



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

PREDICCIÓN DEL NIVEL DE AGUA DEL EMBALSE, BASADO EN REDES NEURONALES, PARA LA MEJORA DE LA PLANIFICACIÓN DE PRODUCCIÓN DE ENERGÍA EN LA CENTRAL HIDROELÉCTRICA AGOYÁN.

GABRIEL ESTEBAN ASQUI SANTILLÁN

Trabajo de Titulación modalidad Proyectos de Investigación y Desarrollo, presentado ante el Instituto de Posgrado y Educación Continua de la ESPOCH, como requisito parcial para la obtención del grado de:

**MAGISTER EN SISTEMAS DE CONTROL Y AUTOMATIZACIÓN
INDUSTRIAL**

Riobamba - Ecuador

Marzo - 2017

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

CERTIFICACIÓN:

EL TRIBUNAL DEL TRABAJO DE TITULACIÓN CERTIFICA QUE:

El Trabajo de Titulación modalidad Proyectos de Investigación y Desarrollo, denominado: "PREDICCIÓN DEL NIVEL DE AGUA DEL EMBALSE, BASADO EN REDES NEURONALES, PARA LA MEJORA DE LA PLANIFICACIÓN DE PRODUCCIÓN DE ENERGÍA EN LA CENTRAL HIDROELÉCTRICA AGOYÁN", de responsabilidad del señor Gabriel Esteban Asqui Santillán, ha sido minuciosamente revisado y se autoriza su presentación.

Ing. Henry Vallejo Vizhuete; MSc.

PRESIDENTE

Ing. Jorge Hernández Ambato; PhD.

DIRECTOR DE TESIS

Ing. Alberto Arellano Aucancela; MSc.

MIEMBRO DEL TRIBUNAL

Ing. Fernando Mejía Peñafiel; MSc.

MIEMBRO DEL TRIBUNAL

DOCUMENTALISTA

SISBIB ESPOCH

Riobamba, Marzo-2017

DERECHOS INTELECTUALES

Yo, Gabriel Esteban Asqui Santillán soy responsable de las ideas, doctrinas y resultados expuestos en este Trabajo de Titulación y el patrimonio intelectual del mismo pertenece a la Escuela Superior Politécnica de Chimborazo.

GABRIEL ESTEBAN ASQUI SANTILLÁN.

CI. 060358715-5

©2016, Gabriel Esteban Asqui Santillán

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor

Yo, Gabriel Esteban Asqui Santillán, declaro que el presente proyecto de investigación, es de mi autoría y que los resultados del mismo son auténticos y originales. Los textos constantes en el documento que provienen de otras fuentes están debidamente citados y referenciados.

Como autor, asumo la responsabilidad legal y académica de los contenidos de este Trabajo de Titulación de Maestría.

GABRIEL ESTEBAN ASQUI SANTILLÁN.

CI. 060358715-5

DEDICATORIA

A Juan, Susana, Nadya, Andrés y Samuel

AGRADECIMIENTO

A la unidad de negocio CELEC EP HIDROAGOYÁN, en especial al Ing. Rafael Rodríguez gerente de producción y al Ing. Santiago Cunalata del departamento de ingeniería, por la apertura y diligencia prestada para la realización del presente tema de tesis.

A la comunidad Python[®], al igual que a mis tutores como a todos los gestores del software libre.

A mi familia por el apoyo y auspicio.

ÍNDICE

CERTIFICACIÓN	II
DERECHOS INTELECTUALES	III
AUTORIZACIÓN	IV
DECLARACIÓN DE AUTENTICIDAD	V
DEDICATORIA	VI
AGRADECIMIENTO	VII
RESUMEN	XV
SUMMARY	XVII
CAPITULO I	
1 INTRODUCCIÓN.	1
1.1 Planteamiento del problema	1
1.2 Formulación del problema.	2
1.3 Preguntas directrices.	2
1.4 Justificación.	2
1.5 Objetivos de la Investigación.	3
1.5.1 <i>Objetivo General.</i>	3
1.5.2 <i>Objetivos Específicos.</i>	3
1.6 Hipótesis.	4
1.6.1 <i>Hipótesis General.</i>	4
1.6.2 <i>Hipótesis Específicas.</i>	4
1.6.3 <i>Identificación de Variables.</i>	4

1.6.4	<i>Operacionalización de Variables.</i>	5
-------	---	---

CAPITULO II

2	MARCO TEÓRICO.	7
2.1	Estado del arte.	7
2.2	Redes Neuronales Artificiales.	9
2.2.1	<i>Modelos computacionales.</i>	11
2.3	Funciones de Activación.	12
2.3.1	<i>Función Escalón.</i>	13
2.3.2	<i>Función Lineal y Mixta.</i>	13
2.3.3	<i>Función Tangente Hiperbólica.</i>	15
2.3.4	<i>Función Sigmoidea,</i>	15
2.4	Tipos de entrenamiento.	16
2.5	Algoritmos de Entrenamiento.	16
2.5.1	<i>La Función de Error.</i>	16
2.5.2	<i>Algoritmo Back-propagation.</i>	19
2.5.3	<i>Algoritmo Broyden Fletcher Goldfarb Shanno (BFGS).</i>	20
2.6	RNA como Predictor.	26
2.6.1	<i>Predicción en un Paso de Tiempo.</i>	26
2.6.2	<i>Predicción en Múltiples Pasos de Tiempo.</i>	27
2.7	RNA como Modelo.	28

CAPITULO III

3	METODOLOGÍA DE LA INVESTIGACIÓN.	34
3.1	Preparación y Análisis de Datos.	34
3.2	Determinación de los Modelos.	41
3.2.1	<i>Predictores.</i>	44
3.2.2	<i>Modelo de la Presa.</i>	50
3.3	Implementación de la Plataforma.	55
3.3.1	<i>Módulo Base de Datos.</i>	59
3.3.2	<i>Módulo Frontal.</i>	60
3.3.3	<i>Módulo IHM.</i>	61

3.3.4	<i>Módulo Modelo.</i>	62
3.3.5	<i>Módulo Predictor.</i>	62
3.3.6	<i>Módulo Usuarios.</i>	62

CAPITULO IV

4	RESULTADOS.	63
4.1	Resultados y Discusión.	63
4.1.1	<i>Docimasia de hipótesis</i>	67

	CONCLUSIONES	73
--	---------------------	-----------

	RECOMENDACIONES	74
--	------------------------	-----------

BIBLIOGRAFÍA

ANEXOS

LISTA DE TABLAS

Tabla 1-1	Identificación de variables.	5
Tabla 2-1	Operacionalización de variables.	6
Tabla 1-3	Pruebas de configuración predictor de temporada invernal fuerte.	46
Tabla 2-3	RMSE promedio de los predictores.	51
Tabla 3-3	RMSE promedio de los modelos.	55
Tabla 4-3	RMSE promedio del modelo de la presa sumado al predictor.	58
Tabla 1-4	RMSE y porcentaje de acierto del predictor de caudal.	64
Tabla 2-4	RMSE y porcentaje de acierto del predictor de nivel.	67
Tabla 3-4	Reprogramaciones y cambios de producción antes y después.	71

LISTA DE FIGURAS

Figura 1-2	Neurona artificial.	10
Figura 2-2	Esquema de una RNA.	11
Figura 3-2	Función escalón.	14
Figura 4-2	Función lineal y mixta.	14
Figura 5-2	Función tangente hiperbólica.	15
Figura 6-2	Función sigmoidea.	15
Figura 7-2	Clasificación de los métodos numéricos llamados de descenso.	18
Figura 8-2	RNA simple de ejemplo.	18
Figura 9-2	Esquema iterativo de minimización utilizando el método de Newton.	22
Figura 10-2	Esquema iterativo método BFGS.	23
Figura 11-2	Elección de paso de Goldstein.	25
Figura 12-2	Esquema iterativo para la elección del paso.	30
Figura 13-2	Esquemas de predicción en un paso	31
Figura 14-2	División de datos históricos.	31
Figura 15-2	Predictor varios pasos.	32
Figura 16-2	Ejemplos predicción varios pasos.	32
Figura 17-2	Diagrama de entrenamiento.	33
Figura 18-2	Diagrama de modelado.	33
Figura 1-3	Asignación de temporadas para el caudal del Pastaza	36
Figura 2-3	Diagrama de bloques del sistema.	41
Figura 3-3	Pantallas de los entrenadores de predictores.	46
Figura 4-3	Ventana de pruebas para el predictor.	50
Figura 5-3	Entrenador de Modelos	52
Figura 6-3	Ventana de pruebas para el modelo de la presa.	53
Figura 7-3	Ventana de pruebas del modelo del nivel de la presa más el predictor.	59
Figura 8-3	Ventana para subir los datos a la BDD.	60

Figura 9-3	Ventana principal frontal con las distintas lengüetas.	61
Figura 1-4	Predicción de caudal con umbral de ocho horas.	66
Figura 2-4	Predicción de nivel con umbral de ocho horas.	69

LISTA DE GRÁFICOS

Gráfico 1-2	Predicción para la cuenca del Tomebamba en Matadero	8
Gráfico 2-2	Comparación de caudales en el río Tomebamba	8
Gráfico 3-2	Caudal real frente a los predichos en el río Pastaza	9
Gráfico 1-3	Caudal rio Pastaza	36
Gráfico 2-3	Caudal temporadas veraniegas 2012-2014	38
Gráfico 3-3	Picos de PSD temporadas veraniegas	39
Gráfico 4-3	Cambios de producción y reprogramaciones por temporada	40
Gráfico 5-3	Análisis Frecuencial año 2005	42
Gráfico 6-3	Dispersión de frecuencias	43
Gráfico 7-3	Errores de entrenamiento y validación	47
Gráfico 8-3	RMSE de los predictores por temporadas usando el método 1	48
Gráfico 9-3	RMSE de los predictores por temporadas usando el método 2	49
Gráfico 10-3	RMSE de los modelos de la presa según las temporadas	54
Gráfico 11-3	RMSE del modelo de la presa sumado al predictor con método 1	56
Gráfico 12-3	RMSE del modelo de la presa sumado al predictor con método 2	57
Gráfico 1-4	RMSE diario del predictor de caudal	65
Gráfico 2-4	RMSE diario del predictor de nivel	68

RESUMEN

El objetivo de la investigación fue predecir el caudal futuro del río Pastaza, para determinar el nivel de embalse de la presa, a fin de planificar la producción de energía eléctrica en la central hidroeléctrica Agoyán. Se utilizó los datos históricos de caudal, nivel y potencia activa; comprendidos desde el año 2005 hasta 2016. Se subió los datos al gestor de base de datos PostgreSQL[®]. Se escogió una muestra no probabilista a la que se aplicó un filtrado de datos, una división por temporadas climáticas y una transformación logarítmica. Se creó una plataforma de software con el lenguaje Python[®], que permitió el entrenamiento de redes neuronales artificiales basados en la librería NeuroLab[®] y el algoritmo Broyden–Fletcher–Goldfarb–Shanno (BFGS). Se implementó dentro de la plataforma los algoritmos de predicción de caudal y modelado de la presa, así como las herramientas para la comprobación de efectividad de los modelos. Se sometió a prueba el sistema durante veinte y dos (22) días obteniéndose un RMSE promedio de 9.96 metros cúbicos por segundo, con un porcentaje de aciertos de 45.45 %, para el predictor de caudal y un RMSE promedio de 0.2735 metros sobre el nivel del mar con un porcentaje de aciertos de 86.36 %; en un umbral de ocho (8) horas de predicción. Se concluyó que el umbral de predicción alcanzado no afecta aún los índices de producción, por lo que se recomienda la implementación de estaciones meteorológicas en las cuencas de los ríos Chambo y Patate.

Palabras clave: <TECNOLOGÍA Y CIENCIAS DE LA INGENIERÍA>, <CONTROL AUTOMÁTICO>, <SERIES TEMPORALES>, <REDES NEURONALES ARTIFICIALES (RNA)>, <PREDICTOR>, <MODELADO>, <PYTHON (SOFTWARE)>, <NEUROLAB (SOFTWARE)>, <CENTRAL HIDROELÉCTRICA>

ABSTRACT

The objective of this research was to predict the future water flow of the Pastaza River in order to determine the reservoir level of the dam, for the purpose of planning the production of electrical energy at the Agoyán hydroelectric power plant. Historical data of the water flow, its level and active power was used ranging from the year 2005 until 2016. The records were uploaded to the database manager PostgreSQL[®]. A non-probabilistic sample was chosen to which a data filter was applied, a division by climatic seasons and a logarithmic transformation. A software platform was created using the Python[®] language, which allowed for the training of artificial neural networks based on the NeuroLab[®] library and the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm for flow prediction and dam modeling, as well as the tools for the testing of effectiveness of the models. The system was tested for twenty-two (22) days obtaining a RMSE average of 9.96 cubic meters per second, with an accuracy percentage of 45.45% for the flow predictor and a RMSE average of 0.2735 meters above the sea level with an accuracy percentage of 86.36%; in a threshold of eight (8) hours of prediction. It was concluded that the forecast threshold reached doesn't yet affect the production indices, for which the implementation of weather stations at the basins of the Chambo and Patate rivers is recommended.

Keywords: <ENGINEERING SCIENCES AND TECHNOLOGY>, <AUTOMATIC CONTROL>, <TEMPORAL SERIES>, <ARTIFICIAL NEURAL NETWORK (ANN)>, <PREDICTOR>, <MODELING>, <PYTHON (SOFTWARE)>, <NEUROLAB (SOFTWARE)>, <HYDROELECTRIC PLANT>

CAPÍTULO I

1 . INTRODUCCIÓN.

El presente capítulo explica la problemática encontrada en la central Hidroeléctrica Agoyán, sobre la planificación de producción de energía eléctrica, planteando variables, indicadores e hipótesis, que determinen si la solución planteada incide favorablemente o no en la problemática descrita.

1.1 Planteamiento del problema

La Empresa Pública Corporación Eléctrica del Ecuador (CELEC EP), dentro de los objetivos estratégicos, dispone el incrementar la disponibilidad y confiabilidad del sistema nacional eléctrico interconectado, así como el incrementar la oferta de servicio eléctrico, por lo que se encuentra construyendo proyectos como Coca Codo Sinclair, Sopladora, Minas San Francisco, Toachi Pilatón, Delsitanisagua, Manduriacu, Quijos, Mazar-Dudas, Ciclo Combinado y proyectos de transmisión entre ellos el Sistema de Transmisión EAT 500 KV (CELEC EP, 2016).

Es importante mencionar que el beneficio de tales obras es considerable al tener mayor producción energética para el país. Sin embargo, debido a que construir un nuevo proyecto hidroeléctrico recae en grandes costes, se deben también buscar alternativas que permitan optimizar los proyectos hidroeléctricos que ya se encuentran generando.

En este ámbito, el Ecuador cuenta con la Central Hidroeléctrica Agoyán, la misma que está ubicada en la provincia de Tungurahua en el km 5 1/2 de la vía que une la ciudad de Baños de Agua Santa con la ciudad de Puyo, en el sector llamado el Agoyán.

En esta central se realizan dos planificaciones diarias una con umbral de veinte y cuatro horas (24h) y otra con umbral de cuarenta y ocho horas (48h) de producción, usando los datos obtenidos por dos sensores, uno de caudal implantado en el río Pastaza y uno de nivel implantado en la presa. Además, se añade la experiencia del operador, la misma

que no siempre tiene éxito en la toma de decisiones acertadas sobre la estimación de producción energética.

La situación antes mencionada, da lugar a que la producción de energía no siempre sea la óptima, teniendo en ocasiones que desperdiciar agua del embalse por planificar una cota de producción baja respecto a los recursos disponibles y que no fueron correctamente estimados.

Por otro lado, puede darse el caso de bajar súbitamente la producción debido a una sobre estimación de cota de producción alta, resultando en un nivel de embalse inadecuado. Estos escenarios constituyen la problemática, actualmente presente en la Central Hidroeléctrica Agoyán, sobre la cual se desarrolla el presente tema de investigación.

1.2 Formulación del problema.

¿La correcta predicción del nivel de agua del embalse basado en redes neuronales, mejorará la planificación de producción de energía en la Central Hidroeléctrica Agoyán?

1.3 Preguntas directrices.

- ¿Qué técnica de predicción basada en red perceptrón multicapa resultó ser la más adecuada?
- ¿Cuál es el algoritmo para definir los parámetros del perceptrón multicapa para la predicción?
- ¿Cuál es el modelo que debe implantarse para la predicción de caudal y nivel del embalse de agua?
- ¿Qué plataforma hardware - software es necesaria para la evaluación del modelo predictivo?
- ¿Qué nivel de exactitud se obtuvo en la predicción del nivel de agua del embalse?

1.4 Justificación.

La producción de energías limpias es una necesidad primordial de la sociedad moderna (REN21, 2015). Una de estas formas de producción es la hidroeléctrica. Sin embargo, la

generación de electricidad en forma limpia no siempre significa un óptimo aprovechamiento del recurso utilizado, que para este caso específico es el agua.

Un ejemplo de producción de energía eléctrica limpia; en el cual se busca la eficiente utilización del recurso natural que la genera, es la solar. Esto se logra con la aplicación de algoritmos de Rastreo del Máximo Punto de Potencia (MPPT, por sus siglas en inglés), logrando transferir la máxima energía generada por los paneles solares hacia la carga (Barros & Mota, 2010).

En otro ámbito la generación mundial de energía hidráulica en 2014 se ha estimado en 3.900 TWh. De igual modo, la industria continuó buscando la innovación en servicios más confiables, eficientes y flexibles (REN21, 2015).

Tomando en cuenta esta creciente necesidad de energía, los objetivos nacionales del buen vivir recalcan en su objetivo número siete el promover la eficiencia y una mayor participación de energías renovables sostenibles como medida de prevención de la contaminación ambiental (Senplades, 2013).

Por otro lado, el objetivo once del plan nacional del buen vivir busca asegurar la soberanía y eficiencia de los sectores estratégicos para la transformación industrial y tecnológica (Senplades, 2013).

Procurando estos objetivos, cualquier esfuerzo investigativo, que mejore el aprovechamiento del agua para la producción de energía eléctrica, como el aquí propuesto, es de suma importancia para dar cumplimiento al plan nacional del buen vivir.

1.5 Objetivos de la Investigación.

1.5.1 *Objetivo General.*

- Predecir el nivel de agua del embalse para la mejora de la planificación de producción de energía en la Central Hidroeléctrica Agoyán.

1.5.2 *Objetivos Específicos.*

- Estudiar el estado del arte de las técnicas de predicción basadas en perceptrón multicapa aplicadas a la predicción de caudal y nivel del embalse de agua.

- Diseñar el algoritmo para definir los parámetros del perceptrón multicapa para la predicción de nivel y caudal.
- Diseñar un modelo predictivo basado en redes neuronales que permita pronosticar el caudal y nivel del embalse de agua.
- Implementar la plataforma hardware - software para la evaluación del modelo.
- Comprobar la efectividad del modelo en la predicción del caudal y del embalse de agua frente a los datos reales e históricos.

1.6 Hipótesis.

1.6.1 *Hipótesis General.*

La adecuada selección de la técnica de predicción basada en red neuronal y aplicada a la predicción del nivel de agua del embalse optimiza la planificación de producción de energía.

1.6.2 *Hipótesis Específicas.*

- El estudio del arte de las técnicas de predicción basadas en red perceptrón multicapa ayuda a seleccionar la metodología adecuada para la predicción de caudal y nivel del embalse de agua.
- El diseño de un modelo predictivo basado en red perceptrón permite pronosticar el caudal y nivel del embalse de agua.
- El modelo es efectivo en la predicción del caudal del río Pastaza y nivel del embalse frente a los datos reales en un 90%.
- El tiempo máximo en el cual el modelo es efectivo es de un día.

1.6.3 *Identificación de Variables.*

Se identifican cuatro variables descritas en la Tabla 1-1.

Tabla 1-1: Identificación de variables.

VARIABLE	TIPO	CONCEPTO
Reprogramaciones semanales de producción.	Cuantitativa. Dependiente.	Cantidad de veces en que la producción se tuvo que cambiar en la semana.
Selección adecuada del tipo de RNA para la predicción del nivel de agua del embalse en la Central Hidroeléctrica Agoyán.	Cualitativa. Independiente.	Es el conjunto de herramientas existentes para llegar a un modelo de caja negra.
El grado de precisión en la predicción del nivel de agua.	Cuantitativa. Dependiente.	Es el porcentaje de aciertos y fallos en la predicción del nivel de agua.
Tiempo máximo en que el modelo se acercó a la realidad.	Cuantitativa.	Cantidad de horas en que el modelo se acercó a la realidad medido en horas, días, semanas, etc.

Realizado por: Gabriel Asqui. 2016

1.6.4 Operacionalización de Variables.

La Tabla 2-1 muestra los indicadores, las técnicas y los instrumentos de cada variable.

Tabla 2-1: Operacionalización de variables.

VARIABLE	INDICADOR	TÉCNICA	INSTRUMENTO FUENTE
Reprogramaciones semanales de producción.	Cantidad de reprogramaciones semanales de producción.	Observación.	Bases de Datos.
La selección adecuada del tipo de RNA para la predicción de nivel de agua del embalse en la Central Hidroeléctrica Agoyán.	Tipos de RNA utilizadas para predicción.	Revisión bibliográfica Prueba y error.	Técnicas de predicción.
El grado de precisión en la predicción del nivel de agua.	% de aciertos. % de fallos.	Observación. Pruebas.	Técnicas Estadísticas.
Tiempo máximo en que el modelo se acercó a la realidad.	Horas, días. % de error.	Observación.	Técnicas Estadísticas.

Realizado por: Gabriel Asqui. 2016

CAPÍTULO II

2. MARCO TEÓRICO.

2.1 Estado del arte.

Los esfuerzos por obtener modelos que expliquen y predigan el comportamiento del caudal de un río dentro del país han sido pocos. Uno de estos casos es el realizado en la cuenca del río Paute por A. Quimbo y R. Vázquez utilizando técnicas de minado de datos como son el Modelo Mecanicista Basado en Datos (DBM) juntamente con el filtro de Kalman. El gráfico 1-2 muestra gráficamente los resultados obtenidos en este estudio. (Quichimbo & Vázquez, 2014).

Por otro lado, J. Veintimilla y F. Cisneros, proponen la utilización de “Artificial Neural Networks” (Redes Neuronales Artificiales, RNA) para la obtención de un modelo que no necesite los rigurosos requerimientos de información de los modelos matemáticos que encuentre una relación entre las entradas y las salidas en una cuenca hidrográfica. Además se hace una comparativa entre distintos algoritmos de entrenamiento como son el Back-Propagation (retro propagación), OWO-HWO y Genéticos, obteniendo los mejores resultados con el algoritmo de Back-Propagation, utilizando los datos de ocho (8) estaciones meteorológicas y prediciendo con los datos de tres (3) días anteriores, datos hasta de un (1) día después, aunque no se especifica el grado de precisión de los mismos ni las métricas que se manejaron para dar un dato como válido (Veintimilla & Cisneros, 2014).

El gráfico 2-2 muestra los resultados de la predicción con los métodos usados por J. Veintimilla y F. Cisneros.

De manera particular, el trabajo de F. Anchaluiza y V. Hinojosa, el mismo que fue basado en datos de caudales de la Central Hidroeléctrica Agoyán, conjuntamente con la aplicación de las técnicas del Razonamiento Inductivo Fuzzy y algoritmos evolutivos, permitió el mejoramiento de la predicción de caudal del río Pastaza en comparación a los métodos

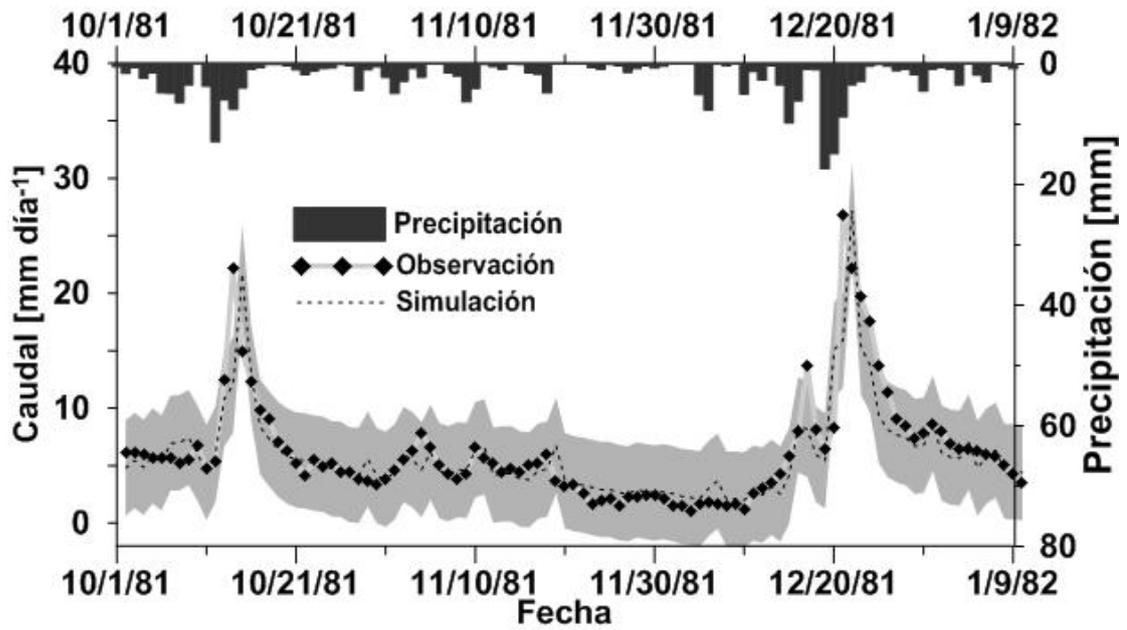


Gráfico 1-2: Predicción para la cuenca del Tomebamba en Matadero.

Fuente: A. Quimbo, R. Vázquez, Predicción de caudales en la cabecera de la cuenca del Paute mediante el modelo DBM.

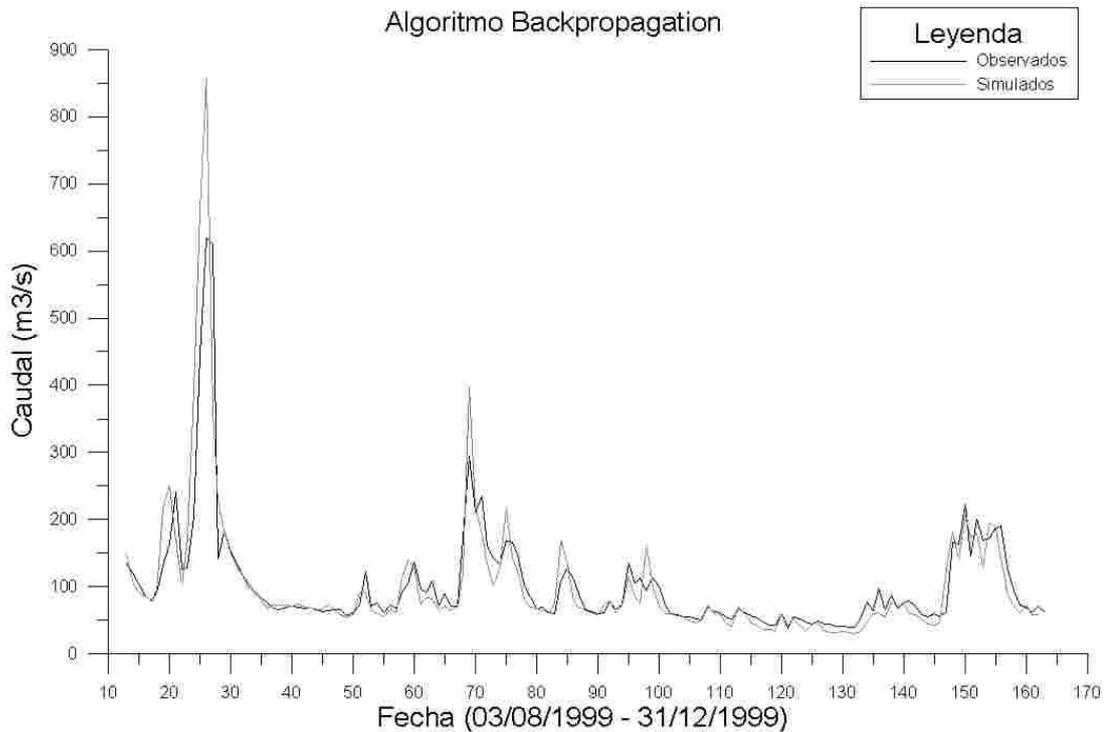


Gráfico 2-2: Comparación de caudales en el río Tomebamba.

