



Universidad de Valladolid
Facultad de Ciencias

TRABAJO FIN DE GRADO

Grado en Estadística

Predicción a corto plazo de precios del mercado eléctrico español

Autora:

D.^ª Amaia Ganuza Galdeano

Tutora:

D.^ª M.^ª del Pilar Rodríguez del Tío

Índice

| | |
|--|----|
| 1. Resumen..... | 5 |
| 2. Abstract | 5 |
| 3. Palabras clave..... | 5 |
| 4. Introducción | 5 |
| 5. Pool eléctrico..... | 5 |
| 6. Explicación del método SOM | 6 |
| 6.1. Arquitectura del SOM..... | 7 |
| 6.2. El algoritmo del SOM..... | 9 |
| 7. Preparación de los datos..... | 10 |
| 7.1. Análisis Descriptivo | 10 |
| 8. Ajuste de modelos SOM..... | 13 |
| 8.1. Modelo con datos desde 2003..... | 13 |
| 8.1.1. Características del modelo elegido | 14 |
| 8.1.2. Predicción..... | 16 |
| 8.2. Modelo con datos desde marzo de 2009..... | 17 |
| 8.2.1. Características del modelo elegido | 17 |
| 8.2.2. Predicción..... | 18 |
| 8.3. Modelo con datos de 2012 | 19 |
| 8.3.1. Características del modelo elegido | 21 |
| 8.3.2. Predicción..... | 22 |
| 9. Ajuste de modelos ARIMA..... | 23 |
| 9.1. Modelos con datos desde marzo de 2009 | 23 |
| 9.2. Modelos con datos de 2012 | 27 |
| 10. Comparación SOM vs ARIMA | 30 |
| 10.1. Modelo con datos desde marzo de 2009..... | 30 |
| 10.2. Modelo con datos de 2012..... | 31 |
| 11. Conclusiones..... | 32 |
| 12. Posibles continuaciones | 33 |
| 13. Bibliografía | 34 |
| 14. Anexos..... | 35 |
| I. Datos utilizados..... | 35 |
| II. Scripts R..... | 35 |
| Lectura de datos..... | 35 |

| | |
|------------------------------|----|
| Descriptivas | 35 |
| Ajuste de modelo SOM | 35 |
| Ajuste de modelo ARIMA | 39 |
| Comparación de modelos | 40 |

1. Resumen

El presente trabajo comienza con una pequeña explicación sobre el funcionamiento del mercado eléctrico español y la manera en que éste es gestionado. Continúa con una breve explicación del método SOM (Self Organizing Map), que utilizaremos en la parte práctica para hacer predicciones de precios.

Para la predicción horaria de los precios del mercado eléctrico se utilizan datos históricos. Para ellos se eligen modelos de predicción obtenidos a partir de dos técnicas diferentes: SOM y ARIMA.

En nuestro ejemplo el método SOM realiza mejores predicciones que el modelo ARIMA ajustado como referencia, sin embargo, hay que tener en cuenta que no se han explotado todos los recursos que ésta última técnica ofrece, puesto que el objetivo principal del presente trabajo es la aplicación del modelo SOM para la predicción de los precios a corto plazo.

2. Abstract

This study starts with a short explanation about the operation of the Spanish electricity market and the way it is managed. It follows with a brief explanation about SOM (Self – Organizing Map) method. After that it presents the practical part.

The historical data of the electricity market prices are used for its prediction. Two prediction models have been developed from two different techniques: SOM and ARIMA.

In our example SOM method makes better predictions than ARIMA model, but it must be taken into account that it has not been exploited all the resources that the latter technique offers, since the main objective of this study is the application of SOM model for the prediction of short-term prices.

3. Palabras clave

Mercado eléctrico, predicción a corto plazo, Self Organizing Map, ARIMA.

4. Introducción

El objetivo de este trabajo es la predicción a corto plazo del precio del mercado eléctrico español con frecuencia horaria. Para ello se cuenta con datos de precio de mercado desde enero de 2003 hasta abril de 2014.

En el presente trabajo, utilizando los datos citados, se comparan las predicciones obtenidas mediante el método “Self Organizing Map” (SOM), que se explicará en el apartado 6, y las obtenidas mediante un modelo ARIMA, técnica hasta ahora más utilizada.

5. Pool eléctrico

El pool o mercado eléctrico, es el sistema a través del cual se negocian los intercambios de energía eléctrica en la península (España y Portugal). El **precio varía cada hora** dependiendo de

lo que los productores y consumidores quieran producir o consumir. Según las necesidades éstos (productores y consumidores) presentan una oferta.

El mercado es gestionado por el OMIE (Operador del Mercado Ibérico de Energía), un operador independiente. Este se articula a través de una sesión diaria y seis intradiarias. La sesión diaria permite la presentación de ofertas para las 24 horas del día siguiente. Los mercados intradiarios se convocan a lo largo del día anterior y del propio de entrega. Las ofertas se introducen a través de Unidades de Oferta (UOF), que integran la producción de una o más centrales representadas por el mismo agente, o la demanda de un conjunto de consumidores suministrados por el mismo comercializador.

Los productores y compradores pueden pactar un intercambio bilateral independiente al mercado, aunque habitualmente la mayor parte acude al pool para realizar las compraventas.

Los vendedores son las centrales de producción de régimen ordinario (gran hidráulica, nuclear y térmicas) y especial (eólica, solar, biomasa, mini-hidráulica). **Los compradores** son la totalidad de los consumidores (domésticos e industriales). Por otro lado hay unidades que trasladan la energía fruto de intercambios internacionales.

Cada sesión y cada hora el OMIE ordena las ofertas recibidas de UOF de menor a mayor precio para la venta y de mayor a menor para la compra, siendo el precio inferior 0 €/MWh y el superior 180,3 €/MWh. El resultado gráfico serían dos curvas donde el eje x es la energía y el eje y el precio. El punto donde se cruzan ambas líneas es el **precio de casación** para esa hora y sesión.

Las centrales nucleares y renovables venden energía a precio aceptante, es decir, 0€/MWh. Esto es debido a que las centrales nucleares tienen baja capacidad de parada y las renovables tienen poca capacidad de almacenaje de energía primaria (viento y sol). El resto de centrales tienen en cuenta sus costes de generación y oportunidad y pueden introducir ofertas complejas.

Una menor demanda de consumo implica una bajada de precios, así como una elevación de la producción renovable puede implicar una fuerte bajada del precio de casación, llegando a cubrirse toda la demanda a precio 0€/MWh. Por lo que **las energías renovables pueden suponer grandes ahorros** en el pool eléctrico, lo que influye en el precio de todos los consumidores.

6. Explicación del método SOM

Los mapas autoorganizados o SOM (por sus siglas en inglés), fueron inventados por Teuvo Kohonen. Son un tipo de red neuronal artificial entrenada mediante aprendizaje no supervisado.

Una de las características más importantes de las redes neuronales es la capacidad que tienen para “aprender” de los datos pasados, obtener información de estos datos, y de esta manera, aprovechar la información para hacer predicciones futuras.

En el estudio de la metodología del SOM para predicción se pueden distinguir dos procesos: entrenamiento y predicción:

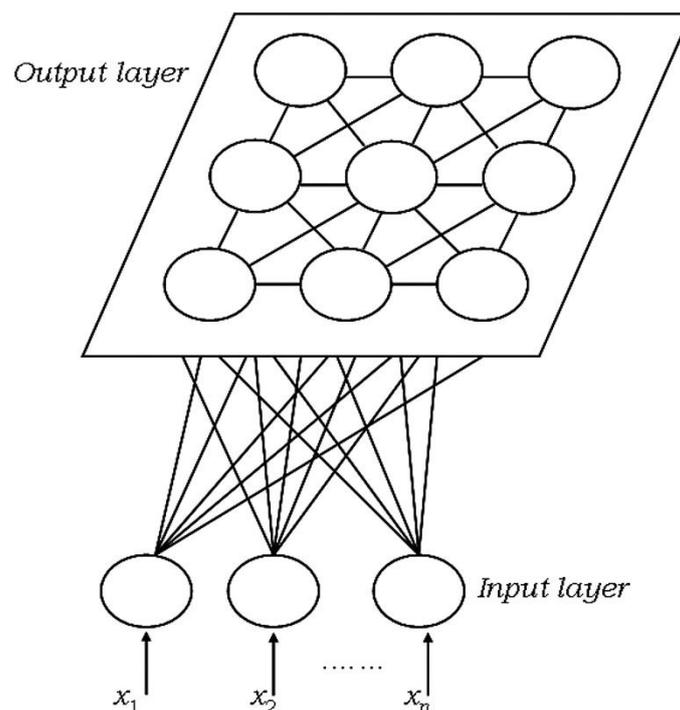
En el proceso de entrenamiento el SOM “aprende” de los datos incluidos, y a través de estos se ajustan los parámetros del modelo. Por entrenamiento se entiende la capacidad de obtener información de las situaciones pasadas, es decir, “aprender” de la experiencia.

Los parámetros que se obtienen en el proceso de entrenamiento son utilizados en la predicción. En el proceso de predicción se incluyen nuevas instancias y a través de los parámetros obtenidos durante el entrenamiento se obtiene una predicción.

Ofrece una manera de representar datos multidimensionales en espacios de dimensiones menores, normalmente en una o dos

6.1. Arquitectura del SOM

Un modelo SOM está compuesto por dos capas de neuronas. La capa de entrada (formada por n neuronas, tantas como variables de entrada haya), se encarga de recibir y transmitir a la capa de salida la información procedente del exterior. La capa de salida (formada por m neuronas) es la encargada de procesar la información y formar el mapa.



La información siempre se propaga desde la capa de entrada a la de salida, es decir, las conexiones entre las capas son siempre hacia adelante. Cada neurona de entrada i está conectada con cada una de las neuronas de salida j mediante un peso w_{ji} . Así, las neuronas de salida tienen asociado un vector de pesos W_j , llamado *codevector*. Este vector se considera un vector de referencia ya que representa a las instancias recogidas en la neurona j . Es de esta manera como el SOM proyecta un espacio de alta dimensión en un mapa de dos dimensiones.

Los *codevectors* son vectores de la misma dimensión que los vectores de entrada (instancias). Cada elemento del vector corresponde al valor que se le asigna a cada una de las variables de

entrada. Se inicializan aleatoriamente y se van actualizando en cada iteración conforme se van incluyendo instancias en la neurona o en las de su alrededor.

En la capa de salida, cada neurona tiene influencia en las de su alrededor. Las neuronas cercanas entre sí pertenecen a una vecindad. El tamaño de la vecindad depende del valor que tome el radio. El SOM utiliza una función de vecindad para preservar las propiedades topológicas del espacio de entrada.

Se pueden analizar varios elementos en el estudio de la arquitectura del SOM. Consideramos en primer lugar, la selección de variables. Seleccionar las variables más relevantes (número y relación) para obtener buenos resultados es uno de los pasos más importantes de todo el proceso. Una mala elección puede hacer que el modelo genere pésimos resultados.

En segundo lugar, citamos la decisión del modo en que se van a transformar los datos. Dependiendo de las características de cada una de las variables la transformación de los datos juega un papel muy importante a la hora de la realización del modelo.

Por último, consideramos la elección de la dimensión de la red, el número de iteraciones, los valores de alfa, el radio y la forma del vecindario (circular o cuadrado).

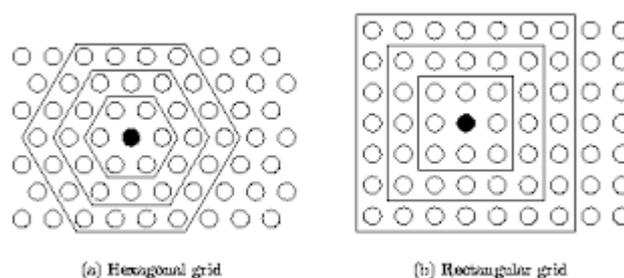
Con respecto a la elección de la dimensión de la red, es una tarea complicada, ya que hay diferentes teorías sobre la correcta elección de esta. Por un lado, se propone utilizar mapas de Sammon para ver la distribución de los datos basados en la función de densidad de forma gráfica. Por otro lado, se utilizan fórmulas como $N^{\circ} \text{ de nodos} = 5 * n^{0.54321}$, siendo n el número de instancias en el modelo.

En cuanto al número de iteraciones, se debe elegir un número tal que el modelo no quede ni infra ajustado ni sobre ajustado.

El parámetro alfa (o coeficiente de aprendizaje) determina cuánto va a ser afectada la vecindad de la BMU (Best Matching Unit, neurona elegida para incluir la instancia). El valor de alfa va decreciendo de manera constante a lo largo de las iteraciones.

El radio define la extensión del vecindario. Si una neurona está dentro de la distancia fijada desde la BMU, entonces esta neurona se considera dentro del vecindario, y su *codevector* será actualizado. Este valor, al igual que el valor de alfa, se reduce a lo largo de las iteraciones.

Por último, queda hablar sobre la forma del vecindario. La configuración usual de las neuronas es un espacio regular de dos dimensiones, en una rejilla hexagonal, en cuyo caso cada neurona tendría 6 vecinos o rectangular, en cuyo caso tendría 8 vecinos.



6.2. El algoritmo del SOM

Durante el entrenamiento el SOM realiza los siguientes pasos:

1. Se selecciona una instancia x al azar del conjunto de datos seleccionado para entrenar. Se calcula su distancia a cada uno de los vectores del codebook (el codebook es el conjunto de vectores de referencia, codevectors). Para calcular estas distancias se suelen utilizar distancias euclídeas.
2. La neurona más cercana, con menor distancia a x , será la BMU y el resto de vectores del codebook se actualiza. Esta actualización está regida por la tasa de aprendizaje (alfa) y el radio elegidos. Conforme se van actualizando los vectores el valor tanto de la tasa de aprendizaje como del radio decrecen.

Los pasos 1 y 2 se repiten hasta que se asignan todas las variables del conjunto de entrenamiento. Cuando el proceso de aprendizaje ha terminado los vectores próximos en distancias euclídeas se encuentran en neuronas adyacentes.

El SOM no siempre se utiliza para predecir. Es más común recurrir al SOM para hacer agrupamientos de datos, es decir, una vez construido el modelo, se clasifican las nuevas instancias en las neuronas del mapa y dependiendo de su ubicación pertenecen a un grupo u otro.

Otra utilidad que se le da a este algoritmo es la detección de fallos, por ejemplo, en maquinaria. En estos casos se comparan los valores del codevector al que se le ha asignado la nueva instancia con los valores reales de ésta, y si se obtienen errores altos en algún punto concreto o tendencias crecientes de error puede significar que el aparato se está estropeando.

Habitualmente, la clasificación de nuevos datos se realiza de la siguiente manera:

1. Una vez obtenido un modelo adecuado, se introducen las instancias que se quieren predecir. Se escoge una instancia y , al igual que en el entrenamiento, se calculan las distancias euclídeas de esta instancia a cada uno de los vectores del *codebook*.
2. La neurona más cercana será a la que se asigne la nueva instancia, pero en este caso los pesos no se actualizan.

En este trabajo, la predicción se realiza de la siguiente manera:

1. Al igual que en el procedimiento anterior, se incluyen las nuevas instancias en el modelo obtenido, pero en este caso las instancias están incompletas (cada variable corresponde a una hora, ver apartado 7), es decir, únicamente se incluyen los valores de las primeras horas de cada día. Si no se incluyera ningún valor, el algoritmo no podría calcular distancias a los codevectors.
2. Una vez asignada la nueva instancia a una neurona, se calcula el valor medio de los precios horarios de las instancias incluidas en dicha neurona en el momento de entrenamiento. Estos valores medios serán la predicción de las horas faltantes en la instancia que se quiere predecir.

Para comparar las distintas pruebas realizadas la medida a comparar es el **error de cuantización** (*quantization error*, QE), éste calcula la distancia media de los vectores

pertenecientes a una neurona al *codevector* de dicha neurona. Al aumentar el número de neuronas (dimensión de la UMatrix), el QE desciende, ya que los datos estarán distribuidos de manera más dispersa por el mapa.

Las medidas que se utilizan para comparar modelos de distinta dimensión son los errores topográficos. En el paquete de R utilizado (kohonen), se cuenta con dos **errores topográficos**: *nodedist* y *bmu*.

- *Nodedist*: distancia media, en términos de las coordenadas (x, y) del mapa, entre los *codevectors* más parecidos de cada una de las neuronas.
- *BMU*: distancia media en términos de las coordenadas (x, y) del mapa entre la unidad que mejor se ajusta (*BMU*) y la segunda que mejor se ajusta, para todos los datos de la muestra.

