initially created for getting the power parameter on any of the turbines and was later adapted to gather data from any parameter.

The function cannot distinguish between the shutdowns or just missing sensor data, that search is done at a later stage and with a different function.

The *xls2matlab* and *getdata* are starting points to create the databases for later processing in Matlab and R, other functions and algorithms were created for different tasks during this research, especially those involving the modelling. Using these two functions, the two years of daily reports are stored for later in the research.

3.3 Data structure

As stated before, this study will focus in two years of data from units 1 and 2 of Berlín geothermal field. A detailed description of the data will follow and visual methods will be used to see it distribution and detect any discernable patterns in it.

Table 3.3 Variables used in the research

Description	**Units**	**Variable name Unit 1**	**Variable name Unit 2**
Turbine output	MW	W1	W2
Turbine steam mass flow at the inlet	Tons/hr	m1	m2
Turbine steam inlet pressure	bar	P_in1	P_in2
Turbine inlet temperature	°C	T_in1	T_in2
Turbine chamber inlet pressure	bar	Pch_in1	Pch_in2
Condenser pressure	bar	Pc1	Pc2
Condenser temperature	°C	Tc1	Tc2
Condenser water level	mm	CL1	CL2
Cooling water mass flow	Tons/hr	mc1	mc2
Cooling water inlet temperature	°C	Tcool_in1	Tcool_in2
Cooling water outlet temperature	°C	Tcool_out1	Tcool_out2
Left control valve position	%	CV_left1	CV_left2
Right control valve position	%	CV_right1	CV_right2

Given that units 1 and 2 share the same design, components and components distribution, the same variables are going to be used to describe both turbines from the operational report, 13 variables are used to described de geothermal power plant as shown in Table 3.3 and these variables are used to get a first look at how the data is distributed.

The variable names shown in Table 3.3 are the same variable names stored in Matlab. For each variable 8760 data points are used, for all 13 descriptive variables becomes 113880 data points. The first approach taken to analyze the data are data visualization methods.

3.4 Data visualization

Data visualization methods give the user an insight of the data and data structure. The first approach taken in this research is to plot a sensor measurement raw data over the selected study period, as shown in Figure 3.3.

It is interesting to see the peaks in the data and going further into the information available, these peaks are physically impossible inside the turbine and are anomalous measurements of the data. If not treated correctly, these peaks can unknowingly influence any model attempted to create, and this type of anomalous measurement only increases it impact with every order of magnitude.

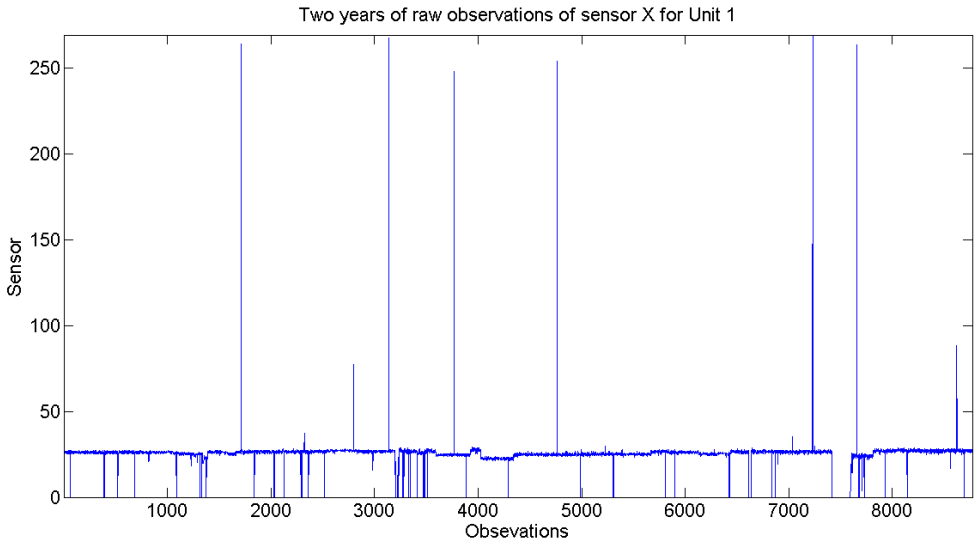

Figure 3.3: Graph of two years of data for a particular sensor (sensor name not shown)

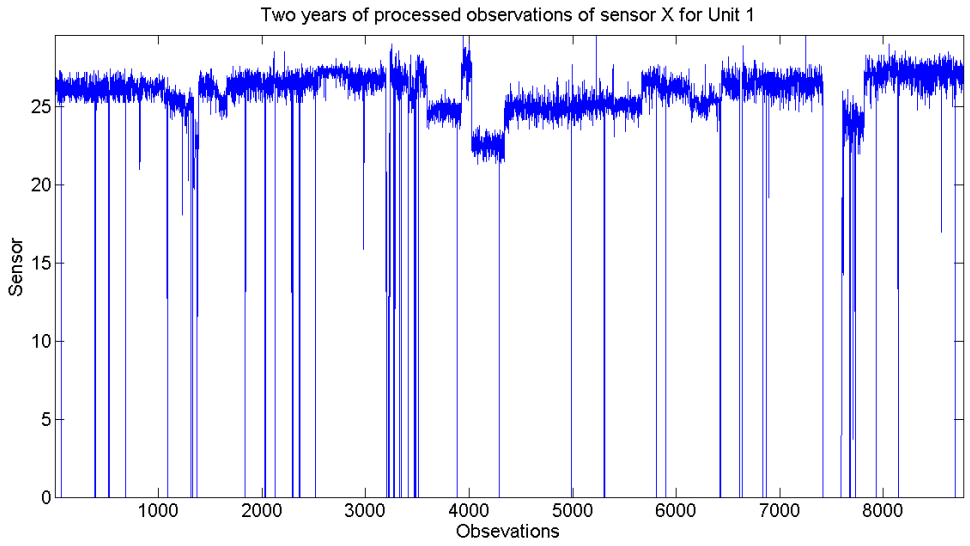

Figure 3.4: Graph of two years of data, annomalous data has been processed. (observations every two hours)

In order to clean the data from these peaks, given the large amount of samples, a simple linear regression was applied to each of this points which were singular in nature, with this method the integrity of the datasets was assured as no other data points were touched and the algorithm also looked for only single occurrences. This algorithm is implemented in the *getdata* function described before, and can be disabled in case an intact data structure wants to be used. The algorithm detects were there are occurrences of peaks of data (8 in the case of the sensor in Figure 3.3), then creates a linear regression using the following and the previous point of this occurrence, then replaces the peak value with the new calculated value.

Figure 3.4 shows the same data as Figure 3.3 but on the latter figure it has processed from the data peaks. While in Figure 3.3 all the information looks stable and with no significant changes, the same data looks less stable and with more variability than it could have initially appeared.

The Vertical lines shown in Figure 3.4 and Figure 3.5 are shutdowns of the turbine, these are detected when there are no readings in all the sensors of the turbine by the algorithm.

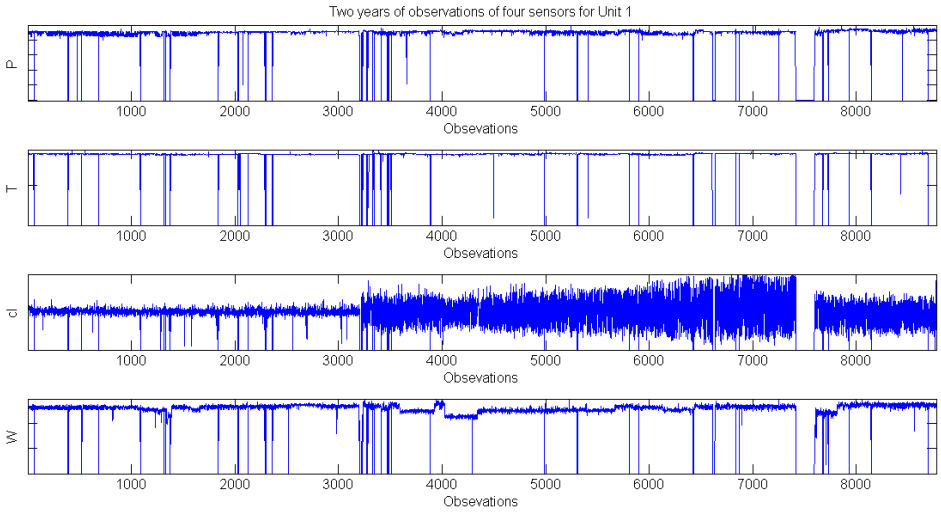

Figure 3.5: Pressure, temperature, condenser level and power output for Unit 1in two years (scale not shown)

This type of behavior in the measurements is expected in the dataset, the vertical lines in the figures show a sudden drop in the sensor measurement, which means that the turbine stopped operating, this can be due to sudden unexpected stops or planned stops in the maintenance schedule.

In order to detect shutdowns in the turbine, the data is compared using an algorithm that will check if there are no readings in the turbine and then call the field report in order to get the operations description of the particular day.

Figure 3.5 shows four measurements of the same turbine and the same period, and while the scale is hidden it shows that there is a relation between the data, but this would be very hard to deduct from looking at figures. The following chapters will start going further into this relations.

3.5 Data patterns

After the visualization methods, the next step taken was to detect any correlation between the variables in the datasets of each turbine.

Two different distinctions are important here, further on in this research when dealing with the whole dataset including the shutdowns of the turbine the data will be called unprocessed dataset, while the data only considering continuous operation will be called processed dataset.

Both Figure 3.6 and Figure 3.7 show the whole correlation matrices for the unprocessed dataset of Units 1 and 2. The 13 x 13 matrix, shows how all the variables of the dataset relate to each other, the diagonal of the matrix, show how the data for that particular variable is distributed.

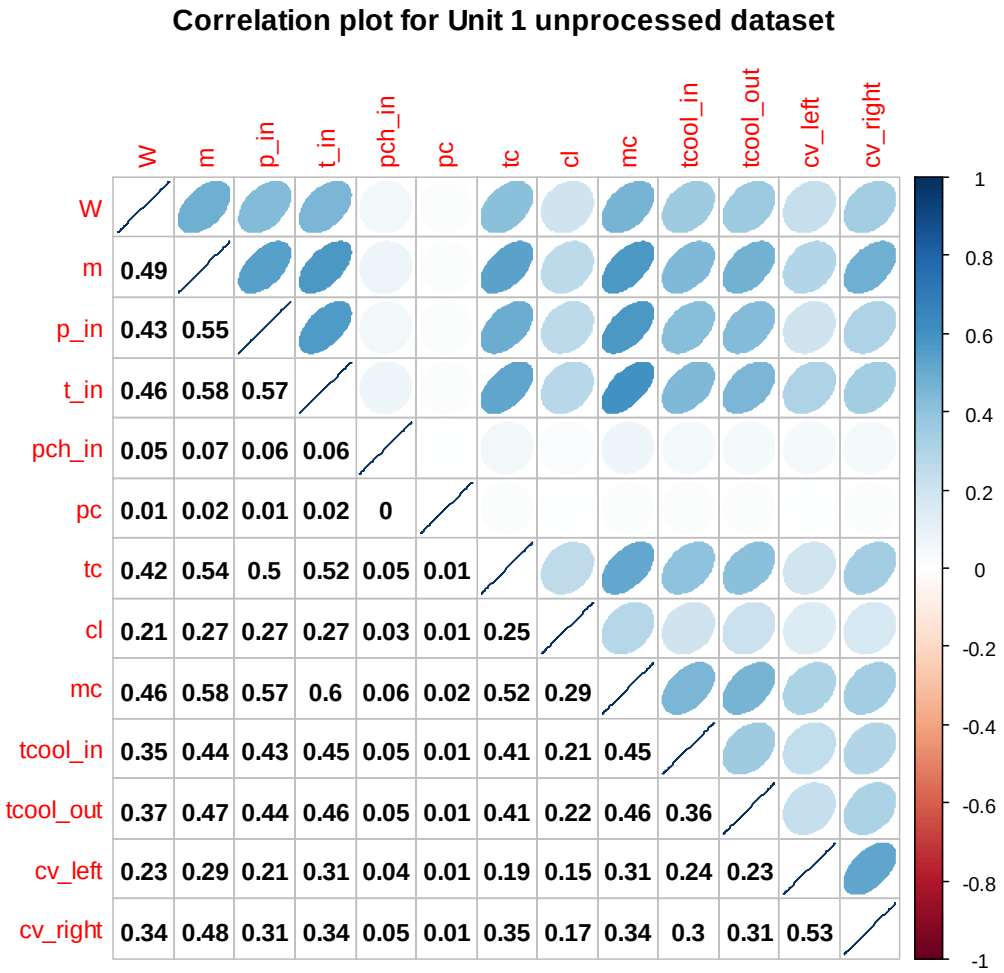

Figure 3.6 Data correlation for Unit 1 with unprocessed data

The upper off-diagonal shows the correlation of any two variables, with 1 being a blue ellipse with a positive slope and -1 being a red ellipse with negative slope. The lower off-diagonal of the matrix shows the Pearson correlation coefficient, which is calculated by the equation 3.1, which will give a value from +1 to -1, were 1 is a total positive correlation, 0 is no correlation and -1 is total negative correlation.

$$\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X * \sigma_Y} \qquad 3.1$$

Where

ρ is the correlation between variables X and Y, *cov* is the covariance between variables X and Y as shown in equation 3.2, σ is the standard deviation

$$cov_{X,Y} = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{(n-1)} \qquad 3.2$$

Where *cov* is the covariance between variables X and Y, X_i and Y_i are the individual observations, $\bar{X}$ and $\bar{Y}$ are the mean values of X and Y observations, *n* is the total number of observations

