**XVIII SIMPOSIO DE INGENIERÍA ELÉCTRICA**

**Predicción de la carga eléctrica para un circuito de 34.5 kV con Redes Neuronales Artificiales**

***Electric Load Forecasting for a 34.5 kV circuit with Artificial Neuronal Networks***

**Alberto Andrés Limonte Ruíz1**

1-Alberto Andrés Limonte Ruiz. UCLV, Cuba. E-mail: limonte@uclv.edu.cu

**Resumen:**

La operación eficiente de los Sistemas Eléctricos hace imprescindible un balance; prácticamente instantáneo, entre la producción de energía eléctrica y su consumo. Resolver adecuadamente ese balance, implica poder predecir con antelación y con la mayor exactitud posible como variará la demanda de energía eléctrica, aspecto sumamente engorroso, pues la demanda muy compleja, y varía en función de variables que van desde aquellas asociadas al clima hasta eventos sociales.

El trabajo tiene como objetivo la implementación de un modelo de predicción de la demanda a corto plazo mediante Redes Neuronales Artificiales (RNA) para un circuito de subtrasmisión de 34.5 kV.

Se justifica la selección del Perceptrón Multicapa (Multiple Layer Perceptron “MLP”) con alimentación hacia delante (Feedforward) utilizando un entrenamiento supervisado con el algoritmo de propagación hacia atrás del error (Backpropagation). El modelo implementado consta de 2 capas ocultas con12 y 8 neuronas respectivamente se entrena con el 98.8% de los datos de demanda registrados en un circuito de 34.5kV de la provincia Villa Clara, dejándose los restantes para la validación del modelo.

La comparación de los valores pronosticados con los valores reales de demanda muestran errores entre 0.14% y 4.47%, aspecto perfectible si se incluyeran variables meteorológicas de cuyos datos no se dispuso

El modelo de RNA implementado en Matlab permite predecir la demanda eléctrica de un circuito de subtrasmisión de 34.5kV con errores aceptables, demostrándose la bondad de este método de inteligencia artificial para la solución de la problemática planteada

***Abstract:***

*The efficient operation of the Electrical Systems makes a balance essential;practically instantaneous, between the production of electrical energy and its consumption. Resolving this balance properly, implies being able to predict in advance and with the greatest possible accuracy how the demand for electric power will vary, an extremely cumbersome aspect, since the demand is very complex, and varies according to variables that range from those associated with climate to social events. .*

*The objective of the work is to implement a short-term demand prediction model using Artificial Neural Networks (ANN) for a sub-transmission circuit of 34.5 kV.*

*The selection of the Multilayer Perceptron (MLP) with feed forward (Feedforward) is justified using a supervised training with the algorithm of backward propagation of the error (Backpropagation). The model implemented consists of 2 hidden layers with 12 and 8 neurons, respectively, with 98.8% of the demand data recorded in a 34.5kV circuit of the Villa Clara province, leaving the remaining ones for model validation.*

*The comparison of predicted values ​​with real demand values ​​shows errors between 0.14% and 4.47%, a perfectible aspect if meteorological variables were included whose data were not available*

*The model of RNA implemented in Matlab allows predicting the electrical demand of a sub-transmission circuit of 34.5kV with acceptable errors, demonstrating the goodness of this method of artificial intelligence for the solution of the problematic raised*

 **Palabras Clave:** Inteligencia Artificial; Demanda Eléctrica; Predicción.

***Keywords:*** *Artificial intelligence; Electric Demand; Prediction.*

**1. Introducción**

A pesar de los desarrollos actuales en los medios de almacenamiento para la energía eléctrica, todavía esa energía no se puede almacenar en cantidades significativas ni por tiempos considerables; ello hace imprescindible un adecuado balance; prácticamente instantáneo, entre la producción de energía eléctrica y su consumo para garantizar una operación eficiente y eficaz de los elementos involucrados en la producción y transporte de la misma. Un problema importante para resolver adecuadamente ese balance, implica poder predecir con antelación como variará la demanda de energía eléctrica, aspecto sumamente engorroso, pues la demanda es sumamente compleja, y es afectada por variables que van desde aquellas asociadas al clima (temperatura, humedad), pasando por elementos culturales hasta eventos sociales.