7. Preparación de los datos

7.1. Análisis Descriptivo

En la tabla 11 (anexo I) se muestran las variables utilizadas para la creación del modelo. La matriz de datos utilizada como datos de entrenamiento en el primer modelo SOM es de dimensión 4110 x 32 (datos desde el 1 de enero de 2003 hasta el 2 de abril de 2014), la utilizada en el segundo modelo es de 1827 x 32 (del 2 de abril de 2009 al 2 de abril de 2014) y por último la utilizada en el tercer modelo, está formada por los 365 días de 2012, es decir, es de dimensión 365x32.

VARIABLES:

- FECHA: día/ mes/ año. Fecha en la que se obtiene cada uno de los precios.
- ANIO: año.
- MES: mes.
- DIASEM: día de la semana en el que se obtiene cada uno de los precios. (L, M, X, J, V, S y D).
- DIA: Día del mes en que se obtiene cada uno de los precios.
- H1-H24: precio del mercado eléctrico en cada una de las 24 franjas horarias. Este precio está medido en €/MWh.

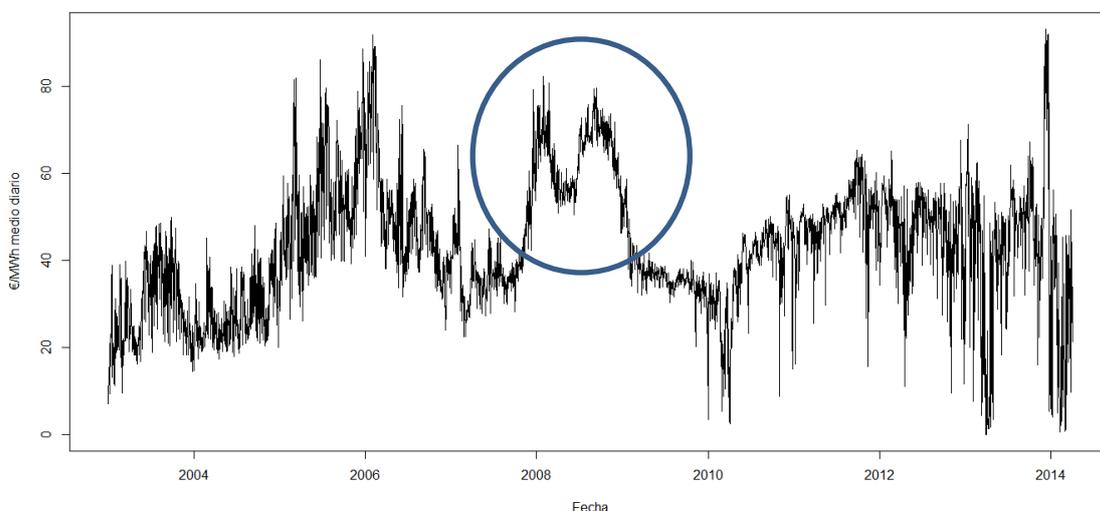


Gráfico 1. Precio medio diario de los datos de entrenamiento (del 1 de enero de 2003 al 2 de abril de 2014).

En el gráfico 1 se observa que la serie varía mucho a lo largo de los años. En 2008 se aprecia un comportamiento diferente debido a la creación de un nuevo real decreto que limitó el crecimiento de la energía solar fotovoltaica y a causa de ello la energía eólica sufrió un aumento considerable.

Puesto que somos conscientes de que la volatilidad de este tipo de series dificulta su ajuste por modelos ARIMA, buscamos un periodo favorable al ajuste del modelo con el único objetivo de utilizarlo como referencia. Por ello se usan los datos a partir del 25 de marzo de 2009 para ajustar modelos mediante esta técnica. En cambio, para la metodología SOM decidimos comenzar probando un primer modelo con los datos desde 2003.

Con el fin de conocer las características principales de los datos, a continuación se representan los diagramas de caja por hora.

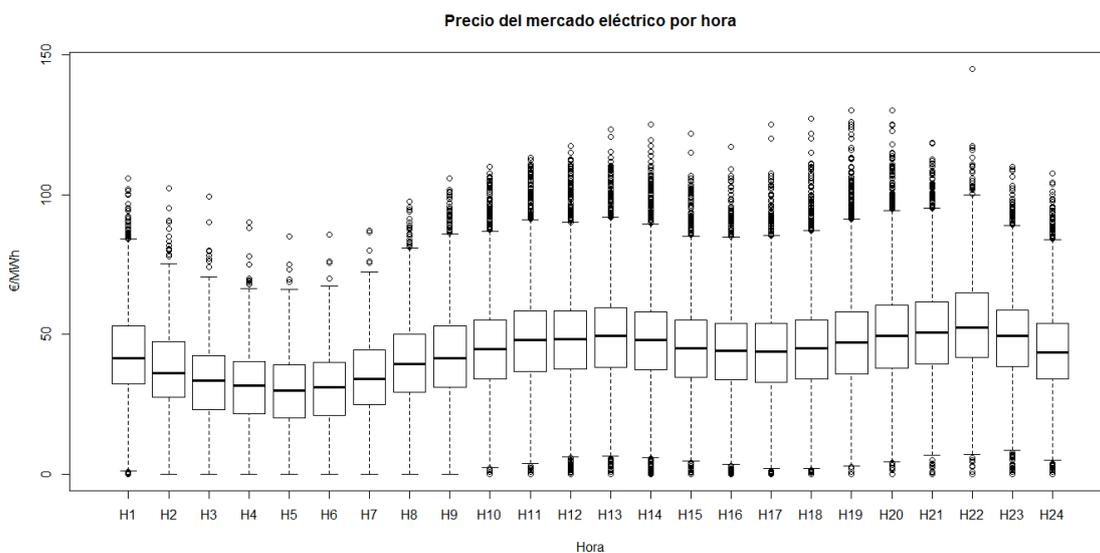


Gráfico 2. Diagrama de cajas múltiple para cada hora.

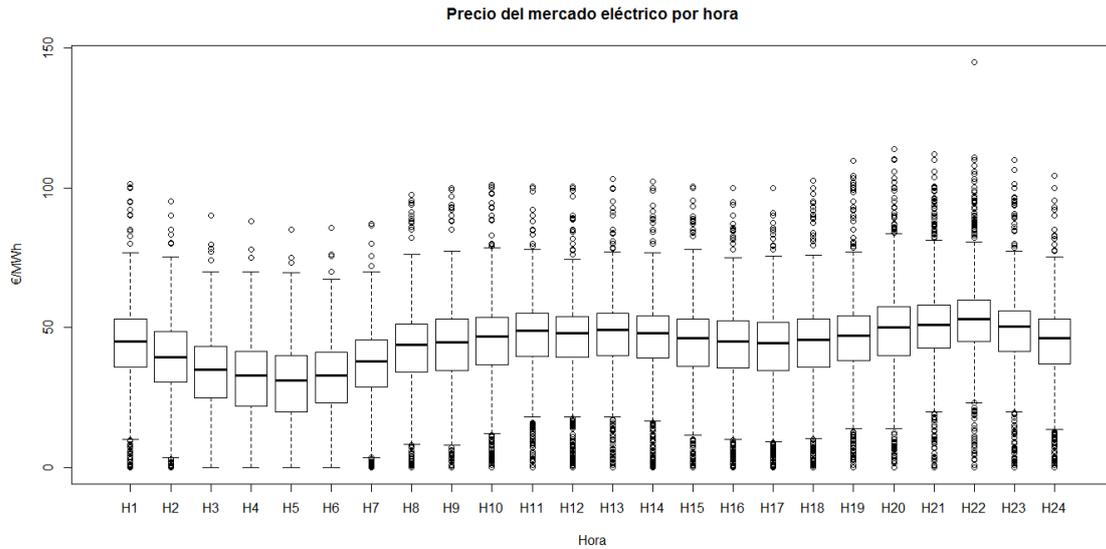


Gráfico 3. Diagrama de cajas múltiple para cada hora con los datos desde el 25 de marzo de 2009.

En los dos gráficos anteriores (2 y 3) se observa la diferencia de precio entre las horas, siendo más barato a las 5 de la mañana y más caro a las 22. Esta característica no cambia al seleccionar los datos de los últimos años.

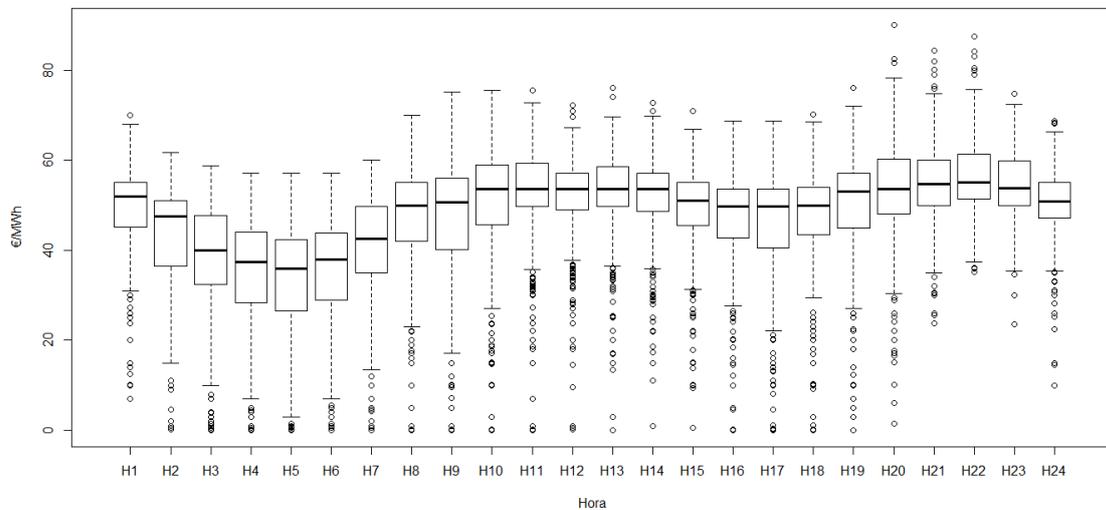


Gráfico 4. Diagrama de cajas múltiple para cada hora de los datos del año 2012.

Los datos de 2012 tienen valores máximos bastante inferiores a los de los datos de varios años en conjunto. Tanto en el gráfico 2 como en el gráfico 3 se observan máximos de casi 150€, mientras que en el gráfico 4 el máximo ronda los 90 €.

8. Ajuste de modelos SOM

8.1. Modelo con datos desde 2003

A pesar de que los precios de los primeros años tienen comportamientos muy diferentes a los últimos se decide comenzar probando con todos los datos, como ya comentamos en el apartado anterior. Por lo tanto se incluyen 4110 vectores de dimensión 24, es decir, datos de las 24 horas de cada uno de los días desde enero de 2003 hasta abril de 2014.

Los datos han sido estandarizados, previamente a la aplicación del modelo SOM, para obtener mejores predicciones y un mejor funcionamiento del método.

Para elegir el mejor modelo, se ha basado la elección en el **error de cuantización (QE)**, así como en los **errores topográficos**.

| Modelo | Nº Iter | Xdim | Ydim | Alfa | Radio | QE | Nodedist | Bmu |
|--------|---------|------|------|----------|-------------------|-----------|----------|----------|
| 1 | 1000 | 50 | 40 | 0.2,0.02 | Default(26.96294) | 0.2687803 | 5.49579 | 6.12549 |
| 2 | 1000 | 50 | 40 | 0.1,0.02 | Default(26.96294) | 0.2952683 | 5.48125 | 6.14257 |
| 3 | 1000 | 50 | 40 | 0.8,0.02 | Default(26.96294) | 0.2221283 | 5.48813 | 6.12848 |
| 4 | 1000 | 50 | 40 | 1,0.02 | Default(26.96294) | 0.2279788 | 5.48795 | 6.13255 |
| 5 | 1000 | 50 | 40 | 0.9,0.02 | Default(26.96294) | 0.2255058 | 5.47524 | 6.12548 |
| 6 | 1000 | 50 | 40 | 0.8,0.01 | Default(26.96294) | 0.2273528 | 5.48951 | 6.12579 |
| 7 | 1000 | 50 | 40 | 0.8,0.03 | Default(26.96294) | 0.2265198 | 5.49547 | 6.13547 |
| 8 | 1000 | 50 | 40 | 0.8,0.02 | 25 | 0.230937 | 5.56841 | 6.02146 |
| 9 | 2000 | 50 | 40 | 0.8,0.02 | Default(26.96294) | 0.2273022 | 6.637683 | 7.892722 |
| 10 | 200 | 50 | 40 | 0.8,0.02 | Default(26.96294) | 0.25558 | 4.336398 | 4.706878 |
| 11 | 10000 | 50 | 40 | 0.8,0.02 | Default(26.96294) | 0.2171209 | 8.418574 | 10.38767 |

Tabla 1. Errores de los modelos probados.

El valor del **radio** se va modificando desde el número dado hasta su valor negativo, cuando la vecindad es menor que uno, solo se actualiza la neurona ganadora. Seleccionando el radio por defecto, este empieza en un valor que cubre las 2/3 partes de todas las distancias neurona a neurona.

En la tabla 2, se observa que el modelo con menor QE obtenido es el modelo 11, pero también presenta unos errores topográficos muy elevados, por otro lado, el modelo con menor error topográfico es el 10. Finalmente se escoge un modelo con errores intermedios, el **modelo 3**.

8.1.1. Características del modelo elegido

U-Matrix

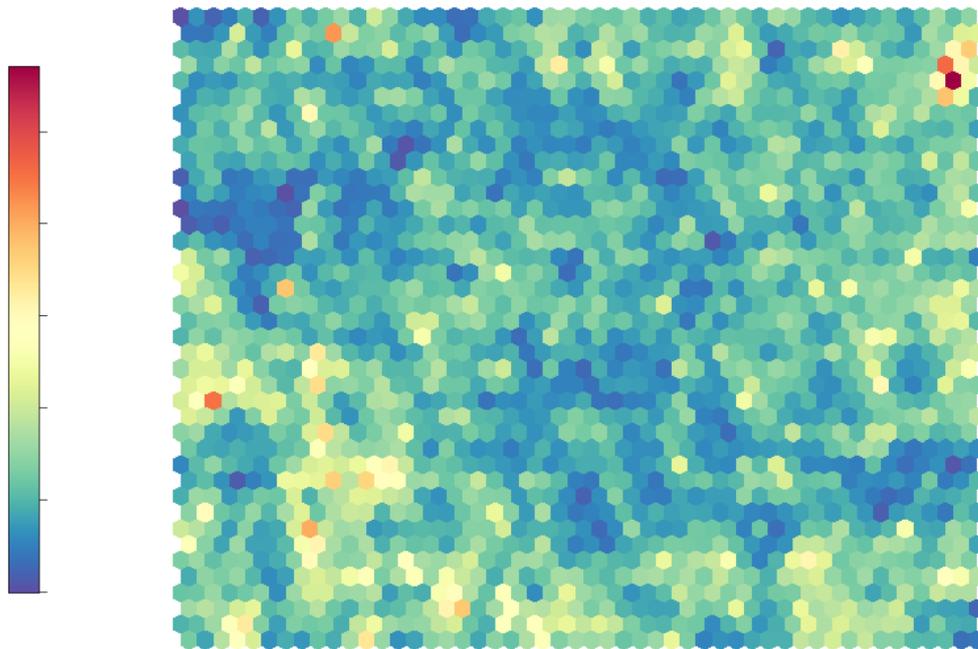


Gráfico 5. U-Matrix del modelo.

La U-Matrix representa la distancia media de cada neurona a las que le rodean. Los colores azules representan menor distancia con las neuronas de alrededor, y las más oscuras (rojizas), representan mayor distancia, es decir, las instancias que se encuentran en estas neuronas se comportan diferente que las de su alrededor.

Destaca una neurona que se encuentra en la parte superior derecha con un tono muy oscuro. Esta neurona contiene instancias que se comportan de manera muy diferente al resto, por esta razón la escala de colores se reduce en gran medida para las demás neuronas.

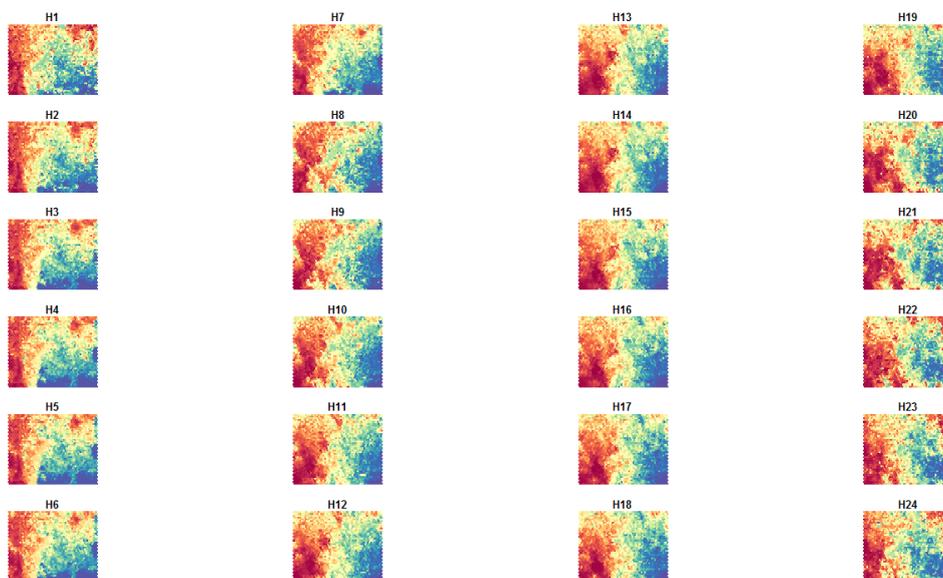


Gráfico 6. Planos de Kohonen de cada una de las variables incluidas en el modelo.