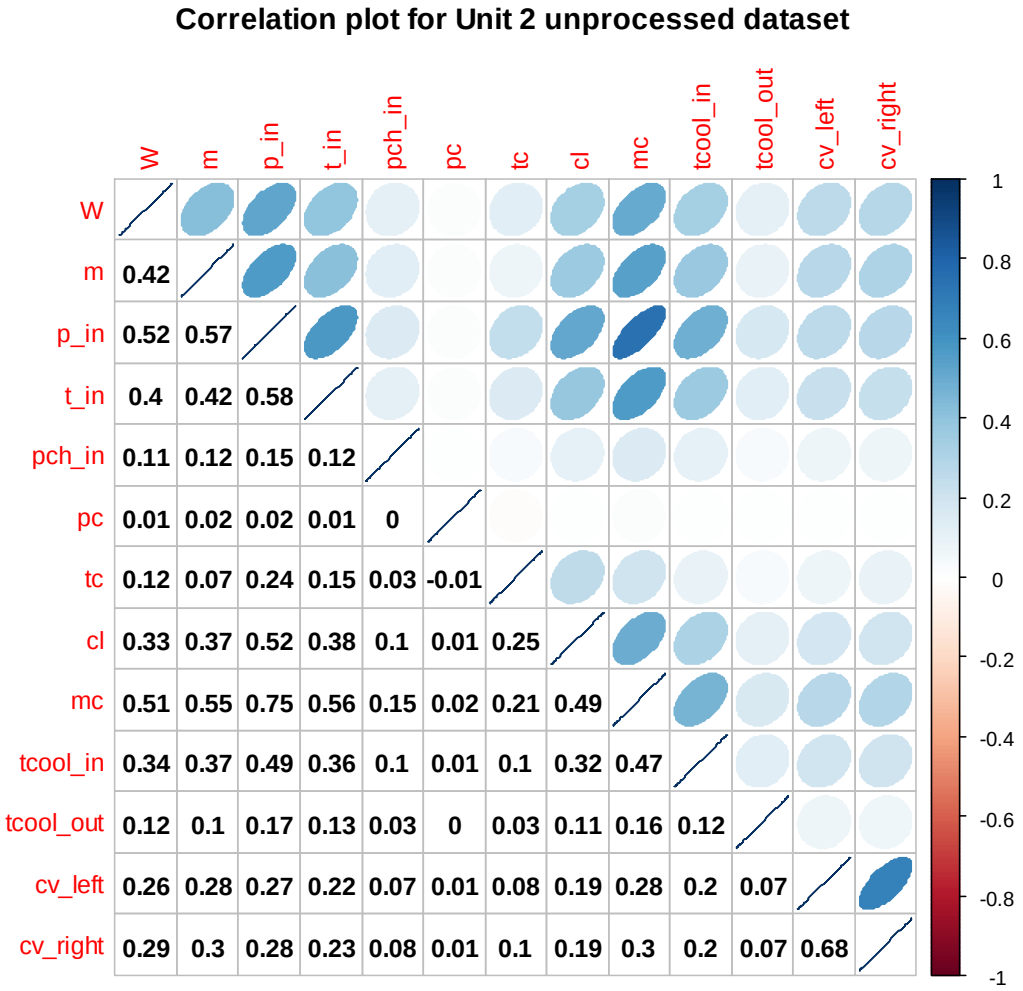

Figure 3.7 Data correlation for Unit 2 using unprocessed data

Comparing both matrices shows that almost all the variables correlate to each other. The data is interesting and shows a logical relation in a power plant, but from a data analysis point of view, offers little to no help in case a very descriptive data model wants to be created. Showing that everything is correlated, can lead to erroneous assumptions. The strongest correlation of two variables being the control valves position.

The most important information from both matrices is that all the variables appear to be positively correlated, and also that some of the variables, like condenser temperature in Unit 2 have a neutral correlation to almost all of the variables.

The data used for both of those matrices is the unprocessed dataset, which means that is a compilation of two years of operation for both turbines, it includes the data points when the turbines where stopped, and the correlation is then affected by the "unknown" amount of zeros in the dataset.

Assuming the data correlation wants to be checked only when the turbine is operating, the next step is to remove the data observations when the turbine was stopped to better understand what is happening.

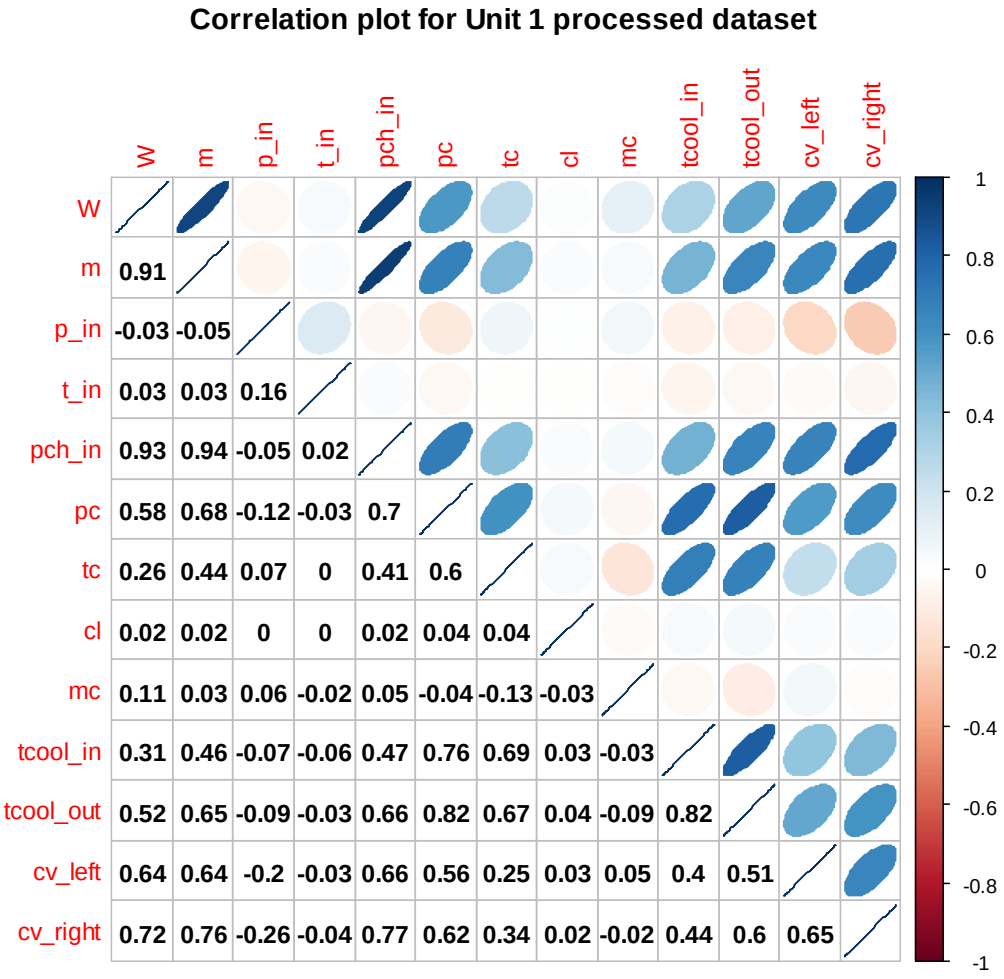

Figure 3.8 Data correlation for Unit 1 using processed data

Figure 3.8 and Figure 3.9 show the correlation matrix for the processed database, is this matrices the correlations are clearer, and not all the variables have a positive correlation, some of them showing negative correlations.

It is clear in both turbines the strong correlation between the power of the turbine (W) and the steam mass flow at the inlet (m). The same for the left and right control valves in the turbines, both show a positive correlation to power (W) and mass flow (m) but a negative correlation to the pressure at the inlet (p_in).

The rest of the variables while showing less correlation weight in the matrix, are still important for the analysis and will still be used for future analysis.

The correlation analysis goal is to see if the variables in both datasets are behaving similarly, this could mean that for Unit 1 and 2 (twin turbines) a general predictive model may be created. Another tool for checking the data similarity between both Unit 1 and Unit 2 datasets is used next.

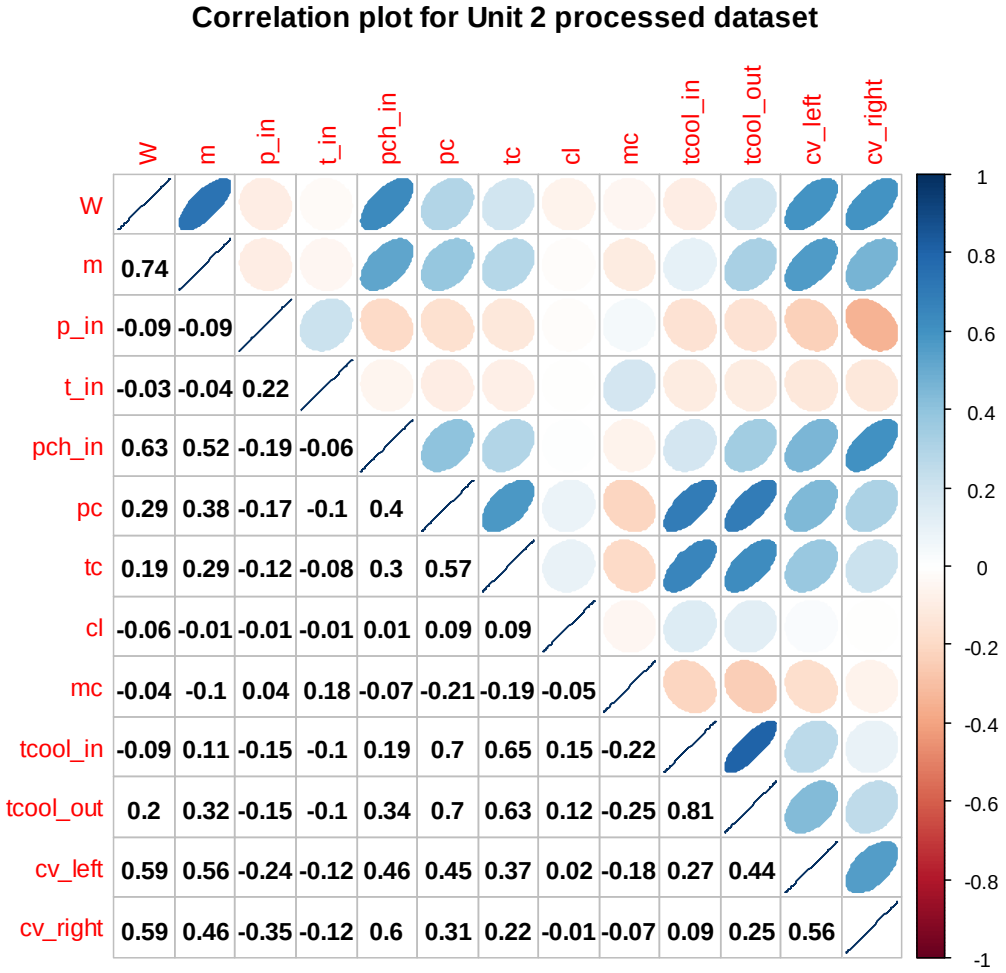

Figure 3.9 Data correlation for Unit 2 using processed data

The next tool used for the data analysis is Principal Component Analysis (PCA), PCA is a statistical tool to detect patterns in data, it is useful when datasets have many explanatory variables and a useful visual representation is very hard to come by. This is done by reducing the whole dataset, to a dataset consisting of two variables (the principal components) and a set of vectors of the original variables projected over the transformed data.

The idea behind PCA is that, when a dataset has many variables, it is possible to reduce the dataset in order to have just two variables without much loss of information, these two variables are the principal components. The principal components are an eigenvalue

decomposition of the covariance matrix, one dataset will have as many principal components as variables are present in the dataset.

In order to compare all the variables in the dataset, the standardized z-score values of each observation was calculated. When having a dataset with different units it is recommended to apply the PCA to the z-score values of the observations in the dataset. A similar process was applied for the PCA analysis than the one done to the correlation matrix, the data was analyzed discarding the observations when there was a shutdown in the turbines.

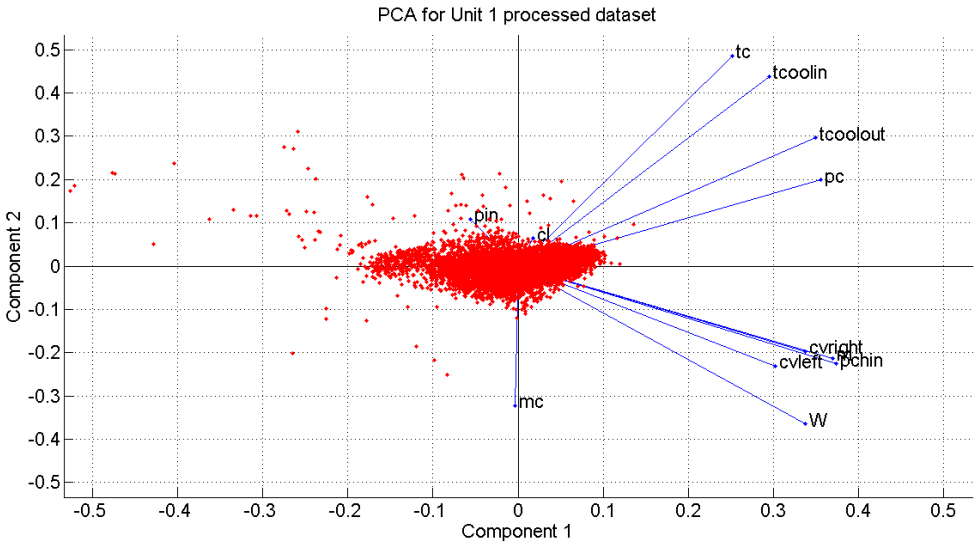

Figure 3.10 Principal component analysis for Unit 1 with processed data

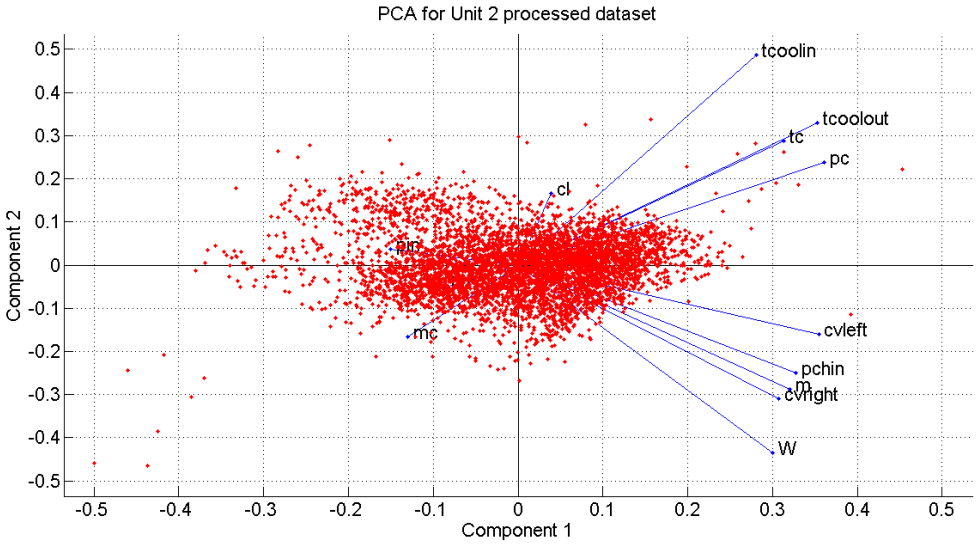

Figure 3.11 Principal component analysis for Unit 2 with processed data

Figure 3.10 and Figure 3.11 show the PCA for the processed datasets for Units 1 and 2 show a better explanation of how the data is distributed. These PCA representations of the databases are very different from each other, the data of Unit 1 being more tightly distributed for Unit 1 than Unit 2, vector distribution is similar but not identical for the variables for both datasets, a unified predictive model should not be created.