La demanda de potencia y el consumo de energía eléctrica son funciones no lineales en el tiempo las cuales presentan distintos valores en los diferentes puntos geográficos de la red. Esto se debe a la naturaleza de los usuarios, a la intensidad y modos de uso de la energía [1]. Una curva característica de la variación de la demanda para un día y una región determinada se muestra en la figura 1.



Figura 1. Curva característica de demanda para 24 horas (elaboración propia)

Para la operación y planificación de la generación, transmisión y distribución de la energíaeléctrica sonesenciales modelos exactos para la predicción de la demanda (también denominada carga eléctrica), ya que esta predicción ayuda a la toma de importantes decisiones, tales como el despacho de carga entre fuentes de generación, la disponibilidad de generación o reserva rodante, la transferencia entre regiones etc., lo que redunda en una operación con mayor eficiencia y eficacia de los diferentes elementos involucrados [2]

La predicción de carga o demanda puede ser dividida en tres categorías de acuerdo con el horizonte temporal de predicción; esto es, a corto plazo, que normalmente está entre una hora y hasta una semana, a mediano plazo; la que se considera normalmente entre una semana y un año y la predicción a largo plazo la que comprende más de un año [2]

La mayoría de los métodos de predicción usan técnicas estadísticas o algoritmos de inteligencia artificial tales como regresión, redes neuronales, lógica difusa y sistemas expertos. Mientras que para la predicción a mediano y largo plazo se emplean ampliamente los métodos denominados de "uso final" y de enfoque econométrico[2]; para la predicción a corto plazo se han desarrollado una serie de métodos que incluyen los denominados modelos de regresión[3], [4],[5],[6]; series temporales[7],[8]; redes neuronales[9],[10],[11], [12],[13], [14];lógica difusa[15], [16], sistemas expertos [17,18] y combinaciones de algunos de los anteriores [19]

 Como se puede apreciar, dentro de las aplicaciones más reportadas para este fin en años recientes, las RNA ocupan un lugar importante, debido a su habilidad para aprender relaciones no lineales y complejas que son difíciles de modelar con técnicas convencionales [13]

El presente trabajo tiene como objetivo la implementación de un modelo de predicción de la demanda a corto plazo mediante Redes Neuronales Artificiales(RNA) para un circuito de subtrasmisión de 34.5 kV.

Existen muchos tipos de RNA, cada una con sus propias características; la red que mejor se ajusta a una determinada aplicación, depende de varios factores, tales como: el tipo de problema en sí mismo, tipos de variables disponibles, cantidad y calidad de los datos, algoritmo de cálculo, etc.

Una RNA simula el comportamiento del cerebro humano. La misma es un procesador masivo distribuido en paralelo formado por unidades procesadoras simples, conocidas como neuronas. Desde el punto de vista de su arquitectura consta de un número de capas que contienen neuronas y los pesos asociados con las conexiones entre dichas neuronas, y en este caso la información es transferida en la denominada alimentación hacia delante (feed-forward). En la figura 2 se muestra un modelo de una RNA. Las unidades simples denominadas neuronas constan de tres elementos básicos, a saber, los pesos de las conexiones, la función de suma de dichos pesos dentro del nodo y la función de transferencia. En el caso mostrado en la figura se tiene 1 capa de entrada, una capa oculta y una capa de salida. Una explicación más detallada de esto puede encontrarse en [20].



Figura 2 Modelo de una RNA (tomada de [13])

Una forma para escoger el tipo de red a implementar, es analizar si el problema cuenta con un conjunto de salida conocido, esta condición restringe las opciones a dos categorías, las redes con aprendizaje supervisado y las de aprendizaje no supervisado [20]

La red neuronal más frecuentemente utilizada y que mejor se ajusta al objetivo planteado es un Perceptrón Multicapa (Multiple Layer Perceptron “MLP”) con alimentación hacia delante (Feed-forward) utilizando un entrenamiento supervisado por medio del algoritmo de retro-propagación del error (Backpropagation).

Esta red de aprendizaje supervisado, emplea un ciclo propagación-adaptación de dos fases. Una vez que se ha aplicado un patrón a la entrada de la red como estímulo, este se propaga desde la primera capa a través de las capas de la red, hasta generar una salida. La señal de salida se compara con la salida deseada y se calcula una señal de error para cada una de las salidas. Las salidas de error se propagan hacia atrás, partiendo de la capa de salida, hacia todas las neuronas de la capa oculta que contribuyen directamente a la salida. Sin embargo las neuronas de la capa oculta solo reciben una fracción de la señal total del error, basándose aproximadamente en la contribución relativa que haya aportado cada neurona a la salida original. Este proceso se repite, capa por capa, hasta que todas las neuronas de la red hayan recibido una señal de error que describa su contribución relativa al error total. Basándose en la señal de error percibida, se actualizan los pesos de conexión de cada neurona, para hacer que la red converja hacia un estado que permita clasificar correctamente todos los patrones de entrenamiento.