En el gráfico 6 se presentan los **planos de Kohonen** de cada una de las variables incluidas en el modelo. Las neuronas oscuras (colores rojizos) son las que presentan valores más elevados del precio de la electricidad. Por otro lado las neuronas con colores azulados son las que contienen instancias con precios más baratos.

La instancia que en la U-Matrix se encuentre, por ejemplo, en la neurona 1 (esquina inferior izquierda), en cada uno de los planos se encuentra en esa misma neurona.

Todos los planos se parecen en gran medida, y van cambiando gradualmente. Si se compara H12 con H24, se observa que la distribución de los valores varía, habiendo más neuronas con colores rojos oscuros en H24.

Las instancias que se encuentran en la parte inferior izquierda de los planos, son días en los que el precio es más caro a cualquier hora del día, en cambio las que se encuentran en la esquina derecha de la parte inferior, presentan precios más bajos.

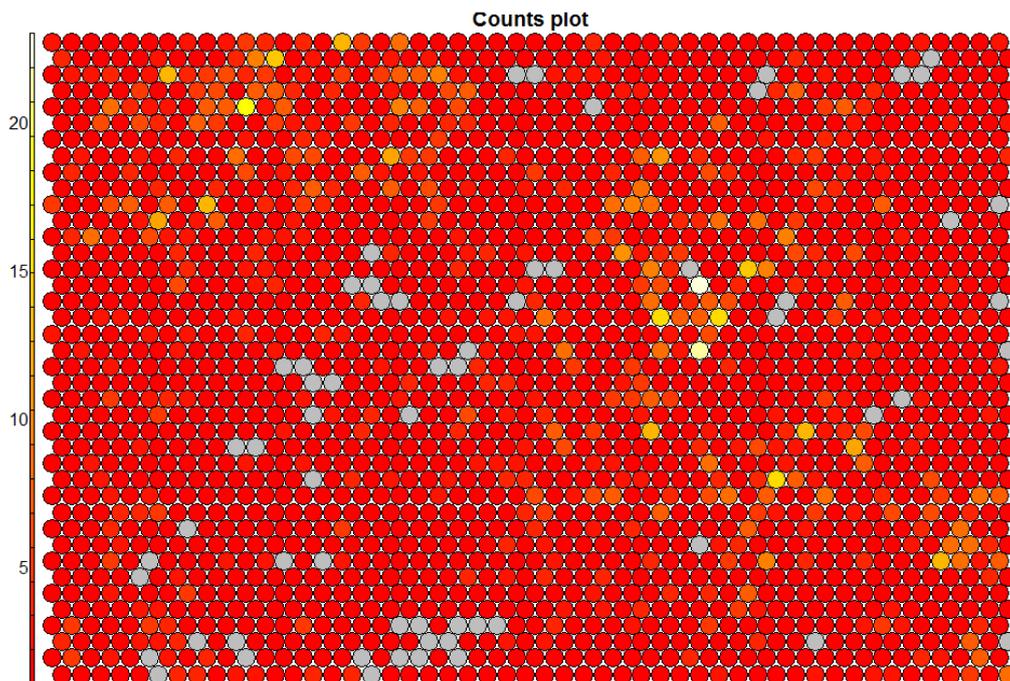


Gráfico 7. Número de instancias por nodo.

En el gráfico 7 se pueden identificar las neuronas con más instancias (blancos y tonos amarillos), así como las neuronas vacías (grises). Cuanto más homogénea sea la distribución de los datos en el mapa mejor será el modelo. En este caso las instancias están bien distribuidas por todo el mapa, siendo 1 el menor número de instancias por neurona y 24 el mayor.

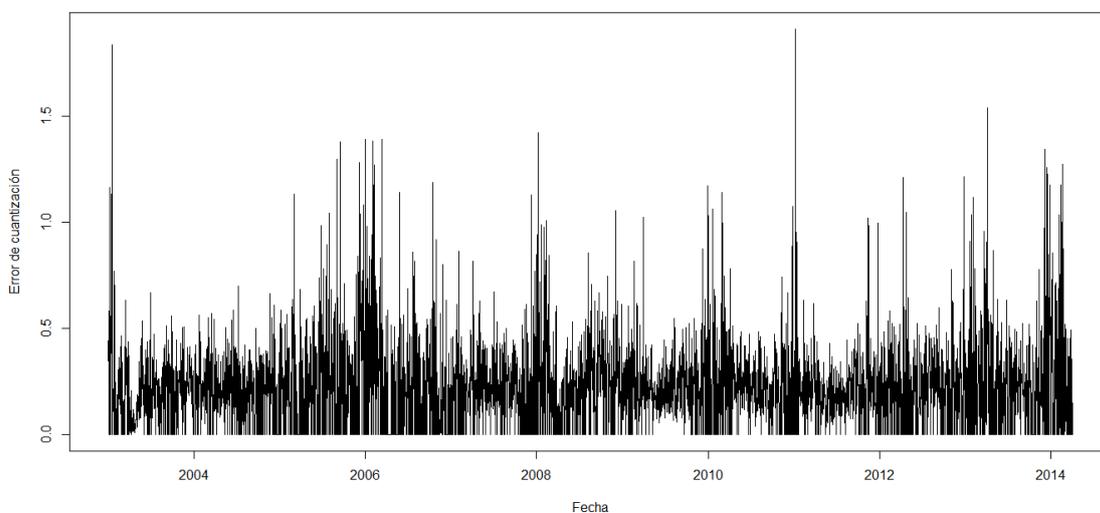


Gráfico 8. Error de cuantización de cada instancia incluida en el modelo.

En el gráfico 8 se observa la distancia euclídea entre cada instancia y el *codevector* de la neurona en la que se encuentra. El error de cuantización del modelo es la media de estas distancias.

Se observa cómo en invierno las distancias son mayores, lo que significa que las predicciones en estos momentos serán peores.

8.1.2. Predicción

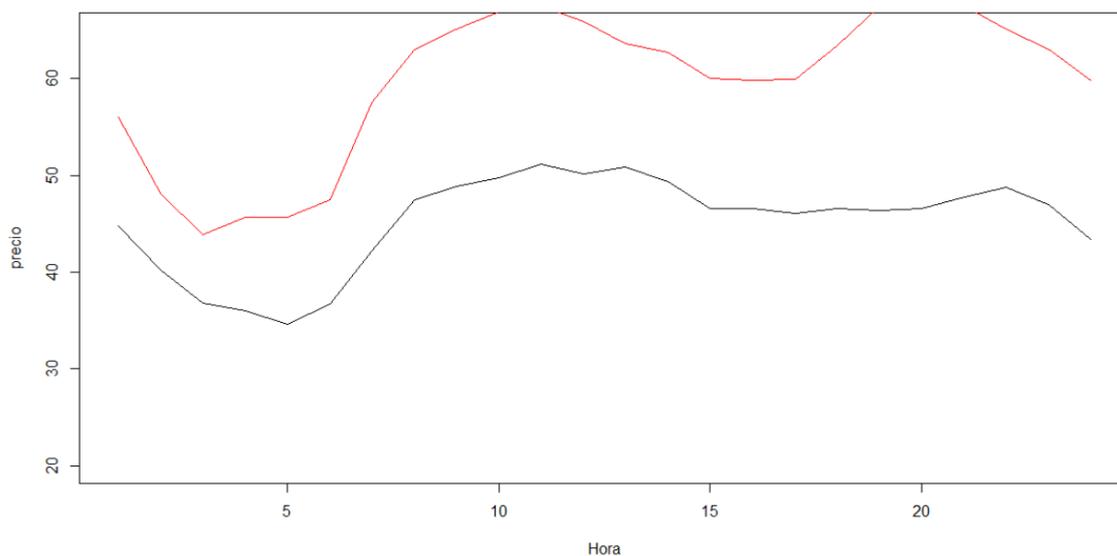


Gráfico 9. Ejemplo de un día de predicción real (negro) y predicho (rojo).

En el gráfico 9 se observa un error de predicción muy alto en todas las horas del día, con una diferencia máxima de 15€ y una diferencia mínima de 7€.

El resto de las predicciones obtenidas de este modelo predicen con un error similar al ejemplo mostrado en el gráfico 9. Por esta razón se decide probar con los datos a partir del 25 de marzo de 2009, que como se ha mencionado anteriormente, son un periodo más favorable a la modelización que los de los primeros años de los que se tienen datos.

8.2. Modelo con datos desde marzo de 2009

Al utilizar menos datos la dimensión del mapa también se debe reducir, por lo tanto las pruebas realizadas a partir de ahora constan de menor número de nodos.

| Modelo | Nº Iter | Xdim | Ydim | Alfa | Radio | QE | Nodedist | Bmu |
|--------|---------|------|------|----------|-------------------|-----------|-----------|-----------|
| 1 | 2000 | 30 | 20 | 0.8,0.02 | Default(15.09967) | 0.4179452 | 10.259874 | 10.324127 |
| 2 | 200 | 30 | 20 | 0.8,0.02 | Default(15.09967) | 0.54766 | 7.355624 | 8.785421 |
| 3 | 1000 | 30 | 20 | 0.8,0.02 | Default(15.09967) | 0.7511175 | 8.872326 | 9.774822 |
| 4 | 1000 | 30 | 20 | 0.8,0.02 | Default(15.09967) | 0.4160533 | 4.968388 | 5.86985 |
| 5 | 10000 | 2 | 2 | 0.8,0.02 | Default(1) | 7.067709 | 1 | 1 |

Tabla 2. Tabla de modelos probados.

En la tabla 3, se observa que el modelo con menor QE obtenido es el modelo 4.

El modelo 5 se realiza con el fin de ver si los datos se agrupaban según días de la semana u otro tipo de agrupamiento más global. Este modelo consta únicamente de 4 neuronas. Como resultado se obtienen unos errores topográficos excesivamente altos y los agrupamientos no son los esperados. Por lo tanto el modelo que se selecciona para hacer la predicción es el **modelo 4**.

8.2.1. Características del modelo elegido

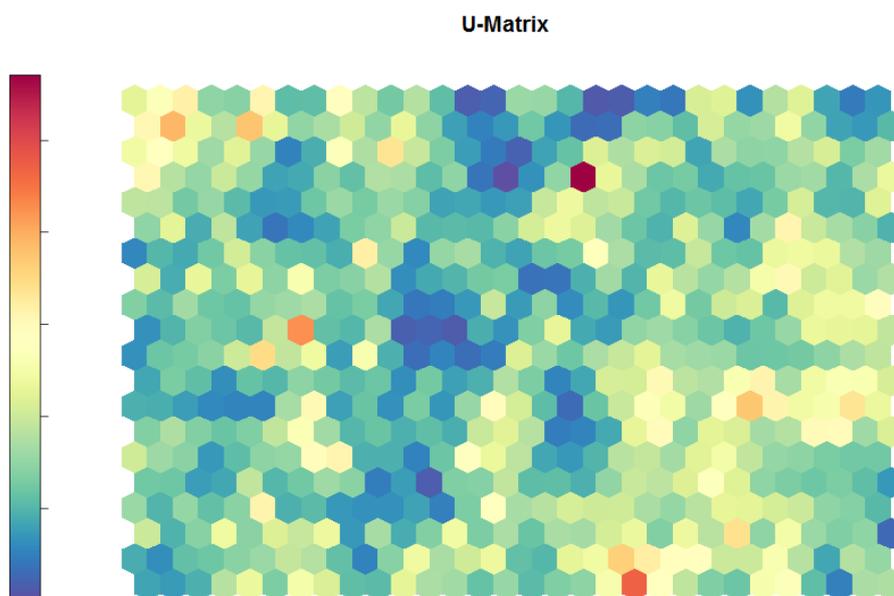


Gráfico 10. U-matrix del modelo.

La U-Matrix de este modelo está más “pixelada” que la mostrada en el gráfico 5, esto es por la menor dimensión del mapa. Los colores no varían tan gradualmente.

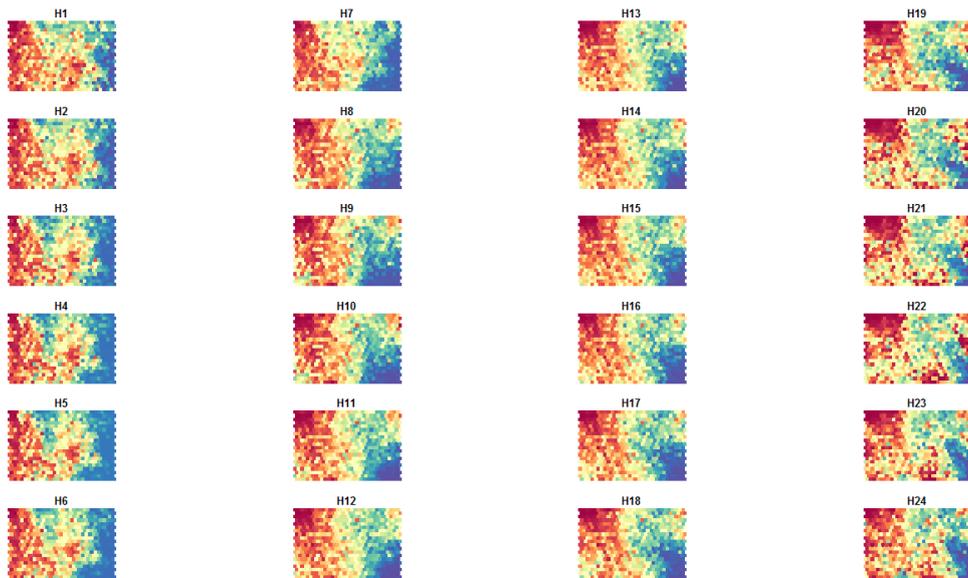


Gráfico 11. Planos de Kohonen de cada una de las variables incluidas en el modelo.

En el gráfico 10 se ve el mismo efecto que en el gráfico 5, los planos van cambiando gradualmente sus tonalidades según se va aumentando la hora. Las últimas horas de la tarde y primeras de la noche son las que tienen precios más elevados en su conjunto y las primeras horas de la mañana son las más baratas.



Gráfico 12. Número de instancias por nodo.

8.2.2. Predicción

Para la **predicción** en la primera prueba se incluyen datos de los días reservados (del 27 de marzo al 2 de abril) hasta las 16h, para predecir las 8 horas posteriores de cada día y en la segunda prueba se incluyen datos de estos días hasta las 19h. De esta manera se ve la diferencia en la predicción al utilizar más información.

Para comparar los errores en la predicción se utilizan errores relativos:

$$Error = \sum_{i=1}^{24} \frac{|\hat{y}_i - y_i|}{y_i}$$

Siendo \hat{y}_i la predicción en la hora i , e y_i el valor real en la hora i .

8.2.2.1. Incluyendo hasta las 16H (inclusive) para hacer la predicción

| Fecha | Día de la semana | Error relativo |
|------------|------------------|----------------|
| 27/03/2014 | Jueves | 2.549915 |
| 28/03/2014 | Viernes | 2.768589 |
| 29/03/2014 | Sábado | 5.137466 |
| 30/03/2014 | Domingo | 6.06153 |
| 31/03/2014 | Lunes | 3.651397 |
| 01/04/2014 | Martes | 12.11031 |
| 02/04/2014 | Miércoles | 2.53586 |

Tabla 3. Errores relativos de la predicción de cada uno de los días.

8.2.2.2. Incluyendo hasta las 19H (inclusive) para hacer la predicción

| Fecha | Día de la semana | Error relativo |
|------------|------------------|----------------|
| 27/03/2014 | Jueves | 2.549915 |
| 28/03/2014 | Viernes | 2.7688589 |
| 29/03/2014 | Sábado | 4.884994 |
| 30/03/2014 | Domingo | 6.06153 |
| 31/03/2014 | Lunes | 3.016961 |
| 01/04/2014 | Martes | 12.38802 |
| 02/04/2014 | Miércoles | 2.53586 |

Tabla 4. Errores relativos de la predicción de cada uno de los días.