P_in, *mc* and *cl* are grouped very close to the origin of both datasets, and behave differently for both turbines, showing that these interactions are not the same for both turbines and almost not represented in the PCA.

3.6 Summary

When dealing with large datasets it is important to take into account the way the observations are provided. A standardized method for storing the data helps to understand better the dataset.

It is suggested some background knowledge of the data being analyzed, in this case the thermodynamic interactions, in the way that the expected interactions in the variables are all according to the physical interactions.

Data visualization techniques are a good first approach when analyzing large datasets, these can give an idea on the variability of the data and some basic correlations between variables.

The idea behind the correlation plots and the PCA for both units is to discover how similar the datasets were, in general it shows that the basic correlations are the same for both datasets, but a general predictive model that will be able to describe both datasets wouldn’t be recommended.

4 Events analysis

This part of the research focus on studying the database in a different manner than what has been previously done, by linking the information of the operational report and the field report. The main focus is now to analyze the operation of the turbine and link it to its data. The field report will now be used to correlate the events to any relevant data that could be registered on the entirety of the dataset.

A graphical method was used as the first approach to discern any similarities between the data preceding events. The *getdata* function comes is used for this part of the research in order to speed the data analysis process.

4.1 Daily data report

The daily information of the power plant operation is stored in the daily field report, which is a report stored in Microsoft Excel format sent daily to the relevant personnel. This report has also the characteristic that it includes any additional data that can be useful to describe any out of the ordinary operations in the geothermal field, this characteristic is not present in the operational report.

A new data acquisition function was created to sort the data from the daily reports to improve the data collection later on.

The field report data was compiled in a database similar to the one used in the operational report. The algorithm used for this purpose is the *"xls2matlab"* describe previously in this document, the only changes made for this were, the data folder and the destination cell array to N3.

After been determining what is considered to be an event, an algorithm was created to acquire the data related and leading to an event. This algorithm uses the *getdata* function previously described.

4.2 Event identification

In order to identify the events in the database, it was determined what was considered to be an event. For this research an event is considered to be any anomalous operation the turbine can show, this particular definition of event will only concern for this part of the research document. Events are divided in two categories: shutdowns and decreased production. Shutdowns are when there is no power production and there are no sensor readings. Decreased production events are when the turbine is producing less than a threshold values, 80% of the mean production during continuous operation.

The first approach to analyze the data was to detect the location of any event in the database and then to get all the relevant data related to that particular event. The data selected was the one used for the creation of the data prediction models together with the variables involved

in said models. A Matlab function was created for this purpose, making use of the improved databases M3 and N3 and function *"getdata"*.

Table 4.1: Pseudo code for event identification algorithm

```
function "eMatrix"
"Get the function inputs"
    Get the operational matrix (M3)
    Get the field matrix (N3)
    Get the list of data sheets in M3 (sheetsNames)
    Get the number of the turbine for which the events are requested (Unit)
"Initialize general variables"
    Set the initial date (startDate)
    Set the last date (endDate)
"Create the data matrix for event detection"
    Call the 13 variables involved in the data models...
        'W' , 'm', 'p_in', 't_in', 'pch_in', 'pc', 'tc',...
        'cl', 'mc', 'tcool_in', 'tcool_out', 'cv_left', 'cv_right'
    Store variables in a data matrix (datap1)
"Create event matrix"
    IF power goes below treshold
        store data in matrix (groups)
    ENDIF
    Locate events in datap1 matrix
    Get the position and duration of each event an store it (Events)
    Initialize the event matrix with the size of Events (eMatrix)
    FOR 1 until the size of Events
        Store the events data in eMatrix
    ENDFOR
```

While the variable eMatrix is called a matrix in reality is a cell array, similar to M3 and N3. It stores data from the field and operational reports, making it easy to correlate the data leading to an event. It is important to note, that this definition of event and the detection criteria was solely based on the power production of the turbine, but changes can be done to the algorithm to adapt it to a different criteria.

The function returns the *eMatrix variable*, a cell array containing the information related to all the events found in turbine and 24 measurements leading to that particular event, the cell array also contains data from the field report observations from the day of the event, the previous day and the following day.

After locating the events a plot was made to try to find any discernable patterns in the date leading to any type of event, shutdowns or a decrease in power production. Plotting the dataset proved unsuccessful for this purpose, no clear signs were shown on both datasets.

The next step taken was to analyze the cause for any of the events that could be registered in the field report database.

Table 4.2 Excerpt from the complete event table for Unit 1 created with the "eMatrix" function (descriptions translated from spanish)

Date	Type of event	Duration of event	Description
'06-Jan-2011 09:00:00'	S	1	Unit 1, shutdown at 08:37 due to failure in PC1 (Condenser or pressure at the condenser) Goes back online at 10:37
'02-Feb-2011 11:00:00'	S	3	No data
'13-Feb-2011 11:00:00'	S	7	Unit 1, shutdown at 11:05-21:56, due to failure in excitation diodes, Unit 1 shutdown 22:01, high position of the shaft
'26-Feb-2011 23:00:00'	S	1	Shutdown in Unit 3
'10-Mar-2011 09:00:00'	D	2	Unit 1, Restricted wells: TR4 restricted to 15 cm of stem, TR5 fully open
'01-Apr-2011 15:00:00'	S	1	Shutdown in Unit 3 at 14:28, failure in after condenser pumps
'13-Apr-2011 11:00:00'	D	6	Unit 1, Restricted wells: TR4s and TR5s restricted to 7 cm of stem
'20-Apr-2011 07:00:00'	S	1	Shutdown in Unit 3 at 06:27, cause: failure in analogue modules due to strong variation in national system
'21-Apr-2011 17:00:00'	S	1	Shutdown in Unit 3 at 11:36-17:31, valve W-303 failure
'22-Apr-2011 19:00:00'	D	12	Unit 1, Restricted wells: TR4s and TR5s restricted to 12 and 9 cm of stem

Table 4.2 was created using the data from the *eMatrix*, directly linked to the location of events in the dataset. The source data which had to be analyzed and translated is depicted partly in the table. The table shows four columns, the time of the event, the type of event S for shutdown or a D for a decrease in production, the duration of the event in data points observed for that particular event and the event description.

The duration of an event can fail to show the actual duration of an event unless it is specifically described in the field report, and also and event can fail to be logged in the operational report unless it is happening during the time the data is logged in the operational report. An event in order to be logged has to meet one of these two requirements, being at least two hour long or happen during the data logging time.

The next step is to try and determine patterns in the events causes and determine if any type of model can be created to predict this type of events.

4.3 Events summary

After creating the event matrix, a closer look into the data collected in the matrix showed that the data is stored in a highly variable manner, therefore each event observation was analyzed individually to obtain a description and a cause.

After determining the characteristics of all the events present in the event matrix, a table summary is presented.

Table 4.3 Total events recorded for Unit 1

Description	**Quantity**	**Average duration (hours)**
Shutdown events	42	5.42
Decrease in production	50	18.16
TOTAL EVENTS FOR UNIT 1	92	

As shown in Table 4.3 the total amount of events is 92, with a decrease in production being on average the longest events compared to the shutdown events. In terms of quantity, although decrease in production is larger, the amount of events is not significantly different.

Table 4.4 Total events recorded for Unit 1 by year and type

Description	**2011**	**2012**
Shutdown events	27	15
Decrease in production	20	30
TOTAL EVENTS FOR UNIT 1	47	45

As shown in Table 4.4 the distribution of events in terms of quantity over the two years analyzed is not significantly different, although in terms of type of event it is. Whereas in 2011 were more shutdown events than decrease production, the opposite happens in 2012 were the shutdown events were half the amount of the decrease in production events.

Even if a shutdown event can seem worse than a decrease in production event, a sustained increase in the later can cause a big loss in energy injection to the power grid in the long run in case these events aren't scheduled with the national energy authority.

For the two types of events, 12 possible causes can be summarized that can produce the event of those 12 just 1 is causing a decrease in power production and the rest are associated with shutdowns. A breakdown list of these causes is shown in Table 4.5 and Table 4.6.

Of the data presented in both tables it is remarkable that for all the events in decreased production, all were caused by a restriction of the wells to some extent, all these restriction were programmed due to wells maintenance or different field activities.

The shutdown events are very different, these events are mainly related to a shutdown in Unit 3, due to the large size of the electric system of that turbine, the whole system becomes unstable and it is common for shutdowns in Units 1 and/or 2 to happen.

Table 4.5 Complete event table for Unit 1

Description	**Quantity**
DECREASED PRODUCTION	
Restricted wells	46
SHUTDOWN	
Shutdown in Unit 3	17
Programmed shutdown of Unit 1	5
Low condenser level	2
Failure in condenser	1
Failure in excitation diodes	1
Strong variations of the system	1
Failure in circulation system	1
Switch failure	1
Valve failure	1
High differential pressure	1
Failure in wells	1
No data	14

It is also very interesting to see that of all the shutdowns, only five of them were programmed and of those 5 only 1 was an overhaul, the rest were small programmed stops for testing that were completed in less than a day.

Table 4.6 Complete event table for Unit 1 by year

Description	**2011**	**2012**
DECREASED PRODUCTION		
Restricted wells	19	27
SHUTDOWN		
Shutdown in Unit 3	12	5
Programmed shutdown of Unit 1	3	2
Low condenser level	2	0
Failure in condenser	1	0
Failure in excitation diodes	1	0
Strong variations of the system	1	0
Failure in circulation system	1	0
Switch failure	1	0
Valve failure	1	0
High differential pressure	0	1
Failure in wells	0	1
No data	5	9

A similar table of recorded events was created for Unit 2, showing the total number of events recorded from the algorithm and the average duration of the events, Table 4.7 shows the total event count for Unit 2, the total amount of events is approximately 60% more than the events of Unit 1, shutdown events been almost the same for Unit 1 in the same period, the average duration of shutdown events is similar to Unit 1 and the average duration of decrease in production is one third of Unit 1.

Table 4.7 Total events recorded for Unit 2

Description	**Quantity**	**Average duration (hours)**
Shutdown events	40	4.65
Decrease in production	110	5.56
TOTAL EVENTS FOR UNIT 2	150	

When dividing the events by year, two different trends happen, shutdown events decreased for 2012 and decrease in production increased for 2012

Table 4.8 Total events recorded for Unit 2 by year and type

Description	**2011**	**2012**
Shutdown events	26	14
Decrease in production	14	96
TOTAL EVENTS FOR UNIT 2	40	110

From the totality of events shown in Table 4.8 for decreased production only one of those was due to repairs in a valve, the other ones were due to restricted wells, in both cases the events were scheduled.

Of the shutdown events, most of them were linked to a shutdown in Unit 3, followed by a high level in the demister and then vacuum loss and programmed shutdown. Unregistered events are almost one third of the shutdown events.

Table 4.9 Complete event table for Unit 2

Description	Quantity
DECREASED PRODUCTION	
Restricted wells	108
Repairs in valve	1
SHUTDOWN	
Shutdown in Unit 3	14
High level in demister	5
Programmed shutdown in Unit 2	3
Vacuum loss	3
Variations in system	1
Voltage failure	1
Valve failure	2
No data	12

Table 4.10 shows the events by year for Unit 2, most of the shutdown events are linked to Unit 3 and of those, more than two thirds happened in 2011, of the recorded events, variations in system, voltage or valve failure were nonexistent during 2012

Table 4.10 Complete event table for Unit 2

Description	2011	2012
DECREASED PRODUCTION		
Restricted wells	13	95
Repairs in valve	1	0
SHUTDOWN		
Shutdown in Unit 3	10	4
High level in demister	4	1
Programmed shutdown in Unit 2	2	1
Vacuum loss	1	2
Variations in system	1	0
Voltage failure	1	0
Valve failure	2	0
No data	5	7

4.4 Events interpretation

When analyzing the events for Units 1 and 2, having 242 events can suggest plenty of data to build a predictive model for event occurrences. The problem with the dataset at hand is that these events are classified by Unit and then leads to having two subsets of events, shutdown events and decreased production events.

When the event classification was done, events like shutdown in Unit 3 and restricted wells were commonly identified as causes for the shutdowns and production decreases, the first type is related to a dataset not used for this research, and the latter is a scheduled event mostly for maintenance purposes.

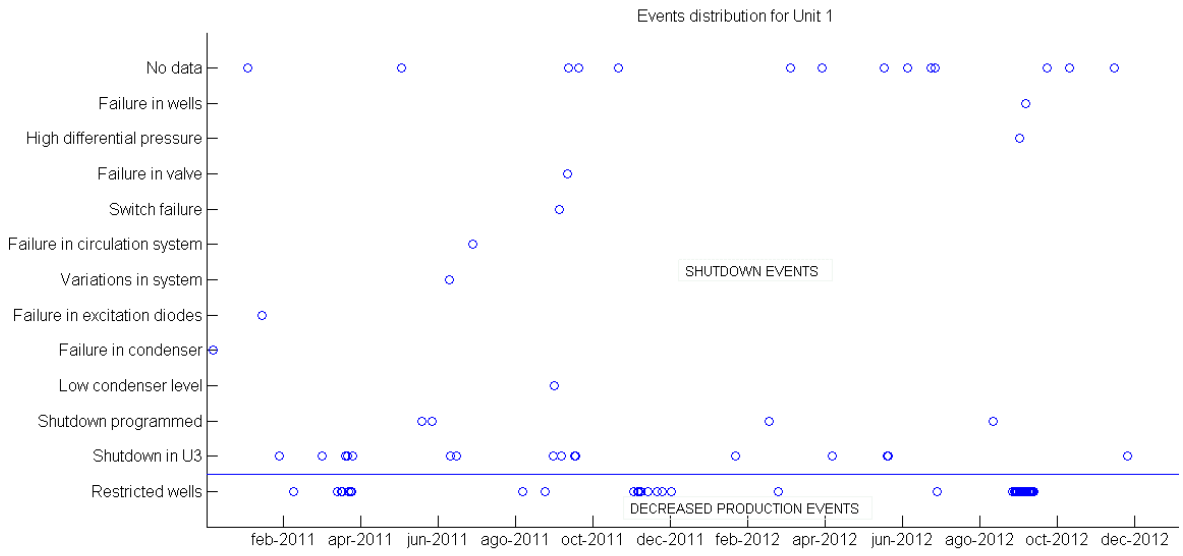

Figure 4.1: Event distribution by type for two years of observations for Unit 1

Identifying the events as shown in Figure 4.1 and Figure 4.2, classifying and counting them helped to realize in this research, that within the dataset used it would not be possible to create predictors for these types of events. Of the events recorded only 10% were unscheduled and there were not more than two occurrences for the unscheduled events, similar with Unit 2 only 8% were unscheduled events.

Of the events recorded, the decreased production events, were mostly scheduled operations in the field that lead to a decreased steam mass flow to the turbine, and then to a decreased production.