**2. Metodología**

Para dar respuesta al objetivo planteado, primeramente se procedió al estudio de las diferentes configuraciones o tipos de RNA. Primeramente se procedió a la preparación de los datos, en este caso se disponía de los valores de demanda horaria registrada de un circuito de 34.5kV durante un año. En la figura 3 se muestran los mismos para todos los días del año en cuestión



Figura 3. Gráficos de carga de todos los días del año con datos medidos disponibles (Fuente [20])

Los mismos son normalizados dividiéndolos entre el valor de la máxima demanda registrada en el año analizado, para obtener valores entre cero y uno.

Una vez definido el tipo de RNA a implementar (Red Multicapa) alimentada hacia adelante (feed-forward) y con retro-propagación del error (backpropagation), es preciso determinar la arquitectura de la misma, es preciso aclarar en este punto que aunque en la literatura se muestran algunos métodos para ello, en este caso se decidió realizarlo por el Método de “prueba y error” con la evaluación de diferentes estructuras y su máximo error admisible. Lo que se persigue es dotar a la red de un número adecuado de neuronas en la capa oculta para que sea capaz de “aprender” las características de las posibles relaciones existentes entre los datos de la muestra. Luego de varias pruebas se observó un mejor desempeño de la red MLP con 12 neuronas en la primera capa oculta y 8 en la segunda capa oculta.

Los valores de entrada a la red se agruparon en el vector de cuatro entradas, que incluyen el mes, el día, la hora y la demanda en el instante anterior al que se desea saber el valor de consumo, para lo cual se calculó un promedio semanal por meses mostrado en la figura 4. Los meses se identifican del 1 al 12 comenzando por enero, los días de la semana del 1 al 7 con el lunes como primer día, y para las horas se asumió la 1:00 am como la hora 1 y luego por su orden hasta las 24:00 pm.



Figura 4. Gráficos de carga con promedios por semanas para los meses del año con los datos medidos disponibles (Fuente [20])

El número de neuronas para la capa de salida se seleccionó de acuerdo al horizonte de predicción. Se efectuaron varias pruebas de comparación y, según resultados analizados, se obtuvo mejores resultados con una neurona en la capa de salida.

Luego se procede a realizar el entrenamiento de las RNA con diferentes arquitecturas, para ello se emplean las facilidades que brinda el Matlab[20]

El proceso de aprendizaje no es fijo para ninguna red neuronal, el éxito consiste en probar con diferentes configuraciones hasta obtener la respuesta deseada. En este caso como ya se ha planteado se escogió una red **4:12:8:1**, es decir, que para un vector de entrada de cuatro dimensiones y esperando una sola salida de red, se tienen 12 neuronas en la primera capa oculta y 8 neuronas en la segunda capa oculta, la misma es representada esquemáticamente en la figura 5.



Figura 5. Diagrama esquemático de la red neuronal implementada (Fuente [20])

Las corridas se realizaron con un código confeccionado sobre MatLab 7.0 a partir del empleo de sus herramientas para el trabajo con las RNA. La red se crea mediante el comando **newff** [20].



Para el entrenamiento se emplearon solo el 98.8% de los valores de entrada, de los 2015 disponibles. Estos valores fueron presentados de forma aleatoria, de manera que no esquematizara comportamientos sino que fuese capaz de diferenciarlos. Los restantes valores se tomaron como prueba del rendimiento de la red o validación del modelo. Es preciso señalar que con un 70% de los valores de datos de entrada es suficiente para el entrenamiento.

Los valores de iniciación de la matriz de pesos se generaron aleatoriamente y, después de varias pruebas, los parámetros mediante los cuales se alcanzó el rendimiento esperado de la red se fijaron en los siguientes valores













Las predicciones fueron generadas para un período de 24 horas, las cuales no representan una característica típica de consumo diario, sino el resultado de la predicción aleatoria en cada punto de cualquier mes, día u hora, lo que muestra una validación general de lo que podría conformar un conjunto de datos anual.