Fuente: J. Ventimilla y F. Cisneros, Redes neuronales artificiales (RNA) aplicadas en la predicción de caudales para intervalos de tiempo horarios.

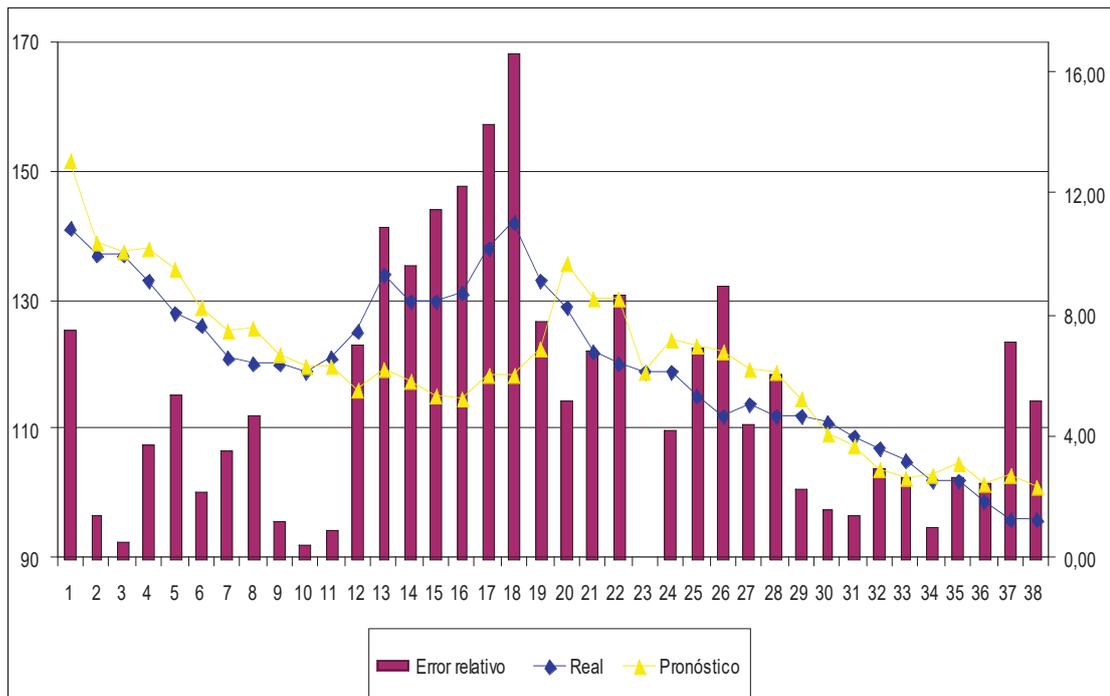


Gráfico 3-2: Caudal real frente a los predichos en el río Pastaza.

Fuente: F. Anchaluja y V. Hinojosa, Pronóstico de caudales de mediano y corto plazo utilizando razonamiento inductivo fuzzy y algoritmos evolutivos-aplicación para las centrales de embalse y centrales de pasada.

estadísticos generalmente utilizados en el sector hidroeléctrico como el ARIMA, en el gráfico 3-2 se aprecian los resultados de este estudio (Anchaluisa & Hinojosa, 2008).

Cabe recalcar que este estudio no implementó ninguna plataforma hardware-software en la empresa para optimizar la producción de energía eléctrica, ni para predecir el caudal o el nivel del embalse que será parte de los objetivos de esta investigación.

2.2 Redes Neuronales Artificiales.

Las Redes Neuronales Artificiales (RNA) son sistemas que constan de un gran número de procesadores simples, llamados neuronas, ligados por conexiones o enlaces los cuales son caracterizados por un peso o valoración (Ponce, 2010).

Cada unidad recibe entradas $x_i, i = 1, \dots, n$, y genera una única salida y_j . Los pesos $w_{ji}, i = 1, \dots, n$, que conectan la entrada i –ésima, con la neurona j –ésima, atenúan o amplifican las señales de entrada, provocando que su suma, más el umbral o sesgo θ_j , asociado a la neurona j –ésima, al pasar por una función de activación, produzca una señal de salida saturada, es decir que a pesar de que el sumatorio sea un número grande positivo o negativo, la salida solo variara dentro de un rango específico (Ponce, 2010; Pérez, 2002).

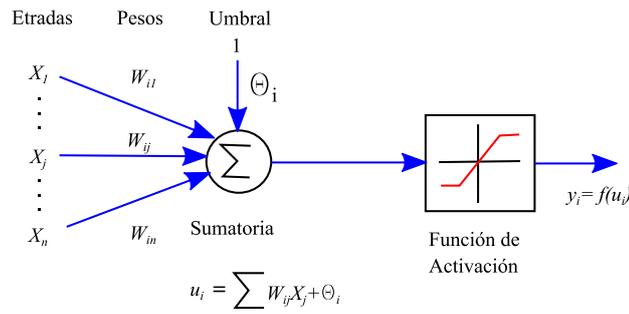


Figura 1-2: Neurona artificial.
Realizado por: Gabriel Asqui. 2016

En la Fig. 1-2 es posible observar el comportamiento antes descrito aplicado a una neurona artificial.

Todas las señales que ingresan a la neurona k pueden ser vistas como una única entrada si se describe utilizando la ecuación 2.1 (Ponce, 2010).

$$u = \sum_{i=1}^n w_{ji}x_i + \theta_j \quad (2.1)$$

Dependiendo del signo del peso se habla de una excitación para el caso positivo y de una inhibición para el caso negativo. Si se añade la variable tiempo a la ecuación 2.1 esta se reescribe como lo muestra la ecuación 2.2 (Ponce, 2010).

$$u(t) = \sum_{i=1}^n w_{ji}x_i(t) + \theta_j \quad (2.2)$$

Los pesos son los encargados de guardar el conocimiento de la red neuronal sobre cierta tarea en cuestión y al proceso mediante el cual se ajustan estos pesos se denomina aprendizaje o entrenamiento. Por lo general el sesgo θ se lo considera como otro peso más conectado a una entrada que siempre tiene el valor de uno es por eso que la salida de cada neurona se la puede reescribir como la ecuación 2.3, por lo que de aquí en adelante solo se lo mencionara o graficará donde sea estrictamente necesario (Ponce, 2010; Pérez, 2002).

$$y = f(u)$$

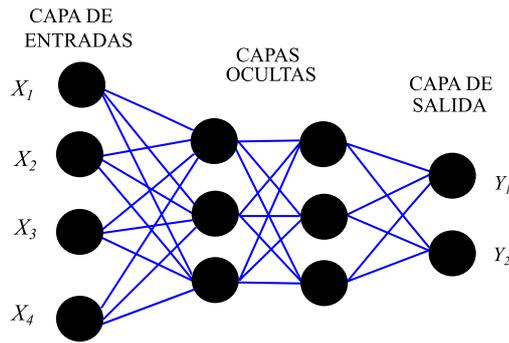


Figura 2-2: Esquema de una RNA.
Realizado por: Gabriel Asqui. 2016

$$y_j(t) = F\left(\sum_{i=1}^n w_{ji}x_i(t) + w_j\right) \quad (2.3)$$

Donde:

$x_i(t)$: Señal de salida de otros nodos o entradas externas.

w_{ji}, w_j : Pesos de las conexiones, sesgo.

$F(*)$: Función de activación no lineal.

La Fig. 2-2. muestra como las RNA poseen una estructura redundante por lo que destacan por su elevada capacidad de producir salidas correctas para entradas no vistas en el entrenamiento (Pérez, 2002).

2.2.1 Modelos computacionales.

Dentro de los modelos computacionales más conocidos de RNA se tiene:

- Perceptrón.- Contiene una única capa de neuronas y una función de activación binaria, basa su aprendizaje en ejemplos y puede realizar tareas de clasificación utilizando la función del umbral para el aprendizaje (Isasi Viñuela & Galván León, 2004).
- Adaline.- Posee una estructura idéntica al perceptrón pero utiliza la regla delta de aprendizaje que compara la salida producida por la red con la salida deseada sin pasar por la función del umbral (Isasi Viñuela & Galván León, 2004).

- **Perceptrón Multicapa.**- Posee varias capas ocultas de neuronas todas ellas conectadas hacia adelante, convirtiéndolo en un aproximador universal. Utiliza funciones de activación no lineales justamente para la resolución de problemas de esta índole, y es el modelo escogido en este trabajo (Isasi Viñuela & Galván León, 2004).
- **Redes Neuronales de base Radial.**- Son redes multicapa con conexión hacia adelante, que contiene una única capa oculta cuyas neuronas activan una distinta zona del espacio de patrones de entrada.
- **Redes Auto organizables.**- Utilizan algoritmos de aprendizaje no supervisado descritos en la sección 2.4, permitiendo le sin recibir información del entorno ajustar sus pesos es decir auto organizarse. (Ponce, 2010)

2.3 Funciones de Activación.

Las funciones de activación $F_k(*)$ establecen el efecto de la entrada total $u(t)$ en la activación de la unidad k . Dichas funciones pueden ser sigmoidea, tangente hiperbólica, escalón entre otras (Ponce, 2010).

Una condición para escoger estas funciones de activación, es que su derivada se puede calcular a partir de la propia función, por ejemplo, si escogemos la función sigmoidea de la ecuación 2.4 y la derivamos aplicando la regla de la cadena obtendremos la ecuación 2.5.

$$f(x) = (1 + e^{-x})^{-1} \quad (2.4)$$

$$\frac{df(x)}{dx} = (1 + e^{-x})^{-2} \cdot e^{-x} \quad (2.5)$$

Podemos notar entonces que si utilizamos las ecuaciones 2.6 2.7 y las sustituimos en 2.5, obtendremos la ecuación 2.8.

$$f(x)^2 = (1 + e^{-x})^{-2} \quad (2.6)$$

$$\frac{df(x)}{dx} = f(x)^2 \cdot e^{-x}$$

Despejando e^{-x} de la ecuación 2.4.

$$e^{-x} = \frac{1}{f(x)} - 1 \quad (2.7)$$

Entonces:

$$\frac{df(x)}{dx} = f(x)^2 \left[\frac{1}{f(x)} - 1 \right]$$

$$\frac{df(x)}{dx} = f(x) [1 - f(x)] \quad (2.8)$$

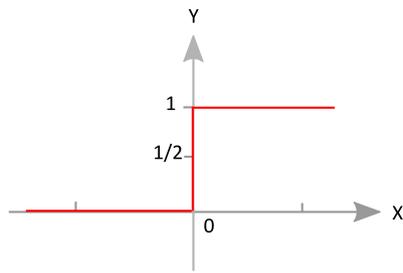
Como se aprecia en la ecuación 2.8, la derivada de una función sigmoidea se puede calcular a partir del valor de la propia función sigmoidea, lo que presenta una gran ventaja, y explica su uso en las RNA. Los siguientes apartados muestran las gráficas y ecuaciones de las funciones de activación más utilizadas.

2.3.1 *Función Escalón.*

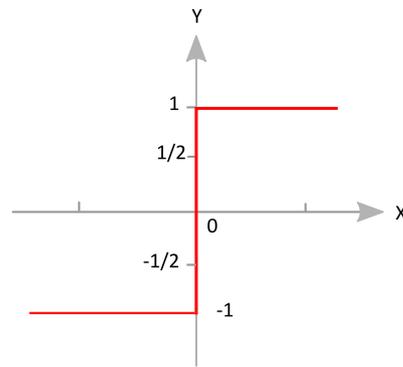
Se asocia a neuronas binarias, cuando la suma es mayor o igual al umbral de la neurona, la salida toma un valor de uno (1) de lo contrario es cero (0) o menos uno (-1) (véase Fig. 3-2) (Ponce, 2010).

2.3.2 *Función Lineal y Mixta.*

Estas funciones, como lo muestra la Fig. 4-2, define un límite inferior y uno superior, si el sumatorio de las entradas son menores al límite inferior la salida es cero (0) o menos uno (-1); en cambio si el sumatorio es superior al límite superior la salida es uno (1); en el caso de que el valor del sumatorio se encuentre entre los dos límites, la salida tomara un valor de acuerdo a la función lineal asociada (Ponce, 2010).



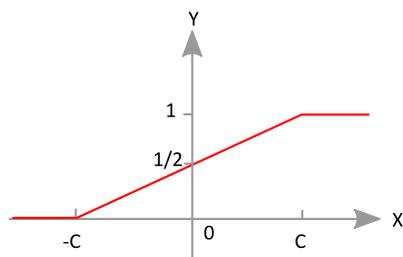
$$F_k(x) = \begin{cases} 1 & \text{si } x \geq 0 \\ 0 & \text{si } x < 0 \end{cases}$$



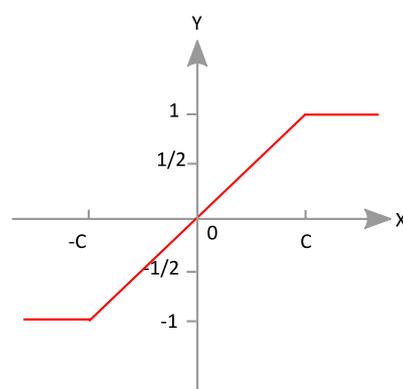
$$F_k(x) = \begin{cases} 1 & \text{si } x \geq 0 \\ -1 & \text{si } x < 0 \end{cases}$$

Figura 3-2: Función escalón.

Fuente: Ponce P. Inteligencia artificial con aplicaciones a la ingeniería.



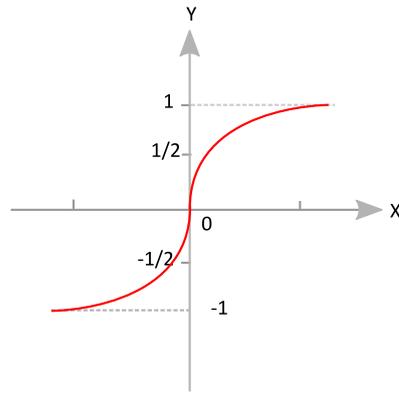
$$F_k(x) = \begin{cases} 1 & \text{si } x < -c \\ 0 & \text{si } x > c \\ \frac{x}{2c} + \frac{1}{2} & \text{en otro caso} \end{cases}$$



$$F_k(x) = \begin{cases} 1 & \text{si } x < -c \\ 0 & \text{si } x > c \\ ax & \text{en otro caso} \end{cases}$$

Figura 4-2: Función lineal y mixta.

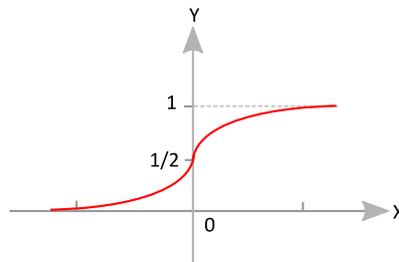
Fuente: Ponce P. Inteligencia artificial con aplicaciones a la ingeniería.



$$F_k(u) = \frac{e^u - e^{-u}}{e^u + e^{-u}}$$

Figura 5-2: Función tangente hiperbólica.

Fuente: Ponce P. Inteligencia artificial con aplicaciones a la ingeniería.



$$F_k(u) = \frac{1}{1 + e^{-u}}$$

Figura 6-2: Función sigmoidea.

Fuente: Ponce P. Inteligencia artificial con aplicaciones a la ingeniería.

2.3.3 Función Tangente Hiperbólica.

Es utilizada generalmente en entrenamientos supervisados, su forma suave, como lo muestra la Fig. 5-2, la caracteriza (Ponce, 2010).

2.3.4 Función Sigmoidea,

Esta función tiende a comportarse como la función escalón, cuando los valores de entrada están alejados del cero, pudiendo utilizar reglas de aprendizaje que fueron definidas para funciones escalón. Su importancia radica, en que su derivada existe en todo el intervalo y siempre es positiva, la Fig. 6-2 muestra su gráfica (Ponce, 2010).

Una red multicapa, siempre tiene funciones de activación no lineales, caso contrario esta es equivalente a una red de una única capa (Izaurieta & Saavedra, 1999).

2.4 Tipos de entrenamiento.

Cuando un algoritmo de aprendizaje, utiliza la salida computada de la red y la compara con una salida deseada conocida, modificando los pesos hasta que el error entre estas dos salidas sea el mínimo se le conoce como algoritmo de aprendizaje supervisado. En cambio, cuando el algoritmo modifica los pesos para que las salidas de la red sean consistentes, es decir, frente a unas entradas parecidas se obtengan la misma salida, a este algoritmo se lo llama aprendizaje no supervisado (Basogain Olabe, 2005).

En el presente trabajo se utiliza el trabajo supervisado.

2.5 Algoritmos de Entrenamiento.

El objetivo de entrenar una red neuronal es que su salida se aproxime a la salida conocida de un sistema en particular. Con este afán existen muchos algoritmos, cuya meta es minimizar el error entre la salida calculada por la RNA con respecto a la salida conocida del sistema. En forma general el problema de optimización se describe como: "dado \mathbb{R}^n como conjunto y una función $f : \mathbb{R}^n \rightarrow \mathbb{R}$, se desea hallar un $\bar{x} \in \mathbb{R}^n$ tal que cumpla" (Hernández, 2006):

$$f(\bar{x}) \leq f(x), \quad \forall x \in \mathbb{R}^n \quad (2.9)$$

"Donde \bar{x} se le conoce como mínimo global de la función f en \mathbb{R}^n y se denota" (Hernández, 2006):

$$\min_{x \in \mathbb{R}^n} f(x)$$

2.5.1 La Función de Error.

Una forma de calcular el error entre una salida conocida s y una salida y calculada por la RNA, es la descrita en la ecuación 2.10.

$$er = \frac{1}{2}(s_1 - y_1)^2 + \frac{1}{2}(s_2 - y_2)^2 \dots + \frac{1}{2}(s_n - y_n)^2 \quad (2.10)$$

En forma general para que un punto $\bar{x} \in \mathbb{R}^n$ sea un mínimo local, de una función f en \mathbb{R}^n , doblemente diferenciable, bastara cumplir las siguientes condiciones (Hernández, 2006):

- Que \bar{x} sea un punto crítico de f .

$$\nabla f(\bar{x}) = \vec{0} \Leftrightarrow \begin{bmatrix} \frac{\partial f(\bar{x})}{\partial x_1} \\ \frac{\partial f(\bar{x})}{\partial x_2} \\ \vdots \\ \frac{\partial f(\bar{x})}{\partial x_n} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (2.11)$$

- Y que la matriz hessiana $\nabla^2 f(\bar{x})$ de f en \bar{x} sea positiva.

$$\nabla^2 f(\bar{x}) = \begin{bmatrix} \frac{\partial^2 f(\bar{x})}{\partial x_1^2} & \frac{\partial^2 f(\bar{x})}{\partial x_1 \partial x_2} & \dots & \frac{\partial^2 f(\bar{x})}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f(\bar{x})}{\partial x_2 \partial x_1} & \frac{\partial^2 f(\bar{x})}{\partial x_2^2} & \dots & \frac{\partial^2 f(\bar{x})}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f(\bar{x})}{\partial x_n \partial x_1} & \frac{\partial^2 f(\bar{x})}{\partial x_n \partial x_2} & \dots & \frac{\partial^2 f(\bar{x})}{\partial x_n^2} \end{bmatrix} > 0 \quad (2.12)$$

Dado que, para hallar un mínimo, es necesario resolver un sistema de ecuaciones no lineales, surge la necesidad de aplicar métodos numéricos. Los métodos numéricos llamados de descenso, se los puede clasificar en tres grupos, como lo indica la Fig. 7-2 (Hernández, 2006).

Para nuestro caso la función del error (ecuación 2.10) solo depende de las salidas y_n calculadas por la RNA, ya que las salidas s_n se las obtiene de los datos históricos, es decir son constantes. Las salidas y_n a su vez dependen de los pesos y los umbrales, como lo indica la ecuación 2.3, por lo que la derivada del error es una derivada parcial con respecto a cada una de los pesos y umbrales que contengan cada salida.

Para ejemplificar estos hechos, se utilizará una red con una entrada, una salida y una capa oculta con dos neuronas, descrita en la Fig. 8-2. La salida de la RNA está definida como

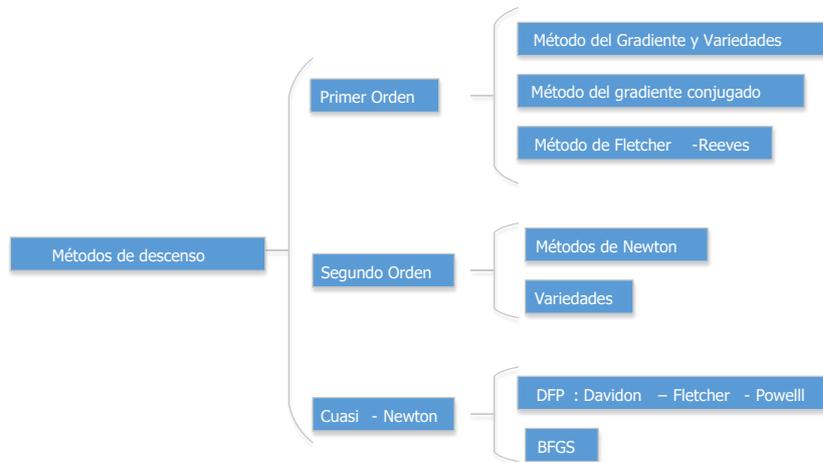


Figura 7-2: Clasificación de los métodos numéricos llamados de descenso.

Fuente: Hernández G. Métodos clásicos de optimización para problemas no-lineales sin restricciones.

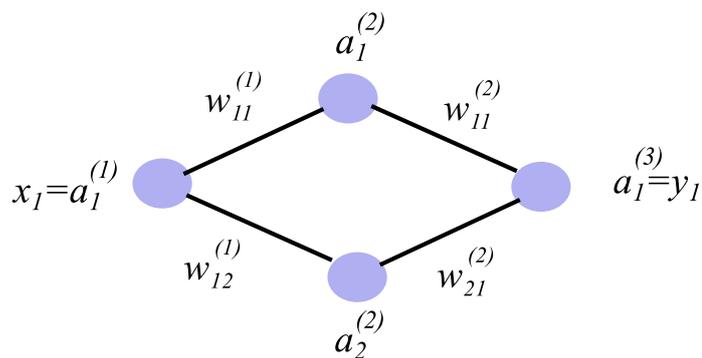


Figura 8-2: RNA simple de ejemplo.

Realizado por: Gabriel Asqui. 2016

lo indica la ecuación 2.13, la cual depende de los pesos w y los umbrales θ , por lo que para minimizar la función del error será necesario derivar parcialmente con respecto a esas variables.

$$y_1 = f \left[\theta_1^{(3)} + w_{11}^{(2)} \cdot a_1^{(2)} + w_{21}^{(2)} \cdot a_2^{(2)} \right]$$

$$a_1^{(2)} = f \left[\theta_1^{(2)} + w_{11}^{(1)} \cdot a_1^{(1)} \right]$$

$$a_2^{(2)} = f \left[\theta_2^{(2)} + w_{12}^{(1)} \cdot a_1^{(1)} \right]$$

$$y_1 = f \left[\theta_1^{(3)} + w_{11}^{(2)} \cdot f \left[\theta_1^{(2)} + w_{11}^{(1)} \cdot a_1^{(1)} \right] + w_{21}^{(2)} \cdot f \left[\theta_2^{(2)} + w_{12}^{(1)} \cdot a_1^{(1)} \right] \right] \quad (2.13)$$

Generalizando la derivada parcial del error estará definida por la ecuación 2.14.

$$\frac{\partial e}{\partial w_*} = \sum_{i=1}^n \frac{\partial e}{\partial y_i} \cdot \frac{\partial y_i}{\partial w_*} \quad (2.14)$$

Siendo w_* el peso o el umbral, con respecto al cual se va a derivar parcialmente.

2.5.2 Algoritmo Back-propagation.

Utiliza el método del descenso del gradiente, que forma parte de los métodos de primer orden, como se indica en la Fig. 7-2. Basa la adaptación de los pesos y umbrales, siguiendo una dirección de búsqueda, que es el negativo del gradiente de la función de error (Isasi Viñuela & Galván León, 2004).

La modificación de los pesos y los umbrales, está definida por la ecuación 2.15 y la ecuación 2.16 respectivamente.

$$w_n = w_v - \alpha \frac{\partial e}{\partial w} \quad (2.15)$$

$$\theta_n = \theta_v - \alpha \frac{\partial e}{\partial \theta} \quad (2.16)$$

Donde:

w_n : Es el nuevo peso calculado.

w_v : Es el peso anterior.

α : Tasa de aprendizaje o paso, entre 0 y 1, influye en la magnitud del desplazamiento en la función del error.

Como se puede evidenciar, el algoritmo de Back-Propagation utiliza solo la primera derivada de la función de error, por lo que se garantiza la convergencia en algún punto mínimo local (Ghosh & Mriganka, 2012).

Del parámetro α depende la rapidez de convergencia del algoritmo, además de prevenir o provocar quedarse atrapado en un mínimo local, por lo que el entrenamiento depende de escoger la correcta tasa de aprendizaje (Ghosh & Mriganka, 2012).

2.5.3 Algoritmo Broyden Fletcher Goldfarb Shanno (BFGS).

Para explicar este algoritmo partiremos del algoritmo de Newton, el cual aproxima la función a optimizar mediante la serie de Taylor hasta orden dos, como lo expresa la ecuación 2.17 (Hernández, 2006).

$$f(x) \approx \Phi(x) = f(\bar{x}) + (x - \bar{x})\nabla f(\bar{x}) + \frac{1}{2}(x - \bar{x})Hf(\bar{x})(x - \bar{x})^T \quad (2.17)$$

Se debe considerar que para que \bar{x} sea un mínimo debe satisfacer la condición de la ecuación 2.9 y cumplir con la condición de la ecuación 2.11. Tomando en cuenta estos criterios, en la ecuación 2.17 se tiene (Hernández, 2006):

$$f(\bar{x}) \leq f(x) = f(\bar{x}) + \frac{1}{2}(x - \bar{x})Hf(\bar{x})(x - \bar{x})^T$$

Es decir:

$$\frac{1}{2}(x - \bar{x})Hf(\bar{x})(x - \bar{x})^T = \frac{1}{2}(\Delta x)Hf(\bar{x})(\Delta x)^T \geq 0$$

Si la matriz hessiana $Hf(\bar{x})$ es definida positiva se cumple que:

$$(\Delta x)Hf(\bar{x})(\Delta x)^T > 0$$

Que corresponde a la condición número dos dada por la ecuación 2.12.

El valor para el siguiente paso estará dado por la ecuación 2.18.

$$x_{k+1} = x_k - [Hf(x_k)]^{-1}[\nabla f(x_k)] \quad (2.18)$$

Ejemplo:

Supongamos $f(x)$ de una sola variable con $f'(x)$ y $f''(x)$ conocidas, se escoge un punto x_k y se aproxima $f(x)$ entorno a ese punto, usando la expansión de Taylor, obtendremos (Hernández, 2006):

$$q_k(x) = f(x_k) + f'(x_k)(x - x_k) + \frac{1}{2}f''(x_k)(x - x_k)^2$$

Si resolvemos el $\min q(x)$ y no el $\min f(x)$, utilizando la condición de la ecuación 2.11 tendremos :

$$\frac{dq_k}{dx} = f'(x_k) + f''(x_k)(x - x_k) = 0$$

Si despejamos x y la re-definimos como x_{k+1} se tiene:

$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)} \quad (2.19)$$

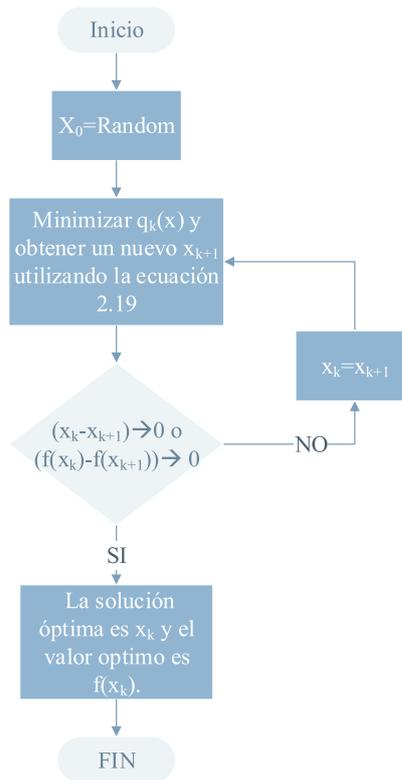


Figura 9-2: Esquema iterativo de minimización utilizando el método de Newton.