Comparando las tablas 3 y 4 se observa que 4 de los 7 días predichos se han incluido en la misma neurona en los dos casos, ya que el error es exactamente el mismo. Dos de los tres errores restantes disminuyen su error al añadir más horas en el modelo, y una predicción lo empeora.

8.3. Modelo con datos de 2012

A continuación utilizamos sólo datos de precios por hora del año 2012

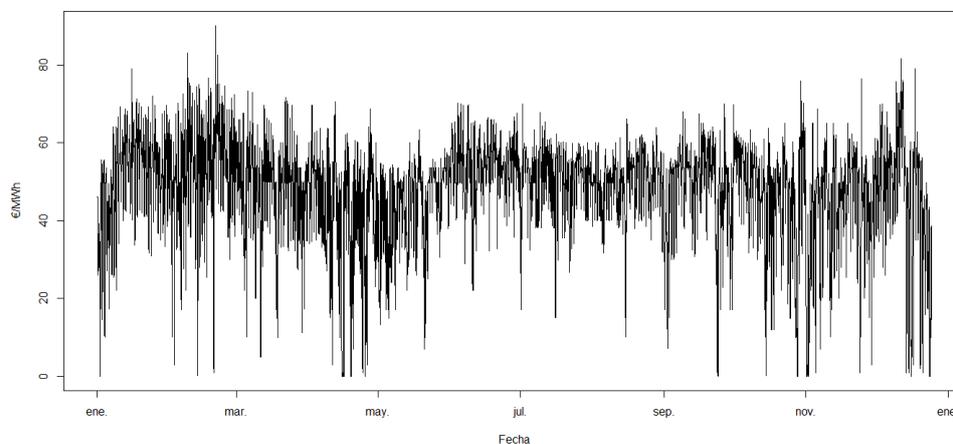


Gráfico 13. Datos horarios de 2012. Datos de entrenamiento.

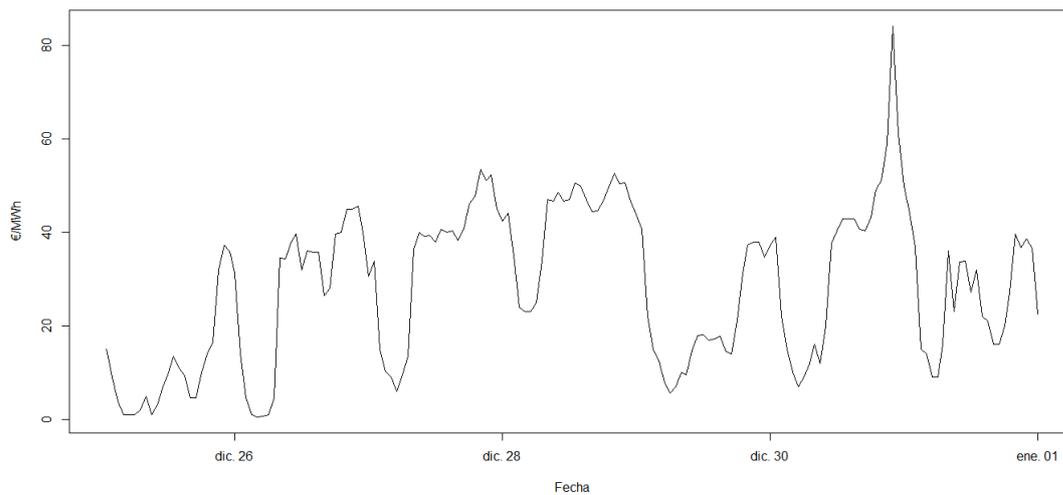


Gráfico 14. Datos horarios de la última semana de 2012. Datos test.

| Modelo | Nº Iter | Xdim | Ydim | Alfa | Radio | QE | NodeDist | Bmu |
|--------|---------|------|------|----------|-------------------|-----------|----------|----------|
| 1 | 1000 | 30 | 10 | 0.8,0.02 | Default(13.22876) | 0.3050869 | 2.386831 | 2.597283 |
| 2 | 1000 | 30 | 10 | 1,0.02 | Default(13.22876) | 0.305412 | 3.465051 | 4.366533 |
| 3 | 1000 | 30 | 10 | 0.5,0.02 | Default(13.22876) | 0.2978662 | 1.845334 | 2.45949 |
| 4 | 10000 | 30 | 10 | 0.5,0.02 | Default(13.22876) | 0.2493407 | 1.769132 | 2.238111 |
| 5 | 10000 | 30 | 10 | 0.1,0.02 | Default(13.22876) | 0.3717596 | 1.662198 | 1.787842 |
| 6 | 10000 | 20 | 10 | 0.1,0.02 | Default(9.165151) | 0.7267258 | 1.540279 | 1.667221 |
| 7 | 10000 | 20 | 10 | 0.5,0.02 | Default(9.165151) | 0.5256219 | 2.460588 | 2.711644 |
| 8 | 10000 | 20 | 10 | 0.8,0.02 | Default(9.165151) | 0.5483184 | 3.347369 | 4.17738 |
| 9 | 10000 | 20 | 15 | 0.8,0.02 | Default(10.53565) | 0.2104314 | 2.194684 | 2.412102 |
| 10 | 10000 | 20 | 15 | 0.5,0.02 | Default(10.53565) | 0.2233712 | 1.78934 | 1.951015 |
| 11 | 10000 | 20 | 15 | 0.3,0.02 | Default(10.53565) | 0.2215798 | 1.890986 | 2.131387 |

Tabla 5. Tabla de modelos probados.

Buscando un equilibrio entre el QE y los errores topográficos se decide elegir el modelo 10.

8.3.1. Características del modelo elegido

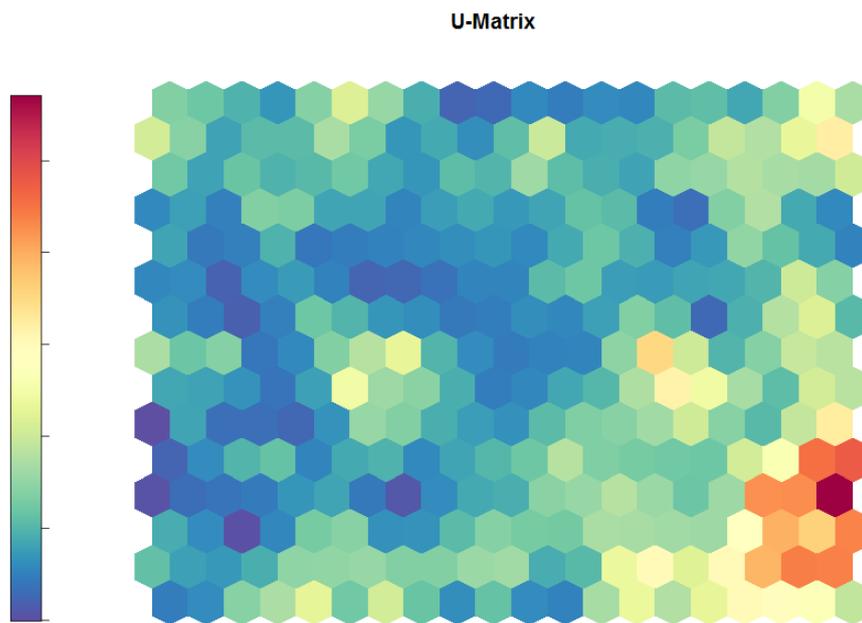


Gráfico 15. U-matrix del modelo.

La U-matrix de este modelo (gráfico 15), presenta una agrupación de neuronas que destacan por sus altas distancias.

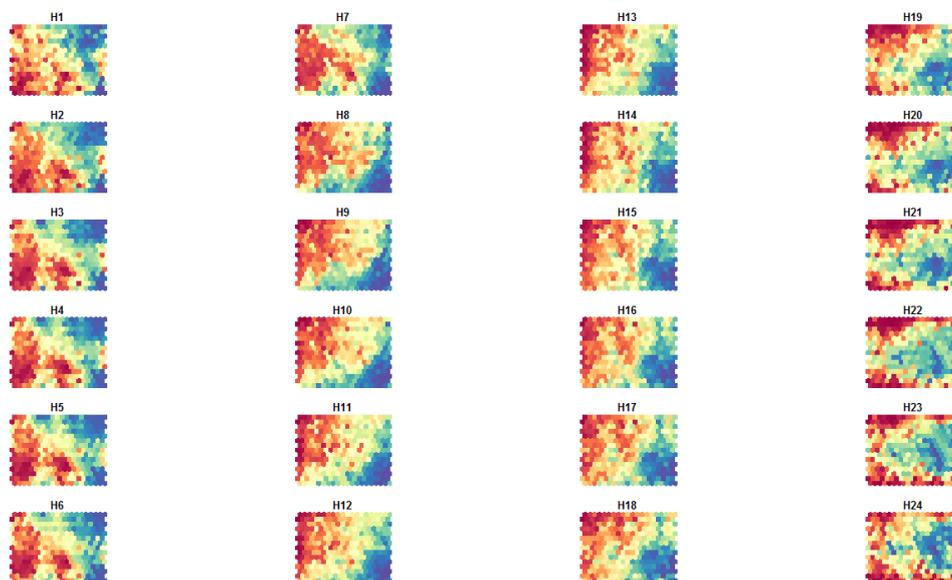


Gráfico 16. Planos de Kohonen de cada una de las variables incluidas en el modelo.

En el gráfico 16 se observa que el grupo de neuronas que en el gráfico 15 tienen altas distancias presenta precios bajos en todas las horas del día.

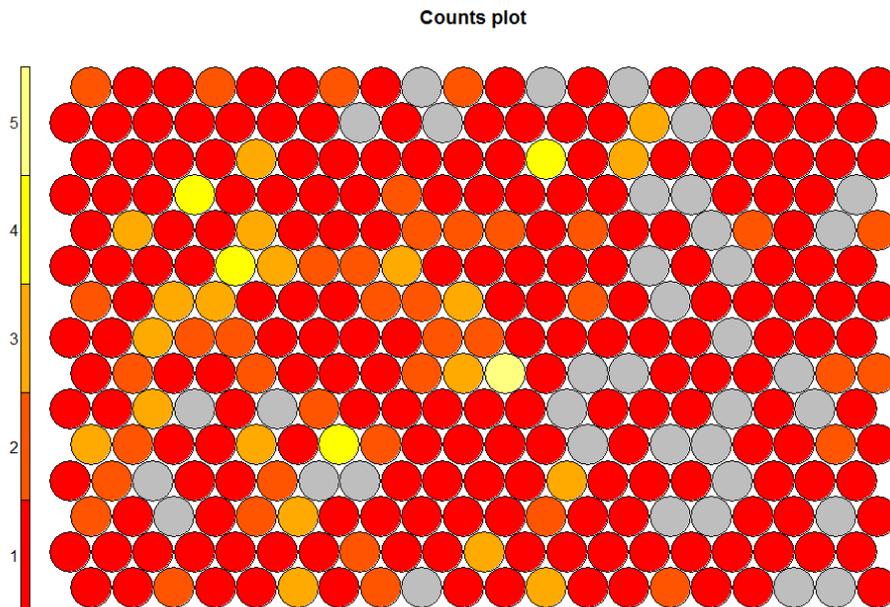


Gráfico 17. Número de instancias por nodo.

En el gráfico 17 se puede apreciar que en la mayor parte de neuronas únicamente hay una instancia, y el máximo número de instancias por neurona es 5.

8.3.2. Predicción

8.3.2.1. Incluyendo hasta las 16H (inclusive) para hacer la predicción

| Fecha | Día de la semana | Error relativo |
|------------|------------------|----------------|
| 25/12/2012 | Martes | 92.06348 |
| 26/12/2012 | Miércoles | 11.7915 |
| 27/12/2012 | Jueves | 3.161893 |
| 28/12/2012 | Viernes | 2.316953 |
| 29/12/2012 | Sábado | 15.69634 |
| 30/12/2012 | Domingo | 6.171875 |
| 31/12/2012 | Lunes | 9.959338 |

Tabla 6. Errores relativos de la predicción de cada uno de los días.

8.3.2.2. Incluyendo hasta las 19H (inclusive) para hacer la predicción

| Fecha | Día de la semana | Error relativo |
|------------|------------------|----------------|
| 25/12/2012 | Martes | 26.06344 |
| 26/12/2012 | Miércoles | 102.1671 |
| 27/12/2012 | Jueves | 3.161893 |
| 28/12/2012 | Viernes | 2.255963 |
| 29/12/2012 | Sábado | 10.93403 |
| 30/12/2012 | Domingo | 19.56767 |
| 31/12/2012 | Lunes | 9.999867 |

Tabla 7. Errores relativos de la predicción de cada uno de los días.

Las predicciones mejoran en 3 de los 7 días al incluir más horas en el modelo. Hay un día que presenta el mismo error y los otros 3 días empeoran la predicción.

En conjunto las predicciones realizadas con el año 2012 únicamente, son peores que las realizadas con el modelo que incluye los datos desde marzo de 2009 hasta abril de 2014.

9. Ajuste de modelos ARIMA

9.1. Modelos con datos desde marzo de 2009

En primer lugar se cambia el formato de los datos para utilizarlos como una serie de frecuencia horaria.

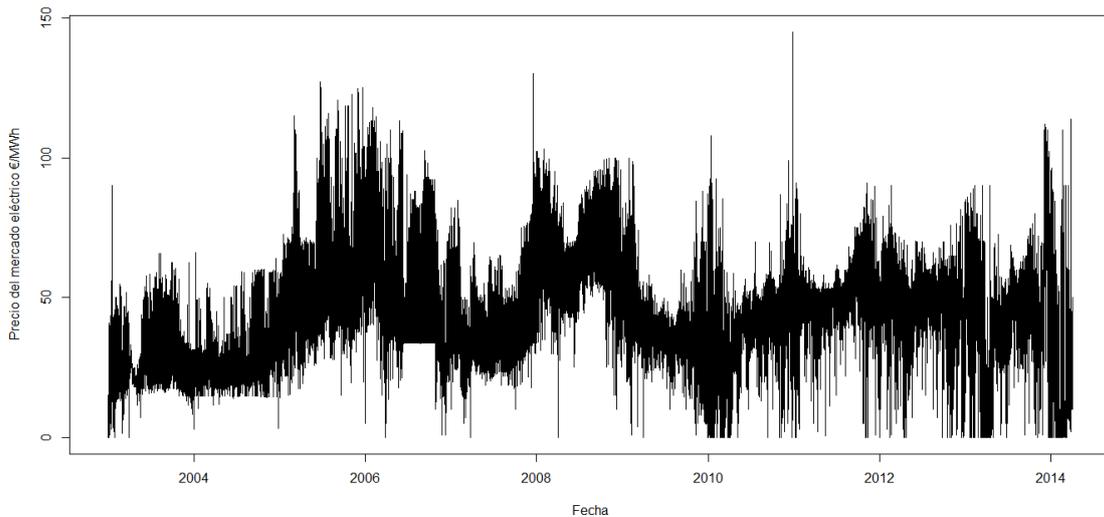


Gráfico 18. Serie de precios horarios del 1 de enero de 2003 hasta el 2 de abril de 2014.

En la serie horaria (gráfico 18), se aprecia mejor la estabilización de los datos a partir de 2009.

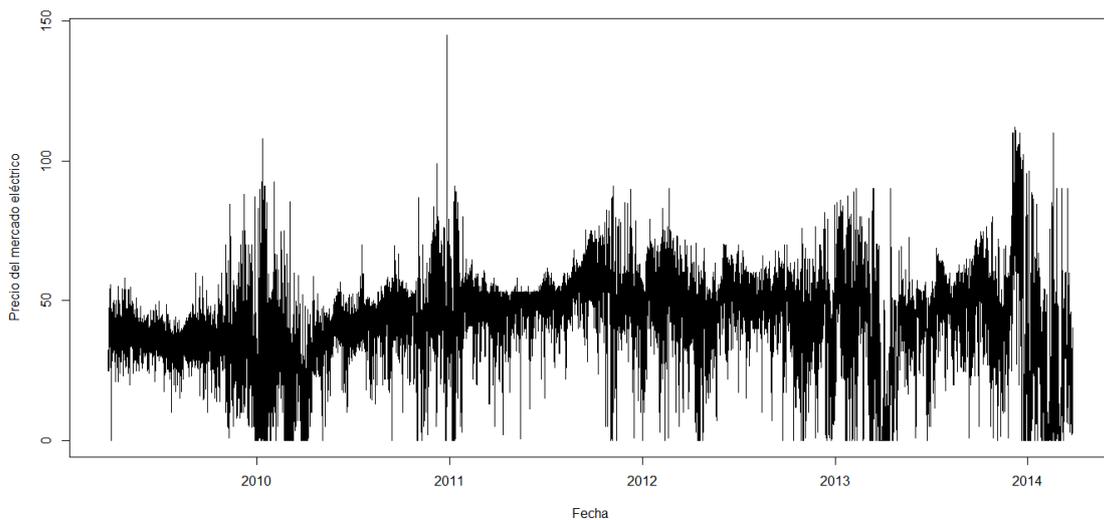


Gráfico 19. Serie de precios horarios del 26 de marzo de 2009 hasta el 25 de marzo de 2014.

