

3.4 PREDICCIÓN DE CONSUMO DE CARGA

3.4.1 Descripción del problema

Un sistema de energía eléctrica debe abastecer de energía a todos los puntos de carga con una buena calidad del servicio. Por lo tanto un sistema eléctrico confiable, el cual asegura buena calidad, debe contar con las siguientes características:

- Entregar energía en forma continua a todos los puntos de carga.
- Los límites de la frecuencia y la tensión deben estar dentro de valores tolerables.
- El sistema debe operar en la medida de lo posible, con costos mínimos y con un mínimo de alteraciones ambientales o ecológicas.

Estas características pueden adquirirse por medio de una planeación exhaustiva del sistema, que permita conocer no sólo su estado actual, sino también las medidas que deben adoptarse para condiciones futuras.

Una de las herramientas útiles en el planeamiento de un sistema eléctrico es la predicción del consumo de carga, la cual permite conocer de antemano la necesidad de expansión del sistema; la finalidad de la predicción siempre será el



mejoramiento del servicio, convirtiéndose en uno de los primeros pasos en cualquier proceso de planeamiento de un sistema eléctrico.

Al hablar de predicción de carga, resulta útil aclarar que como carga se asume todo equipo que demanda energía del sistema de energía eléctrica, tales como lámparas, electrodomésticos, motores eléctricos, hornos eléctricos, etc. De esta manera se tienen varios tipos de carga:

- Motores en general
- Equipos de calentamiento
- Equipos electrónicos
- Equipo de iluminación

Desde el punto de vista del sistema de energía eléctrica, las cargas pueden ser separadas en tres (3) grupos funcionales:

- Cargas domiciliarias
- Cargas industriales
- Cargas comerciales

Estas cargas presentan características muy diferentes con relación al tamaño, simetría (1Φ o 3Φ), constancia de la carga y el período de funcionamiento.



La predicción de consumo de carga refleja las necesidades futuras de una población; esta previsión debe ser lo más ajustada a la realidad, ya que unos valores inferiores a los reales causarían deficiencias en la prestación del servicio en el futuro y un pronóstico de necesidades superior al real, motiva la inversión prematura en instalaciones que no tendrán un aprovechamiento inmediato.

La proyección del suministro de energía se hace con base en el consumo, aplicando porcentajes de pérdidas que pueden obtenerse de un análisis de los registros históricos (que normalmente se presentan en forma estadística), o por similitud con otros sistemas. En general las pérdidas tienden a disminuir a causa de las mejoras progresivas, que se introducen en los sistemas de transmisión, subtransmisión y distribución. En forma similar al consumo de energía, la proyección de la demanda pico se obtiene sumando las demandas máximas coincidentes en hora pico [37].

En la tabla 3.4.1 se observa el comportamiento típico de consumo de carga de una población, a la cual se realizó un seguimiento hora a hora durante una semana tipo. Se ha tomado el día lunes como 1 y en su orden el día domingo como 7, se asumió la una de la mañana como la hora 0 y las doce de la noche como la hora 23. Los datos correspondientes al consumo de la población se encuentran en (kW).



HORA	0	1	2	3	4	5	6	7	8	9	10	11
Lunes	1,2300	1,0889	1,0289	0,9879	0,9879	1,1050	1,3729	1,6649	1,6649	2,1569	2,3230	2,3659
Martes	1,6049	1,4389	1,3631	1,3559	1,3439	1,3890	1,5699	1,7750	2,0180	2,1900	2,3359	2,3630
Miércoles	1,6630	1,4689	1,3890	1,3751	1,3611	1,4140	1,6040	1,8009	2,0739	2,2301	2,3649	2,3990
Jueves	1,7299	1,5159	1,4320	1,3931	1,3909	1,4310	1,6140	1,8170	2,0989	2,2260	2,3810	2,3741
Viernes	1,7129	1,4569	1,3461	1,2880	1,2331	1,1911	1,1570	1,1700	1,2139	1,3370	1,4799	1,5740
Sábado	1,7130	1,2821	1,1820	1,1220	1,0961	1,1059	1,1710	1,2751	1,4121	1,5450	1,7110	1,7410
Domingo	1,4140	1,3250	1,2249	1,2239	1,1240	1,0191	0,9989	0,9989	0,9790	1,0150	1,1271	1,2271