Realizado por: Gabriel Asqui. 2016

Luego de lo cual se puede definir un esquema iterativo que nos lleve al mínimo, como lo muestra la Fig. 9-2(Hernández, 2006).

Como observamos en el método de Newton es necesario calcular el inverso de la matriz Hessiana como lo muestra la ecuación 2.18, lo que conlleva un alto costo computacional (Hernández, 2006). El algoritmo BFGS que forma parte de los algoritmos llamados cuasi-Newton, en cambio basa su búsqueda de un mínimo dentro de la función, utilizando una matriz S de dirección y de tamaño $n \times n$, siendo n el número de variables de la función, cuyo fin es el mismo que la inversa de la matriz hessiana del método de Newton, pero sin tener que calcular la matriz hessiana exacta (Ghosh & Mriganka, 2012).

El valor del siguiente paso se calcula usando la ecuación 2.20.

$$x_{k+1} = x_k - t_k S_k g_k \quad g_k = \nabla f(x_k) \quad (2.20)$$

El esquema iterativo para el método BFGS se muestra en la Fig. 10-2.

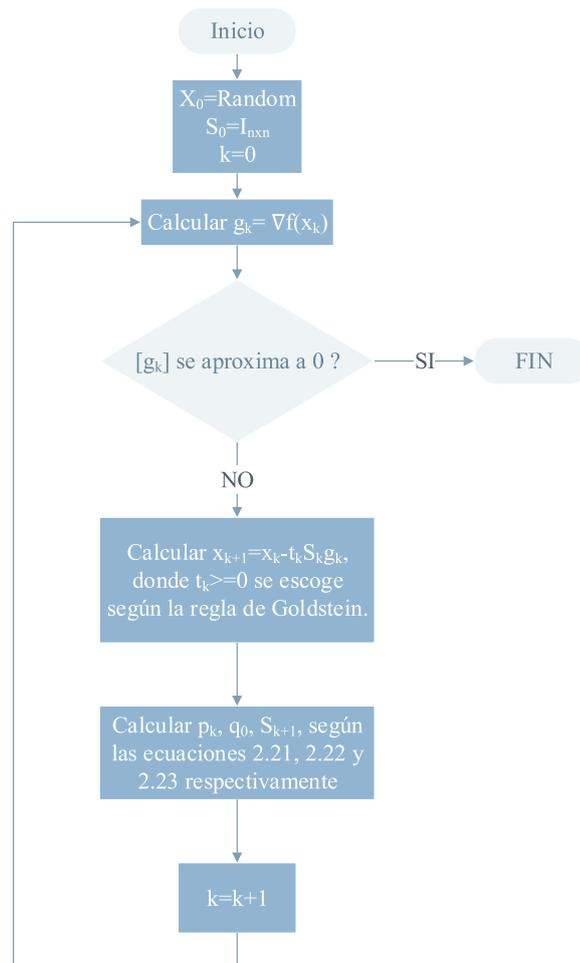


Figura 10-2: Esquema iterativo método BFGS.
Realizado por: Gabriel Asqui. 2016

$$p_k = x_{k+1} - x_k \quad (2.21)$$

$$q_k = g_{k+1} + g_k \quad (2.22)$$

$$S_{k+1} = S_k + \left[I + \frac{(q_k)^T S_k q_k}{(p_k)^T q_k} \right] \frac{p_k (p_k)^T}{(p_k)^T q_k} - \frac{[p_k (q_k)^T S_k^T + S_k q_k (p_k)^T]}{(p_k)^T q_k} \quad (2.23)$$

Regla de Goldstein.

Permite la elección del tamaño de paso en cada iteración, para cualquier método de optimización seleccionado, determinando tres criterios. Siendo k la iteración de un método de optimización, $t_k \geq 0$ el tamaño del paso y D_k la dirección de descenso tendremos (Hernández, 2006):

$$x_{k+1} = x_k + t_k D_k$$

Para determinar el paso t_k se calcula un intervalo $[t_g, t_d]$, $0 \leq t_g \leq t_d$, tal que $t_k \in [t_g, t_d]$ (Hernández, 2006).

Para definir los tres criterios sea:

$$h(t) = f(x_k + tD_k) - f(x_k), \quad t \geq 0$$

$$h'(t) = \nabla f(x_k + tD_k)^T D_k$$

$$h'(0) = \nabla f(x_k)^T D_k < 0$$

$$h(t) \leq 0$$

Ademas:

$$0 < m_1 < m_2 < 0$$

Los criterios son:

Criterio (a): $m_2 h'(0)t \leq h(t) \leq m_1 h'(0)t$

Criterio (b): $h(t) > m_1 h'(0)t$

Criterio (c): $h(t) < m_2 h'(0)t$

En la Fig. 11-2 se ejemplifica la elección del paso de Goldstein, en donde en azul se representa la función de error, $h(0)$ es el punto escogido para aproximar dicha función usando la serie de Taylor hasta la segunda derivada, $th'(0)$ es la tangente en dicho punto, y los rayos $th'(0)m_2$ y $th'(0)m_1$ sirven para ir ajustando el paso de descenso hasta el mínimo local.

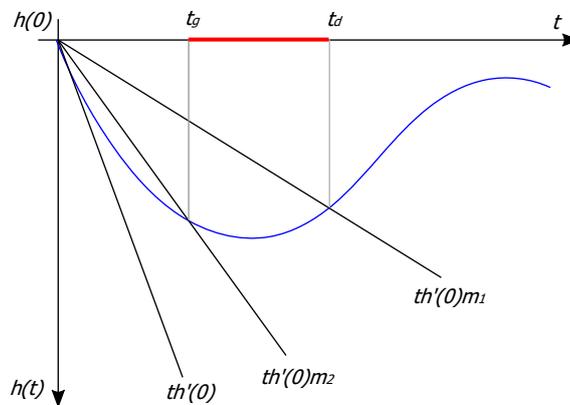


Figura 11-2: Elección de paso de Goldstein.

Fuente: Hernandez G. Métodos clásicos de optimización para problemas no-lineales sin restricciones.

El esquema iterativo del algoritmo general de cálculo de paso se muestra en la Fig. 12-2 (Hernández, 2006).

Debido a que algoritmos como el Back-Propagation, presenta el problema de una lenta convergencia, se escoge el algoritmo BFGS que presenta una convergencia mas rápida, a pesar de tener un consumo mayor de memoria (Ghosh & Mriganka, 2012).

2.6 RNA como Predictor.

Las series temporales son conjuntos de datos de un suceso a lo largo del tiempo, donde su evolución temporal no depende del tiempo explícitamente sino de los valores de la serie en instantes anteriores o de otras variables temporales (Isasi Viñuela & Galván León, 2004).

En la mayoría de series temporales es difícil llegar a una ecuación que describa la relación explícita entre el valor de la serie en un instante de tiempo y sus valores anteriores, por lo que las RNA han sido utilizadas ampliamente por sus características, como son la capacidad de aproximar y capturar relaciones a partir de un conjunto de ejemplos sin saber la distribución de los datos, el construir relaciones no lineales, el construir relaciones a partir de información incompleta o con ruido y por su facilidad de construir y utilizar (Isasi Viñuela & Galván León, 2004).

Un ejemplo de pronóstico de series de tiempo es el presentado por S. Joekes, E. Barbosa y W. Robledo, en donde se comparan procedimientos estadísticos tradicionales frente a las RNA con conexiones feed-forward (Hacia adelante), y algoritmo de entrenamiento Back-Propagation, donde se aprecia que los mejores pronósticos lo obtienen las RNA (Joekes *et al.*, 2005).

Tanto el perceptrón multicapa como las redes de base radial encuentran relaciones entre patrones de entrada-salida independientes de la variable de tiempo. Sin embargo, esta particularidad no impide utilizarlos para tratar información temporal. Para lograrlo se debe dar como entrada a la red además del valor del patrón en un determinado espacio de tiempo, una cierta historia de dicho patrón, utilizando una secuencia finita temporal en el pasado. Otra opción es utilizar redes de neuronas recurrentes, en donde la salida dependa de las entradas y de los estados anteriores, evitando la necesidad de proveer a la red de una secuencia de valores anteriores (Isasi Viñuela & Galván León, 2004).

2.6.1 Predicción en un Paso de Tiempo.

La predicción en un paso de tiempo, como muestra la Fig. 13-2, puede seguir dos esquemas. El primero, mostrado en la Fig. 13-2 (a), utiliza los datos entre los instantes t y $t - d$ anteriores para obtener el dato en el instante $t + 1$, siendo t el instante actual, mientras

que el segundo, mostrado en la Fig. 13-2 (b), utiliza los datos de los instantes entre t y $t - d$ anteriores para obtener el dato en el instante $t + u + 1$ (Isasi Viñuela & Galván León, 2004).

En el momento de entrenamiento de la RNA, se deben dividir los datos históricos, en datos de entrada y en datos objetivos. Los datos de entrada serán los datos que se entregarán como entradas de la RNA, y los datos objetivos serán los datos que se compararán con la salida de la RNA para el ajuste de los pesos. La división de datos para los dos esquemas de la predicción en un paso de tiempo se muestra en la Fig. 14-2.

2.6.2 Predicción en Múltiples Pasos de Tiempo.

La predicción en múltiples pasos de tiempo, pretende predecir el comportamiento de la serie hasta un umbral de predicción indicado por un número natural. Esto puede conseguirse utilizando los esquemas de la Fig. 15-2 (Isasi Viñuela & Galván León, 2004).

El esquema de la Fig. 15-2 (a) re-alimenta la salida del predictor en un paso, de la Fig. 13-2 (a), para conseguir múltiples predicciones hasta el umbral indicado por lo que de aquí en adelante se lo llamará ***Método 1*** (Isasi Viñuela & Galván León, 2004).

Por otro lado, el esquema de la Fig.15-2 (b), utiliza los datos entre los instantes $t - u$ y $t - u - d$ anteriores, los cuales forman una ventana, para obtener el dato del instante $t + 1$, luego la ventana se mueve un paso y con los datos entre los instantes $t - u + 1$ y $t - u - d + 1$, obtiene el dato del instante $t + 2$, este proceso se repite hasta alcanzar el umbral indicado, de aquí en adelante se lo llamará ***Método 2*** (Isasi Viñuela & Galván León, 2004).

La Fig. 16-2 da un ejemplo de la aplicación de los dos métodos de predicción en varios pasos, tomando secuencias en forma de vectores.

La división de datos de entrada y datos objetivo para el entrenamiento de la RNA para predicción en múltiples pasos de tiempo sigue el mismo esquema de la Fig. 14-2a para el Método 1 y el esquema de la Fig. 14-2b para el Método 2.

2.7 RNA como Modelo.

La identificación de sistemas es el arte de construir modelos matemáticos de sistemas dinámicos. Esta tarea es ligada ampliamente a la comunidad de Control, pero fue construida a partir de técnicas estadísticas como son la regresión lineal y no-lineal (Garrido, 1999).

El modelo, construido a partir de las entradas y salidas observadas es capaz de representar o simular el comportamiento dinámico del proceso. Los modelos cuya construcción se basan en los datos observados se los conocen como NARMA (Nonlinear Auto-Regressive Moving Average) para el caso de procesos dinámicos no lineales. El uso de RNA suele utilizarse debido a que son más tolerables al ruido, que los modelos construidos con polinomios (Isasi Viñuela & Galván León, 2004).

La ecuación 2.24 describe el modelo NARMA (Isasi Viñuela & Galván León, 2004).

$$y(t) = f(y(t-1), \dots, y(t-n_y); u(t-1), \dots, u(t-n_u)) \quad (2.24)$$

Donde:

$$y(t) = \begin{bmatrix} y_1(t) \\ \vdots \\ y_m(t) \end{bmatrix}; u(t) = \begin{bmatrix} u_1(t) \\ \vdots \\ u_r(t) \end{bmatrix}$$

Las señales $y(t)$ y $u(t)$, son las salidas y las entradas del sistema respectivamente; n_y y n_u son los valores máximos históricos de las salidas y las entradas, necesarios para modelar; y $f(*)$ es la función no lineal que se desea aproximar mediante la RNA, utilizando para nuestro caso el algoritmo de entrenamiento BFGS (Chen *et al.*, 1990).

La Fig. 17-2, muestra el diagrama de entrenamiento de la RNA, en donde se escogen los datos históricos del caudal y potencia como entradas discretas, y el nivel del embalse discreto histórico como salida del sistema PRESA.

Estas tres señales, en conjunto se convierten en entradas de la RNA. La salida de nivel predicha por la RNA se compara con la salida real de nivel, obteniéndose el error del

modelo, que es usado para ajustar los pesos y umbrales de la RNA, mediante el algoritmo BFGS.

Una vez entrenada la RNA, esta puede ser utilizada para simular el comportamiento del sistema PRESA, para lo cual se debe re-alimentar la propia salida de la RNA, debido a que en entrenamiento se utilizaba los datos $y(k + 1)$, no disponibles en la simulación. Todo este proceso es representado en la Fig. 18-2.

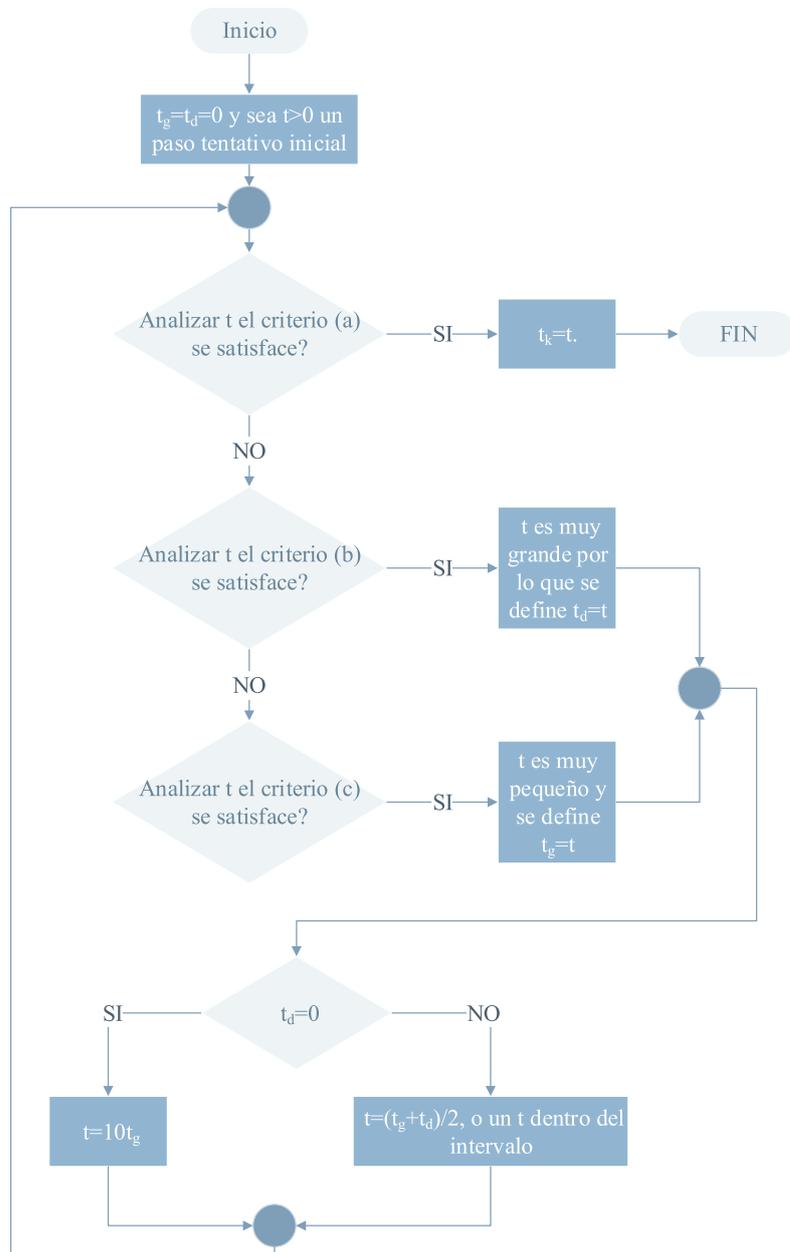


Figura 12-2: Esquema iterativo para la elección del paso.
 Realizado por: Gabriel Asqui. 2016

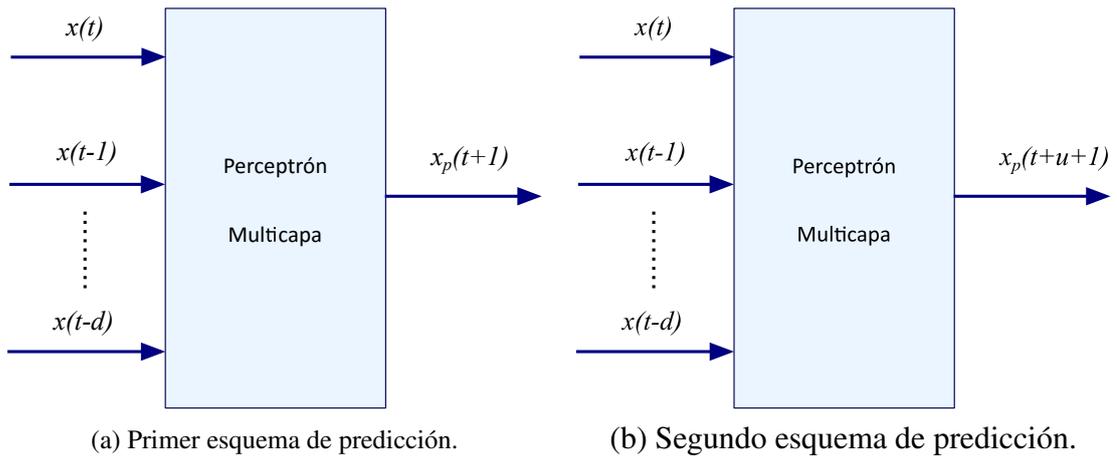


Figura 13-2: Esquemas de predicción en un paso
 Realizado por: Gabriel Asqui. 2016

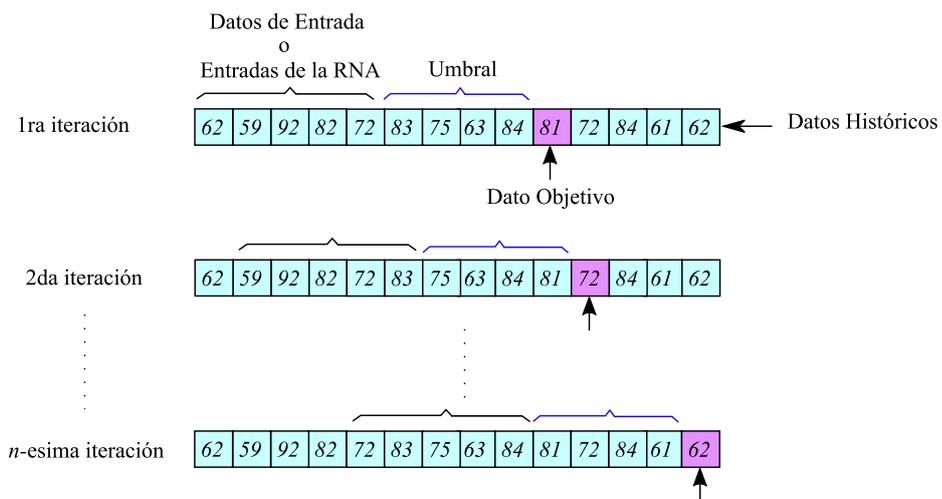
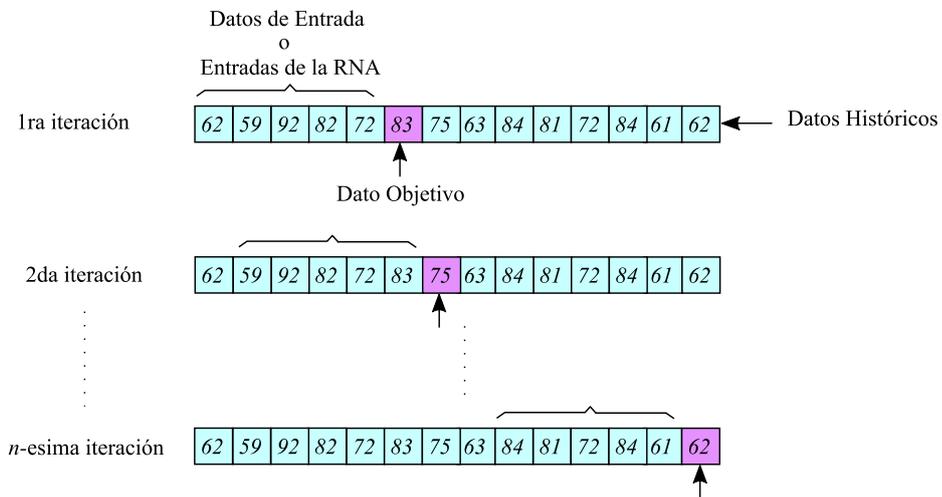


Figura 14-2: División de datos históricos.
 Realizado por: Gabriel Asqui. 2016

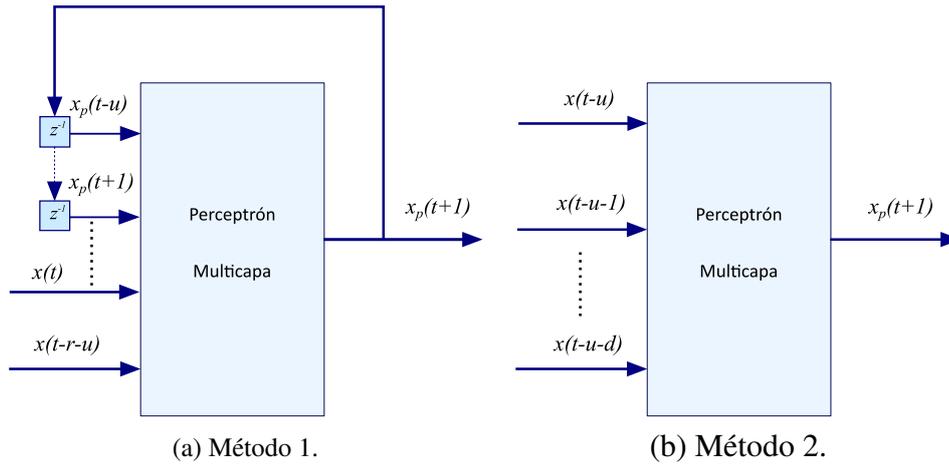
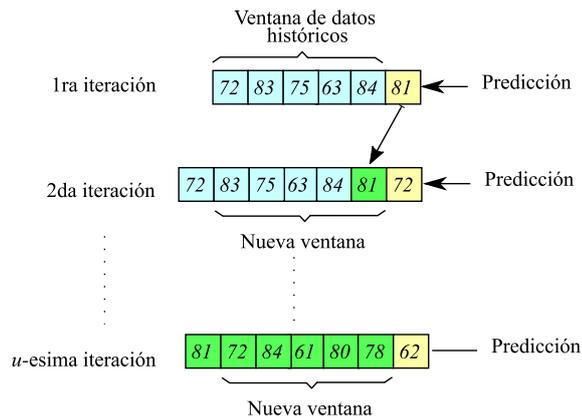
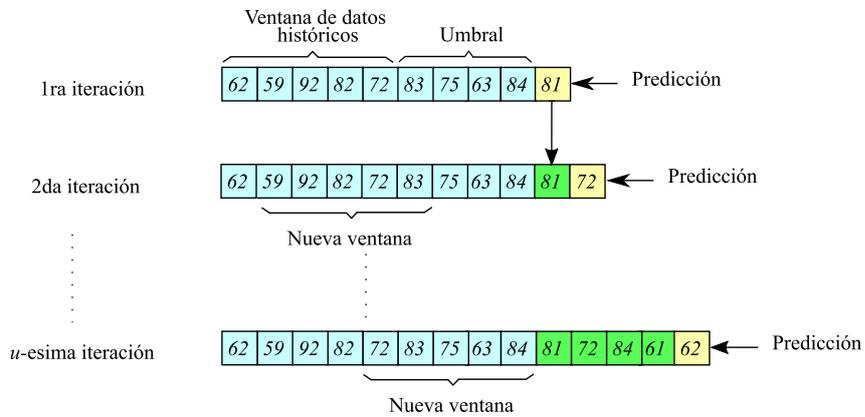


Figura 15-2: Predictor varios pasos.
Realizado por: Gabriel Asqui. 2016



(a) Predicción de varios pasos usando el método 1.



(b) Predicción de varios pasos usando el método 2.

Figura 16-2: Ejemplos predicción varios pasos.
Realizado por: Gabriel Asqui. 2016

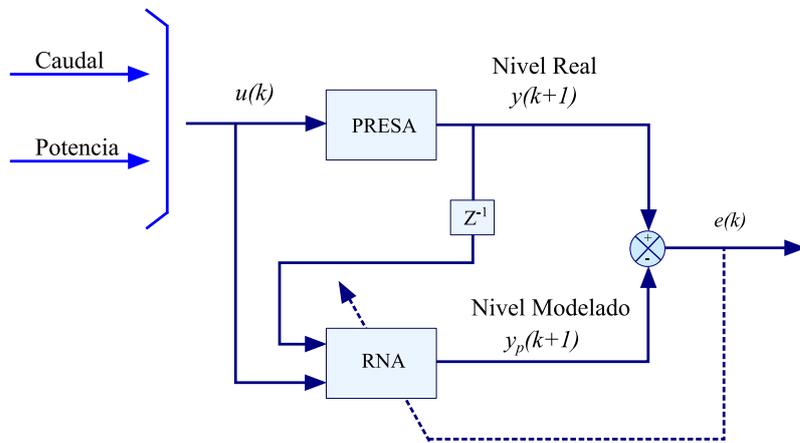


Figura 17-2: Diagrama de entrenamiento.
 Realizado por: Gabriel Asqui. 2016

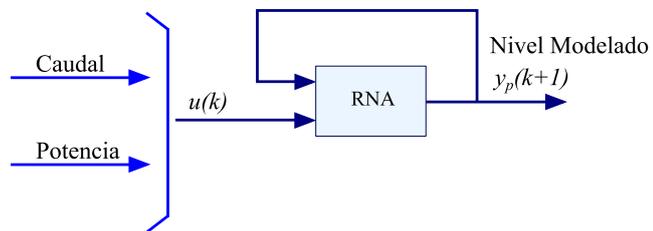


Figura 18-2: Diagrama de modelado.
 Realizado por: Gabriel Asqui. 2016

CAPÍTULO III

3 . METODOLOGÍA DE LA INVESTIGACIÓN.

Con el afán de plasmar todos los conceptos mostrados en el capítulo dos, sobre predicción y modelamiento de sistemas físicos, este capítulo muestra la metodología utilizada para analizar los datos históricos, de caudal nivel y potencia activa total generada, y de esta forma obtener una aplicación de software que realice la predicción del nivel de embalse de la central Hidroeléctrica Agoyán a través de la predicción de caudal, con el fin de alcanzar posibles mejoras en la planificación de producción de energía eléctrica.

3.1 Preparación y Análisis de Datos.

La central Hidroeléctrica Agoyán posee un sistema SCADA (Supervisory Control And Data Acquisition) que adquiere los datos de los sensores de nivel de embalse y de caudal del río Pastaza.

