ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA (ICAI)

MÁSTER EN INGENIERÍA DE TELECOMUNICACIÓN

DEFINITION OF ANOMALY INDICATORS AND CONDITION PROGNOSIS IN COMPONENTS OF A HYDROPOWER PLANT

Autor: Beatriz García Alejo

Director: Miguel Ángel Sánz Bobi

Madrid

Julio 2018

Declaro, bajo mi responsabilidad, que el Proyecto presentado con el título

DEFINICIÓN DE INDICADORES DE ANOMALÍA Y CONDICIÓN DE PROGNOSIS EN UNA PLANTA HIDRÁULICA

en la ETS de Ingeniería - ICAI de la Universidad Pontificia Comillas en el

curso académico 2017/18 es de mi autoría, original e inédito y

no ha sido presentado con anterioridad a otros efectos.

El Proyecto no es plagio de otro, ni total ni parcialmente y la información que ha sido

tomada de otros documentos está debidamente referenciada.

Fdo.: Beatriz García Alejo Fecha: ……/ ……/ ……

Autorizada la entrega del proyecto

EL DIRECTOR DEL PROYECTO

Fdo.: Miguel Ángel Sánz-Bobi Fecha: ……/ ……/ ……

AUTORIZACIÓN PARA LA DIGITALIZACIÓN, DEPÓSITO Y DIVULGACIÓN EN RED DE PROYECTOS FIN DE GRADO, FIN DE MÁSTER, TESINAS O MEMORIAS DE BACHILLERATO

1º. Declaración de la autoría y acreditación de la misma.

El autor D. Beatriz García Alejo DECLARA ser el titular de los derechos de propiedad intelectual de la obra: Definición de indicadores de anomalía y condición de prognosis en una planta hidráulica, que ésta es una obra original, y que ostenta la condición de autor en el sentido que otorga la Ley de Propiedad Intelectual.

2º. Objeto y fines de la cesión.

Con el fin de dar la máxima difusión a la obra citada a través del Repositorio institucional de la Universidad, el autor **CEDE** a la Universidad Pontificia Comillas, de forma gratuita y no exclusiva, por el máximo plazo legal y con ámbito universal, los derechos de digitalización, de archivo, de reproducción, de distribución y de comunicación pública, incluido el derecho de puesta a disposición electrónica, tal y como se describen en la Ley de Propiedad Intelectual. El derecho de transformación se cede a los únicos efectos de lo dispuesto en la letra a) del apartado siguiente.

3º. Condiciones de la cesión y acceso

Sin perjuicio de la titularidad de la obra, que sigue correspondiendo a su autor, la cesión de derechos contemplada en esta licencia habilita para:

a) Transformarla con el fin de adaptarla a cualquier tecnología que permita incorporarla a internet y hacerla accesible; incorporar metadatos para realizar el registro de la obra e incorporar "marcas de agua" o cualquier otro sistema de seguridad o de protección.
b) Reproducirla en un soporte digital para su incorporación a una base de datos electrónica, incluyendo el derecho de reproducir y almacenar la obra en servidores, a los efectos de garantizar su seguridad, conservación y preservar el formato.
c) Comunicarla, por defecto, a través de un archivo institucional abierto, accesible de modo libre y gratuito a través de internet.
d) Cualquier otra forma de acceso (restringido, embargado, cerrado) deberá solicitarse expresamente y obedecer a causas justificadas.
e) Asignar por defecto a estos trabajos una licencia Creative Commons.
f) Asignar por defecto a estos trabajos un HANDLE (URL *persistente).*

4º. Derechos del autor.

El autor, en tanto que titular de una obra tiene derecho a:

a) Que la Universidad identifique claramente su nombre como autor de la misma
b) Comunicar y dar publicidad a la obra en la versión que ceda y en otras posteriores a través de cualquier medio.
c) Solicitar la retirada de la obra del repositorio por causa justificada.
d) Recibir notificación fehaciente de cualquier reclamación que puedan formular terceras personas en relación con la obra y, en particular, de reclamaciones relativas a los derechos de propiedad intelectual sobre ella.

5º. Deberes del autor.

El autor se compromete a:

a) Garantizar que el compromiso que adquiere mediante el presente escrito no infringe ningún derecho de terceros, ya sean de propiedad industrial, intelectual o cualquier otro.
b) Garantizar que el contenido de las obras no atenta contra los derechos al honor, a la intimidad y a la imagen de terceros.
c) Asumir toda reclamación o responsabilidad, incluyendo las indemnizaciones por daños, que pudieran ejercitarse contra la Universidad por terceros que vieran infringidos sus derechos e intereses a causa de la cesión.
d) Asumir la responsabilidad en el caso de que las instituciones fueran condenadas por infracción

de derechos derivada de las obras objeto de la cesión.

6º. Fines y funcionamiento del Repositorio Institucional.

La obra se pondrá a disposición de los usuarios para que hagan de ella un uso justo y respetuoso con los derechos del autor, según lo permitido por la legislación aplicable, y con fines de estudio, investigación, o cualquier otro fin lícito. Con dicha finalidad, la Universidad asume los siguientes deberes y se reserva las siguientes facultades:

- La Universidad informará a los usuarios del archivo sobre los usos permitidos, y no garantiza ni asume responsabilidad alguna por otras formas en que los usuarios hagan un uso posterior de las obras no conforme con la legislación vigente. El uso posterior, más allá de la copia privada, requerirá que se cite la fuente y se reconozca la autoría, que no se obtenga beneficio comercial, y que no se realicen obras derivadas.
- La Universidad no revisará el contenido de las obras, que en todo caso permanecerá bajo la responsabilidad exclusive del autor y no estará obligada a ejercitar acciones legales en nombre del autor en el supuesto de infracciones a derechos de propiedad intelectual derivados del depósito y archivo de las obras. El autor renuncia a cualquier reclamación frente a la Universidad por las formas no ajustadas a la legislación vigente en que los usuarios hagan uso de las obras.
- La Universidad adoptará las medidas necesarias para la preservación de la obra en un futuro.
- La Universidad se reserva la facultad de retirar la obra, previa notificación al autor, en supuestos suficientemente justificados, o en caso de reclamaciones de terceros.

Madrid, a 11 de Julio de 2018

ACEPTA

Fdo. Beatriz García Alejo

Motivos para solicitar el acceso restringido, cerrado o embargado del trabajo en el Repositorio Institucional:

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA (ICAI)

MÁSTER EN INGENIERÍA DE TELECOMUNICACIÓN

DEFINITION OF ANOMALY INDICATORS AND CONDITION PROGNOSIS IN COMPONENTS OF A HYDROPOWER PLANT

Autor: Beatriz García Alejo

Director: Miguel Ángel Sánz Bobi

Madrid

Julio 2018

Agradecimientos

A mi familia.

DEFINICIÓN DE INDICADORES DE ANOMALÍA Y CONDICIÓN DE PROGNOSIS EN COMPONENTES OF A PLANTA HIDRÁULICA

Autor: García Alejo, Beatriz.
Director: Sánz Bobi, Miguel Ángel.
Entidad Colaboradora: ICAI – Universidad Pontificia Comillas y SINTEF.

RESUMEN DEL PROYECTO

Este proyecto extiende el campo de *Business Intelligence* (BI) al área de energía hidroeléctrica con el fin de mejorar las operaciones funcionales y de mantenimiento de una turbina Kaplan específica. Tal tarea se ha logrado mediante el desarrollo de patrones normales de comportamiento y detección de anomalías de algunas de las variables clave de la central hidroeléctrica que influyen en su optimización.

Palabras clave: Turbina, Hidroeléctrico, MLPNN, SVM, RBF, Acumuladores de aceite, Patrones normales de comportamiento, Detección de anomalías, Estimación

1. Introducción

La energía hidroeléctrica es una de las principales fuentes de electricidad renovables que generó el 16,4% de la electricidad producida en el mundo en 2016 [1]. El óptimo y buen funcionamiento de las plantas hidroeléctricas es crítico para poder aumentar esta cifra. Sin embargo, la mayor parte de sus componentes son difíciles de inspeccionar. Por esta razón, este proyecto desarrolla modelos de BI que utilizan datos SCADA (valores promedio de una hora) para estimar los patrones normales de comportamiento de algunas de las variables clave de una planta determinada. Esto ayudaría a anticipar anomalías y realizar prevenciones para mejorar las condiciones óptimas de la estación.

2. Definición del proyecto

El objetivo principal de este proyecto es predecir patrones de comportamiento normales de algunas variables principales de una estación hidroeléctrica específica basados en sus datos reales. Éstos se han desarrollado mediante el uso de Redes Neuronal Perceptrón Multicapa (MLPNN), Máquinas de Vector de Soporte (SVM) y Funciones de Base Radial (RBF), donde MLPNN resulta ser el más preciso para predecir los atributos en cuestión. Los dos últimos métodos proporcionan consistencia a los resultados anteriores. A pesar de que utilizan distintos algoritmos matemáticos, todos los métodos aprenden el comportamiento de las variables a predecir en la fase de entrenamiento, en la que los valores esperados son conocidos. Después, éstos corrigen los conocimientos adquiridos con otros conjuntos de datos, donde se desconoce el resultado esperado [2].

Posteriormente se han desarrollado detectores de anomalías e indicadores del estado de salud, cuyos resultados son acordes a los previos patrones normales de comportamiento y son útiles para predecir posibles anomalías de la planta y medir su rendimiento.

3. Descripción del modelo/sistema/herramienta

Los modelos descritos en la sección anterior se han desarrollado para algunas de las principales variables que más influyen en el rendimiento de la estación: acumuladores

de aceite para el mecanismo de los álabes (AC1-GV, AC2-GV, AC3-GV), acumuladores de aceite para la turbina (AC1-TUR, AC2- TUR) y el tanque central de aceite (AC-TANK). Independientemente del método de BI, las entradas para el pronóstico de cada variable se han seleccionado en la fase de extracción de características, teniendo en cuenta el funcionamiento de la planta y el resultado de una determinada función de Matlab. Adicionalmente, AC-TANK también se ha estudiado con las predicciones de los otros modelos como entradas.

La Figura 1 muestra un diagrama que representa los modelos desarrollados con MLPNN. El mismo esquema se ha seguido con los métodos SVM y RBF.

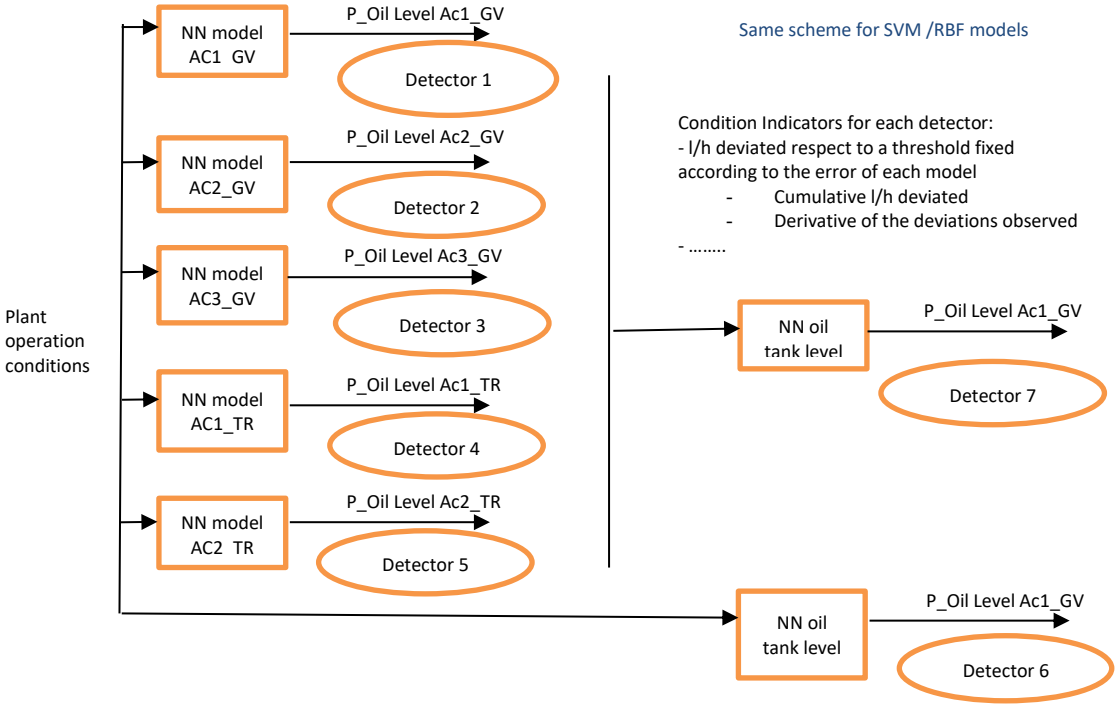


Figura 1. Esquema del desarrollo del proyecto para el método de MLPNN

Finalmente, se han llevado a cabo el detector de anomalías y dos indicadores del estado de salud para cada variable estimada y todos los métodos utilizados. El primer indicador está relacionado con la frecuencia de anomalías, pues mide el número de anomalías estimadas por hora. Lo importante de sus gráficas es el número de fluctuaciones: cuanto más sean, más anomalías hay. Por otro lado, el segundo indicador se centra en mostrar la gravedad de la anomalía representando el cambio de error en el tiempo. Por lo tanto, en este caso, lo interesante de sus gráficas es la altura de las fluctuaciones: cuanto más altas son, mayor diferencia entre los datos estimados y reales hay.

4. Resultados

La Figura 2 presenta los resultados de la predicción del patrón de comportamiento normal (a la izquierda), el detector de anomalías (en el medio) y los indicadores del estado de salud (a la derecha) obtenidos para el modelo AC1-GV. Los resultados son muy precisos para este caso: a la izquierda, el error en la predicción del patrón de comportamiento normal tiene una distribución normal centrada en cero. Además, el error máximo es menor al 1% (3 litros). En segundo lugar, AC1-GV presenta solo algunas pequeñas anomalías, ya que ni siquiera alcanzan el 1% de error. Finalmente, a partir de los resultados de los indicadores del estado de salud, las anomalías no son muy frecuentes

(primer gráfico a la derecha) y son pequeñas (segundo gráfico a la derecha), ya que ambos gráficos presentan inusuales fluctuaciones cortas.

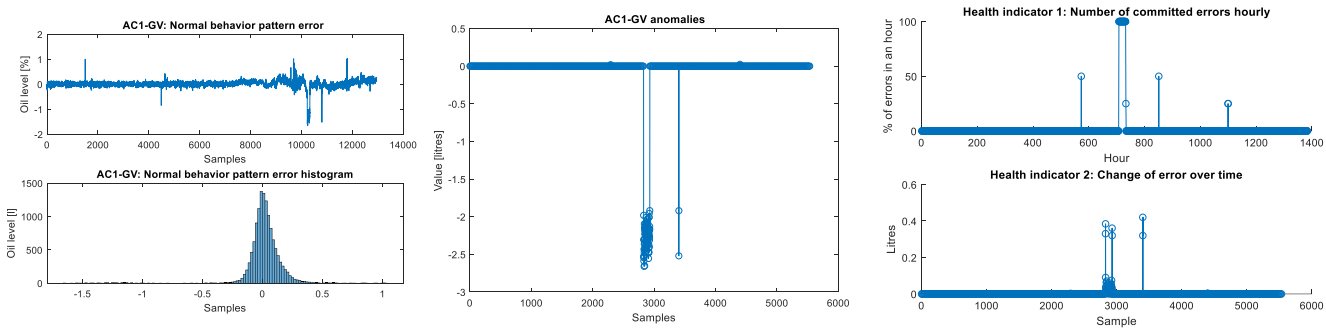


Figura 2. Resultados obtenidos de AC1-GV con MLPNN: patrones de comportamiento normal, anomalías e indicadores del estado de salud

El mismo procedimiento se ha llevado a cabo para los métodos SVM y RBF. Sus resultados no son tan exitosos como los de MPLNN, pero son lo suficientemente precisos como para consolidarlos y obtener conclusiones firmes (ver Tabla 1).

	MLPNN	*SVM*	*RBF*
Número de anomalías [%]	1.93%	9.44%	7.05%
Promedio de las anomalías [litros]	1.2	3.36	1.17

Table 1. Comparison between MPLNN, SVM and RBF models for AC1-GV

Paralelamente, los resultados del resto de las variables estudiadas se han comparado, de forma que mientras que algunos atributos obtienen resultados robustos y exitosos, como AC1-GV, otros no. Estos últimos presentan anomalías debido a posibles fugas de aceite.

5. Conclusiones

En general, las predicciones de patrones de comportamiento normales han logrado los principales objetivos de esta tesis, así como detectores de anomalías e indicadores de salud. Sin embargo, algunas de las variables estudiadas, como AC3-GV, presentan anomalías que pueden deberse a una fuga de aceite. Tal comportamiento influye en el nivel del tanque central de aceite, por lo que también presenta anomalías frecuentes. Las anomalías no significan que las predicciones de patrones de comportamiento normales sean incorrectas, sino que algunos atributos sufren un comportamiento anormal.

En consecuencia, este proyecto da lugar a investigación que explique el origen de estas anomalías como trabajo futuro, así como la aplicación de los modelos desarrollados a nuevos conjuntos de datos para consolidar los resultados. Además, este proyecto conduce a la implementación de los modelos desarrollados en la planta hidroeléctrica para que pueda ejecutarse a lo largo del tiempo.

6. Referencias

[1] World Energy Council, «Energy sources: Hydropower,» 2017. [En línea]. Disponible: https://www.worldenergy.org/data/resources/resource/hydropower.

[2] C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

DEFINITION OF ANOMALY INDICATORS AND CONDITION PROGNOSIS IN COMPONENTS OF A HYDROPOWER PLANT

Author: García Alejo, Beatriz.
Supervisor: Sánz Bobi, Miguel Ángel.
Collaborating Entity: ICAI – Pontifical University of Comillas and SINTEF.

ABSTRACT

This project extends Business Intelligence (BI) field to hydroelectric power area in order to improve functional and maintenance operations of a specific Kaplan turbine. Such task has been achieved by the normal behavior patterns and anomalies detection development for some of the key hydroelectric plant's variables which influence its optimization.

Keywords: Turbine, Hydroelectric, MLPNN, SVM, RBF, Oil accumulators, Normal behavior patterns, Anomalies detection, Estimation.

1. Introduction

As the leading renewable electricity source, hydropower importance is increasing over time. Indeed, it generated 16.4% of the electricity produced in the world from all sources in 2016 [1]. Such relevant amount reveals the importance of working on getting the best operation in hydropower plants. However, most components from these stations are difficult to inspect. For this reason, this project develops some Business Intelligence models which uses SCADA data (one-hour average values) to estimate normal behavior patterns of some variables from a determined plant. This would help to anticipate anomalies and conduct some preventions to keep on with the optimal station conditions.

2. Project definition

This project aims to develop estimations of some normal behavior patterns based on retrieved data of some principal variables of a specific hydropower station. They are achieved by using Business Intelligence methods: Multilayer Perceptron Neural Network (MLPNN), Support Vector Machines (SVM) and Radial Basis Functions (RBF), where MLPNN results to be the most accurate one for the attributes to predict. The two latter ones are required in order to provide MPLNN output with consistency. Despite they calculate estimations basing them on different mathematical algorithms, all methods start learning variables behavior by previous training phase, when expected values are known. Then, they correct their acquired knowledge with other data sets, where expected output is unknown [2].

Once normal behavior patterns are obtained, anomalies detectors and health indicators have been programmed. They support previous result and are useful to predict abnormal behavior of the plant as well as to measure its performance.

3. Model description

Described models in previous section have been developed for several different attributes from the hydroelectric plant station. These are considered to be some of the principal variables which most influence on the station performance and they are guide vanes

mechanism oil levels accumulators (AC1-GV, AC2-GV, AC3-GV), turbine runner accumulators (AC1-TUR, AC2-TUR), and oil tank hub (AC-TANK). Regardless the Business Intelligence method, inputs for every variable's prognosis have been selected in the feature extraction phase by considering plant's operation and a Matlab function's results.

Moreover, apart from estimating normal behavior patterns considering real data as inputs, AC-TANK has also been studied with the others predicted models as inputs.

Figure 1 depicts a diagram which represents developed models with MLPNN. Same scheme has been followed with SVM and RBF methods.

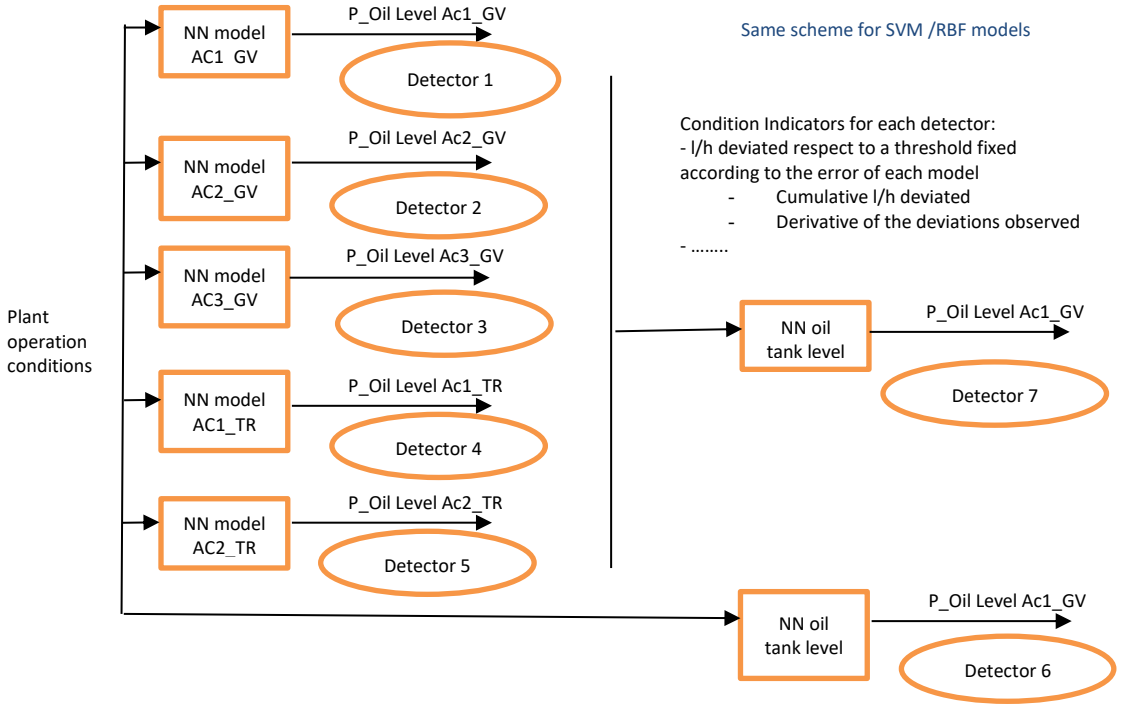


Figure 1. Diagram of the project developed tasks with MLPNN.

Finally, anomalies detector and two health indicators have been carried out for every estimated variable and all used methods. The first health indicator defines a signal which measures how many anomalies are predicted hourly in percentage. Therefore its result is related to anomalies frequency. The important point of its graphs is the number of fluctuations: the more, the more anomalies are. On the other hand, second health indicator is focused on showing how serious anomaly is by showing how error is changing over time. Hence, for this one, the most important point to be focus on its graphs is the fluctuations' height: the higher they are, the greater difference between estimated and real data is.

4. Results

Figure 2 presents results from the normal behavior pattern prediction (on the left), anomalies detector (on the middle) and health indicators (on the right) obtained for the AC1-GV model. Results are very accurate for this case: on the left, error on normal behavior pattern prediction has a normal distribution centered in cero. Besides, maximum error is less than 1% (3 litres). Secondly, in line with previous output, AC1-

GV presents only few small anomalies, since they don't even reach 1% of error. Finally, from health indicators results, anomalies are not very frequent (first graph on the right) and they are small (second graph on the right), since both graphs present unusual short fluctuations.

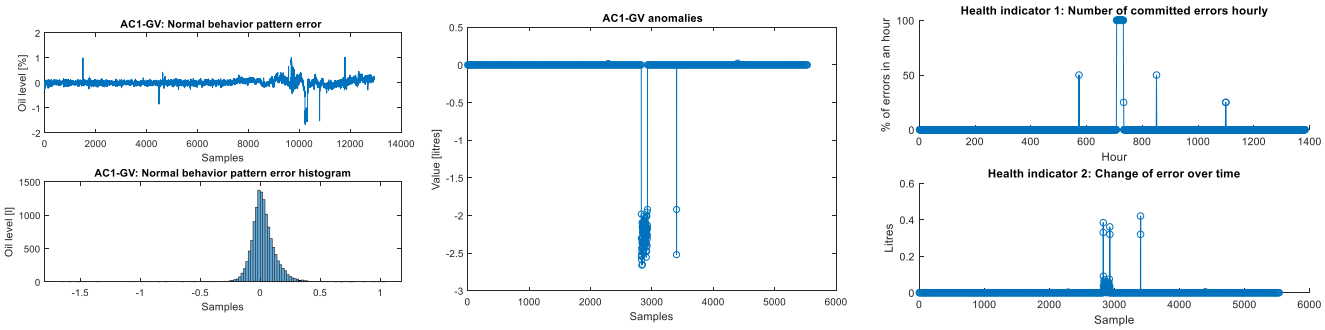


Figure 2. MLPNN Results for AC1-GV: predicted normal behavior pattern, anomalies detection and health indicator

Same procedure has been followed for SVM and RBF methods. Their results are not as successful as MPLNN's ones, but they are accurate enough to strengthen them and obtain firm conclusions (See Table 1).

	MLPNN	*SVM*	*RBF*
Number of anomalies [%]	1.93%	9.44%	7.05%
Mean of the anomalies (litres)	1.2 litres	3.36 litres	1.17 litres

Table 2. Comparison between MPLNN, SVM and RBF models for AC1-GV

Similarly, results for the rest studied target variables have been compared. While some gets robust and successful results, as AC1-GV, others don't. These latter ones may present anomalies due to oil leakages which need further research.

5. Conclusions

All in all, normal behavior patterns predictions have achieved the main objectives of this thesis, as well as anomalies detectors and health indicators, as shown in previous section. However some target variables, such as AC3-GV present burst anomalies which may be due to oil leakage. Such abnormal behavior influences oil tank hub level and results also reveal usual anomalies. Anomalies don't mean that normal behavior patterns predictions are wrong but that some attributes are suffering from abnormal behavior.

Consequently, further research which explains the origin of these anomalies is required as future work, as well as applying models to more unknow data sets to ensure results. Besides, this project leads to the implementation of the developed research in the plant so that it can be executed over time.

6. References

[1] World Energy Council, «Energy sources: Hydropower,» 2017. [En línea]. Available: https://www.worldenergy.org/data/resources/resource/hydropower.

[2] C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

Thesis Index

Figures Index

Tables Index

Chapter 1. INTRODUCTION

1.1 PROJECT MOTIVATION

Nowadays, energy can be seen in every corner, since the moment when somebody gets up in the morning and switches off the alarm until they go to bed in the night and turn off the lights. Hence, energy is considered to be one of the most important society's dependence.

Within time, most energy has been produced by CO_2, oil and other fossil combustibles. However it has been demonstrated that abusing their use brings noxious consequences worldwide, such as the global warming or the greenhouse gases' emissions, which are provoking a hole in the O_3 layer. Popular awareness related to this issue is increasing every day and, as a result, society has already started using more convenient products for the environment and substituting these sources by renewable energies.

Nevertheless, renewable resources are almost new and therefore, they are not as well-developed as the traditional electricity sources. This means that popular worry about the World welfare is not enough, but a hard work and development of renewable energies is also needed.

At this point, projects like this master thesis born with the main goal of getting as most profit from renewable energies as possible. This project is focused on the hydropower, which is one of the most valuable renewable energies due to its high availability. In addition, hydropower energy's responses to any fluctuation in demand, either shortage or abundance, as well as its change in the power grid are fast.

This source is being exploited and studied in several regions, but Norway is one of the countries which most enjoys the presence of water. This is the reason why SINTEF, a Norwegian company from Trondheim, is getting advantage of this fact and investing in projects which enhance hydropower plants' performances. This action is not only achieved

by improving the mechanics of the turbines and others parts of any plant, but also by measuring some variables periodically, such as hourly, obtaining big data sets, and doing a thorough research of their relationship. Additionally, inspecting components and others part of a hydropower plant system is a difficult process. So models based on machine learning from data sets retrieved in the plant in order to monitor the system condition as well as to detect failures are valuable.

This big amount of data retrieved by sensors on hydropower plants have fed this project, which uses Business Intelligence methods to analyze it and get to some conclusion for improving their performance. Furthermore, this data is already logged in the Supervisory Control And Data Acquisition (SCADA) [1], which eases the work. Particularly, this project is focused on a specific plant, called E4, close to Trondheim and has the main objective of analyzing different measured variables in order to get a diagnosis of its normal behaviour and a model which describes its most optimum performance. Moreover, it will conclude with a research of anomalies in its performance, so that a health indicator could be developed.

Chapter 2. PRECEDENT WORK

Some professionals started the implementation of Business Analytics in the determined E4 hydropower plant some months before this project have started. Their main purpose was similar to the ones described for this thesis in Chapter 4. : to develop models and algorithms for condition monitoring and the detection of faults in hydropower equipment [1]. Both projects are based on machine learning and artificial intelligence. Indeed, they belong to the research project MonitorX - "Optimal utilization of hydropower asset lifetime by monitoring of technical condition and risk" [1].

However this work finished by the prediction of some attributes such as the power generated in the power plant and the oil tank level. The error for those predictions were minimal so that contributed to infer the plat was suffering from some oil leakages.

As a result, this project was born as a continuation of the already initiated research. Moreover at this point it is reasonable that the thesis is focused on the different oil levels of the hydropower plant.

Chapter 3. DESCRIPTION OF THE HYDRAULIC POWER PLANT

This work covers cases that are from a specific hydropower plant in Norway which uses a Kaplan turbine for the electric energy production.

The main feature of a Kaplan turbine is its adjustable runner blades. They have the main advantage of maintaining a high efficiency at the same time they are operating with different water flows at varying head which origin different power outputs [2].

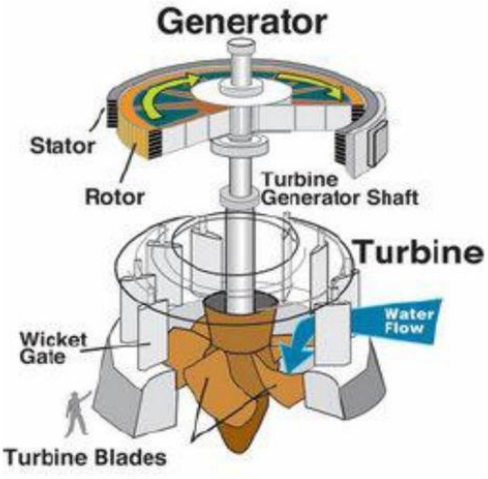


Figure 3. Kaplan turbine [3]

The functioning of the Kaplan turbine is based on the control taken over by turbine runner blades (turbine blades) and wicket gates (guide vanes) operation (Figure 3) [1]. Both turbine blades and wicket gates are the mentioned adjustable runner blades. Their opening angle is controlled by a turbine regulator with a correlated movement between them. The regulator adjusts their position according a combination curves for both the turbine runner and the guide vanes which is predefined based on some information about head and flow. Both components of the Kaplan turbine can be adjustable thanks to a high-pressure hydraulics

acting mechanism. It contains a high-pressure unit (HPU) and an accumulator tank which provides oil to let servomotors act [1].

The high-pressure hydraulic system is mainly composed by [1] [2]:

- Governator oil sump tank with oil pumps (HPU – High Pressure Unit)
- Pressure accumulator banks. One bank for runner blades and one bank for the wicket gates
- Hydraulic oil cooling/heating system
- Wicket gate control system
- Runner blade control system
- Quick stop / Emergency stop system
- Oil system for runner hub. The runner hub is the lowermost part of the runner (Figure 1), which appears below the turbine blades with a cone-shape.

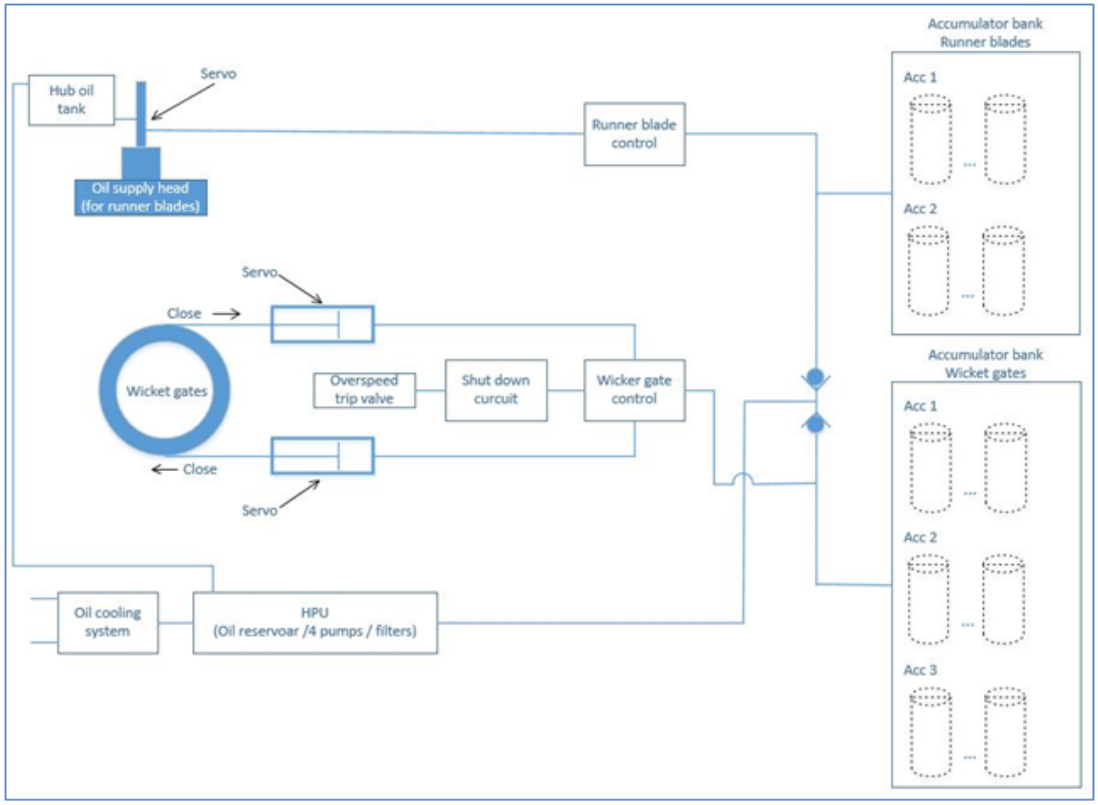


Figure 4. Simplified view of a hydraulic system. [1]

The oil sump must be large enough to store all the oil in the hydraulic system. However oil is distributed in all the different hydraulic system component during operation so only a little percentage of oil is in the reservoir. In spite of this oil distribution, a minimum oil level in the sump is required in order to avoid the HPU pumps become dry.

