

Comparación de modelos predictivos para el consumo eléctrico diario, LSTM y SARIMA

Claudio López de Lérica E.

I. INTRODUCCIÓN

Al hablar sobre el funcionamiento en tiempo real del sistema eléctrico, existen algunos aspectos que se asumen dados. Uno de estos, y de los más relevantes, es la necesidad de mantener en equilibrio (igualdad) el monto de energía inyectado y consumido de la red en cada momento, para mantener la frecuencia de la misma dentro de un rango aceptable. Dadas las limitantes físicas de los mecanismos de control, resulta de extrema importancia estimar de alguna manera cuanto energía requerirá el sistema en la siguiente ventana de tiempo, por lo que se utilizan diversos modelos de predicción de demanda.

Otro aspecto donde es muy relevante la predicción de la demanda es en relación a las decisiones económicas, como es la inversión en nuevas plantas de generación o expansión del sistema de transmisión. Este problema considera granularidad semestral y anual. Así, queda claro que la predicción de la demanda es inherente al problema de servicios energéticos. Además, queda en evidencia el amplio rango de ventanas horarias en el que se ocupa, el cuál va desde los minutos hasta los años.

En el contexto anterior, resulta interesante entender, aplicar, y poder comparar distintos métodos o modelos utilizados para llevar a cabo esta tarea. En general estos se separan en modelos auto regresivos, y de aprendizaje de máquinas [1]. Sin embargo, esta categorización es discutible pues los primeros también tienen proceso de “aprendizaje”. En particular, en la bibliografía se menciona como tradicional para resolver este problema el modelo ARIMA (autoregresivo) y relacionados.

En este trabajo se busca implementar y comparar el desempeño de los modelos SARIMA y una red LSTM (Long short-term memory) para la predicción “a un paso” del consumo eléctrico con resolución diaria. SE escogieron dichas herramientas ya que la naturaleza secuencial de las series de tiempo, y la componente estacional del consumo eléctrico (semanal y mensual), advierten como conveniente el uso de modelos con enfoque en la “periodicidad” y/o “memoria”.

II. MARCO TEÓRICO

A. Modelo SARIMAX

El nombre ARIMA viene de la composición de siglas AR-I-MA, en donde AR corresponde a AutoRegressive, I corresponde a Integrated, y MA corresponde a Moving Average. Un modelo más general, SARIMAX, incluye características de estacionalidad y variables exógenas, con S correspondiente a Seasonal y X correspondiente a exogenous factors.

Se denomina modelo auto regresivo pues, al no considerar variables independientes, se busca general una regresión con la misma variable dependiente (y) de p tiempos anteriores (orden p), como se presenta a continuación

$$y_t = \beta_0 + \beta_1 \cdot y_{t-1} + \dots + \beta_p \cdot y_{t-p} + \epsilon_t \quad (1)$$

$$y_t = f(y_{t-1}, \dots, y_{t-p}, \epsilon_t) \quad (2)$$

Un modelo MA, en el contexto ARIMA, refiere al uso únicamente de los errores para q tiempos anteriores (orden q), lo que se expresa a continuación

$$y_t = \beta_0 + \epsilon_t + \phi_1 \cdot \epsilon_{t-1} + \dots + \phi_q \cdot \epsilon_{t-q} \quad (3)$$

$$y_t = f(\epsilon_t, \epsilon_{t-1}, \dots, \epsilon_{t-q}) \quad (4)$$

Para aplicar un modelo AR, MA o ARMA, se requiere que la función (data) sea estacionaria. Para corroborar esto se utiliza “Augmented Dick-Fuller test” lo que, en pocas palabras, requiere que la relación entre dos datos Y_t e Y_{t-1} no dependa del tiempo t de observación. La exigencia típica del teste son $p - values < 5\%$.

Si la función llegase a ser no estacionaria, entonces se puede generar una serie estacionaria diferenciando (orden d) la original ($y_t^* = y_t - y_{t-1} \forall t$). Para agregar satisfactoriamente esta acción al modelo, se debe tener la noción de integrar (I) el resultado.

Finalmente, debido al conocimiento previo del problema, se agrega una variable de estacionalidad (Seasonal) por la naturaleza del consumo. Además, se ignora la posibilidad de agregar información exógena (X) para trabajar únicamente con el dataset. Con lo anterior, se obtiene un modelo SARIMA.