Estos datos son guardados dentro de una Base de Datos (BDD). Sin embargo, debido a que no fue posible acceder a la BDD generada en el SCADA por ser un sistema propietario, los datos entregados fueron los recogidos por los operadores desde el 01/01/2005 hasta el 31/04/2016, en formato *.xlsx y *.xlsm.

Por otro lado, otra dificultad que se encontró, fue que las plantillas usadas para guardar los datos iban cambiando a través de los años por lo que fue necesario; primero llevar todos los datos a un solo formato, y segundo crear una BDD propia utilizando el software PostgreSQL[®].

Las librerías para Python[®] como son: OpenPyXl[®], para el manejo de archivos de formato *.xlsx y *.xlsm, junto a Psycopg[®] que comunica Python[®] con PostgreSQL[®] hicieron la tarea de llevar los 102.702 datos por variable, de los formatos *.xlsx y *.xlsm, a la BDD de PostgreSQL[®], mucho más sencilla.

Los datos que se suben a la BDD con sus respectivas unidades son los siguientes:

- Nivel de embalse en metros sobre el nivel del mar.
- Caudal del río Pastaza en metros cúbicos por segundo.
- Potencia total activa generada en Mega Vatios.
- Potencia total reactiva generada en Mega Volt-Amperio Reactivo.

De las dos potencias guardadas, solo se tomará en cuenta para el análisis, la potencia total activa, ya que es la que tiene influencia en la apertura y cierre de los álabes de la turbina, provocando el consumo o no del agua del embalse, sin embargo se suben los datos de potencia reactiva para futuros proyectos.

Estos 102.702 datos por variable constituyen el universo de datos a analizar, de los cuales se extrae una muestra no probabilista, con el objeto de que la RNA no aprenda comportamientos ilógicos, tomando en cuenta los siguientes criterios:

- Se eliminan los datos de caudal iguales a cero o menores a $35 \text{ m}^3/\text{s}$ debido a que estos datos corresponderían a errores al tipear ya que el caudal por datos históricos nunca a sido menor a los $35 \text{ m}^3/\text{s}$ y los datos iguales a cero corresponden a los caudales que no se registran cuando la central no está generando.
- Se eliminan los datos de potencia activa superiores a 156 MW, correspondientes a errores al tipear ya que la capacidad máxima de producción es de 156 MW.
- Se eliminan los datos de nivel superiores a 1651m.s.n.m, correspondientes a errores al tipear ya que el nivel máximo son los 1651m.s.n.m.

Luego de la aplicación de estos filtros la muestra no probabilista la conforman 93.874 datos.

El gráfico 1-3 muestra un ejemplo del caudal anual del río Pastaza para el año 2005, nótese que existen temporadas de alto y bajo caudal por lo que la Fig. 1-3 muestra las tres temporadas climáticas asignadas al caudal dentro de la empresa, que son: invernall fuerte, invernall ligera y verano.

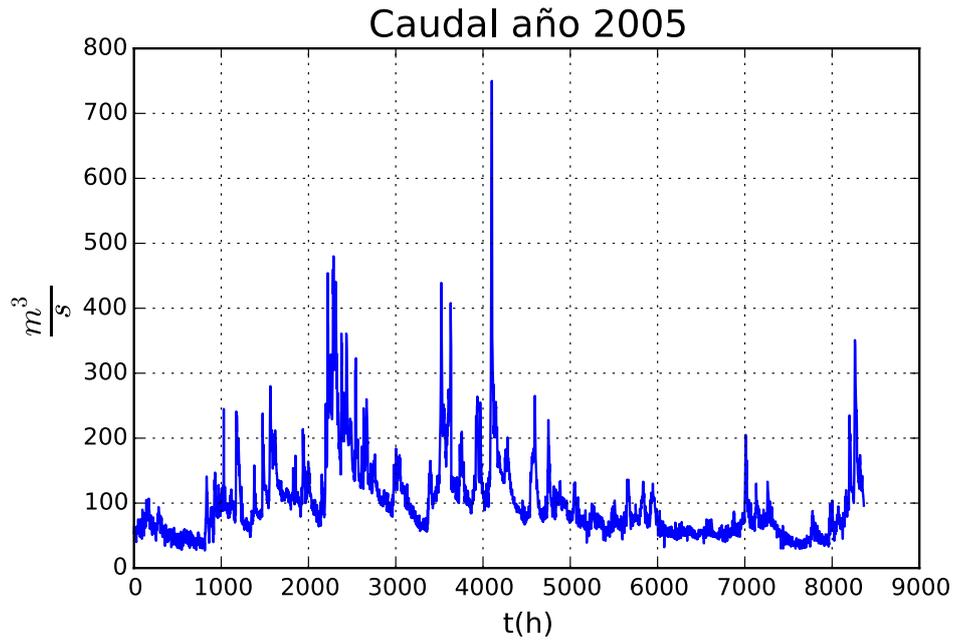
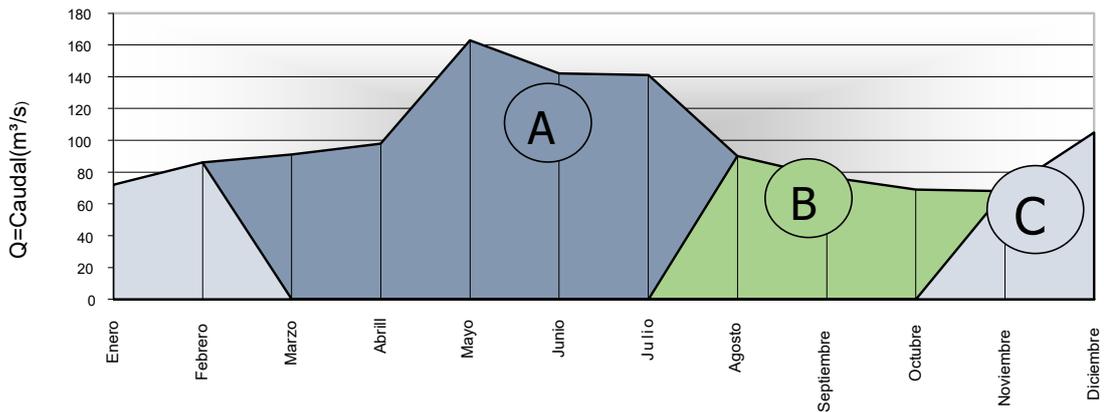


Gráfico 1-3: Caudal río Pastaza.
Realizado por: Gabriel Asqui. 2016

PERIODO	MESES
INVERNAL FUERTE	MARZO - JULIO
INVERNAL LIGERA	AGOSTO - OCTUBRE
VERANO	NOVIEMBRE - FEBRERO



Meses	Ene.	Feb.	Mar.	Abr.	Mayo	Junio	Julio	Agos	Sept.	Oct.	Nov	Dic.
Q(m³/s)	72	86	91	98	163	142	141	90	77	69	68	105

Figura 1-3: Asignación de temporadas para el caudal del Pastaza.

Fuente: CELEC-EP

Estas temporadas tienden a repetirse cada año por lo que se escogen tres (3) años con temporadas muy distintas para el análisis, reduciendo la muestra no probabilista a 8300 datos promedio por temporada con los que se trabajan en el presente trabajo.

Dado que la serie temporal (ST) del caudal no es estacionaria, es necesario aproximar su comportamiento a una serie temporal estacionaria (STE) retirando los componentes de tendencia, esto se consigue aplicando una transformación preliminar de los datos utilizando una función, que para nuestro caso es la logarítmica $[\ln(x_1), \dots, \ln(x_n)]$.

De esta forma se penaliza las magnitudes grandes y se favorece a las pequeñas logrando una ST mas uniforme. El gráfico 2-3 muestra un ejemplo de la transformación mencionada para las temporadas veraniegas de los años 2012 a 2014 (Brockwell & Davis, 2002).

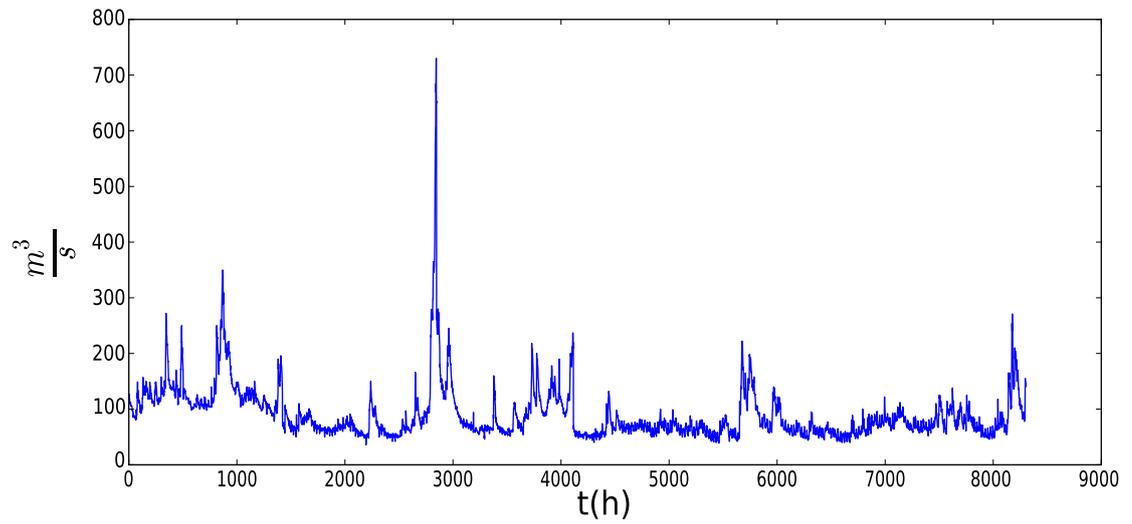
Los gráficos 3-3 (a) y 3-3 (b) muestra como la transformación de la ST ayuda a acentuar los picos de PSD, ya que la aproxima a una STE. Se debe buscar una transformación que permita el retorno a datos originales sin pérdidas importantes de información.

Para comprobar la mejora en la planificación de la producción es necesario hacer dos definiciones, de acuerdo a las sugerencias de los especialistas de la empresa:

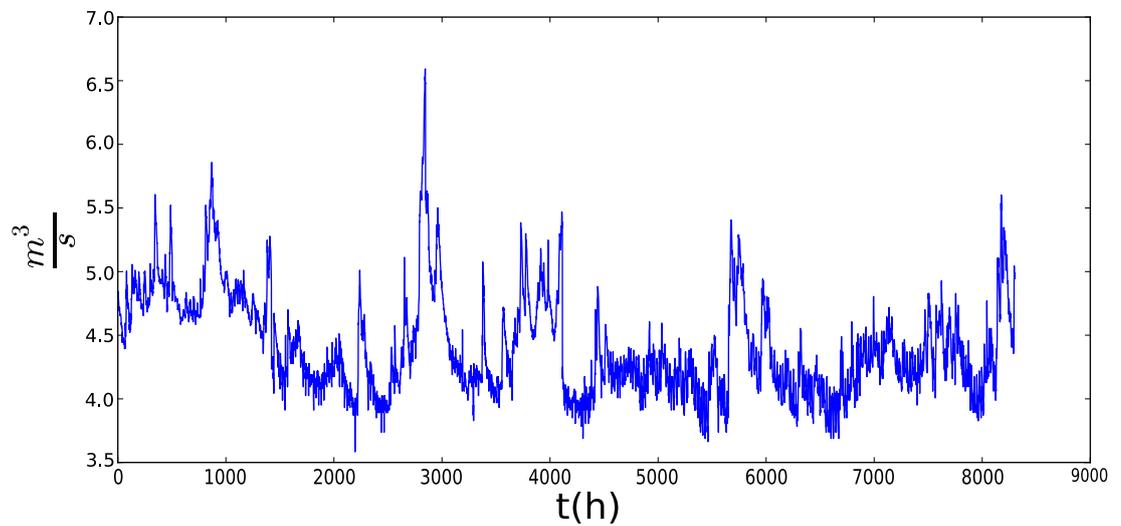
- Cambio de Producción.- Cuando se modifica la producción a partir de $\pm 5\text{MW}$.
- Reprogramaciones.- Cuando se modifica la producción a partir de $\pm 20\text{MW}$.

El gráfico 4-3 muestra los cambios de producción y reprogramaciones por temporada de cada año, donde se aprecia que la temporada verano tiene un incremento sustancial en el número de cambios de producción y reprogramaciones, este aumento se debe a que en verano el caudal de río es bajo y les es difícil, según los especialistas de la central, dar una programación adecuada de producción.

En cambio, en las temporadas invernal fuerte e invernal ligera es más sencillo determinar la producción diaria ya que los caudales son altos y superan los $120 \text{ m}^3/\text{s}$ necesarios para producir a plena carga, por lo que evidencian una reducción en estos dos índices. Nótese que la temporada invernal ligera no tiene datos en el año 2006, ya que la central estuvo parada.



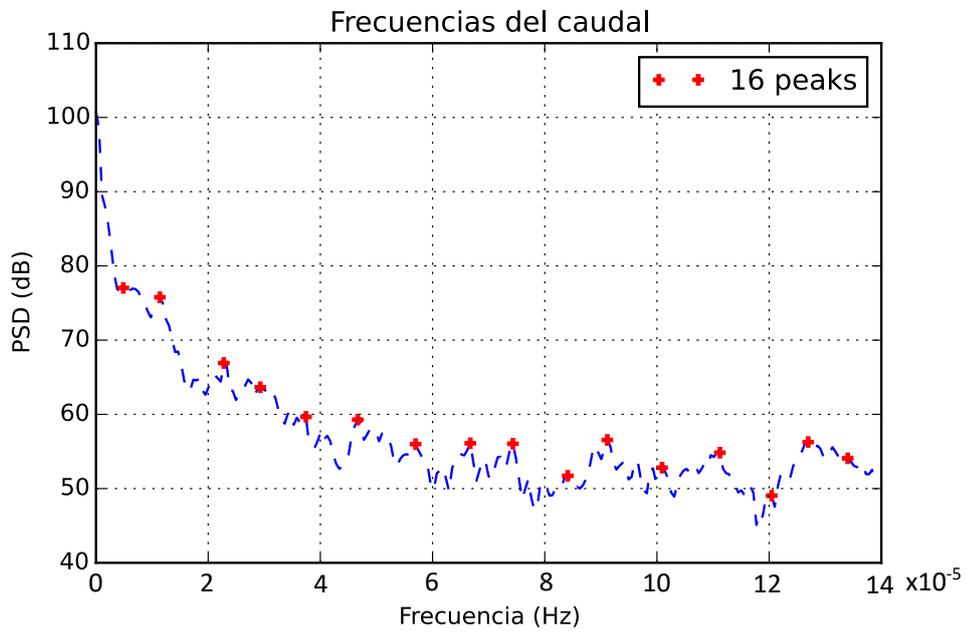
(a) Caudal temporadas veraniegas sin transformación.



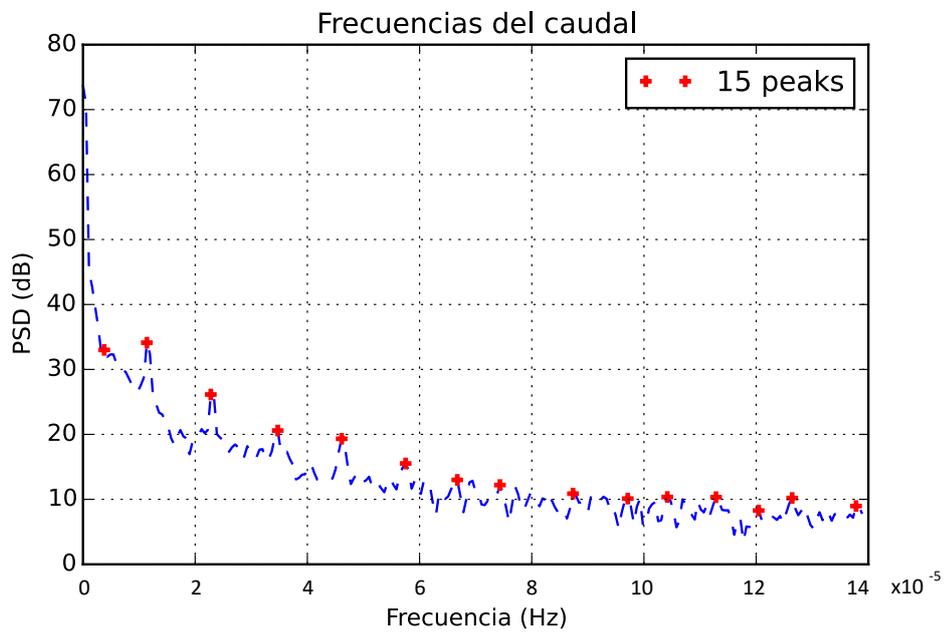
(b) Caudal temporadas veraniegas con transformación logarítmica.

Gráfico 2-3: Caudal temporadas veraniegas 2012-2014.

Realizado por: Gabriel Asqui. 2016

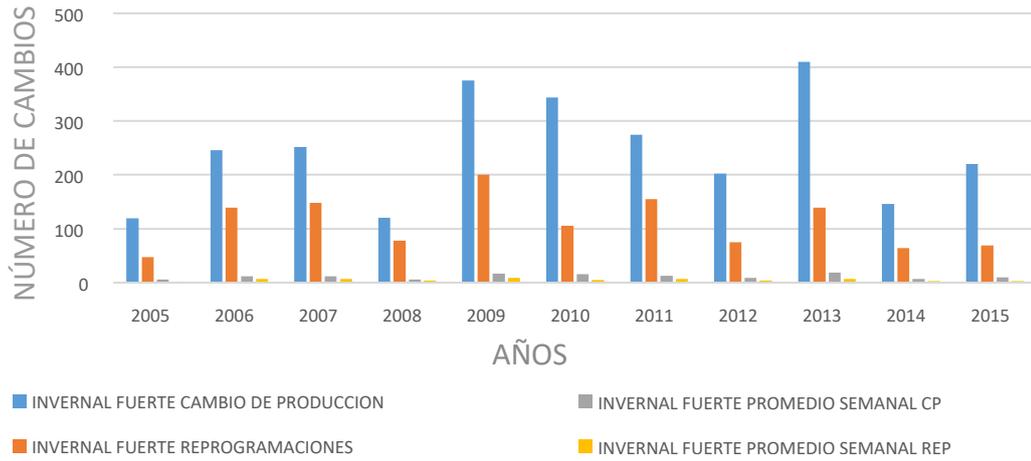


(a) Picos de PSD sin transformación.

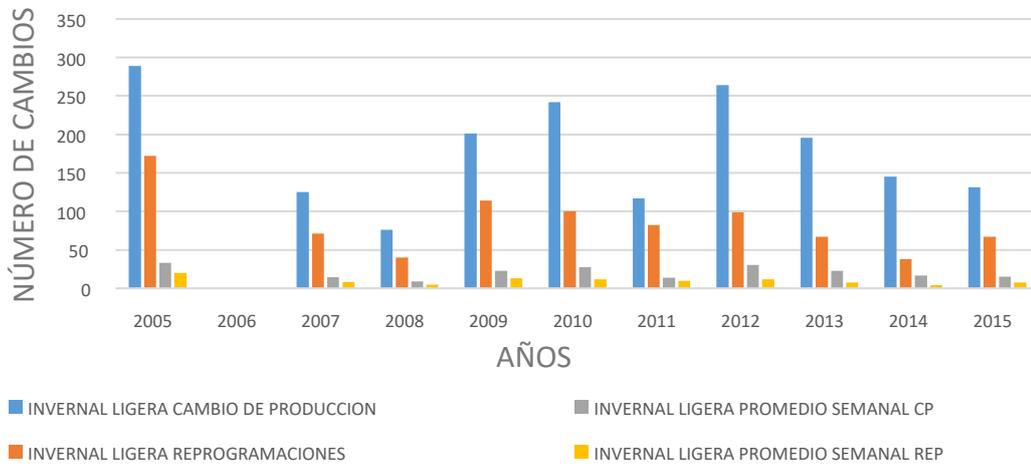


(b) Picos de PSD con transformación.

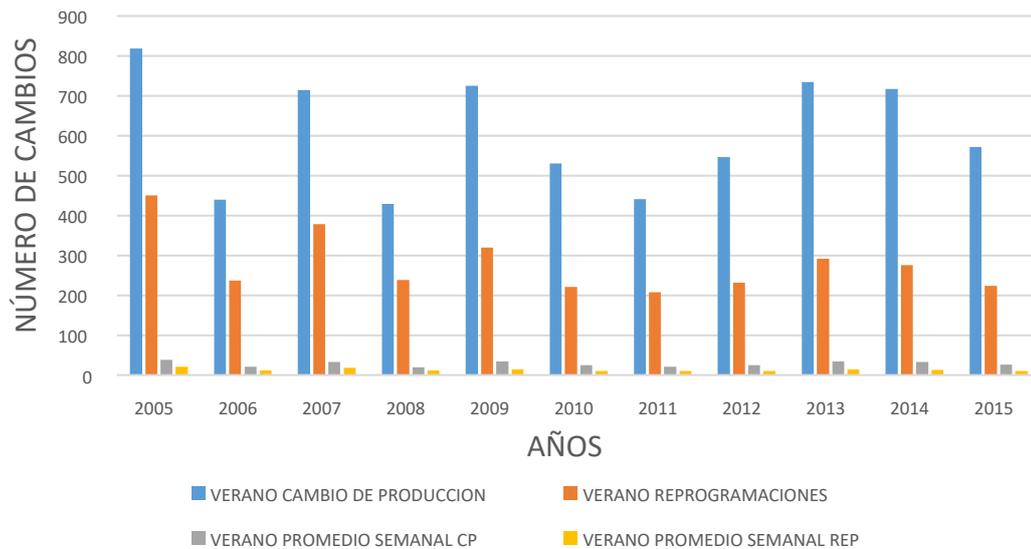
Gráfico 3-3: Picos de PSD temporadas veraniegas.
Realizado por: Gabriel Asqui. 2016



(a) Temporada Invernal Fuerte.



(b) Temporada Invernal Ligera.



(c) Temporada Verano.

Gráfico 4-3: Cambios de producción y reprogramaciones por temporada.
Realizado por: Gabriel Asqui. 2016

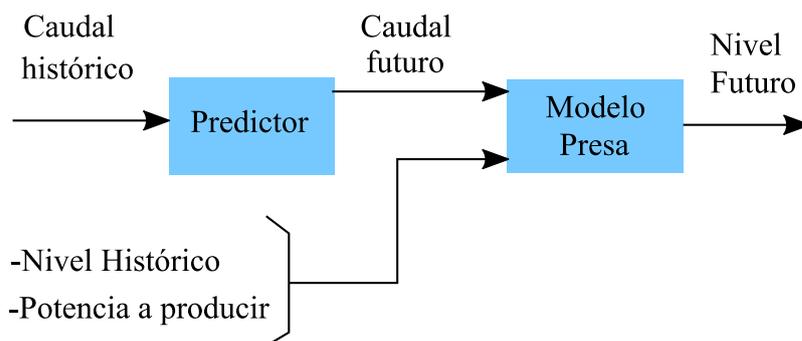


Figura 2-3: Diagrama de bloques del sistema.
Realizado por: Gabriel Asqui. 2016

3.2 Determinación de los Modelos.

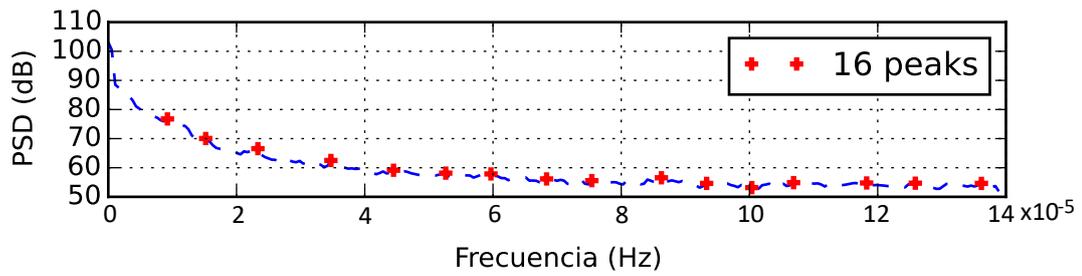
Para obtener la predicción del nivel del embalse como lo muestra la Fig. 2-3, es necesario primero determinar un modelo predictor de caudal que arroje los datos futuros de caudal en base a los datos históricos y segundo alimentar estos datos futuros de caudal al modelo de la presa junto con los niveles históricos y la potencia activa a generar, para de esta manera determinar el comportamiento del nivel futuro del embalse.

Tanto para el modelo predictor de caudal como para el modelo presa, es necesario determinar de forma aproximada el número de neuronas de entrada como el de las capas ocultas de la RNA, para luego aplicar los algoritmos de entrenamiento descritos en el capítulo dos, con esta meta procedemos a realizar un análisis de frecuencias, en las señales de caudal, nivel y potencia activa total generada (Zaldivar *et al.*, 2000).

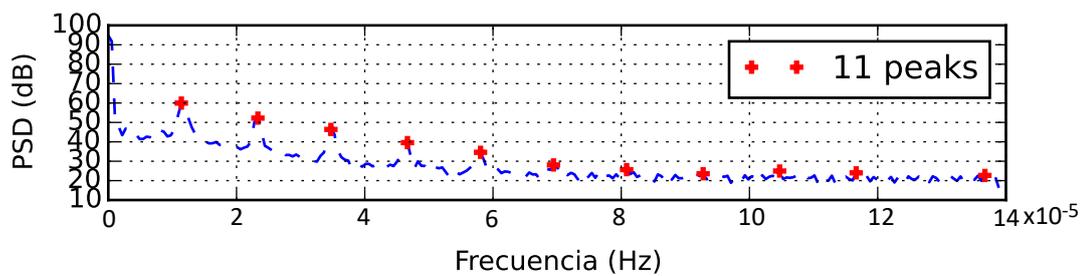
El gráfico 5-3 muestra un ejemplo de los picos del PSD (power spectral density) de las distintas señales para el año 2005, estos picos muestran cuanto de la señal esta contenida en cierta frecuencia, mientras más grande sea el pico más señal estará contenida en esa frecuencia.

Las frecuencias de los picos determinan aproximadamente el número de neuronas para las entradas y las capas ocultas, en el gráfico 6-3 se muestra la dispersión de frecuencias en horas desde el año 2005 hasta el 2015.

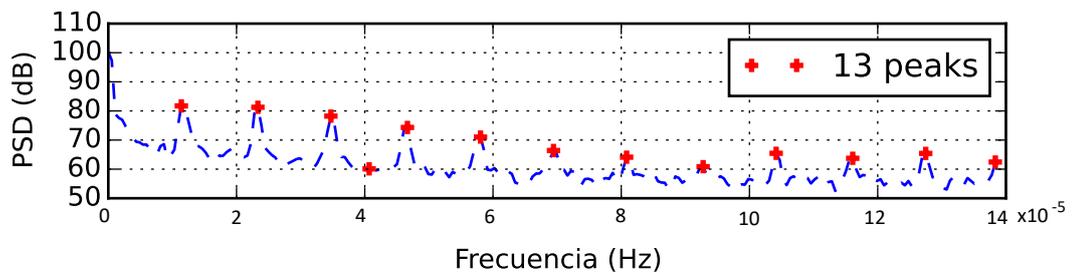
Se escoge una configuración de red con dos capas ocultas para el predictor como para el modelo de la presa, ya que en pruebas presentaron mejor comportamiento que las de una sola capa oculta.



(a) Señal de Caudal.