According to Figure 4, HPU is located at the floor of the hydraulic system and its main components are the oil reservoir or sump, oil pumps (with filter system to maintain the oil clean), filters, valves and coolers. Its tasks are to provide wicket gates and runner blades with high-pressure oil and supply oil to the control system, which consists in five accumulator banks. Besides, HPU counts on with monitorization of the oil level, water-in-oil content and temperature [1].

Cooling and heating systems are required in order to cool the oil during operation and heat it during passive state.

Wicket gates are controlled by two servos that stimulate the control ring which surrounds them. Gates are open or close depending on this ring action. For example, they close when the ring becomes clockwise, which means some stones may be stuck in between. Guides vanes' angle are set up to increase or decrease the water which is flowing to the turbine runner [2].

Related to the runner blade regulation, it is set up by using another servo actuator which is located in the runner hub. The oil that is supplied or returned to this servo is routed through the center of the turbine shaft, passing by the oil supply head at the top of the shaft [1] [2].

Furthermore, there is a system stop the turbine due to emergency reasons in the hydraulic system. It can be activated either manually or automatically when the turbine is overspeeding [1] [2].

Last, the turbine hub contains oil and is characterized by a higher oil pressure than the surrounding water pressure to prevent water from getting in the hub. The oil in the hub comes

from the HPU reservoir. However both hub oil and its tank are not part of the high-pressure circuit so if a leakage in the turbine runner blade servo exists, the oil level in the hub oil tank gets affected and will an alarm goes on. [1]

All in all, water gets in from the head level of the weir to the turbine through a spiral. Due to its hydrodynamic profile, water is lead to the guide vanes, which are configurated by the described system to be open with the optimal angle. Then, water reaches the runner and hydraulic energy is transformed into mechanical energy. The turbine is also dynamically adjusted to obtain much power as possible. Finally, water is guided back to the tail level of the river via some pipes through hydro generator shaft. [4]

Further information related to this kind of hydropower plant can be found in the source [2].

Chapter 4. PROJECT DEFINITION

4.1 JUSTIFICATION

The origin of this project lies on the high importance renewable energy sources are gaining every day. More specifically, Norway is a country which is especially focused on hydropower due to its immense quantities of water.

In this sense, the justification of this thesis is the study of maximizing the profit of hydropower plants by using Business Intelligence methods. This is applied to the plants once they have already been built and every mechanical part is finished and ready to perform its determined function to get energy from the water movement.

Not only Business Intelligence would help to improve the hydropower plants performance but also to define their normal behavior and identifying anomalies. These would make workers to be aware of any existing failure at the present moment so that they could set oil level, reestablish some elements, restart the plant or carry out any required action to correct their behavior and maximize the power profit.

Moreover, some reports may be exported as getting the health indicators of the plants. They would be useful to monitor their behavior and analyzing if they have been getting better or worse results over time. Workers would be able to compare the different situations when the profit was maximized and when it wasn't, obtaining some conclusions about which elements are not working properly.

This project is focused on a specific plant which has a Kaplan turbine and it is considered to be the first step to improve its performance by analyzing the data already logged in SCADA as hourly mean values and that sensors are retrieving periodically. Consequently it is one of the first steps to implement Business Intelligence methods in the area. Indeed it is the seed to another project which would consist in collecting more valid data and doing a research to

find the perfect location to add a microchip which could do the determined math from the retrieved values to output a thorough prediction. This would make possible the anticipation to any failure in the plant and would improve much more the profit and the plant performance.

4.2 OBJETIVES

The principal objective of this project is to develop a prognosis and detection of anomalies of the hydropower plant behaviour based on its monitoring information. Moreover, in order to achieve this one some other goals must be encountered.

To be specific, the variables which are going to be studied are the following ones:

- Accumulator 1 guide vanes operating mechanism oil level (AC1-GV).
- Accumulator 2 guide vanes operating mechanism oil level (AC2-GV),
- Accumulator 3 guide vanes operating mechanism oil level (AC3-GV),
- Accumulator 1 turbine runner oil level (AC1-TUR),
- Accumulator 2 turbine runner oil level (AC2-TUR),
- Hub tank oil (AC-TANK), based on real values from other attributes and from predictions of previous listed items.

First step to achieve the principal aim of this thesis consists in analyzing, preparing and filtering data in order to ensure that only valid data would be taken to the project development.

Furthermore, relationships among the different attributes must be studied in order to determine how much they could explain to the value of the outputs that are desired to be predicted. This target is required in order to discriminate those variables which do not contribute to the prediction of any important attribute.

Last described goals are mandatory in order to develop an accurate prediction of some variables, such as the three guide vanes accumulators' oil levels, the two turbine runner

accumulators' oil levels and the tank oil level. These predictions form the base to develop the prognosis of the mentioned oil levels in the short-term evolution.

Thanks to the definition of the normal behavior of the different oil levels, anomalies would be easily detected. That is why another goal would be the development of a program which could identify and make people aware of the abnormal behaviour in the hydropower plant. However, the program would not tell the reason of the anomaly but only the variable which is suffering from atypical values.

Finally, another program would be carried out in order to let compare the plant behaviour with itself over time. This program would be based on the development of a health indicator for every oil level in the plant.

Besides, it must be mentioned that every objective would be developed in three different Business Intelligence models in order to strengthen the results and conclusion of this thesis.

All in all, Figure 6 presents a diagram which summarizes the final result that this project ends with. It is only the representation of the outputs for one of the methods that are used, Multilater Perceptron Neural Network (MLPNN). However this scheme should be repeated with Support Vector Machines (SVM) and Radial Basis Functions (RBF) to make conclusions consistent.

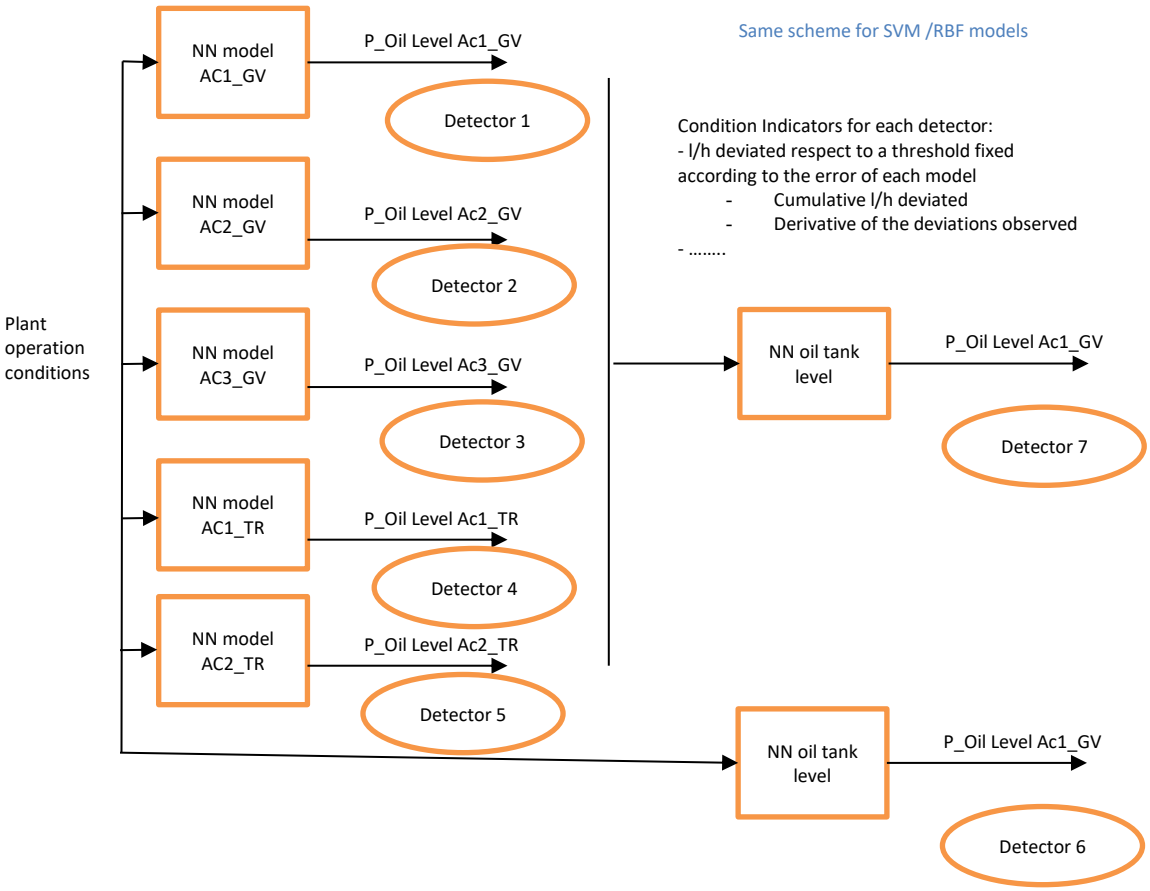


Figure 5. Diagram of the output of this project with MLPNN.

4.3 METHODOLOGY

In order to achieve the previous objectives described in the last section, a thorough scheme as the one explained below has been followed:

1. ***Analysis and filtering of data.***

   After observing the data: the name of the attributes, the maximum and minimum values they may have, its boxplot, histograms, etc., enough knowledge has been obtained in order to filter the outliers and zeros. They are not needed in the rest of the project because their information would provide error since they are not considered normal values for the variables.

2. ***Feature extraction.***

   This stage is a required step before the study of the variables prediction because thank to the feature extraction it is possible to know how every attribute may influence to the others. In other words, this step facilitates the best inputs to predict different outputs for the developed Business Intelligence models.

   This step also includes the validation of the training and testing sets, which means that every value of the input from the testing sets must be included in the values that the training set contains for every specific input attribute.

3. ***Definition of a prognosis.***

   Different oil levels are predicted in this phase according to the result of the feature extraction. The normal behavior is defined by using Multilayer Neural Networks, basically. However this method is supported by Support Vector Machines and Radial Basis Functions. In every developed model, different executions are tried in order to select the most accurate one, with their attributes, such as the number of neurons, adjusted.

4. ***Anomalies detection.***

   Once the prognosis for every oil level is defined and can be predicted, anomalies detection is carried out. This step is also developed for every different Business Intelligence method which have been used in the previous phase in order to strengthen the results.

5. ***Health Indicator.***

   Finally, health indicators are developed for every output that has been predicted in the project and for every model which has been used for that task. The goal of this stage is to be able to infer if the hydropower plant has been improving or not over time. If not, professional would have to work on it and correct the failures.

4.4 PLANIFICATION AND ECONOMIC ESTIMATION

This section presents the planification followed during the project development by a Gantt Diagram [5] and an economic estimation through different tables and calculations.

4.4.1 GANTT CHART

A Gantt Chart is helpful to plan the duration of every phase in a project since it takes into account the dependences among them as well as the available resources. The following Figure 6 shows the summary of the thesis' principal tasks.

	Nombre	Duracion	Inicio	Terminado
1	**Study of the hydropower plant**	7,875 days?	**22/01/18 9:00**	**31/01/18 17:00**
4	**Review of the state of art**	16 days?	**1/02/18 8:00**	**22/02/18 17:00**
8	**Analysis of data**	5 days?	**23/02/18 8:00**	**1/03/18 17:00**
12	Feature extraction	1 day?	2/03/18 8:00	2/03/18 17:00
13	Validation of inputs	1 day?	2/03/18 9:00	5/03/18 9:00
14	Analysis of Data Report	3 days	5/03/18 9:00	8/03/18 9:00
15	**Definition of a prognosis**	19 days?	**8/03/18 9:00**	**4/04/18 9:00**
24	**Anomalies detection**	13 days?	**5/04/18 8:00**	**23/04/18 17:00**
33	**Health indicator**	15 days	**24/04/18 8:00**	**14/05/18 17:00**
42	Conclusions and Future Work	4 days	11/06/18 8:00	14/06/18 17:00
43	Masther Thesis Report	15 days	15/06/18 8:00	5/07/18 17:00
44	Presentation	1 day?	17/07/18 7:00	17/07/18 17:00

Figure 6. Summary of the project main phases planification.

To continue, every phase is scrolled out in order to show in different images it planification in more detail:

	Nombre	Duracion	Inicio	Terminado
1	**Study of the hydropower plant**	7,875 days?	**22/01/18 9:00**	**31/01/18 17:00**
2	Plant elements and performance	4,875 days?	22/01/18 9:00	26/01/18 17:00
3	Description of the hydropower plant	3 days?	29/01/18 8:00	31/01/18 17:00
4	**Review of the state of art**	16 days?	**1/02/18 8:00**	**22/02/18 17:00**
8	**Analysis of data**	5 days?	**23/02/18 8:00**	**1/03/18 17:00**
12	Feature extraction	1 day?	2/03/18 8:00	2/03/18 17:00
13	Validation of inputs	1 day?	2/03/18 9:00	5/03/18 9:00
14	Analysis of Data Report	3 days	5/03/18 9:00	8/03/18 9:00
15	**Definition of a prognosis**	19 days?	**8/03/18 9:00**	**4/04/18 9:00**
24	**Anomalies detection**	13 days?	**5/04/18 8:00**	**23/04/18 17:00**
33	**Health indicator**	15 days	**24/04/18 8:00**	**14/05/18 17:00**
42	Conclusions and Future Work	4 days	11/06/18 8:00	14/06/18 17:00
43	Masther Thesis Report	15 days	15/06/18 8:00	5/07/18 17:00
44	Presentation	1 day?	17/07/18 7:00	17/07/18 17:00

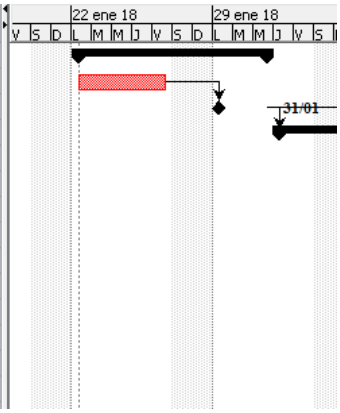


Figure 7. Study of the hydropower plant phase planification.

	Nombre	Duracion	Inicio	Terminado
1	**Study of the hydropower plant**	7,875 days?	**22/01/18 9:00**	**31/01/18 17:00**
4	**Review of the state of art**	16 days?	**1/02/18 8:00**	**22/02/18 17:00**
5	Failure modes in Kaplan hydropower	8 days	1/02/18 8:00	12/02/18 17:00
6	Patterns recognition techniques	7 days	13/02/18 8:00	21/02/18 17:00
7	State of art review	1 day?	22/02/18 8:00	22/02/18 17:00
8	**Analysis of data**	5 days?	**23/02/18 8:00**	**1/03/18 17:00**
12	Feature extraction	1 day?	2/03/18 8:00	2/03/18 17:00
13	Validation of inputs	1 day?	2/03/18 9:00	5/03/18 9:00
14	Analysis of Data Report	3 days	5/03/18 9:00	8/03/18 9:00
15	**Definition of a prognosis**	19 days?	**8/03/18 9:00**	**4/04/18 9:00**
24	**Anomalies detection**	13 days?	**5/04/18 8:00**	**23/04/18 17:00**
33	**Health indicator**	15 days	**24/04/18 8:00**	**14/05/18 17:00**
42	Conclusions and Future Work	4 days	11/06/18 8:00	14/06/18 17:00
43	Masther Thesis Report	15 days	15/06/18 8:00	5/07/18 17:00
44	Presentation	1 day?	17/07/18 7:00	17/07/18 17:00

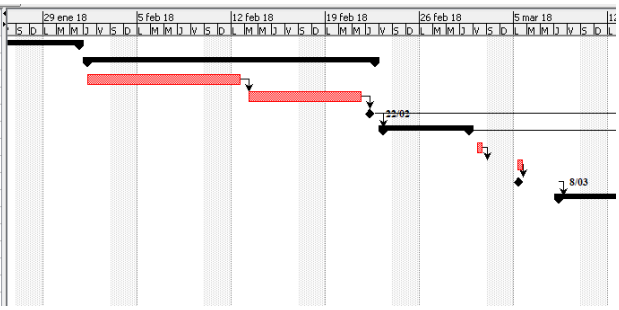


Figure 8. Review of the state of art phase planification.

	Nombre	Duracion	Inicio	Terminado
1	**Study of the hydropower plant**	7,875 days?	**22/01/18 9:00**	**31/01/18 17:00**
4	**Review of the state of art**	16 days?	**1/02/18 8:00**	**22/02/18 17:00**
8	**Analysis of data**	5 days?	**23/02/18 8:00**	**1/03/18 17:00**
9	Data collection	1 day?	23/02/18 8:00	23/02/18 17:00
10	Filtering of data	1 day?	26/02/18 8:00	26/02/18 17:00
11	Definition of training and testing sets	3 days	27/02/18 8:00	1/03/18 17:00
12	Feature extraction	1 day?	2/03/18 8:00	2/03/18 17:00
13	Validation of inputs	1 day?	2/03/18 9:00	5/03/18 9:00
14	Analysis of Data Report	3 days	5/03/18 9:00	8/03/18 9:00
15	**Definition of a prognosis**	19 days?	**8/03/18 9:00**	**4/04/18 9:00**
24	**Anomalies detection**	13 days?	**5/04/18 8:00**	**23/04/18 17:00**
33	**Health indicator**	15 days	**24/04/18 8:00**	**14/05/18 17:00**
42	Conclusions and Future Work	4 days	11/06/18 8:00	14/06/18 17:00
43	Masther Thesis Report	15 days	15/06/18 8:00	5/07/18 17:00
44	Presentation	1 day?	17/07/18 7:00	17/07/18 17:00

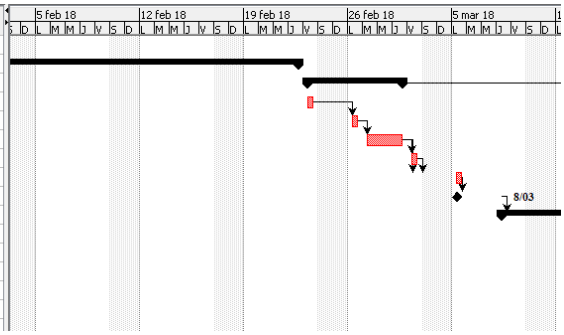
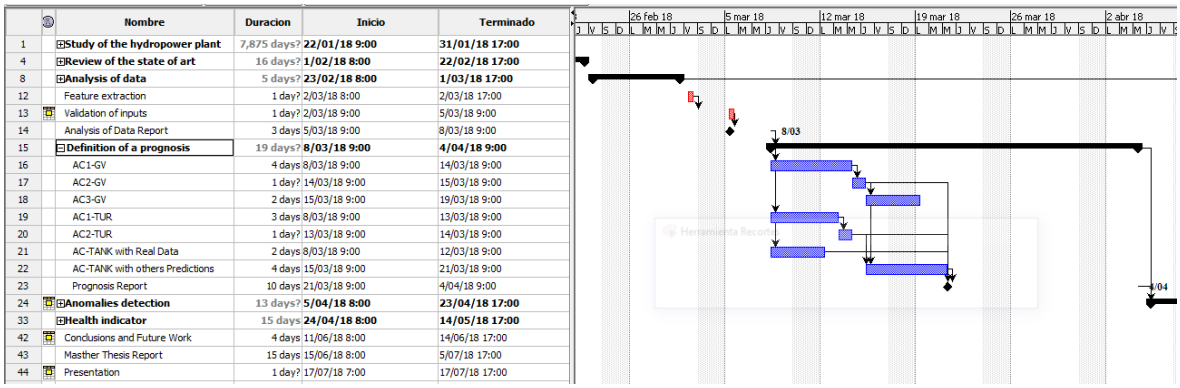
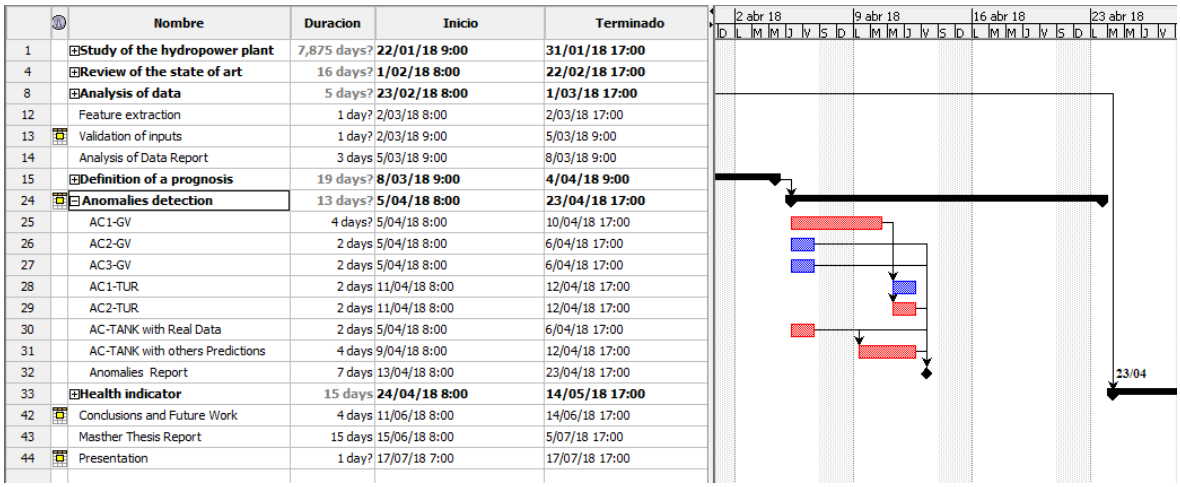


Figure 9. Analysis of data, Data collection and filtering and Definition of training and testing sets phases planification.



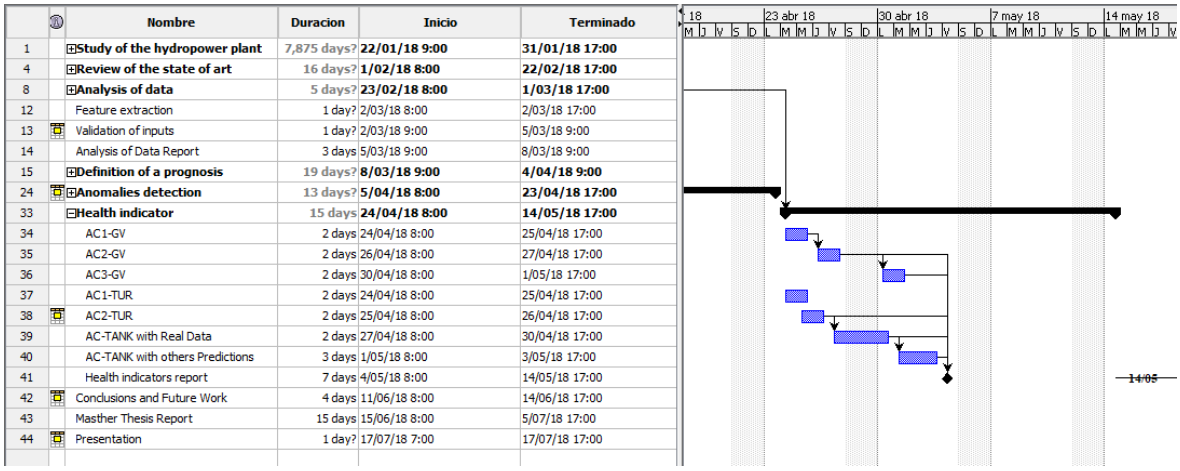
	Nombre	Duracion	Inicio	Terminado
1	**Study of the hydropower plant**	7,875 days?	**22/01/18 9:00**	**31/01/18 17:00**
4	**Review of the state of art**	16 days?	**1/02/18 8:00**	**22/02/18 17:00**
8	**Analysis of data**	5 days?	**23/02/18 8:00**	**1/03/18 17:00**
12	Feature extraction	1 day?	2/03/18 8:00	2/03/18 17:00
13	Validation of inputs	1 day?	2/03/18 9:00	5/03/18 9:00
14	Analysis of Data Report	3 days	5/03/18 9:00	8/03/18 9:00
15	**Definition of a prognosis**	19 days?	**8/03/18 9:00**	**4/04/18 9:00**
16	AC1-GV	4 days	8/03/18 9:00	14/03/18 9:00
17	AC2-GV	1 day?	14/03/18 9:00	15/03/18 9:00
18	AC3-GV	2 days	15/03/18 9:00	19/03/18 9:00
19	AC1-TUR	3 days	8/03/18 9:00	13/03/18 9:00
20	AC2-TUR	1 day?	13/03/18 9:00	14/03/18 9:00
21	AC-TANK with Real Data	2 days	8/03/18 9:00	12/03/18 9:00
22	AC-TANK with others Predictions	4 days	15/03/18 9:00	21/03/18 9:00
23	Prognosis Report	10 days	21/03/18 9:00	4/04/18 9:00
24	**Anomalies detection**	13 days?	**5/04/18 8:00**	**23/04/18 17:00**
33	**Health indicator**	15 days	**24/04/18 8:00**	**14/05/18 17:00**
42	Conclusions and Future Work	4 days	11/06/18 8:00	14/06/18 17:00
43	Masther Thesis Report	15 days	15/06/18 8:00	5/07/18 17:00
44	Presentation	1 day?	17/07/18 7:00	17/07/18 17:00

Figure 10. Definition of prognosis phase planification.



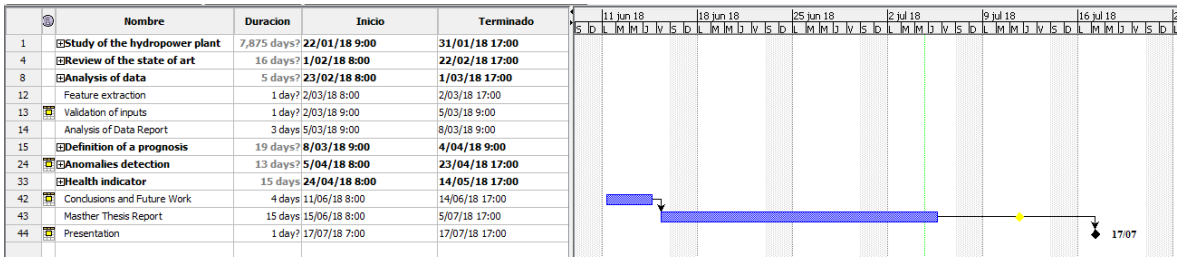
	Nombre	Duracion	Inicio	Terminado
1	**Study of the hydropower plant**	7,875 days?	**22/01/18 9:00**	**31/01/18 17:00**
4	**Review of the state of art**	16 days?	**1/02/18 8:00**	**22/02/18 17:00**
8	**Analysis of data**	5 days?	**23/02/18 8:00**	**1/03/18 17:00**
12	Feature extraction	1 day?	2/03/18 8:00	2/03/18 17:00
13	Validation of inputs	1 day?	2/03/18 9:00	5/03/18 9:00
14	Analysis of Data Report	3 days	5/03/18 9:00	8/03/18 9:00
15	**Definition of a prognosis**	19 days?	**8/03/18 9:00**	**4/04/18 9:00**
24	**Anomalies detection**	13 days?	**5/04/18 8:00**	**23/04/18 17:00**
25	AC1-GV	4 days?	5/04/18 8:00	10/04/18 17:00
26	AC2-GV	2 days	5/04/18 8:00	6/04/18 17:00
27	AC3-GV	2 days	5/04/18 8:00	6/04/18 17:00
28	AC1-TUR	2 days	11/04/18 8:00	12/04/18 17:00
29	AC2-TUR	2 days	11/04/18 8:00	12/04/18 17:00
30	AC-TANK with Real Data	2 days	5/04/18 8:00	6/04/18 17:00
31	AC-TANK with others Predictions	4 days	9/04/18 8:00	12/04/18 17:00
32	Anomalies Report	7 days	13/04/18 8:00	23/04/18 17:00
33	**Health indicator**	15 days	**24/04/18 8:00**	**14/05/18 17:00**
42	Conclusions and Future Work	4 days	11/06/18 8:00	14/06/18 17:00
43	Masther Thesis Report	15 days	15/06/18 8:00	5/07/18 17:00
44	Presentation	1 day?	17/07/18 7:00	17/07/18 17:00

Figure 11. Anomalies detection phase planification.



	Nombre	Duracion	Inicio	Terminado
1	**Study of the hydropower plant**	7,875 days?	**22/01/18 9:00**	**31/01/18 17:00**
4	**Review of the state of art**	16 days?	**1/02/18 8:00**	**22/02/18 17:00**
8	**Analysis of data**	5 days?	**23/02/18 8:00**	**1/03/18 17:00**
12	Feature extraction	1 day?	2/03/18 8:00	2/03/18 17:00
13	Validation of inputs	1 day?	2/03/18 9:00	5/03/18 9:00
14	Analysis of Data Report	3 days	5/03/18 9:00	8/03/18 9:00
15	**Definition of a prognosis**	19 days?	**8/03/18 9:00**	**4/04/18 9:00**
24	**Anomalies detection**	13 days?	**5/04/18 8:00**	**23/04/18 17:00**
33	**Health indicator**	15 days	**24/04/18 8:00**	**14/05/18 17:00**
34	AC1-GV	2 days	24/04/18 8:00	25/04/18 17:00
35	AC2-GV	2 days	26/04/18 8:00	27/04/18 17:00
36	AC3-GV	2 days	30/04/18 8:00	1/05/18 17:00
37	AC1-TUR	2 days	24/04/18 8:00	25/04/18 17:00
38	AC2-TUR	2 days	25/04/18 8:00	26/04/18 17:00
39	AC-TANK with Real Data	2 days	27/04/18 8:00	30/04/18 17:00
40	AC-TANK with others Predictions	3 days	1/05/18 8:00	3/05/18 17:00
41	Health indicators report	7 days	4/05/18 8:00	14/05/18 17:00
42	Conclusions and Future Work	4 days	11/06/18 8:00	14/06/18 17:00
43	Masther Thesis Report	15 days	15/06/18 8:00	5/07/18 17:00
44	Presentation	1 day?	17/07/18 7:00	17/07/18 17:00

Figure 12. Health indicator phase planification



	Nombre	Duracion	Inicio	Terminado
1	**Study of the hydropower plant**	7,875 days?	**22/01/18 9:00**	**31/01/18 17:00**
4	**Review of the state of art**	16 days?	**1/02/18 8:00**	**22/02/18 17:00**
8	**Analysis of data**	5 days?	**23/02/18 8:00**	**1/03/18 17:00**
12	Feature extraction	1 day?	2/03/18 8:00	2/03/18 17:00
13	Validation of inputs	1 day?	2/03/18 9:00	5/03/18 9:00
14	Analysis of Data Report	3 days	5/03/18 9:00	8/03/18 9:00
15	**Definition of a prognosis**	19 days?	**8/03/18 9:00**	**4/04/18 9:00**
24	**Anomalies detection**	13 days?	**5/04/18 8:00**	**23/04/18 17:00**
33	**Health indicator**	15 days	**24/04/18 8:00**	**14/05/18 17:00**
42	Conclusions and Future Work	4 days	11/06/18 8:00	14/06/18 17:00
43	Masther Thesis Report	15 days	15/06/18 8:00	5/07/18 17:00
44	Presentation	1 day?	17/07/18 7:00	17/07/18 17:00

Figure 13. Last phases planification.

4.4.2 ECONOMIC ESTIMATION

In line with the planification, the economic estimation can be calculated by considering the final cost of the project. This means that the budget includes not only the hardware and software costs but also the personal ones.

In terms of hardware, the hydropower plant is not considered as part of the project since it already existed before its beginning. Neither the sensors that retrieve the data and the servers which store the data are considered due to the fact that they were implemented in the hydropower plant before this thesis was requested. Hence, hardware cost can be summed up to a laptop. This information is complemented in Table 3:

Component	*Features*	*Price*
Laptop	Manufacturer: Samsung Modelo: RC530 Processor: Intel Core i7- 6500U; 2,5 GHz Memoria RAM: 8,00 GB Sistema Operativo: Windows 10	€800

Table 3. Hardware components and costs.

Related to the software, Matlab and Excel are the programs which have been used. Because they have been used for an Academic purpose, their licenses have no cost. Otherwise, Matlab would have been €250 with the version R2017b annually and Excel, as a Microsoft program, would have cost €10 with the most updated version, of 2017.

Finally, referring to the personal costs, the project needs a junior analyst and a project manager. Their salaries, which are shown in Table 4, would compose the total amount of this area.

Professional	*Earnings (€/hour)*	*Worked Hours*	*Final Earnings*
Junior Analyst	€30	240 h	€7,200
Project Manager	€55	40 h	€2,200
Total			€9,400

Table 4. Personal costs.

Consequently, the final cost reaches €10,200 without considering V.A.T., as Table 5 presents. The prices rise up to €12,174 taking taxes into account.

Concept	*Amount*
Hardware costs	€800
Software costs	€0
Personal costs	€9,400
V.A.T. (21%)	€1,974
Total	**€12,174**

Table 5. Summary of the total costs.

Chapter 5. TECHNOLOGY DESCRIPTION AND BUSINESS INTELLIGENCE MODELS

This project aims to find the Normal Behavior Patterns for some variables of a hydropower plant station. This is a Machine Learning application and is based on discovering data regularities through computational algorithms in order to develop abilities to classify data into different categories [6].

According to Bishop [6], for every algorithm to develop a Normal Behavior Pattern a training phase is needed in order to let the model learn the data regularities. However, usually a pre-processing of data is required to make training scenario easier. This pre-phase is called feature extraction and has the main goal of retrieving the relationship among data attributes.

Once the model is able to classify data, it tries to estimate unknown values. At the same time, the model is constantly correcting its knowledge to output more accurate predictions.

In this project, Normal Behavior Patterns of some variables have been developed by using Mulilayer Perceptron Neural Netwok, Support Vector Machines and Radial Basis Function Netwoks, based on linear models for classification, Sparse Kernel Machines for regression and Kernel Methods, respectively.