Queda por mencionar que, para detectar qué hiper parámetros aplican mejor al problema, se recurre al criterio de menor AIC (Akaike Information Criterion). Se extrae un porcentaje de modelos que logren los menores valores AIC, pues en este conjunto es más probable encontrar el modelo que se ajuste mejor dentro de las opciones propuestas.

B. Redes neuronales LSTM

Las redes LSTM son un caso particular de RNN. Estas últimas se caracterizan por tener un ciclo cerrado sobre sí misma, lo que convierte a la salida de un tiempo t en una entrada para un tiempo $t + 1$. Esto permite que la información asociada pueda persistir en el tiempo. A continuación se presenta de manera ilustrativa este funcionamiento.

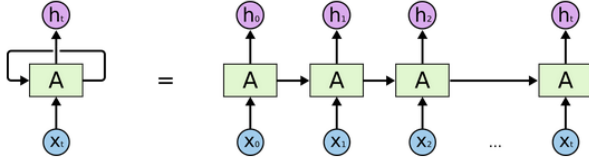


Fig. 1. RNN equivalente a una secuencia (en diferentes tiempos) conectadas por la salida.

Ahora bien, a pesar de que tengan la virtud de retener, a medida que el espaciado entre periodos con información relevante aumenta, las RNN's convencionales pierden la capacidad de conectar esta misma (info.) y aprender como consecuencia. Es en este punto donde la estructura LSTM cobra importancia, la cuál se muestra en la fig. 7.

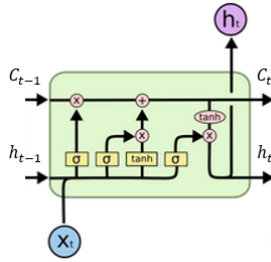


Fig. 2. Representación interna LSTM para una entrada en tiempo t .

Se puede distinguir que, además de la entrada propia de la variable x_t se tienen como inputs la salida del tiempo anterior h_{t-1} y el estado interno del tiempo anterior C_{t-1} . Este último, como queda expreso, no se presenta como output de la configuración, sino que cumple el rol de memoria interna.

En pocas palabras, C_{t-1} cruza a través de una función que decide si es necesario recordar esa información u olvidarla (signo \times). Posterior a esto, la fracción recordada de el estado anterior se suma (signo $+$) a la nueva propuesta de estado interno, calculada a través de h_{t-1} y x_t . Finalmente C_t se filtra para obtener h_t , y paralelo a esto se transmite de manera bruta al siguiente periodo.

Como se mencionó, la gran utilidad de esta herramienta es la posibilidad de retener información por largos periodos de tiempo lo que, dada la periodicidad del problema de consumo eléctrico, la hace naturalmente atractiva para explorar.

C. Medida de error: MAPE (Mean Absolute Percentage Error)

Se utilizó esta medida exclusivamente porque se construye para ser independiente de la escala. Así, es posible comparar el desempeño del modelo auto regresivo SARIMA con un modelo LSTM, ya que los datos de entrada del segundo se normalizan previo al proceso mientras que para el primero no es necesario.

Se define como

$$MAPE = \frac{1}{n} \sum_{t=1}^n \left| \frac{e_t}{real_t} \right| \times 100 \quad (5)$$

Donde e_t cooresponde al error para el tiempo t , esto es $real_t - forecast_t$.

III. METODOLOGÍA

Se trabajó con una base de datos oficial del RTO PJM desde Kaggle [2], la cuál contiene el consumo eléctrico de distintos sectores de PJM con resolución horaria. Dado el desarrollo e implementación de sistemas de autoconsumo y almacenamiento, resulta interesante analizar el consumo energético con resolución diaria (horaria agregada en un día). Con esto, se elije el sector domiciliario (*DOM*) por su marcada naturaleza estacional, el cuál cuenta con una base de 4843 datos, desde 01-05-2005 hasta el 03-08-2018.

Antes de aplicar cualquier método se realizó un pre procesamiento. Se eliminó únicamente la primera y la última fecha, puesto que no se tenían los 24 datos horarios correspondientes originales, y por tanto la suma diaria distaba del promedio del total de datos. Con esto, quedan 4841 datos a disposición. Además, se verificó la inexistencia de outliers y fechas repetidas.