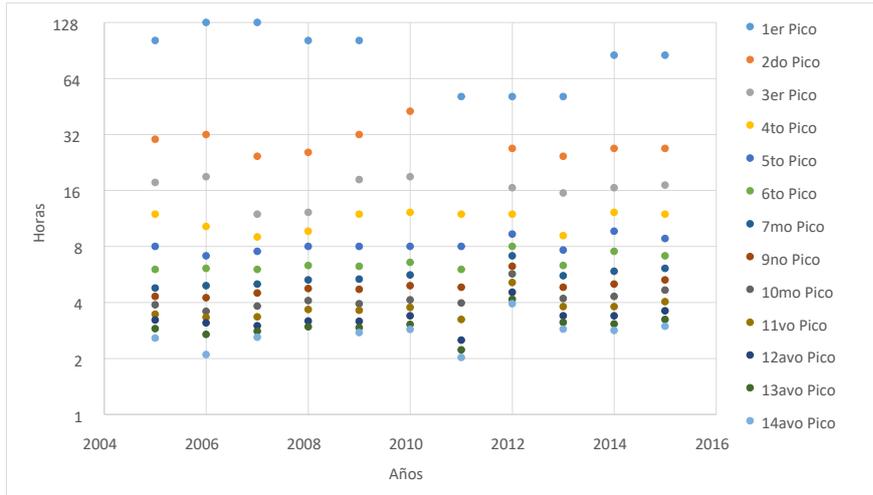


(b) Señal de Nivel.

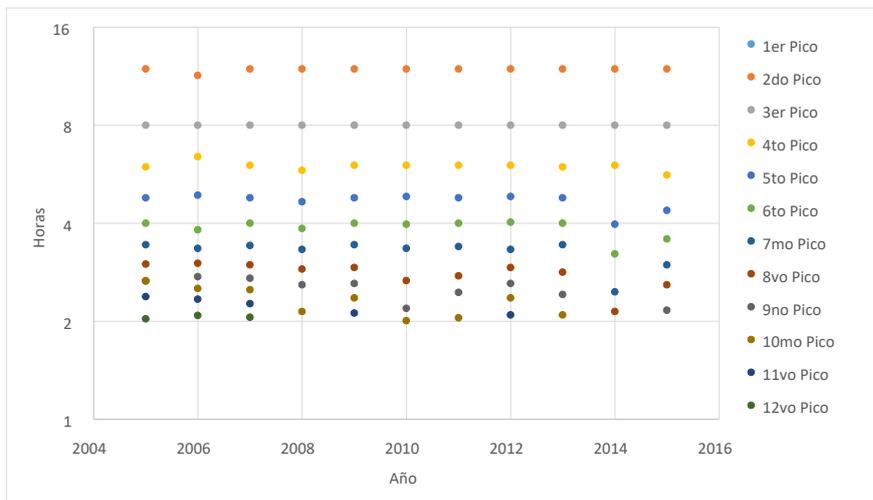


(c) Señal de Potencia Activa.

Gráfico 5-3: Análisis de frecuencias año 2005.
Realizado por: Gabriel Asqui. 2016



(a) Frecuencia en horas de los picos del PSD de la señal de caudal.



(b) Frecuencia en horas de los picos del PSD de la señal de nivel.



(c) Frecuencia en horas de los picos del PSD de la señal de potencia activa.

Gráfico 6-3: Dispersión de frecuencias.
Realizado por: Gabriel Asqui. 2016

Para obtener la red con mejores resultados, es necesario ir combinando el número de neuronas, determinadas por los picos de PSD, tanto para las entradas como para las capas ocultas, e ir entrenando dichas redes.

Se entrenan las redes durante 100 épocas con el algoritmo bfgs proporcionado por las librerías de NeuroLab[®], no es necesario realizar más épocas de entrenamiento, para determinar que red tendrá mejor comportamiento, ya que con este número de entrenamientos se puede ver la tendencia del error, es decir si el error disminuye drásticamente entre configuraciones.

Todas estas pruebas son almacenadas en una tabla dentro de la BDD para su posterior análisis. Una vez encontrada la red con el mejor comportamiento se la entrena durante 1000 épocas, lo que toma alrededor de 5 horas, para el caso de los predictores y 48000 épocas para el caso de los modelos, lo que toma 48 horas, utilizando un computador con procesador Intel(R) Core(TM) i7-4702MQ CPU @ 2.20GHz y memoria de 15,9 GB.

La razón por la que existe esta diferencia de número de épocas será descrita en las siguientes secciones.

3.2.1 *Predictores.*

La planificación de producción se la envía al Centro Nacional de Control de Energía (CENACE) hasta las nueve de la mañana de cada día, para el día siguiente, es decir con un umbral de predicción de 48 horas.

Sin embargo, para la planificación diaria, dentro de la central, con un umbral de 24 horas, se utiliza los datos proyectados el día anterior y las condiciones climáticas de ese día pudiendo o no haber un ajuste en lo planificado.

La planificación de producción para el día, umbral de 24 horas, en la actualidad depende de la experiencia del operador, para cumplir dos objetivos:

1. No tener que cambiar a cada momento la producción.
2. No quedarse sin agua en el embalse para las horas pico de 6pm a 9pm.

El presente trabajo pretende alcanzar el umbral de 24 horas de la planificación diaria; sin embargo, se realizaron pruebas para el umbral de 48 horas, además del umbral de 8 horas que es lo que dura un turno de operación.

El utilizar las temporadas climáticas, implica entrenar un predictor por temporada y por método de predicción obteniéndose seis predictores en total.

Una vez determinada la configuración más adecuada, como se explicó en el apartado anterior, se entrena la red y su configuración se la guarda en una tabla dentro de la BDD.

El guardado de la configuración de la RNA es necesaria debido a que se necesita saber el número de neuronas de entrada que se usó en el entrenamiento en el momento de la predicción.

Para el entrenamiento de los predictores por temporada se juntan los datos de tres años con temporadas históricas muy distintas, pretendiendo así que la red aprenda el mayor número de casos diferentes posibles.

Por ejemplo, se juntan todas las temporadas invernales de tres años y se entrena una red bajo el método número uno. La Fig. 3-3 muestra la pantalla dedicada a este propósito dentro de la aplicación, se debe notar que para el caso de los predictores del método dos es necesario escoger el umbral de predicción, ya que con este dato se apuntan los datos objetivos que la red debe aprender como se explica en el apartado 2.6.2 del capítulo 2.

Cabe recalcar que se realizan entrenamientos por temporadas con el afán de que la red aprenda el comportamiento del río Pastaza según la temporada, y comprobar si de esta forma su predicción es más efectiva dentro de la temporada entrenada.

La Tabla 1-3 muestra unos ejemplos de las configuraciones probadas para el predictor de temporada invernal fuerte.

Al entrenar una RNA se trabaja con dos errores, el de entrenamiento, que nos dice que tanto se acercan los datos producidos por la RNA a los datos objetivo dados para el entrenamiento, y el de validación que nos indica cuanto se parecen los datos producidos por la RNA frente a los datos que no están en el conjunto de entrenamiento, siendo el error de validación el factor a tomar en cuenta para determinar la configuración adecuada (Isasi Viñuela & Galván León, 2004).



Figura 3-3: Pantallas de los entrenadores de predictores.

Realizado por: Gabriel Asqui. 2016

Tabla 1-3: Pruebas de configuración de red para el predictor de temporada invernal fuerte.

Predictor Temporada Invernal Fuerte					
Neuronas de Entrada	Primera Capa Oculta	Segunda Capa Oculta	Error de entrenamiento	Error de validación	Epoch
2	6	2	9,9150286	44,216942	100
3	6	3	7,7328248	27,162733	100
3	88	3	15,389417	32,756832	100
6	88	6	7,8679037	28,508532	100
6	102	6	13,750329	28,458172	100
16	102	16	13,935171	23,938168	100

Realizado por: Gabriel Asqui. 2016

El gráfico 7-3 muestra que, aunque el error de entrenamiento siga disminuyendo el de validación puede estancarse o subir debido al sobre entrenamiento de la RNA. Por esta razón es necesario implementar dentro del algoritmo de entrenamiento una etapa de validación y de control, que grabe la RNA con menor error de validación y no la con menor error de entrenamiento para no caer en el sobre entrenamiento. Esto explica la diferencia de épocas de entrenamiento para el caso del predictor y del modelo.

Para el caso de la implementación de los entrenadores de predictores, se utilizó la programación multi-hilo, utilizando las librerías de PyQt[®], de esta forma se evita que el sistema operativo termine cerrando la aplicación debido al tiempo que toma el solo dar una iteración a los lazos, dentro del algoritmo de entrenamiento; lo que provoca que aparezca el

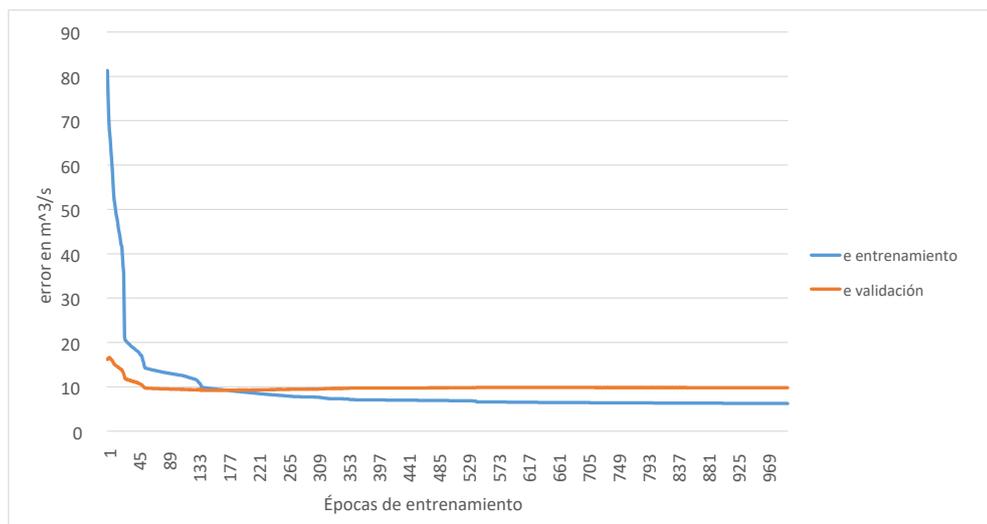


Gráfico 7-3: Errores de entrenamiento y validación.
Realizado por: Gabriel Asqui. 2016

mensaje *"la aplicación no responde"*.

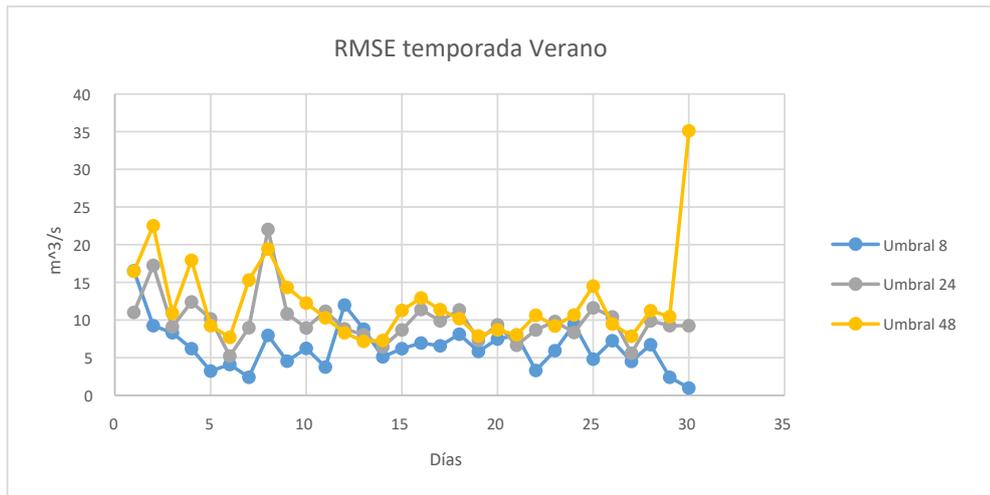
Para comprobar la efectividad del predictor se implementa la pantalla de la Fig. 4-3 en la cual, según la fecha escogida, se hace una predicción y se la compara con el caudal real histórico.

También se va modificando el umbral de predicción de esta forma se evidencia que al ir aumentando el umbral de predicción, el error RMSE (Root Mean Square Error) también lo hace como lo muestra el gráfico 8-3, en el cual se visualiza el RMSE, utilizando el Método 1 tomando treinta días de prueba.

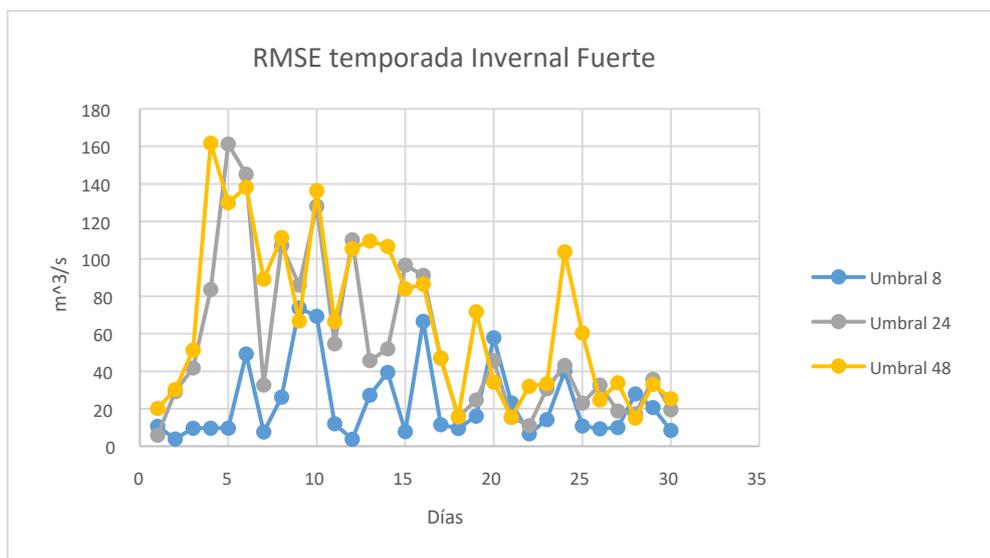
El umbral para la predicción en múltiples pasos del Método 1 es fácilmente variable por no estar ligado al entrenamiento, sin embargo, en el Método 2 este no es el caso, por lo que si se desea variar el umbral hay que entrenar redes específicas para el umbral requerido (véase apartado 2.6.2 del capítulo 2). El gráfico 9-3 muestra el incremento del RMSE para el método 2.

Para los dos métodos los días escogidos por temporada fueron:

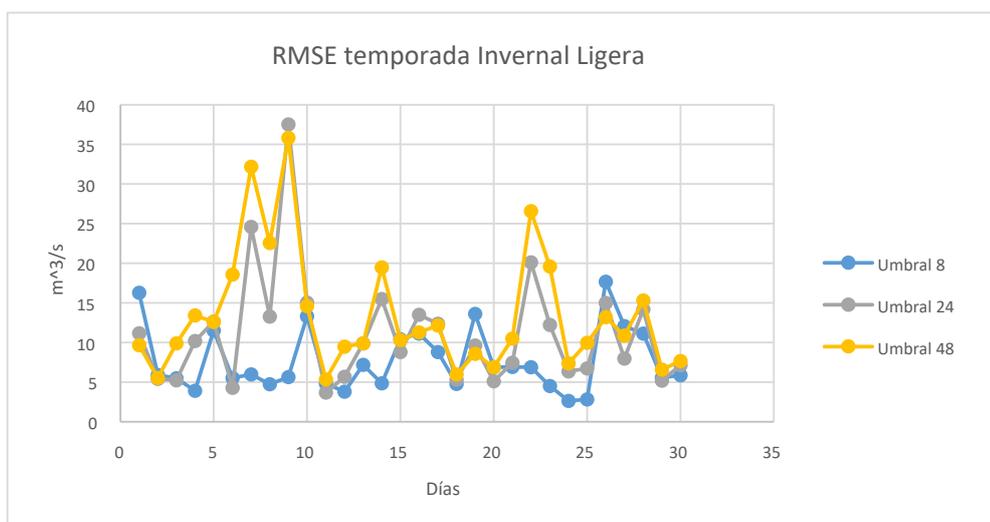
- Temporada Verano: desde el 04/01/2005 hasta el 02/02/2005.
- Temporada Invernal Fuerte: desde el 01/04/2005 hasta el 30/04/2005.
- Temporada Invernal Ligera: desde el 01/08/2005 hasta el 30/08/2005.



(a) RMSE temporada verano método 1.

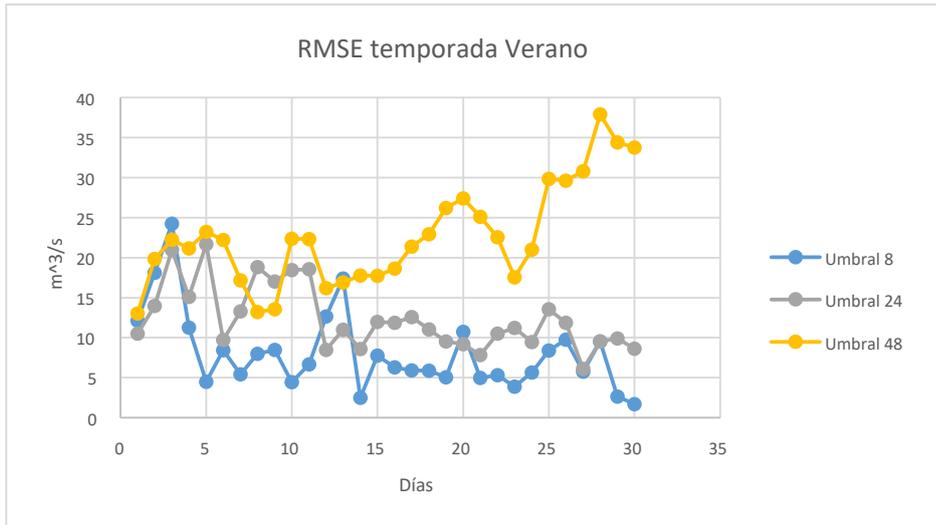


(b) RMSE temporada invernal fuerte método 1.

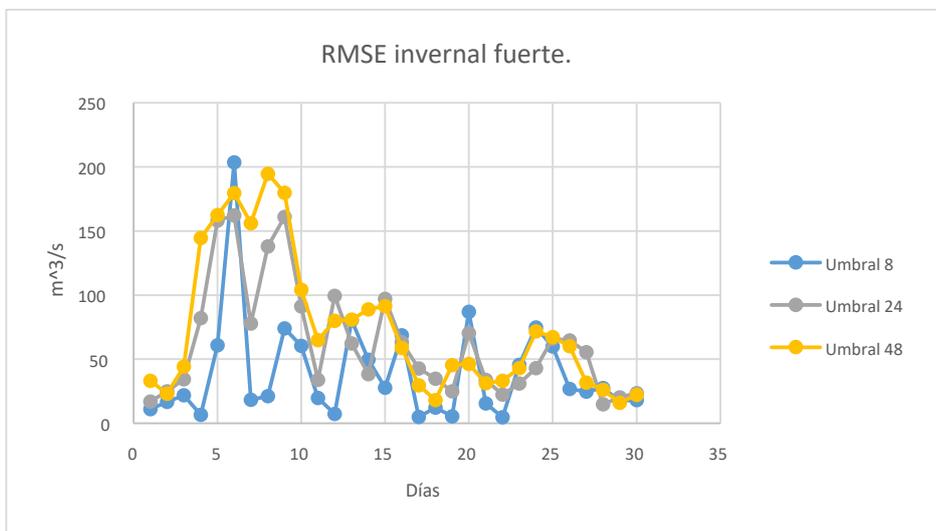


(c) RMSE temporada invernal ligera método 1.

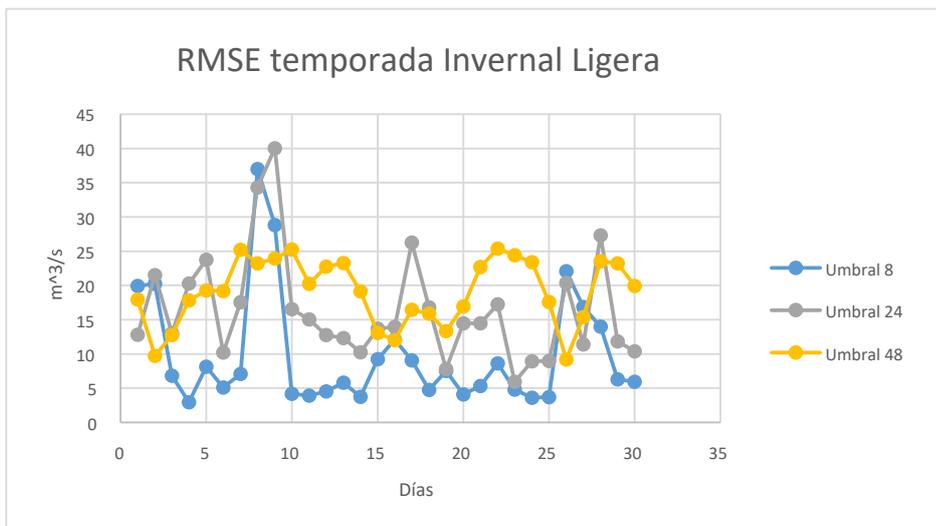
Gráfico 8-3: RMSE de los predictores por temporadas usando el método 1. Realizado por: Gabriel Asqui. 2016



(a) RMSE temporada verano método 2.



(b) RMSE temporada invernal fuerte método 2.



(c) RMSE temporada invernal ligera método 2.

Gráfico 9-3: RMSE de los predictores por temporadas usando el método 2. Realizado por: Gabriel Asqui. 2016



Figura 4-3: Ventana de pruebas para el predictor.
Realizado por: Gabriel Asqui. 2016

En la Tabla 2-3 se puede observar el RMSE promedio de las distintas temporadas y con los distintos métodos, en donde se evidencia que para nuestro caso el Método 1 es el más efectivo sin embargo tiende a tener un incremento sustancial en el RMSE cuando aumenta el umbral, el Método 2 en cambio es más tolerante al incremento del umbral, aunque sea menos efectivo.

El porcentaje de acierto se mide con los caudales que tienen un RMSE menor o igual a $\pm 8 \text{ m}^3/\text{s}$ del valor real, ya que esta diferencia de caudal no tiene mayor incidencia en el nivel de embalse según los expertos de la central, y se lo puede considerar como una predicción válida.

3.2.2 Modelo de la Presa.

Se entrenan tres modelos correspondientes a las distintas temporadas. Dentro de la aplicación el administrador del sistema tiene acceso al entrenador mostrado en la Fig. 5-3, que cuenta al igual que los entrenadores de los predictores, con la seguridad de que si en un entrenamiento la red no converge, o se interrumpe el entrenamiento, esta RNA no

Tabla 2-3: RMSE promedio de los predictores.

Temporada	Método	Umbral horas	RMSE Promedio m^3/s	RMSE mínimo m^3/s	RMSE máximo m^3/s	% de acierto
Verano	1	8	9,28313565	0,99914968	16,537056	76,6666667
		24	15,8874415	5,2117352	22,049847	16,6666667
		48	6,43704070	7,1774287	35,131275	16,6666667
Invernal Fuerte	1	8	23,11047675	3,7633877	73,742752	16,6666667
		24	55,0400548	5,8884201	161,21941	3,3333333
		48	68,0038468	15,150201	161,83633	0
Invernal Ligera	1	8	7,75825607	3,005923	22,561712	70
		24	11,47793412	4,6444907	20,699396	20
		48	13,0153187	6,4760461	27,809381	10
Verano	2	8	8,11219078	1,7124765	24,257475	60
		24	12,3580327	6,1119671	21,67638	6,6666667
		48	22,5986711	13,040374	37,890461	0
Invernal Fuerte	2	8	39,0751780	4,6721721	203,49275	16,6666667
		24	62,9550717	14,803472	162,13844	0
		48	74,28110527	15,927778	194,61774	0
Invernal Ligera	2	8	9,87973111	2,9526362	37,008835	60
		24	16,3383327	5,9296336	40,039402	6,6666667
		48	19,0691933	9,1870117	25,366697	0

Realizado por: Gabriel Asqui. 2016

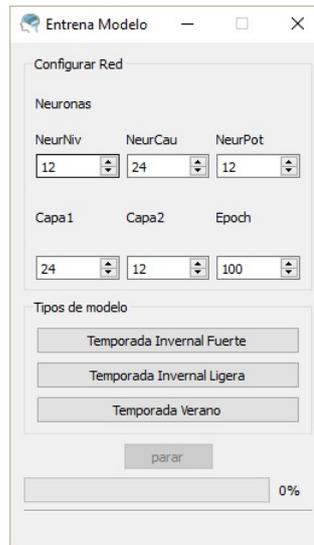


Figura 5-3: Entrenador de Modelos

Realizado por: Gabriel Asqui.
2016

es guardada, garantizando que el sistema no almacene nunca redes sin entrenar y que los recursos de computación no sean desperdiciados.

Al igual que en los predictores se utiliza programación multi-hilo para evitar el colapso de la aplicación, así como el análisis de frecuencias para determinar en forma aproximada las neuronas de entradas de las señales de caudal, nivel y potencia además del número de neuronas para la primera y segunda capa oculta.

Para realizar las pruebas de los modelos se escoge un año al azar, en este caso el 2009, con las siguientes fechas según la temporada:

- Verano: del 01/01/2009 al 04/02/2009, los días entre el nueve y trece no se toman en cuenta por estar parada la central.
- Invernal Fuerte: del 01/04/2009 al 30/04/2009.
- Invernal Ligera: del 01/08/2009 al 30/08/2009.

Las pantallas implementadas para las pruebas son dos:

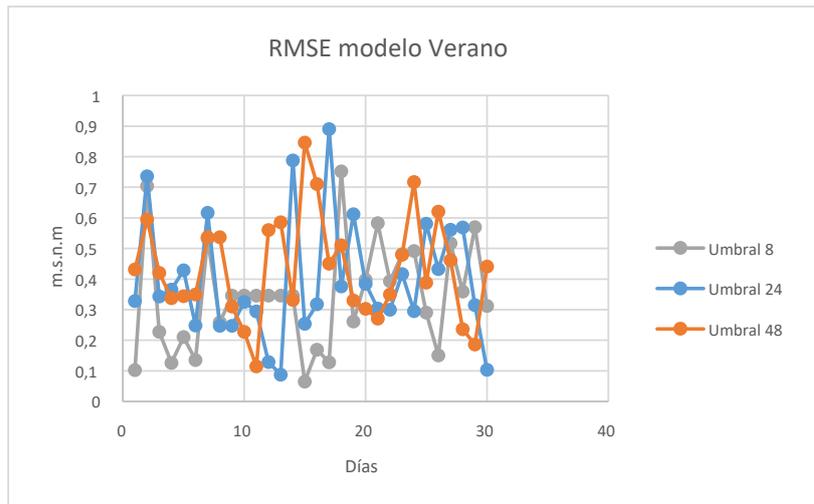
- Primero la pantalla de la Fig. 6-3, en la cual se utiliza los datos históricos reales de caudal, potencia y los datos entregados por el modelo de nivel de la RNA siguiendo



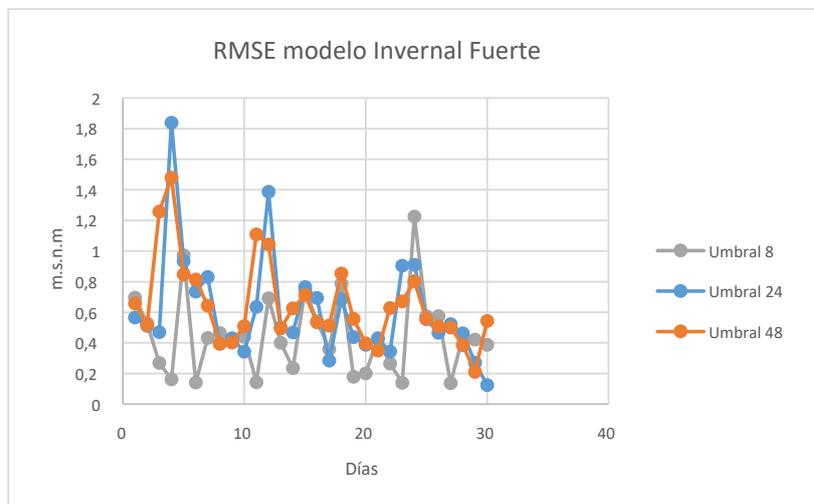
Figura 6-3: Ventana de pruebas para el modelo de la presa.
Realizado por: Gabriel Asqui. 2016

el esquema de la Fig. 18-2, para obtener datos de nivel entregados por los modelos, y compararlos con los datos reales de nivel histórico, según la fecha seleccionada; de esta forma solo se comprueba la exactitud de los modelos de la presa. El gráfico 10-3 muestra el RMSE de las pruebas realizadas durante treinta días para los umbrales de interés por temporada y la Tabla 3-3 muestra el promedio de RMSE de los treinta días. Como se evidencia en los promedios de la Tabla 3-3 los modelos son menos propensos al cambio de umbral que los predictores. El porcentaje de aciertos se mide a partir de los niveles modelados con RMSE menor a 0.5 m.s.n.m.