5.1 MULTILAYER PERCEPTRON NEURAL NETWORK (MLPNN)

MLPNN is a good algorithm for this project due to the large scale of data, which sometimes origin problems related to the data basis function adaptation in others methods. [6]

Neurons are the structural constituents of the brain, which is characterized by its impressive calculation capacity. In fact, in many cases it is said that a simile of the brain is a very complex, non-linear and parallel computer. Its qualities are managed through a sophisticated

network of neurons with synaptic interactions. A biological neuron is composed of the apical dentrites, which are synaptic inputs that receive stimuli from other neurons, the body of the neuron cell, basal dentrites, the axon and the synaptic terminals, which are the ones that stimulate others neurons. [7]

The main goal of this neural network model is to classify the data and compare the result with other classifiers which had been already used.

The learning of a neural network consists of two stages: training and validation. The first one is a process by which the weights of the synaptic connections of the neurons are adapted through a continuous stimulation of the environment. The kind of learning is determined by the way in which the changes in the parameters occur. The sequence of events in learning is [7]:

1. Stimulation of the neural network by the environment

2. Modification of the strength of the connections

3. Network response in this new situation

The learning algorithms can be developed by error correction.

There are several neural network models and some of them are going to be explained:

McCulloch-Pitts cells are considered to be precursor neurons of all types of neurons and neural networks that came later. The first type of neuron that was born, as its concept is understood, it is the Perceptron of Rosenblatt in 1958. The perceptron is similar to a cell of McCulloch-Pitts in its structure but not in its functioning. This is because it is a neuron with adjustable synaptic weights and a threshold and the output of the neuron is +1 or -1. Further, the inputs are real and the neuron allows to classify a set of inputs in two separable classes automatically.

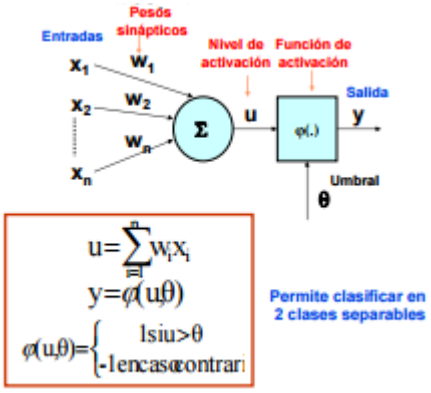


Figure 14. Perceptron model [7].

The perceptron neuron was devised as a classifier of examples. It is a class discriminant. It uses the convergence theorem so that in cases where there is a set of examples described by n characteristics and belonging to one of two possible linearly separable classes, the perceptron neuron can learn the examples and find a hyperplane of separation between them. The parameters of that hyperplane will correspond to the weights of the perceptron neuron capable of separating those classes and would be a solution to the equation [7]:

$$w_1 \cdot x_1 + w_2 \cdot x_2 + \cdots + w_n \cdot x_n - \theta = 0$$

Therefore, the learning of a perceptron neuron consists in finding a vector w so that it separates class C1 from class C2, or, if classes C1 and C2 are linearly separable classes, then there exists a vector of weights w such that:

$w^T x \geq 0$ for each input vector $x \in C1$

$w^T x < 0$ for each input vector $x \in C2$

The perceptron uses the weight adaptation algorithm, which consists of the following steps: Initialization, Activation, Calculation of the response and Adaptation of the weight (only when there is error in the learning of the neurons). This process is repeated in a loop. The key to the learning of the perceptron is in the way in which the weights are modified when the desired output does not correspond to the one given by the neuron. The change in weights may be positive or negative (increasing or decreasing) depending on the error that has been

made according to the desired response. This way of learning is based on the learning concept according to Hebb's rule. This rule states that the change in weights must be proportional to the product of the stimulus of the neuron by its output (in this case the desired) [7].

The perceptron has two main problems. On the one hand, it can only classify 2 linearly separable classes and, on the other hand it does only take into account whether there is an error or not, but not its magnitude.

In this way, it makes sense to talk about another neuron model: almost at the same time that Frank Rosenblatt exposes his perceptron (1958), Bernard Widrow and his student Marcian Hoff introduced a similar neuron model but established a different way of learning from the one of the perceptron and that later served as the basis for the main methods of learning by error correction. The neuron was called ADALINE and it learned with the Delta rule.

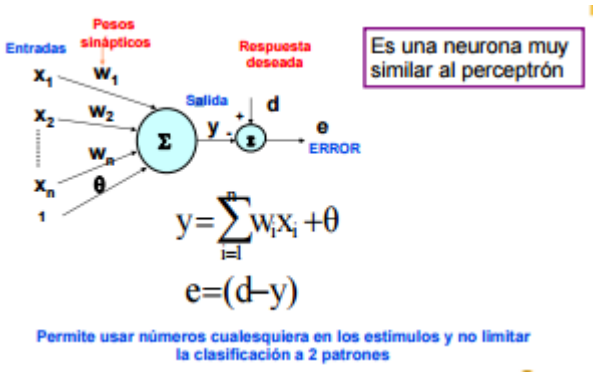


Figure 15. ADALINE neuron model [7].

The objective of the Delta Rule is to obtain an output value y = d when a stimulus x(p) is introduced. If there is a training set that has m examples, the learning would consist of adjusting the weights w_i in such way that the error made with all the examples is as small as possible, as a whole. This leads to evaluate the global error as the mean square error. The Delta rule tries to minimize this global error for the set of examples by means of an iterative process in which the examples are presented one by one and modifying the parameters of

the neuron (weights) by means of the gradient descent method. In this method, the change to be made in each weight would be proportional to the derivation of the error committed when passing an example p by the neuron in relation to the weight considered [7].

Unlike the perceptron, the learning steps of the ADALINE neuron are the following ones [7]:

1. Random initialization of weights
2. Stimulate the model with an example or an input pattern.
3. Calculation of the output y and compare it with the expected one d: d(p) – (p)
4. For all weights, multiplication of the difference from above by the determined input and consider it with the learning rate η
5. Set the new weight adding up the result from the step number 4 to the previous weight
6. In case it has converged, finish the process. Otherwise, if there are still more examples to adjust, keeping on doing the loop. If the examples have been finished and he model has not converged yet, start again to introduce them.

It can be said that the main differences between the perceptron and ADALINE models are that the output of the first one is binary, while in the ADALINE neuron it is real. In the perceptron the weights are modified when the output of the network and the expected output are of different classes, while in ADALINE the weights are always modified by the real difference between expected and actual output. In addition, in the ADALINE neuron there is a measure of how much the neuron is wrong and in the perceptron it is only determined if it has been wrong or not. Finally, in the ADALINE there is a learning rate. In the perceptron it always has a value of 0 or 1. However, despite its differences, there is a way to incorporate the delta rule into the perceptron, which mainly affects the way to calculate the change in weights [7]:

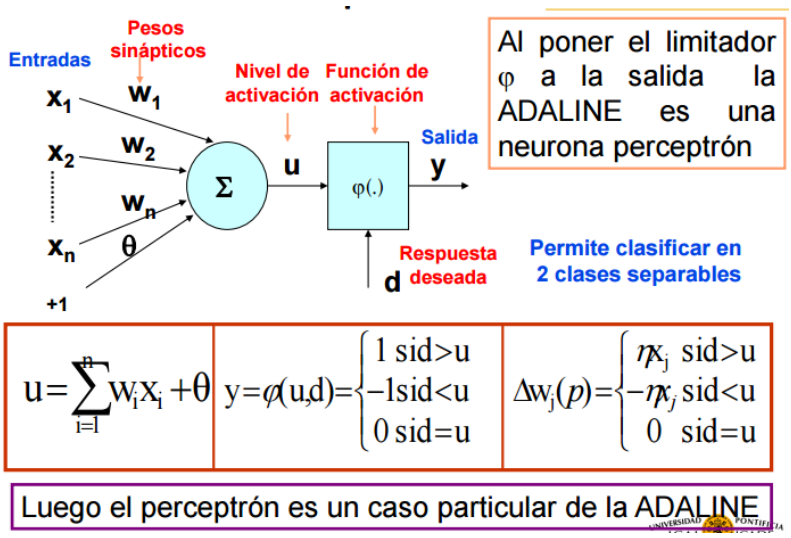


Figure 16.Incorporation of the Delta rule into the perceptron [7].

Although this latter method is an improvement of the two ones described previously, it is still a perceptron, so it can not properly classify a set of examples that belong to classes that are not linearly separable. In order to do so, you would need two lines or hyperplanes. Solving this problem with an ADALINE neuron is not possible either since the learning process reaches a point where it is unable to improve. This is translated into a wrong learned classification by the neuron.

As a solution, the concept of a network of neurons with several layers (Multilayer Perceptron) is introduced. Its effect is to transform the input space into a new dimension so that after that transformation, the examples are distributed in a way that they can be linearly separated.

In the multilayer perceptron some function signals are existed, which are input signals that are propagated across the network from the input to the output. Besides, it also exists the error signals, which are those that originate in the output layer and propagate to the entrance.

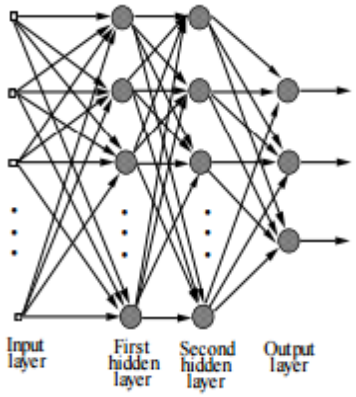


Figure 17. Multilayer perceptron model [7]

This model uses the Back-propagation algorithm, which is characterized by the way the error is calculated: $e_j(n) = d_j(n) - y_j(n)$.

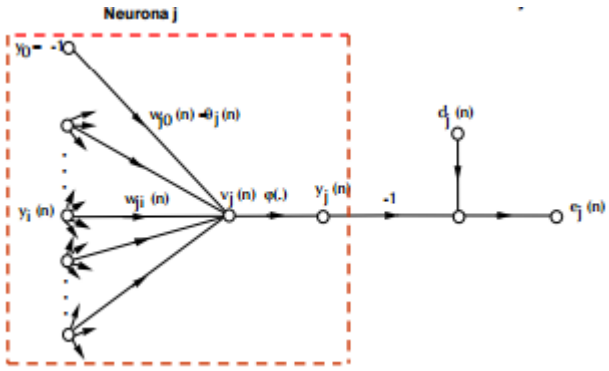


Figure 18. Back-Propagation algorithm model [7]

This algorithm is based on the concept which states that the weight $w_{ji}(n)$ is proportional to the actual gradient:

$$\frac{\partial \varepsilon(n)}{\partial w_{ji}(n)}, where$$

$\varepsilon(n) = \frac{1}{2}\sum_{j \epsilon C} e_j^2(n)$, such that C is the set of neurons of the output layer [7].

Moreover, Figure 18 shows the following parameters:

$$v_j(n) = \sum_{i=0}^{p} w_{ji}(n) y_i(n)\,, where\ p\ is\ the\ number\ of\ stimulus\ to\ the\ neuron\ j$$

$$y_j(n) = \varphi_j\left(v_j(n)\right), which\ is\ the\ output\ of\ the\ neuron$$

The objective is to look for the weights correction which must be done by these formulas:

$$\frac{\partial \varepsilon(n)}{\partial w_{ji}(n)} = -e_j(n)\,\varphi'_j(v_j(n))\,y_i(n)$$

$$\Delta w_{ji}(n) = -\eta \frac{\partial \varepsilon(n)}{\partial w_{ji}(n)}$$

$$\frac{\partial \varepsilon(n)}{\partial w_{ji}(n)} = \frac{\partial \varepsilon(n)}{\partial e_j(n)} * \frac{\partial e_j(n)}{\partial y_j(n)} * \frac{\partial y_j(n)}{\partial v_j(n)} * \frac{\partial v_j(n)}{\partial w_{ji}(n)}$$

$$* \quad \frac{\partial \varepsilon(n)}{\partial e_j(n)} = e_j(n) \qquad \left(\varepsilon(n) = \frac{1}{2}\sum_{j \in C} e_j^2(n)\right)$$

$$* \frac{\partial e_j(n)}{\partial y_j(n)} = -1 \qquad \left(e_j(n) = d_j(n) - y_j(n)\right)$$

$$* \frac{\partial y_j(n)}{\partial v_j(n)} = \varphi'_j(v_j(n)) \qquad \left(y_j(n) = \varphi_j(v_j(n))\right)$$

$$* \frac{\partial v_j(n)}{\partial w_{ji}(n)} = y_i(n) \qquad \left(v_j(n) = \sum_{i=0}^{P} w_{ji}(n)\, y_i(n)\right)$$

Figure 19. Back-Propagation algorithm's ecuations

Last equation from Figure 19 takes into account the gradient decline among the weights spaces with the sign "-". The final formula results in:

$$\Delta w_{ji}(n) = \eta \delta_j(n)\, y_i(n)$$

Considering that:

$\delta_j(n) \rightarrow$ gradiente local definido como:

$$\delta_j(n) = -\frac{\partial \varepsilon(n)}{\partial e_j(n)} * \frac{\partial e_j(n)}{\partial y_j(n)} * \frac{\partial y_j(n)}{\partial v_j(n)} = +e_j(n)\, \varphi_j'(v_j(n))$$

El ajuste de Δw_{ji} depende de $e_j(n)$ error de salida de la neurona j

Two cases exist in this algorithm: that the neuron j is in the output layer or in the hidden layer. For the first option, the error can be calculated by [7]:

$$e_j(n) = d_j(n) - y_j(n)$$

$$e_j(n) \Rightarrow \delta_j(n) \Rightarrow \Delta w_{ji}(n)$$

On the other hand, when the neuron j is part of the hidden layer,

$$\delta_j(n) = -\frac{\partial \varepsilon(n)}{\partial y_j(n)} * \frac{\partial y_j(n)}{\partial v_j(n)} = -\frac{\partial \varepsilon(n)}{\partial y_j(n)}\, \varphi_j'(v_j(n))$$

Moreover, the back-propagation algorithm presents a learning rate η whose value makes weights vary proportionally to its value: the bigger the learning rate is, the more the weights

vary. For this reason, it is important the generalized delta rule, since it let increase the learning rate avoiding the instability risk.

At the same time, for this method, it is recommended to normalize or scale the input and output patterns to the network by means of a linear transformation in the intervals [0, 1] or [-1, 1], depending on whether a sigmoidal activation function is used. or hyperbolic tangent. This avoids numerical problems by disparity of scales of the characteristics of the examples and homogenizes the learning at the same time that it facilitates the procurement of the output that then it is necessary to re-scale if one wants to know its real value.

The steps that this algorithm follows are listed below [7]:

- Step 0: To normalize (highly recommendable).
- Step 1: Random initialization with values close to zero and networking thresholding for the weights.
- Step 2: Take an example n from the training set [x(n), d(n)] and propagate it forward in order to obtain the output y(n).
- Step 3: To evaluate the error e(n) for the n case.
- Step 4: To modify the weights and thresholds according to:

4.1. Calculation the local gradient δ for every neuron from the output layer.

4.2. Modification of the weights in the output layer.

4.3. Calculation of the local gradient δ for the neurons of the hidden layer, from the closest one to the output one until the input one (retropropagation).

Step 5: To repeat steps 2, 3 and 4 for all the examples of the training set, completing an iteration or learning cycle.

Step 6: To evaluate the total error carried out by the network. It is also called the training error.

Step 7: To repeat steps 2, 3, 4, 5 and 6 until reaching a minimum training error. For this task m learning cycles are required.

For the criterion of for the learning algorithm, the following facts must be taken into account [7]:

- The learning algorithm converges when the change in the mean square error per cycle is sufficiently small (0.1% to 1% per cycle or epoch)
- When it has been achieved learning algorithm must be less than a threshold
- When the network parameters (weights) do not move
- When generalization gives good results

The training process can be developed in two different ways: Continuous process or learning based on example or model and Batch learning or bach. Referring to the first one, the weights are updated after the presentation of each example of the training set. The network learns from cycle to cycle (or epoch to epoch) until the stabilization of weights and convergence of learning error to a minimum. It is good to present the training examples in random order from one epoch to another. In batch or batch learning the weights are updated after the presentation of all the examples of a cycle [7].

Regarding the capacity of generalization that neurons may acquire, it is necessary that during the learning the neural network extracts the characteristics of the examples, in order to be able to respond correctly to unknown examples. This is known as network generalization capability.

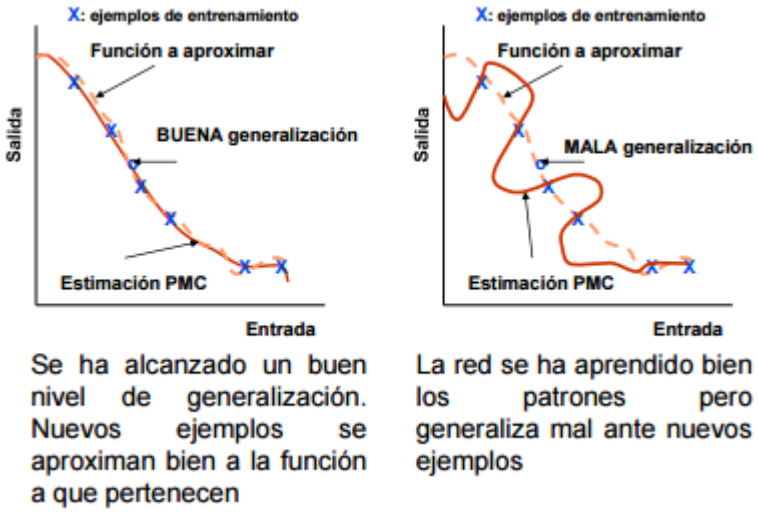


Figure 20. Example of how a neuron can acquire generalization of the patterns [7]

For the neuron to perform the learning process well, there must be two sets of examples: training for the network and modifying its weights and thresholds, and validation, in order to measure the generalization capacity of the network. Both must have different but representative examples of knowledge models to be learned. Proportion of examples around 60% training and 40% validation. As the training error is analyzed throughout the learning cycles, the evolution of the validation error must be analyzed. The validation error is analyzed at the same time as the training error. Every certain number of training cycles, a set of validation examples must be presented to the network and the error committed on said set must be calculated. Sometimes a rigorous training can override the generalization capacity and therefore it is better to perform the calculation of the training error and the validation error in parallel [7].

A very common error in the acquisition of generalization in neurons is the overlearning, which occurs when the network has learned well the examples of the training set (too well) but is not able to generalize or respond adequately to the examples of the set of validation. Overlearning may be due to an excessive number of learning cycles, little diversity in the information of the examples of the training set or an excessive number of hidden neurons.

This method uses the backpropagation algorithm, which is efficient but can encounter different types of problems [7]:

- Local minima: When the surface that defines the error based on the parameters of the network has many valleys and ridges, there is a risk that the error minimization process may end up at a local minimum. This is undesirable especially if you stay away from the global minimum. To solve it, you can increase the number of hidden neurons somewhat, decrease the rate of learning throughout the learning process or add noise to the method used to decrease the gradient.

- Saturation or paralysis: It occurs when the total input to a neuron in the network takes very high values (positive or negative). Since the activation functions have horizontal asymptotes, if the input to a neuron has a high value, it becomes saturated reaching a maximum or minimum activation value. If an output neuron is saturated, then y_j (n) = 0 or 1 and the parameters do not change and the sum of errors is constant. It looks like a local minima problem, but then the error may grow. To solve it, one should avoid that the neuron works in saturation.

In Business Intelligence it is required to support strongly the conclusions inferred in a model. For this reason, the predictions made by the different Neural Networks are going to be supported by two more models: Support Vector Machine and Radial Basis function.

5.2 SUPPORT VECTOR MACHINE (SVM)

SVM is a kernel-based algorithm which has sparse solutions. This means that predictions for new inputs are dependable on the kernel function evaluated at a subset of the training data points [7, 6].

The main feature of SVM is its parameters determination, which corresponds to a convex optimization problem, which is translated into the fact that no local solution is a global optimum too. [6]

SVM is a good algorithm to be used when the attribute to be classified or estimated is non-linear. This is because Kernel functions are able to solve a problem by translating data from a linear space to another one where the hyperplane solution is linear. Then, Kernel functions translate back the solution to the original linear space [8]. This is exemplified in Figure 21.

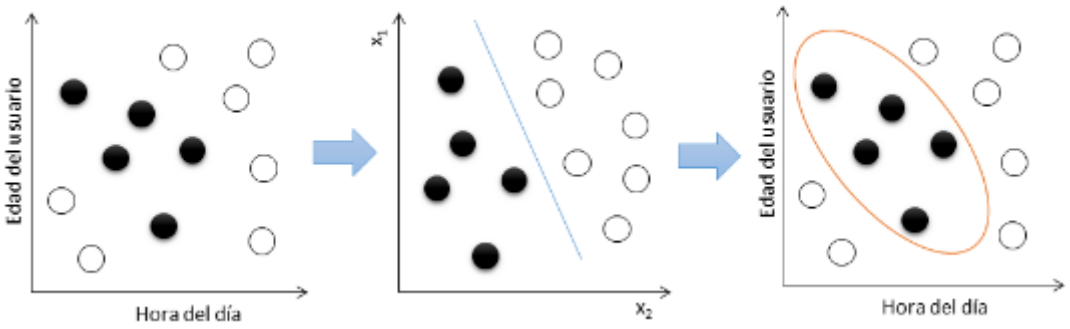


Figure 21. Categorical example of how SVM works [8]

For the depicted example in Figure 19 it may be possible to use another simpler algorithm to classify data. That's why Figure 22 complements this information by a numerical example. In this one, data should be classified by a curve instead of by a line.

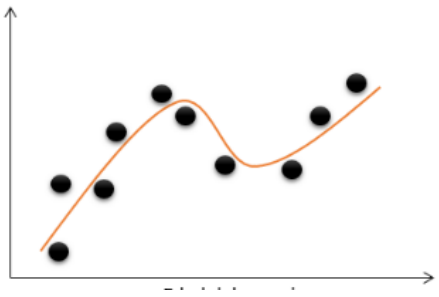


Figure 22. Numerical exampleof how SVM works [8]

Main advantages of SVM are [9]:

- Effectiveness when spaces have several dimensions.
- It is memory efficient because it uses a subset of training points in the decision function.
- It is versatile.

However if the number of features is much greater than the number of samples overfitting may happen. Besides, SVMs don't provide probability estimations, as MLPNN does, used for cross-validation [9].

Main differences in comparison with Neural Networks are listed in Table 6[10].

Neural Networks	**Support Vector Machines**
- Hidden layers transform spaces of any dimension. - Research space has multiples locals minimum. - Training is computationally hard. - Efficient classification. - Number of hidden layers and nodes are designed. - Accurate for typical problems	- Kernels transform data to spaces with a large dimensions number. - Only a global minimum in the research space. - Efficient classification - Cost parameter and Kernel function are designed. - Accurate for typical problems.

Table 6. Neural Network vs SVM [10].

Further information can be found in Bishop book, reference [6]. There, mathematical equations of SVM are explained, together with examples and more details.

5.3 RADIAL BASIS FUNCTION (RBF)

RBF is a pattern recognition technique whose main difference with the previous ones is that during the prediction or testing phase, training data points are not discarded but kept. [6] Therefore, it involves storing the training set and it is fast during training period but slow in the prediction one.

RBF is another Neural Network based on Gaussian functions by the casting into an equivalent dual representation. The main goal of this algorithm is find a smooth function f(x) that fits every target value exactly. Nevertheless, in normal behavior patterns estimations, target values are noisy. For this reason, RBF was expanded from regularization theory. [6]

This method is computationally expensive when predictions are made for new data points since one basis function is associated with every data point. Consequently, regularization phase is required. Regularization is based on stabilizing the solution by means of some auxiliary functions such as smoothness constrains. [11]

Figure 23 presents an example of a typical RBF model. Its main difference with MLPNN is that neurons from the unique hide layer calculate the Euclidean distance between the weights vector and the input. A radial function based on gaussians is applied to this difference. [11]

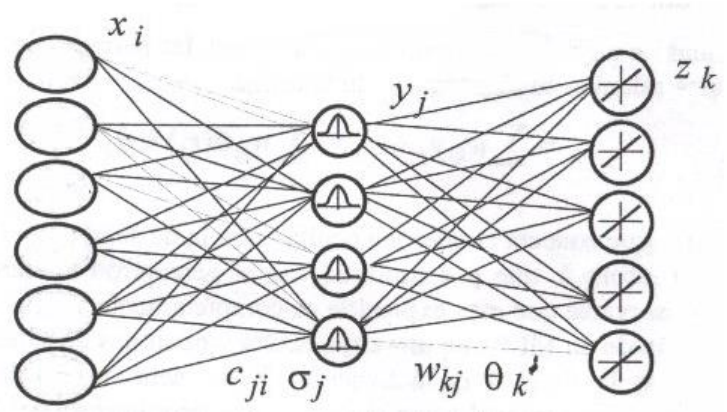


Figure 23. Typical RBF architecture [11]

On the other hand, weights are updated similarly to the MLPNN: by taking previous predictions to the current one into account and using an activation function.

Main advantages are listed below [12]:

- A mapping from input to output exists for all input values.
- The mapping is unique.
- The mapping function is continuous.

On the other hand, RBF also presents some drawbacks [12]:

- Noise or imprecision data adds uncertainty to reconstruct the mapping uniquely.
- Not enough training data to reconstruct the mapping uniquely due to degraded generalization performance.
- Regularization is needed.

All in all, while MLPNN uses the weighted sum of its input values, RBF Neural Network measures the inputs' similarities to real values from the training set. Then, a classification is performed. In this method, every neuron stored an example from the training set to base their calculations on forecasting. These examples are prototypes. Classification is carried out by comparing the prototypes with the input, after passing neurons output through a Gaussian activation function whose center is the prototype vector instead of the mean value of the bell curve. Besides, the Gaussian activation function has a parameter that should be configured in the model development. This is related to the bell-shape width. This parameter is called spread in Matlab and is explained in Chapter 7. . [13]

Finally, Table 7 presents the main differences between MLPNN and RBF:

Multilayer Perceptron Neural Network	***Radial Basis Function Neural Network***
- Sigle or multiple hidden layers - Nonlinear hidden layer and linear or nonlinear output layer - Argument of hidden units: scalar product - Global approximator -	- Single hidden layer - Nonlinear hidden layer and linear output layer - Argument of hidden units: Euclidean normalization - Global approximators

Table 7. Comparison between MPLNN and RBFNN [11]

Further information, especially related to mathematical formulas which RBF method is based on can be found in sources as Bishop's book, with reference [6] in this project's bibliography.

Chapter 6. PROJECT DEVELOPMENT

This Chapter describes the previous steps that are required in order to obtain the patterns of normal behavior as well as the anomalies detection and the health indicators of the different oil levels variables. They are summarized in Table 8:

Task	***Description***
Attributes definition	Acquisition of attributes features: meaning, measurement units, type of values they have.
Data sets definition	Division of the entire data set into two subperiods: one for the training stage (models know expected values) and another one for the testing phase (expected values are unknown).
Data pre-analysis	Statistical parameters of the attributes are calculated in order to remove outliers.
Feature extraction	Study of which attributes explain the most the behavior of the target variables to estimate.

Table 8. Project development tasks.

6.1 DATA SOURCE DESCRIPTION

6.1.1 ATTRIBUTES DEFINITION

Before starting to program, it is required a thorough knowledge of the data source. It has 17 attributes that contains both general operational data about the E4 hydropower station and data related to the Kaplan turbine regulator. Every datum is the mean value of all the numbers measured in an hour. That is why the start time is always at every hour o'clock sharp.

Because the station on study is in Norway, attributes are identified by Norwegian names. However the table below (Table 9) also shows the translation in English and the variables' measurements.

General operational data about the E4 hydropower station

	Name in English	**Name in Norwegian (in the original Excel sheet)**	**Measurement unit**
	Start time	Start time	Date and hour
1	Power	EMBRETFO4_G1	MW
2	Rotational speed of turbine in % of nominal rotational speed	EMBRETFO4_A1_TURTALL	%
3	Guide vane position	EMBRETFO4_LEDEAPPARAT	(opening) in %
4	Water flow	EMBRETFO4_A1_VANNFØRING	m^3/s
5	Bearing cooling water temperature	EMBRETFO4_G1_VANN_LAGRKJØLER	ºC
6	Headwater level	EMBRETFO4_DAM_OVERVANN	moh (water level in the river above the power station)
7	Tailwater level	EMBRETFO4_DAM_VST_UNDERVAN	moh (water level in river below the power station)

Data related to the Kaplan turbine regulator

	Name in English	**Name in Norwegian (in the original Excel sheet)**	**Measurement unit**
	Start Time	Start Time	Date and hour
8	Turbine runner_oil pressure	G1_Løpehjul_oljetrykk	bar
9	Guide vanes_oil pressure	G1_Ledeapparat_oljetrykk	bar
10	Leakage oil tank_oil level	G1_Lekkoljetank_oljenivå	%

11	Oil tank_oil level	G1_Oljekasse_oljenivå	%
12	Oil tank_temperature	G1_Oljekasse_temperatur	ºC
13	Accumulator 1 guide vanes operating mechanism_oil level	G1_Ledeapparat_akku_1_oljenivå	%
14	Accumulator 2 guide vanes operating mechanism_oil level	G1_Ledeapparat_akku_2_oljenivå	%
15	Accumulator 3 guide vanes operating mechanism_oil level	G1_Ledeapparat_akku_3_oljenivå	%
16	Accumulator 1 turbine runner_oil level	G1 _Løpehjul_akku_1_oljenivå	%
17	Accumulator 2 turbine runner_oil level	G1_Løpehjul_akku_2_oljenivå	%

Table 9. Measured signals in the E4 hydropower station. [15]

It is important to take the numbers which index the signals into account since at some points the attributes are referred by them in order to make graphs clearer.

6.1.2 TRAINING AND TESTING DATA SETS

During the project development three data sets have been received in total. The first one contains data from the 1st of January of 2015 until the 31st of May of 2017. Because this is a long period, it has been divided into two subsets: one forms the training set and the other one is the first testing set. Then, the training set would be composed by data from the 1st of January of 2015 until the 13th of February at 9:00 h at first whereas the testing set would be from the 13th of February of 2017 at 10:00 h until 31st of May of 2018.

However, if data of the hold set is plot (Figure 24), it can be inferred that some attributes adopt unusual values before the sample number 10610, which matches with the date of 9th of April of 2016.

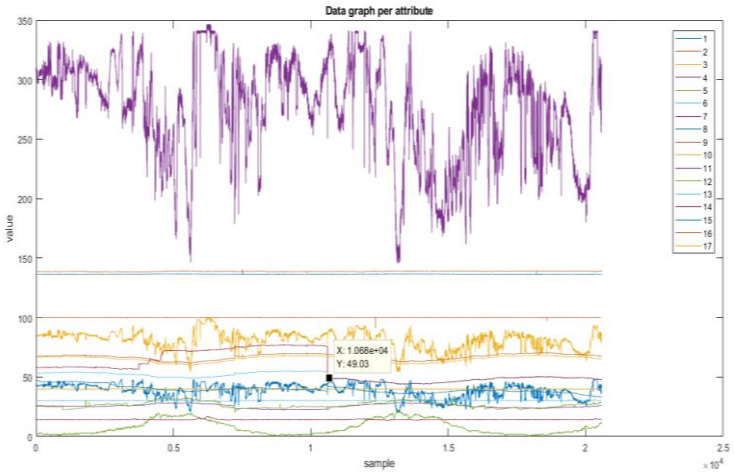


Figure 24. First data set plot by attributes.

This conclusion is better observed when only the signals that are going to be predicted are represented (Figure 25):

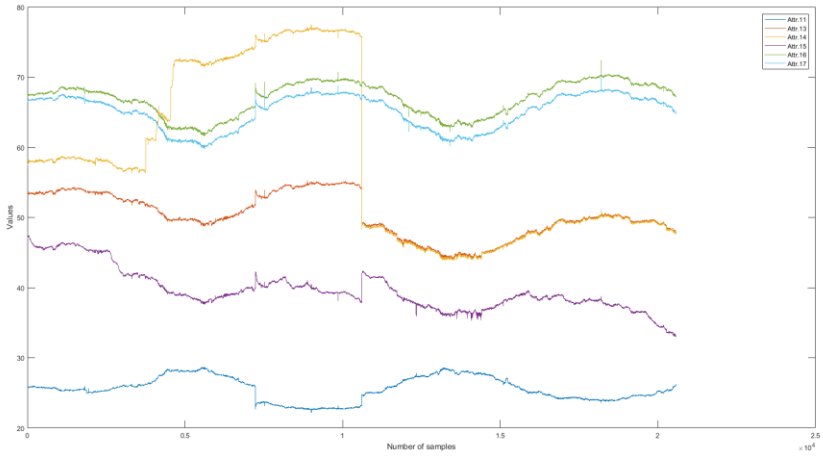


Figure 25. Different oil level signals from the first data set.

On the other hand, there are two more testing data sets apart from the mentioned one. The first set includes all values from the 1st of June of 2017 until the 31st of October of 2017

whereas the second one contains the data which corresponds to the period that starts on the 1st of November of 2017 and ends on the 21st of May of 2018. Nevertheless this latter set is different from the other ones since instead of containing the hourly mean value of every attribute, it contains a single value which has been retrieved during the determined hour and randomly chosen.