For an event predictor to be created a larger dataset with more occurrences of the same events would be necessary. It is important to remember that in between each observation there is a difference of 2 hours, in this time the system can heavily fluctuate, and with an increase in the data frequency of logging, a predictive model could be created.

The same principle applies to the shutdown events, where there are even less observations of shutdowns and most of them being related to Unit 3. A larger dataset including, logged data from Unit 3 could increase the possibilities of prediction and flexibility of the current models created for this particular research.

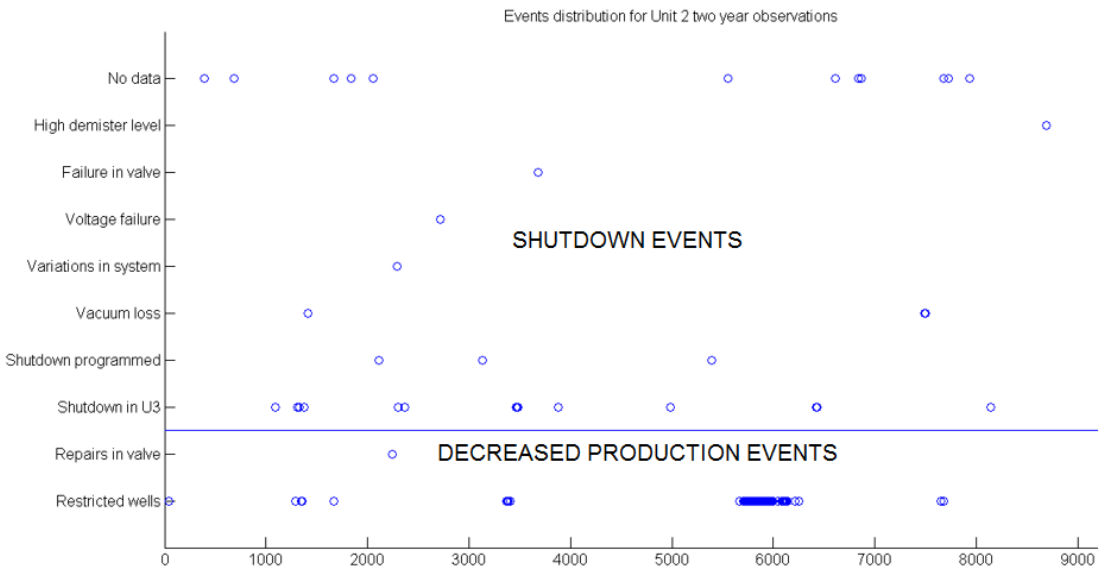

Figure 4.2: Event distribution by type for two years of observations for Unit 2

4.5 Summary

The success of an event prediction task can be based on the variability of the dataset, in the sense that while having plenty of data to describe the power operation, the datasets lack enough observations of unexpected operation. While this is a good thing for the power plant, meaning that the operation is reliable, it doesn't offer enough observations to attempt event prediction. Even having plenty of observations cannot guarantee a success in an event predictor, further research can focus on this topic.

Future work on this matter can include the characterization of the distribution of anomalous operation, for instance, detecting the type of distribution of the unexpected shutdowns in Unit 3 followed. This is a very interesting field to study that can link this research with maintenance planning.

5 Predictive models

A discussion about the datasets used to create the predictive models is offered in this chapter. The thermodynamic model is discussed after the datasets and then the predictive models created from the data are described.

The comparison of the performance of all the models is discussed in the next chapter: Models comparison, to focus the discussion on the models and their creation to this chapter.

5.1 Datasets

This section will describe the data prediction models created from the dataset. A split of 70/15/15 was used for both Unit 1 and Unit 2 databases and both databases were used discarding the observations were the turbine were stopped. It is important also to point out that the training data was randomized in order to improve generalization in the observations when creating the predictive models (regression trees, random forest and linear models).

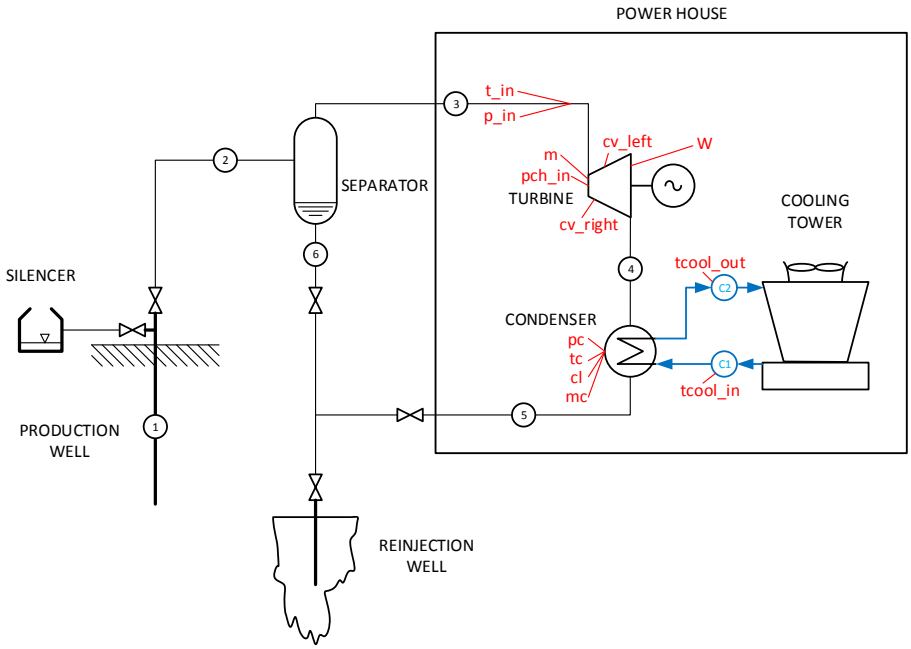

Figure 5.1: Location of sensors in the power plant of the variables considered for the models

The split stands for how the data was divided into training set, validation set and testing set. The training set consist of 70% of the data and it is used to create the predictive models, the validation set consist of 15% of the data and it is used to select the best trained model, the rest of the split 15% belongs to the testing set, as it name implies, the final model will be tested on this data, the figures and graphs shown in this subsection are created from this data.

The variables considered for all the models are the same described in Table 3.3 and are shown in Figure 5.1. The data coming from this variables is stored in the operational report with 2 hour intervals for the whole dataset.

The regression tree model and the random forest model, were created using rattle package on R, which is a data mining package developed for R. The linear models, minimum squared and min Akaike Information Criterion (AIC) model were created using custom made code in R. A general explanation on the creation of this models is offered.

5.2 The thermodynamic model

A thermodynamic model of Unit 1 and Unit 2 was created to compare the predictive models with a physical model. The thermodynamic properties are calculated using REFPROP fluids library on Matlab and the proprietary library on EES.

A general thermodynamic model was created in EES to show the general cycle of a single flash power plant. This model considers the components found in the geothermal field, wells, cyclone separators, pipelines and the power house. This model is shown for schematic purposes in Figure 5.2 and Figure 5.3.

The EES model calculates a thermodynamic balance set of equations that describe the Unit 1 and Unit 2, whereas the Matlab function is an algorithm that gathers the information coming from the reports. Figure 5.2 shows the schematic of the EES thermodynamic model.

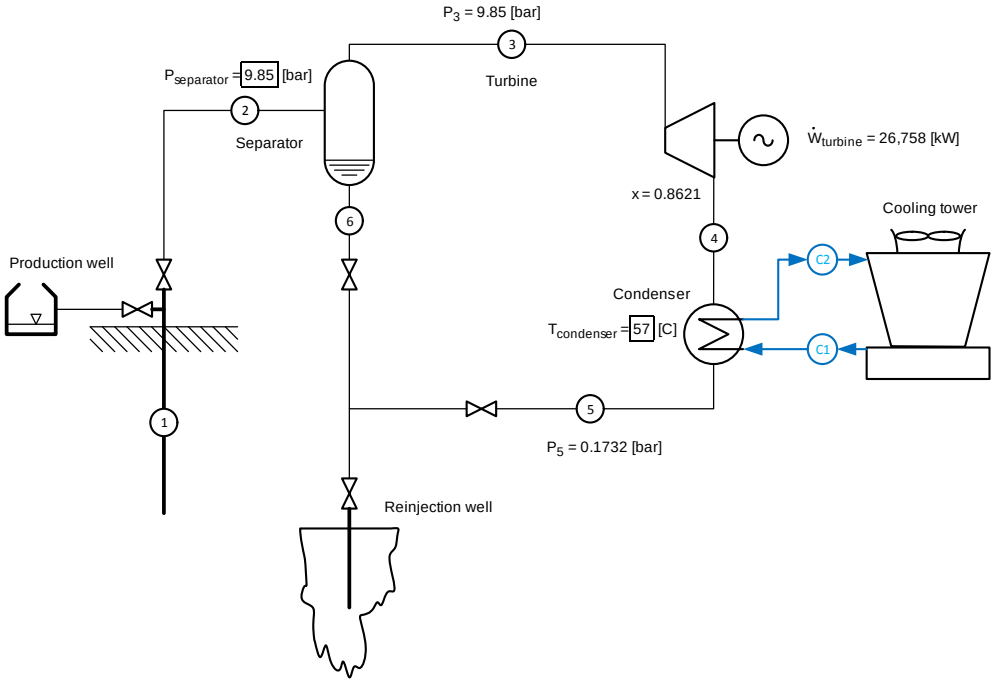

Figure 5.2: Schematic of the modeled thermodynamic cycle

The higher pressure of the cycle, is defined by the separation pressure ($P_{separator}$), the pressure in the separator will be distributed along the pipeline carrying the vapor into the turbine (P_3) as shown in equation 5.1 and also the reinjection pipeline in this scenario (P_6). The separation

process is also defined by the heat and mass transfer equations, equation 5.2 denotes the mass balance, equation 5.3 denotes the heat transfer between the fluid at the inlet of the separator and the liquid and vapor phase at the outlet of the separator.

$$P_{separator} = P_3 = P_6 \quad 5.1$$

$$\dot{m}_2 = \dot{m}_3 + \dot{m}_6 \quad 5.2$$

$$\dot{m}_2 * h_2 = \dot{m}_3 * h_3 + \dot{m}_6 * h_6 \quad 5.3$$

Where P is pressure in kPa, m is mass flow in kg/s, h is enthalpy in kJ*kg/K

The turbine is the next component of the power plant following the path of the vapor. When calculating the turbine power output, the efficiency of the turbine needs to be taken into account in the heat balance.

The efficiency of the turbine is calculated as the energy at the inlet minus the energy at the outlet. Since the turbine is not a perfect adiabatic process, the real output of the turbine is obtained considering the isentropic efficiency (η), shown in equation 5.4. The power output of the turbine is defined by equation 5.5.

$$\eta_t = \frac{h_3 - h_4}{h_3 - h_{4s}} \quad 5.4$$

$$\dot{W}_t = \dot{m}_3 * (h_3 - h_4) \quad 5.5$$

Where η_t is the efficiency of the turbine, $\dot{W}_t$ is power output of the turbine in kW

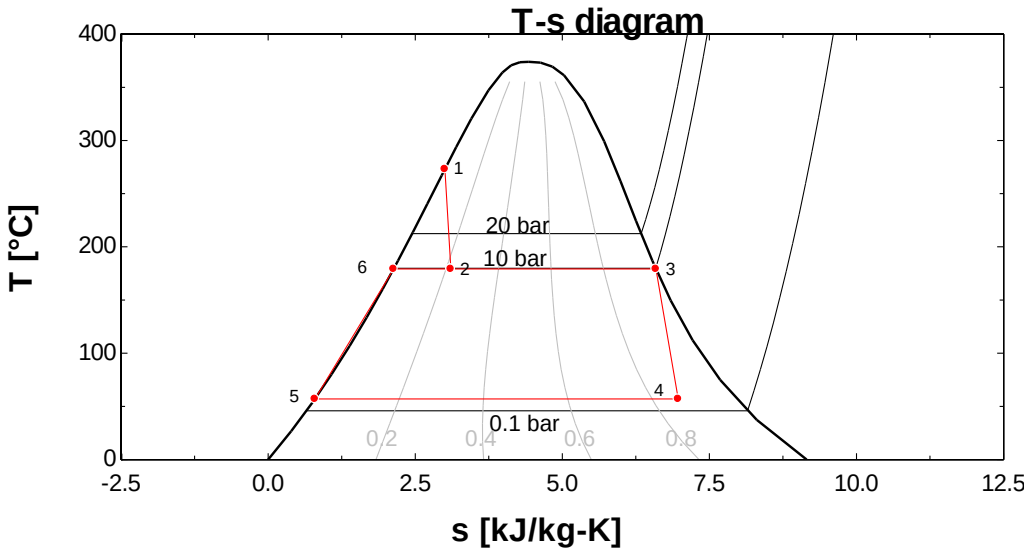

Figure 5.3: T-s diagram of the single flash thermodynamic model

The subscript *s* in the enthalpy of the 4 stage denotes the isentropic enthalpy of the stage. The properties of stage 4 are defined by the temperature of the condenser. The cycle can also be followed in the T-s diagram (Temperature – entropy) as shown in Figure 5.3.

The Matlab function created for the operational data uses an algorithm and a series of equations to obtain the turbine output by reading a dataset. The Matlab function, as stated before, is combined with REFPROP to calculate the desired value, power output in the case of this research.

The function sfcalc was created to calculate a general thermodynamic plant based on the inlet and outlet conditions of the turbine, it also calculates the efficiency of the turbine based on the observed power of the turbine. The following is a pseudo code for the *sfcalc* function.

Table 5.1: Pseudo code for thermodynamic model algorithm

Function "sfcalc"
"Initialize variables"
Set Inlet temperature (t_in)
Set Inlet pressure (p_in)
Set Inlet steam mass flow (m_in)
Set Condenser temperature (p_c)
Set Observed power of the turbine (W)
Set Type of fluid analyzed ("water")
"Calculate thermodynamic properties"
Call REFPPROP function
Calculate Entropy and Enthalpy at the inlet (s_inlet, h_inlet)
Calculate Isentropic enthalpy (h_outs)
Calculate Liquid and Vapor enthalpy at the outlet (hl_out, hv_out)
Calculate Vapor enthalpy at the inlet (hv_in)
"Calculate function outputs using thermodynamic balance equations"
Calculate Turbine work (W_calc1)
Calculate Turbine efficiency (eta_calc1)

The function created to model the cycle uses 6 input variables and it is also modeled after the way data is treated in the operational reports, the function gives 4 outputs. To obtain the thermodynamic properties for water, the function calls the REFPROP library, in this case a function created to read and process arrays or matrices.

Although the function calculates the theoretic efficiency based on the observed power values of the turbine, these variables are not used later in this research, but it is also calculated to consider it for further research.