HORA	12	13	14	15	16	17	18	19	20	21	22	23
Lunes	2,3731	2,2311	2,1560	2,2080	2,2949	2,3741	2,5000	2,4340	2,3560	2,0000	1,9890	1,8080
Martes	2,3359	2,1560	2,0799	0,1651	2,2551	2,3671	2,4770	2,4310	2,3540	2,2100	1,7085	1,7000
Miércoles	2,3580	2,2000	2,1231	2,1749	2,2049	2,3349	2,4640	2,3780	2,4140	2,0040	1,8582	1,7071
Jueves	2,3021	2,1459	2,0581	2,0809	2,1651	2,2380	2,2820	2,1540	2,1020	1,9950	1,9040	1,8590
Viernes	1,5951	1,5771	1,5629	1,5320	1,5440	1,6380	1,7310	1,7480	1,7921	1,8321	1,8620	1,7930
Sábado	1,7129	1,6200	1,1570	1,5831	1,6251	1,6251	1,8950	1,9040	1,9310	1,9360	1,9580	1,748
Domingo	1,2950	1,3130	1,2909	1,2600	1,2669	1,3631	1,1530	1,6020	1,6440	1,6150	1,5030	1,4990

Tabla 3.4.1 Datos de consumo de carga de una población durante una semana

Estos datos son una recopilación de un promedio histórico de diferentes años, del consumo de la población escogida como muestra, con ellos podemos determinar la tendencia del consumo de los usuarios de esta población. El objetivo es entrenar una red neuronal que aprenda los datos anteriores y a partir de ellos esté en capacidad de predecir lo que sucederá en cualquier día de la semana a una hora determinada en años futuros.

El comportamiento de estos usuarios se visualiza en la figura 3.4.1, en donde puede notarse las tendencias que pueden llegar a causar congestión en la central en caso de no preverse con anticipación:



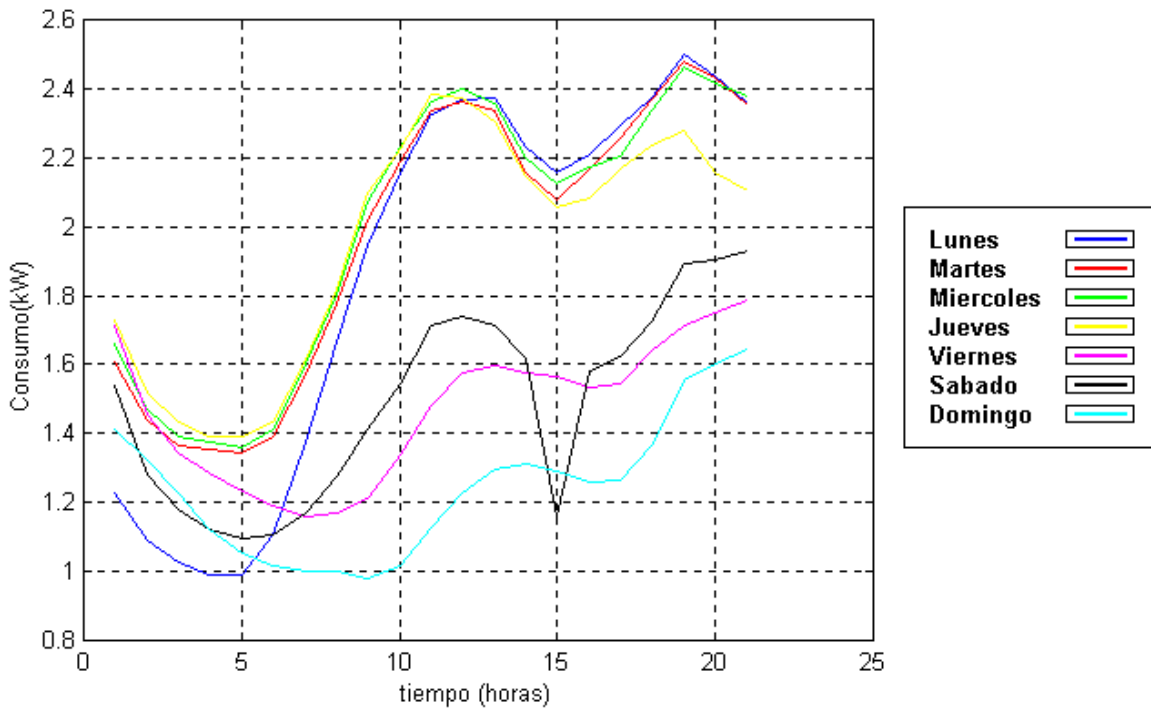


Figura 3.4.1 Curvas de Carga

3.4.2 Justificación del tipo de red.

El algoritmo Backpropagation es un algoritmo de aprendizaje supervisado, el cual necesita conocer cuál es la salida esperada (columnas interiores de la tabla 3.4.1) asociada a cada una de las entradas (columnas referentes al día y a la hora en la tabla 3.4.1), que actualiza pesos y ganancias siguiendo la regla de pasos descendientes descrita en la sección 2.3.3

$$W^m(k+1) = W^m(k) - \alpha s^m (a^{m-1})^T \tag{3.4.1}$$

$$b^m(k+1) = b^m(k) - \alpha s^m \tag{3.4.2}$$



Una de las mayores ventajas de las redes multicapa, y en especial de la red Backpropagation, es que pueden aproximar cualquier función si se escoge una adecuada configuración para la red [16] y un adecuado número de neuronas en la capa oculta, escogencia que depende de la experiencia del desarrollador de la red. La red Backpropagation es un excelente aproximador de funciones, aunque es imposible determinar una configuración exacta de la red para cada aplicación.

El proceso de aprendizaje no es fijo para ninguna red neuronal, el éxito consiste en probar con diferentes configuraciones hasta obtener la respuesta deseada; para esta aplicación se escogió una red 2:12:8:1, es decir que para un vector de entrada de dos 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.

Un esquema de la red puede observarse en la figura 3.4.2, la cual muestra las características de los datos de entrada y salida de la red. Allí puede verse que es necesario ingresar a la red el día y la hora para los cuales se desea conocer el valor de demanda pico de acuerdo a la convención de la tabla 3.4.1, las condiciones de entrada a la red pueden ser tan complejas como se quiera, es decir, la entrada puede convertirse fácilmente en un vector de cuatro componentes que incluya además del día y la hora, el mes y el año para los cuales se requiere predecir el valor de demanda; existen también otras opciones como convertir la entrada en un vector de tres componentes donde se incluya hora, día y tipo de carga para el cual se desea prever el consumo.



La elección de los patrones de entrada debe realizarse dependiendo de las necesidades explícitas que se tengan en el momento de hacer la predicción de carga, de la forma en que vaya a procesarse la información de salida de la red y de la cantidad y calidad de la información disponible.