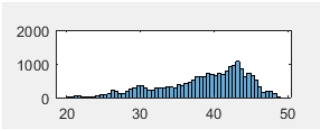
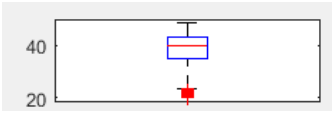
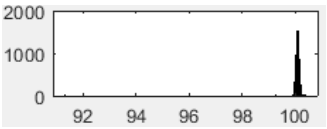
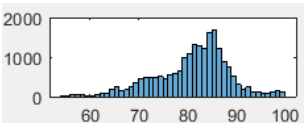
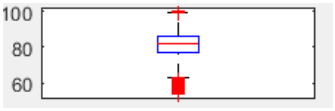
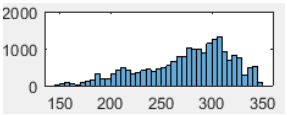
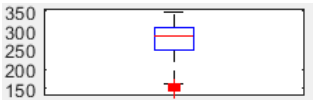
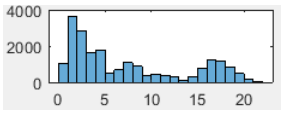
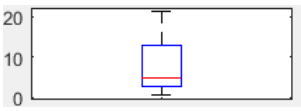
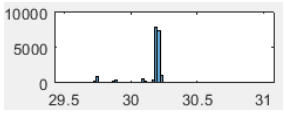
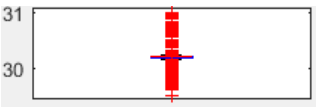
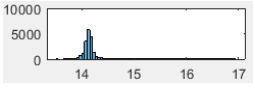
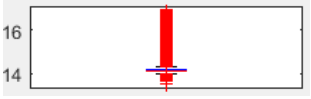
The following table sums up the period which are included in every group of data (Table 10):

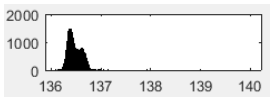
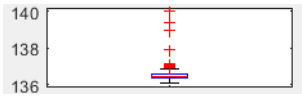
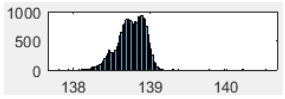
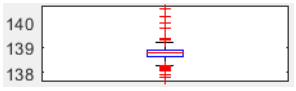
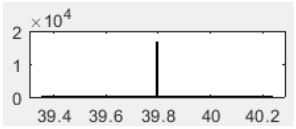
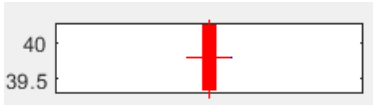
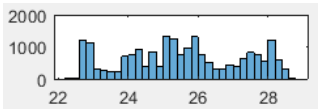
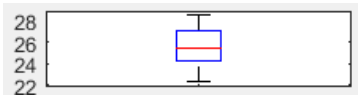
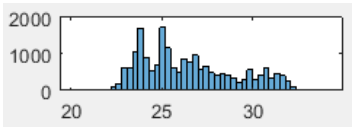
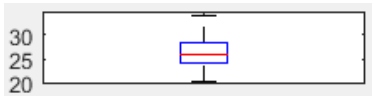
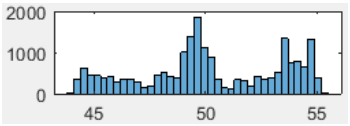
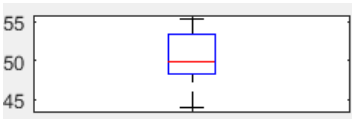
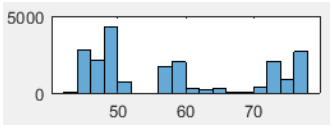
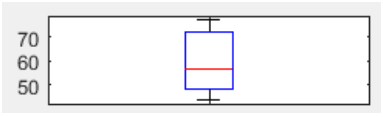
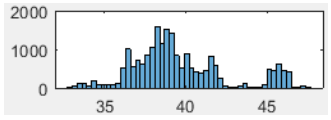
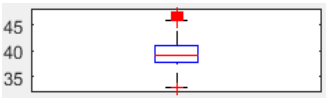
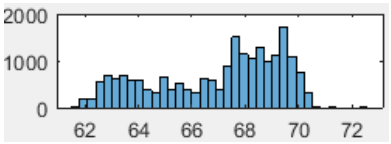
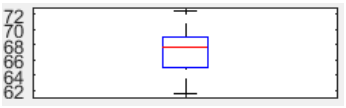
Data set	**ID**	**Period**	**Number of valid samples**	**Proportion of total data**	**Total training and testing proportions**
Training set	tr	9/April/2016 – 13/Feb/2017	7391	48.6%	48.6%
First testing set	ts1	13/Feb/2017 – 31/May/2017	2575	16.9%	51.4%
Second testing set	ts2	1/June/2017 – 31/Oct/2017	2963	19.5%	
Third testing set	ts3	31/Oct/2017 – 21/May/2018	2277	15%	

Table 10. Details of every data set.

6.2 DATA PRE-ANALYSYS

The data pre-analysis consists in filtering them by removing out the outliers and others values which do not provide useful information to the models to predict the outputs. This step has been carried out by observing the boxplots and histograms of every attribute. They are plot in the following table (Table 11).

Attr.	*Boxplot*	*Histogram*
1.Power		
2.Rot.Sp.Tur.		
3.GVPos		
4.WaterFlow		
5.B.C.Wat.Temp		
6.Head		
7.Tail		

8.Tur.OilPres		
9.GVOilPres		
10.LeakOilTank		
11.OilTank		
12.OilTankTemp		
13.OilGV1		
14.OilGV2		
15.OilGV3		
16.OilTur1		

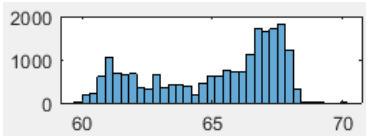
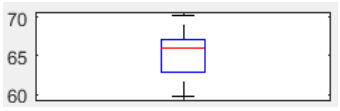
17.OilTur2		

Table 11. Attributes' boxplots and histograms

Table 11 shows that some attributes have outliers that must be removed out from the sets. In this section the filtering of the first attribute, which is the power level, is going to be explained in detail. However more information about this step would be added in Annex A. This one is especially referred to those variables which are not commented below as the power produced.

Taking graphs from Table 11 that are referred to the power into account, the first attribute should be filtered in order to obtained just the main values which provide the most useful information. This means that the Business Intelligence models should only receive those values which are considered as normal measurements in the hydropower plant.

Both boxplot and histogram of the power are presented below in a bigger size in order to help to understand the filtering (Figure 26):

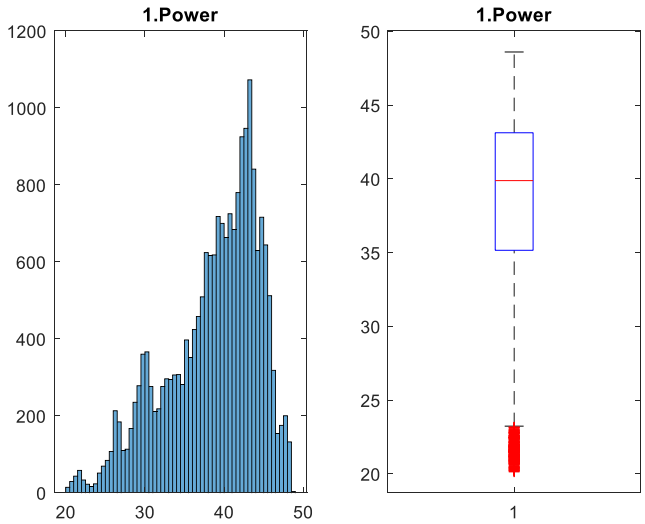


Figure 26. Histogram and boxplot of the attribute 1.Power

From Figure 26 it is concluded that there are not significant outliers. Indeed, the outliers which appear in read in the boxplot are not very far away from the first quartile, which is 23 MW. Therefore, all rows whose power value is below 20 MW must be taken out of the data set.

Despite the data filtering due to the leftover attributes is commented in Annex A, the final result is summarized in Table 12:

Due to attribute	*Condition*	*Due to attribute*	*Condition*
1.Power	<20 MW	10.LeakOilTank	<39 %
2.Rot.Sp.Tur.	<97 %	11.OilTank	No condition
3.GVPos	<54 %	12.OilTankTemp	<16 %
4.WaterFlow	<140 m^3/s	13.OilGV1	No condition
5.B.C.Wat.Temp	No condition	14.OilGV2	No condition
6.Head	<29 MOH	15.OilGV3	>46%
7.Tail	<13 MOH	16.OilTur1	No condition
8.Tur.Oil.Pres	<136 bar	17.OilTur17	No condition
	>137 bar		
9. GVOilPres	<134 bar		

Table 12. Filtering conditions according to every attribute.

6.3 FEATURE EXTRACTION

As it has been mentioned out, the algorithms that have been used in this project to predict different variables, consists in classifying an output signal taking into account the values of some input signals. Therefore it is required to know which attributes are the ones which most explain the desired output. In order to guess so, it has been resorted to the `relieff` function from Matlab. This function returns a histogram which tells the predictor importance weight of every input attribute for explaining the output.

In this sense, the result obtained for the first guide vane accumulator is depicted in the Figure below (Figure 27):

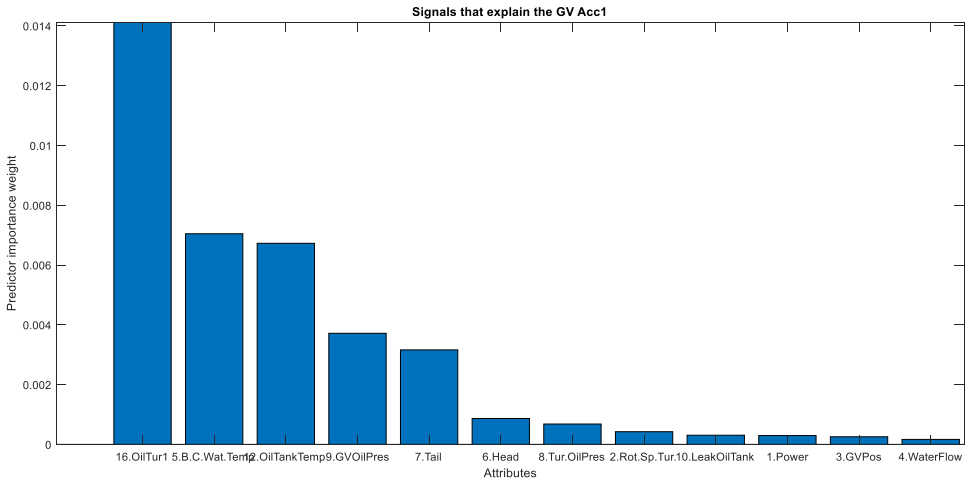


Figure 27. Predict importance for the output AC1-GV behavior.

According to Figure 25 possible inputs to estimate the accumulator 1 guide vanes operating mechanism oil level may be:

1. Attr. 5: Bearing cooling water temperature
2. Attr. 6: Headwater level
3. Attr. 7: Tailwater level
4. Attr. 8: Turbine runner oil pressure
5. Attr. 9: Guide vanes oil pressure
6. Attr. 12: Oil tank temperature
7. Attr. 16: Accumulator 1 turbine runner oil level

Next step consists in selecting the concrete inputs that should be used in the models to predict this determined output. This decision must consider a limitation: input signals from every testing set must adopt any value as long as the same attribute adopts it in the training set. If this requirement is not being accomplished, the model would be considered not valid.

This requirement would be explained below with the bearing cooling water temperature attribute:

It seems the training set of this attribute includes every value that belongs to both testing 1, testing 2 and testing 3 sets (Figure 28). However when the probability density function (PDF) is depicted (Figure 29), it can be inferred that most values of the second period of testing are included in the tail of training set PDF. As a result, it may be better to neglect this attribute. Fortunately, though according to Figure 27 this attribute may be the second one which may explain most of the neural network model's output, the first one explains more than the double of this one. This means that ignoring this attribute as an input for the models may not be significant and it has been demonstrated during the models' developments. Additionally, discarding this input would contribute to have simpler models.

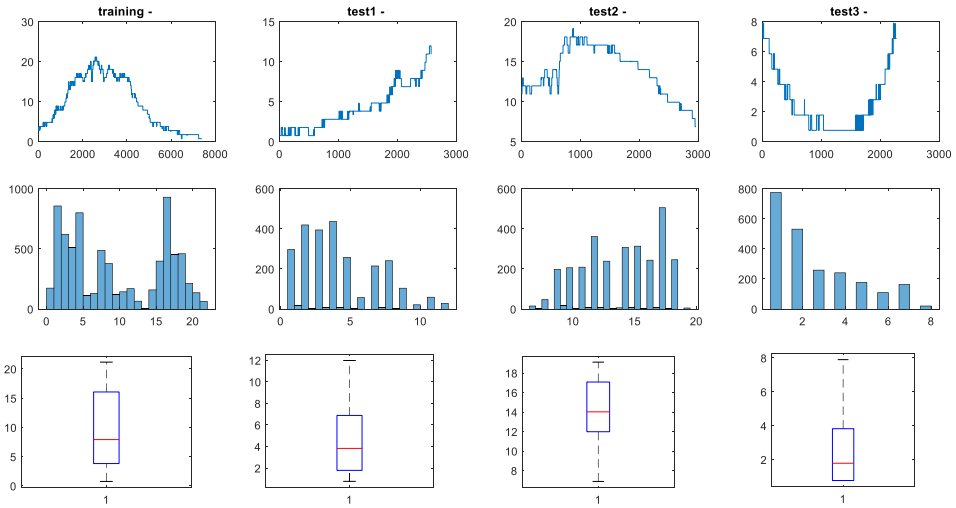


Figure 28. Bearing cooling water temperature graphs by training and testing periods.

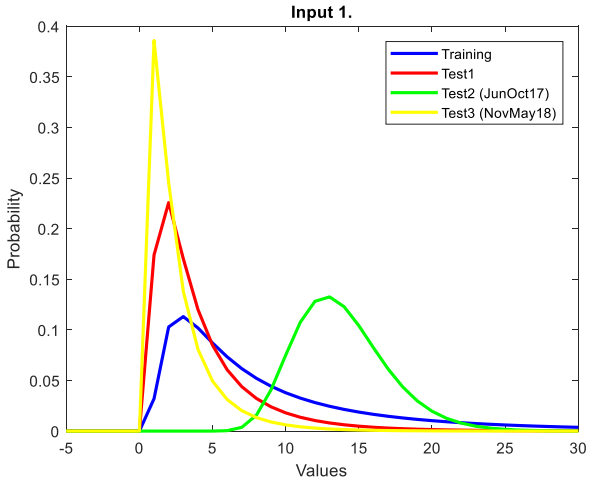


Figure 29. Probability Density Function by training and testing periods of 5.Bearing cooling water temperature.

Similarly to the bearing cooling water temperature attribute, this procedure was repeated for every possible input which appeared to be significant for the prediction of AC1-GV. In this case, every suggested input mentioned in the list above accomplished the requirement, so the final decision was the same one.

Parallelly, this step has been repeated for all the outputs which are desired to be estimated: AC2-GV, AC3-GV, AC1-TUR, AC2-TUR and AC-TANK; from the execution of the `relieff` function. The final result for them is shown in Table 13:

Output	*Suggested Inputs (`relieff` result)*	*Final inputs*
AC1-GV AC2-GV AC3-GV	5.B.C.Wat.Temp 6.Head 7.Tail 8.Tur.OilPres 9.GVOilPres 12.OilTankTemp 16.OilTur1	6.Head 7.Tail 8.Tur.OilPres 9.GVOilPres 12.OilTankTemp 16.OilTur1
AC1-TUR AC2-TUR	2.Rot.Sp.Tur. 5.B.C.Wat.Temp 7.Tail 8.Tur.OilPres 9.GVOilPres 12.OilTankTemp 13.OilGV1	7.Tail 8.Tur.OilPres 9.GVOilPres 12.OilTankTemp 13.OilGV1
AC-TANK (with real data)	1.Power 6.Head 7.Tail 8.Tur.OilPres	1.Power 6.Head 7.Tail 8.Tur.OilPres

	9.GVOilPres	9.GVOilPres
	12.OilTankTemp	12.OilTankTemp
	13.OilGV1	13.OilGV1
	16.OilTur1	16.OilTur1

Table 13. Final inputs for the developed models by predicted outputs.

More information about this section, especially referred to accumulators 2 and 3 guide vanes operating mechanism oil level, accumulators 1 and 2 turbine runner oil level and the hub tank oil, can be found in Annex A.

Chapter 7. RESULTS

In this Chapter the main results are commented. It is divided into three parts. The first one is oriented to the normal behavior models or prognosis of the oil level guide vanes operating mechanism accumulators, the turbine runner accumulators and the hub tank oil. Secondly, detection of anomalies scenario is explained and finally, the third subsection is referred to the health indicators.

Only the first guide vanes operating mechanism accumulator and some additional interesting information related to any other variable are presented in this Chapter since results which correspond with others signals are achieved in a similar way. Nevertheless those that are not included, are commented in detail in Annex B.

Finally, every result has been developed by using the three Business Intelligence models which were explained in Chapter 5. .

7.1 NORMAL BEHAVIOUR MODELS

The main method for developing the normal behavior models is the MPLNN. However SVM and RBF are also used in order to provide more accuracy and consistency to the outcome. All of them use the defined training set to learn the output signals' behaviors. Then, once the models have acquired the enough knowledge to estimate the variables, the models' accuracies are studied by measuring the errors and comparing the predictions with real data. The main difference between the two steps for the models to learn the attributes behaviours is that they already know the expected value before the estimation is calculated during the training stage while during the testing ones real data is unknown.

AC1-GV is the signal which is going to be exemplified in order to explain all the methods. Rest attributes are explained in detail in Annex B.

7.1.1 AC1-GV

7.1.1.1 MLPNN

In terms of MLPNN, regardless the number of neurons, the error committed by the model is minimum when parameters have the values described below:

- As it has been described, the neural network learns how to classify the data by three stages: training, validation and testing. The first one consists in studying how the input signals affect the value of the output by knowing the result. During validation stage, the neural network applies the knowledge learned in the previous stage and analyses the error and set the parameters improving the result. Its goal is to determine a stopping point for the back-propagation algorithm. Finally, the testing phase has the objective of assessing the performance of a fully-trained classifier and measuring the error rate after choosing the final model. In this way, the neural network model for estimating level of oil in the guide vanes accumulator 1 has randomly divided the training set data into three groups, one per phase: 75% of the dataset is used in the training stage, 10% is for the validation and the 15% left for the testing:

```
net(i).net.divideParam.trainRatio = 75/100;
net(i).net.divideParam.valRatio = 10/100;
net(i).net.divideParam.testRatio = 15/100;
```

- The maximum iterations or cycles which the same data may be processed by the neural network is 10000:

```
net(i).net.trainParam.epochs = 10000;
```

- The learning rate, called η in the thoretical part of this work, is equal to 0.1%:

```
net(i).net.trainParam.lr = 0.001;
```

- The training status is shown every 50 epochs:

```
net(i).net.trainParam.show = 50;
```

- The minimum performance that the neural network may have is 1e-5:

```
net(i).net.trainParam.goal = 1e-5;
```

- The maximum fail index can be 50:

```
net(i).net.trainParam.max_fail =50;
```

These parameters are summarized in Table 14:

Parameter Name	***Value***
Training Ratio	75%
Validation Ratio	10%
Testing Ratio	15%
Maximum number of iterations	10000
Learning rate	0.001
Observing the performance every	50
Minimum performance	1e-5
Maximum fail	50

Table 14. AC1-GV Multilayer Perceptron Neural Network parameters.

The previous parameters were based on others models used for this same hydropower station which had the goal of predicting the behavior of some other signals that are measured in order to anticipate the anomalies [3].

Then, a study about number of neurons the network should have to get the most accurate result has been done. This would help to get the minimal error. In this way, a loop has been programmed to observe the results obtained with 8, 12, 16, 20 and 14 neurons.

In order to measure the error, it has been declared three variables in Matlab: one for allocating the value of the mean error, another one for the standard error and the last one for the Mean Absolute Error (MAE). The difference among them is that the first one sums up every error and divides the total sum by the total number of iterations. In other words, it calculates the mean value of the difference of every expected or real datum and the obtained one:

$$mean\ error = \frac{|expected\ value - output\ obtained|}{number\ of\ observations}$$

The standard error (SE) is the standard deviation of its sampling distribution:

$$SE = \frac{Standard\ deviation\ of\ the\ error}{\sqrt{number\ of\ observations}}$$

The picture below may help to understand its meaning. It is an example for a variable with an unbiased normally distributed error and it depicts the proportion of samples that would be between 0, 1, 2 and 3 standard deviations above and below the actual value.

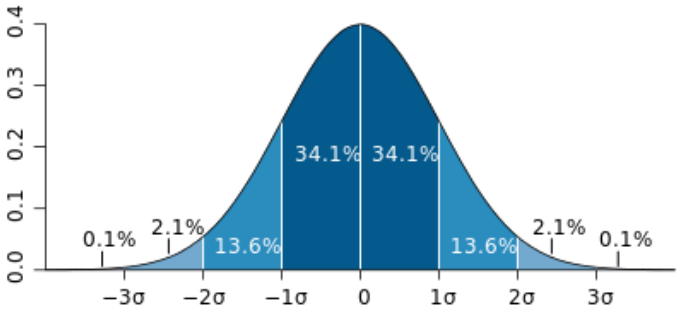


Figure 30. Some help to understand the Standard Error.

The MAE is a measure of difference between two continuous variables.

$$MAE = \frac{\sum_1^N |expected\ value - output\ obtained|}{N}$$

, with N being the number of observations.

Figure 31 shows the transition of the three explained error types for both training and testing phases. According to it, it seems the best result is obtained when the model has 16 neurons. This is the main reason why it has been chosen this number of neurons.

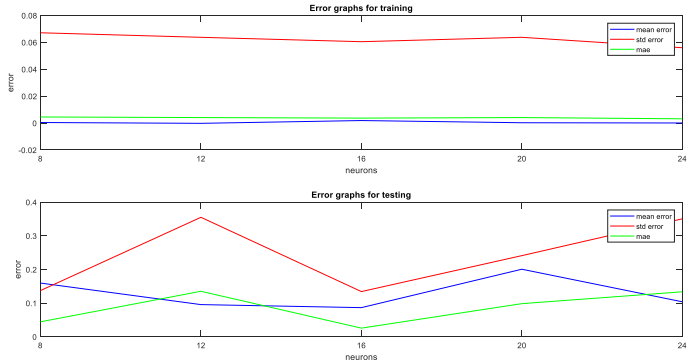


Figure 31. Error measures for the different neural network models

It can be inferred from the graphs depicted in the Figure 32 below that the prediction which has been calculated by the neural network is very accurate and close to the real data despite the bearing cooling water temperature attribute has been neglected as an input. The histograms show the error during the training phase doesn't even reach 0.5%. Besides, during the first period of testing, the error is approximately 1%; and 2% for the second testing set. Error is greater for the last testing set mainly because the retrieved data are not a mean of every taken measurement during evert hour. All in all, the result seems to be valid.

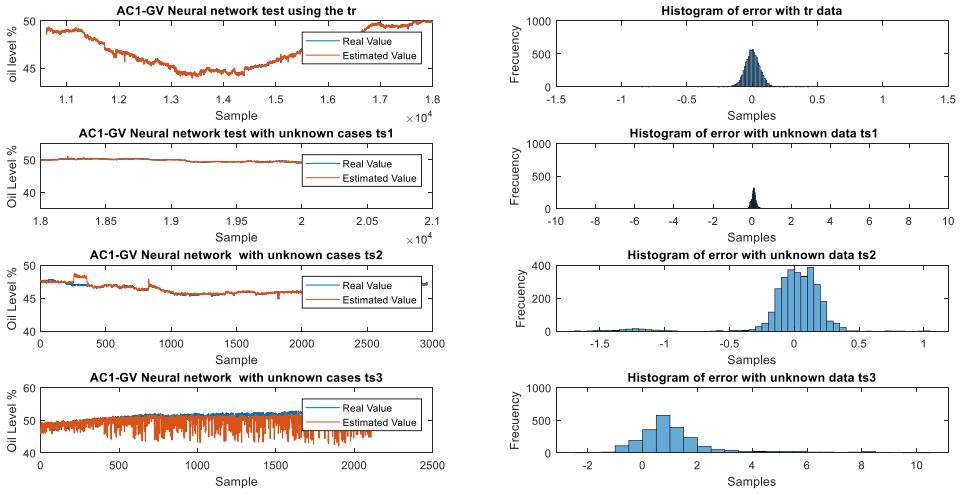


Figure 32. Neural network model result for AC1-GV with 16 neurons for training and testing scenarios.

Additionally, Figure 33 complements the information referred to this MLPNN model, which provides more graphs about the check how accurate the chosen neural network model for predicting the oil level in the guide vanes accumulator number one from June to October of 2017 (second testing period). Despite the prediction seems to be worse than the one made for the testing 1 period, it is not. The biggest error doesn't reach the 2% either, as commented above, and the expected values are almost completely overlapped with the predicted ones. Furthermore, the error graph is close to zero in most of the samples.

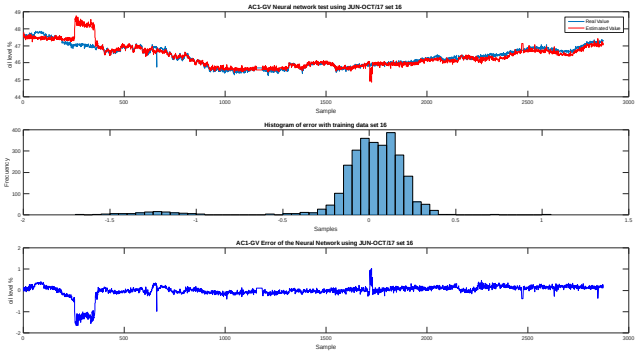


Figure 33. Prediction of the AC1-GV NN with 16 neurons for the second period of testing

As a curiosity, the commented model can be compared with another one which has a different number of neurons, such as 8 and 20. Observing Figure 34 and Figure 35, which only show the results for the training and the first testing sets, it is verified that 16 neurons give a better approximation than 8 and 20. There is no such a big difference between 8 and 16 neurons, since the histograms show almost the same percentages of error: less than 0.5% for the training stage and approximately 1% for the testing one. Nevertheless during the latter depicted period, there are some points which are not as accurately predicted as the chosen model does at. These points are some samples between the number 20,000 and the number 20,500.

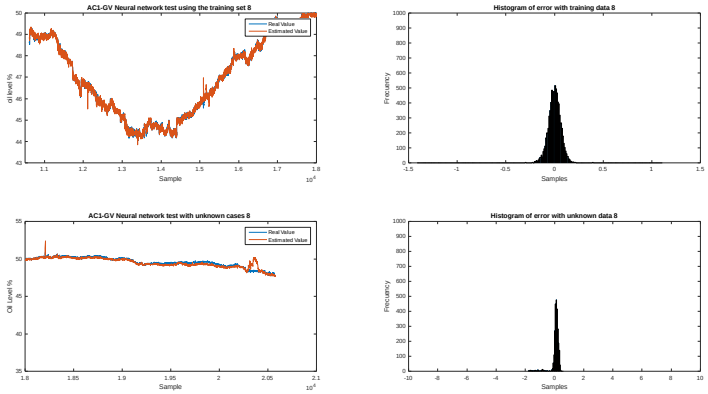


Figure 34. Neural network model result for AC1-GV with 8 neurons for training and testing scenarios

In contrast, the model with 20 neurons shows a bigger difference from the one with 16 neurons (Figure 33, which only shows results from training and the first testing sets). This may be because the network may have over-learned and makes a poorer prognosis. In this way, even the error of the training phase is still smaller than the 0.5%, during the testing, from the histogram some samples with more than the 4% of error are observed.

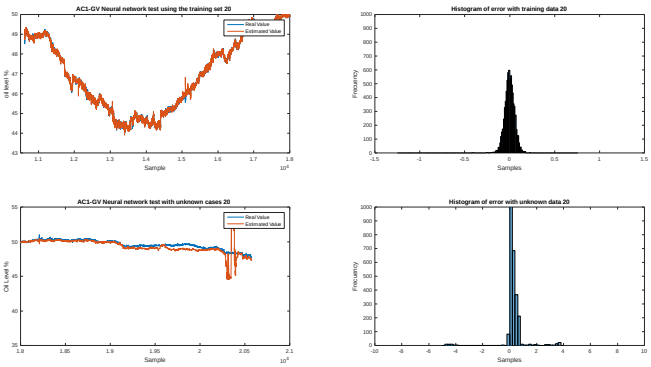


Figure 35. Neural network model result for AC1-GV with 20 neurons for training and testing scenarios.

Furthermore, another important note in this model is that because the model has still good results, ignoring the attribute of bearing cooling water temperature as an input, not only has contributed to get a valid model but also a simpler one. In order to confirm that it didn't

explain significantly the output of the level of oil in the guide vanes accumulator 1, Figure 36 is included below. It shows the best result obtained, including the same inputs as in the valids model, adding the bearing cooling water temperature for the training and the first testing period:

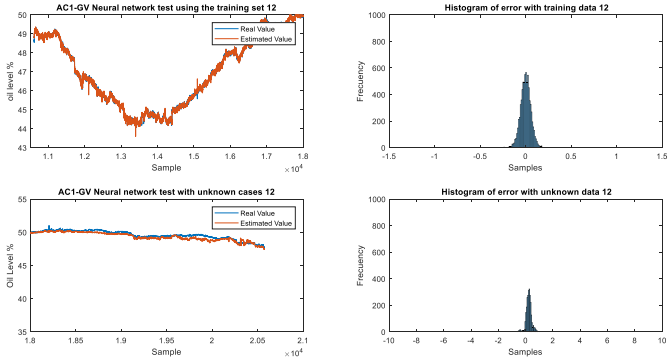


Figure 36. Not valid neural network model result for AC1-GV with 12 neurons for training and testing scenarios since it takes into account the bearing cooling water temperature as an input

As it has been pointed out at the beginning of this document, every model in Business Intelligence may be supported by others. Therefore, the following sections are focused on how the oil level of the guide vanes accumulator 1 is predicted by Support Vector Machines and Radial Basis Functions.

7.1.1.2 SVM

Once again, the first step to develop this model is to set up are the determined parameters. They are mentioned below [16]:

- The model is retrained using the standardized data.
- The program is solved by the solver called Sequential Minimal Operation (SMO)[i].
- The kernel function used in the model is polynomial type which means that it follows the following mathematical formula:

$$G(x_j, x_k) = (1 + x_j{}' x_k)^q$$

- The KernelScale parameter is set to auto. The software divides all elements of the predictor matrix X by the value of this parameter. With the value given in this model, the software selects an appropriate scale factor using a heuristic procedure. This heuristic procedure uses subsampling, so estimates can vary from one to another.

Below, there is Table 15 which summarizes the already explained parameters:

Parameter Name	*Value*
Re-training data	Standardized
Solver	Sequential Minimal Operation (SMO)
KernelScale	Auto

Table 15. AC1-GV SVM parameters.

In Figure 37 there are twelve graphs: the first row of three of them are focused on the training data, the second one is referred to the first period of testing, the third one is related to the second testing period, from June to October of 2017 and the last one has to be with the last testing period, which is from November of 2017 to May of 2018.

For every phase, the comparison between the real data and the prediction, the histogram of the error and the determined graph of the error has been depicted. It is inferred that the error is bigger using SVMs than MLPNN, especially when predicting the data from June to October of 2017 (testing 2 period) and from November of 2017 to May of 2018 (testing 3 period). Nonetheless, as already mentioned, the reason why the error increases sharply in this last period may be because the set contains random measurements of different instants instead of hourly mean values.

The error for the training dataset is almost zero. Its density probability function follows a normal distribution with mean equal to 0. The maximum error in the prediction of the training dataset is almost 0.3%. For the first unknown dataset (testing 1 period) the approximation made by the SVM is still accurate since the error is almost zero, except for some samples, which makes the histogram not to be centered in zero. Finally, though at some

samples the error for the second unknown dataset is bigger than the 15%, it is almost zero in most points, as it can be observed in the last graph of Figure 37.

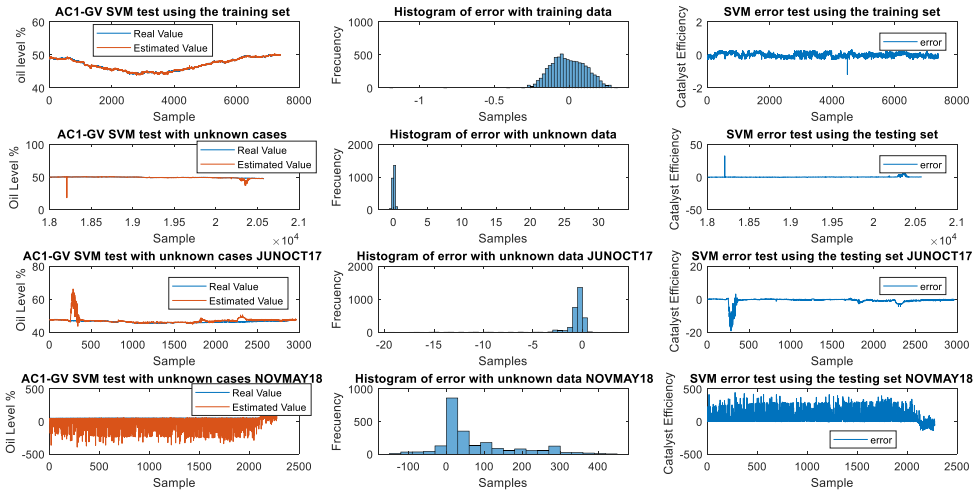


Figure 37. Results for the prediction made by SVM for the AC1-GV for the tr, ts1 and ts2 sets.

All in all, the SVM model has enforce the obtained result from the MLPNN: accumulator 1 of the guide vanes' oil level can be explained by some signals which are being measured in the hydropower plant.

7.1.1.3 RBF

Finally, in order to re-strengthen the results obtained, a Radial Basis Function model has been carried out. It just consists in demonstrating by a new different way that the output that is being studied can be predicted by the determined inputs.

Configuring RBF models is simple since only a parameter has to be set up, which is the spread. The spread is the value which is used to set the extension of the Gaussian probability distribution, as it is shown in Figure 38. The larger it is, the less sensitive the neural network will be.