The function uses a similar set of equations as the ones used in the EES model and also calculates the turbine efficiency based on the actual rated turbine output, it uses an equation considering the Baumann rule for wet turbines efficiency (DiPippo, 2007).

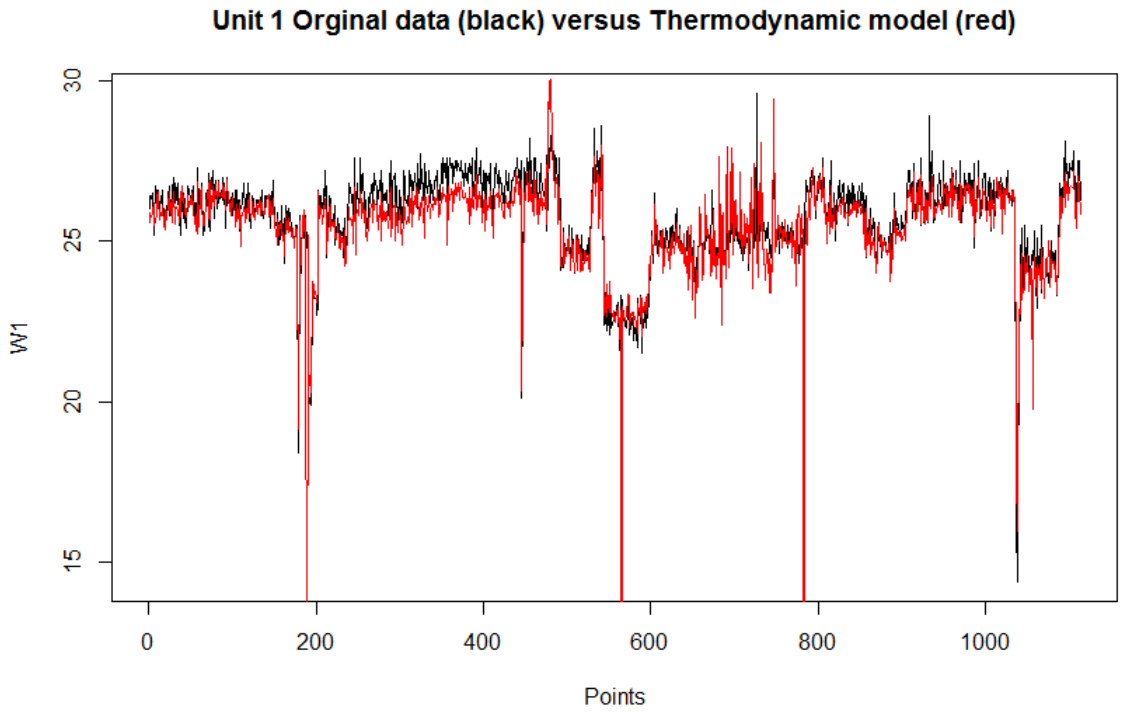

Figure 5.4: Thermodynamic model for Unit 1 data

Figure 5.4 and Figure 5.5 show the general thermodynamic model performance over the testing dataset and while it shows very good accuracy, it fails to predict over the data at some points.

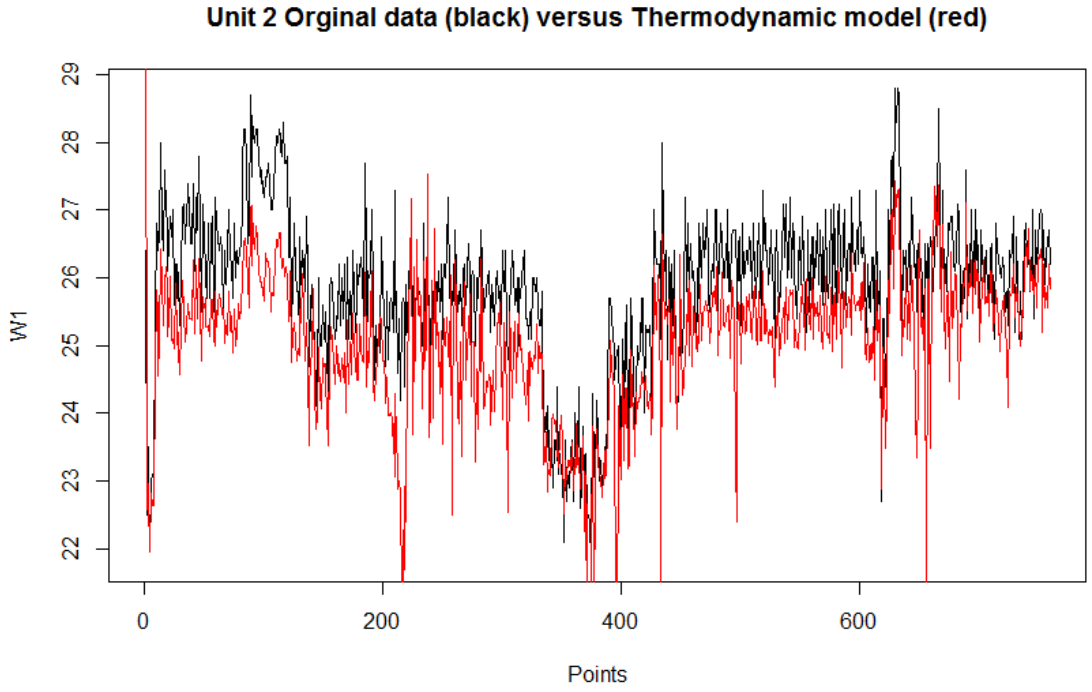

Figure 5.5: Thermodynamic model for Unit 2 data

5.3 Regression trees

Decision trees is a general term used for two data mining tools, classification trees and regression trees. Classification trees are used when the target function is a categorical variable, while regression trees are used when the target function is of numerical values, hence the use of regression trees in this research.

The techniques were developed in the 1980s, and since then they were one of the most widely used tools, due to the simplicity of the resulting model and simplicity to follow. The simplicity of the rules created from a decision tree, makes them one of the choice when considering classification problems.

A regression tree learns from a dataset to create a basic set of rules which can be followed.

A general example of a regression tree is shown in Figure 5.6, the dataset used for this example calculates a computer hard drive speed based on the size of the data package it requests. The tree is followed from the top, showing the answers at the bottom.

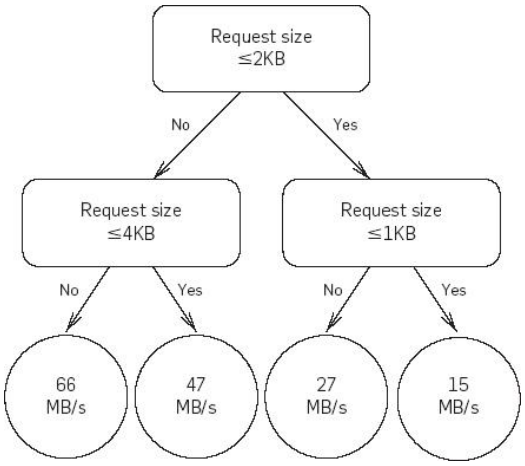

Figure 5.6: Regression tree example (Meisner et. al., 2009)

In this case the hard drive speed is determined following the tree from top to bottom, if the data package request size is bigger than 2KB the left branch is followed, the if the requested data package is smaller than 4KB the tree will "predict" that the hard drive speed is 47 MB/s, the goal of this particular tree is to learn the performance of a disk drive for sequentially-read data (Meisner et. al., 2009)

R is used for creating the regression trees in this research, more specifically the rpart package in R. Rpart stands for Recursive partitioning for classification, regression and survival trees, this algorithm follows closely most of the functionality of the 1984 book Classification and Regression Trees by Breiman et. al.

The method for constructing a regression tree is very similar to the one used for constructing a decision tree, it goes as follows:

- Selecting the node splits, based on information gain

- Deciding to continue splitting a node or declare it terminal
- Assigning each node with a function value

The difference between constructing decision or regression trees is the node split criteria, given that the information gain is measured differently for decision or regression trees.

The algorithm iterates over all the attributes present in the database, in this case, over the 12 independent variables and then selects the attributes that gives the most information gain, the predicted value here will be the average of all the observations falling into that node. (Breiman et. al, 1984)

For this particular research the anova method is used for the splitting criterion, this criterion will decide which variables gives the best split of the data, maximizing the value of equation 6.1, as shown in equations 5.6 and 5.7 (Therneau et. al, 2014)

$$SS_T - (SS_L + SS_R) \tag{5.6}$$

Where

$$SS_T = \sum (y_i - \bar{y})^2 \tag{5.7}$$

Where SS_T is the sum of squares for the node SS_L and SS_R are the sum of squares for the left and right "son" (subsequent) nodes.

Each split will create a smaller subset of the data, where all the variables are analyzed again until a stop criteria is reached, where further splits of the data will only overfit the training dataset and will show poor generalization. In some cases, to prevent overfitting of the tree model, the biggest available decision tree is created, and then the terminal nodes are analyzed and some of them are removed to improve generalization, this technique is called pruning. In this case, instead of pruning, a stop criteria is used, this helps improving computational time because it prevents further iterations over splits that will offer no help to improve the fit.

The stop criteria for growing the tree used in this case is the complexity parameter, the default CP = 0.01 is in rattle. The CP is the factor by which the program will attempt a new split of the data, if the improvement of the prediction does not increase by the CP factor, the program will not attempt to increase the tree further. In the case of anova, each split has to increase the overall R-squared by at least CP each step. (Therneau et.al., 2015)

Table 5.2: Tree rules for Unit 1

nodes (sorted by level) 1	2	3	4	Split	n (obs in this branch)	Value of target (W)	
1)				root	5189	25.77782	
	2)			pch_in< 7.935	1363	24.14435	
		4)		m< 36.2345	15	12.98667	*
		5)		m>=36.2345	1348	24.26851	
			10)	pch_in< 7.505	353	22.72465	*
			11)	pch_in>=7.505	995	24.81623	*
	3)			pch_in>=7.935	3826	26.35973	
		6)		m< 50.643	1892	25.88032	
			12)	pch_in< 8.125	765	25.53416	*
			13)	pch_in>=8.125	1127	26.11529	*
		7)		m>=50.643	1934	26.82873	
			14)	pch_in< 8.585	1241	26.61824	*
			15)	pch_in>=8.585	693	27.20569	*

* denotes a terminal node

Table 5.2 shows the set of rules generated for Unit 1 dataset, it is interesting to note here that the algorithm only selected two of all the explanatory variables, steam mass flow at the inlet (m) and the pressure in the chamber of the turbine (pch_in).

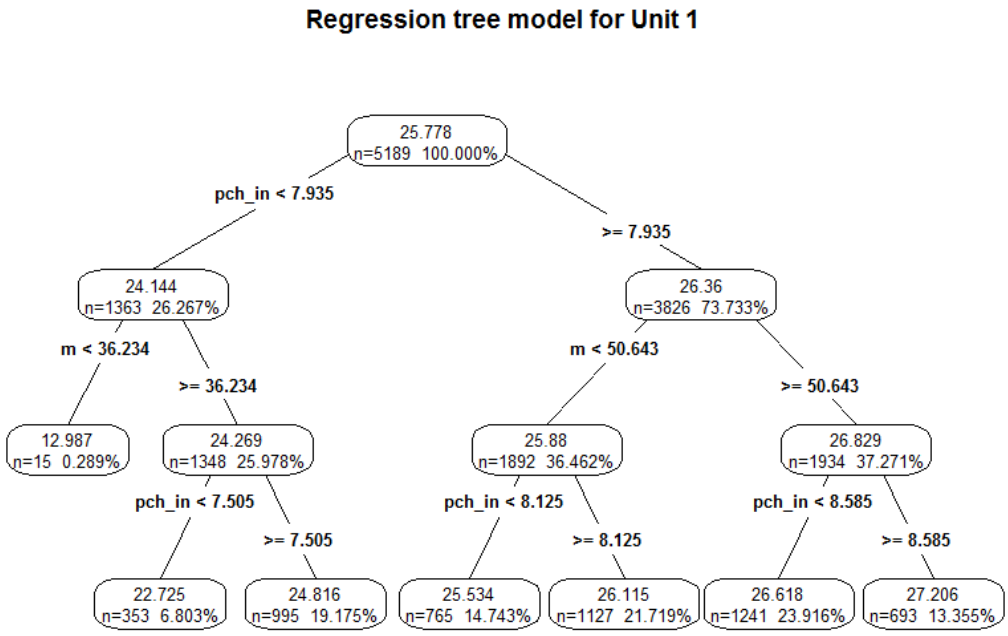

Figure 5.7: Regression tree for Unit 1

Table 5.2 shows the nodes sorted by levels, from node 1 (the root node) on level 1 then a rule on level 2 needs to be followed and so on until level 4. The split denotes the rule which will be followed to continue to the next node level. N denotes the number of observations

from the training set that fall into this particular rule, in general, all the observations in the same level should have a sum equal to the total of observations. Value target denotes, the final value of the regression tree function, but the value will ultimately be predicted by the terminal nodes (the ones with an asterisk next to it).

Each node in the figure shows 3 values, on top is the target value (W), n is the number of observations on the training dataset that fall in this particular node and the percentage represents the fraction of the whole training dataset that the particular node represents.

A graphical representation of the set of rules shown before can be found in Figure 5.7, the set of rules can be followed by inspecting the root node, the one on top, then following each branch and comparing the current variables, the function finally predicts the target value.

Figure 5.8 shows the model applied to the testing dataset. The comparison of all the models will be done at la later in this document, for now, it is important to note that the model can predict fairly accurate the testing dataset, only failing to do so in some of the peaks of the data. Another interesting thing about the model is the way it looks over the real data, looking like steps in some fashion, this phenomenon is because of the characteristic rules that govern the model.

Figure 5.8 Regession tree model results of Unit 1 data

Table 5.3: Tree rules for Unit 2

nodes (sorted by level)				Split	n (obs in this branch)	Value of target (W)	
1	**2**	**3**	**4**				
1)				root	3540		
	2)			m< 49.1015	1007	24.67597	
		4)		cv_right< 20.5	310	23.39194	
			8)	m< 44.4375	61	22.23607	*
			9)	m>=44.4375	249	23.6751	*
		5)		cv_right>=20.5	697	25.24706	
			10)	pch_in< 7.975	364	24.89368	*
			11)	pch_in>=7.975	333	25.63333	*
	3)			m>=49.1015	2533	26.29491	
		6)		m< 51.714	1704	25.99805	
			12)	m< 50.4705	788	25.73255	*
			13)	m>=50.4705	916	26.22645	*
		7)		m>=51.714	829	26.9051	
			14)	pch_in< 8.725	602	26.66757	*
			15)	pch_in>=8.725	227	27.53502	*

* denotes a terminal node

Table 5.3 and Figure 5.9 show the regresion model for Unit 2 database, the main difference here is that the regression tree model selects three variables, steam mass flow at the inlet (m x10), pressure in the chamber of the turbine and an additional one right control valve possition (cv_right).