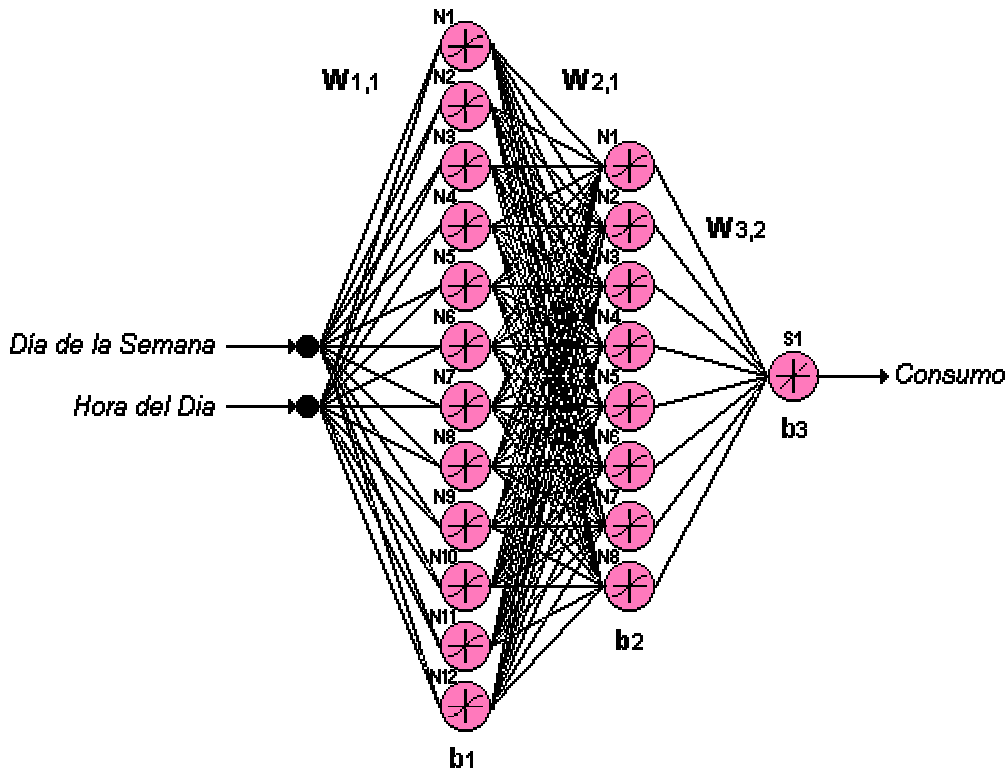


Figura 3.4.2 Red Backpropagation 2:12:8:1 para predicción del consumo de carga

Cualquier cambio que se realice en los patrones de entrenamiento exige una codificación diferente del vector de entrada y a su vez cambia las condiciones generales de la red, pero el proceso de entrenamiento sigue siendo igual. En esta aplicación se utilizó la configuración de la figura 3.4.2 porque el objetivo es mostrar el funcionamiento general de una red Backpropagation, mostrando las bondades de este tipo de red para aproximar funciones (en este caso las curvas de carga de



la población en estudio) y su gran capacidad para predecir comportamientos futuros de patrones nunca antes presentados en la etapa de aprendizaje.

3.4.3 Entrenamiento de la red

El código de entrenamiento para esta red, es desarrollado con base en la herramienta de redes neuronales del Matlab. La red es creada mediante el comando *newff* para creación de redes Backpropagation, con las siguientes características:

```
net=newff([0 1;0 1],[12,10,1],  
          {'tansig','tansig','purelin'},'trainlm');
```

Los valores de iniciación de la matriz de pesos se generaron aleatoriamente; después de varias pruebas, los parámetros que determinan el entrenamiento se fijaron en los siguientes valores, mediante los cuales se alcanzó el rendimiento óptimo de la red.

```
net.trainParam.show =10;  
net.trainParam.epochs =1000;  
net.trainParam.goal =1e-5;  
net.trainParam.lr=0.075;  
net.trainParam.mem_reduc =1;  
net.trainParam.min_grad =1e-12;
```

Una explicación del significado de los comandos anteriores, puede encontrarse en el anexo A.



Los valores de entrada a la red se agruparon en el vector de dos entradas p , (tabla 3.4.1), las cuales describen el día y la hora en que desea saberse el valor de consumo. La red se entrenó sólo con 180 valores de entrada de los 206 que conforman todo el set de entrenamiento, estos valores fueron normalizados porque de esa forma se obtuvo una mejor generalización de la red en el proceso de aprendizaje; el proceso de normalización consiste en dividir cada uno de los valores de demanda de la tabla 3.4.1 por el valor de demanda máximo, de tal forma que el mayor valor de entrada a la red será uno:

$$\text{Demanda normalizada} = \frac{\text{Valor de la demanda}}{\text{Demanda máxima}}$$

El valor esperado, el valor de la potencia pico en un instante determinado, es representado por el escalar t

```
t=t';  
[net,tr]=train(net,p,t);
```

Los restantes 26 valores de los patrones de entrenamiento se tomaron como valores de prueba del rendimiento de la red y se agruparon en el vector p_1 , siendo t_1 el vector de salidas esperadas.

```
p1=p1';  
a = sim(net,p1);
```



Luego de varias iteraciones el error cayó por debajo de 2×10^{-5} , el desempeño del error medio cuadrático puede observarse en la figura 3.4.3.