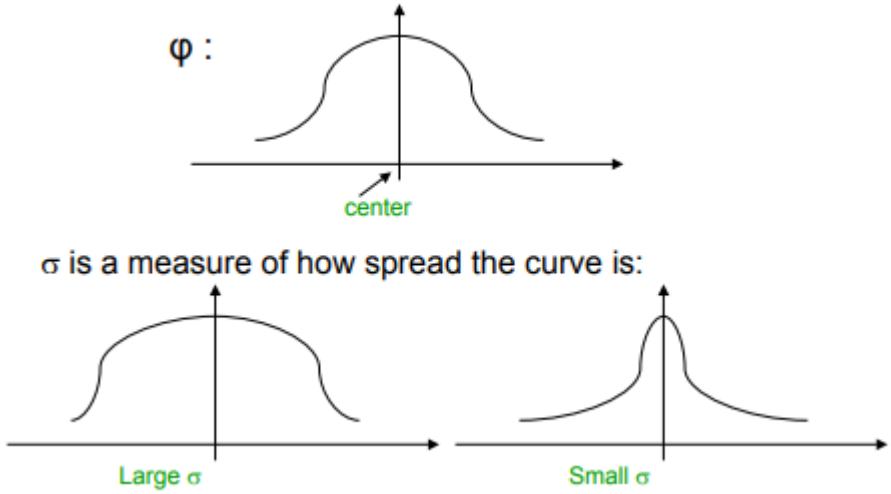


Figure 38. Definition of the spread parameter [17]

In order to choose the most appropriate spread value, it has been developed a loop in which the parameter adopts a value from 15 to 65. For estimating the oil level of the accumulator 1 guide vanes, it seems the best result is obtained when the spread is equal to 25 (Figure 39):

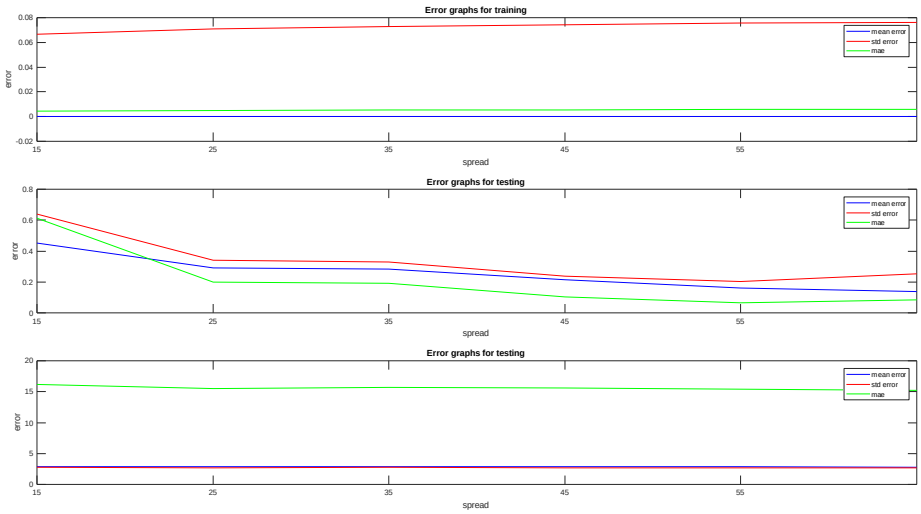


Figure 39. Error measurements for the different RBF Neural Networks developed

The function used in Matlab is `newrbe`. The main drawback of this function is that it produces as many hidden neurons as there are input vectors. It may be for this reason why predictions for the testing sets are not very accurate, though the training phase has good results. This issue can also be inferred from Figure 40. It shows the comparison between the real and the estimated data, the graph of their difference and its histogram. These graphs are referred to the model which has a spread of 25, the one with best results.

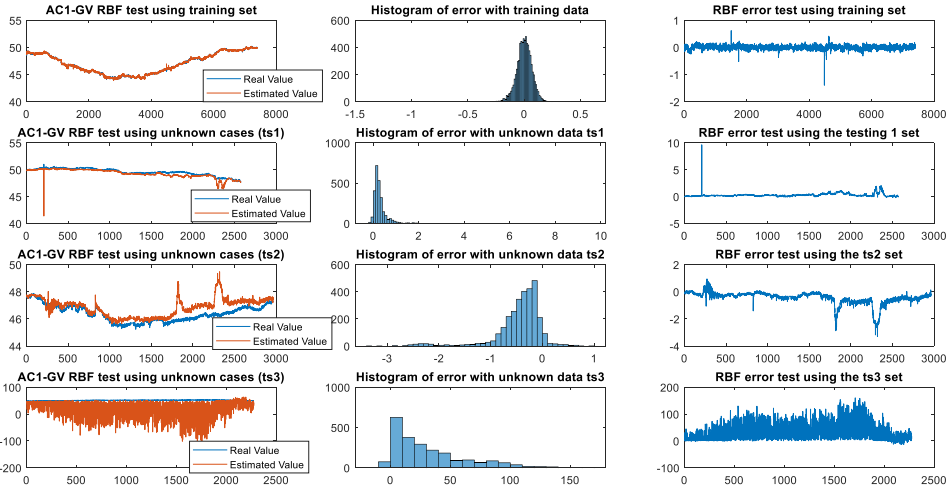


Figure 40. Result obtained in the RBF neural network with spread = 25

For spread equal to 25 the maximum training error is around 0.5%, while for the first period of testing it increases until the 10%, but just in a sample. Regardless this outlier, the maximum error for this period can be considered 2%. By contrast, for the second period of testing, the error is seriously soared until the 15%. Indeed, it still increases reaching big amounts during the last testing period. Furthermore, the comparison between real and predicted data are another representation of the fact that error is increasing by the further you go from the training period in terms of time.

Despite the error is bigger for this model, predictions are valid enough in order to support the Multilayer Perceptron Neural Network estimation. The reason why this happens is because the RBF is not the most appropriate model to carry this approximation out.

Just to justify that the best approach for RBF neural networks was the previous described one, below the result obtained for a spread equal to 55 from the training and the first two testing sets can be found (Figure 41). The first period of testing and training one have similar predictions than the previous RBF model. Nevertheless, predictions get much worse for the second period of training. That's the reason why it has not been chosen as the best RBF Neural Network model to estimate AC1-GV.

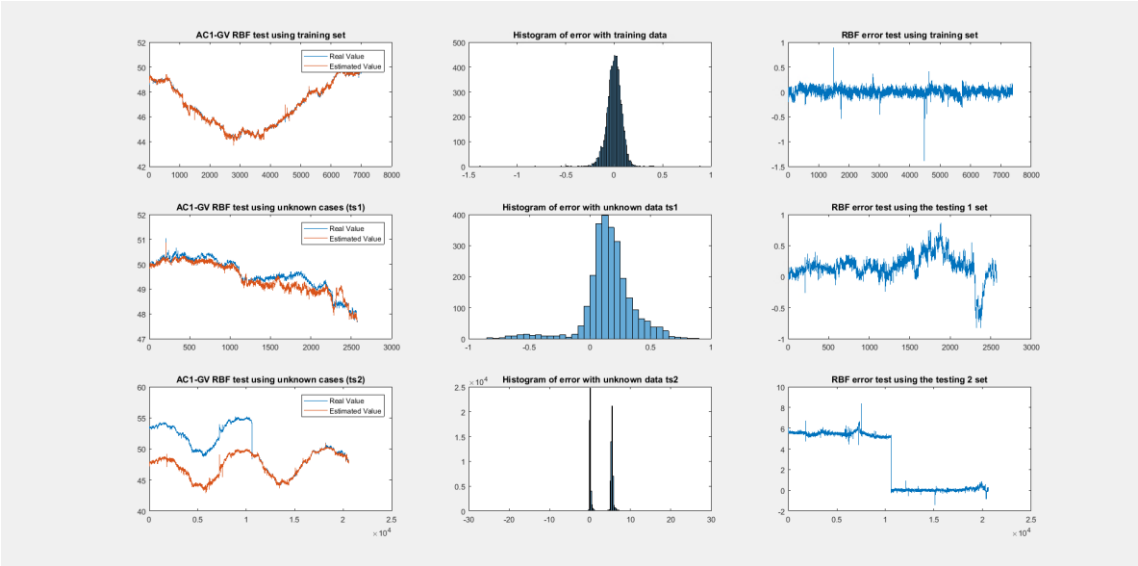


Figure 41. Result obtained in the RBF neural network with spread = 55 for tr, ts1 and ts2 data sets.

7.1.1.4 Comparison among MLPNN, SVM, RBF

In this section the three previous models are compared. However, due to the peculiarity of last testing data set, which encompasses values from November of 2017 to May of 2018, it is not considered. In this way graphs will be clearer, since for such data set error is incredibly bigger than in the rest ones.

Figure 42 shows the comparison between real and predicted data for every model. As it can be observed, objectives have been accomplished since for every Business Intelligence method that has been used, both estimated and real values are very similar. This Figure also ensures that MLPNN is the method which best predicts the oil level, since it is the one which presents the maximum similarities between estimation and reality.

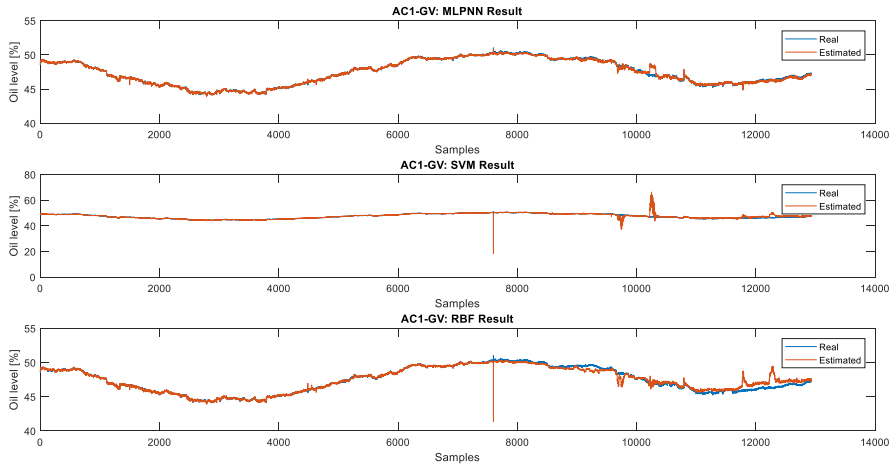


Figure 42. Comparison of AC1-GV prediction of developed models.

Besides, Figure 43 shows the difference between the two variables which have been represented in the last Figure 42 for every developed model. Then, it can be observed from Figure 43 that for both MLPNN, SVM and RBF predictions error is null. This is another confirmation about the obtained accuracy in the developed project to estimate the attribute AC1-GV.

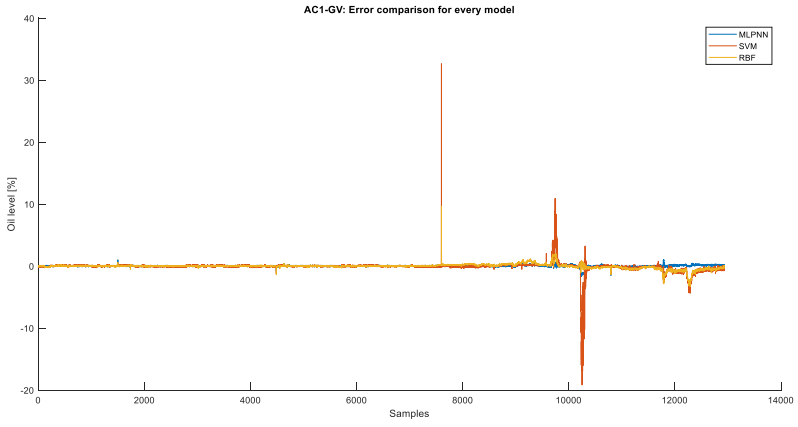


Figure 43. Comparison of AC1-GV committed error of developed models.

To conclude, main objectives related to predict variables have been successfully achieved.

7.2 ANOMALIES DETECTION

When real-world is being analysed, as in Business Intelligence jobs, it arises the need og determining which instances are not similar to all others. Such ones are known as anomalies and they are determined by the process call anomaly detection. They can be caused by "by errors in the data but sometimes are indicative of a new, previously unknown, underlying process" [18].

In this project an anomaly is defined as the abnormal behavior of the variable so that the error of a sample for the testing set is either greater than the maximum or smaller than the minimum error committed during the training set. In other words, the signal is considered to have a normal behavior as long as the difference between the estimation and the real data is included in the interval (minimum error during training set, maximum error during training set). Consequently, the anomalies signal will be set to zero when it is observed a normal behavior. Otherwise the anomalies' value will be equal to the committed error for that sample.

Another detail to consider is that results are shown in percentages, so that 1% of diversion in anomalies means that for that sample, 3 litres are diverted.

Parallelly to the normal behavior models, only ACGV1 anomalies are explained in this section. However, AC3-GV is also explained in order to demonstrate that not only punctual anomalies are detected in this project, but also serious failures that strongly influences the hydropower plant operation. Others outputs are commented in Annex B.

7.2.1 AC1-GV

7.2.1.1 MLPNN

Considering Figure 32, the developed MLPNN model achieved an accurate estimation of the AC1-GV, except for the last testing period, from November of 2017 to May of 2018. For this reason, only a little number of samples are expected to show an anomaly.

Figure 44 shows a graph which depicts the anomalies when the model is predicting the oil level with unknown data. In this variable the three testing periods have been joined so that from the first sample until the sample number 2575 data is referred to the testing 1 period, from the sample number 2576 until the sample number 55378 data are part of the second testing period (from June to October of 2017) and anomalies from the testing 3 period is represented from the sample 55379 until the end (from November of 2017 to May of 2018).

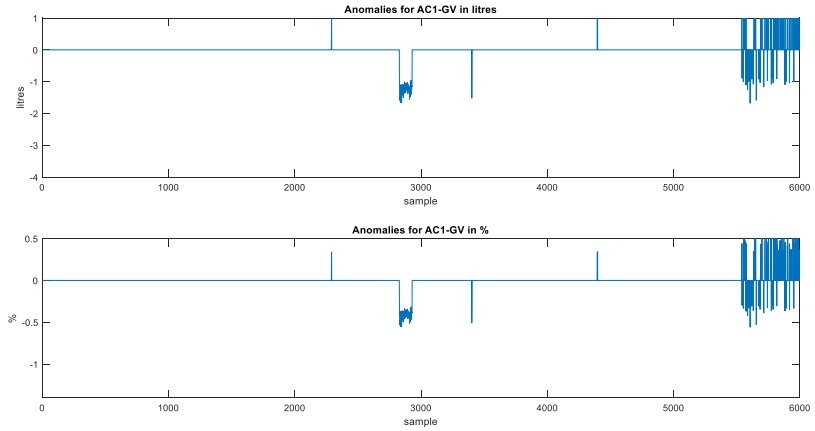


Figure 44. Anomalies detector for the prediction of AC1-GV with the MLPNN model.

Figure 45 confirms that prediction for the last testing period is very different from the others. The reason why this may be happening is because real values are not the mean of all the measurements which are taken for an hour. Consequently, anomalies detection results are commented regardless this data set.

Because prediction is very accurate, anomalies signal values is mostly composed by zeros except for those that are between samples 2829 and 2929, related to the beginning of the second testing period. This small period, when the oil level seems to have an abnormal behavior, may be related to the part of the third graph of Figure 41 where red and blue lines are very different from each other (samples between 0 and 500 from June to October 2017 under the title "AC1-GV Neural network with unknown cases ts2). Besides, sample number 3402 shows another big anomaly in the oil level signal.

In total, when data are unknown, oil level of the guide vanes accumulator 1 is being predicted correctly except in 1.93% of samples, which are the ones that present anomalies. This means that only 107 samples out of 5538 are being predicted wrongly. Moreover, the error has an absolute mean value of 2.173 litres, which means a little bit less than 1%, which is reasonable observing testing 1 and testing 2 histograms from Figure 42.

On the other hand, the number of anomalies during the ignored testing period in this section is equal to 1058 samples out of 2277 measurement this set contains, which means that 46% of the data which belong to the period from November of 2017 to May of 2018 present an abnormal behavior. This fact is another proof of the fact that in Business Intelligence data should be consistent and uniform. Errors are soared during this period because their meaning is not the same as the one from the others sets.

Additionally, from Figure 45 it can be also inferred that last testing period has a greater number of anomalies in comparison with the previous two, which are represented in blue and red in the graph of the left.

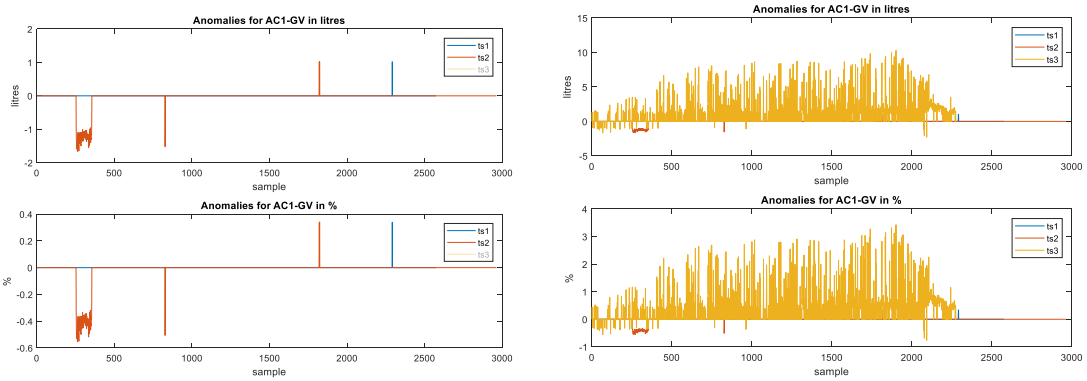


Figure 45. Comparison of data anomalies among the three testing periods.

7.2.1.2 SVM

Support Vector Machine is the second model which has been used to predict the oil level of the guide vanes accumulator 1. According to Figure 37 from Normal Behavior Model Chapter , results are a little worse than the obtained ones from the MLPNN but very similar.

Actually, same samples which were inaccurately predicted are, approximately, the ones which show bigger errors in SVM estimation, such as those at the beginning of the second period of testing.

Once the prediction has been carried out, anomalies are calculated following the same methodology which has been mentioned: the prediction of a sample is considered to be an anomaly when its error is either bigger than the absolute maximum committed during the training set or smaller than the absolute minimum.

For this model, errors during the third testing period are so big that they are not represented in the graphs. Otherwise, anomalies of others periods can be well appreciated, since they are very small in comparison with the ignored ones. Another reason why ts3 is ignored is that, as it has been stated, its measurements are not a mean hourly value, as data from other sets are.

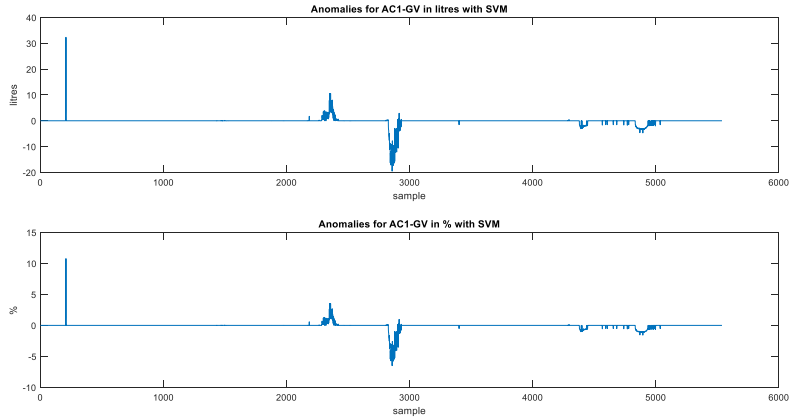


Figure 46. Anomalies detector for the prediction of AC1-GV with the SVM model for ts1 and ts2.

It can be inferred from Figure 46 that the number of anomalies has risen regarding the MLPNN model. For SVM model, without considering ts3, there are 523 out of 5538 samples that are considered as abnormal. This means 9.44% of testing period samples are not being

correctly predicted with SVM. In addition to this numbers, the anomalies have a mean of 0.317 litres.

In case the third testing period is taken into account, anomalies soar sharply (See Figure 47).

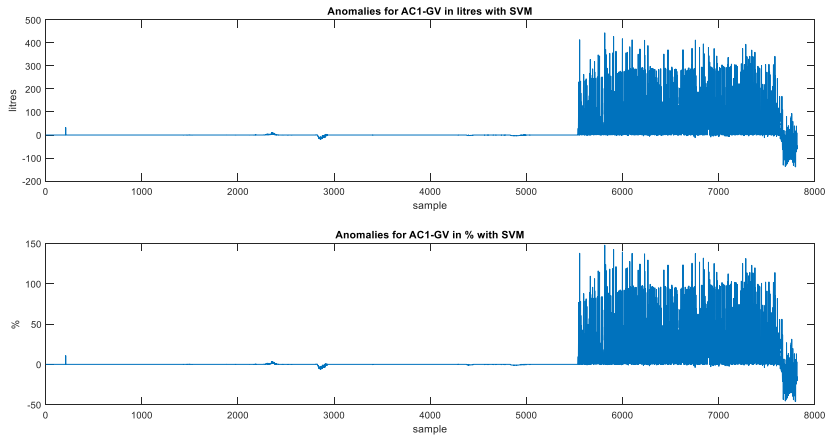


Figure 47. Anomalies detector for the prediction of AC1-GV with the SVM model.

Indeed, the percentage of samples which present anomalies rise from the previous 9.44% to 34.4%, being 67.5 litres of oil the mean value of error in the estimation.

7.2.1.3 RBF

The last model which has been carried out to predict the guide vanes accumulator 1 oil level is the RBF. Similarly to the previous models, from Figure 40, which depicts training and testing sets' results it can be inferred that they are not precise and accurate as the obtained in MLPNN.

Figure 48 shows the anomalies variable based on the obtained results from Matlab and regardless data from November of 2017 to May of 2018, corresponding to testing 3 period. The reasons why this measure has been taken are the same ones already mentioned in the SVM section. By observing Figure 46, it seems RBF prediction presents approximately the

same number of anomalies as SVM. However, a thorough comparison is written in the next part of the thesis.

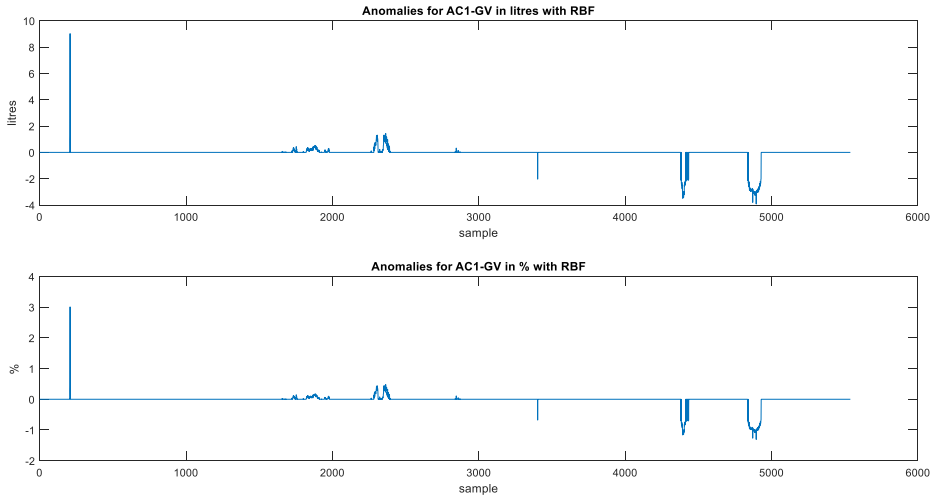


Figure 48. Anomalies detector for the prediction of AC1-GV with the SVM model for ts1 and ts2.

Indeed, 7.05% of samples (390 out of 5538) are considered part of the abnormal behavior, which is close to the previous 9% obtained by using SVM model. Nonetheless, the mean value of the anomalies is greater: 1.17 litres. If last testing period is taken into account, such number will be much higher: 33.3% of samples would be anomalies and their mean value is equal to 27.9 litres.

7.2.1.4 Comparison among MLPNN, SVM and RBF

The aim of this section is to compare the results which have been obtained in the three developed models: MLPNN, SVM and RBF. As it has been carried out in SVM and RBF models, values which belong to the third testing period, from November of 2017 to May of 2018, are neglected in this comparison.

Firstly, Table 16 compares the percentage of anomalies that are detected with every model as well as the mean value of them. As it has been commented on before, the best result is obtained with the MLPNN. Then, SVM identifies a larger amount of anomalies, which may be greater too due to the mean value. And last, RBF shows better result than the SVM but

very similar to it, since the amount is practically the same. However the mean is almost the half of the SVM's anomalies. This means that RBF's anomalies may be smaller.

	MLPNN	*SVM*	*RBF*
Number of anomalies [%]	1.93%	9.44%	7.05%
Mean of the anomalies (litres)	1.2 litres	3.36 litres	1.17 litres

Table 16. Comparison of AC1-GV anomalies detector result.

In order to understand the numbers of Table 16, Figure 49 depicts the anomalies signal detected by the three models. Again, it shows the difference between the most accurate prediction, carried out employing the MLPNN model, with the two others. Both SVM and RBF seem to identify anomalies at the same samples approximately. Nevertheless, SVM anomalies are greater. This is reasonable according to Table 16 numbers.

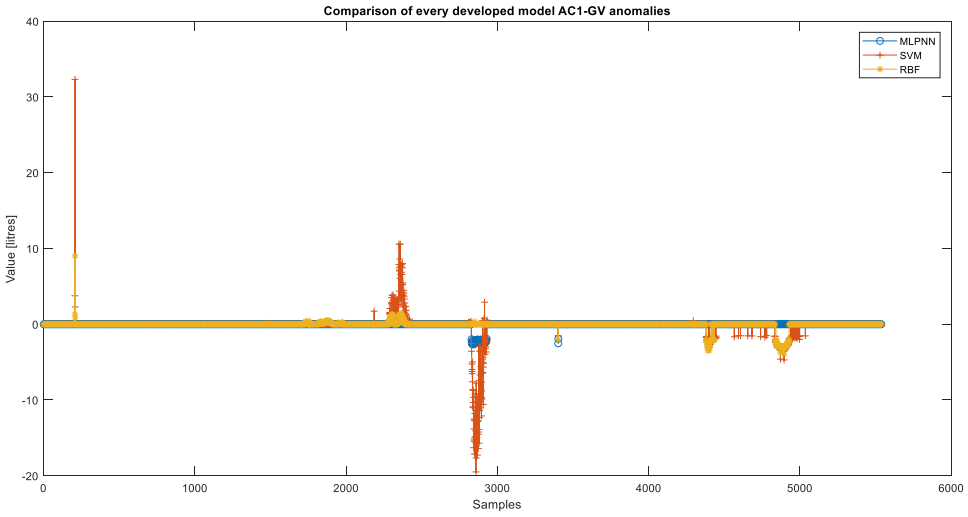


Figure 49. AC1-GV anomalies detected by MLPNN, SVM and RBF models.

7.2.2 AC3-GV

AC1-GV demonstrated that the developed proyect is able to predict punctual anomalies. However it can also anticipate serious failures that complicate the station performance. For instance, AC3-GV normal behavior patterns shew an unusual behavior: estimations were very different from real data. This, together with anomalies detection and health indicators results presented that AC3-GV is suffering from significant oil leakages.

Consequently anomalies are numerous, as observed in Figure 95, regardless the model. Indeed, though last testing set has not been included in represented graphs, more than half of the samples are big anomalies according to Table 17. As a result, more than half of the samples show errors in their predictions. Besides, anomalies average value is big.

	MLPNN	*SVM*	*RBF*
Number of anomalies [%]	59.8%	60.7%	76.58%
Mean of the anomalies (litres)	6.9 litres	9.3 litres	7.4 litres

Table 17. Comparison of AC3-GV anomalies detector result.

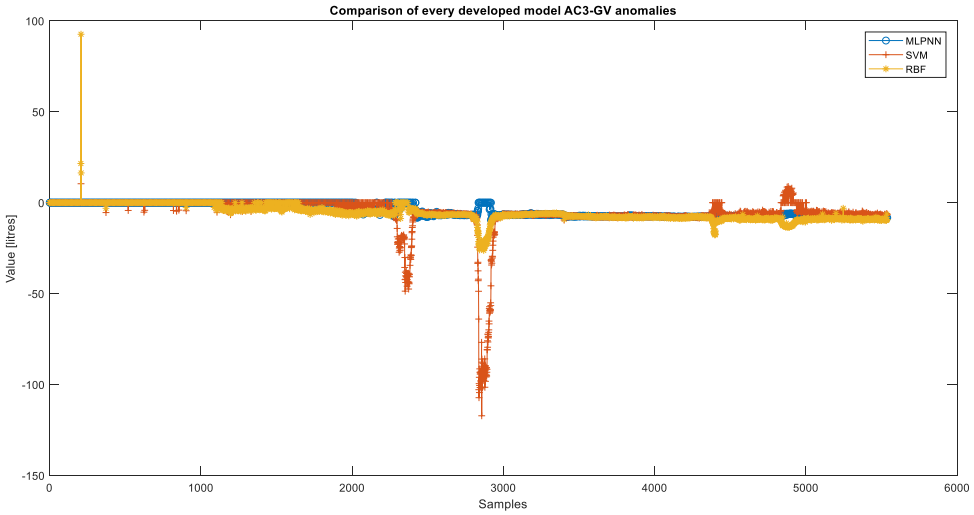


Figure 50. AC3-GV anomalies detected by MLPNN, SVM and RBF models.

7.3 HEALTH INDICATORS

Health indicators are required in any Business Intelligence project in order to measure its performance and progress over time towards the goals that must be accomplished [19]. It helps the developers know how similar estimations are in comparison with real data. It is also useful to study the times a limit of error is surpassed and establish an alarm whose meaning would be either revising the station performance or making an update of the model's development.

The developed health indicator in this project is calculated in two different ways:

On the one hand the first one is based on presenting the number of errors that the model commits hourly when real data values are unknown.

On the other hand, the aim of the second health indicator is to show the error change during the testing period. The way it has being calculated is by constructing a vector which stores the slope value during this period.

Only AC1-GV and AC3-GV health indicators are explained in this section. The first one is an example of a variable whose behavior follows its prognosis. AC3-GV has been added in order to present that significant failures are also predicted. Additional information related to the others variables can be found in Annex B.

Besides, it should be reminded that last period of testing, corresponding to dates between November of 2017 and May of 2018, presents high error rates, and consequently, the health indicator is expected to have a lot of fluctuations. Because it is due to the data nature, that has different meaning than the others data sets, it is going to be mostly ignored in the comments.

7.3.1 AC1-GV

7.3.1.1 MLPNN

Considering the obtained results from this model (Figure 32) and the anomalies detector, and regardless the testing 3 data set (from sample 5524), the worst expected period according the first developed heath indicator (See Figure 51) is the one which is referred to the abnormal behavior at the beginning of the second period of testing (See non-zero values in Figure 49). Besides, it can be inferred that during ts1 and ts2 periods most errors in predictions are punctually, except for a short period, close to the hour number 700, which is coincides with the beginning of ts2, as mentioned. Whereas after the hour number 1381, when ts3 starts, most estimations are wrong since in Figure 51 they present a 100% of errors, as expected due to the data inconsistency in comparison with the previous data sets.

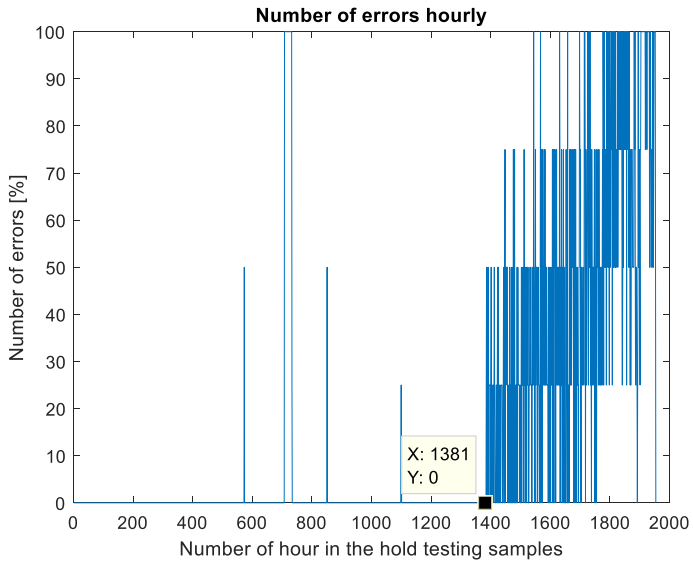


Figure 51. First AC1-GV health indicator for MLPNN model: Number of errors per hour.

The second health indicator is depicted in Figure 52. It presents the change of error in two ways: the first one ignores the symmetric slope the gradient adopts every time an error in the prediction exists and it is shown the absolute slope. The second graph from Figure 52

presents the change of error showing when it is being negative or positive (prediction is greater or smaller than real data) and it also presents the recovery of the slope (symmetry of non-zero values): when errors stop existing on the estimation of the variable, the error signal comes back to its zero state, which means a perfect prediction.

Both health indicators are coherent regarding each other since fluctuations in the number of hour coincide with fluctuations on the number of sample. The conversion from number of sample to number of hour consists in dividing by four.

Hence, considering both health indicators, the developed prediction of AC1-GV variable is accurate until October of 2017, when the second testing data set ends, since it is equal to zero in most of the samples. On the other hand, another model should be developed if the retrieved data from the data sets stop being the hourly mean, as it happens from November of 2017, when the last testing set begins. This can be seen by observing the fluctuations that both health indicators suffer from at the end (from the hour number 1381 in the first one and from the sample 5542 in the second one).

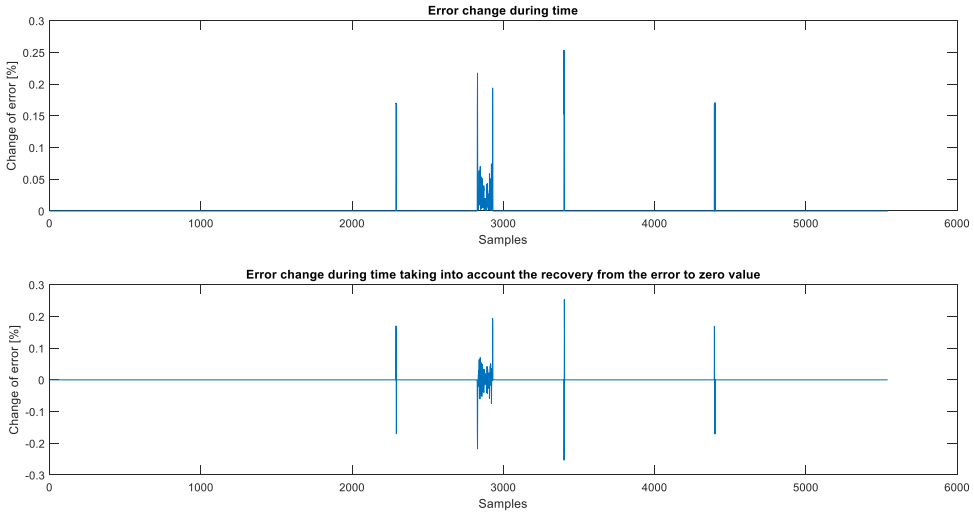


Figure 52. Second AC1-GV health indicator for MLPNN model: Error change in % of litres.