**3. Resultados y discusión**

La convergencia de la red o modelo implementado se logró en todos los casos para una cantidad de corridas que oscila entre 5 y 10. El tiempo del entrenamiento se encuentra alrededor de 10 minutos.

Luego de varias iteraciones el error cayó por debajo de 2 x 10-3, el desempeño del error medio cuadrático puede observarse en la figura 5.



Figura 5. Iteraciones vs. error en una predicción de consumo de carga (Fuente [20])

Los parámetros que definen los pesos (W) para cada una de las capas de la red entrenada se presentan en las tablas 1, 2, y 3

Tabla 1. Valores de los pesos (W) entre las neuronas de la capa de entrada y la primera capa oculta

| **W1=net.IW{1,1}** |
| --- |
|  | **1** | **2** | **3** | **4** |
| **N1** | -0.0511 | 0.0025 | -0.0601 | -2.9586 |
| **N2** | -0.0004 | -0.0019 | 0.1257 | -1.7328 |
| **N3** | 0.2447 | -0.0036 | -0.1552 | -3.4324 |
| **N4** | 1.7186 | 0.1907 | -0.3352 | -3.4799 |
| **N5** | 0.0762 | 0.0514 | -0.7763 | 11.2109 |
| **N6** | 0.0062 | 0.0012 | 0.3733 | 1.2379 |
| **N7** | -0.2511 | 0.0048 | -0.0096 | -1.3445 |
| **N8** | 0.2654 | -0.0120 | -0.0426 | -0.3238 |
| **N9** | -0.0201 | -0.0151 | 0.4812 | 1.4717 |
| **N10** | -0.0014 | 0.0010 | -0.1323 | 0.2690 |
| **N11** | 0.0042 | 0.0123 | -0.4890 | 1.9043 |
| **N12** | -1.7590 | -0.1519 | 0.8674 | 2.9530 |

Tabla 2. Valores de los pesos (W) entre las neuronas de la primera capa oculta y las neuronas de la segunda capa oculta

| **W2=net.LW{2,1}** |
| --- |
|  | **1** | **2** | **3** | **4** | **5** | **6** |
| **N1** | -6.9155 | 7.8634 | 1.6779 | -0.0933 | -1.3493 | 6.116 |
| **N2** | -0.6461 | 4.185 | -0.7133 | -0.0671 | 0.1802 | 1.7352 |
| **N3** | -3.8668 | -26.3418 | 6.1262 | -0.1602 | 2.8873 | -4.4884 |
| **N4** | 7.5866 | -5.0431 | 2.4609 | 0.2931 | -1.7038 | -1.2798 |
| **N5** | -0.3003 | 4.1035 | -0.5498 | -0.056 | 0.0851 | 1.5043 |
| **N6** | 1.5105 | 8.4485 | -4.4696 | 0.344 | -0.4875 | 7.4599 |
| **N7** | 13.8262 | -1.2127 | 15.0625 | -0.1425 | 13.2399 | 15.0768 |
| **N8** | -3.8447 | 11.8385 | -0.4712 | -0.2472 | 0.2655 | 5.8395 |
| **W2=net.LW{2,1}** |
|  | **7** | **8** | **9** | **10** | **11** | **12** |
| **N1** | 2.5249 | -1.1145 | -6.5936 | 12.8574 | -6.1818 | -0.566 |
| **N2** | -0.9006 | -0.3639 | -1.9936 | 6.762 | -1.0082 | 0.3182 |
| **N3** | 9.3216 | 10.2767 | -6.1087 | -30.6213 | -2.3125 | 1.5473 |
| **N4** | 3.0168 | 2.6536 | 11.648 | -12.4201 | 10.0164 | -0.119 |
| **N5** | -0.6471 | -0.1533 | -1.5407 | 6.1234 | -0.6872 | 0.2942 |
| **N6** | -4.893 | -1.789 | -4.3335 | 19.7642 | -1.4707 | 2.8447 |
| **N7** | 2.7473 | 3.3179 | 1.1072 | -8.4182 | -15.1473 | 0.3859 |
| **N8** | -0.8075 | -0.714 | -4.071 | 15.3487 | 7.0114 | -0.0177 |

Tabla 2. Valores de los pesos (W) entre las neuronas de la capa de salida y las neuronas de la segunda capa oculta

| **W3=net.LW{3,1}** |
| --- |
|  | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** |
| **N1** | -0.1029 | -0.2205 | 1.1001 | 0.2796 | 0.2537 | -1.0455 | -0.0863 | 1.0944 |