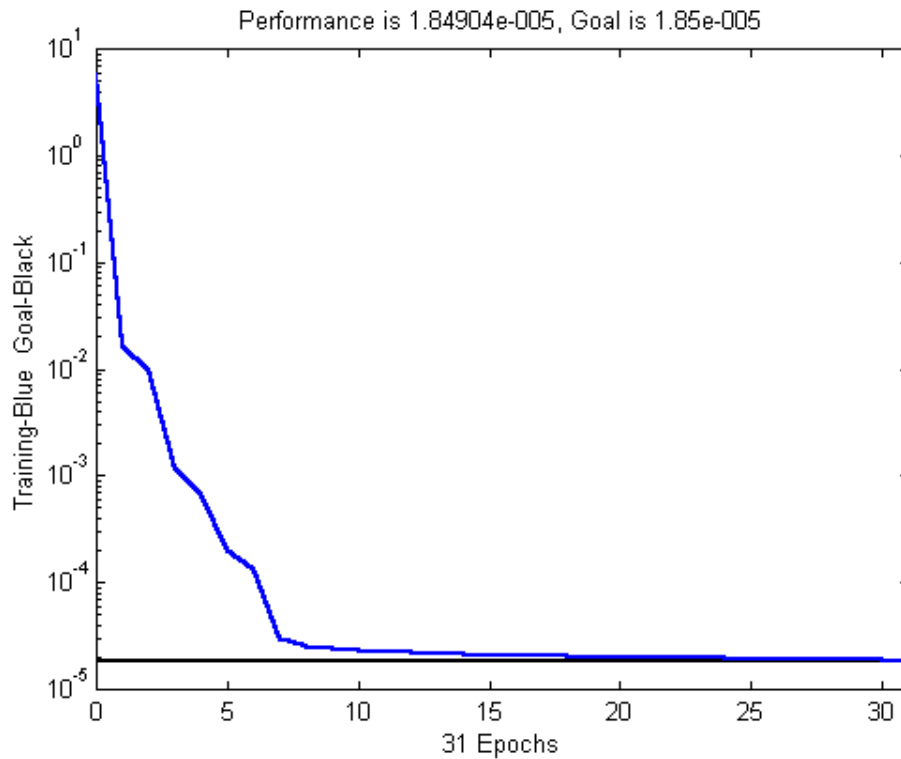


Figura 3.4.3 Iteraciones Vs Error en una predicción de consumo de carga

Los parámetros de la red entrenada se presentan en la tabla 3.4.2

W1=net.IW{1,1}

	1	2
N1	-21,9482	3,0064
N2	-0,4403	14,0061
N3	-30,1225	-14,4532
N4	6,8063	9,6593
N5	-0,6548	10,661
N6	-32,5167	-16,0543
N7	-31,5269	28,1193
N8	-26,5558	-10,0917
N9	2,4681	-26,1821
N10	7,306	1,6967
N11	-0,0242	9,3016
N12	-5,2355	0,211



W2=net.LW{2,1}

	1	2	3	4	5	6
N1	-0,3267	18,5006	1,4482	7,3519	5,56	-63,61
N2	-1,7789	-34,377	132,102	56,1894	-36,79	5,4474
N3	2,9001	-1,6303	31,2892	2,0873	-3,0091	-1,3828
N4	-0,8462	8,6876	4,0463	6,0138	2,3158	-0,2587
N5	12,0037	0,1605	60,885	148,871	2,5194	106,22
N6	-98,9806	105,356	115,699	0,5065	95,0003	-17,001
N7	-5,6485	136,978	15,7418	43,3306	-13,133	-23,351
N8	-0,7708	-80,889	1,397	100,628	5,1579	-0,1767

	7	8	9	10	11	12
N1	72,191	-27,627	42,4595	0,9163	-2,2968	57,8689
N2	180,2947	-253,47	-92,087	0,066	3,5167	-77,476
N3	3,3586	2,9584	1,2985	-0,0981	0,0207	-1,3115
N4	29,6843	0,8862	-10,697	0,4033	-1,1097	25,1813
N5	10,1653	0,8031	-30,127	0,2075	2,0099	3,7721
N6	46,6346	112,561	-28,56	6,8125	96,3854	172,322
N7	-58,043	-16,587	-10,875	-35,975	0,3041	74,7964
N8	94,3186	-1,8353	-1,4035	0,0731	-0,1007	33,8039