All in all, it is concluded than even though there are some anomalies in the prediction of the guide vanes accumulator 1 oil level, most of the samples are being accurately calculated, being based on the input data without knowing the expected result.

7.3.1.2 SVM

Similarly to MLPNN, two health indicators have been developed. The first one depicts the number of committed errors every hour when estimating AC1-GV with SVM most accurate model. Results can be observed in Figure 37. This graph ignores the health indicator over the third testing period, which corresponds to the month between November of 2017 and May of 2018 for the same reasons as it has been done in section 7.2.1.2, SVM's Anomalies Detection: when they are included results from the others data sets can not be appreciated because of their huge difference.

Because prediction is less accurate than with the MLPNN, Figure 53, which shows the first health indicator result, presents more peaks than Figure 51. It seems the prediction for the testing periods not only has the big errors committed in the MLPNN (Figure 52 contains the same peaks that Figure 54 presents), but also some others little ones, since SVM's health indicator has more fluctuations.

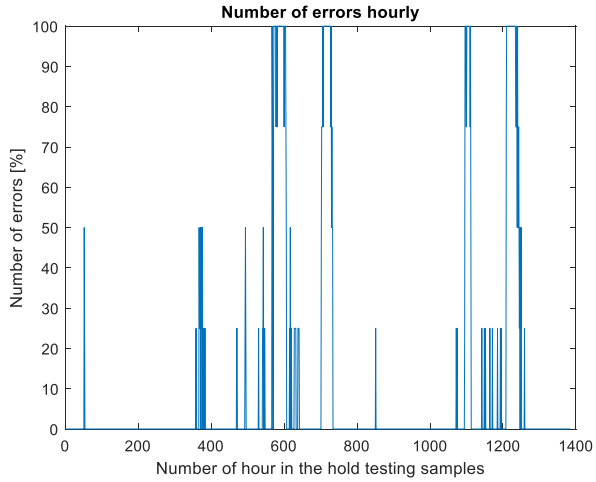


Figure 53. First AC1-GV health indicator for SVM model for ts1 and ts2: Number of errors per hour

Second health status indicator consists in measuring how the error is changing during time, hourly to be specific. It appears that there are three long sub-periods when oil level has been suffering from an abnormal behavior. Moreover, there are some several punctual anomalies, such as the first one, almost at the beginning of the testing set.

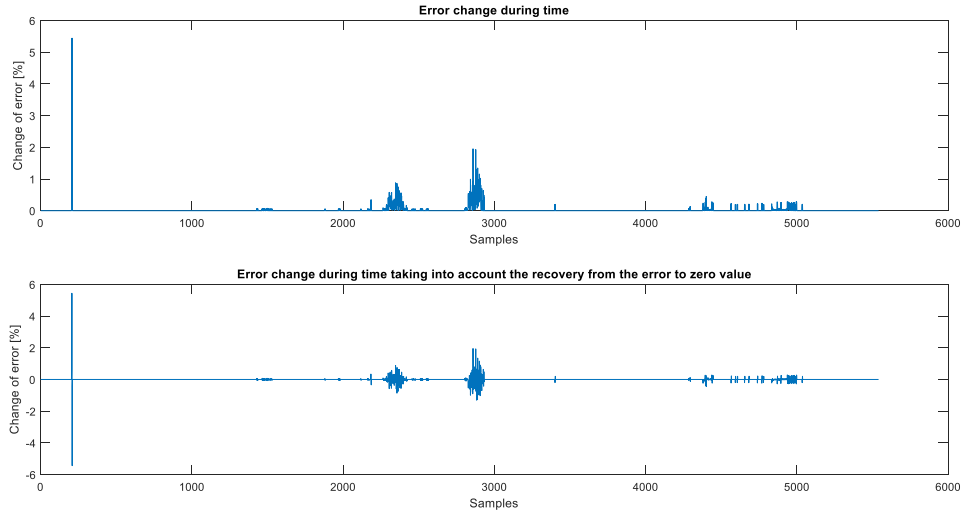


Figure 54. AC1-GV wealth status indicator for SVM model for ts1 and ts2: Error change

To sum-up, SVM predictions are worse than the ones result from the MLPNN model. This is the reason why more anomalies have been found in this section and also the mean of the anomalies is greater.

7.3.1.3 RBF

The same two health indicators have been developed for the las used model in this project: RBF. Similarly to the SVM model, last testing period has been ignored to explain the results in order to observe clearer the outcome for the others, since the ts3 doesn't contain consistent data with the rest of data.

The first health indicator depicts the number of errors the model commits every hour. As it was expected, results are not very different from the ones obtained with SVM model.

Nevertheless, as RBF model generates a less accurate prediction according to previous sections, the health indicator presents more and higher fluctuations, representing the fact that there are more anomalies.

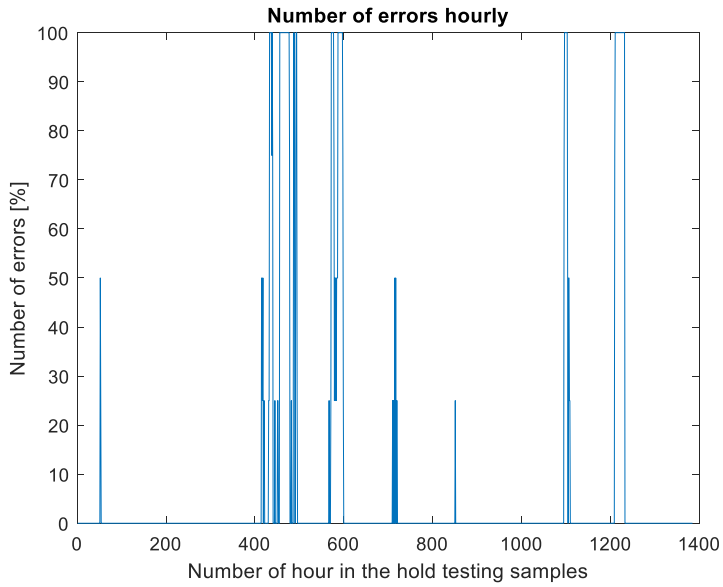


Figure 55. First AC1-GV health indicator for RBF model for ts1 and ts2: Number of errors per hour

The second health indicator shows how the error evolves during the prediction period. The main difference with the SVM model is that in the RBF's prediction there are more short periods of abnormal behavior rather than punctual anomalies. Consequently, there are more rises and falls in the graphs from Figure 55.

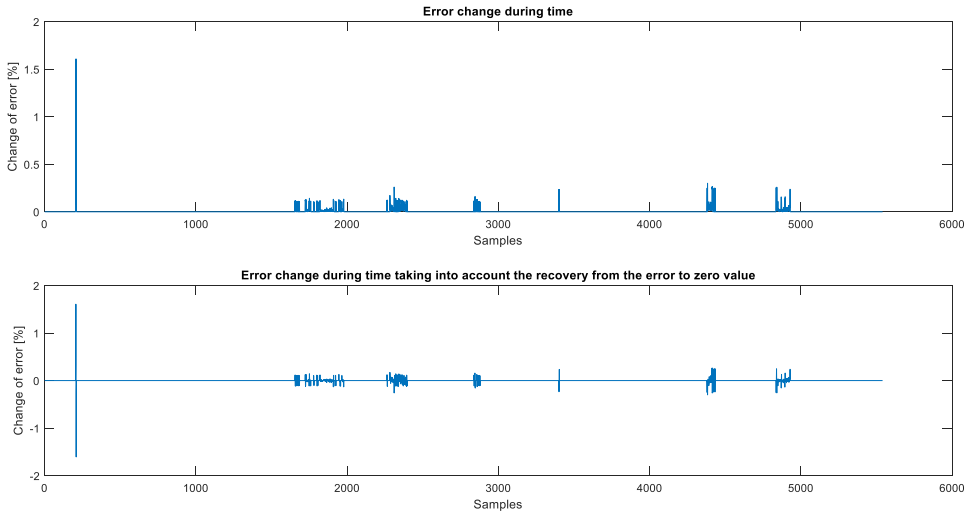


Figure 56. AC1-GV health indicator for SVM model for ts1 and ts2: Error change

All in all, RBF predictions are also worse than MLPNN and this is the reason why more anomalies have been found in this section and also the mean of the anomalies is greater. Nevertheless, error is not so big, so RBF technology is a good method to support the principal one, which is MLPNN.

7.3.1.4 Comparison among MLPNN, SVM and RBF

Finally, all health indicators already commented are compared in order to extract some conclusions clearer. In this comparison, for the same reasons as explained above, date from November of 2017 to May of 2018 is not considered in this section. This period corresponds to the third testing set.

Figure 57 and Figure 58 compare the health indicators of the three models. Between samples 2000 and 3000 (in hours, this period is from hour number 300 to hour number 700 approximately) appears to be the worst behavior of the oil level variable because the three models detect short periods of anomalies. Same happens between sample 4000 and 5000, or between hour 1050 and 1300. Nonetheless, during this latter mentioned period, MLPNN only detects punctual anomalies. These two figures strengthen the conclusion which states that MLPNN is the best model to predict the oil level of the guide vanes accumulator 1, and

that SVM and RBF are worse; as well as that RBF's anomalies are smaller but more frequent than with SVM.

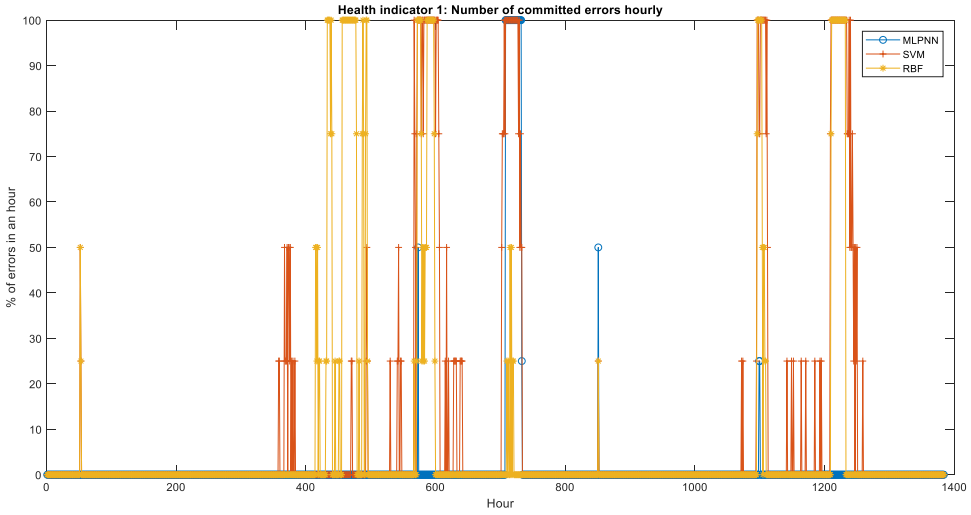


Figure 57. AC1-GV health indicator for MLPNN, SVM and RBF models: Number of errors hourly.

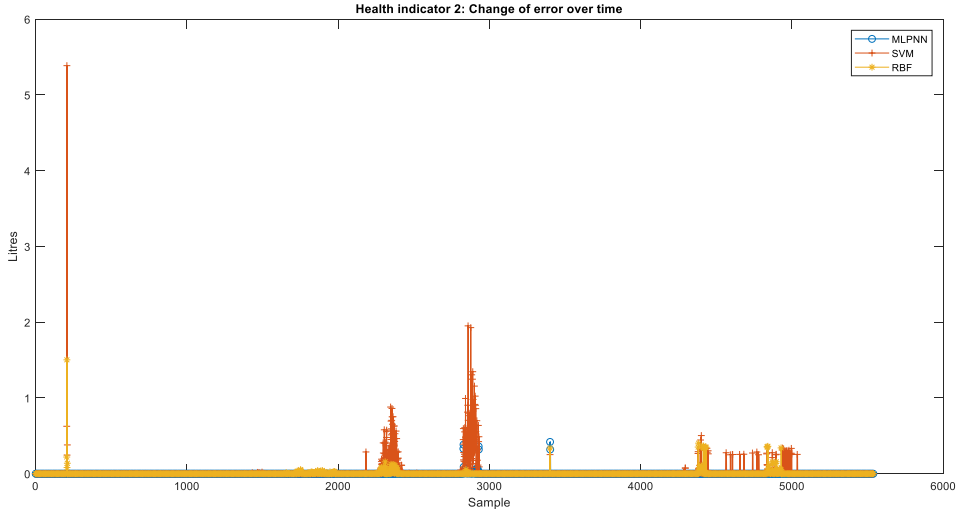


Figure 58. AC1-GV health indicator for MLPNN, SVM and RBF models: Error change.

All in all, three models are accurate to predict the guide vanes accumulator 1 oil level as long as original data is the mean value of the taken measurements hourly since anomalies

are never greater than 1.5% of the tank accumulator. Besides, the frequency of the times the signal suffers from any abnormal behavior is neither that much, since the worst case is the result obtained with SVM and it only was less than 10% of the samples. Most important conclusion to outstand is that MLPNN has the best results, far from the ones obtained with the others models.

7.3.2 AC3-GV

In line with the oil leakages that AC3 suffers from, several fluctuations are appreciated in both health indicators regardless the model (Figure 59). Such rises and falls reveal that plant workers must be alarmed because of AC3-GV behavior, since health indicators demonstrate a large number of anomalies (first health indicator) and how significant they are (second health indicator).

Figure 59. AC3-GV health indicators for MLPNN, SVM and RBF models.

All in all, despite the errors and anomalies, objectives for AC3-GV variable have been successfully achieved since according to real values it is suffering from oil leakages.

Chapter 8. CONCLUSIONS AND FUTURE WORK

This project describes a methodology for the forecast normal behavior models and anomalies detection behavior conditions in the short-term for some determined Kaplan turbine components.

Normal behavior models (or patterns) have been developed by observing common relationships that exist between a group of input variables and the corresponding output of a specific variable which is predicted. Input variables are chosen by taking the hydropower plant performance during operation and an algorithm results from Matlab into account.

Targeted variables' normal behavior models have been estimated by employing MPLNN, SVM and RBF algorithms, thought the first one's results are the most accurate. However SVM and RBF methods are required to strengthen MLPNN conclusions.

The main aim of the project, which consists in detecting abnormal behavior which may cause the possible failure mode in the station, has been successfully achieved, considering that "abnormal behavior is any significant derivation or difference between the predicted output of the models and its corresponding real observations" [3]. This conclusion is observed in the presented normal behavior models, related with the oil tank level, considered from two different perspectives, such as real values and others estimations; and oil level in the two banks of accumulators of both turbine runner and guide vanes. After training the models, their learned knowledge is tried with new unknown operation samples. During this application, models keep on learning and correcting the forecasted patterns. This process is repeated for every algorithm in order to gain consistency in their results.

Over all, from described results in Chapter 7. , it is inferred that the predicted oil levels were almost equal to the expected values, presenting nor larger errors than 3% got most cases. Besides, while some accumulators oil levels, such as AC1-GV seemed to adopt the expected real values always, others, such as AC3-GV, didn't follow similar values to the predicted

ones. The difference between expected and real values of this latter variable reveals the existence of possible oil leakages. These conclusions were carried out by analyzing normal behaviors deviations.

As a result, when any oil level accumulator, either related to guide vanes or turbine runner operation, is suffering from such kind of anomalies, they are also presented in the oil tank hub, where every oil flow is taken from. That is the reason why real and predicted data are also different at some samples from each in graphs from Annex B related to AC-TANK variable. Therefore, a closer monitoring is needed to search for the cause of this anomaly.

Additionally the oil tank hub's normal behavior pattern was also developed by considering its temperature and others oil level accumulators' already calculated normal behavior patterns as inputs. However this model suggests a deeper study than what it has been developed with more real data to reach firm conclusions.

Anomalies detector and health indicators have been carried out for every studied variable and used algorithm to complement the normal behavior models. Their results are reasonable according the obtained prognosis. Their results are graphs that help to visualize if anomalies are happening punctually or in burst. Besides, they measure the difference between real and expected values, which provide information to set some alarms.

In this sense, punctual anomalies are shown in those variable anomalies detectors and health indicators where predicted patterns coincide with real behavior for most samples, such as AC1-GV, whereas anomalies burst are shown in those variables which seemed to have oil leakages over time, such as AC3-GV.

Besides, burst anomalies are present in every model when period from November of 2017 to June of 2018 are analysed, since data are not one-hour average values, as the training set which learning proceed has been done with. This lies on the importance of the data consistency and constant meaning.

All in all, this project leads to the implementation of the developed research in the plant so that it can be executed over time. Nonetheless, it is suggested to still verify its accuracy with

additional real data in order to improve the prognosis and reduce committed errors, so that anomalies would be better identified, and health indicator would gain precision. Moreover, the project may be completed by searching for the main causes of the anomalies in order to be able to carry a documentation out and prepare some solutions guide to cope with them. Such suggestions would contribute, together with the developed project to maximize the hydropower station profit.

BIBLIOGRAPHY

[1] World Energy Council, «Energy sources: Hydropower,» 2017. [En línea]. Available: https://www.worldenergy.org/data/resources/resource/hydropower.

[2] C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

[3] M. Á. Sanz-Bobi, T. Welte y L. Eilertsen, «Anomaly indicators for hydropower plant components based on patterns of normal behavior,» 2018.

[4] A. Tapper, «A fatigue investigation in a Kaplan,» 2016. [En línea]. Available: http://www.diva-portal.se/smash/get/diva2:946471/FULLTEXT01.pdf.

[5] Wikipedia, «Kaplan turbines,» 2012. [En línea]. Available: https://en.wikipedia.org/wiki/Water_turbine.

[6] P. G. I. F. Liliana Topliceanu, «Functional Problems and Maintenance Operations of Hydraulic Turbines,» *TEM Journal*, vol. 5, nº 1, pp. 32-37, 2016.

[7] Wikipedia, «Gantt chart,» 2017. [En línea]. Available: https://en.wikipedia.org/wiki/Gantt_chart.

[8] M. Á. Sánz-Bobi, «Business Intelligence course: Neural Networks Notes,» 2016. [En línea]. Available: www.sifo.comillas.edu.

[9] Analítica Web, «Machine Learning y Support Vector Machines: porque el tiempo es dinero,» 2016. [En línea]. Available: https://www.analiticaweb.es/machine-learning-y-support-vector-machines-porque-el-tiempo-es-dinero-2/.

[10] Scikit Learn, «Support vector machines (SVMs,» [En línea]. Available: http://scikit-learn.org/stable/modules/svm.html.

[11] Wikipedia, «Support Vector Machines,» 2018. [En línea]. Available: https://es.wikipedia.org/wiki/M%C3%A1quinas_de_vectores_de_soporte.

[12] Department of Electrical Engineering National Cheng Kung University, «Radial Basis Function,» 2005. [En línea]. Available: https://www.slideshare.net/kylin/section5-rbf.

[13] ibiblio, «Redes de función de base radial (RBF),» 2003. [En línea]. Available: https://www.ibiblio.org/pub/linux/docs/LuCaS/Presentaciones/200304curso-glisa/redes_neuronales/curso-glisa-redes_neuronales-html/x185.html.

[14] C. McCormick, «Radial Basis Function Network (RBFN) Tutorial,» 2013. [En línea]. Available: http://mccormickml.com/2013/08/15/radial-basis-function-network-rbfn-tutorial/.

[15] SINTEF, «Memo: E4,» Trondheim, 2017.

[16] MathWorks, «Matlab: fitrsvm,» https://se.mathworks.com/help/stats/fitrsvm.html, 2018.

[17] Università degli Studi di Torino, «Radial-Basis Function Networks,» [En línea]. Available: http://www.di.unito.it/~botta/didattica/RBF_1.pdf.

[18] R. Chalapathy, A. K. Menon y S. Chawla, «Anomaly detection using One-Class Neural Networks,» 18 February 2018. [En línea]. Available: https://arxiv.org/pdf/1802.06360.pdf.

[19] H. Oei, «Defining Your Business Intelligence: Key Performance Indicators,» 2016. [En línea]. Available: https://www.5xtechnology.com/why-business-intelligence-

blog/bid/103212/why-should-i-care-about-business-intelligence-key-performance-indicators.

[20] Wikipedia, «Sequential Minimal Optimization,» 2018. [En línea]. Available: https://en.wikipedia.org/wiki/Sequential_minimal_optimization.

ANNEX A – PROJECT DEVELOPMENT

A.1. DATA PRE-ANALYSIS

Data pre-analysis for all attributes except the power is explained in detail in this Annex.

In this way, the executed data filtering is based on the graphs from Table 11. These are presented below again in larger sizes in order to give a better observation.

- ***Attribute 2. Rotational Speed Turbine***

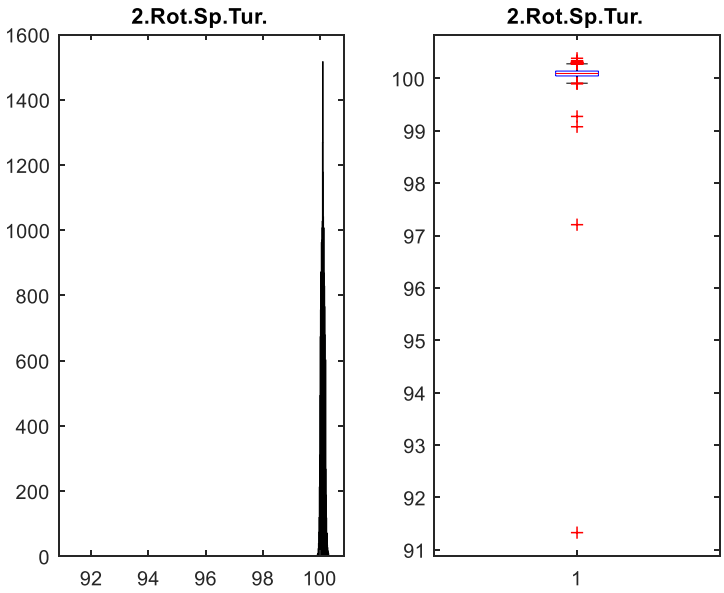


Figure 60. Histogram and boxplot of the attribute 2.Rotational speed of turbine

Figure 60 shows that this attribute has several outliers. In fact, the furthest one is close to 91% while more of the 45% of the data is between 99 and 101%.

As a result, every row whose value on the rotational speed of turbine is below 97% are removed.

- ***Attribute 3. Guide vane position (opening)***

Even though this attribute presents some outliers, they are not very different from the values which are equal to the first and the third quartiles that represents the 25% and 75% respectively (Figure 61). Consequently, the filtering condition according to the guide vane position signal is that values must be greater than 54%.

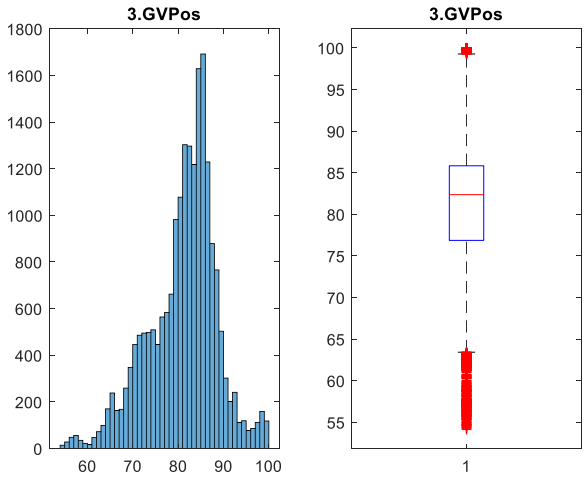


Figure 61. Histogram and boxplot of the attribute 3.Guide vane position

- ***Attribute 4. Water flow***

According to Figure 62, the scenario for the water flow is very similar to the previous one. Therefore, the condition set by this attribute is that no water flow data must be below 140 m^3/s.

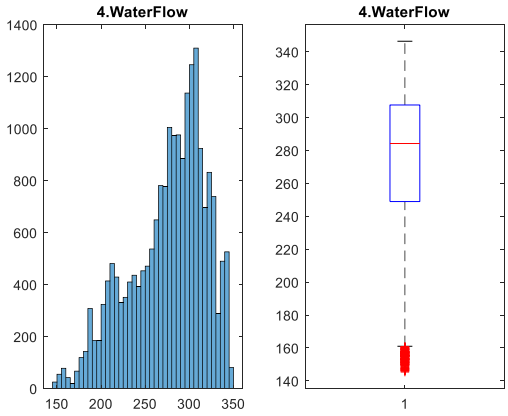


Figure 62. Histogram and boxplot of the attribute 4. Water flow

- ***Attribute 5. Bearing cooling water temperature***

Figure 63 presents that this attribute does not have any outliers. That is the reason why no condition is set by the bearing cooling water temperature to filter any data.

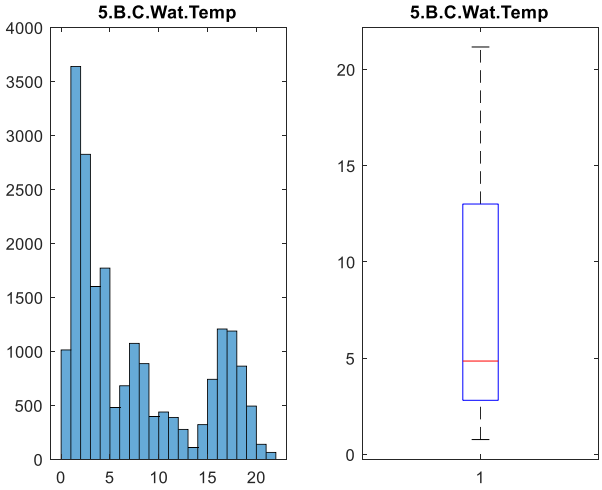


Figure 63. Histogram and boxplot of the attribute 5.Bearing cooling water temperature

- ***Attribute 6. Headwater level***

Half of the headwater level values take part between 30.18 and 30.21 MOH (Figure 64). Besides, those points that are below 30.16 MOH or above 30.24 MOH are considered

outliers. Hence, the rows whose headwater level values are not in the mentioned interval have been discarded.

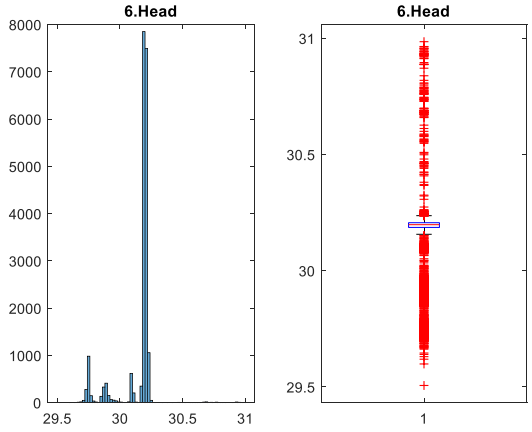


Figure 64. Histogram and boxplot of the attribute 6.Headwater level

- ***Attribute 7. Tailwater level***

Even though it seems this attribute has several outliers according to the boxplot from Figure 65, it can be inferred from the histogram that it follows approximately a normal distribution. For this reason, only those outliers that are almost isolated are discarded, which means that the condition consists in filtering values which are over 13 MOH.

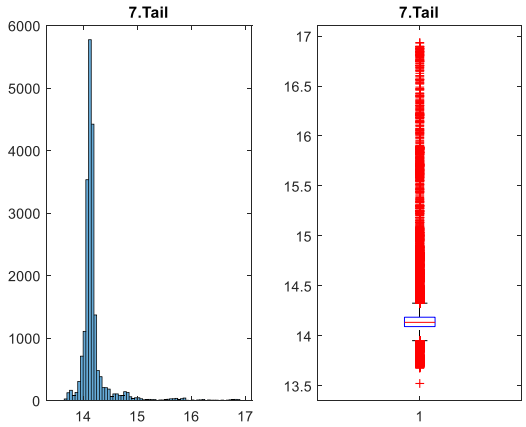


Figure 65. Histogram and boxplot of the attribute 7.Tailwater level

- ***Attribute 8. Turbine runner oil pressure***

Turbine oil runner attribute has also a distribution similar to a normal one (Figure 66). However some outliers should be discarded, such as those that are below 136 bar or greater than 137 bar.

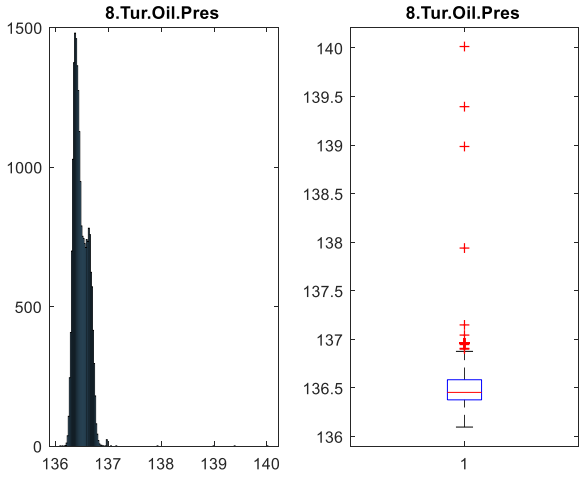


Figure 66. Histogram and boxplot of the attribute 8.Turbine runner oil pressure

- ***Attribute 9. Guide vanes oil pressure***

Due to the bell-shaped the histogram presents and the confirmation to this fact by observing the small proportion of outliers from the boxplot of Figure 67, no condition is added according to the guide vanes oil pressure.

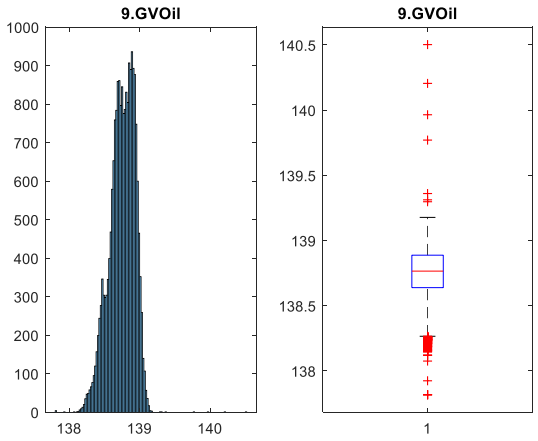


Figure 67. Histogram and boxplot of the attribute 9.Guide vanes oil pressure

- ***Attribute 10. Leakage oil tank oil level***

This attribute shows that both first and third quartiles as well as the mean and the median share the same value (Figure 68). However it can be observe that in spite of presenting most of its numbers equal to 39.8%, there are outliers distributed in a continuous way. Hence, only those data which have a zero value in this attribute have being discarded.

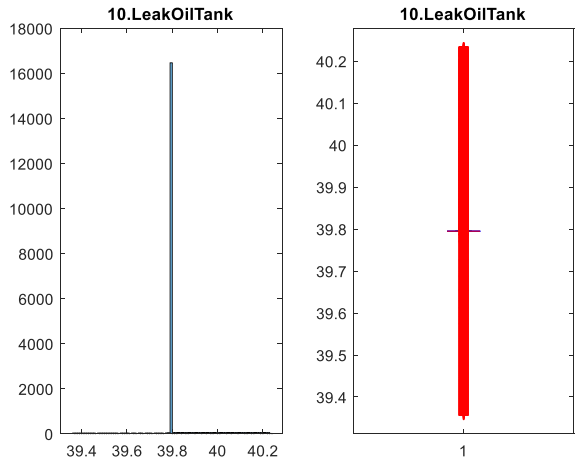


Figure 68. Histogram and boxplot of the attribute 10.Leakage oil tank oil level

- ***Attribute 11. Oil tank oil level***

Due to the lack of outliers in the oil tank oil level, no conditions are added. See Figure 69.

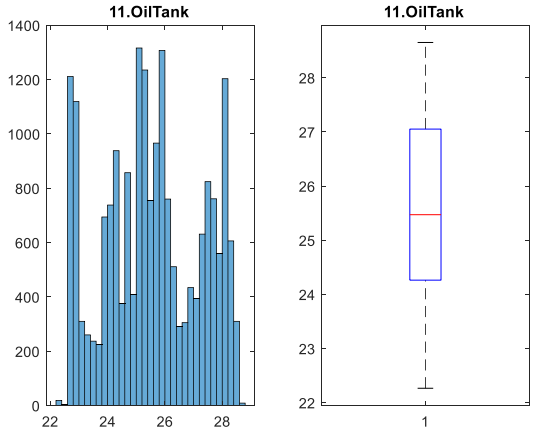


Figure 69. Histogram and boxplot of the attribute 11.Oil tank oil level

- ***Attribute 12. Oil tank temperature***

According to the graphs from Figure 70, only those rows which have a value equal to zero in the oil tank temperature have been filtered.

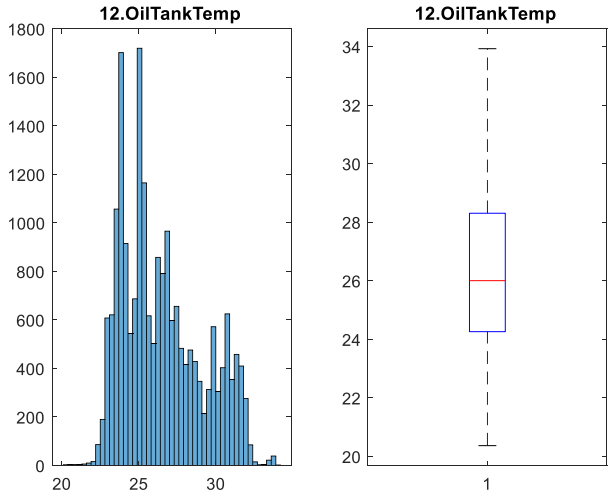


Figure 70. Histogram and boxplot of the attribute 12.Oil tank temperature

- ***Attribute 13. Accumulator 1 guide vanes operating mechanism oil level***

Due to the lack of outliers in the accumulator 1 guide vanes operating mechanism oil level, no conditions are added. See Figure 71.