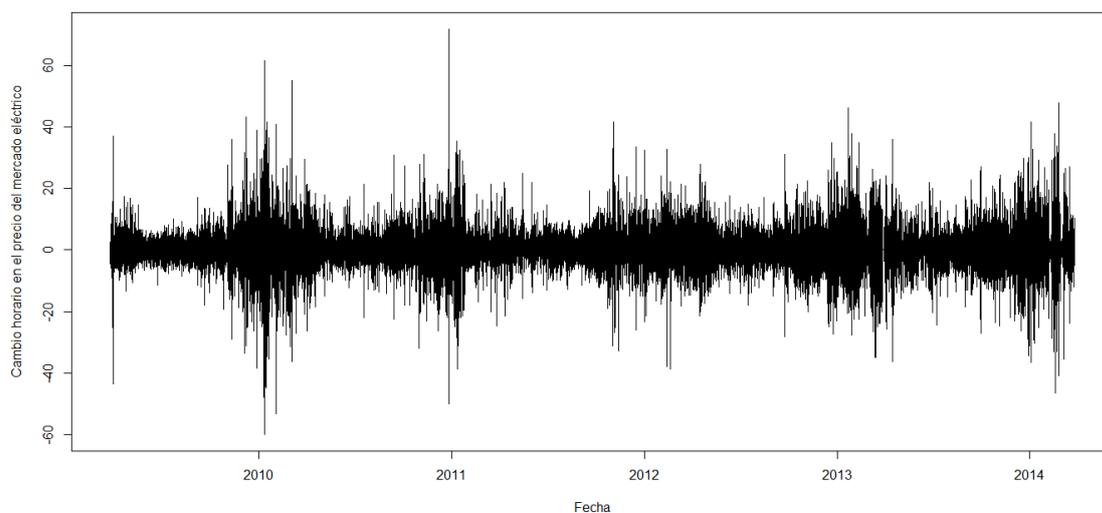


Gráfico 20. Cambio horario en el precio del mercado eléctrico.

Se observa cómo en los días de invierno los precios presentan más variación que en el resto del año.

Para una primera idea se representan las funciones ACF y PACF de la serie en bruto y diferenciada.

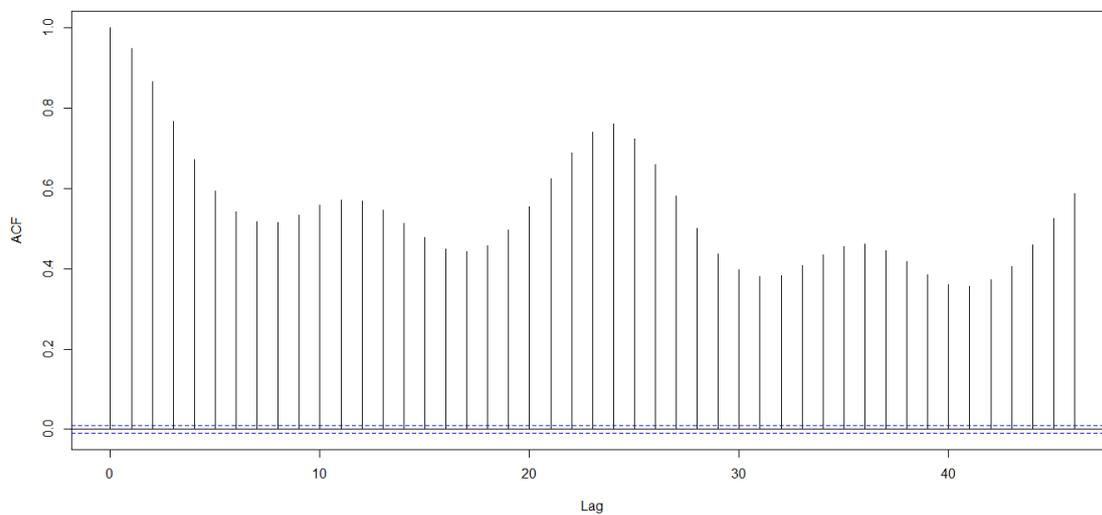


Gráfico 21. Función de autocorrelación de la serie.

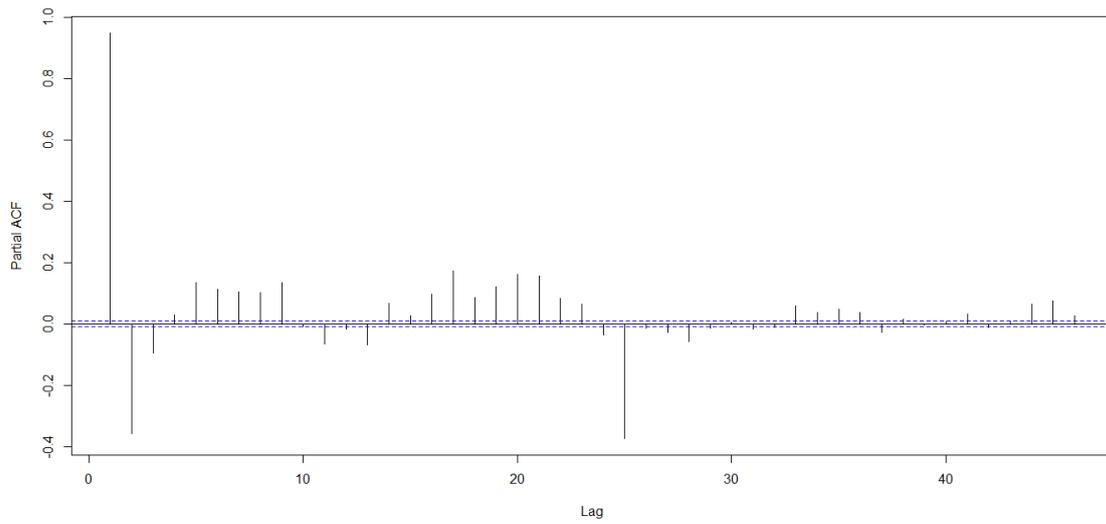


Gráfico 22. Función de autocorrelación parcial de la serie.

ACF de la diferencia de primer grado

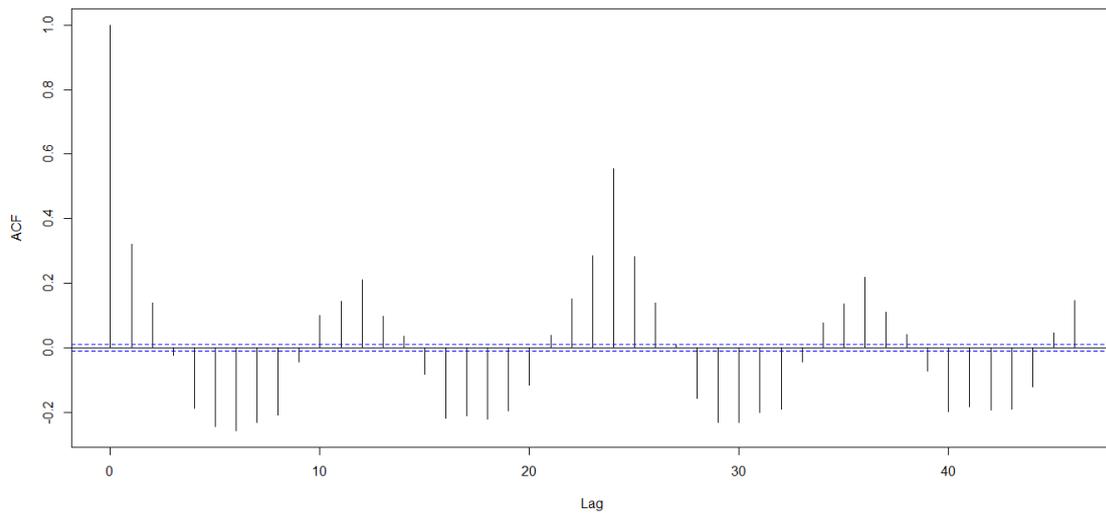


Gráfico 23. Función de autocorrelación de la serie diferenciada.

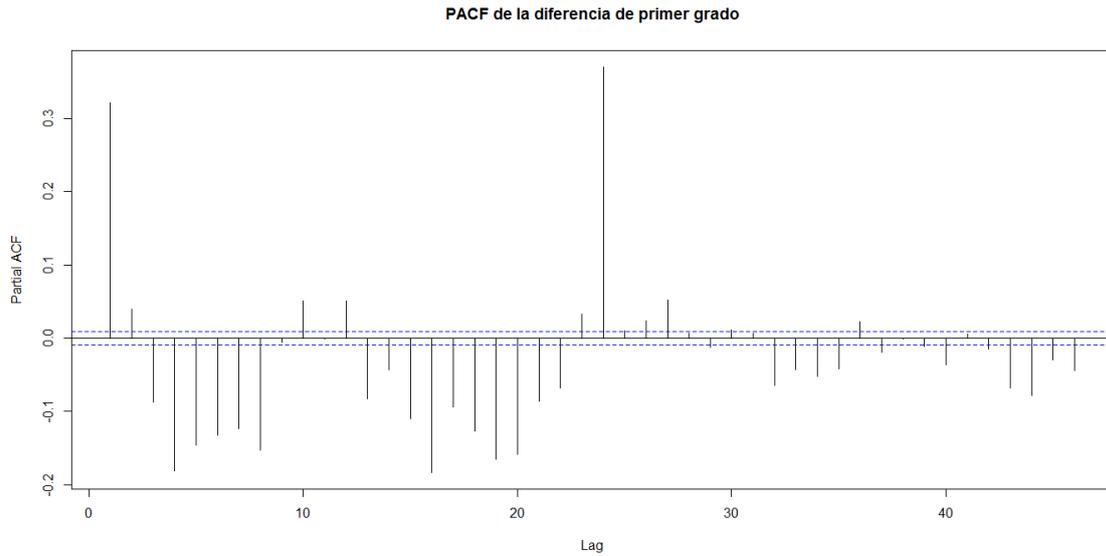


Gráfico 24. Función de autocorrelación parcial de la serie diferenciada.

En el gráfico 21 se observa la estacionalidad día-noche (cada 12 horas) y la diaria.

A continuación se prueban diferentes modelos y se compara sus AICs.

| Modelo ARIMA | AIC |
|---|-----------|
| ARIMA(4,1,3) | 257.816,1 |
| ARIMA(24,1,3) | 277.791,4 |
| ARIMA(24,2,3) | 277.920,4 |
| ARIMA(24,1,7) | 277.595 |
| ARIMA(0,1,1)(0,1,1)₂₄ | 267.839,9 |
| ARIMA(1,1,0)(1,1,0)₂₄ | 280.576,3 |
| ARIMA(1,1,0)(7,1,0)₂₄ | 264.339,7 |

Se representa la función de autocorrelación de los residuos del mejor modelo, el ARIMA (1,1,0)(7,1,0)₂₄, para comprobar si tienen apariencia de ruido blanco.

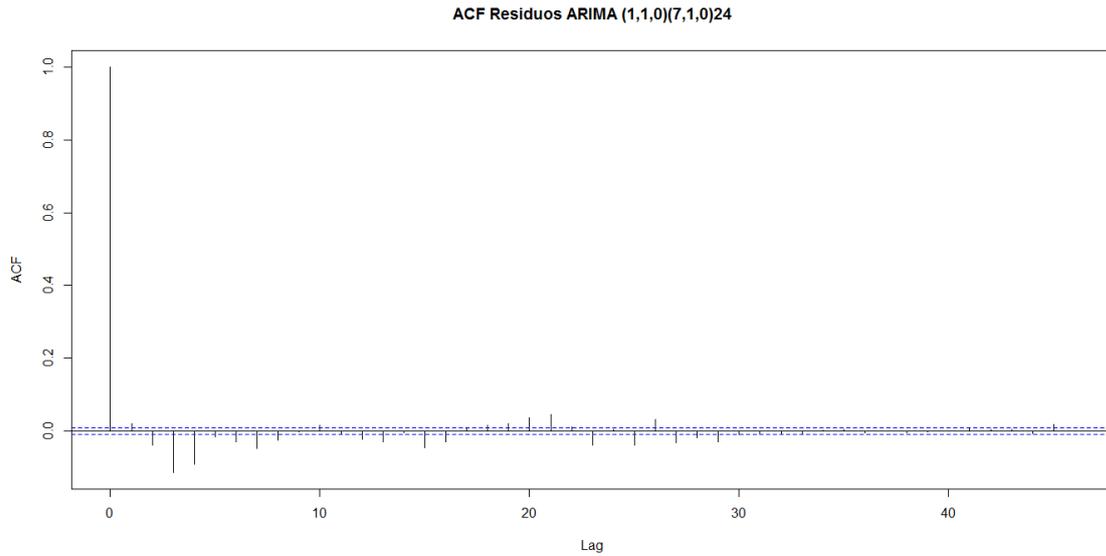


Gráfico 25. Función de autocorrelación de los residuos del modelo ARIMA (1,1,0)(7,1,0)₂₄.

9.2. Modelos con datos de 2012

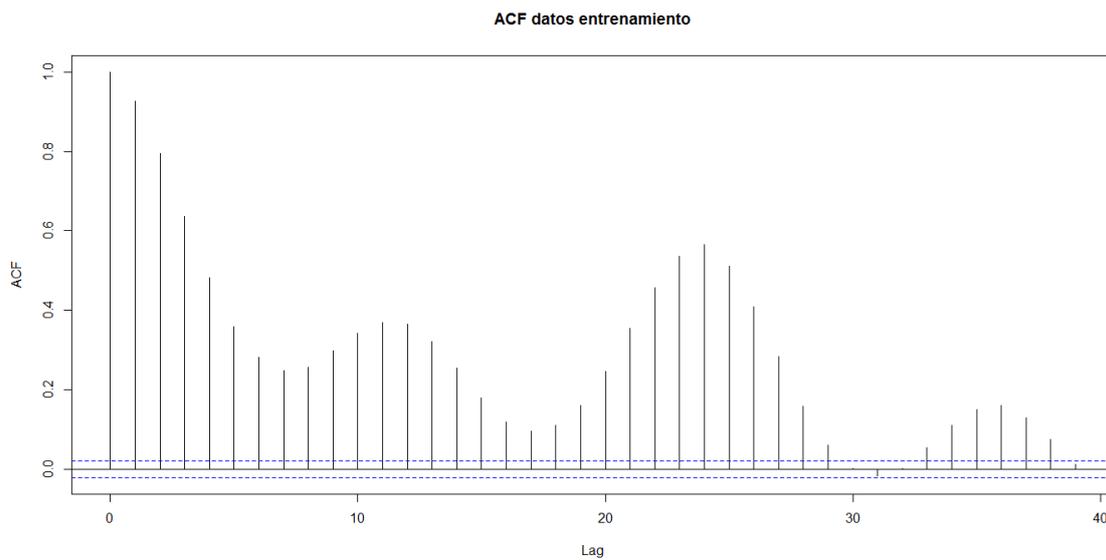


Gráfico 26. ACF de los datos de entrenamiento.

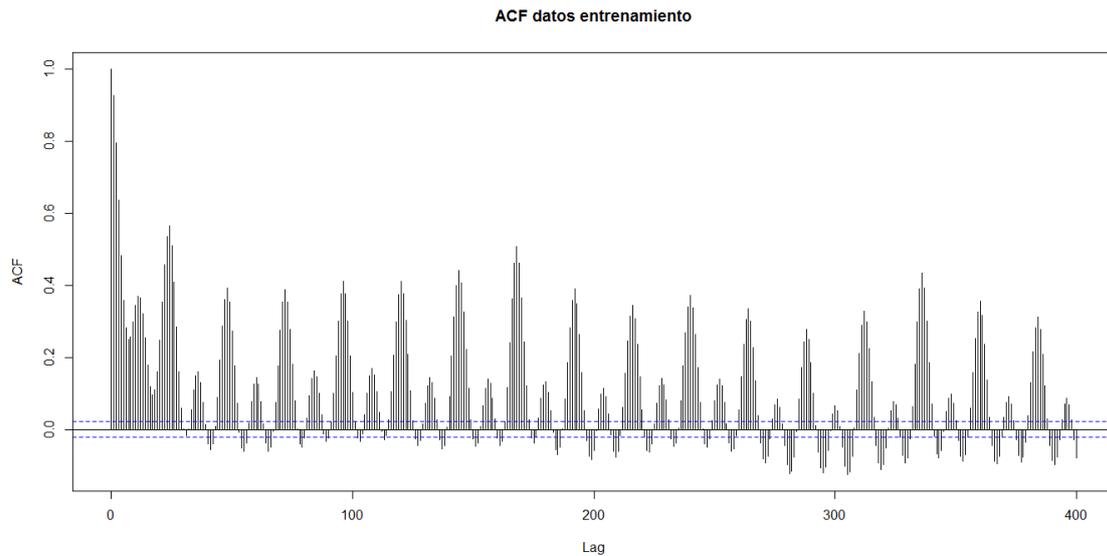


Gráfico 27. ACF de los datos de entrenamiento con 400 desfases.