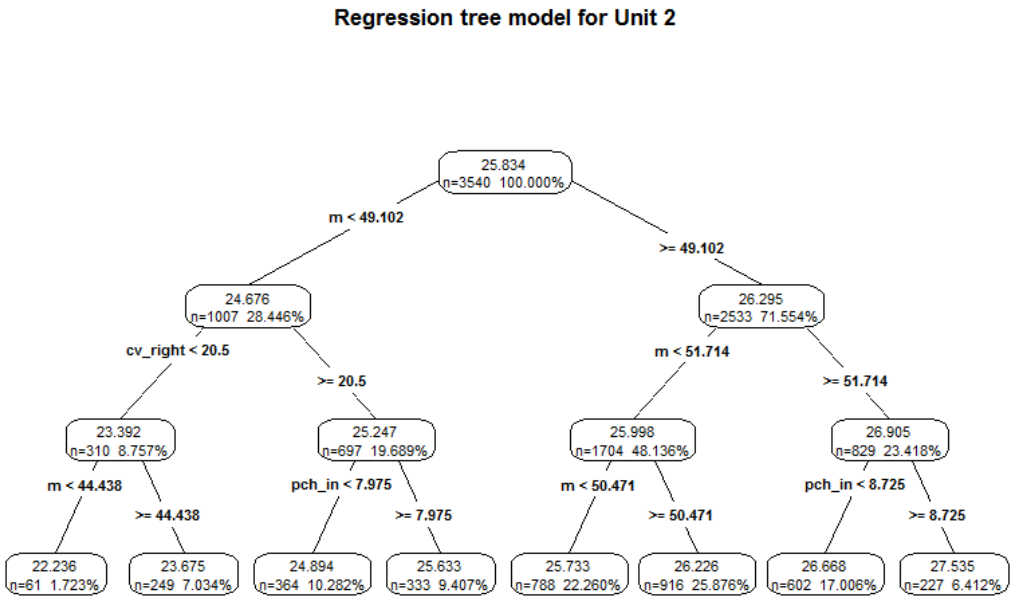

Figure 5.9: Regression tree for Unit 2

As with Unit 1 regression tree, mass flow and prresure in the chamber were selected to create the model from the algorithm, which means that these variables explian the dataset very good from the regression tree perspective, and serve the purpose of the regression tree model.

Similarly for Unit 1 the representation of the model applied to the testing dataset for Unit 2 data is shown in Figure 5.10, the main difference here is that the random data split has less peaks than the data for Unit 1.

The prediction distribution is also similar for Unit 1 and Unit 2 for the regression tree models, making it a very good tool for estimating values, and simple to explain in a quick fashion. Regression trees, in this particular research have proven to be an integral part of the starter pack of any data mining researcher.

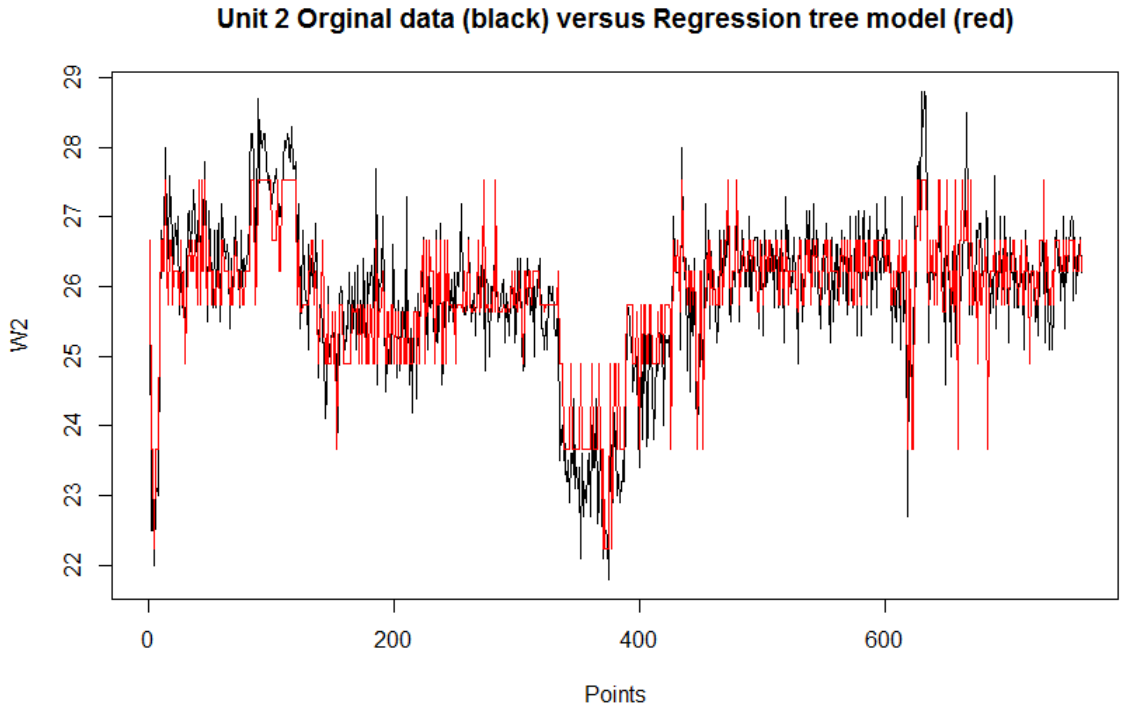

Figure 5.10 Regession tree model results of Unit 2 data

5.4 Random forest

The random forest technique comes from the idea of having many regression tree models created over a single dataset, instead of having just one model to predict the data. The random forest models usually produces better (more accurate) models because the group of trees reduces the instability observed when having just one.

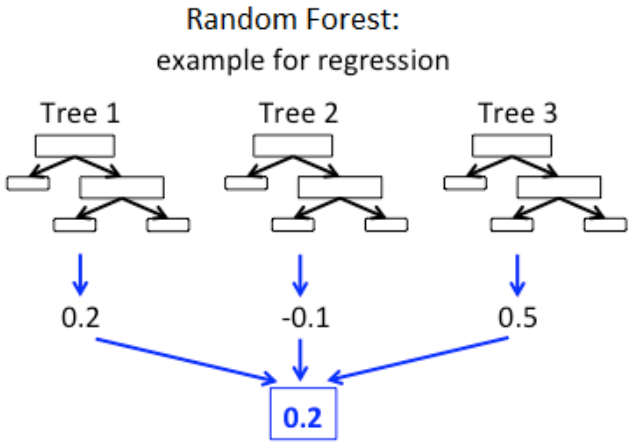

Figure 5.11: General schematic of a Random forest model (Bradley and Amde, 2015)

Essentially random forest are a collection of decision trees that individually predict the value of the target function, and then the whole forest votes for the final predicted value, as shown in Figure 5.11. the random forest technique handles changes in the data quite well, and it is very robust to variables that have little relationship to the target variable. It is very important to note, that this particular technique uses un-pruned decision trees. The resulting decision of a random forest model will have very little disturbance over small changes in the data.

The final predicted value of the random forest is the mean of all the predicted values from every tree. (Breiman, 2001)

As with the regression tree model, a random forest model was created for both turbines datasets. The number of trees selected for both models was 500, in order to show how the accuracy of the forest behaves as the number of trees in the forest increases.

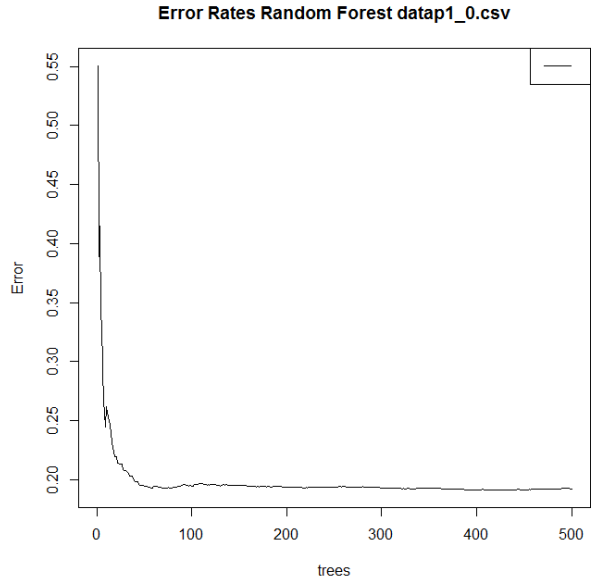

Figure 5.12 Error rate for random forest model for Unit 1

It is important to note that the prediction accuracy will ultimately be based on the dataset distribution, the logical tendency will be towards a stagger in the accuracy regardless of certain number of trees.

Figure 5.12 and Figure 5.13 show how the error rate for both turbine random forest models, these graphs plot the error rate progressively with the number of trees used, this is useful to decide the number of trees used when computation times are considered. Regardless of how quickly the model achieves its top accuracy it is interesting to note that overall, the model for Unit 2 is more precise.

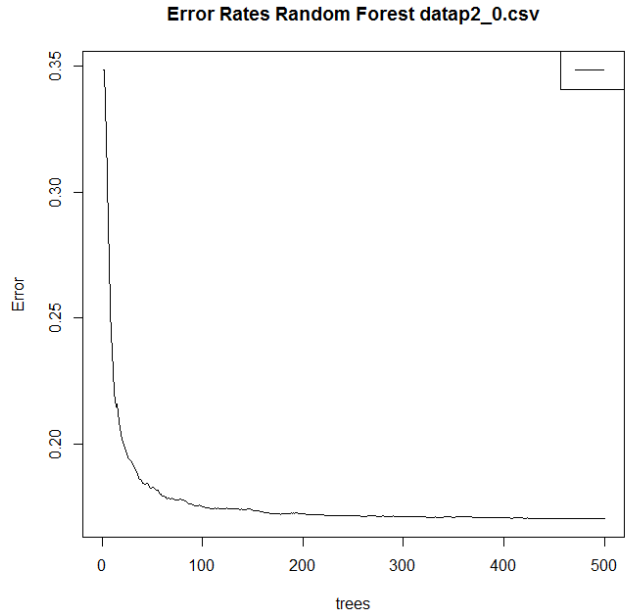

Figure 5.13 Error rate for random forest model for Unit 2

Although in practice, when a much larger dataset is considered, the number of trees used in the random forest will impact the computation time. While this a very efficient algorithm, when used for real time prediction, the trees number can be an important factor to consider when executing the model if the computational time is a concern, because the algorithm will calculate the result predicted from each tree individually.

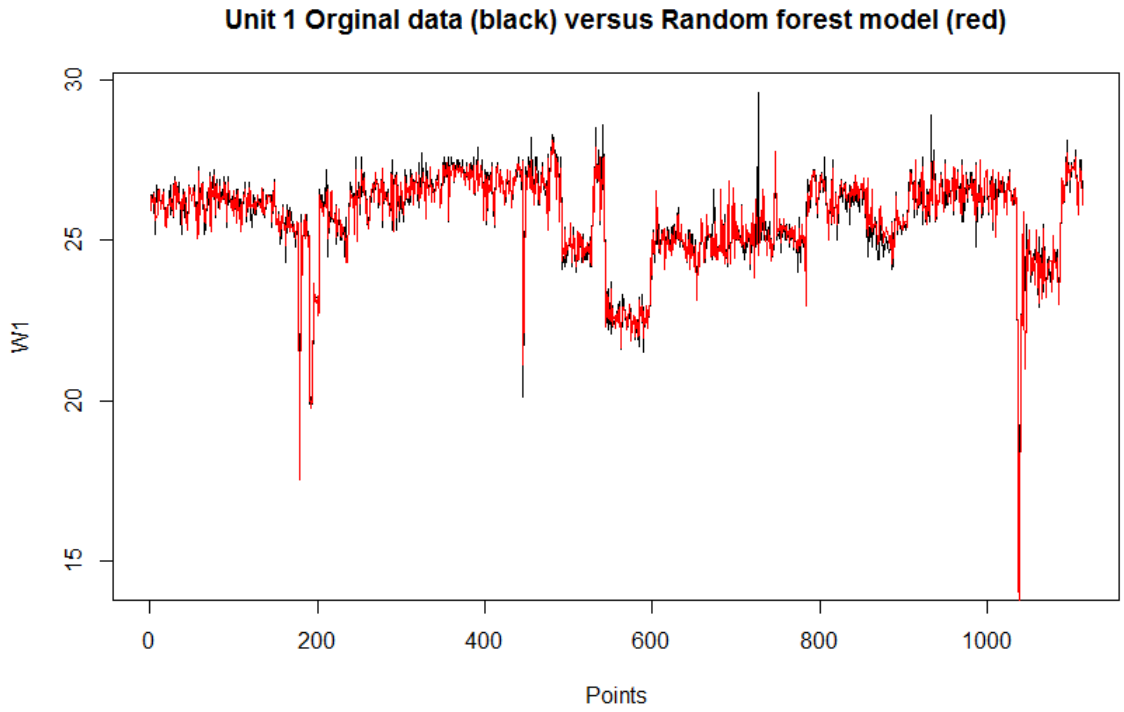

Figure 5.14 Random forest model result for Unit 1

The next step, as with the regression tree models is to test the random forest, for that the testing dataset will be used. Figure 5.14 and Figure 5.15 show the performance of the random forest model

When plotting the results of the random forest models is inevitable to compare them to those of the regression tree models. Both figures show how the random forest predicts the target variable with a very good accuracy, failing only on some peaks but without much sacrifice of the performance, still the value obtained would be very close to the actual value, also its important to remember that the data is not sequential and that the target values were randomized.