Para series de tiempo con comportamiento estacional, es relevante separar la data *train/val/test* en periodos. Así, *train* = [2005 : 2016], *val* = [2017] y *test* = [2018], como se muestra en la figura a continuación.

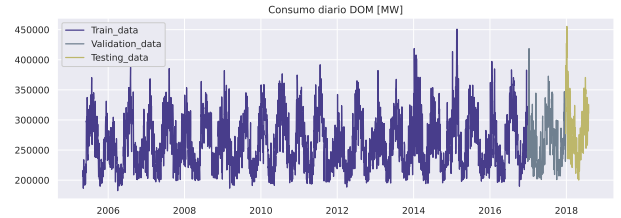


Fig. 3. Data sin procesamiento.

Se propone como experimento el ajuste de hiperparámetros de ambas herramientas para una predicción "a un paso", lo que quiere decir que luego de calcular el pronóstico \hat{y}_t , para calcular \hat{y}_{t+1} se considera el valor real y_t y no el estimado. Esto hace sentido en la realidad para dimensionar o configurar instalaciones de autoconsumo y almacenamiento. Lo anterior para una posterior comparación de desempeños.

A. Metodología SARIMA

- Verificar si la serie de tiempo es estacionaria (\neq estacional) con ADFuller y análisis cualitativo, tanto para el set de train como validación. De no ser así, es necesario imponer $d = 1$ como se explicó antes (I).
- Se impone un periodo de estacionalidad $S = 7$, puesto que se conoce la naturaleza periódica semanal del consumo eléctrico.
- Se propone un vector de búsqueda para hiperparámetros $p = q = P_s = Q_s = [0, 1, 2]$ y $D_s = [0, 1]$, considerando capacidad computacional, para posteriormente probar las combinaciones con menores AIC registradas. Se propone

también un 5% de total de configuraciones probadas (162), lo que da un total de 8.

- Elección de modelo con menor MAPE en el set de validación, y análisis de residuos. Particularmente densidad de probabilidad, Q-Q plot y autocorrelación.
- Aplicación del modelo elegido a testing y cálculo de desempeño.

B. Metodología LSTM

- Normalización de data para evitar problemas en el entrenamiento por las dimensiones.
- Se propuso una estructura con 3 capas de LSTM (siendo estas mismas inputs y outputs). Además, debido a que la cantidad de datos de entrenamiento se considera pequeña (cerca de 4000), se quiso evitar una gran cantidad de pesos asociados, por lo que se limitó a un total de 10 neuronas por capa. Finalmente, se añadió un “dropout” de 15% a capa capa como valor típico, para evitar overfitting.
- En cuanto a los parámetros de entrenamiento, se propuso un “lag” de 7 tiempos para el cálculo del paso siguiente. Esto debido a la naturaleza del problema.
- Finalmente, se entrenó hasta 200 épocas analizando los valores de MAPE de validación y test para posibles mejoras del modelo (posible overfitting, o posibilidad de agregar estructuras para complejizar el modelo), y estimar un valor razonable de épocas para el modelo final.
- Aplicación del modelo elegido a set de test, y cálculo de desempeño.

IV. RESULTADOS Y ANÁLISIS

A. Resultados y Análisis SARIMA

Al realizar ADFuller test se comprobó que la data no es estacionaria, pues se obtuvo p - values mayores a 5% en el set de validación. En particular 15.59%. Esto era esperable pues se trabaja con data estacional (énfasis en diferencia entre estacionaria y estacional). Así, se termina de justificar $d = 1$.

En segundo lugar se analiza el gráfico de autocorrelación, donde se observa claramente el efecto semanal (fig. 4), lo que argumenta el uso de $S = 7$.

Como se mencionó, de los 162 modelos obtenidos a partir de la combinación de parámetros se escogió aquellos pertenecientes al 5% con menor AIC . Es relevante mencionar que este valor (AIC) asigna un puntaje tanto al desempeño del modelo, como a los requerimientos para procesar este. Así, es importante la selección de un grupo para observar el desempeño real, y no escoger exclusivamente a partir del menor AIC pues posiblemente no correspondan.