En los gráficos 26 y 27 se ve como hay estacionalidad cada 12 horas, diaria (cada 24 horas) y, aunque menos marcada, también semanal (cada 7 días).

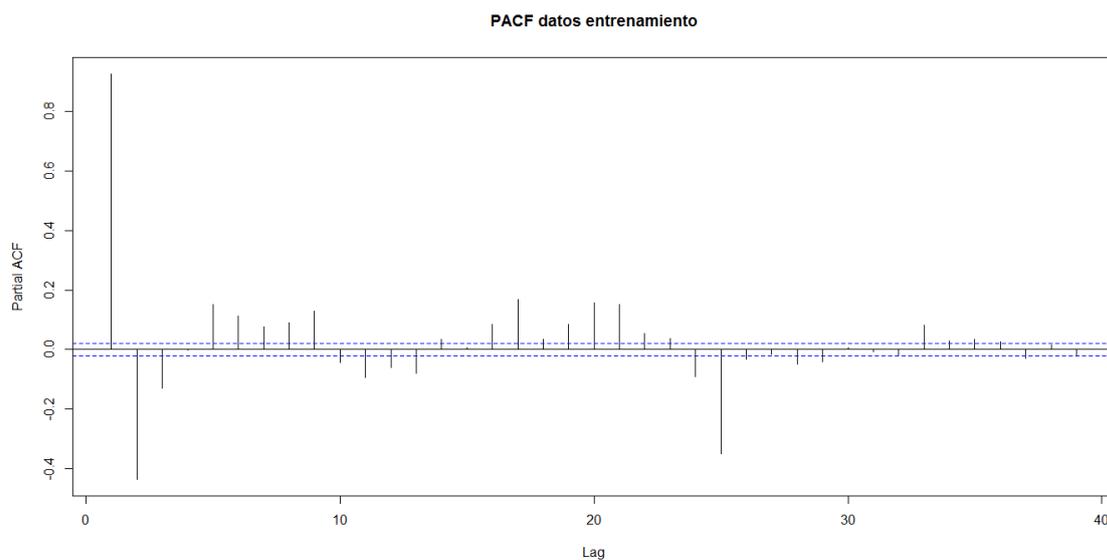


Gráfico 28. PACF de los datos de entrenamiento.

Para empezar a ajustar modelos se utiliza una función del paquete “forecast” de R llamada “auto-arima” que devuelve el mejor modelo ARIMA (de entre una selección) teniendo en cuenta el AIC, AICc o BIC.

Esta función sugiere un modelo ARIMA (4,1,3). Además de este modelo, se prueban otros con diferentes combinaciones de parámetros. De cada uno de estos modelos se observan las funciones de autocorrelación y autocorrelación parcial de sus residuos, además de los valores del AIC, AICc y BIC:

| Modelo | AIC | AICc | BIC |
|--|-----------------|----------------|-----------------|
| ARIMA(4,1,3) | 49239.01 | 49239.02 | 49295.5 |
| ARIMA(4,1,3)(0,1,0)₂₄ | 49925.39 | 49925.41 | 49981.86 |
| ARIMA(4,1,3)(0,1,1)₂₄ | 45472.26 | 45472.28 | 45535.79 |
| ARIMA(4,1,3)(1,1,1)₂₄ | 45341.03 | 45341.05 | 45411.61 |
| ARIMA(4,1,3)(2,1,1)₂₄ | 45395.4 | 45395.44 | 45473.05 |
| ARIMA(4,1,3)(0,1,0)₁₆₈ | 47811.65 | 47811.67 | 47867.99 |
| ARIMA(4,1,3)(0,1,1)₁₆₈ | 44348.08 | 44348.1 | 44411.45 |

Tabla 8. Medidas de error de cada uno de los modelos probados.

Tras valorar cada uno de los modelos mostrados en la tabla 8, se decide optar por el modelo ARIMA (4,1,3)(0,1,1)₁₆₈.

A continuación se obtienen las siguientes predicciones

| Fecha | Día de la semana | Error relativo |
|-------------------|------------------|----------------|
| 25/12/2012 | Martes | 107.3531 |
| 26/12/2012 | Miércoles | 112.1341 |
| 27/12/2012 | Jueves | 4.839952 |
| 28/12/2012 | Viernes | 9.844089 |
| 29/12/2012 | Sábado | 8.920575 |
| 30/12/2012 | Domingo | 8.436704 |
| 31/12/2012 | Lunes | 8.471479 |

Tabla 9. Errores relativos de cada día predicho con cada uno de los modelos ejecutados.

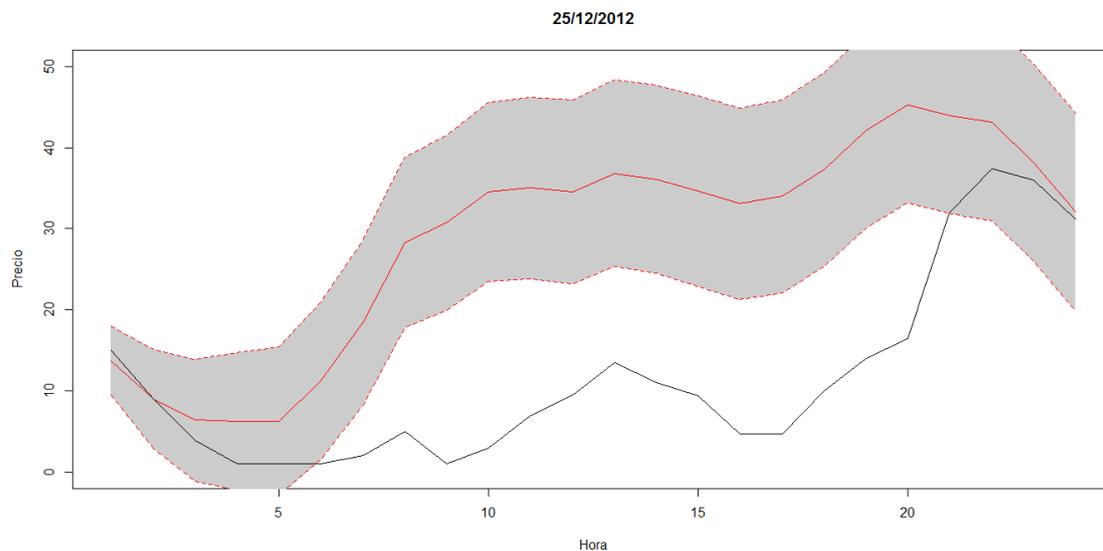


Gráfico 29. Precio del mercado eléctrico el 25/12/2012 real (negro) y predicho (rojo) con bandas de confianza del 80%.

25/12/2012

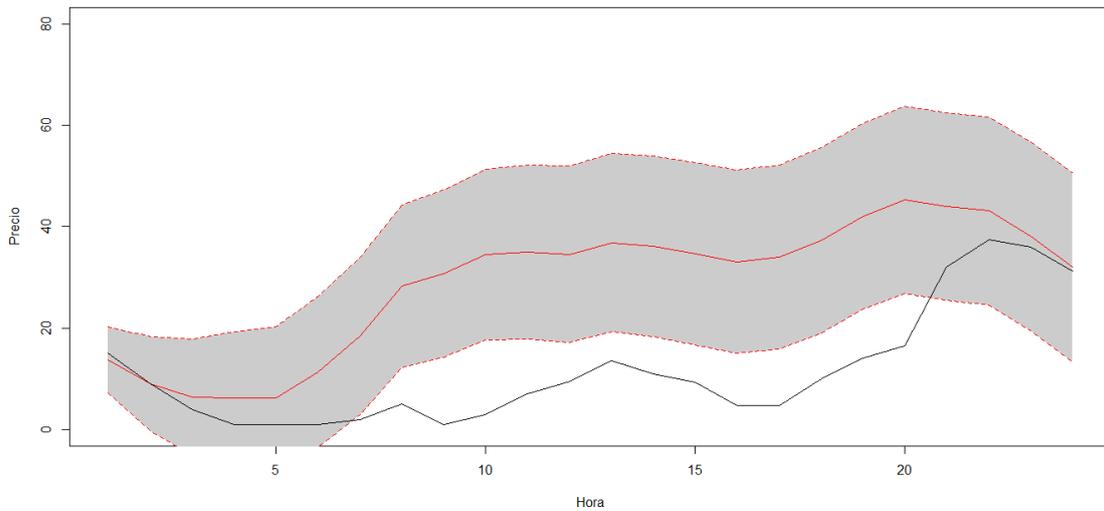


Gráfico 30. Precio del mercado eléctrico el 25/12/2012 real (negro) y predicho (rojo) con bandas de confianza del 95%

10. Comparación SOM vs ARIMA

10.1. Modelo con datos desde marzo de 2009

Características del modelo SOM:

- Dimensión: 30x20
- Iteraciones: 1000
- Alfa: (0.8,0.02)

Modelo ARIMA (1,1,0)(7,1,0)₂₄.

| Fecha | Día de la semana | Error SOM 16H | Error SOM 19H | Error ARIMA |
|-------------------|------------------|---------------|---------------|-------------|
| 25/12/2012 | Martes | 2.549915 | 2.549915 | 104.254558 |
| 26/12/2012 | Miércoles | 2.768589 | 2.7688589 | 85.215684 |
| 27/12/2012 | Jueves | 5.137466 | 4.884994 | 56.321057 |
| 28/12/2012 | Viernes | 6.06153 | 6.06153 | 10.576843 |
| 29/12/2012 | Sábado | 3.651397 | 3.016961 | 11.547449 |
| 30/12/2012 | Domingo | 12.11031 | 12.38802 | 90.142407 |
| 31/12/2012 | Lunes | 2.53586 | 2.53586 | 8.258379 |

Tabla 10. Errores relativos de cada día predicho con cada uno de los modelos ejecutados.

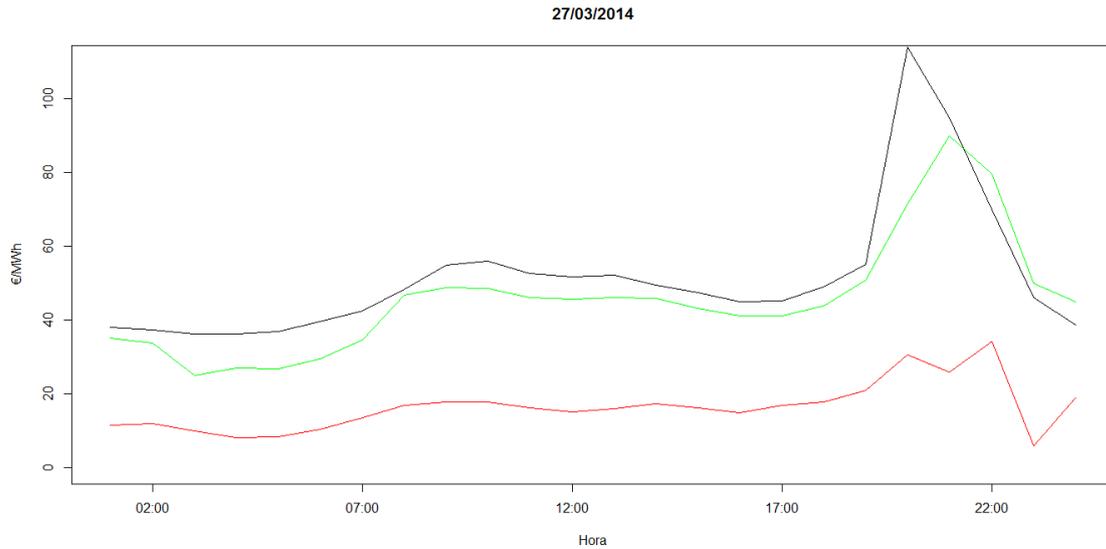


Gráfico 31. Precio del mercado eléctrico real (negro), predicho con SOM (verde) y predicho con ARIMA (rojo).

10.2. Modelo con datos de 2012

Características del modelo SOM:

- Dimensión: 20x15
- Iteraciones: 10000
- Alfa: (0.5,0.02)

Modelo ARIMA (4,1,3)(0,1,1)₁₆₈.

| Fecha | Día de la semana | Error SOM 16H | Error SOM 19H | Error ARIMA |
|-------------------|------------------|---------------|---------------|-------------|
| 25/12/2012 | Martes | 92.06348 | 26.06344 | 107.3531 |
| 26/12/2012 | Miércoles | 11.7915 | 102.1671 | 112.1341 |
| 27/12/2012 | Jueves | 3.161893 | 3.161893 | 4.839952 |
| 28/12/2012 | Viernes | 2.316953 | 2.316953 | 9.844089 |
| 29/12/2012 | Sábado | 15.69634 | 10.93403 | 8.920575 |
| 30/12/2012 | Domingo | 6.171875 | 19.56767 | 8.436704 |
| 31/12/2012 | Lunes | 9.959338 | 9.999867 | 8.471479 |

Tabla 11. Errores relativos de cada día predicho con cada uno de los modelos ejecutados.

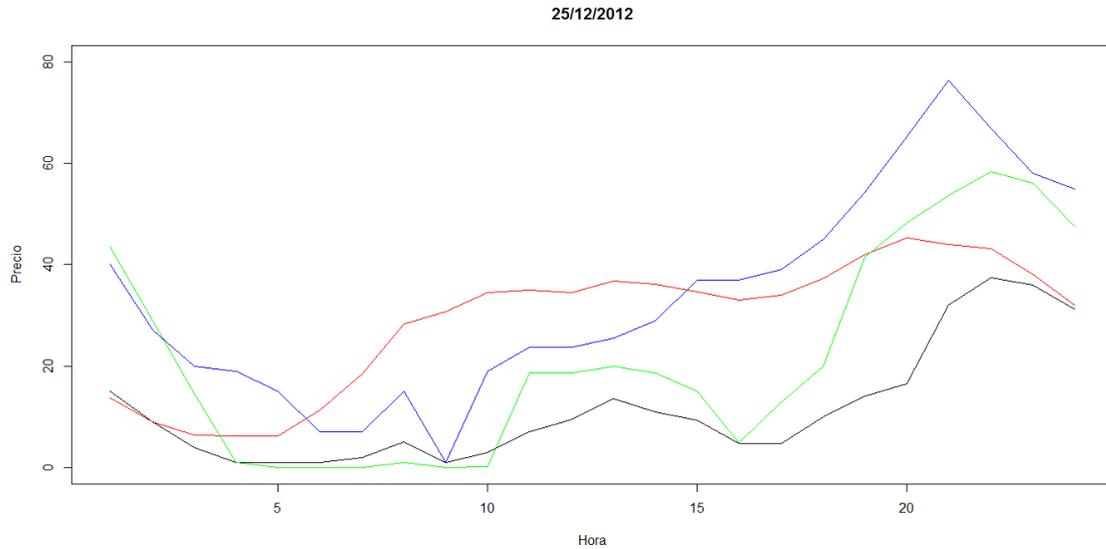


Gráfico 32. Precio del mercado eléctrico real (negro), predicho con SOM hasta 16H(azul), hasta 19H(verde), y predicho con ARIMA (rojo). 25/12/2012.

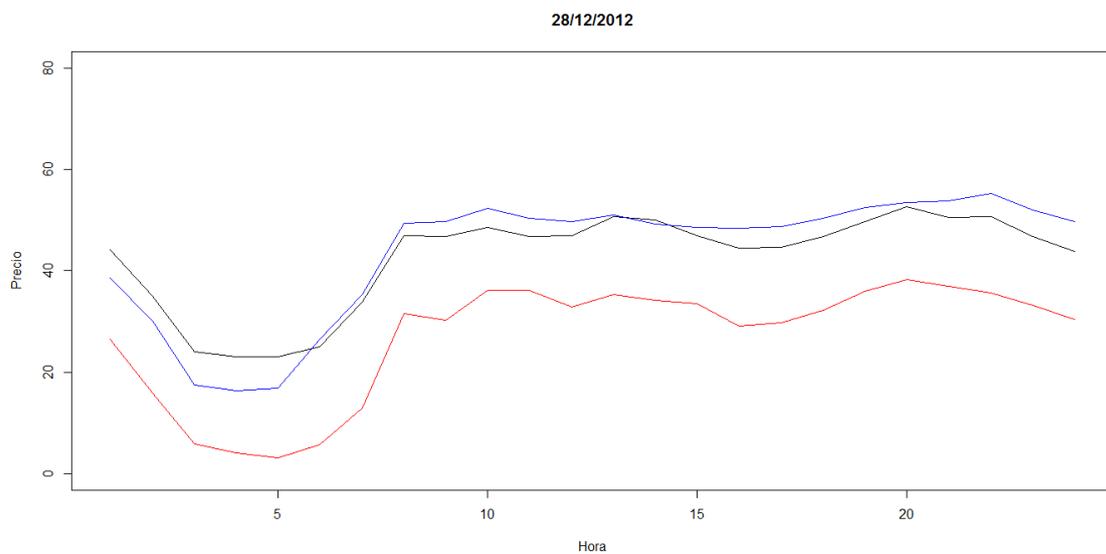


Gráfico 33. Precio del mercado eléctrico real (negro), predicho con SOM hasta 16H(azul), hasta 19H(verde), y predicho con ARIMA (rojo). 28/12/2012. La línea azul y la verde están superpuestas.