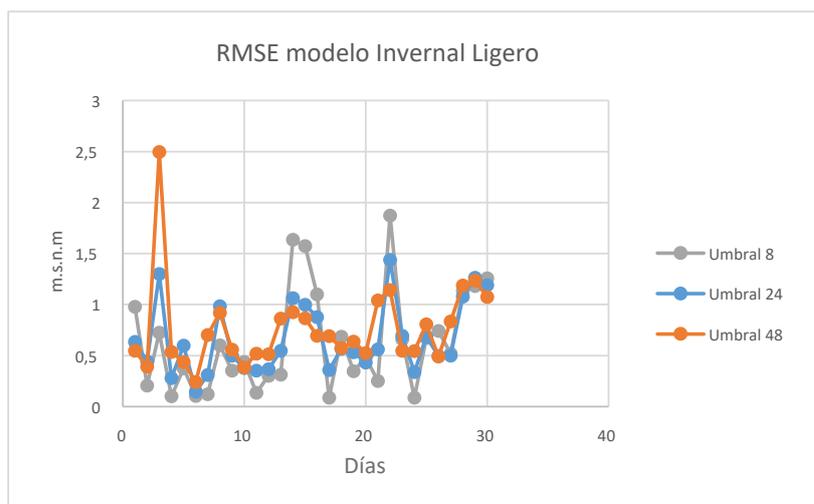
- Segundo la pantalla de la Fig. 7-3, en la cual se utiliza la predicción de caudal, según el tipo de predictor escogido, y se introduce esta señal de caudal como entrada para el modelo, además de la potencia histórica y la re-alimentación del nivel modelado por la RNA, de esta forma se comprueba la exactitud del modelo de la presa, sumado al predictor de caudal. El gráfico 11-3 y 12-3 muestra el RMSE alcanzado para umbrales de estudio utilizando el Método 1 y 2 respectivamente y la Tabla 4-3 muestra el RMSE promedio de los treinta días.



(a) Modelo temporada verano.



(b) Modelo temporada Invernal Fuerte.



(c) Modelo temporada Invernal Ligera.

Gráfico 10-3: RMSE de los modelos de la presa según las temporadas.

Realizado por: Gabriel Asqui. 2016

Tabla 3-3: RMSE promedio de los modelos.

Temporada	Umbral horas	RMSE Promedio m.s.n.m	Min RMSE m.s.n.m	Max RMSE m.s.n.m	% de acierto
Verano	8	0,343295	0,0653	0,7525	80
	24	0,396470	0,0877	0,8898	73,333
	48	0,432766	0,11466	0,8463	66,667
Invernal Fuerte	8	0,446870	0,1366	1,2259	66,667
	24	0,609343	0,1237	1,8390	50
	48	0,650778	0,21114	1,4794	26,667
Invernal Ligera	8	0,634457	0,0876	1,8737	50
	24	0,663904	0,1480	1,4392	43,333
	48	0,764402	0,2413	2,4964	16,667

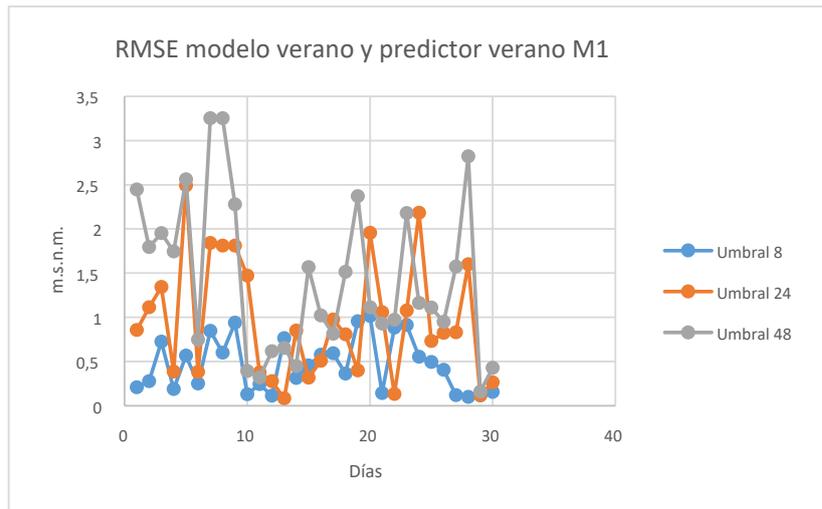
Realizado por: Gabriel Asqui. 2016

Como se puede notar en la Tabla 4-3, el error del predictor es propagado al modelo de la presa, y aumenta conforme el umbral de predicción crece lo que lleva a tener un RMSE que en el peor de los casos sobrepasa los dos (2) metros promedio.

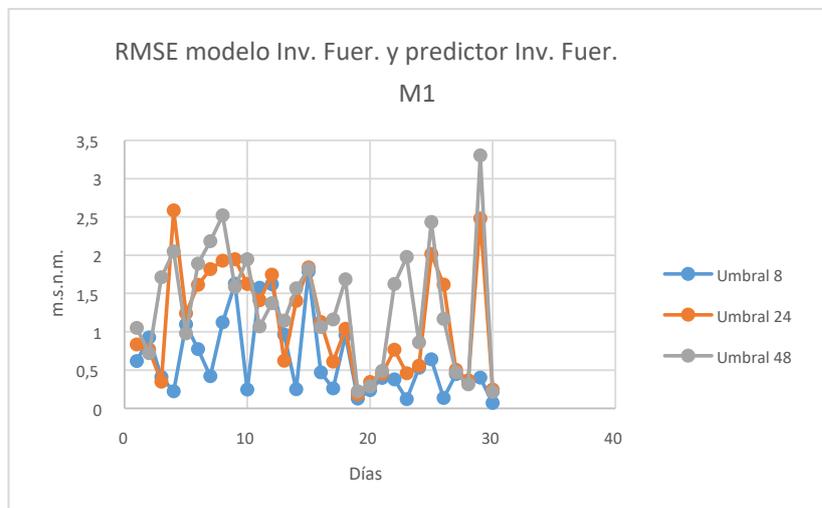
3.3 Implementación de la Plataforma.

La plataforma de software consta de seis (6) módulos: BaseDatos, Frontal, IHM, Modelo, Predictor y Usuarios, los cuales fueron desarrollados usando Python(x,y). Una mejor descripción de estos módulos se realizará a continuación:

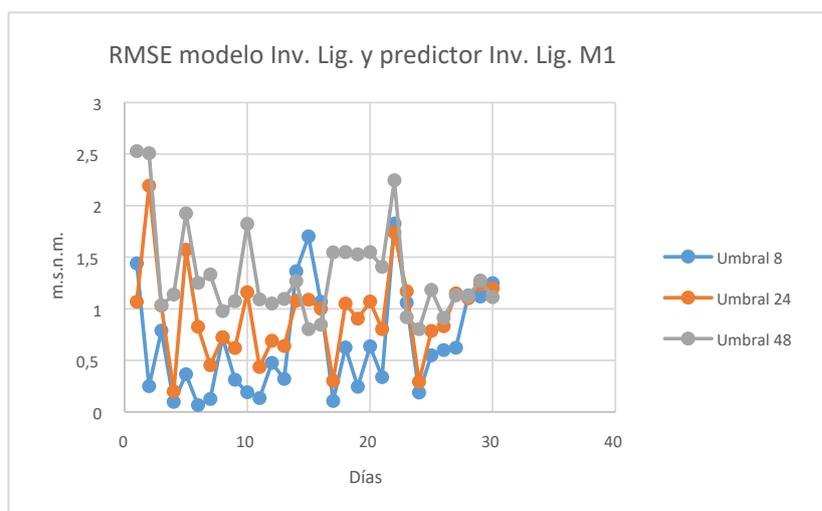
- BaseDatos.- Contiene las funciones para acceder a los archivos en formato *.xlsx y *.xlsm, como para grabar estos datos en la BDD o modificarlos.
- Frontal.- Contiene todas las funciones para los botones del panel frontal de la aplicación.
- IHM.- Contiene los diseños de las interfaces gráficas de todos los widgets.
- Modelo.- Contiene los algoritmos de entrenamiento y prueba de los modelos para la presa.
- Predictor.- Contiene los algoritmos de entrenamiento y prueba del predictor de caudal
- Usuarios.- Contiene las funciones para mantenimiento de usuarios, cifrado de claves de acceso y guardado en BDD.



(a) RMSE Temporada Verano.



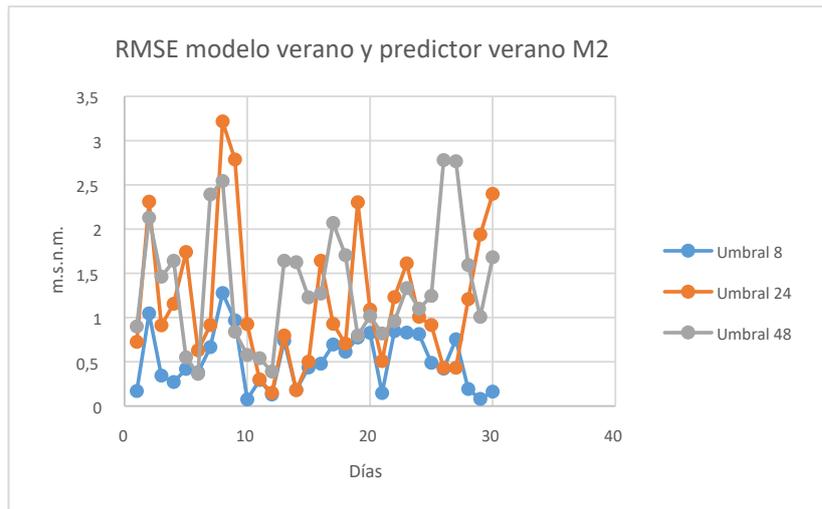
(b) RMSE Temporada Invernal Fuerte.



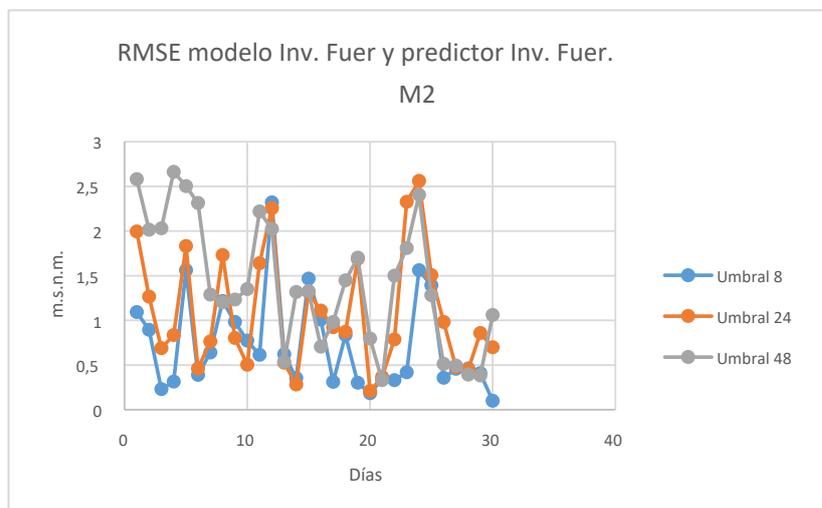
(c) RMSE Temporada Invernal Ligera.

Gráfico 11-3: RMSE del modelo de la presa sumado al predictor con método 1.

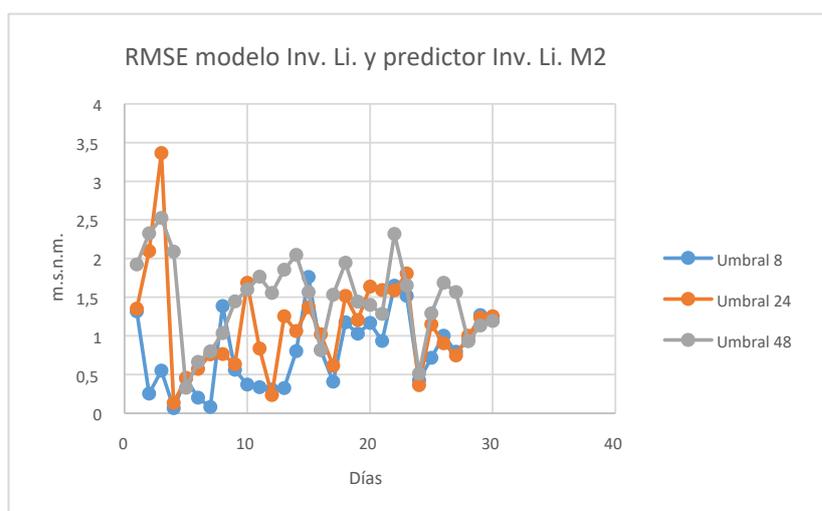
Realizado por: Gabriel Asqui. 2016



(a) RMSE Temporada Verano.



(b) RMSE Temporada Invernal Fuerte.



(c) RMSE Temporada Invernal Ligera.

Gráfico 12-3: RMSE del modelo de la presa sumado al predictor con método 2.

Realizado por: Gabriel Asqui. 2016

Tabla 4-3: RMSE promedio del modelo de la presa sumado al predictor.

Temporada	Método	Umbral horas	RMSE Promedio m.s.n.m.	Min RMSE m.s.n.m	Max RMSE m.s.n.m	% de acierto
Verano	1	8	0,466923	0,0991	1,0135	56,667
		24	0,96230	0,0824	2,4920	33,333
		48	1,437973	0,1595	3,2522	16,667
Invernal Fuerte	1	8	0,638395	0,0698	1,7876	56,667
		24	1,148365	0,1813	2,5829	23,333
		48	1,362053	0,2129	3,3035	20
Invernal Ligera	1	8	0,656023	0,0656	1,8251	46,667
		24	0,945998	0,1980	2,1924	16,667
		48	1,333420	0,8012	2,5293	0
Verano	2	8	0,516222	0,0742	1,2768	56,667
		24	1,185806	0,1488	3,2151	20
		48	1,364847	0,3639	2,7773	6,6667
Invernal Fuerte	2	8	0,73030	0,0998	2,3189	50
		24	1,089777	0,2096	2,5596	20
		48	1,412877	0,3283	2,6629	13,333
Invernal Ligera	2	8	0,792970	0,0662	1,7597	36,667
		24	1,139764	0,1315	3,3636	13,333
		48	1,473689	0,3282	2,5229	3,3333

Realizado por: Gabriel Asqui. 2016

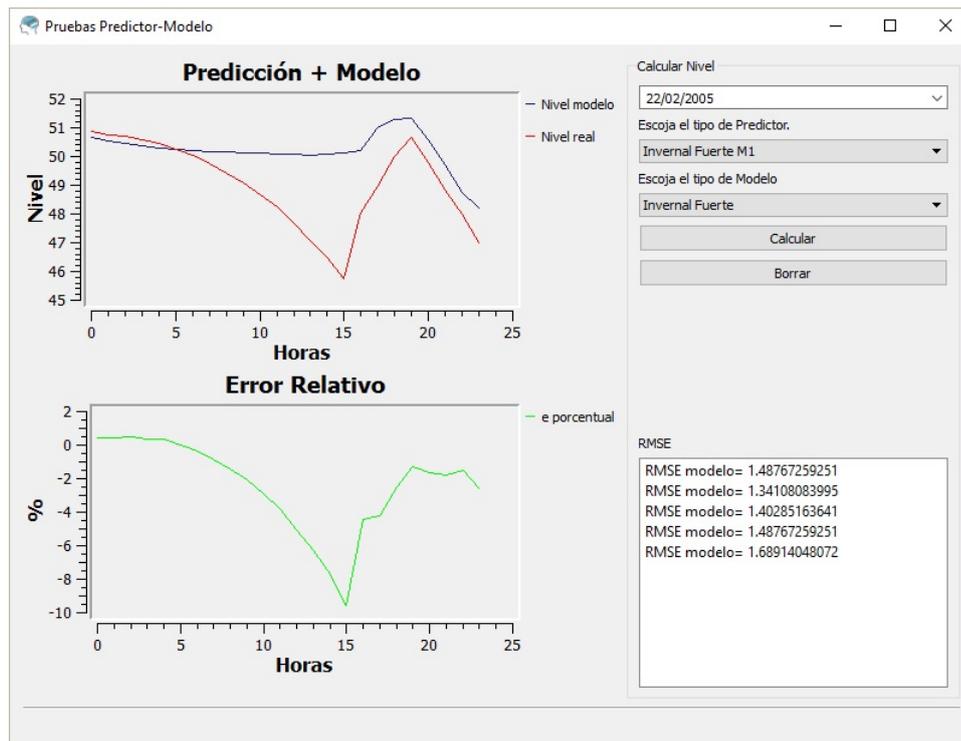


Figura 7-3: Ventana de pruebas del modelo del nivel de la presa más el predictor de caudal.

Realizado por: Gabriel Asqui. 2016

- Lanzador.- Gestiona las comunicaciones entre los distintos módulos.

3.3.1 Módulo Base de Datos.

Utiliza las librerías de Psycopg[®] y PyQt[®] para establecer comunicación con la BDD, que consta de cuatro (4) tablas:

- ag_op.- Guarda los datos de fecha, hora, nivel de embalse, caudal, potencia activa total, potencia reactiva total, potencia activa por unidad y potencia reactiva por unidad.
- nets.- Guarda los datos de las redes entrenadas como son: número de entradas, neuronas de la primera capa oculta, neuronas de la segunda capa oculta, mínimo error alcanzado en el entrenamiento, neuronas de nivel, neuronas de caudal y neuronas de potencia.
- pruebas.- Guarda los datos de las pruebas de entrenamiento como son: tipo de modelo o predictor, número de entradas, neuronas de la primera capa oculta, neuronas de la segunda capa oculta, número de epoch de entrenamiento, mínimo error alcanzado en el entrenamiento, neuronas de nivel, neuronas de caudal y neuronas de potencia.

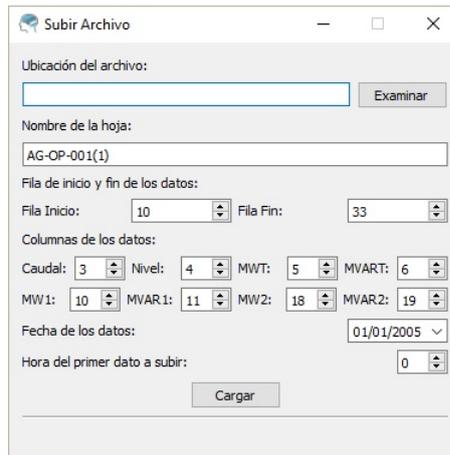


Figura 8-3: Ventana para subir los datos a la BDD.

Realizado por: Gabriel Asqui. 2016

- usuario.- Guarda el nombre de usuario y la clave de acceso para los administradores del sistema cifrados con algoritmo AES.

Psycopg[®] utiliza librerías propias, mientras que PyQt[®] utiliza dll para la comunicación con la BDD, particular que hay que tomar en cuenta ya que estos dll dependen de la arquitectura del computador en el cual se va a instalar la aplicación. Para pasar los datos de los archivos *.xlsx y *.xlsm, se implementa la pantalla de la Fig. 8-3 en donde se escoge: nombre de archivo, hoja, fecha y hora del primer dato, hora del último dato, así como las columnas de donde se encuentran los distintos datos.

3.3.2 Módulo Frontal.

Se implementa los algoritmos que controlan los botones de las cuatro (4) pestañas de la pantalla principal mostrada en la Fig. 9-3 y cuyas funciones se describen a continuación:

- Pestaña Caudales.- Pone en marcha el predictor escogido y lanza una predicción para las próximas horas según el umbral escogido y la muestra en forma de gráfico.
- Pestaña Embalse.- Permite al operador escoger: la potencia activa deseada, el predictor y el modelo a utilizar; luego de lo cual lanza una predicción del nivel de embalse con el umbral escogido basándose en los datos arrojados por el módulo predictor y modelo. Los resultados se muestran en forma de gráfica.

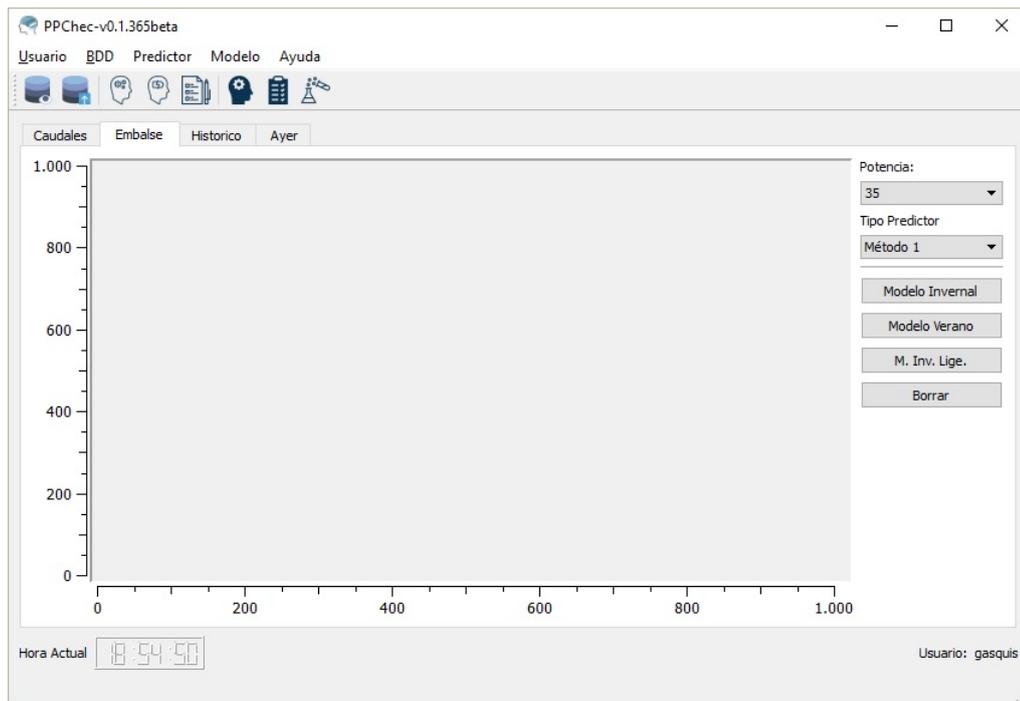


Figura 9-3: Ventana principal frontal con las distintas lengüetas.

Realizado por: Gabriel Asqui. 2016

- Pestaña Histórico.- Muestra los datos históricos anuales de potencia activa, nivel o caudal a conveniencia del operador.
- Pestaña Ayer.- Toma los datos históricos y les resta veinticuatro (24) datos, con estos datos disminuidos se lanza una predicción de caudal o nivel, y se la compara con los veinticuatro (24) datos disminuidos que pasan a constituir los datos reales. Así se puede comparar los datos calculados frente a los reales, de un día anterior al último dato ingresado en la BDD.

3.3.3 Módulo IHM.

Guarda los diseños de las distintas pantallas en formato *.ui, dado por el programa Qt Designer, así como los distintos archivos generados por el script pyuic4 que convierte el formato *.ui a *.py, y cuya conversión es necesaria para poder ser llamadas por un lanzador de formato *.pyw. dentro de la aplicación.

También guarda los íconos de la ventana frontal, y el archivo para enlazarlos con la aplicación con extensión *.qrc, el cual es convertido a *.py con el script pyrcc4.

3.3.4 *Módulo Modelo.*

Implementa los algoritmos y funciones descritos en el capítulo dos, en los archivos `fun-Model.py` y `model.py` escritos en Python[®] puro con excepción de las librerías de NeuroLab[®] optimizadas con `scipy` que utiliza librerías implementadas en C para conseguir mejores velocidades de procesamiento.

Además, se conecta a la BDD obteniendo los datos necesarios para luego de normalizarlos, formar las ventanas de entradas y objetivos para el entrenamiento y pruebas de la RNA que se implantara como modelo.

3.3.5 *Módulo Predictor.*

Se conecta a la BDD, para obtener los datos para la predicción, los cuales primero son transformados y normalizados, para luego ser divididos y formar las ventanas de entradas y objetivos tanto para el entrenamiento como para las pruebas.

3.3.6 *Módulo Usuarios.*

Contiene las funciones que permiten primero verificar que el usuario a crear no se encuentre en la BDD, segundo cifrar la clave de acceso y guardarla. De esta forma evitar que, aun ingresando a la BDD, se sepa cual es la clave de los administradores y tercero encargarse de la autenticación de los usuarios y desbloqueo de los menús y botones que dan acceso total de la aplicación.

La plataforma de hardware consta de una PC core i7 con sistema operativo Windows 7[®].

En el anexo A se pueden apreciar los distintos diagramas de flujo generales implementados sin embargo si se desea mas detalle se puede revisar el código de la aplicación libre que esta bajo licencia GPLv3. y se puede consultar el manual de usuario.

CAPÍTULO IV

4 . RESULTADOS.

Como se mostró en el Capítulo 3 en la Tabla 4-3, los resultados en pruebas fuera de línea del predictor de caudal junto al modelo de la presa, fueron aceptables en un umbral de ocho (8) horas, disminuyendo su eficacia a medida que el umbral crecía.

Esta sección toma los resultados del capítulo 3 como referencia para la puesta en producción del sistema y muestra los resultados obtenidos luego de su implantación y la incidencia en los índices de producción que se mostraron en el gráfico 4-3 y que corresponden a los cambios de producción y reprogramaciones antes de la implantación del sistema.

4.1 Resultados y Discusión.

Es importante mencionar que, la meta de umbral planteado dentro de las hipótesis al inicio de este trabajo fue de 24 horas. Sin embargo, se demostró que dicho umbral para el caso de los predictores de caudal solo generaba el 20% de aciertos para el Método 1 y 6.66% para el Método 2 (véase Tabla 2-3), en la temporada invernal ligera donde se llevó a cabo las pruebas y un 43.33% de aciertos para el caso del modelo de la presa (véase Tabla 3-3).

Debido a esto, se analizaron casos con umbrales de menor valor (8 horas). De esta manera, los mejores modelos de presa y predictor con umbrales de ocho (8) horas se enviaron a producción en la Central Hidroeléctrica Agoyán durante veintidós (22) días (desde el 1/09/2016 hasta el 22/09/2016), proporcionando los resultados mostrados en la Tabla 1-4, en la cual se muestran también los resultados utilizando el modelo invernal ligera y verano para los mismos datos.

Se justifican las pruebas con el predictor de caudal de temporada verano debido a que los caudales estaban por debajo de los $120 \text{ m}^3/\text{s}$ en los días de prueba y que caracterizan a la temporada verano y no a la temporada invernal ligera. Nótese en la Tabla 1-4 que

Tabla 1-4: RMSE y porcentaje de acierto del predictor de caudal.

Temporada	Método	Umbral horas	RMSE Promedio m^3/s	RMSE mínimo m^3/s	RMSE máximo m^3/s	% de acierto
Invernal	1	8	11,6818	3,2472	28,0660	31,8182
Ligera	2	8	20,3372	4,6773	60,0067	13,6364
Verano	1	8	9,9671	1,2864	25,7748	45,4545
	2	8	15,4519	4,0203	52,8206	36,3636

Realizado por: Gabriel Asqui. 2016

el RMSE decrece en tanto que el porcentaje de aciertos aumenta utilizando el predictor de caudal de temporada verano y que el método con mejor comportamiento para nuestro caso es el Método 1.