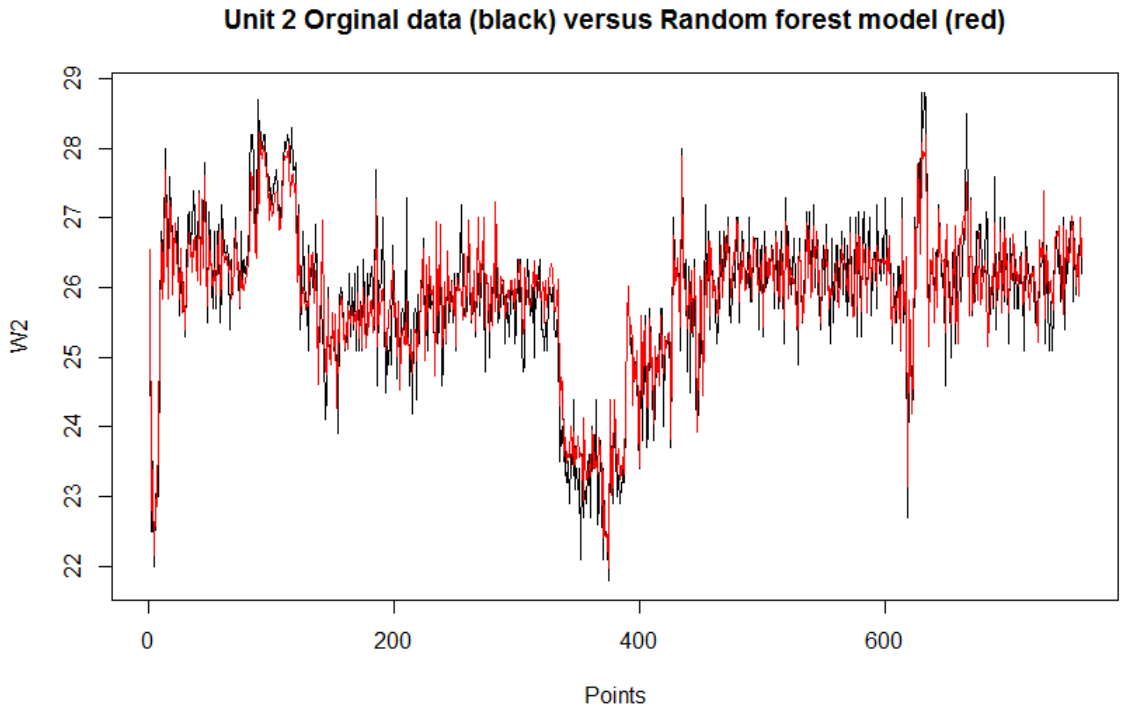

Figure 5.15 Random forest model result for Unit 2

5.5 Linear regression models

This part consist of regression models created by every possible combination of the 12 explanatory variables in the dataset (Table 4.3 and Figure 6.1) plus an intercept, all the models were created using the same training set as with the regression tree and random forest models. Two comparison methods were used to score every model created, the models were scored over the validation dataset.

5.5.1 Minimum validation error

The first method for comparing the models was the smallest validation accuracy (error), in here the sum of the absolute value of the difference between the observed values and the calculated values is stored, and the model with the smallest validation accuracy is selected. This model should be the one with the highest accuracy of all the models created in the algorithm and compared with this technique.

The validation accuracy is calculated with the equation 5.8 over the validation dataset

$$validation = \sum_{i=1}^{n} \left| f(X_1, \dots, X_p)_i - Y_i \right| \qquad 5.8$$

Where n is the number of observations predicted, Y_i is the observed value, f_x is the predicted value, p is the number of variables used in the model

A simple algorithm to create all possible combinations for the explanatory variables was created in R. Then the algorithm creates a linear model using each combination and the testing dataset, to test it over the validation dataset, the scores for both validation accuracy and AIC values are stored in a matrix for later selection.

After the loop, the algorithm selects the models with the smallest validation accuracy and AIC value. The coefficients of the minimum validation accuracy models are shown in Table 5.4, both equations are using most of the explanatory variables and are tested later on in the testing dataset.

Table 5.4: Variable coefficients for min val model for Unit 1 and Unit 2

Description	**Variable**	**Unit 1 Value**	**Unit 2 Value**
(intercept)		4.14796	13.53941
Turbine steam mass flow at the inlet	m	0.28140	0.17915
Turbine steam inlet pressure	p_in	0.09442	0.24034
Turbine inlet temperature	t_in	---	---
Turbine chamber inlet pressure	pch_in	1.46542	0.54719
Condenser pressure	pc	-0.78210	24.99239
Condenser temperature	tc	-0.06048	---
Condenser level	cl	---	---
Cooling water mass flow	mc	0.00079	0.00025
Cooling water inlet temperature	tcool_in	-0.07567	-0.45088
Cooling water outlet temperature	tcool_out	-0.03255	0.11211
Left control valve position	cv_left	0.00878	0.03983
Right control valve position	cv_right	0.00453	0.03707

Equations 5.9 and 5.10 show the formulas from the minimum validation accuracy method, each of the variables in it, has to be multiplied by its coefficient.

$$W_1 = m_1 + p\_in_1 + pch\_in_1 + pc_1 + tc_1 + mc_1 + tcool\_in_1 + tcool\_out_1 + cv\_left_1 + cv\_right_1 + \text{intercept} \quad (5.9)$$

$$W_2 = m_2 + p\_in_2 + pch\_in_2 + pc_2 + mc_2 + tcool\_in_2 + tcool\_out_2 + cv\_left_2 + cv\_right_2 + intercept \quad (5.10)$$

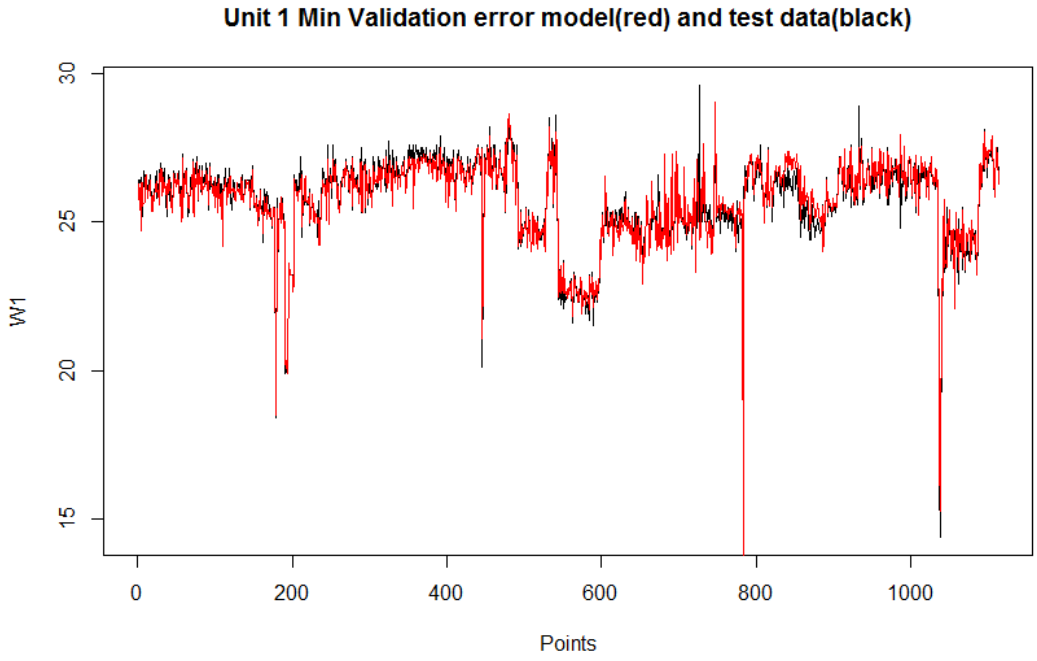

Figure 5.16 Min validation accuracy model of Unit 1 data

Figure 5.16 and Figure 5.17 show how the models with the smallest validation accuracy predict the testing dataset. These models show an accuracy comparable to that of the random forest. The main difference here is that this type of model take les computational time than the random forest. Even though both model take into account most of the explanatory variables from the problem, they seem to show very good generalization over the unseen database.

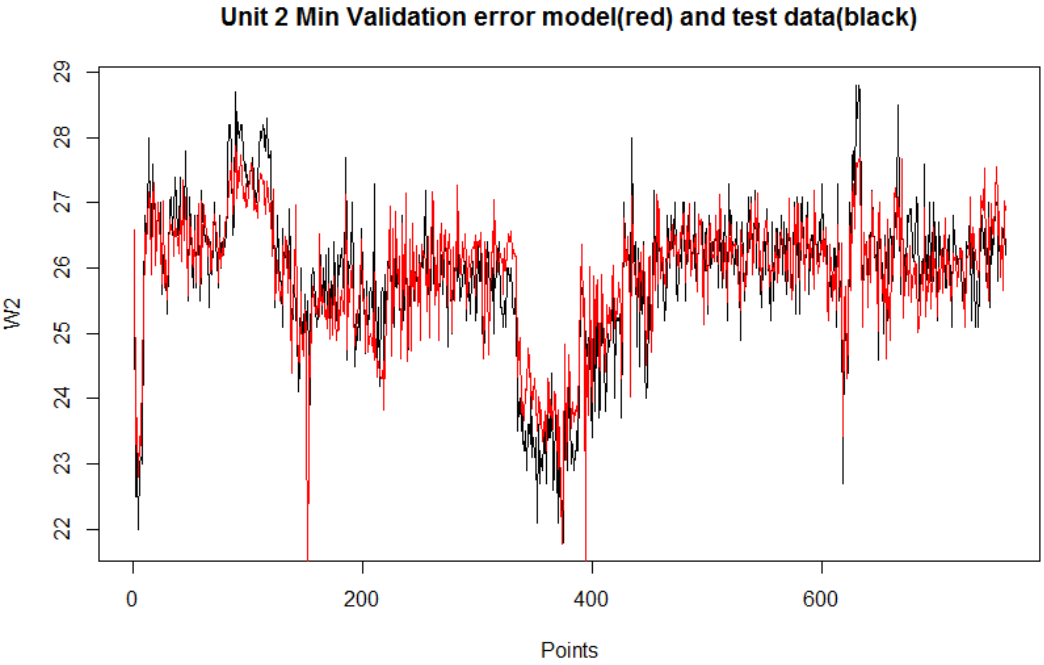

Figure 5.17 Min validation accuracy model of Unit 2 data

5.5.2 Akaike information criterion

The Akaike Information Criterion (AIC) is used to compare models with different complexities, it is a useful tool to select a model that makes a good "explanation" of the data while penalizing the model's complexity in order to prevent poor generalization. When two models are compared and the AIC for each model has been computed, the selected model should be the one with the smallest AIC value.

The AIC is calculated with the equation 5.12 over the validation dataset

$$AIC = n * \log\left(\frac{SS(E)}{n}\right) + 2 * d \qquad 5.11$$

Where n is the number of observations predicted, d is the number of parameters on the current evaluated model, The SS(E) is the sum of squared errors calculated with the equation 5.12

$$SS(E) = \sum_{i=1}^{n} \left(Y_i - f(X_1, \ldots, X_p)_i\right)^2 \qquad 5.12$$

Table 5.5 show the coefficients for the minimum AIC models for both Units 1 and 2. As described before, the selection criteria is to pick the model which gives the smallest AIC value, when comparing several models for the same data.

Table 5.5: Variable coefficients for min AIC model for Unit 2

Description	Variable	Unit 1 Value	Unit 2 Value
(intercept)		4.18957	8.35096
Turbine steam mass flow at the inlet	m	0.28130	0.17557
Turbine steam inlet pressure	p_in	0.09520	0.17321
Turbine inlet temperature	t_in	---	0.02539
Turbine chamber inlet pressure	pch_in	1.46216	0.53025
Condenser pressure	pc	---	23.73854
Condenser temperature	tc	-0.06067	0.05871
Condenser water level	cl	---	-0.00337
Cooling water mass flow	mc	0.00079	0.00026
Cooling water inlet temperature	tcool_in	-0.07725	-0.48143
Cooling water outlet temperature	tcool_out	-0.03354	0.10469
Left control valve position	cv_left	0.00867	0.03841
Right control valve position	cv_right	0.00444	0.03686

Interestingly, while the AIC penalizes for each additional term in the equation, the model with the smallest AIC value for Unit 2 uses the whole set of explanatory variables plus the intercept.

$$W_1 = m_1 + p\_in_1 + pch\_in_1 + tc_1 + mc_1 + tcool\_in_1 + tcool\_out_1 + cv\_left_1 + cv\_right_1 + intercept \quad 5.13$$

$$W_2 = m_2 + p\_in_2 + t\_in_2 + pch\_in_2 + pc_2 + tc_2 + cl_2 + mc_2 + tcool\_in_2 + tcool\_out_2 + cv\_left_2 + cv\_right_2 + intercept \quad 5.14$$

Equations 5.13 and 5.14 show the formulas from the minimum AIC method for both Units 1 and 2, each of the variables in it, has to be multiplied by its coefficient.

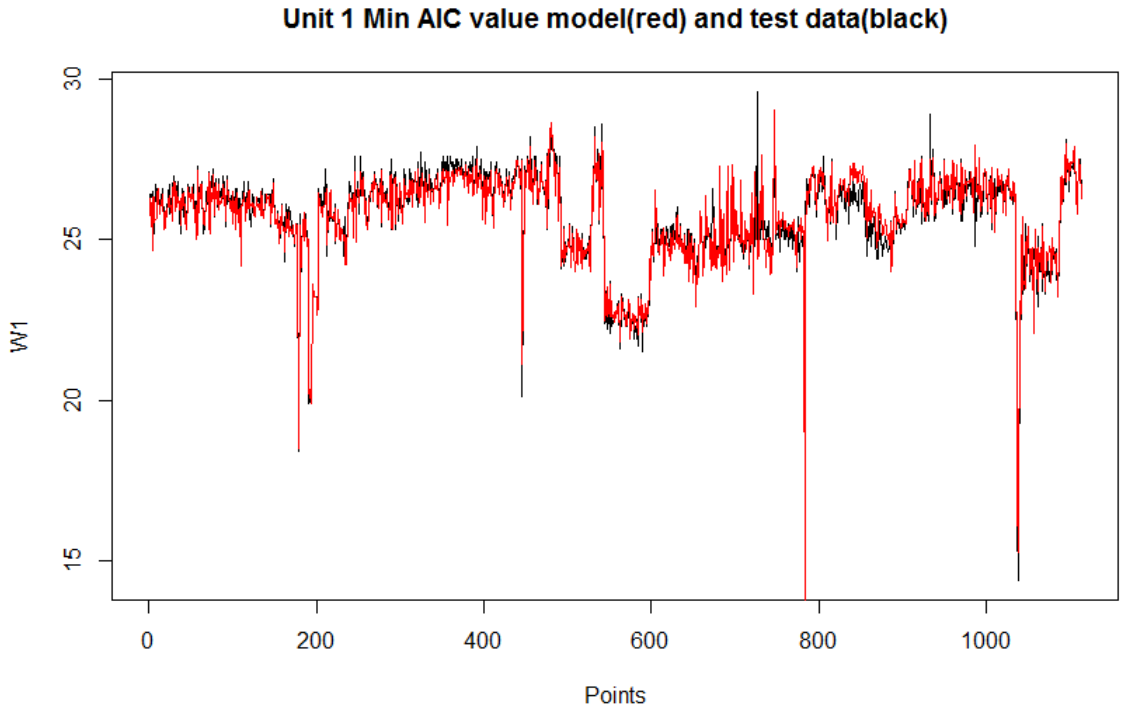

Figure 5.18 min AIC model for Unit 1 data

Figure 5.18 and Figure 5.19 show the performance of the minimum AIC model for both datasets, the performance is very similar to the random forest model and the smallest validation accuracy model.