11. Conclusiones

La conclusión más clara que se ha podido sacar es la dificultad que supone el logro de un buen modelo de predicción del precio del mercado eléctrico a corto plazo, la alta volatilidad de los datos dificulta esta tarea en gran medida. Por otro lado, la ejecución de modelos ARIMA con la librería utilizada, ha resultado especialmente costosa debido a su lentitud.

Finalmente, las predicciones obtenidas tanto con el método SOM como con los modelos ARIMA no han sido todo lo precisas que nos hubiera gustado. Pero si hay que decantarse por uno de los dos modelos, creemos que el método SOM es el que genera mejores predicciones a corto plazo.

Además, se ha comprobado que el modelo SOM con más datos históricos genera mejores predicciones que el realizado con los datos de un único año. A diferencia de lo que ocurre en el modelo ARIMA.

12. Posibles continuaciones

Este trabajo puede tener numerosas vertientes. En primer lugar, cabría comentar las diferentes maneras en las que se pueden dividir los datos a la hora de buscar un modelo adecuado que explique la serie. Una opción sería separar los días laborables del resto, ya que el precio de la electricidad depende de la demanda y está comprobado que en días festivos la demanda se reduce. Otra opción es dividir los días por horas, es decir, hacer diferentes modelos según una serie de fragmentos horarios. En esta opción se agruparían las horas que tengan comportamientos más similares.

Además de las divisiones de los datos, se pueden probar otros algoritmos como redes neuronales, SVM o Random Forest entre otros.

Otra cuestión que se podría añadir al estudio es la inclusión en el modelo de nuevas variables que pudieran afectar al precio del mercado eléctrico como la velocidad del viento, las horas de sol, la temperatura ambiente, reservas hidráulicas de los embalses, etc.

13. Bibliografía

- Barreto, G.A., (2008). Time Series Prediction with the Self-Organizing Map: A Review. *Studies Comput. Intell.*, vol. 77, pp. 135-158.
- Bohari, Z.H., Azemy, H.S., Nasir, M.N.M., Baharom, M.F., Sulaima, M.F., Jali, M.H. (2015). Reliable short term load forecasting using Self Organizing Map (SOM) in deregulated electricity market. *Journal of Theoretical and Applied Information Technology* Vol.9 No 3.
- Carpinteiro, O.A.S., Alves da Silva, A.P. (2001). A Hierarchical Self-Organizing Map Model in Short-Term Load Forecasting. *Journal of Intelligent and Robotic Systems* 31, pp. 105-113
- Domínguez, M., Fuertes, J.J., Díaz I., Cuadrado, A.A., Alonso, S., Morán, A. (2011). Analysis of electric power consumption using Self-Organizing Maps. *The 18th IFAC World Congress Milano (Italy)*.
- Espinola Vílchez, R. (2004). Predicción a corto plazo de los precios de la energía eléctrica. Tesis doctoral. Universidad de Castilla – La Mancha.
- Kalinic, H., Matić, F., Mihanović, H., Vilibić, I., Zagar, N., Jasenko, B., Tudor, M. (2015). Comparison of two meteorological models using self-organizing maps. *OCEANS 2015 Genova*, pp 1-6
- Kumar Aggarwal, S., Mohan Saini, L., Kumar, A. (2009). Electricity price forecasting in deregulated markets: A review and evaluation. *International Journal of Electrical power and Energy Systems*. Vol. 31, pp. 13-22.
- Li, Y., Pan, F. (2013). Application of Improved SOM Neural Network in Manufacturing Process Quality Control. *Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering*.
- Martínez-Álvarez, F., Troncoso, A., Asencio-Cortés, G., Riquelme, J.C. (2015). A Survey on Data Mining Techniques Applied to Electricity-Related Time Series Forecasting. *Energies*, vol. 8, pp. 13162-13193.
- Martínez-Álvarez, F., Troncoso, A., Riquelme, J.C., Aguilar-Ruiz, J.S. (2010). Energy Time Series Forecasting Based on Pattern Sequence Similarity. *IEEE Transactions on Knowledge and Data Engineering*, Vol. 23, pp. 1230-1243.
- Martínez Álvarez, F. Riquelme Santos, J.C. (2007). Análisis de las series temporales de los precios del mercado eléctrico mediante técnicas de clustering. Tesis doctoral Universidad de Sevilla.
- Muñoz, A., Cruz, A., Zamora, J.L., Espínola, R. (2009). Modelos de predicción a corto plazo del precio del mercado eléctrico español. Universidad pontificia Comillas de Madrid
- Nagi, J., Yap, K.S., Tiong, S.K., Ahmed, S.K. (2008). Electrical Power Load Forecasting using Hybrid Self-Organizing Maps and Support Vector Machines. *The 2nd International Power Engineering and Optimization Conference*. Pp. 51-56.

Pözlbauer, G. Survey and Comparison of Quality Measures for Self-Organizing Maps.
 Recuperado 21/05/2016 de <http://www.ifs.tuwien.ac.at/~poelzbauer/publications/Poe04WDA.pdf>

Reston Filho, J.C., Affonso, C.M., L. de Oliveira, C.L. (2012). Short-Term Energy Price Prediction Multi-Step-Ahead in the Brazilian Market Using Data Mining. *Advances in Data Mining Discovery and Applications (Chapter 10)*, pp. 219-234

Valero, S., Senabre, C., López, M., Aparicio, J., Gabladón, A., Ortiz, M. (2012). Comparison of Electric Load Forecasting between Using SOM and MLP Neural Network. *Journal of Energy and Power Engineering 6*, pp. 411-417.

Yan, X. (2014). Forecasting Mid-Term Electricity Market Clearing Price Using Support Vector Machines. *IET Generation, Transmission & Distribution. Vol. 8*, pp. 1572-1582

14. Anexos

I. Datos utilizados

| | FECHA | ANIO | MES | DIASEM | DIA | H1 | H2 | H3 | H4 | H5 | H6 | H7 | H8 | H9 | H10 | H11 | H12 |
|---|------------|-------|-------|--------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-----------|-------|-------|
| 1 | 01/01/2003 | 2003 | 1 | M | 1 | 15.00 | 10.97 | 8.22 | 5.24 | 2.65 | 2.13 | 2.06 | 0.02 | 0.00 | 0.00 | 0.77 | 2.10 |
| 2 | 02/01/2003 | 2003 | 1 | J | 2 | 8.33 | 4.20 | 2.87 | 2.63 | 2.56 | 2.56 | 3.51 | 5.15 | 10.00 | 17.17 | 20.00 | 21.02 |
| 3 | 03/01/2003 | 2003 | 1 | V | 3 | 14.27 | 9.47 | 5.08 | 3.57 | 3.01 | 3.01 | 4.61 | 9.41 | 12.83 | 16.27 | 17.62 | 19.66 |
| 4 | 04/01/2003 | 2003 | 1 | S | 4 | 30.36 | 15.27 | 11.11 | 8.99 | 6.34 | 5.15 | 4.68 | 5.15 | 8.24 | 12.83 | 16.10 | 17.62 |
| 5 | 05/01/2003 | 2003 | 1 | D | 5 | 17.57 | 11.11 | 0.97 | 3.50 | 0.87 | 1.07 | 2.51 | 0.87 | 1.47 | 3.57 | 1.47 | 6.47 |
| 6 | 06/01/2003 | 2003 | 1 | L | 6 | 17.62 | 12.84 | 5.37 | 6.17 | 4.50 | 4.03 | 4.03 | 2.97 | 5.08 | 8.47 | 9.47 | 15.02 |
| | H13 | H14 | H15 | H16 | H17 | H18 | H19 | H20 | H21 | H22 | H23 | H24 | MAX | MIN | AVG | | |
| 1 | 3.50 | 5.33 | 6.33 | 4.20 | 4.20 | 5.15 | 12.00 | 15.40 | 15.50 | 16.27 | 15.02 | 15.42 | 16.27 | 0.00 | 6.978333 | | |
| 2 | 21.02 | 20.00 | 17.62 | 17.62 | 18.01 | 20.46 | 40.81 | 39.95 | 21.00 | 19.51 | 17.00 | 16.61 | 40.81 | 2.56 | 15.400417 | | |
| 3 | 19.60 | 17.62 | 16.20 | 16.11 | 16.50 | 17.62 | 30.36 | 30.36 | 25.00 | 19.73 | 17.50 | 17.43 | 30.36 | 3.01 | 15.118333 | | |
| 4 | 17.62 | 16.18 | 15.50 | 14.75 | 14.43 | 15.75 | 19.83 | 26.00 | 26.57 | 19.92 | 18.30 | 19.61 | 30.36 | 4.68 | 15.262500 | | |
| 5 | 12.00 | 12.82 | 12.00 | 6.17 | 5.57 | 12.00 | 17.62 | 17.62 | 18.50 | 20.00 | 18.50 | 20.00 | 20.00 | 0.87 | 9.343750 | | |
| 6 | 16.40 | 15.25 | 15.00 | 10.87 | 9.27 | 12.00 | 21.00 | 30.36 | 34.22 | 39.97 | 32.54 | 33.39 | 39.97 | 2.97 | 15.243333 | | |

Tabla 12. Cabecera de los datos utilizados.

II. Scripts R

Lectura de datos

```
library(kohonen)
data<-read.csv("2003-2014Bienv1.csv", sep=";")
test<-read.csv("2015v1.csv",sep=";")
```

Descriptivas

```
Fecha<-strptime(data$FECHA, "%d/%m/%Y")
plot(Fecha,data[,32], type="l", ylab="€/MWh medio diario")
```

```
Fecha1<-strptime(test$FECHA,"%d/%m/%Y")
plot(Fecha1,test[,32], type="l", ylab="€/MWh medio diario", xlab="Fecha")
```

```
summary(data)
summary(test)
data2012<-data[3288:3653,]
boxplot(data2012[,c(1:5,30:32)], xlab="Hora",ylab="€/MWh")
```

Ajuste de modelo SOM

```
library(kohonen)
setwd("~/Grado de Estadística/2015/TFG/datos")
data<-read.csv("2003-2014Bienv1.csv", sep=";")
names(data)
```

```

#----Descriptivas datos de entrenamiento-----
Fecha<-strptime(data$FECHA, "%d/%m/%Y")
plot(Fecha[2284:length(Fecha)],data[2284:length(Fecha),32
], type="l", ylab="Precio")
plot(Fecha[2284:length(Fecha)],data[2284:length(Fecha),8],
type="l", ylab="Precio")
#-----Filtrar variables-----
d<-data[2284:4103,-c(2:5,30:32)] #Desde abril de 2009
hasta final (menos una semana)
summary(d)
d<-na.omit(d)
d1<-d[,-1]

d_matrix <- as.matrix(scale(d1))
#-----Modelo SOM-----

aGrid <- somgrid(xdim =30, ydim = 20, topo="hexagonal")
set.seed(15324)
system.time(aSom <- som(data=d_matrix,
grid=aGrid,rlen=1000, alpha=c(0.8,0.02),keep.data=TRUE))
names(aSom)
aSom$radius
plot(aSom, type="counts")
plot(aSom, type="dist.neighbours")
s<-seq(1,600,1)
w<-cbind(s,aSom$grid$pts)
#-----Datos Test-----

test<-data[4104:length(Fecha),-c(2,3,5,30:32)]
test<-data[4104:length(Fecha),-c(2,3,5,30:32)]
names(test)

t1<-na.omit(test)
t2<-t1[,-c(1,2)]
dtest<-as.matrix(scale(t2,center= attr(d_matrix,
"scaled:center"),
scale=attr(d_matrix,"scaled:scale")))
#----se incluyen hasta las 16H para la predicción-----
summary(dtest)
set.seed(123)
a<-map(aSom,dtest[,-c(17:24)])
names(a)
#----Gráficos por nodo----
head(a$unit.classif)
qw<-d[aSom$unit.classif==a$unit.classif[1,]]
curvmed<-colMeans(qw[,-c(1)])
plot(curvmed, type="l", ylab="Precio", xlab="Hora",
ylim=c(30,110))
lines(as.numeric(t2[1,]), col="red")
Error<-sum(abs(curvmed-t2[1,])/t2[1,])
Error
#-----28/03/2014-----
head(a$unit.classif)
qw<-d[aSom$unit.classif==a$unit.classif[2,]]
curvmed<-colMeans(qw[,-c(1)])
plot(curvmed, type="l", ylab="Precio", xlab="Hora",
ylim=c(25,70))
lines(as.numeric(t2[2,]), col="red")
Error<-sum(abs(curvmed-t2[2,])/t2[2,])
Error
#-----29/03/2014-----
qw<-d[aSom$unit.classif==a$unit.classif[3,]]
curvmed<-colMeans(qw[,-c(1)])
plot(curvmed, type="l", ylab="Precio", xlab="Hora",
ylim=c(0,70))
lines(as.numeric(t2[3,]), col="red")
Error<-sum(abs(curvmed-t2[3,])/t2[3,])
Error
#-----30/03/2014-----
head(a$unit.classif)
qw<-d[aSom$unit.classif==a$unit.classif[4,]]
curvmed<-colMeans(qw[,-c(1)])
plot(curvmed, type="l", ylab="Precio", xlab="Hora",
ylim=c(0,70))
lines(as.numeric(t2[4,]), col="red")
Error<-sum(abs(curvmed-t2[4,])/t2[4,])
Error
#-----31/03/2014-----
qw<-d[aSom$unit.classif==a$unit.classif[5,]]
curvmed<-colMeans(qw[,-c(1)])
plot(curvmed, type="l", ylab="Precio", xlab="Hora",
ylim=c(0,70))
lines(as.numeric(t2[5,]), col="red")
Error<-sum(abs(curvmed-t2[5,])/t2[5,])
Error
#-----01/04/2014-----
head(a$unit.classif)
qw<-d[aSom$unit.classif==a$unit.classif[6,]]
curvmed<-colMeans(qw[,-c(1)])
plot(curvmed, type="l", ylab="Precio", xlab="Hora",
ylim=c(0,70))
lines(as.numeric(t2[6,]), col="red")
Error<-sum(abs(curvmed-t2[6,])/t2[6,])
Error
#-----02/04/2014-----

#----Hasta las 19h-----
set.seed(123)
a<-map(aSom,dtest[,-c(20:24)])
names(a)
#----Gráficos por nodo----
#-----27/03/2017-----
head(a$unit.classif)
qw<-d[aSom$unit.classif==a$unit.classif[1,]]
qw
curvmed<-colMeans(qw[,-c(1)])
plot(curvmed, type="l", ylab="Precio", xlab="Hora",
ylim=c(0,110))
lines(as.numeric(t2[1,]), col="red")
Error<-sum(abs(curvmed-t2[1,])/t2[1,])
Error
#-----28/03/2014-----
qw<-d[aSom$unit.classif==a$unit.classif[2,]]
qw
curvmed<-colMeans(qw[,-c(1)])
plot(curvmed, type="l", ylab="Precio", xlab="Hora",
ylim=c(0,110))
lines(as.numeric(t2[2,]), col="red")
Error<-sum(abs(curvmed-t2[2,])/t2[2,])
Error
#-----29/03/2014-----
qw<-d[aSom$unit.classif==a$unit.classif[3,]]
qw