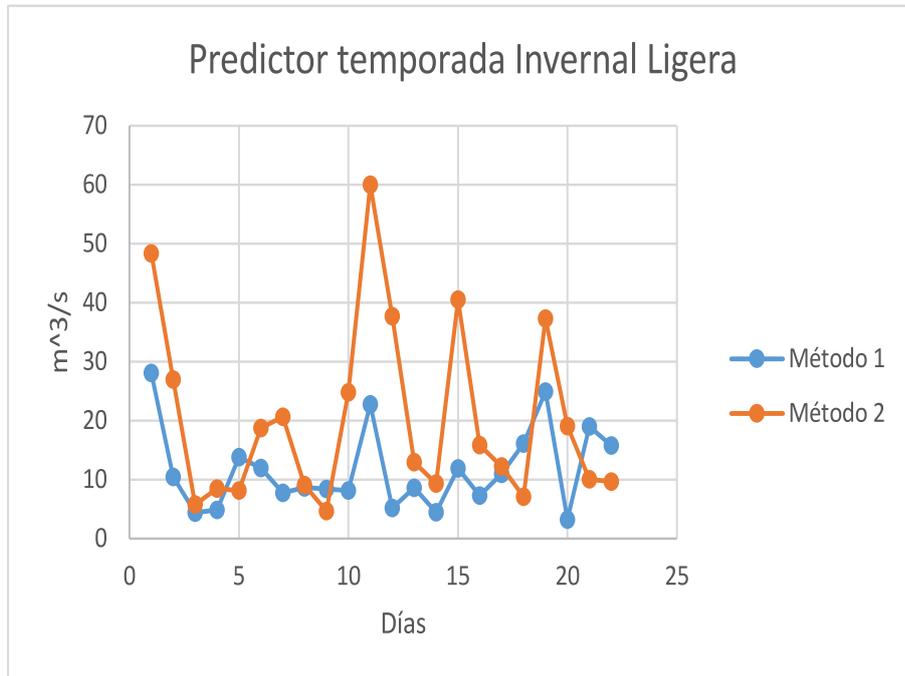
El gráfico 1-4 muestra el RMSE diario alcanzado por el predictor de caudal en los veintidós (22) días de prueba donde se nota claramente la diferencia de precisión entre los dos métodos de predicción.

La Fig. 1-4 muestra un ejemplo del mejor y peor caso predicho versus los datos reales, con su respectivo error usando el predictor de caudal de temporada invernal ligera, obteniéndose un RMSE de $3.24 m^3/s$ en el mejor de los casos y un RMSE de $28.06 m^3/s$ en el peor de los casos.

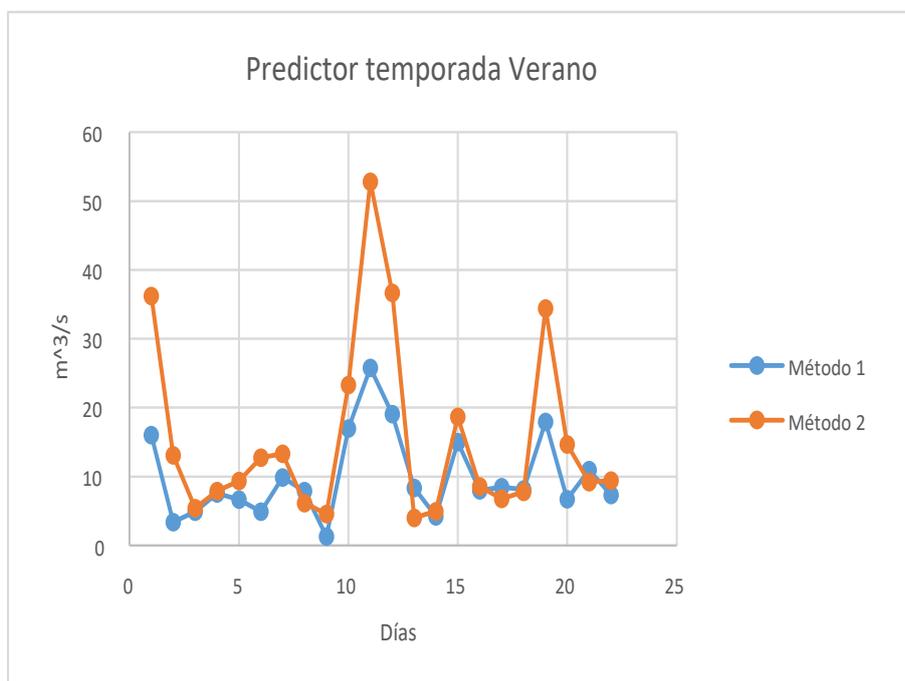
Como se vio en el capítulo tres, se obtuvo predictores para la temporada invernal fuerte, invernal ligera y verano, siendo esta última una de las más importantes para la central Hidroeléctrica Agoyán debido a que el comportamiento del río Pastaza es difícil de predecir en esta temporada, mientras que en las temporadas invernales los caudales se mantienen superiores a los $120m^3/s$ necesarios para obtener la máxima producción y la planificación de producción es fácilmente predecible.

Para el caso de la predicción de nivel del embalse el porcentaje de aciertos en los veintidós (22) días de prueba se muestran en la Tabla 2-4, donde nuevamente para nuestro caso el Método 1 tiene la mejor respuesta alcanzando el 86.3636% con el modelo de presa de temporada verano junto al predictor de caudal correspondiente a la misma temporada.

El gráfico del RMSE diario del predictor de nivel se lo puede ver en el gráfico 2-4, en



(a) RMSE diario utilizando el predictor Invernal Ligero.

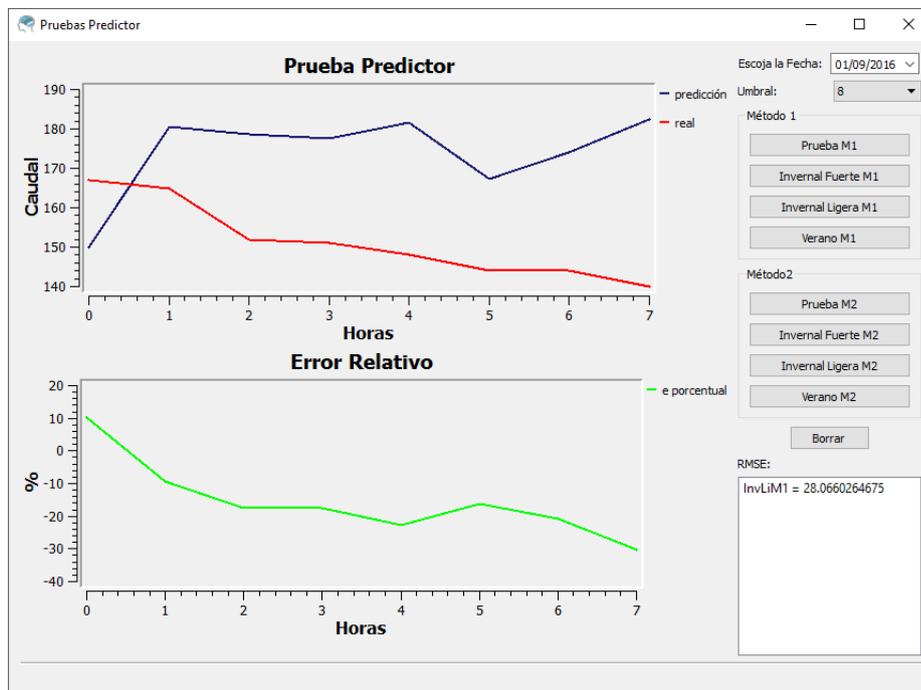


(b) RMSE diario utilizando el predictor Verano.

Gráfico 1-4: RMSE diario del predictor de caudal.
Realizado por: Gabriel Asqui. 2016



(a) Mejor caso utilizando el predictor de temporada invernal ligera método 1.



(b) Peor caso utilizando el predictor de temporada invernal ligera método 1.

Figura 1-4: Predicción de caudal con umbral de ocho horas.
Realizado por: Gabriel Asqui. 2016

Tabla 2-4: RMSE y porcentaje de acierto del predictor de nivel.

Temporada	Método	Umbral horas	RMSE Promedio m.s.n.m	RMSE mínimo m.s.n.m	RMSE máximo m.s.n.m	% de acierto
Invernal	1	8	0,2540	0,0414	1,0099	81,8182
Ligera	2	8	0,6653	0,1111	2,1662	59,0909
Verano	1	8	0,2735	0,0389	1,2878	86,3636
	2	8	0,4112	0,1075	2,0242	72,7273

Realizado por: Gabriel Asqui. 2016

donde se aprecia de manera gráfica las diferencias de precisión entre los dos métodos de predicción.

Un ejemplo del peor y mejor caso para el predictor de nivel utilizando el Método 1 y la temporada invernal ligera se lo puede ver en la Fig. 2-4, en donde se muestra el nivel predicho versus el nivel real, obteniendo en el mejor caso un RMSE de 0.0454 m.s.n.m. y en el peor caso un RMSE de 1.009 m.s.n.m.

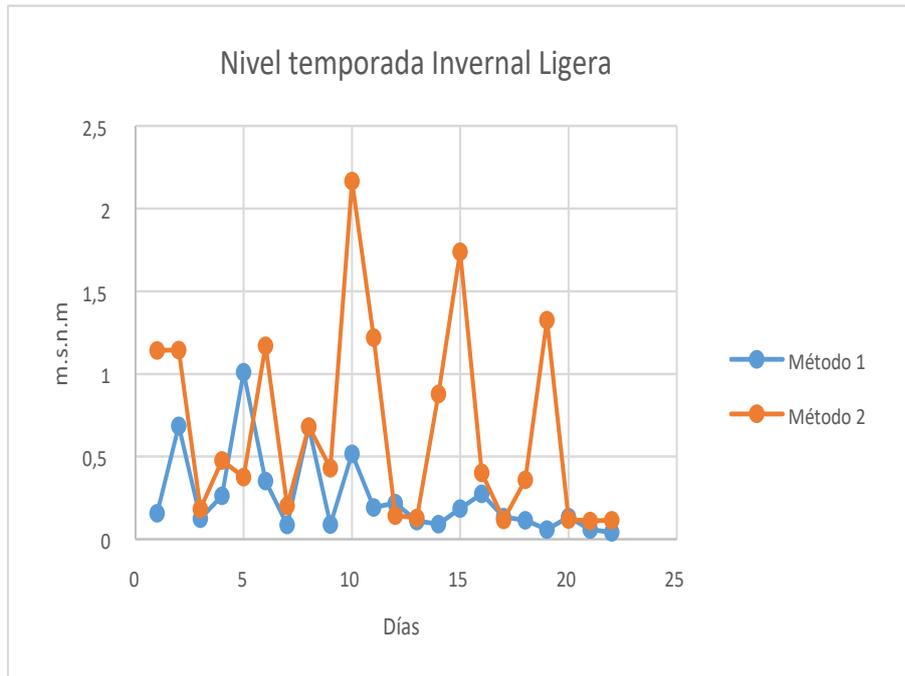
4.1.1 Docimasia de hipótesis

Al inicio de la investigación se planteó la hipótesis alternativa (H_a) que indicaba que la adecuada selección de la técnica de predicción basada en redes neuronales y aplicada a la predicción de nivel de agua del embalse permitiría optimizar la planificación de producción de energía. Por otro lado, la hipótesis nula (H_0), sería la contradicción de H_a .

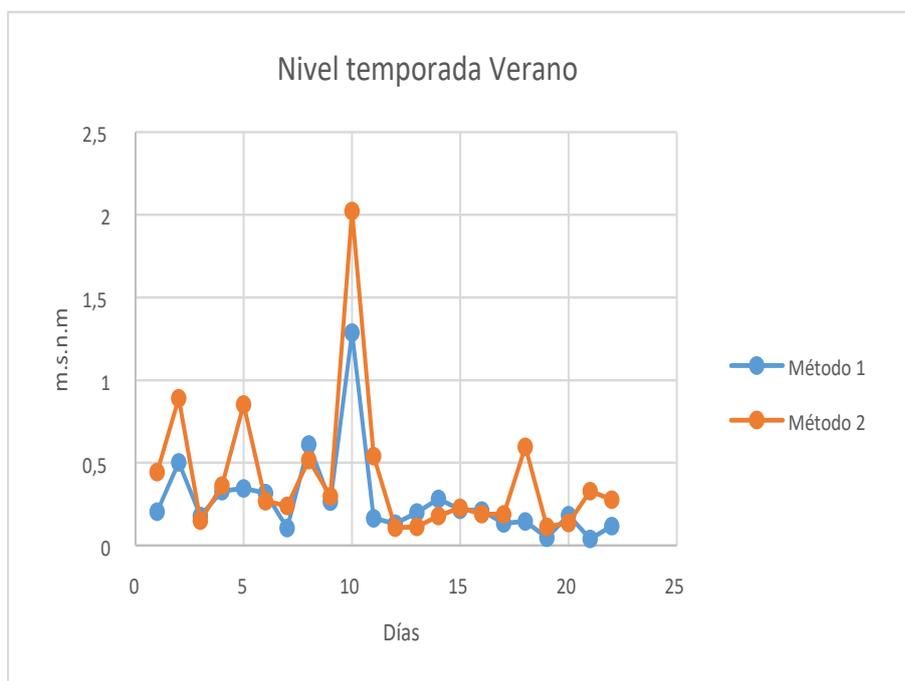
Por lo que se escoge dos indicadores que son el número de reprogramaciones y el número de cambios de producción semanales, para establecer una comparación de la influencia del sistema predicción en la planificación de producción, obteniéndose como parámetros de la hipótesis nula:

$$H_0 : u_1 = 22,06$$

$$H_0 : u_2 = 14,78$$

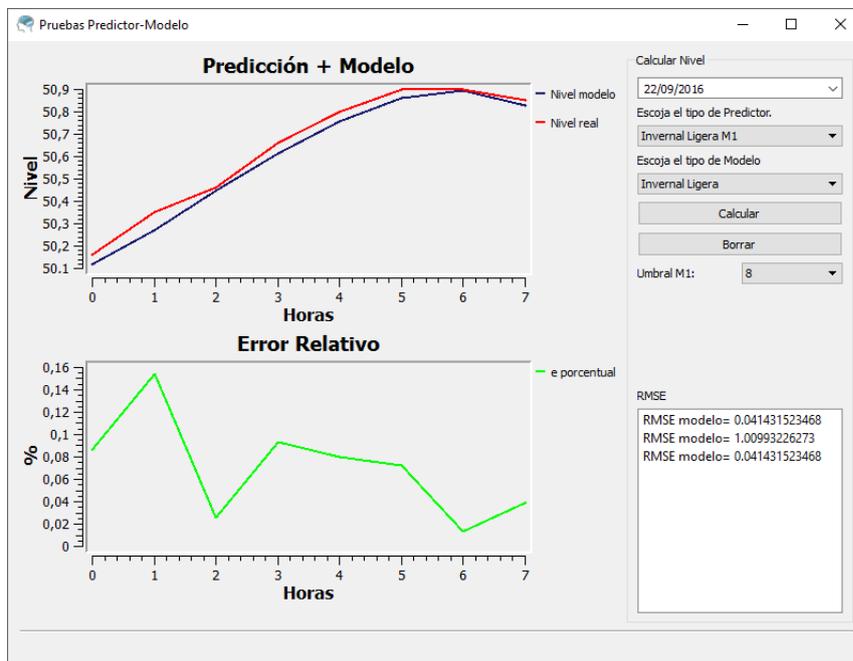


(a) RMSE diario de nivel con predictor invernal Ligero.

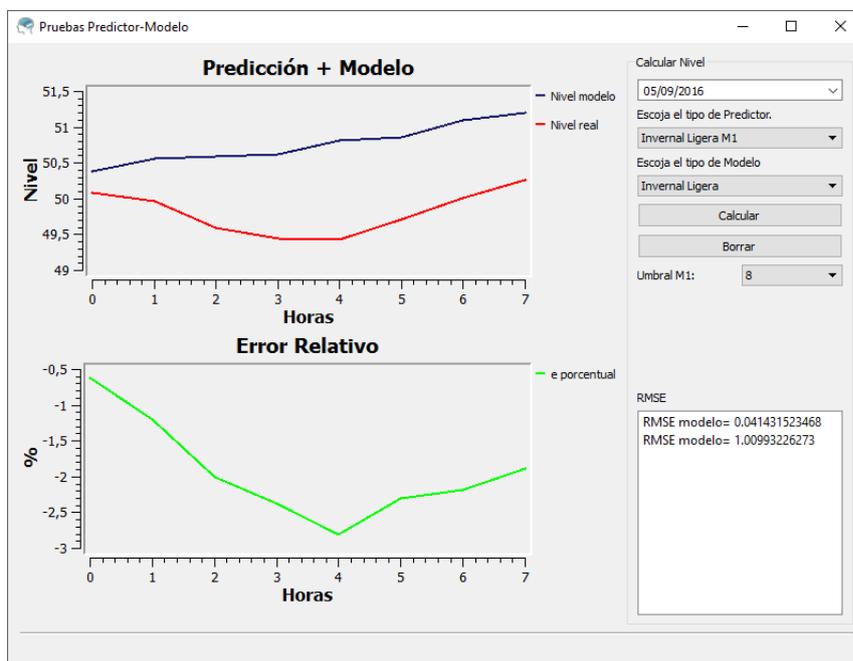


(b) RMSE diario de nivel con predictor Verano.

Gráfico 2-4: RMSE diario del predictor de nivel.
Realizado por: Gabriel Asqui. 2016



(a) Mejor caso utilizando predictor de nivel invernal ligero método 1.



(b) Peor caso utilizando predictor de nivel invernal ligero método 1.

Figura 2-4: Predicción de nivel con umbral de ocho horas.
Realizado por: Gabriel Asqui. 2016

Donde:

u_1 : Promedio de cambios de producción semanales antes de la implantación del sistema.

u_2 : Promedio de reprogramaciones semanales antes de la implantación del sistema.

Para que la hipótesis alternativa sea aceptada se tendría que cumplir que:

$$H_a : u_1 < 20,495$$

$$H_a : u_2 < 9,75$$

Dejándonos una zona de rechazo de hipótesis de cola inferior.

Se establece un nivel de significación del 1% es decir $\alpha = 0,001$, para minimizar la probabilidad de cometer un error de tipo I es decir descartar la hipótesis nula cuando esta es verdadera y se calcula la región de rechazo que es el inverso de distribución normal estándar evaluada en α ; dando como resultado $t_\alpha \leq -2,33$.

Luego de la implantación del sistema se obtienen los promedios indicados en la Tabla 3-4. Siendo la cuasi desviación estándar de la muestra $S_1 = 9,50$ para los cambios de producción y $S_2 = 6,65$ para las reprogramaciones, se procede a calcular el estadístico de contraste t Student, usando la ecuación 4.1, debido a que los días que se permitió realizar las pruebas fueron pocos y la muestra es pequeña.

$$t_i = \frac{\bar{X} - u_i}{\frac{S}{\sqrt{n}}} \quad (4.1)$$

Donde:

\bar{X} : Promedio de la muestra.

u_i : Parámetro de la hipótesis nula.

S : Cuasi desviación estándar de la muestra.

Tabla 3-4: Reprogramaciones y cambios de producción antes y después de la implantación del sistema para la temporada verano.

	Promedio cambios de producción semanales \bar{X}_1	Promedio reprogramaciones semanales \bar{X}_2
Antes de la implantación.	22,06	14,78
Luego de la implantación.	30,66	16.33

Realizado por: Gabriel Asqui. 2016

n : Número de muestras.

Resultando:

$$t_1 = 1,85 \rightarrow t_1 \not\leq -2,33$$

$$t_2 = 1,71 \rightarrow t_2 \not\leq -2,33$$

Por lo tanto, no se puede rechazar la hipótesis nula o lo que es lo mismo aceptar la hipótesis alternativa ya que el estadístico de contraste no recayó en la región de rechazo y no existió una disminución en ninguno de los dos indicadores. Esto es comprensible ya que el umbral efectivo es de ocho horas lo que implica que mínimo se hará un (1) cambio cada ocho (8) horas pudiendo existir hasta tres (3) cambios en veinticuatro (24) horas dependiendo de la situación.

Es menester entonces la implementación de sensores en los afluentes del río Pastaza que ayuden con la disminución del error en la predicción del caudal, así como en la ampliación del umbral de dicha predicción, de esta forma se influiría de manera positiva en la predicción del nivel y en la disminución de los indicadores de producción.

Sin embargo, es importante observar que, aunque el sistema de predicción desarrollado no contribuyo de manera particular a la mejora de los indicadores, si evidencio que los procesos realizados al interior de la Central Hidroeléctrica Agoyán, para la programación de producción energética, son los más adecuados.

Otro aspecto positivo es la línea base que este proyecto marca en la industria de la producción energética al interior de la central dejando como resultado una plataforma tecnológica que de soporte para la toma de decisiones.

Por último se puede rescatar que ahora se cuenta con un simulador que permite calcular la variación de nivel según la potencia escogida a generar, además de un entrenador de redes neuronales y una BDD propia.

El hecho de que los datos de caudal sean muy dispersos en las temporadas invernales, con datos mínimos y máximos muy alejados, hace que al momento de transformarlos y normalizarlos se pierda precisión, por lo que se podría limitar el rango de la señal de caudal ya que para el caso de la central Agoyán cuando los caudales superan los $120 \text{ m}^3/\text{s}$ ya no interesa saber la forma del caudal pues la central produce a plena carga.

Si bien el modelo NARMA nos dio una respuesta aceptable en el umbral de ocho horas frente a los datos reales, queda abierta a discusión la implementación de modelos NARMAX.

El análisis de PSD de la señal, fue primordial a la hora de escoger el número de neuronas de entrada y capas ocultas de la red. Sin embargo, queda abierta a investigación nuevas formas de determinar con mayor precisión estos parámetros.

Hay que tener en cuenta que para el caso de las centrales Hidroeléctricas no solo se debe buscar predecir la tendencia del caudal si no que la forma del caudal predicho se parezca al del real, sobre todo en épocas de sequía, ya que esto puede ser la diferencia entre aprovechar el recurso agua o desperdiciarlo.

La implementación de software libre es primordial para el desarrollo de herramientas que impulsen el desarrollo del país, sin tener que estar ligado a las ataduras del software de pago y que al momento se encuentran a la par con el software libre.

CONCLUSIONES.

- El estudio del arte de las técnicas de predicción basadas en la red perceptrón multicapa ayudó a la selección de la metodología adecuada para la predicción de caudal y nivel del embalse de agua para la producción de energía en la central hidroeléctrica Agoyán.
- El análisis del PSD de la señal, permite definir de forma aproximada los parámetros del perceptrón multicapa, como son número de entradas y número de neuronas por capa, para la predicción de nivel y caudal.
- La correcta preparación como aplicar la transformada logarítmica y filtrado de los datos antes del entrenamiento de una RNA, es primordial para que la red aprenda los casos adecuados.
- El modelado del sistema PRESA se basa en el modelo NARMA y debido a que los datos de salida son re-alimentados a la entrada, son propensos a aumentar su error, aunque no de forma significativa, cuando el umbral de predicción crece (véase Tabla 3-3).
- El predictor de nivel suma a su propio error el error de la predicción de caudal, por lo que el porcentaje de efectividad frente al umbral de ocho horas de predicción decrece (comparece Tabla 2-3 y Tabla 4-3).
- El predictor de caudal es efectivo en un umbral de ocho (8) horas en comparación a las veinticuatro (24) horas.
- La velocidad de entrenamiento está ligada a la cantidad de datos que se presente a la RNA y a las características de procesamiento del pc que se utilice.
- Los indicadores de reprogramaciones ni cambios de producción disminuyeron (véase Tabla 3-4) sin embargo, se tiene una plataforma base, que consta de un entrenador de redes neuronales, alimentador de BDD y simulador de nivel frente a la potencia activa a producir seleccionada por el operador.

RECOMENDACIONES.

- Debido a que los datos de caudal obtenidos se basan en los recogidos por los operadores, es necesario la implementación de un sistema de recolección de datos propio, paralelo al sistema SCADA propietario implementado en la central, que se alimente directamente de los sensores. De esta forma evitar los errores de lectura o tipeo de los datos. Se debe trabajar en la mejora de captura de datos, por parte del sensor de caudal en la cabecera del río Pastaza y en la implantación de sensores en los afluentes que permitan ampliar el umbral de predicción.
- Si bien la parte física contribuiría de gran manera al mejoramiento del sistema queda abierta la posibilidad de utilizar otro tipo de redes como las recurrentes, u otro método de preparación de datos antes de entrenar las redes.
- La aplicación presentada constituye una semilla para futuras investigaciones, ya que en el caso de aparecer nuevas metodologías en el ámbito de la predicción y modelado se puedan implementar modificando dichos módulos del sistema.
- Para mejorar el tiempo de entrenamiento se propone utilizar programación paralela utilizando GPUs.

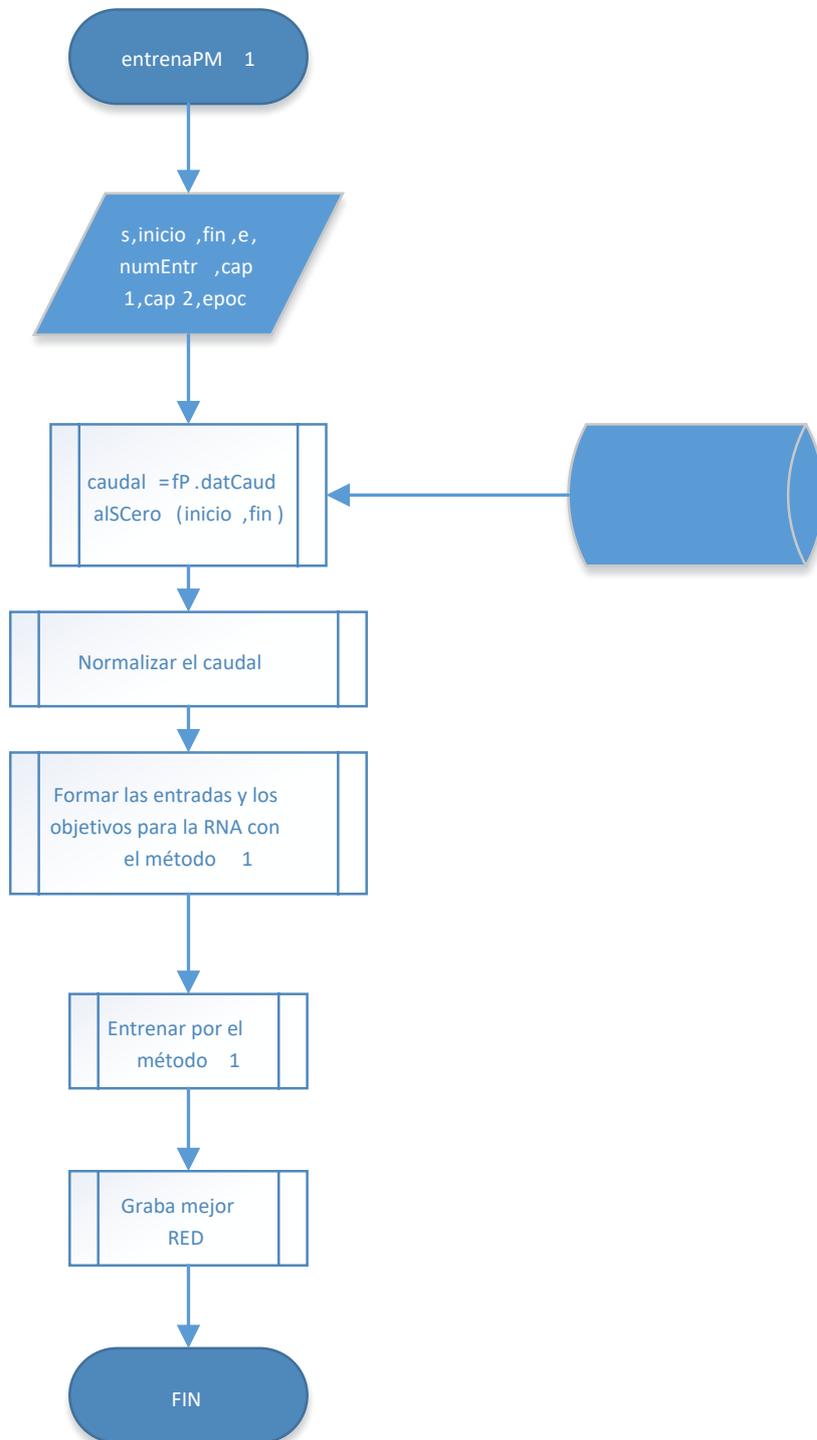
Bibliografía

- Anchaluisa, Freddy, & Hinojosa, Victor. 2008. Pronóstico de caudales de mediano y corto plazo utilizando razonamiento inductivo fuzzy y algoritmos evolutivos-aplicación para las centrales de embalse y centrales de pasada. *Revista técnica energía*, 147–154.
- Barros, José, & Mota, Alexandre. 2010. Maximum power point tracker applied in batteries charging with photovoltaic panels. *Industrial Electronics, 2008. ISIE 2008. IEEE International Symposium*, 202 – 207.
- Basogain Olabe, Xabier. 2005. Redes neuronales artificiales y sus aplicaciones. *Medicina intensiva*, **29**(1), 13–20.
- Brockwell, Peter J, & Davis, Richard A. 2002. *Introduction to time series and forecasting*. 2 edn. New York: Springer.
- CELEC EP. 2016. *Informe rendicion de cuentas 2015*. Tech. rept. CELEC EP, QUITO.
- Chen, S., Billings, S. A., & Grant, P. M. 1990. Non-linear system identification using neural networks. *International Journal of Control*, **51**, 1191–1214.
- Garrido, Santiago. 1999. Identificación , estimación y control de sistemas no-lineales mediante RGO. *Control*.
- Ghosh, Arka, & Mriganka, Chakraborty. 2012. Hybrid optimized back propagation learning algorithm for multi-layer perceptron. *International Journal of Computer Applications*, **57**(December), 1–6.
- Hernández, Gonzalo. 2006. Métodos clásicos de optimización para problemas no-lineales sin restricciones. 1–14.
- Isasi Viñuela, Pedro, & Galván León, Inés. 2004. *Redes de neuronas artificiales: un enfoque práctico*. Madrid: PEARSON EDUCACIÓN S. A.
- Izaurieta, Fernando, & Saavedra, Carlos. 1999. Redes neuronales artificiales. *Charlas de física*, 1–15.