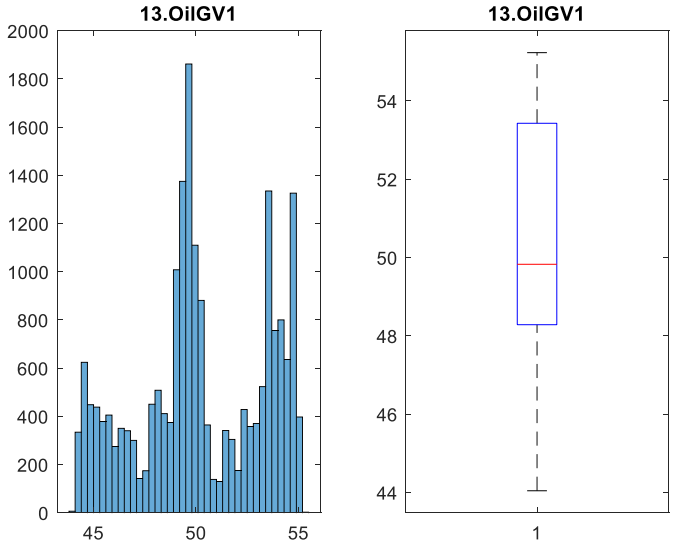


Figure 71. Histogram and boxplot of the attribute 13.Acc1 guide vanes oil level

- ***Attribute 14. Accumulator 2 guide vanes operating mechanism oil level***

Similarly to the previous attribute, the lack of outliers in the accumulator 2 guide vanes reveals that no conditions are added. See Figure 72.

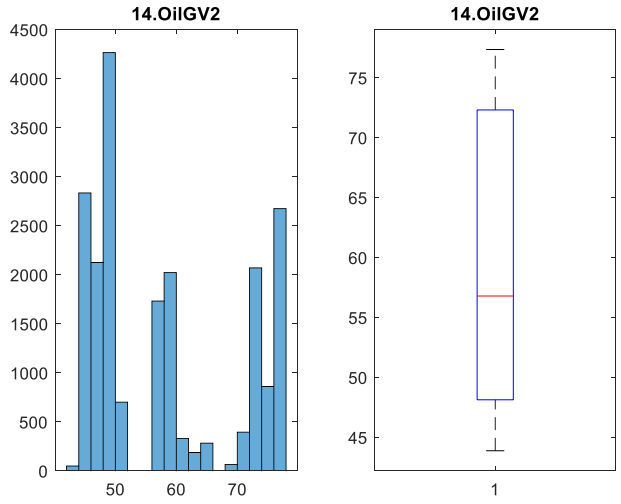


Figure 72. Histogram and boxplot of the attribute 14.Acc2 guide vanes oil level

- ***Attribute 15. Accumulator 3 guide vanes operating mechanism oil level***

This variable presents some outliers in the greatest values. Moreover last research inferred an abnormal behavior of this oil level due to some oil leakages. Therefore it is very important to filter those values that are greater than 46% (Figure 73).

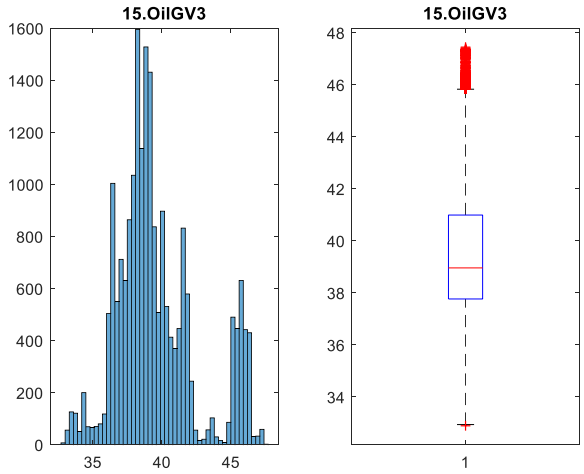


Figure 73. Histogram and boxplot of the attribute 15.Acc3 guide vanes oil level

- ***Attribute 16. Accumulator 1 turbine runner oil level***

Once again, no outliers are shown in the graphs of Figure 74, correspondent to the accumulator 1 turbine runner oil level. For this reason, there isn't any filtering condition related to this attribute.

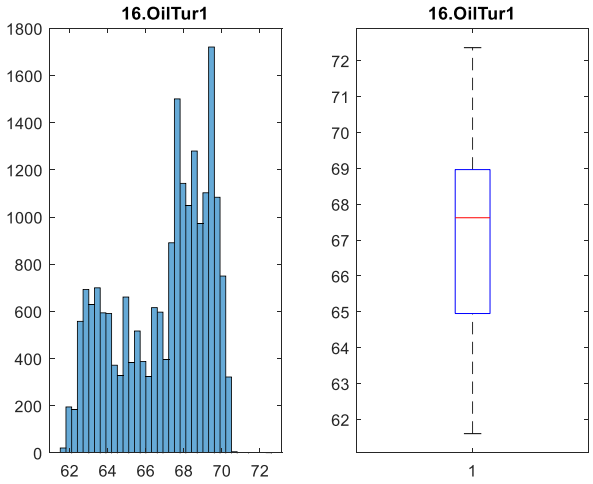


Figure 74. Histogram and boxplot of the attribute 16.Acc1 turbine runner oil level

- ***Attribute 17. Accumulator 2 turbine runner oil level***

The oil level from the accumulator 2 of the turbine runner hasn't any outliers (See Figure 75). Therefore, no conditions have been added.

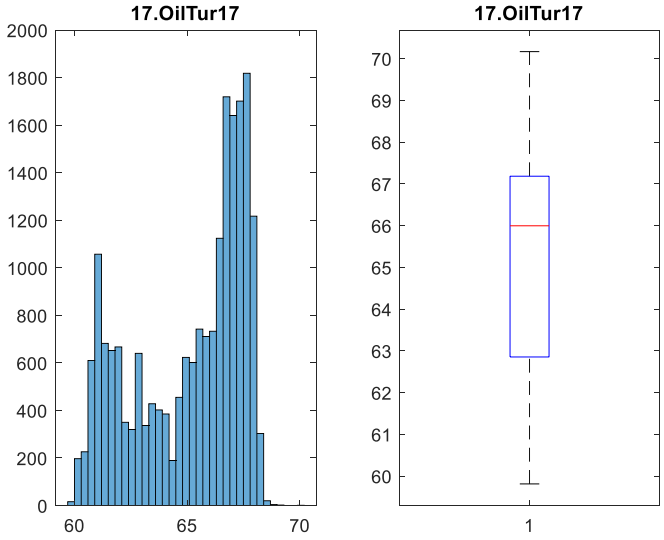


Figure 75. Histogram and boxplot of the attribute 17.Acc2 turbine runner oil level

A.2. FEATURE EXTRACTION

This section presents an explanation in detail about the feature extraction and complements the information commented on Chapter 4.

According to the Table 13, all oil accumulators of the guide vanes operating mechanism have the same suggested and final inputs to the models. Similarly to oil accumulators turbine runner. Hence, they are going to be explained by grouping them into guide vanes and turbines accumulators.

- ***AC1-GV, AC2-GV, AC3-GV***

The suggested outputs for these signals are listed below:

- 5.B.C.Wat.Temp
- 6.Head
- 7.Tail
- 8.Tur.OilPres
- 9.GVOilPres
- 12.OilTankTemp
- 16.OilTur1

In order to prove that the three outputs share those items, both AC2-GV and AC3-GV `relieff` results are going to be shown (Figure 76 and Figure 77). The ones from AC1-GV were already depicted in Chapter 6.3.

Despite both Figures X and X depict that the attribute accumulator 2 turbine runner oil level would be a good input since it appears as the second most important one, it is proportional to the accumulator 1 turbine runner oil level, which means they are redundant. For this reason accumulator 2 turbine runner oil level was discarded.

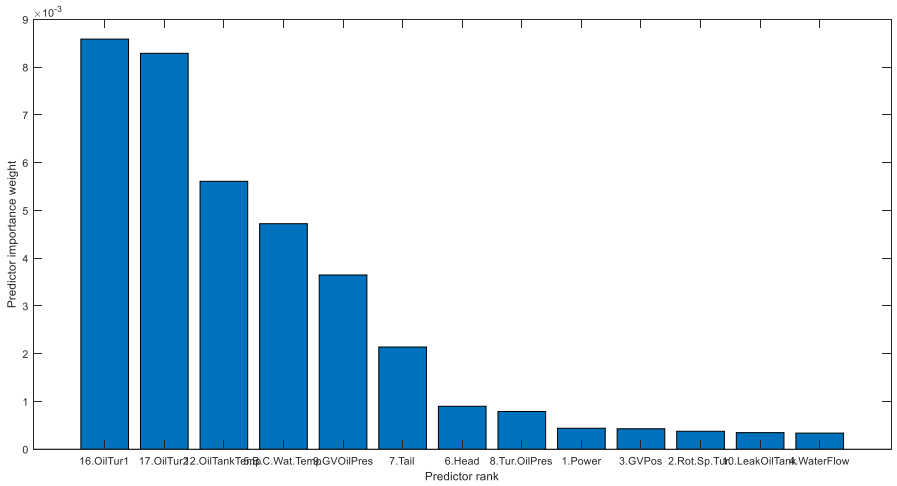


Figure 76. Predict importance for the output AC2-GV behavior.

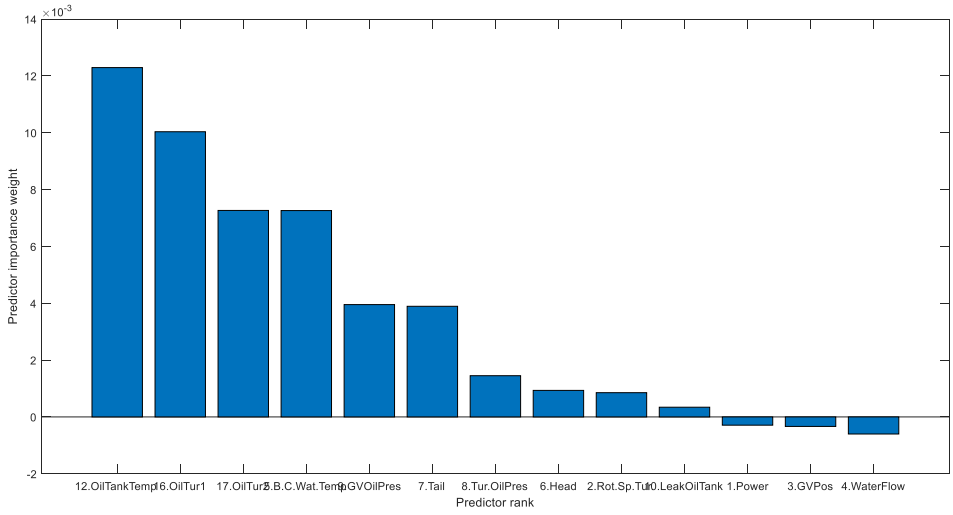


Figure 77. Predict importance for the output AC3-GV behavior.

To continue, from that list of above, only the first item (5.B.C.Wat.Temp) was discarded because of some reasons which were explained in the Chapter 6.3. The rest of the proposed inputs compose the input signals to predict the determined oil levels because they accomplish the needed requirement related to the training and testing sets. To demonstrate so, Figure 78,Figure 79, Figure 80, Figure 81, Figure 82 and Figure 83 are added below. They depict the PDF by training and testing sets for every input:

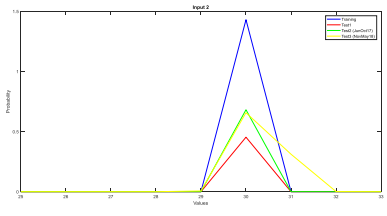


Figure 78. Probability Density Function by training and testing periods of 6.Headwater level.

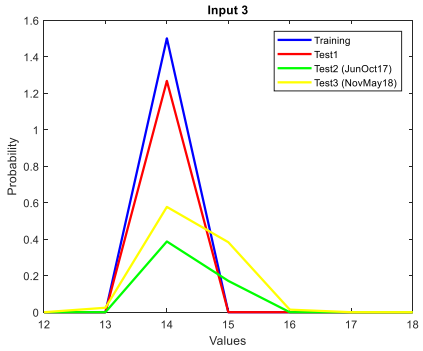


Figure 79. Probability Density Function by training and testing periods of 7.Tailwater level.

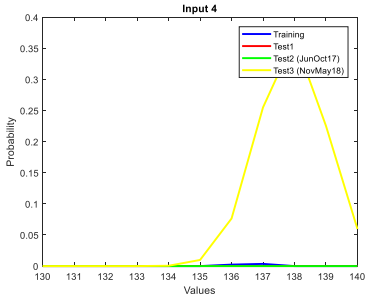


Figure 80. Probability Density Function by training and testing periods of 8.Turbine runner oil pressure.

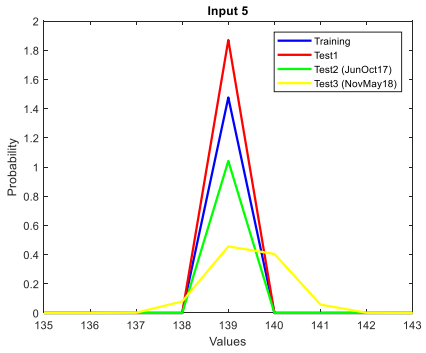


Figure 81. Probability Density Function by training and testing periods of 9.Guide vanes oil pressure.

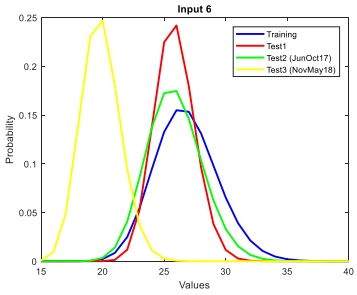


Figure 82. Probability Density Function by training and testing periods of 12.Oil tank temperature.

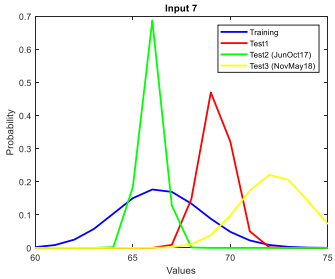


Figure 83. Probability Density Function by training and testing periods of 16.Accumulator 1 turbine runner oil level.

From the Figure 78,Figure 79, Figure 80, Figure 81, Figure 82 and Figure 83 it is inferred that the fourth test set does not reach the determined requirement for every input signal. This information was ignored since prediction for that period it is expected to be unusual. The reason for this is because, as mentioned before, values from this data set are not the hourly mean of every value but a random one which had occurred during a precise hour.

- ***AC1-TUR, AC2-TUR***

For these outputs, the suggested inputs according to the `relieff` function are the following ones:

- 2.Rot.Sp.Tur.
- 5.B.C.Wat.Temp
- 7.Tail
- 8.Tur.OilPres
- 9.GVOilPres
- 12.OilTankTemp
- 13.OilGV1

From this list, the items which were discarded were: 2. Rotational speed turbine and 5. Bearing cooling water temperature. The reason why the latter one is not part of the final inputs for these output signals has been described in Chapter 6.3. On the other hand, the rotational speed turbine was neither part of the inputs attributes because its resulted weight from the `relieff` function was small. This fact makes the concerned variable to be redundant with the others valid inputs.

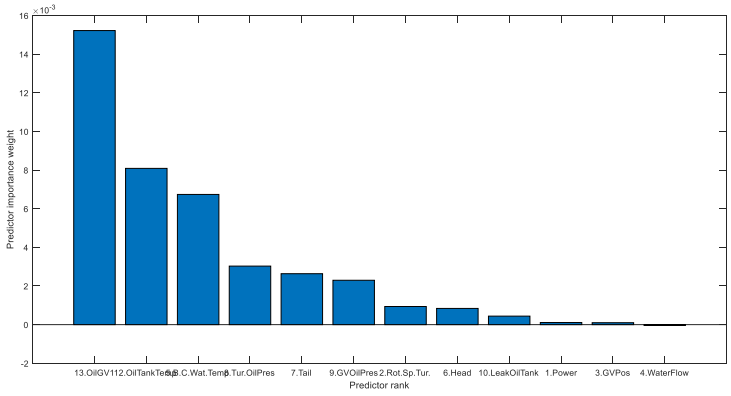


Figure 84. Predict importance for the output AC1-TUR behavior.

Besides, from Figure 84 it is inferred that it is essential that the attribute of the accumulator 1 guide vanes operating mechanism oil level is valid as an input since it is the one which has

the highest meaning of the output. Fortunately, Figure 85 shows that testing values are included the in the interval which embraces the training data set.

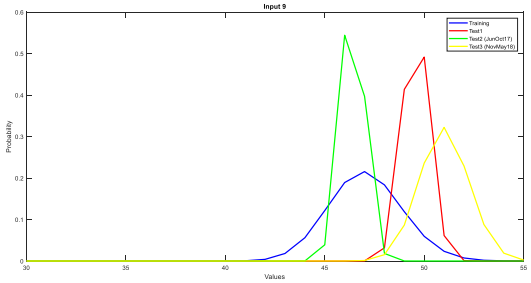


Figure 85. Probability Density Function by training and testing periods of 13.Accumulator 1 guide vanes oil level.

- ***AC-TANK (Hub) based on real data***

Finally, when `relieff` function was applied to the hub tank oil suggested inputs ended being the final ones. They were the following ones:

- 1.Power
- 6.Head
- 7.Tail
- 8.Tur.OilPres
- 9.GVOilPres
- 12.OilTankTemp
- 13.OilGV1
- 16.OilTur1

Similarly to the previous explanations, all of them result valid to be the final inputs.

- ***AC-TANK (Hub) based on forecasts***

This section consists in estimating the same latter variable, hub tank oil, but instead of considering real values of AC1-GV, AC2-GV, AC1-TUR and AC2-TUR attributes as inputs, it takes their predictions for every determined model. Nevertheless, after carrying out a thorough study, better results came up when adding real values from oil tank temperature (attribute number 12 from data sets) and removing predictions from AC3-GV, due to its studied oil leakages, which will be explained later in Annex B. This changes respect to Figure 6 can be better understood by visualizing diagram from Figure 86. In this Figure, this research is referred to the output where Detector 7 tag is allocated.

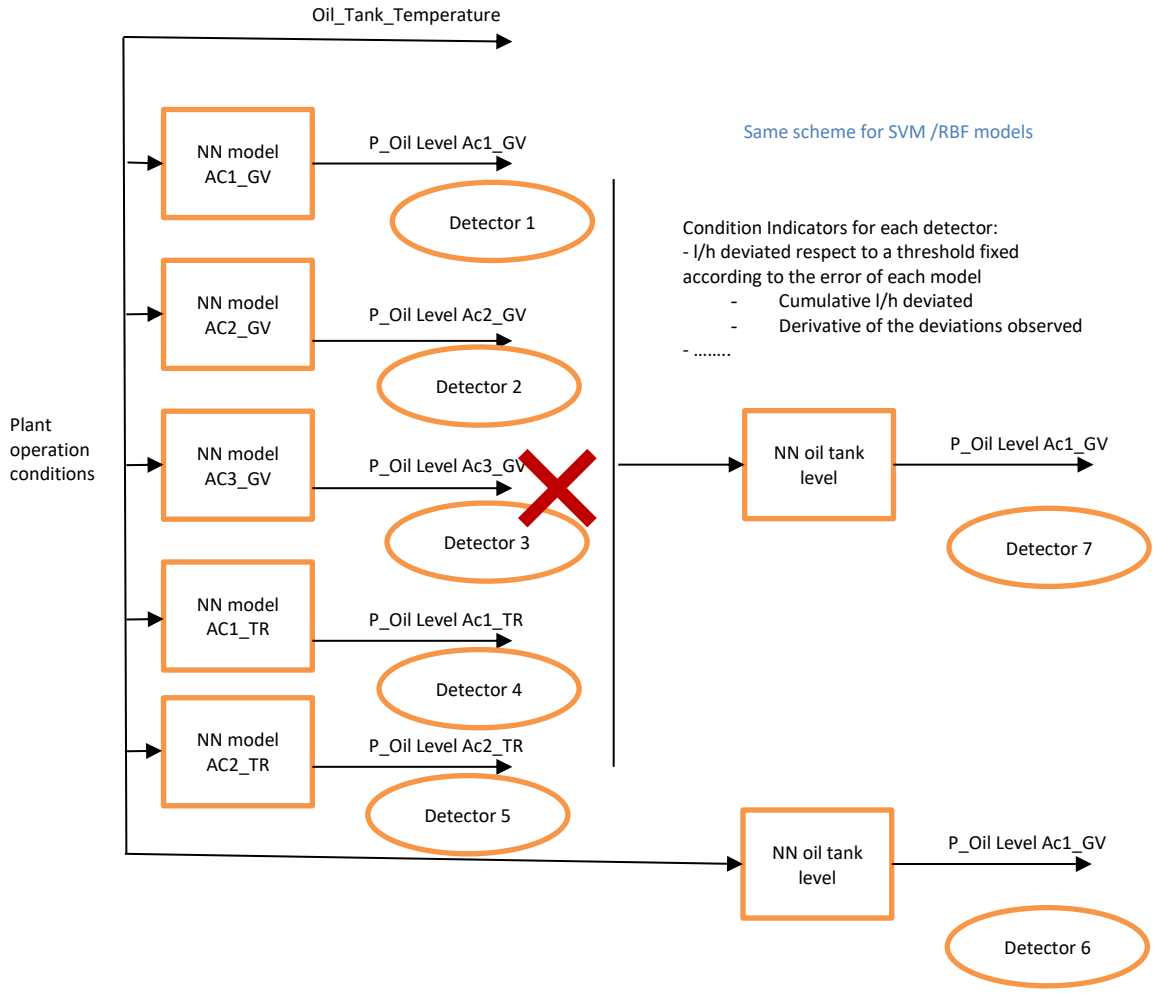


Figure 86. Re-Definition of inputs for AC-TANK based on predictions study.

For this scenario, number of valid real values has been reduced because inputs for this variable are the output of the already trained MLPNN, SVM and RBF models, which means inputs are the prediction that have been developed for the training data sets. In this sense, to estimate the AC-TANK from others variables' forecasted values, the training set is the set which was the first testing set (from February to May of 2017) and the testing sets are the periods from June to October of 2017 and from November of 2017 to May of 2018. However this latter one is not used in the research because its values are single random ones instead of mean hourly values. Table 18 updates shows this shift.

Previous models data sets				***AC-TANK based on predictions as inputs***			
Data set	***ID***	***Period***	***% of data***	***Data set***	***ID***	***Period***	***% of data***
Known data in tr		Real Data		***Known data in tr***		Predictions from previous tr set.	48.6%
Training set	tr	9/April/2016 – 13/Feb/2017	48.6%	***Training set***	tr	13/Feb/2017 – 31/May/2017	16.9%
First testing set	ts1	13/Feb/2017 – 31/May/2017	16.9%	***First testing set***	ts1	1/June/2017 – 31/Oct/2017	19.5%
Second testing set	ts2	1/June/2017 – 31/Oct/2017	19.5%	***Second testing set***	ts2	31/Oct/2017 – 21/May/2018	15%

Third testing set	ts3	31/Oct/2017 – 21/May/2018	15%		

Table 18. Re-Definition of data sets for AC-TANK based on predictions study.

Consequently, because length of every data set is cut off, it is expected to get more error and more anomalies in the forecasts than in the others studies. As soon as more data is retrieved, results will improve.

ANNEX B – RESULTS

Annex B is a complement of Chapter 7, which is related to results from the developed models that estimate some attributes from the data sets as well as from the analysis of the predictions by anomalies detectors and health indicators.

Only one estimated variable has been explained in Chapter 7, AC1-GV. Then, the others studied attributes are commented in this annex.

The way results are explained is by directly compare the error obtained from the developed MLPNN, SVM and RBF models, describing its corresponding parameters in a Table, since the followed methodology has been the same one for AC1-GV, which is defined in Chapter 7.

It must be pointed out that because of the inconsistent of testing 3 data set in comparison with the others, it has not been considered. This has help to provide clearer graphs in Annex B, since during this period predictions are so inaccurate that errors sharply increase. The reasons why this happens is not because of the estimation accuracy but because real data has been retrieved in a different way than the training and others testing sets.

B.1. AC2-GV

B.1.1. NORMAL BEHAVIOR MODELS

AC2-GV prediction reaches its best result when MLPNN, SVM and RBF's parameters are set up as it appears in Table 19.

MLPNN parameters		
	Parameter Name	**Value**
ML PN	Training Ratio	75%
	Validation Ratio	10%

	Testing Ratio	15%
	Maximum number of iterations	10000
	Learning rate	0.001
	Observing the performance every	50
	Minimum performance	1e-5
	Maximum fail	50
	Number of neurons	12
	SVM parameters	
	Parameter Name	**Value**
SVM	Re-training data	Standardized
	Solver	Sequential Minimal Operation (SMO)
	KernelScale	Auto
	RBF parameters	
	Parameter Name	**Value**
RBF	Spread	45

Table 19. AC2-GV MLPNN, SVM and RBM models' parameters.

As a proof, below there is Figure 87, which depicts the graphs of real and estimated values sorted by method:

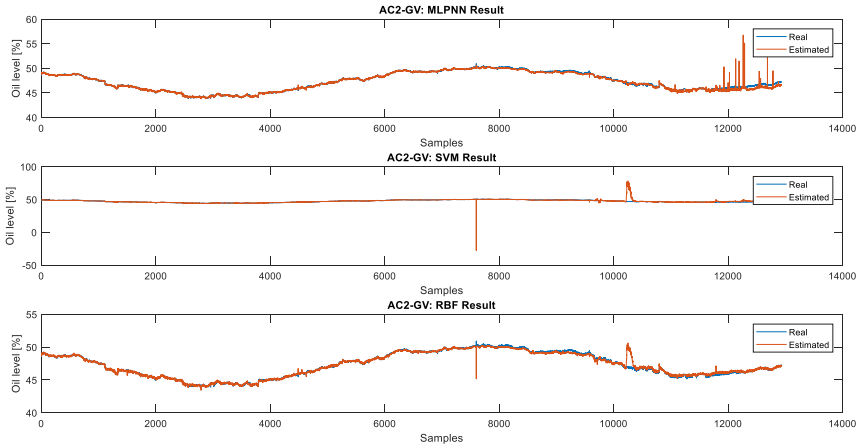


Figure 87. Comparison of AC2-GV prediction of developed models.

Figure 87 shows that both predicted and expected values are very similar since they are almost completely overlapped in every graph. It is inferred that the best method is MLPNN but though SVM and RBF present higher error rate, its result is accurate enough to support the MLPNN outcome.

Figures 84 and 85 depict some error data. The first one depicts the error variables whereas the second one presents error histograms. From both of them it can be inferred that SVM is the model which worst predicts AC2-GV output. However, its maximum error is 2%, which is acceptable. As a result, AC2-GV normal behavior has been successfully predicted.

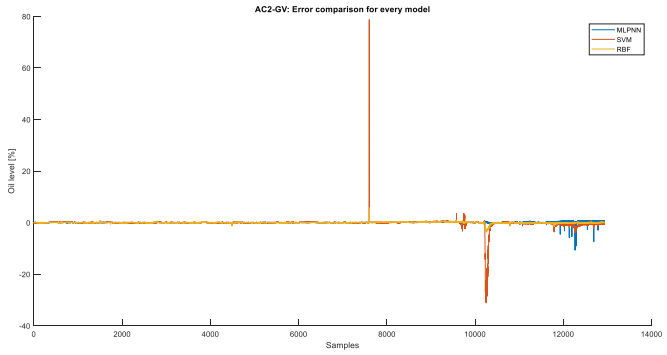


Figure 88. Comparison of AC2-GV committed error of developed models.

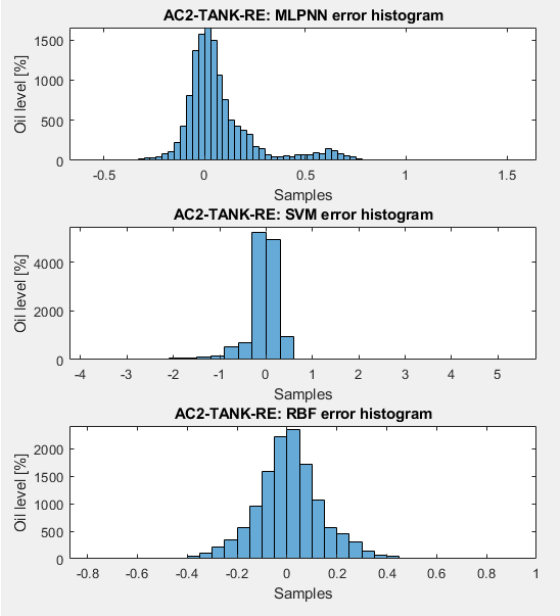


Figure 89. AC2-GV prediction error histograms by method.

B.1.2. ANOMALIES DETECTION

In line with normal behaviour patterns' results, when anomalies are calculated, it is observed that most samples are equal to zero. In Figure 90 color orange, corresponding to SVM anomalies is the one which presents worst results, as it was expected according what has been already mentioned.

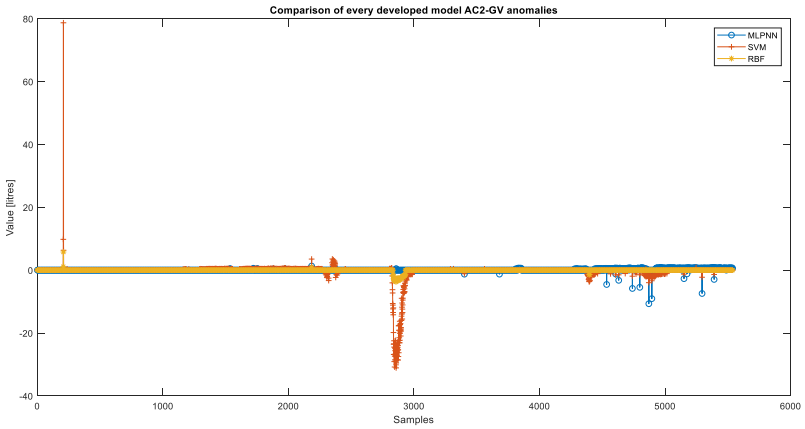


Figure 90. AC1-GV anomalies detected by MLPNN, SVM and RBF models

Table 20 sums up anomalies result of every developed model. While MLPNN has several anomalies with small errors, RBF presents more samples equal to real values. However when RBF's estimation is an error, it is very different from real value. On the other side, SVM has several anomalies whose margin between prediction and reality is almost like RBF's one.

	MLPNN	*SVM*	*RBF*
Number of anomalies [%]	18.5%	21.7%	1.8%
Mean of the anomalies (litres)	0.627 litres	2.34 litres	2.5 litres

Table 20. Comparison of AC2-GV anomalies detector result.

In case last testing period have been included the number of anomalies would have increased, until 37, 43.5 and 26.9% for MLPNN, SVM and RBF, respectively.

B.1.3. HEALTH INDICATORS

Similarly to AC1-GV, two health indicators have been developed. The first one is depicted in Figure 91 and it shows the number of errors every model commits hourly. The more fluctuations they present, the less accurate the prediction is. That's why SVM, which worst results achieved is the curve which more rises and falls has.

However, at the end of the October of 2017 (last samples in the graph) MLPNN seems to start failing more. This graph may be complemented with the error histogram or the second health indicator, because despite MLPNN predictions are not being as accurate at the end than as at the beginning of the testing period, they are small errors, as it has been pointed out.

Related to SVM, similarly to others conclusions, its curve has punctual fluctuations.

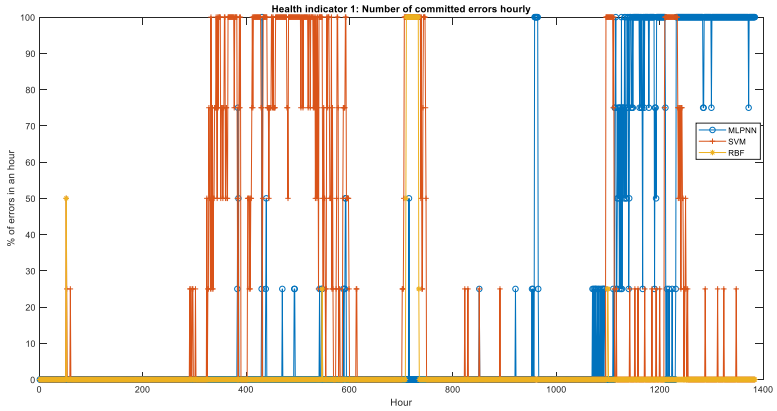


Figure 91. AC1-GV health indicator for MLPNN, SVM and RBF models: Number of errors hourly

Second health indicator is useful to measure, being aware that some samples are inaccurate, how inaccurate they are. In this sense, worries should wake up when there are several large fluctuations in Figure 92. As expected, and in line with anomalies, SVM is the model which is presenting larger number and size of fluctuations.

RBF graph just contains little and punctual rises an MLPNN presents burst errors at the end of the testing period that are no bigger than 2 litres, which means MLPNN's results are acceptable because it is less than 1%.

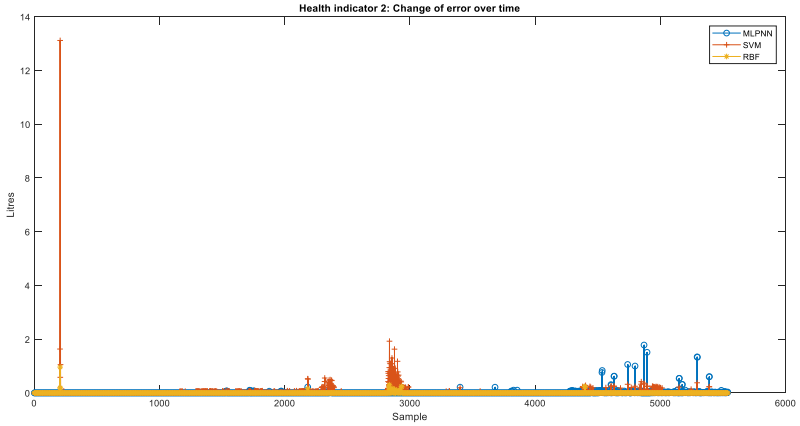


Figure 92. AC1-GV health indicator for MLPNN, SVM and RBF models: Error change

B.2. AC3-GV

B.2.1. NORMAL BEHAVIOR MODELS

Best AC3-GV behavior models have been achieved when MLPN, SVM and RBF methods are configured as stated in Table 21.

	MLPNN parameters	
	Parameter Name	**Value**
MLPNN	Training Ratio	70%
	Validation Ratio	20%
	Testing Ratio	10%
	Maximum number of iterations	500
	Learning rate	0.9
	Observing the performance every	100
	Minimum performance	1e-5
	Maximum fail	10
	Number of neurons	16
	SVM parameters	
	Parameter Name	**Value**
SVM	Re-training data	Standardized
	Solver	Sequential Minimal Operation (SMO)
	KernelScale	Auto
	RBF parameters	
	Parameter Name	**Value**
RBF	Spread	35

Table 21. AC3-GV MLPNN, SVM and RBM models' parameters.