W3=net.LW{3,2}

	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

b1=net.b{1}

N1	17,6647
N2	-12,2797
N3	36,1202
N4	-11,3385
N5	-4,2961
N6	29,8648
N7	-12,2397
N8	5,5126
N9	16,6252
N10	-2,721
N11	-0,2757
N12	4,0912

b2=net.b{2}

N1	35,736
N2	-58,2132
N3	-28,9712
N4	29,135
N5	7,4498
N6	-119,529
N7	20,4235
N8	84,9714

b3=net.b{3}

N1	-0.5760
-----------	---------

Tabla 3.4.2 Resultados obtenidos con la red entrenada



En la tabla 3.4.3, se muestran seis de los valores de la simulación de la red; los valores de a_1 corresponden a los valores entregados por al red y los valores de t_1 corresponden a los valores esperados.

a_1=						
1,341	1,3925	1,1588	1,1699	1,2124	1,339	1,4776
t_1=						
1,3439	1,389	1,157	1,17	1,2139	1,337	1,4799
e=						
0,0029	0,0035	0,0018	0,0001	0,0015	0,002	0,0023
$e_{máx}$=			e_{min} =			
0,0035			1,19E-04			

Tabla 3.4.3 Valores de la simulación

Los comandos, que evaluaron el desempeño de la red son los siguientes, con ellos se calculó el error en la simulación

```
e=abs (t1-a1)
emax=max (e)
emin=min (e)
```

En la tabla 3.4.3 puede observarse el trabajo final de la red y la excelente labor de aproximación que realiza. Ilustrar el resultado para todos los patrones de prueba resulta un poco extenso, por lo tanto se escogieron los patrones más representativos, donde puede verse que sin ningún problema la red ha



aproximado patrones que no se le habían presentado durante el entrenamiento, garantizando el éxito de la red para realizar predicciones de consumo de carga.

De los datos entregados en el entrenamiento, el error más significativo equivale a 0.0035, es decir 3.5kW, valor muy aceptable teniendo en cuenta la disparidad de los datos de entrada, ya que como se observa de la figura 3.4.1 la población tiene un comportamiento bastante aleatorio, 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 son muy diferentes cada día

Después de probar los algoritmos *traingd*, *traingda*, (ver anexo A) se comprobó que el algoritmo *trainlm*, correspondiente al método de Levenberg Marquardt garantizaba 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.

El proceso de entrenamiento, involucró experimentar con muchos tipos de redes tratando de encontrar no solo una configuración óptima, si no también un algoritmo que además de rapidez garantizara estabilidad en el resultado final. La red escogida en la figura 3.4.2 es bastante robusta e involucra un gran número de parámetros (149 en total), sin embargo alcanzó convergencia en menos tiempo que otras redes escogidas 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, ésta aparente dificultad de las redes neuronales puede aprovecharse para resaltar una de sus grandes ventajas: La adaptabilidad de esta teoría a diferentes lenguajes de programación, de tal forma que con unos pocos comandos los parámetros de la red pueden ser transformados sin mayor problema, adoptando una configuración totalmente diferente.

Redes con el tipo de estructura, como la utilizada en esta aplicación pueden emplearse para predecir la necesidad de una reconfiguración del sistema en un día y una hora específica. Es posible también que los resultados de una red como ésta, se utilicen para adoptar estrategias de restauración del servicio en diferentes horas y diferentes días, pues al conocer los valores de demanda es fácil determinar comportamientos críticos y dar prioridad en el restablecimiento, dejando por fuera a la menor cantidad de usuarios posible.

Las diferentes aplicaciones para las que sea necesario implementar una predicción de consumo de carga, pueden derivarse fácilmente de ésta adaptando el código fuente a los patrones con los que se desea realizar la predicción, así como se mencionó anteriormente, lo único necesario es variar la estructura del vector de entrada p adecuándolo a la cantidad de entradas que se vayan a ingresar (hora, día, mes, año, tipo de carga) a la red y también establecer con claridad que tipo de información de salida se espera dependiendo del propósito de la predicción.