En la tabla I, se muestra el subconjunto 5% junto con sus valores de AIC y $MAPE$ para el test de validación.

De la tabla se observa que el modelo con menor valor de $MAPE$ no corresponde al modelo con menor valor de AIC , lo que se justifica en lo explicado recientemente.

Finalmente, se aplica el modelo $(2, 1, 0) \times (2, 0, 2, 7)$ entrenado sobre el conjunto test, con lo que se obtiene un $MAPE = 6.9359\%$

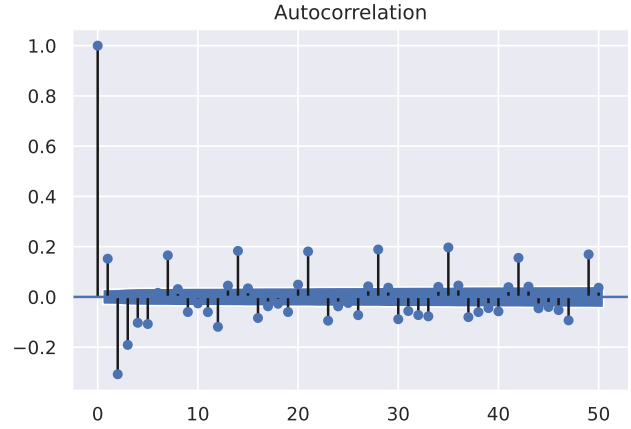


Fig. 4. Gráfico de autocorrelación.

TABLE I
VALORES DE AIC Y MAPE PARA 5% MENOR.

$(q, d, p) \times (Q_s, D_s, P_s, S)$	AIC	MAPE (val)
$(1, 1, 2) \times (2, 0, 2, 7)$	95159.45	6.5386
$(2, 1, 2) \times (1, 0, 2, 7)$	95204.95	6.5962
$(2, 1, 1) \times (2, 0, 1, 7)$	95292.79	6.5826
$(2, 1, 1) \times (2, 0, 2, 7)$	95250.32	6.5221
$(1, 1, 2) \times (2, 0, 1, 7)$	95266.34	6.5225
$(1, 1, 2) \times (1, 0, 2, 7)$	95220.40	6.5954
$(2, 1, 1) \times (1, 0, 2, 7)$	95302.69	7.5336
$(2, 1, 0) \times (2, 0, 2, 7)$	95353.22	6.3742

B. Resultados y Análisis LSTM

Debido a la cantidad de requerimientos teóricos, tanto de la herramienta SARIMA como de la herramienta LSTM, se optó por analizar exclusivamente la cantidad de épocas para entrenar, habiendo argumentado la utilización de algunos otros parámetros en la sección anterior.

En la figura 5 se observa el error $MAPE$ para train y validación, después de cada época. Es importante notar que, aunque se perciba un aumento en la diferencia a medida que se crece la cantidad de épocas de entrenamiento, el eje abscisa ($MAPE$) da cuenta que esta distinción es pequeña, al menos para el rango de épocas evaluado.

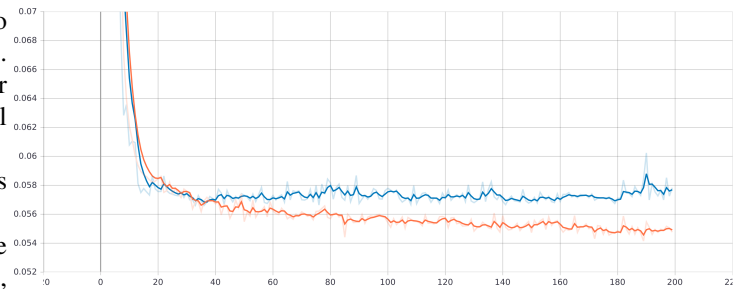


Fig. 5. Valores $MAPE$ por época entrenada, suavizado. $MAPE$ Validación: Azul, $MAPE$ Naranja: Train.

Si bien no se distingue comportamiento de overfitting, en

el sentido de empeoramiento del desempeño de validación, al menos existe un estancamiento a partir de las 40 épocas. Así, se escoge este último valor como del hiperparámetro, con $MAPE = 5.7\%$. Esto hace sentido, considerando la poca cantidad de data para el conjunto de entrenamiento.