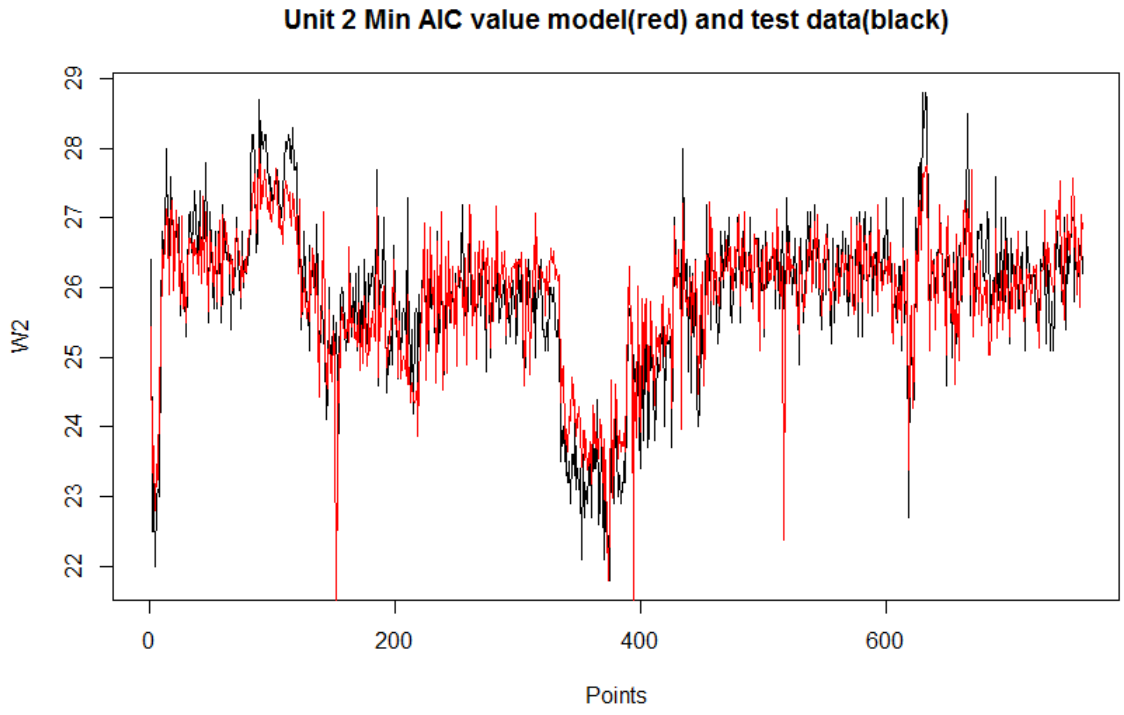

Figure 5.19 min AIC model for Unit 2 data

5.6 Summary

When attempting to create predictive models from big datasets, it is very important to organize data in a way that will make this task easier, this approach helps to save a lot of time in the analyzing stages.

Regression trees have proven to be a very handy tool to get the overall performance of the target function (Power in this case) and can help to create a set of rules later on to update the turbine control system. Random forest, along with the linear regression models are the strongest of the predictive techniques, very robust against unseen data, which makes them perfect candidates for this type of tasks.

6 Predictive models comparison

When model performance is so similar that it cannot be compared easily from just a plot of the predicted data and the real data, further analysis is required in order to select the best model. The models will be compared between them and with the thermodynamic model described earlier.

Using the graphical method can be very useful for selecting different models, but when models show very similar performance, random forest and linear regression models for example, it is very hard to select them over just by comparing them using figures, for that purpose, four different methods were selected.

6.1 Mean absolute error

The mean absolute error (MAE) calculates the sum of the absolute difference between the predicted values and the observed values and divides it between the total number of observations. Equation 6.1 shows the formula for MAE calculation

$$MAE = \frac{\sum_{i=1}^{n}\left(\left|Y_i - f(X_1, \dots, X_p)_i\right|\right)}{n} \quad 6.1$$

Where f_x is the model prediction, Y is the observed value, n is the number of observations

The mean absolute error compares the distribution of the differences of the predicted and the observed values against zero. The lower the MAE, the more accurate the model. The MAE was calculated for each model and the results are shown in Table 6.1.

Table 6.1: Mean absolute error values for Units 1 and 2 models

Model	Unit 1	Unit 2
Minimum validation accuracy	0.01623	0.02024
Minimum AIC value	0.01621	0.02049
Regression tree	0.02036	0.02000
Random forest	0.01026	0.01393
Thermodynamic model	0.03784	0.03965

The table shows that for both Units, the best predictor is the random forest model. While the worst predictor is the thermodynamic model, if the thermodynamic model is not considered, the regression tree is the worst predictor for Unit 1 and the minimum AIC model is the worst for Unit 2.

6.2 Mean square error of prediction

The mean square error of prediction (MSEP) calculates the squared sum of the difference between the observed values between the model predicted values divided by the total number of observations. The MSEP is one of the most common methods to measure the predictive accuracy of a model. It is calculated with equation 6.2

$$MSEP = \frac{\sum_{i=1}^{n}\left(Y_i - f(X_1, \dots, X_p)_i\right)^2}{n} \quad 6.2$$

Where f_x is the model prediction, Y is the observed value, n is the number of observations

When using the MSEP as a comparison tool between models, the MSEP needs to be calculated for each model, and the model with the smallest value shows that is a better predictor than the others.

Table 6.2: Mean square error of prediction values for Units 1 and 2 models

Model	Unit 1	Unit 2
Minimum validation accuracy	0.29304	0.31143
Minimum AIC value	0.29262	0.3192
Regression tree	0.46145	0.30399
Random forest	0.11708	0.14739
Thermodynamic model	1.59344	1.1951

Similarly with the MAE the MSEP shows that the best predictor is the random forest model. While the worst predictor again is the thermodynamic model, if the thermodynamic model is not considered, the regression tree is the worst predictor for Unit 1 and the minimum AIC model is the worst for Unit 2.

6.3 Coefficient of Model Determination

The coefficient of model determination (CD) is the ratio of the total variance of observed data to the squared of the difference between model-predicted and mean of the observed data. Equation 6.3 shows how the CD is calculated

$$CD = \frac{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}{\sum_{i=1}^{n}\left(f(X_1, \dots, X_p)_i - \bar{Y}\right)^2} \quad 6.3$$

Where f_x is the model prediction, Y is the observed value, n is the number of observations, $\bar{Y}$ is the mean value of the observations

If the CD is used to compare predictive models, the value closes to unity is the better predictor. Table 6.3 shows a modified version of this value using the formula *abs(1-CD)* to in this case the smallest the value, the better the predictor.

The CD shows quite different results than the MAE and MSEP for both Units. For Unit 1 the best predictor is the random forest model, while the worst predictor is still the

thermodynamic model, and the regression tree coming on second. For Unit 2, the best predictor is the thermodynamic model, with the random forest showing a very similar result, the worst predictor for Unit 2 is the decision tree model.

Table 6.3: Coefficient of model determination values for Units 1 and 2 models

Model	Unit 1	Unit 2
Minimum validation accuracy	0.01884	0.34858
Minimum AIC value	0.01985	0.32615
Regression tree	0.03319	0.44222
Random forest	0.00244	0.31459
Thermodynamic model	0.34116	0.31228

6.4 Modeling efficiency

The modeling efficiency (MEF) is interpreted as the proportion of variation explained by the predicted values. Equation 6.4 shows how MEF is calculated.

$$MEF = 1 - \frac{\sum_{i=1}^{n}\left(Y_i - f(X_1, \dots, X_p)_i\right)^2}{\sum_{i=1}^{n}(Y_i - \bar{Y})^2} \qquad 6.4$$

Where f_x is the model prediction, Y is the observed value, n is the number of observations, $\bar{Y}$ is the mean value of the observations

If the model prediction was perfect, the value of MEF would be equal to one, and if the MEF is lower than zero the fitted values predict the data worse than using just the mean.

Table 6.4: Model efficiency values for Units 1 and 2 models

Model	Unit 1	Unit 2
Minimum validation accuracy	0.8559	0.76045
Minimum AIC value	0.8561	0.75447
Regression tree	0.77308	0.76618
Random forest	0.94243	0.88663
Thermodynamic model	0.21643	0.08074

The MEF show similar results to those of MAE and MSEP, the best predictor for both Units is the random forest model. While the worst predictor in both cases is the thermodynamic model, followed by the regression tree for Unit 1 and minimum AIC model for Unit 2, but in case of Unit 2, both linear models and regression tree show very similar performance.

6.5 Model selection

In general the random forest model proved to be the one with better performance for both Unit 1 and 2. While this type of model requires data preparation, it can easily be implemented for data prediction, and in case a different approach for data capture is selected later on in the power plant, it can be adapted to a different data scheme.

If the thermodynamic model is not taken into account for both Units, in general the regression tree is the worst performer, this doesn't mean that the method is bad, but that for this particular dataset it does not adapts well.

6.6 Summary

This chapter shows only four of comparison methods available to compare models. The comparison method ends up being a mixture of factors that the researcher needs to take into account, the size of the dataset, the type of data used to create the models, the number of observations, the frequency of observations in case the model is to be used for online predictions.

It is also clear that the different models behave differently for both datasets, reinforcing the previous idea that the datasets are very different and a general predictive model would be hard to come by

7 Discussion and future work

Reviewing the research question, if a predictive model can be created using operational data, the answer is, it can be done, and this is shown in chapter seven, where the comparison of all the predictive models and the thermodynamic model is done.

This research shows that the geothermal power plants databases have rich information from operation that is not being actively studied, at least not in El Salvador.

Further analysis can go into the thermodynamic model to update it and consider reasons why it is less accurate to predict the data at some points.

An event analysis of Unit 1 and Unit 2 datasets is done. Correlation plots and principal components analysis are useful tools to detect data patterns that are not possible to detect just using visualization methods.

Checking the event distribution for Unit 1 and Unit 2, showed that there are not noticeable patterns in the data that may suggest a link between unscheduled events. Further research can be done in Berlin Geothermal field, by increasing the data logging frequency of the operational and field data, to check hidden patterns that cannot be discovered with the current datasets.

Limitations

The datasets used to create this models only considered the data from the power plant, due to the fact that the interval of the observations in the field was different, a more comprehensive model including the geothermal field can be created, using a different dataset. Also this research focused on predicting the power of the turbine, but predictive models can also be created for other sensors in the power plant.

An event predictor could not be created from this particular datasets, due to the small sample size of unscheduled events. A future attempt with a larger dataset and more frequent observations can be done. Having a larger dataset does not guarantee that an event predictor can be created.

Very little information on predictive methods for geothermal power plants was found for this research. Information for this type of studies is very restricted, due to the private character of the datasets. This research attempts to be one of the first data predictions approaches to geothermal power plants.

The predictive models created from the datasets, consider the assumption that the sensors are calibrated to meet standard requierments.

Future work

This research intends to be applied to Berlin geothermal field, the predictive model will be tested used up to date data. The final goal will be to create a model of the whole geothermal

field including the power plant, if the data capture conditions can be met. Having a model with the added conditions in the geothermal wells can be a great tool for creating expansion scenarios or production planning. A predictive model for each component and sensor will be created.

Checking the way the predictive models select variables can help updating the general thermodynamic model, by comparing which variables may be overlooked in the model being currently in use.

Future work regarding this research is described as follow:

- A predictive model will be created using a different dataset with more frequent observations and a larger number of sensors including the geothermal field and also testing different prediction techniques. The objective of this will be to effectively predict small variations on oncoming data, to preemptively adjust the system.
- A general model will be created for the main systems in the power plant in order to get the design parameters, this can help to check consistencies in the components being used in the power plant and to check if those comply with the manufacturer specifications.
- An on-line predictive model will also be created in order to detect variations in the thermodynamic system, which could help detecting operation problems that are difficult to detect. Scaling in pipelines could be detected with a proper dataset for instance avoiding the need of constant thickness measurements for instance.

References

Bradley, J., and Amde, M., 2015: Random Forests and Boosting in Mllib, website: https://databricks.com /blog/2015/01/21/random-forests-and-boosting-in-mllib.html

Breiman L., 2001: Random forest – Random features, Thecnical Report 567, Sept., 1999, Statistics Department University of California Berkeley, 29 pp.

Breiman L., Friedman J. H., Olshen R. A., and Stone, C. J., 1984: *Classification and Regression Trees.* Chapman and Hall, London, New York, Washington, D.C. 359 pp.

DiPippo, R., 2007: *Geothermal power plants. Principles, aplications, case studies and environmental impact* (2nd ed.). Butterworth Heineman, Elsevier, Kidlingiton, UK, 493 pp.

Guidos, J., and Burgos, J., 2012: Geothermal Activity and Development in El Salvador-Producing and developing. *Presented at "Short course on Geothermal Development and Geothermal Wells"*, organized by UNU-GTP and LaGeo, in Santa Tecla, El Salvador, March 11-17, 2012, 12 pp.

Hernandez Murga, C.B., 2012: *Aquifer fluid compositions at the Berlín geothermal field, El Salvador in 2012.* UNU-GTP, Iceland, report 12, 169-202 pp.

Henríquez Miranda, J.L., 1997: *Berlín geothermal project, preliminary power plant design.* UNU-GTP, Iceland, report 7, 173-194.

JICA, 2012: *The Project for Master Plan for the Development of Renewable Energy in the Republic of El Salvador*. Nippon Koey Co., Japan, final report.

Meisner, M. P., Wachs, M., Sambasivan, R. R., Zheng, A. X., and Ganger, G. R., 2009: *Modeling the Relative Fitness of Storage*, Carnegie Mellon University, 12 pp. website: http://pdl.cmu.edu/PDL-FTP/SelfStar/sigmetrics07.pdf

Molnar, P. and Sykes, L.R., 1969: Tectonics of the Caribbean and Middle America regions from focal mechanisms and seismicity. *Geological Society of America Bulletin, 80*, 1639-1684.

Rodriguez, A., and Herrera, A., 2003: Geothermal in El Salvador. *Geothermal Resources Council, Bulletin, July-Aug*. 159-162 pp.

SIGET, 2014: *Boletín de estadísticas eléctricas N° 15 2013*. Gerencia de Electricidad, San Salvador, website: http://estadisticas.cne.gob.sv/images/boletines/Boletines_SIGET/SIGET_2013.pdf.

Tedeschi, L. O., 2004: Assessment of the Adequacy of Mathematical Models. *Workshop on Mathematical Model Analysis and Evaluation, Sassari, Italy*, 28 pp.

Therneau, T. M., Atkinson, E. J. and Mayo Foundation, 2015: *An Introduction to Recursive Partitioning Using the RPART Routines*. Rstudio documentation, 62 pp.

UT, 2015: *Technical data base 2015*. Unidad de Transacciones S.A. de C.V., website: www.ut.com.sv

Yhat, 2013: Fitting & Interpreting Linear Models in R, website: http://blog.yhathq.com/posts/r-lm-summary.html