```

```

curvmed<-colMeans(qw[,-c(1)])
plot(curvmed, type="l", ylab="Precio", xlab="Hora",
ylim=c(0,110))
lines(as.numeric(t2[3,]), col="red")
Error<-sum(abs(curvmed-t2[3,])/t2[3,])
Error
#-----30/03/2014-----
qw<-d[aSom$unit.classif==a$unit.classif[4,]]
qw
curvmed<-colMeans(qw[,-c(1)])
plot(curvmed, type="l", ylab="Precio", xlab="Hora",
ylim=c(0,110))
lines(as.numeric(t2[4,]), col="red")
Error<-sum(abs(curvmed-t2[4,])/t2[4,])
Error
#-----31/03/2014-----
qw<-d[aSom$unit.classif==a$unit.classif[5,]]
qw
curvmed<-colMeans(qw[,-c(1)])
plot(curvmed, type="l", ylab="Precio", xlab="Hora",
ylim=c(0,110))
lines(as.numeric(t2[5,]), col="red")
Error<-sum(abs(curvmed-t2[5,])/t2[5,])
Error
#-----01/04/2014-----

qw<-d[aSom$unit.classif==a$unit.classif[6,]]
qw
curvmed<-colMeans(qw[,-c(1)])
plot(curvmed, type="l", ylab="Precio", xlab="Hora",
ylim=c(0,110))
lines(as.numeric(t2[6,]), col="red")
Error<-sum(abs(curvmed-t2[6,])/t2[6,])
Error

#-----02/04/2014-----
qw<-d[aSom$unit.classif==a$unit.classif[7,]]
qw
curvmed<-colMeans(qw[,-c(1)])
plot(curvmed, type="l", ylab="Precio", xlab="Hora",
ylim=c(0,110))
lines(as.numeric(t2[7,]), col="red")
Error<-sum(abs(curvmed-t2[7,])/t2[7,])
Error
#----Modelo con datos de 2012-----
#----Modelos con año 2012---
#----Modelos con año 2012---
setwd("~/Grado de Estadística/2015/TFG/datos")
data<-read.csv("2003-2014Bienv1.csv", sep=";")
Fecha<-strptime(data$FECHA, "%d/%m/%Y")
Fecha2012<-Fecha[3288:3653]
data2012<-data[3288:3653,]
plot(Fecha2012,data2012$AVG, type="l", xlab="Fecha",
ylab="€/MWh", main="2012")
d<-data2012[1:359,-c(2:5,30:32)] # se quita la última
semana
summary(d)
d<-na.omit(d)
d1<-d[,-1]

d_matrix <- as.matrix(scale(d1))

```

```

#-----Modelo SOM-----
library(kohonen)

aGrid <- somgrid(xdim =20, ydim = 15, topo="hexagonal")
#Probar primero con rlen=1 para ver si no hay problema de
espacio.

set.seed(15324)
system.time(aSom <- som(data=d_matrix,
grid=aGrid,rlen=10000, alpha=c(0.5,0.02),keep.data=TRUE))
names(aSom)
aSom$radius
plot(aSom, type="counts")
plot(aSom, type="dist.neighbours")
source("somComponentPlanePlottingFunction.R")
source("Map_COUNTY_BMU.R")
source("UMatrixBonita.R")

dev.off()
par(mar= c(1,1,1,1))
cplanelay <- layout(matrix(1:8, nrow=2))
vars <- colnames(aSom$data)
for(p in vars) {
  plotCplane(som_obj=aSom, variable=p, legend=FALSE,
type="Quantile")
}
plot(0, 0, type = "n", axes = FALSE, xlim=c(0, 1),
ylim=c(0, 1), xlab="", ylab="")
par(mar = c(0, 0, 0, 2))
image.plot(legend.only=TRUE,
col=rev(designer.colors(n=260, col=brewer.pal(11,
"Spectral"))), zlim=c(-1.5,1.5))
dev.off()
plotUmat(aSom)
s<-seq(1,300,1)
w<-cbind(s,aSom$grid$pts)
testSOM<-data2012[360:nrow(data2012),-c(2,3,5,30:32)]
names(test)

t1<-na.omit(testSOM)
t2<-t1[,-c(1,2)]

dtest<-as.matrix(scale(t2,center= attr(d_matrix,
"scaled:center"),
scale=attr(d_matrix,"scaled:scale")))

summary(dtest)

a<-map(aSom,dtest[,-c(17:24)])
names(a)
head(a$unit.classif)
#-----25/12/2012-----
qw<-d[aSom$unit.classif== a$unit.classif[1,]]
qw
curvmed<-colMeans(qw[,-c(1)])
#----real vs predicho----
plot(curvmed, type="l", ylab="Precio",
xlab="Hora",ylim=c(0,110))
lines(as.numeric(t2[1,]), col="red")

Error<-sum(abs(curvmed-t2[1,])/t2[1,])
Error

```

```

#-----26/12/2012-----
qw<-d[aSom$unit.classif== a$unit.classif[2,]]
qw
curvmed<-colMeans(qw[,-c(1)])
#----real vs predicho----
plot(curvmed, type="l", ylab="Precio",
xlab="Hora",ylim=c(0,110))
lines(as.numeric(t2[2,]), col="red")

Error<-sum(abs(curvmed-t2[2,])/t2[2,])
Error
#-----27/12/2012-----
qw<-d[aSom$unit.classif== a$unit.classif[3,]]
qw
curvmed<-colMeans(qw[,-c(1)])
#----real vs predicho----
plot(curvmed, type="l", ylab="Precio",
xlab="Hora",ylim=c(0,110))
lines(as.numeric(t2[3,]), col="red")

Error<-sum(abs(curvmed-t2[3,])/t2[3,])
Error
#-----28/12/2012-----
qw<-d[aSom$unit.classif== a$unit.classif[4,]]
qw
curvmed<-colMeans(qw[,-c(1)])
#----real vs predicho----
plot(curvmed, type="l", ylab="Precio",
xlab="Hora",ylim=c(0,110))
lines(as.numeric(t2[4,]), col="red")

Error<-sum(abs(curvmed-t2[4,])/t2[4,])
Error
#-----29/12/2012-----
qw<-d[aSom$unit.classif== a$unit.classif[5,]]
qw
curvmed<-colMeans(qw[,-c(1)])
#----real vs predicho----
plot(curvmed, type="l", ylab="Precio",
xlab="Hora",ylim=c(0,110))
lines(as.numeric(t2[5,]), col="red")

Error<-sum(abs(curvmed-t2[5,])/t2[5,])
Error
#-----30/12/2012-----
qw<-d[aSom$unit.classif== a$unit.classif[6,]]
qw
curvmed<-colMeans(qw[,-c(1)])
#----real vs predicho----
plot(curvmed, type="l", ylab="Precio",
xlab="Hora",ylim=c(0,110))
lines(as.numeric(t2[6,]), col="red")

Error<-sum(abs(curvmed-t2[6,])/t2[6,])
Error
#-----31/12/2012-----
qw<-d[aSom$unit.classif== a$unit.classif[7,]]
qw
curvmed<-colMeans(qw[,-c(1)])
#----real vs predicho----
plot(curvmed, type="l", ylab="Precio",
xlab="Hora",ylim=c(0,110))
lines(as.numeric(t2[7,]), col="red")

Error<-sum(abs(curvmed-t2[7,])/t2[7,])
Error
#-----Desde las 20H-----
set.seed(123)
a<-map(aSom,dtest[,-c(20:24)])
names(a)
head(a$unit.classif)

qw<-d[aSom$unit.classif== a$unit.classif[1,]]
qw
curvmed<-colMeans(qw[,-c(1)])
plot(curvmed, type="l", ylab="Precio",
xlab="Hora",ylim=c(0,110))
lines(as.numeric(t2[1,]), col="red")

Error<-sum(abs(curvmed-t2[1,])/t2[1,])
Error
#-----26/12/2012-----
qw<-d[aSom$unit.classif== 183,]
qw
curvmed<-colMeans(qw[,-c(1)])
#----real vs predicho----
plot(curvmed, type="l", ylab="Precio",
xlab="Hora",ylim=c(0,110))
lines(as.numeric(t2[2,]), col="red")
Error<-sum(abs(curvmed-t2[2,])/t2[2,])
Error
#-----27/12/2012-----
qw<-d[aSom$unit.classif== a$unit.classif[3,]]
qw
curvmed<-colMeans(qw[,-c(1)])
#----real vs predicho----
plot(curvmed, type="l", ylab="Precio",
xlab="Hora",ylim=c(0,110))
lines(as.numeric(t2[3,]), col="red")

Error<-sum(abs(curvmed-t2[3,])/t2[3,])
Error
#-----28/12/2012-----
qw<-d[aSom$unit.classif== a$unit.classif[4,]]
qw
curvmed<-colMeans(qw[,-c(1)])
#----real vs predicho----
plot(curvmed, type="l", ylab="Precio",
xlab="Hora",ylim=c(0,110))
lines(as.numeric(t2[4,]), col="red")

Error<-sum(abs(curvmed-t2[4,])/t2[4,])
Error
#-----29/12/2012-----
qw<-d[aSom$unit.classif== a$unit.classif[5,]]
qw
curvmed<-colMeans(qw[,-c(1)])
#----real vs predicho----
plot(curvmed, type="l", ylab="Precio",
xlab="Hora",ylim=c(0,110))
lines(as.numeric(t2[5,]), col="red")

Error<-sum(abs(curvmed-t2[5,])/t2[5,])
Error

```

```
#-----30/12/2012-----
qw<-d[aSom$unit.classif== a$unit.classif[6,]
qw
curvmed<-colMeans(qw[,c(1)])
#----real vs predicho----
plot(curvmed, type="l", ylab="Precio",
xlab="Hora",ylim=c(0,110))
lines(as.numeric(t2[6,]), col="red")

Error<-sum(abs(curvmed-t2[6,])/t2[6,])
Error
#-----31/12/2012-----
```

```
qw<-d[aSom$unit.classif== a$unit.classif[7,]
qw
curvmed<-colMeans(qw[,c(1)])
#----real vs predicho----
plot(curvmed, type="l", ylab="Precio",
xlab="Hora",ylim=c(0,110))
lines(as.numeric(t2[7,]), col="red")

Error<-sum(abs(curvmed-t2[7,])/t2[7,])
Error
```

Ajuste de modelo ARIMA

```
#----Lectura de datos----
setwd("~/Grado de Estadística/2015/TFG/datos")
data<- read.csv("2003-2014Bienv1.csv", sep=";")
Fecha<-strptime(data$FECHA, "%d/%m/%Y")
#----Crear variable Fecha con la hora en cada día----
Fecha1<-
seq(ISOdate(2003,1,1,0),ISOdate(2014,4,2,23),by="hour")
names(data)
var1<-as.numeric()
for(i in 1:nrow(data)){
  dia1<-data[i,]
  var<-t(dia1)[-c(1:5,30:32)]
  var1<-append(var1,var)
}
plot(Fecha1, var1, type="l", xlab="Fecha", ylab="Precio del
mercado eléctrico €/MWh")
datos2<-cbind(Fecha1,as.numeric(var1))

#----ARIMA---
#---Selección de datos desde 26 de marzo del 2009----
dat0914<-datos2[54769:nrow(datos2)-168,]
test<-datos2[98473:nrow(datos2),]
Fecha<-as.POSIXct(dat0914[,1], origin="1970-01-01")
FechaTest<-as.POSIXct(test[,1], origin="1970-01-01")
plot(Fecha,dat0914[,2],type="l",xlab="Fecha",ylab="Precio
del mercado eléctrico")
plot(Fecha[-1],diff(dat0914[,2]), type="l",
xlab="Fecha",ylab="Cambio horario en el precio del
mercado eléctrico")
acf(dat0914[,2], main="")
pacf(dat0914[,2], main="")
acf(diff(dat0914[,2]),main="ACF de la diferencia de primer
grado")
pacf(diff(dat0914[,2]),main="PACF de la diferencia de
primer grado")
#---pruebas de modelos---
library(forecast)
auto.arima(dat0914[,2])
arima313<-Arima(dat0914[,2],order=c(3,1,3))
acf(arima313$residuals)
arima2413<-Arima(dat0914[,2],order=c(24,1,3))
acf(arima2413$residuals, main="ACF Residuos ARIMA
(24,1,3)")
arima011011<-Arima(dat0914[,2],
order=c(0,1,1),seasonal=list(order=c(0,1,1),period=24))
acf(arima011011$residuals,main="ACF Residuos ARIMA
(0,1,1)(0,1,1)24")
```

```
arima110110<-Arima(dat0914[,2],
order=c(1,1,0),seasonal=list(order=c(1,1,0),period=24))
acf(arima110110$residuals,main="ACF Residuos ARIMA
(1,1,0)(1,1,0)24")
arima110710<-Arima(dat0914[,2],
order=c(0,1,1),seasonal=list(order=c(7,1,1),period=24))
acf(arima110710$residuals,main="ACF Residuos ARIMA
(1,1,0)(7,1,0)24")
res<-residuals(arima110710)
Box.test(res, lag=16, fitdf=4, type="Ljung")
#----Predicción-----
#---26/3/2014---
dat260314<-datos2[98449:98472,]
Fecha2<-as.POSIXct(dat260314[,1], origin="1970-01-01")
qwe<-forecast(arima110710, h=24), xlim=c(44010,44033),
ylim=c(-25,60)
plot(Fecha2,dat260314[,2],type="l", xlab="Hora",
ylab="€/MWh", main="26/03/2014", ylim=c(-20,60))
lines(Fecha2,qwe$mean, xlim=c(44010,44033), col="red")
#---Error de predicción---
plot(Fecha2,(dat260314[,2]-qwe$mean), type="b",
main="Error de predicción")
abline(h=0, col="white")
abline(h=5,col="red")
abline(h=-5,col="red")

#----Modelo 2012-----
acf(dat2012[,2], main="ACF datos
entrenamiento")
pacf(dat2012[,2], main="PACF datos
entrenamiento")
library(forecast)
AA<-auto.arima(dat2012[,2])
arima413010<-Arima(dat2012[,2],
order=c(4,1,3),
seasonal=list(order=c(0,1,0), period=
24))
arima413010
acf(arima413010$residuals,main="ACF
Residuos ARIMA (4,1,3)(0,1,0)24")
pacf(arima413010$residuals,main="PACF
Residuos ARIMA (4,1,3)(0,1,0)24")
arima413011<-Arima(dat2012[,2],
order=c(4,1,3),seasonal=list(order=c(0,1,
1),period=24))
arima413011
acf(arima413011$residuals,main="ACF
Residuos ARIMA (4,1,3)(0,1,1)24")
```

```

pacf(arima413011$residuals,main="ACF
Residuos ARIMA (4,1,3) (0,1,1)24")
arima413111<-Arima(dat2012[,2],
order=c(4,1,3),seasonal=list(order=c(1,1
,1),period=24))
arima413111
acf(arima413111$residuals,main="ACF
Residuos ARIMA (4,1,3) (1,1,1)24")
pacf(arima413111$residuals,main="PACF
Residuos ARIMA (4,1,3) (1,1,1)24")
arima413211<-Arima(dat2012[,2],
order=c(4,1,3),seasonal=list(order=c(2,1
,1),period=24))
arima413211
res413211 <- residuals(arima413211)
tsdisplay(res413211,main="Residuos ARIMA
(4,1,3) (2,1,1)24")
arima413010168<-Arima(dat2012[,2],
order=c(4,1,3),seasonal=list(order=c(0,1
,0),period=168))
arima413010168
res413010168 <-
residuals(arima413010168)
tsdisplay(res413010168,main="Residuos
ARIMA (4,1,3) (0,1,0)168")
arima413011168<-Arima(dat2012[,2],
order=c(4,1,3),seasonal=list(order=c(0,1
,1),period=168))
arima413011168
res413011168 <-
residuals(arima413011168)

tsdisplay(res413011168,main="Residuos
ARIMA (4,1,3) (0,1,1)168")
qwe<-forecast(arima413011168), h=24)
Error<-sum(abs(qwe$mean[1:24]-test[1:24,2])/test[1:24,2])
Error
Error<-sum(abs(qwe$mean[25:48]-
test[25:48,2])/test[25:48,2])
Error
Error<-sum(abs(qwe$mean[73:96]-
test[73:96,2])/test[73:96,2])
Error
Error<-sum(abs(qwe$mean[97:120]-
test[97:120,2])/test[97:120,2])
Error
Error<-sum(abs(qwe$mean[121:144]-
test[121:144,2])/test[121:144,2])
Error
Error<-sum(abs(qwe$mean[145:168]-
test[145:168,2])/test[145:168,2])
Error
plot(test[1:24,2], type="l", xlab="Hora", ylab="Precio",
main="25/12/2012", ylim=c(0,50))
newx <- seq(1, 24, length.out=24)
polygon(c(rev(newx), newx), c(rev(qwe$upper[1:24,1]),
qwe$lower[1:24,1]), col = 'grey80', border = NA)
lines(qwe$mean[1:24], col="red")
lines(qwe$upper[1:24,1],lty="dashed",col="red")
lines(qwe$lower[1:24,1],lty="dashed",col="red")
lines(test[1:24,2])

```

Comparación de modelos

```

plot(test[1:24,2], type="l", xlab="Hora", ylab="Precio", main="25/12/2012", ylim=c(0,80))
lines(qwe$mean[1:24], col="red")
lines(curvmed, col="green")
lines(curvmed1, col="blue")

```