Los parámetros que definen las ganancias para cada una de las capas de la red entrenada se presentan en la tabla 4

Tabla 4. Valores de las ganancias de las neuronas de cada capa

| **b1=net.b{1}** |  | **b2=net.b{2}** |  | **b3=net.b{3}** |
| --- | --- | --- | --- | --- |
| **N1** | 1.3318 | **N1** | -0.3374 | **N1** | 0.5009 |
| **N2** | -1.0655 | **N2** | -0.345 |  |
| **N3** | 0.1476 | **N3** | 13.3519 |
| **N4** | -9.6389 | **N4** | 4.5972 |
| **N5** | 0.8859 | **N5** | -0.211 |
| **N6** | -6.109 | **N6** | 2.0739 |
| **N7** | 2.5117 | **N7** | 8.1314 |
| **N8** | -0.824 | **N8** | 11.4073 |
| **N9** | -3.3484 |  |
| **N10** | 1.8991 |
| **N11** | 2.4495 |
| **N12** | -4.9732 |

La figura 6 muestra los resultados logrados en el entrenamiento con una cantidad de patrones superior a 13 veces los pesos de la red.

****

Figura 6. Resultados del entrenamiento del modelo (Fuente [20])

Por su parte en la figura 7 se muestra el resultado gráfico de la predicción para un patrón que no se había presentado a la misma durante su entrenamiento, lo que demuestra que sin ningún problema la red reproduce otros patrones, evidenciando el éxito de la red para realizar predicciones de demanda o carga eléctrica.



Figura 7. Resultados de una prueba con el modelo de Red implementado

Sobre los resultados puede señalarse que después de probar los algoritmos *traingd*, se comprobó que el algoritmo *trainlm*, correspondiente al método de Levenberg Marquardt garantiza una alta velocidad de aprendizaje y una excelente generalización de los patrones de entrenamiento que no se le habían presentado inicialmente, y por eso se escogió para completar el proceso de entrenamiento.

De los datos estimados para el caso de prueba mostrado, el error más significativo equivale a 4.77 %, es decir 0.921 MW, valor y aceptable teniendo en cuenta el comportamiento de los datos de entrada, como se observa en la figura 7, pues aunque se conserva la forma de onda, indicando que el comportamiento es similar todos los días de la semana, los valores de la demanda pico (cuyo valor máximo registrado en el año fue de 20.5 MW) son muy diferentes cada día.

El proceso de entrenamiento, involucró experimentar con muchos tipos de RNA tratando de encontrar no solo una configuración con menores errores, sino también un algoritmo que además de rapidez garantizara estabilidad en el resultado final. La red escogida es bastante robusta e involucra un gran número de parámetros, sin embargo alcanzó convergencia en menos tiempo que otras redes probadas en las cuales se trabajó con una sola capa oculta compuesta por un gran número de neuronas. De esta forma se confirma que no existe un procedimiento establecido para determinar el modelo de red que debe emplearse en cada aplicación y que sólo con la práctica puede determinarse cuál es la configuración de red que garantiza mejores resultados.

**4. Conclusiones**

Los resultados demuestran la eficacia de la aplicación del modelo de RNA implementado para la prediccion de la demanda al circuito de subtransmision seleccionado, tal como lo muestran las gráficas comparativas de los resultados obtenidos mediante los procesos de validacion y prueba con lo cual se considera cumplido el objetivo propuesto para el trabajo.

La precisión de la predicción, a pesar de no haber incluido otras variables que influyen en el consumo energético como son la temperatura , la humedad, y no hacer distinciones entre dias laborables y no laborables es considerable, al observarse un error no mayor al 5% entre la demanda pronosticada por el modelo y la real medida en una determinada hora. Estos parámetros deberan ser introducidos en el vector o capa de entrada en el futuro, para un incremento en la precision del pronóstico. También a futuro sería conveniente evaluar otros métodos que se reportan en la literatura para establecer comparaciones que permitan fundamentar mejor el empleo del tipo de red aquí descrito.

**5. Referencias bibliográficas**

[1] ORBEA, L.d.P.H., Pronóstico de la Demanda de Muy Corto Plazo utilizando Inteligencia Artificial. 2007. p. 3.