Normal behavior pattern of AC3-GV suffers from several large different samples with real variable, if Figure 93 is observed. Error histograms aren't bell-shaped, and they are not centered in zero.

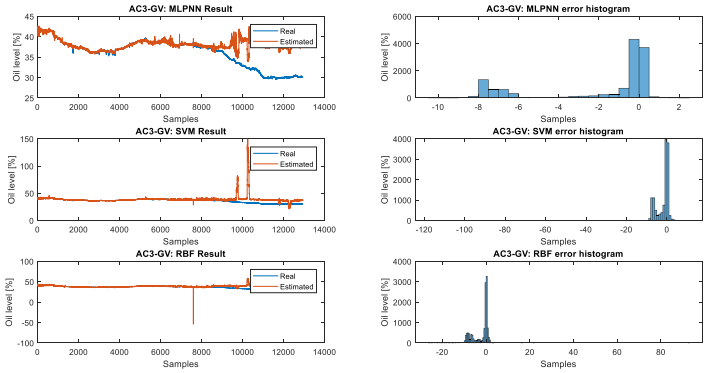


Figure 93. Comparison of AC3-GV prediction of developed models.

It may be that models have not learned the appropriate AC3-GV behavior and that is the reason why estimations are failing. However, according to Figure 94, which depicts real values of AC1-GV, AC2-GV and AC3-GV, this latter one may be suffering from some oil leakages. AC3-GV's mean is much lower and it stops being proportional to the previous one from sample number 800 of the second testing period. This point is also when error is sharply soared in previous Figure 93.

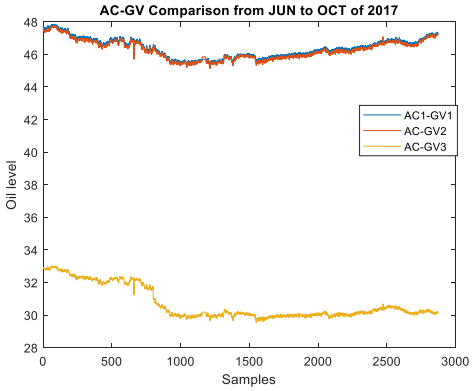


Figure 94. Oil level guide vanes accumulators 1, 2 and 3 signals during the second testing period

B.2.2. ANOMALIES DETECTION

As a consequence of the oil leakages, anomalies are numerous, as observed in Figure 95, regardless the model.

Moreover, though last testing set has not been included in represented graphs, more than half of the samples are big anomalies (Table 22).

	MLPNN	*SVM*	*RBF*
Number of anomalies [%]	59.8%	60.7%	76.58%
Mean of the anomalies (litres)	6.9 litres	9.3 litres	7.4 litres

Table 22. Comparison of AC3-GV anomalies detector result.

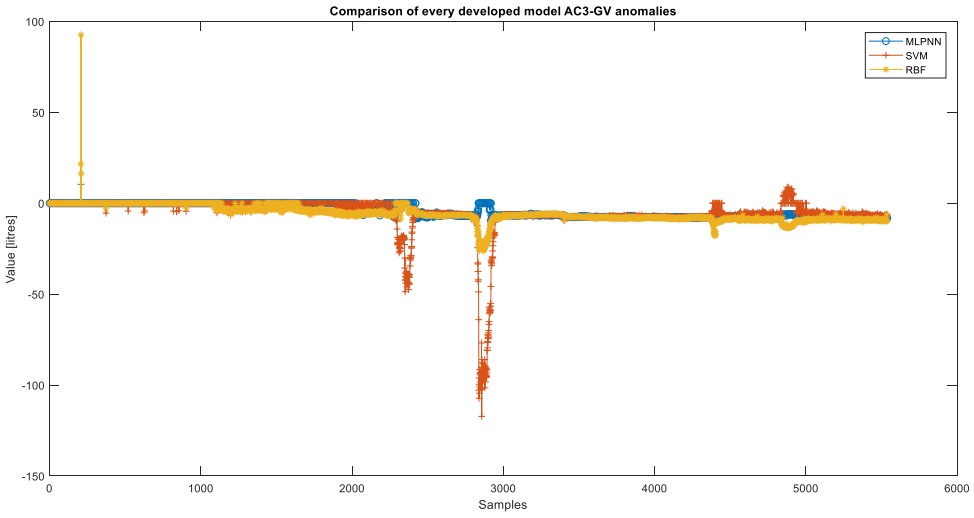


Figure 95. AC3-GV anomalies detected by MLPNN, SVM and RBF models.

B.2.3. HEALTH INDICATORS

Another consequence of the abnormal behavior is the several fluctuations both health indicators presents for every model (Figure 96). Such rises and falls reveal that plant workers

must be alarmed for AC3-GV behavior, since they reveal the large number of anomalies (first health indicator) and their gravity (second health indicator).

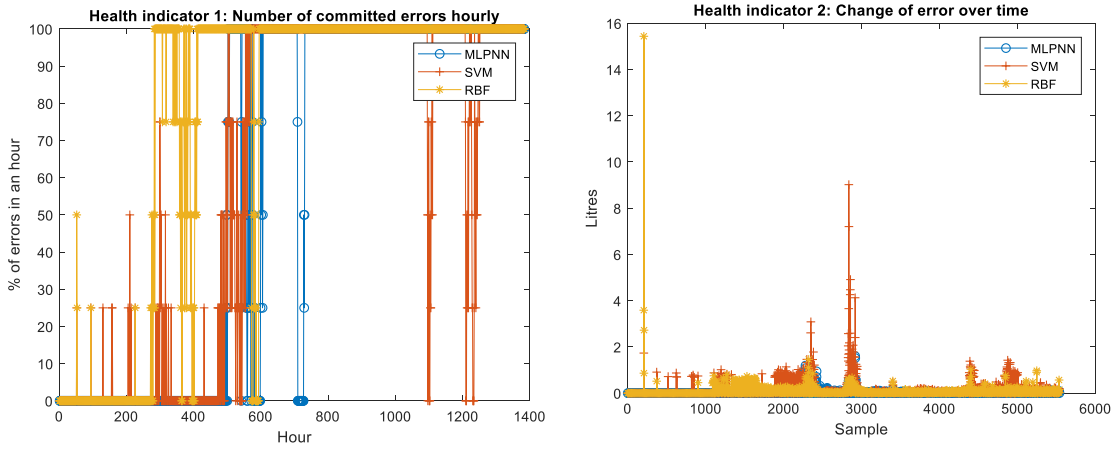


Figure 96. AC3-GV health indicators for MLPNN, SVM and RBF models.

All in all, despite the errors and anomalies, objectives for AC3-GV variable have been successfully achieved since according to real values it is suffering from oil leakages.

B.3. AC1-TUR

B.3.1. NORMAL BEHAVIOR MODELS

Table 23 contains how MLPNN, SVM and RBF models should be configured in order to obtain the best results for getting an accurate AC1-TUR normal behavior model.

MLPNN parameters		
	Parameter Name	**Value**
MLPNN	Training Ratio	80%
	Validation Ratio	10%
	Testing Ratio	10%
	Maximum number of iterations	10000
	Learning rate	0.001
	Observing the performance every	50
	Minimum performance	1e-5
	Maximum fail	30

	Number of neurons	12
	SVM parameters	
	Parameter Name	**Value**
SVM	Re-training data	Standardized
	Solver	Sequential Minimal Operation (SMO)
	KernelScale	Auto
	RBF parameters	
	Parameter Name	**Value**
RBF	Spread	25

Table 23. AC1-TUR MLPNN, SVM and RBM models' parameters.

Except for some SVM samples, Figure 97 shows that estimations are very alike to real values.

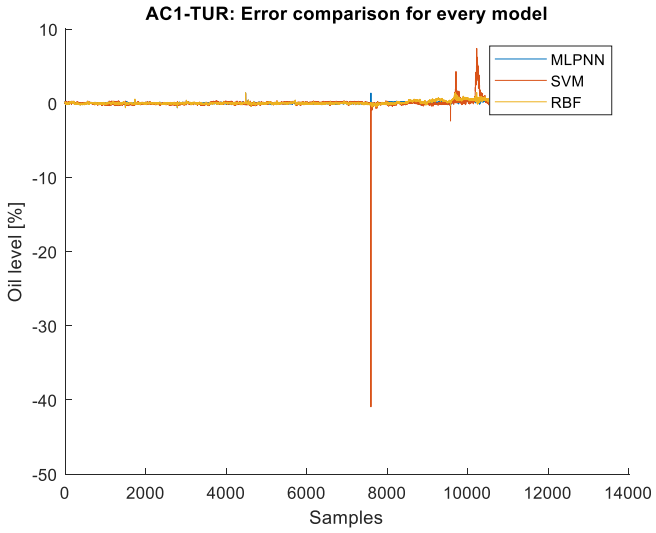


Figure 97. Comparison of AC1-TUR committed error of developed models.

B.3.2. ANOMALIES DETECTION

Figure 98 presents results from the developed anomalies detector of every model. Besides, Table 20 provides further information about them.

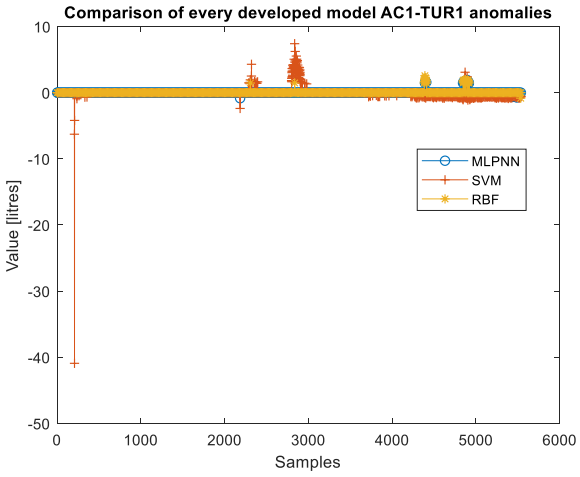


Figure 98. AC1-TUR anomalies detected by MLPNN, SVM and RBF models.

Both Figure 98 and Table 24 shows that best results are obtained by using MLPNN, since it has few small errors. RBF is the method which strongly supports MLPNN estimation due to its successful result. Finally, SVM seems to be the model with most anomalies. However, they are small errors, so it is also compatible with others predictions.

	MLPNN	*SVM*	*RBF*
Number of anomalies [%]	1.2%	33.8%	1.75%
Mean of the anomalies (litres)	1.5 litres	0.93 litres	1.74 litres

Table 24. Comparison of AC1-TUR anomalies detector result.

B.3.3. HEALTH INDICATORS

Both health indicators are depicted in Figure 99. They are reasonable according what has been concluded in terms of AC1-TUR, since SVM's health indicators are those which more fluctuations present. This is translated into the fact that SVM predicts more anomalies. RBF's health indicators also have more and higher rises and falls than MLPNN, which is

normal since this latter one is the best method to estimate this variable, in line with others sections.

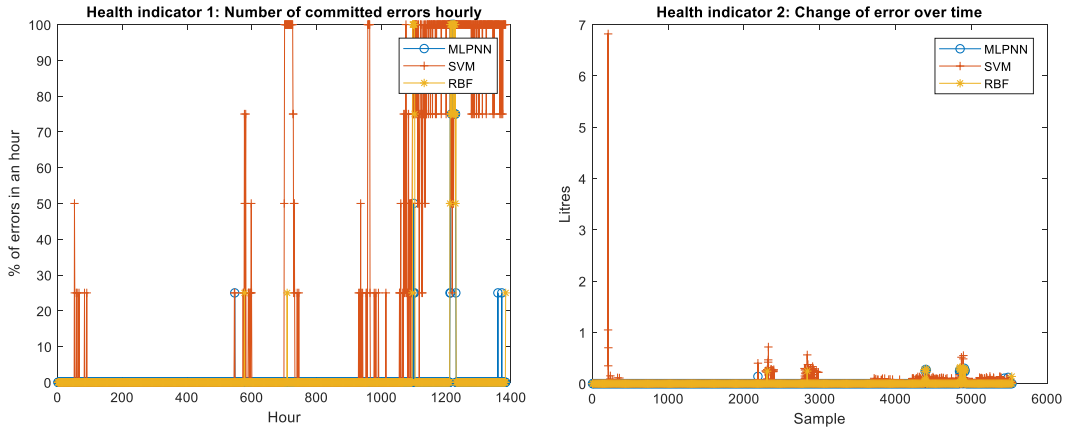


Figure 99. AC1-TUR health indicators for MLPNN, SVM and RBF models.

B.4. AC2-TUR

B.4.1. NORMAL BEHAVIOR MODELS

AC2-TUR normal behavior prediction has been carried out with the values from Table 25, which are sorted by MLPNN, SVM and RBF methods.

2. MLPNN parameters		
	Parameter Name	**Value**
MLPNN	Training Ratio	80%
	Validation Ratio	10%
	Testing Ratio	10%
	Maximum number of iterations	10000
	Learning rate	0.001
	Observing the performance every	50
	Minimum performance	1e-5
	Maximum fail	30

	Number of neurons	8
	SVM parameters	
	Parameter Name	**Value**
SVM	Re-training data	Standardized
	Solver	Sequential Minimal Operation (SMO)
	KernelScale	Auto
	RBF parameters	
	Parameter Name	**Value**
RBF	Spread	45

Table 25. AC2-TUR MLPNN, SVM and RBM models' parameters.

Figure 100 depicts results obtained. For every method predictions are acceptable because errors' histograms present that MLPNN error is not greater than 0.5, and RBF's one doesn't surpass the 0.5%. RBF model is the method which commits several small errors in the estimation. Parallelly, SVM model has some punctual errors which make histogram not be centered in zero. However, it presents the most important part of the bell-shape close to this value. Hence, despite it has calculated wrong a small number of samples, most are similar or equal to real values.

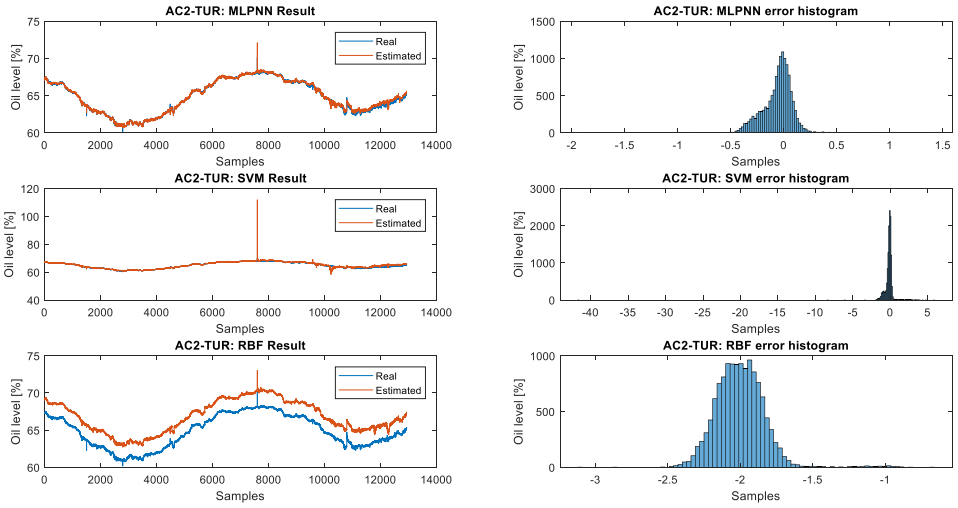


Figure 100. Comparison of AC2-TUR estimation and committed error of developed models.

B.4.2. ANOMALIES DETECTION

Anomalies detector results are in line with previous comments since SVM model is the method with higher anomalies' values (Figure 101).

On the other hand, MLPNN anomalies are few and far between and so small that are hidden by SVM and RBF's anomalies curve. Finally, RBF's anomalies are usual and constant, as it was inferred from Figure 96.

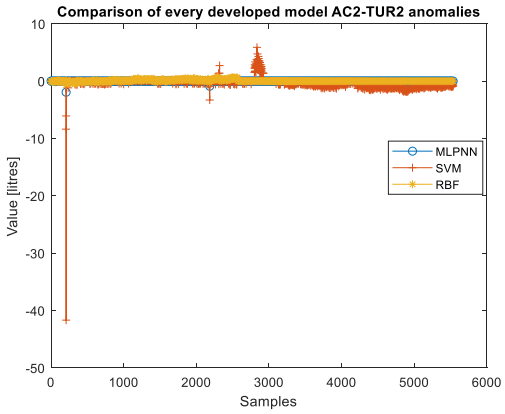


Figure 101. AC2-TUR anomalies detected by MLPNN, SVM and RBF models.

Table 26 approves previous conclusions:

	MLPNN	*SVM*	*RBF*
Number of anomalies [%]	0.397%	40%	46%
Mean of the anomalies (litres)	0.574 litres	0.95 litres	0.182 litres

Table 26. Comparison of AC2-TUR anomalies detector result.

To sum-up, considering that 3 litres is 1% of the accumulator's capacity, AC3-GV have been successfully calculated.

B.4.3. HEALTH INDICATORS

In terms of health indicators, whose outputs are depicted in Figure 102, follow normal behavior pattern and anomalies detection's results.

In this sense, orange is the color which is most outstand in graphs because it is corresponding to SVM method and it is the one which most errors commits. Hence, it was expected to suffer from several fluctuations in health indicators.

MLPNN has small constant fluctuations which explain the few number of anomalies the prediction presents.

Finally, RBF has irregular fluctuations because its prediction has a large deviation. However, in the second health indicator anomalies seem to be soft since yellow curve doesn't reach high quantity of litres.

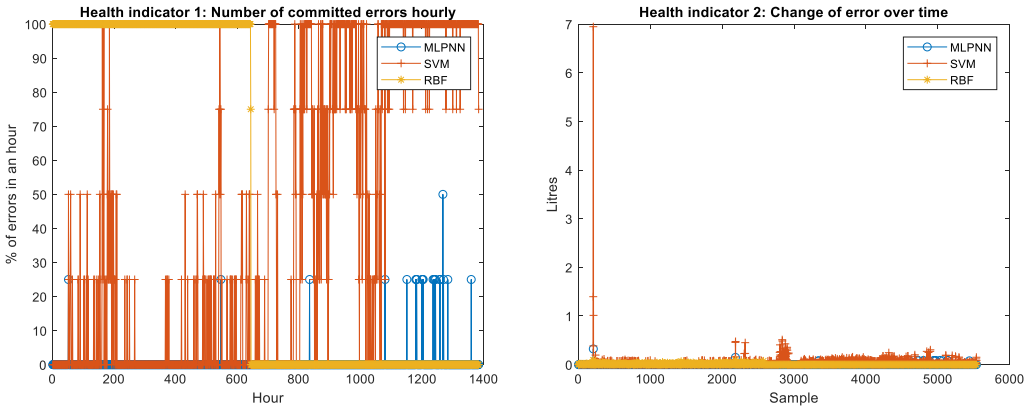


Figure 102. AC2-TUR health indicators for MLPNN, SVM and RBF models.

B.5. AC-TANK WITH REAL DATA

This section explains how the hub tank oil based on real data from other attributes as inputs. Once again, same procedure which has been followed to estimate others variables has been carried out.

B.5. AC-TANK WITH REAL DATA

B.5.1. NORMAL BEHAVIOR MODELS

Best results have been achieved when MLPNN, SVM and RBF methods are configured as Table 23 indicates:

3. MLPNN parameters		
	Parameter Name	**Value**
MLPNN	Training Ratio	80%
	Validation Ratio	10%
	Testing Ratio	10%
	Maximum number of iterations	10000
	Learning rate	0.001
	Observing the performance every	50
	Minimum performance	1e-5
	Maximum fail	30
	Number of neurons	8
	SVM parameters	
	Parameter Name	**Value**
SVM	Re-training data	Standardized
	Solver	Sequential Minimal Operation (SMO)
	KernelScale	Auto
	RBF parameters	
	Parameter Name	**Value**
RBF	Spread	45

Table 27. AC-TANK with real data as inputs MLPNN, SVM and RBM models' parameters.

According to Figure 103, estimations are not as accurate as the previous ones, probably because of its further complexity. Nonetheless error histograms show they are not as different from real values, since they are centered in zero. Besides the maximum error MLPNN, SVM and RBF commit are 0.5, 1.8 and 2% respectively in most samples. Only some punctual ones from SVM and RBF models exceed these numbers. This can be also observed in Figure 104.

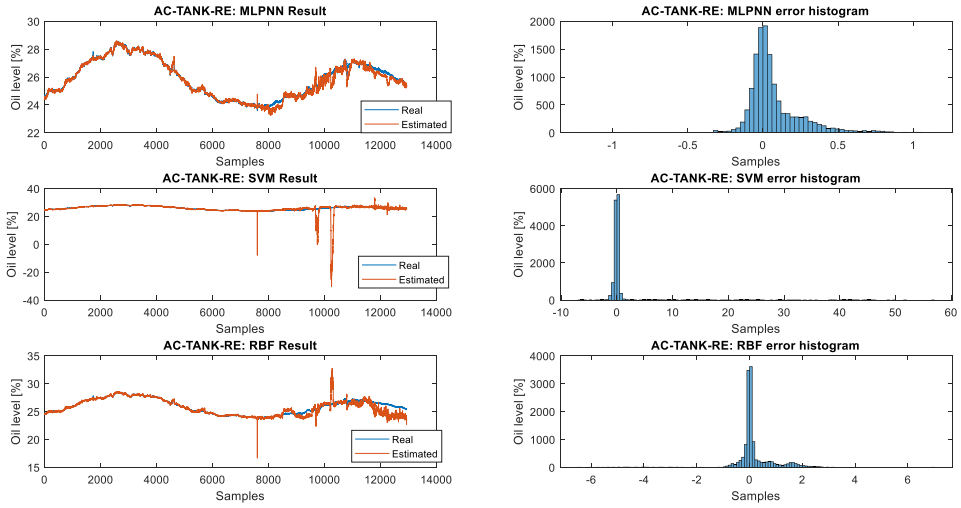


Figure 103. Comparison of AC-TANK with real data as inputs result sorted by developed models.

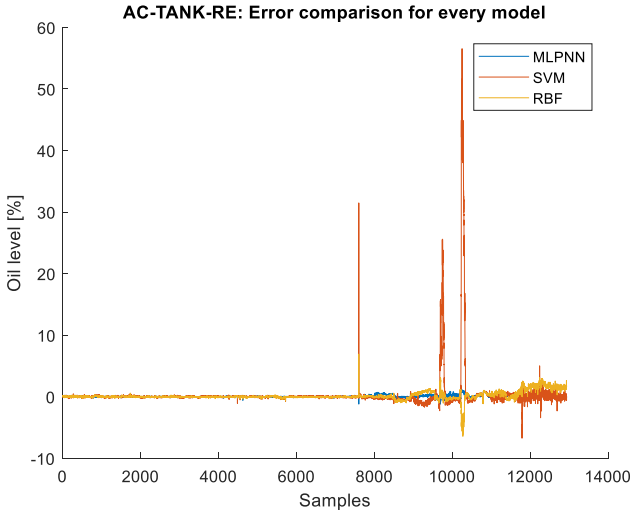


Figure 104. Comparison of AC-TANK with real data as inputs committed error of developed models.

B.5.2. ANOMALIES DETECTION

Table 28 reassert the fact that estimating hub tank oil level is not as successful as others, since it presents a higher percentage of anomalies in the variables. However MLPNN is still

valid for this attribute prediction since it seems estimation are not so different from reality, according mean value of the anomalies. Same happens for RBF, which supports MLPNN results. SVM has worse results and more anomalies consequently. There it is not as appropriate as RBG to strengthen MLPNN results.

	MLPNN	*SVM*	*RBF*
Number of anomalies [%]	13%	18.2%	40.4%
Mean of the anomalies (litres)	0.6 litres	7.1 litres	1.43 litres

Table 28. Comparison of AC-TANK with real data as inpus' anomalies detector result.

Anomalies are better visualized in Figure 105. It confirms what has been commented before. Besides it is observed that the further the sample is from the training set in time, the bigger the error is.

As a result, considering testing sets that have been studied, estimations are acceptable but they need to be tried with more data to ensure if models can be implemented from now on in the plant station.

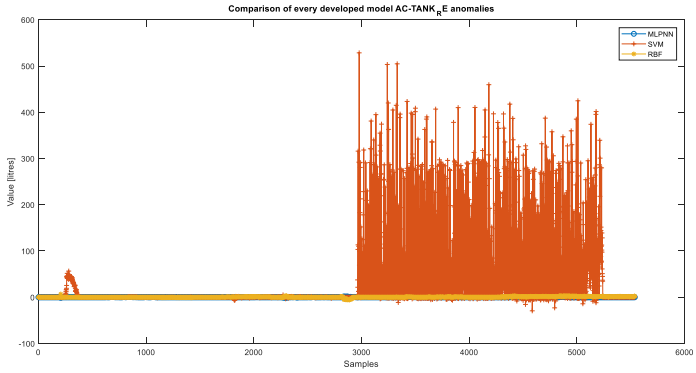


Figure 105. AC-TANK with real data as inputs anomalies detected by MLPNN, SVM and RBF models.

B.5.3. HEALTH INDICATORS

Health indicators are compatible with previous conclusions since most samples show that they suffer from a completely abnormal behavior (First health indicator on the graph of the left). Moreover, from sample 3000, when second testing period starts, error is steeply increasing.

Then, though at the beginning of the testing set, differences between estimations and real values are not big, more samples are needed in order to conclude if the developed models are right.

However it may be also possible that anomalies are right, since the hub tank oil is affected by AC3-GV oil leakages, which start at the beginning of testing 2 too.

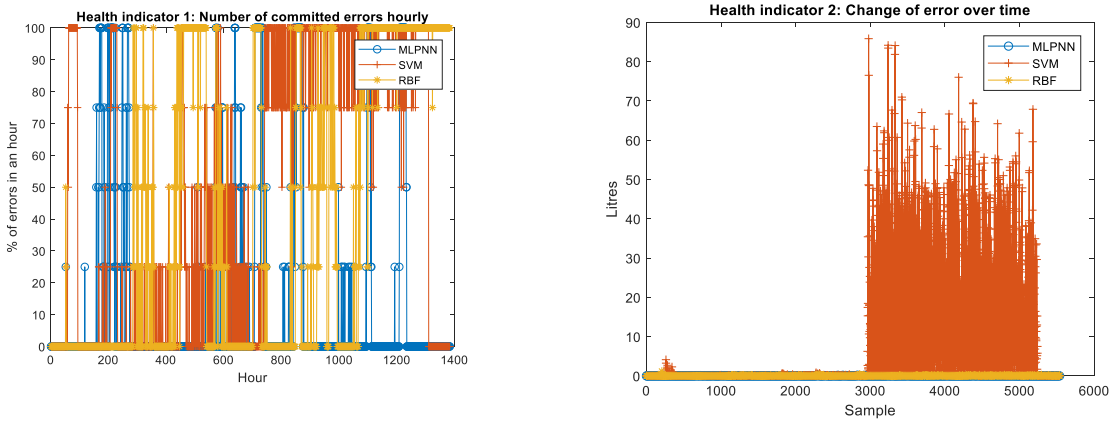


Figure 106. AC-TANK with real data health indicators for MLPNN, SVM and RBF models.

B.6. AC-TANK WITH ESTIMATED DATA

B.6.1. NORMAL BEHAVIOR MODELS

MLPNN parameters		
	Parameter Name	**Value**
MLPNN	Training Ratio	75%
	Validation Ratio	10%
	Testing Ratio	15%
	Maximum number of iterations	10000
	Learning rate	0.01
	Observing the performance every	50
	Minimum performance	1e-5
	Maximum fail	40
	Number of neurons	24
	SVM parameters	
	Parameter Name	**Value**
SVM	Re-training data	Standardized
	Solver	Sequential Minimal Operation (SMO)
	KernelScale	Auto
	RBF parameters	
	Parameter Name	**Value**
RBF	Spread	25

Table 29. AC1-TANK (based on others forecasts) MLPNN, SVM and RBM models' parameters.

As expected, results are worse than previous explained models. Nonetheless forecasts of the AC-TANK normal behavior obtained by using MLPNN method is not very inaccurate because its maximum error is 0.8%.

SVM predicts some outliers which makes the graph to look like it has a lot of predicted zeros. These samples are the ones which makes error histogram tail so long.

RBF results are neither as good as MPLNN nor as bad as SVM's ones. Therefore, more real data is needed in order to infer consistent conclusions.

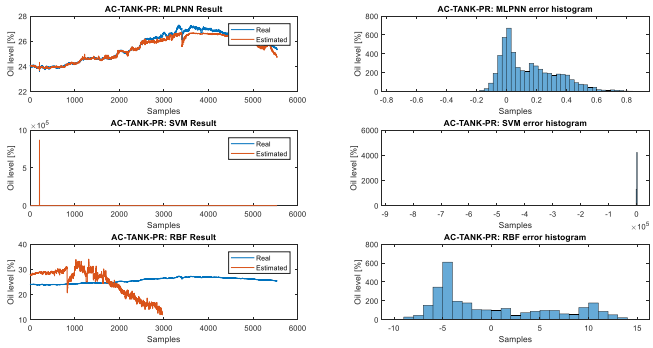


Figure 107. Comparison of AC-TANK (based on others forecasts) prediction of developed models.

B.6.2. ANOMALIES DETECTION

Figure 108 presents anomalies detector results for every method that has been employed. They are reasonable, considering outcome from normal behavior patterns. In this sense, MLPNN shows usual little anomalies. Parallelly SVM seems to not have any anomaly until the end of the set but this is due to the fact that last anomalies are related to such big errors. Finally, RBF presents similar results to MLPNN at he beginning of the testing period. Nevertheless anomalies start to increase at the middle.

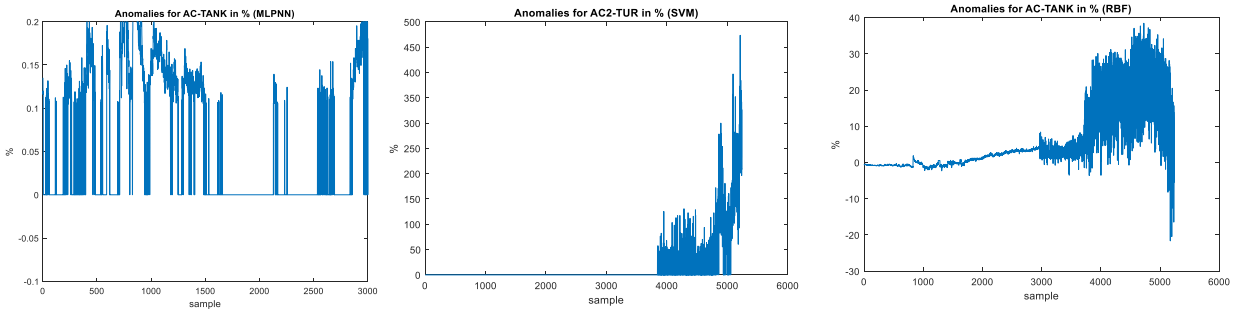


Figure 108. AC-TANK with estimations as inputs anomalies detected by MLPNN, SVM and RBF models.

As a result, it is strengthen the conclusion which was stated about the need of more data to gain consistency and accuracy.

B.6.3. HEALTH INDICATORS

Finally, health indicators results are depicted in Figure 109, sorted by MPLNN, SVM and RBF methods.

Once again, it can be inferred that MLPNN estimation is acceptable, and so it is RBF's prediction until the middle of the testing set, since even though there are fluctuations in the first health indicator, they are small in the second one. Such fluctuations are identified with small frequent errors or estimated anomalies. On the other hand, according to SVM's prediction, AC-TANK is suffering from severe anomalies since fluctuations of second health indicator reach 150 litres. However, this is not supported by the others models.

Consequently, more data is required to reach a clear conclusion and infer if models have succeeded in their forecasts or not.

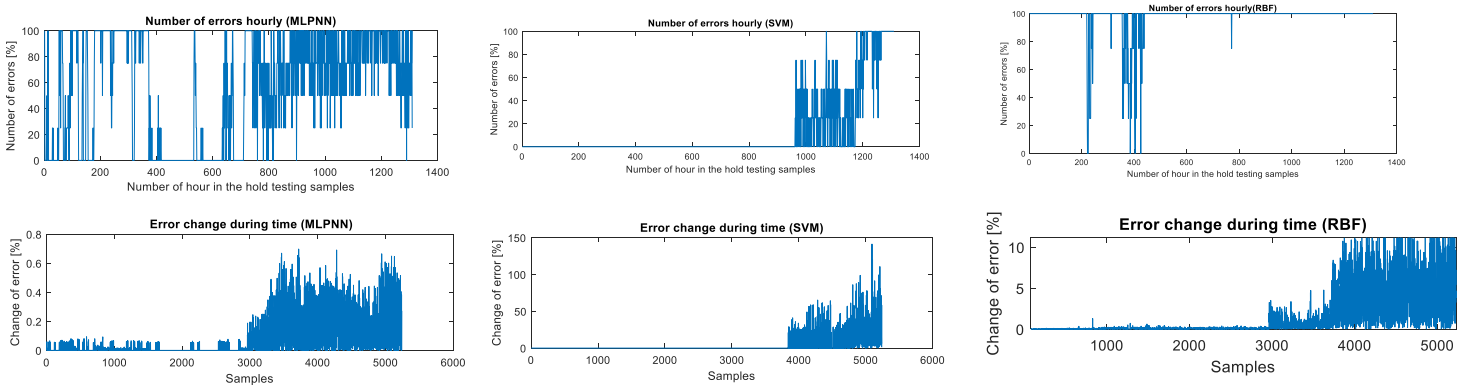


Figure 109. AC-TANK (based on forecasts) health indicators for MLPNN, SVM and RBF models.

[i] SMO: Considering a binary classification problem with a dataset (x_1, y_1), ..., (x_n, y_n), where x_i is an input vector and $y_i \in \{-1, +1\}$ is a binary label corresponding to it. A soft-margin support vector machine is trained by solving a quadratic programming problem, which is expressed in the dual form as follows:

$\max \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i y_j K(x_i, x_j) \alpha_i \alpha_j$, subject to $0 \leq \alpha_i \leq C, for\ i = 1, 2, \ldots, n;\ \sum_{i=1}^{n} y_i \alpha_i = 0$ [13].