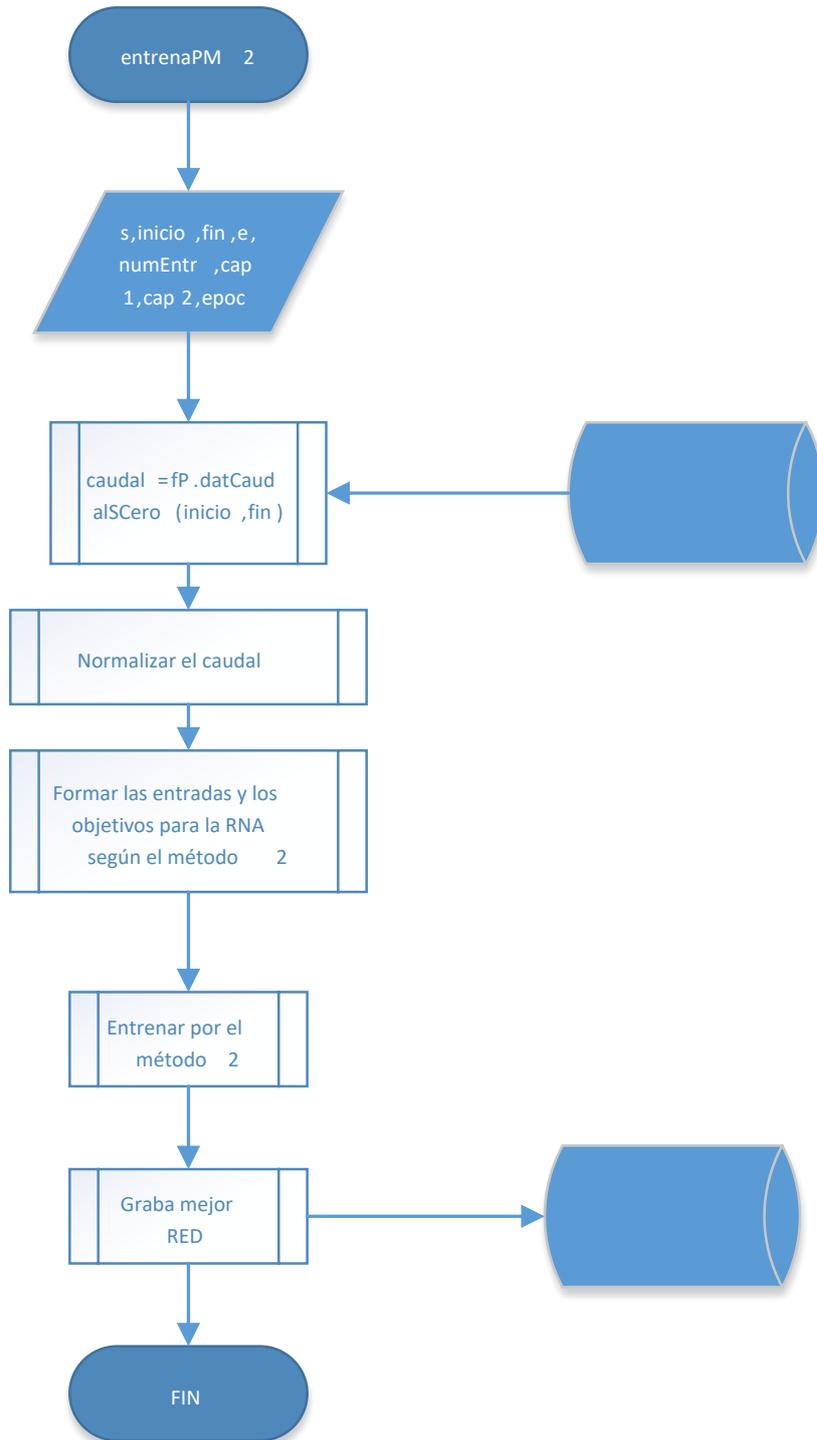
- Joeques, Silvia, Barbosa, Emanuel, & Robledo, Walter. 2005. Modelado y pronóstico de una serie de tiempo contaminada empleando redes neuronales y procedimientos estadísticos tradicionales. *Revista de la sociedad Argentina de estadística*, **9**, 1–20.
- Pérez, J. 2002. *Modelos predictivos basados en redes neuronales recurrentes de tiempo discreto*. Tesis doctoral, Universidad de Alicante.
- Ponce, Pedro. 2010. *Inteligencia artificial con aplicaciones a la ingeniería*. Primera edn. México: Alfaomega Grupo Editor S. A. de C. V.
- Quichimbo, A, & Vázquez, R F. 2014. Predicción de caudales en la cabecera de la cuenca del Paute mediante el modelo DBM. 125–134.
- REN21. 2015. *Renewables 2015 global status report*. Tech. rept. REN21, Milan.
- Senplades. 2013. *Resumen plan nacional buen vivir*. Segunda edn. QUITO: Senplades.
- Veintimilla, Jaime, & Cisneros, Felipe. 2014. Redes neuronales artificiales (RNA) aplicadas en la predicción de caudales para intervalos de tiempo horarios. *Revista Tecnológica ESPOL*, **27**(Diciembre), 37–49.
- Zaldivar, J, Gutiérrez, E, & Galván, I. 2000. Forecasting high waters at Venice lagoon using chaotic time series analysis and nonlinear neural networks. *Journal of hydroinformatics*, **2**(2000), 61–84.

ANEXO A

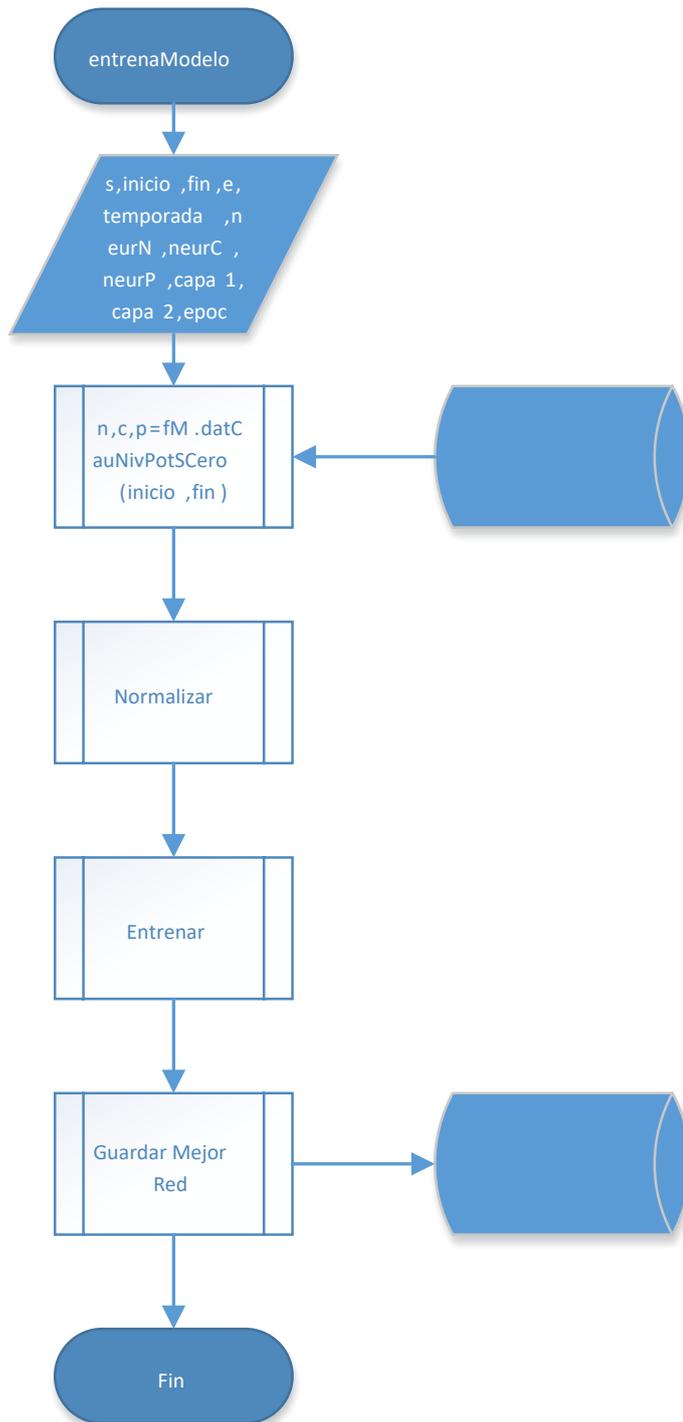
ENTRENADOR PREDICTOR MÉTODO 1.



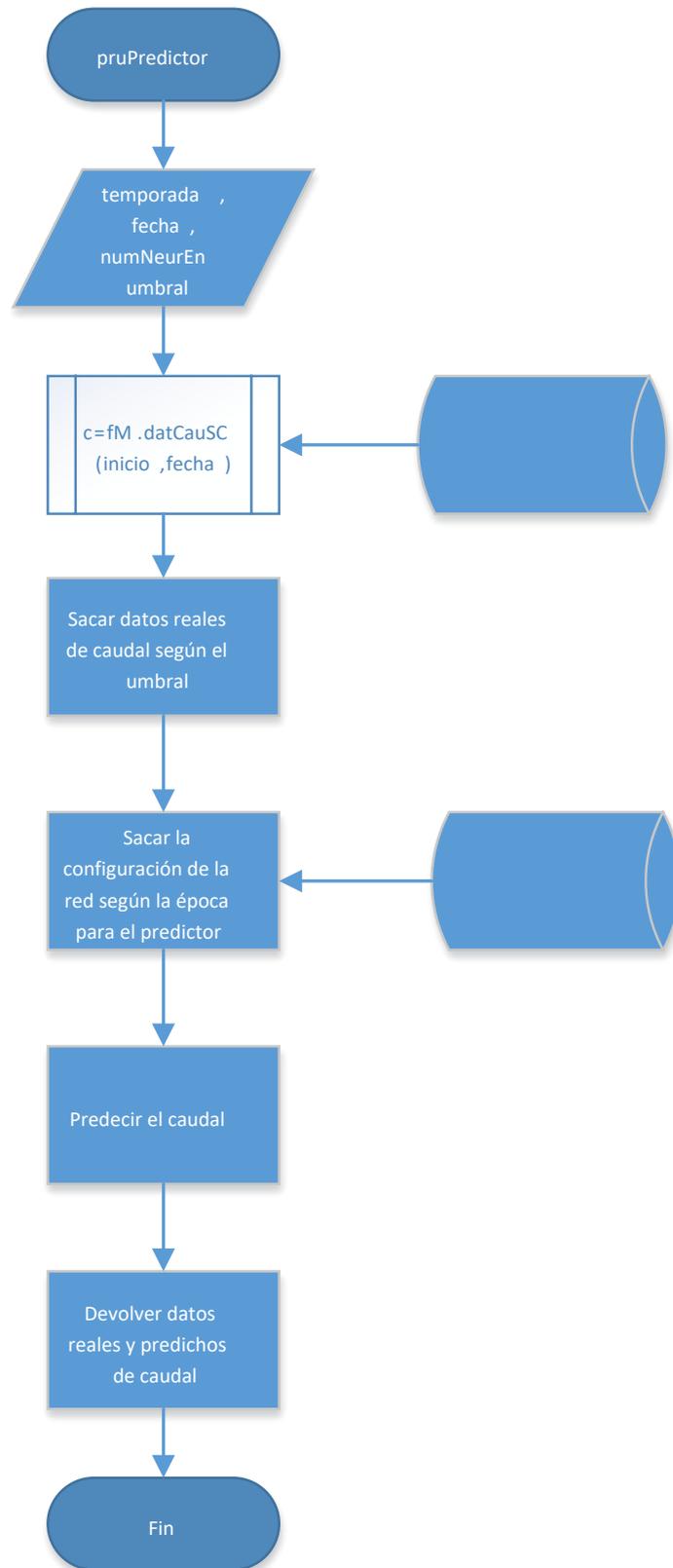
ENTRENADOR PREDICTOR MÉTODO 2.



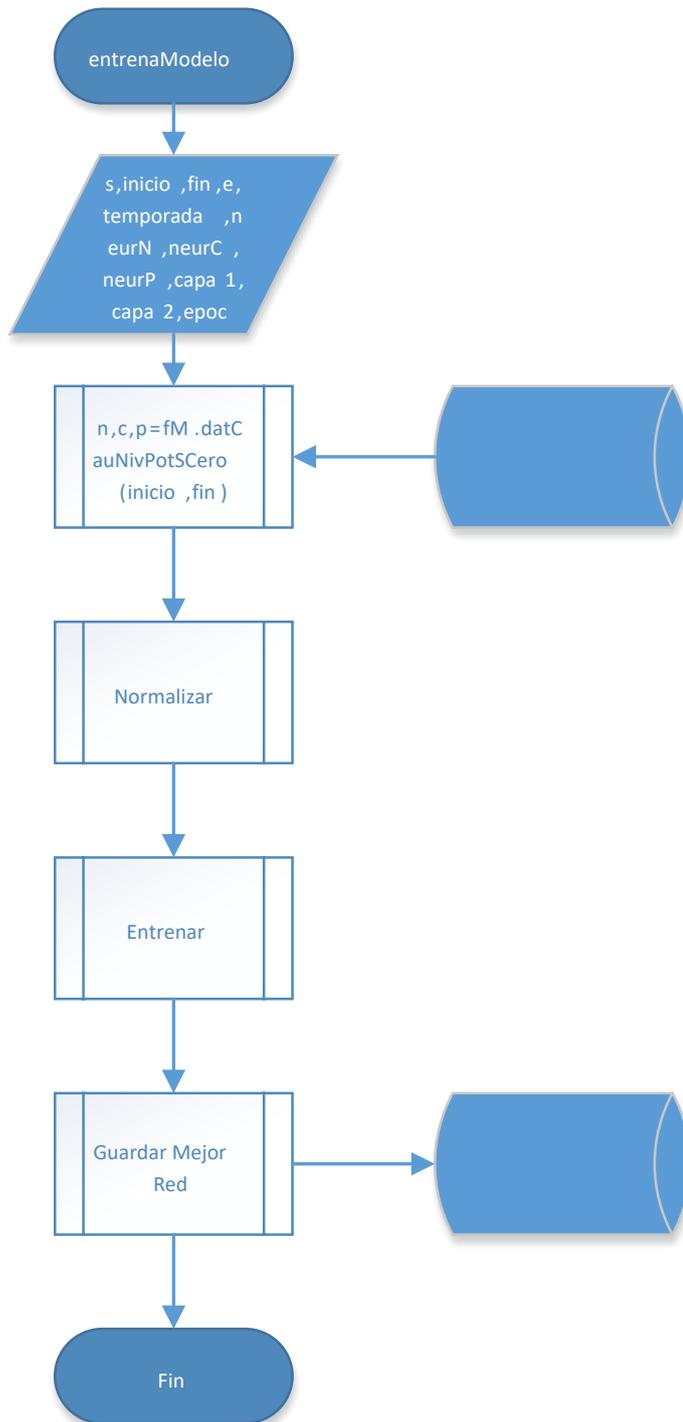
ENTRENADOR MODELO.



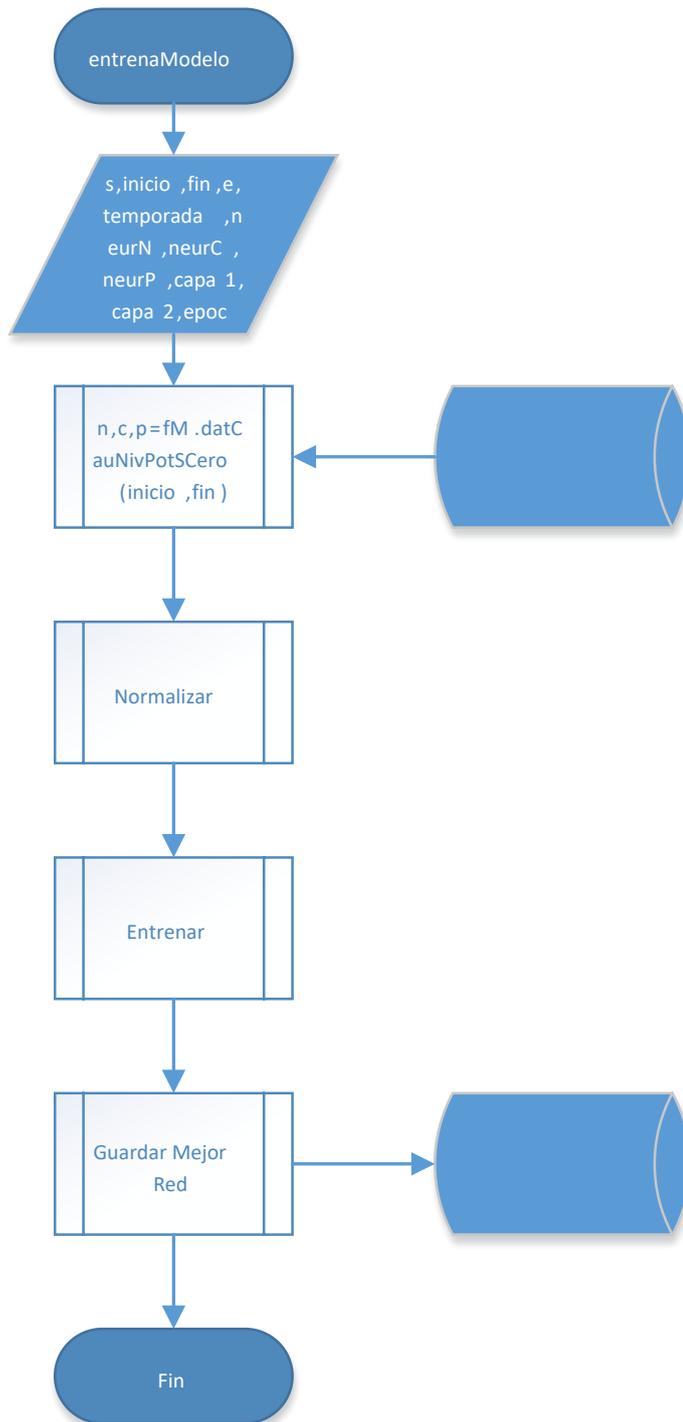
PRUEBAS PREDICTOR.



PRUEBAS MODELO.



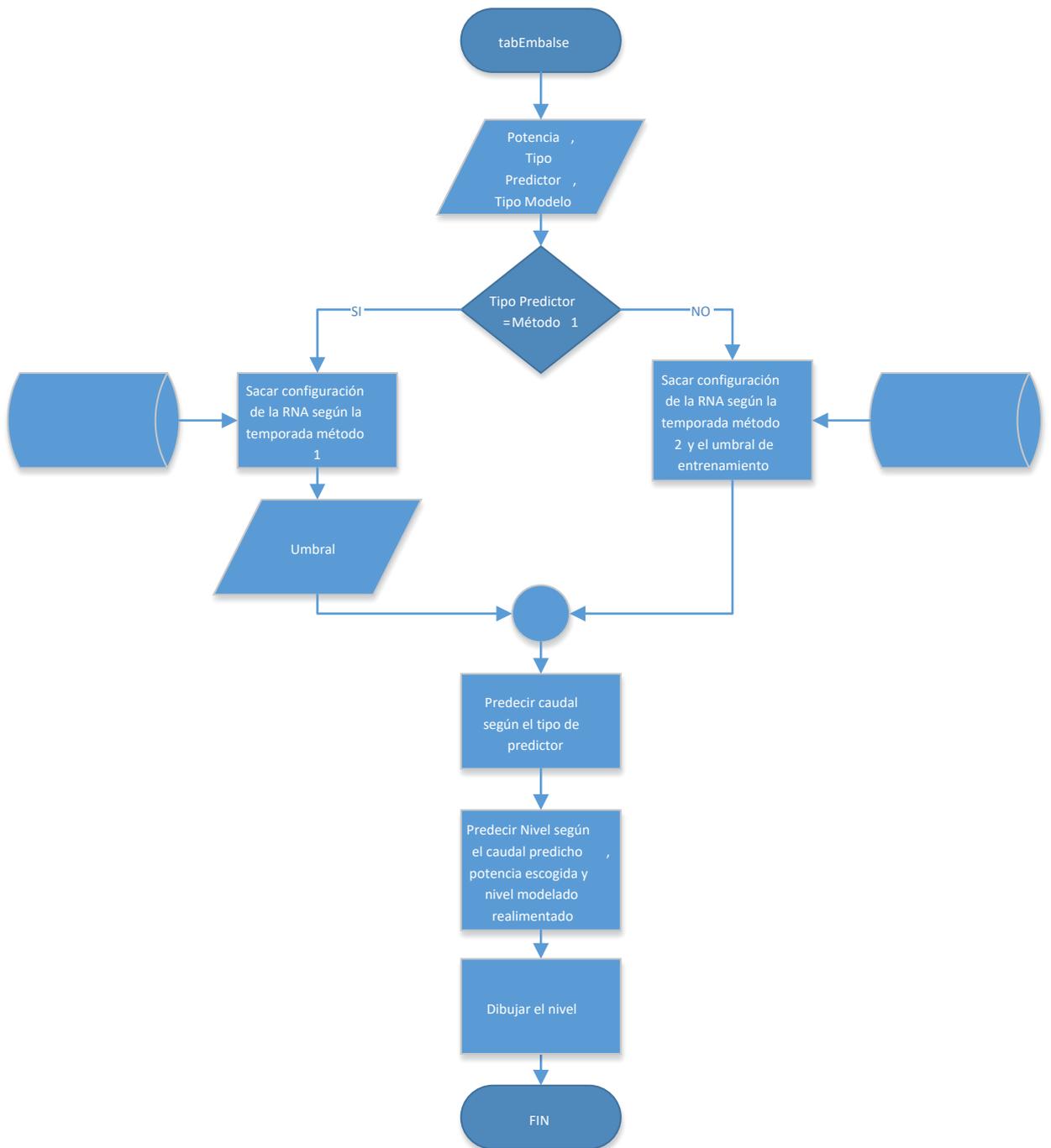
PRUEBAS PREDICTOR Y MODELO.



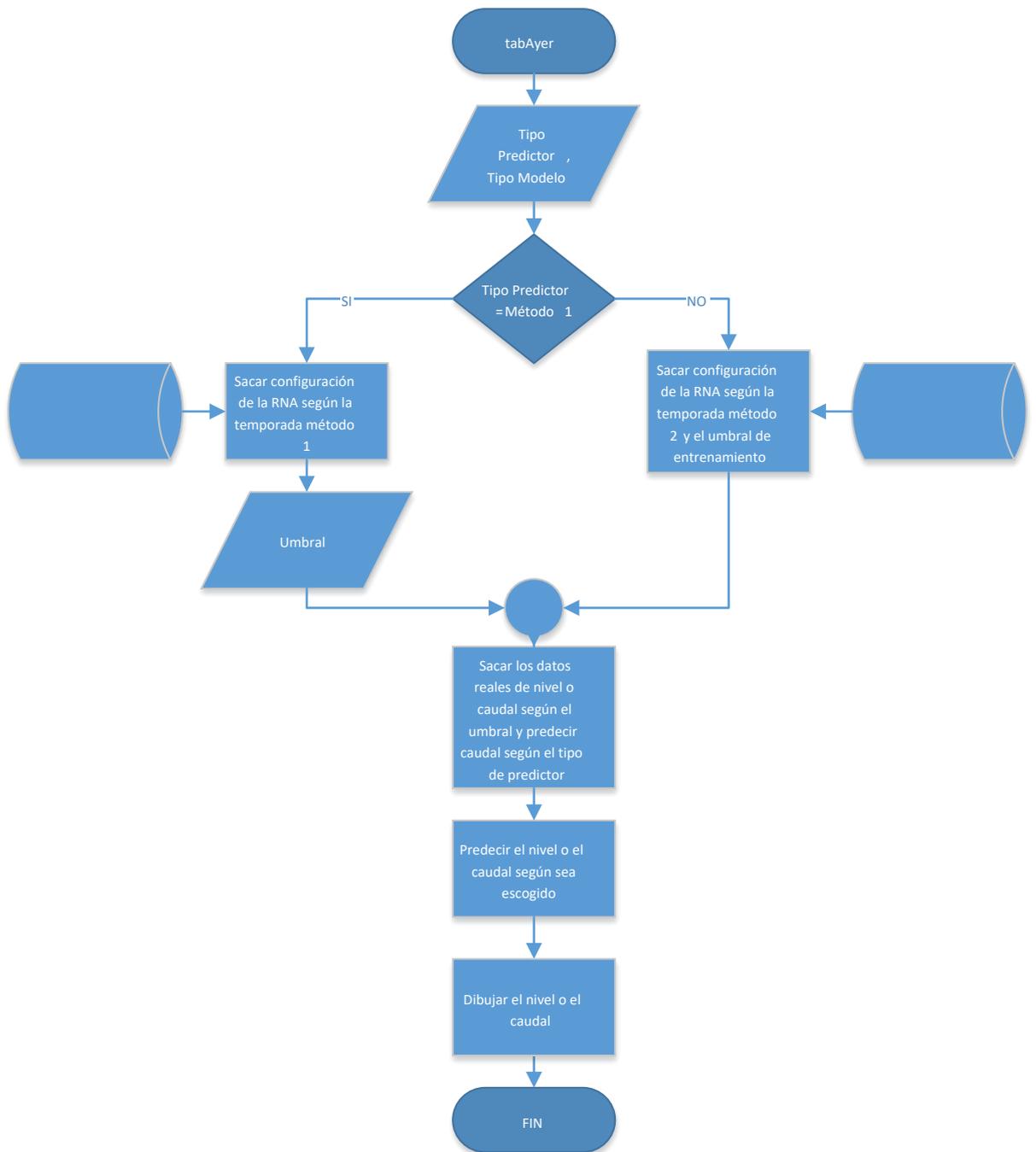
PESTAÑA CAUDAL.



PESTAÑA EMBALSE.



PESTAÑA AYER.



PESTAÑA HISTÓRICO.