[2] E.A Feinberg, D Genethliou, “Chapter 12 Load forecasting”, Applied Mathematics for Power Systems, pp.269-282. http://www.ams.sunysb.edu/~feinberg/public/lf.pdf

[3] W. Charytoniuk, M.S. Chen, and P. Van Olinda. Nonparametric Regression Based Short-Term Load Forecasting. IEEE Transactions on Power Systems, 13:725–730, 1998.

[4] T. Haida and S. Muto. Regression Based Peak Load Forecasting using a Transformation Technique. IEEE Transactions on Power Systems, 9:1788–1794, 1994.

[5] O. Hyde and P.F. Hodnett. An Adaptable Automated Procedure for Short-Term Electricity Load Forecasting. IEEE Transactions on Power Systems, 12:84–93, 1997.

[6] S. Ruzic, A. Vuckovic, and N. Nikolic. Weather Sensitive Method for Short-Term Load Forecasting in Electric Power Utility of Serbia. IEEE Transactions on Power Systems, 18:1581–1586, 2003.

[7] M.Y. Cho, J.C. Hwang, and C.S. Chen. Customer Short-Term Load Forecasting by using ARIMA Transfer Function Model. Proceedings of the International Conference on Energy Management and Power Delivery, 1:317–322, 1995.

[8] H.T. Yang, C.M. Huang, and C.L. Huang. Identiﬁcation of ARMAX Model for Short-Term Load Forecasting: An Evolutionary Programming Approach. IEEE Transactions on Power Systems,11:403–408, 1996

[9] M. Peng, N.F. Hubele, and G.G. Karady. Advancement in the Application of Neural Networks for Short-Term Load Forecasting. IEEE Transactions on Power Systems, 7:250–257, 1992.

[10] A.G. Bakirtzis, V. Petridis, S.J. Kiartzis, M.C. Alexiadis, and A.H. Maissis. A Neural Network Short-Term Load Forecasting Model for the Greek Power System. IEEE Transactions on Power Systems, 11:858–863, 1996.

[11] A.D. Papalexopoulos, S. Hao, and T.M. Peng. An Implementation of a Neural Network Based Load Forecasting Model for the EMS.IEEE Transactions on Power Systems, 9:1956–1962, 1994.

[12] J W. Taylor and R. Buizza. Neural Network Load Forecasting With Weather Ensemble Predictions. IEEE Transactions on Power Systems, vol. 17, no. 3, august 2002

[13] S.Sing, S. Hussain and M. AbizBazaz. Short Term Load Forecasting Using Artificial Neural Network. Proceedings of the Fourth International Conference on Image Information Processing (ICIIP).2017

[14] A. Khotanzad, R.A. Rohani, T.L. Lu, A. Abaye, M. Davis, and D.J. Maratukulam. ANNSTLF–A Neural-Network-Based Electric Load Forecasting System. IEEE Transactions on Neural Networks, 8:835–846, 1997.

[15] S.J. Kiartzis and A.G. Bakirtzis. A Fuzzy Expert System for Peak Load Forecasting: Application to the Greek Power System. Proceedings of the 10th Mediterranean Electrotechnical Conference, 3:1097–1100, 2000.

[16] V. Miranda and C. Monteiro. Fuzzy Inference in Spatial Load Forecasting. Proceedings of IEEE Power Engineering Winter Meeting,2: 1063–1068, 2000.

[17] K.L. Ho, Y.Y. Hsu, F.F. Chen, T.E. Lee, C.C. Liang, T.S. Lai, and K.K. Chen. Short-Term Load Forecasting of Taiwan Power System using a Knowledge Based Expert System. IEEE Transactions on Power Systems, 5:1214–1221, 1990.

[18] S. Rahman and O. Hazim. Load Forecasting for Multiple Sites: Development of an Expert System-Based Technique. Electric Power Systems Research, 39:161–169, 1996.

[19] H.T. Yang and C.M. Huang. A New Short-Term Load Forecasting Approach using Self-Organizing Fuzzy ARMAX Models. IEEE Transactions onPowerSystems, 13:217–225, 1998.

[20] Y. García López. Predicción de la Demanda empleando Redes neuronales Artificiales. Trabajo de diploma. Tutores: Yordemys Santana y Alberto Limonte Ruiz. Universidad Central ¨Marta Abreu¨de Las Villas. Julio 2009. Visitado en http://dspace.uclv.edu.cu/bitstream/handle/123456789/6906/Yunier García López.pdf