Se verifica que la distribución de los residuos es similar a una normal, por lo que se aprueba este modelo.

Aplicando este modelo sobre el conjunto test, se obtiene un valor de $MAPE = 6.74\%$

C. Resultados y Análisis Generales

A continuación, en la figura 6 y figura 7, se presentan de manera ilustrativa los resultados de la aplicación del modelo SARIMA y LSTM respectivamente.

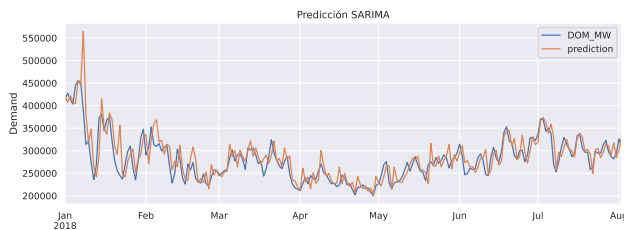


Fig. 6. Presentación modelo SARIMA sobre test.

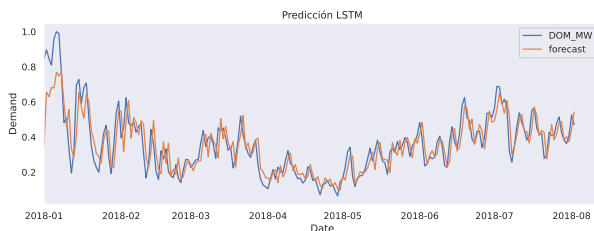


Fig. 7. Presentación modelo LSTM sobre test.

Se puede observar que ambos modelos son capaces de realizar un seguimiento a la data original, siempre teniendo presente que se está trabajando con predicción a un paso.

A pesar de lo esto, queda en evidencia una gran diferencia para los primeros valores de predicción y data real para el modelo LSTM. Esto se puede explicar debido a una dificultad en la manipulación de los datos en el código. Con los limitados conocimientos y capacidades que se poseen no fue posible separar los procesos de entrenamiento y utilización del modelo. Esto funciona bien cuando la data es secuencia (*train-val*), sin embargo al utilizar *train-test* existía un gap entre los conjuntos. Es probable que la red haya reaccionado a este salto abrupto entre los periodos y causado un desacople mayor.

Con todo lo anterior, la red LSTM supera en desempeño al modelo SARIMA, a pesar de las pocas pruebas que se le hicieron a los hiperparámetros con la primera herramienta.

V. CONCLUSIONES

En el presente proyecto se presentó una primera aproximación a la aplicación de un modelo estadístico SARIMA y una red LSTM para la predicción a un paso de consumo eléctrico, y la comparación de sus resultados.

Si bien la red LSTM superó el desempeño del modelo estadístico SARIMA, la diferencia resulta posible de disminuir o aumentar afinando de mejor manera ambas herramientas. En el caso de SARIMA se puede expandir el rango de búsqueda de hiperparámetros, mientras que en el caso de LSTM se pueden modificar otros hiperparámetros además de la cantidad de épocas, incluso la arquitectura.

Se encontraron diversas dificultades, de las cuales se puede destacar la profundidad teórica, la aplicación de las librerías, y la naturaleza de las series de tiempo. Esta última resulta interesante pues agrega consideraciones propias del problema a resolver, al margen de la herramienta que se ocupe. Por último, cabe mencionar la transparencia del uso de SARIMA, en cuanto a la construcción del modelo matemático para llegar al resultado. A diferencia de este, una red LSTM podría ser más simple, pero posee un comportamiento similar a una caja negra.

Como trabajo futuro se propone profundizar más en los límites de cada uno de estos modelos por separados.

An excellent style manual for science writers is [1].

REFERENCES

- [1] Vitor Cerqueira and Luis Torgo and Igor Mozetic, "Evaluating time series forecasting models: An empirical study on performance estimation methods"
- [2] Hourly Energy Consumption; <https://www.kaggle.com/robikscube/hourly-energy-consumption>
- [3] A. Shadkam, "Using SARIMAX to forecast electricity demand and consumption in university buildings," T, University of British Columbia, 2020.
- [4] Michael Phi; "Illustrated Guide to LSTM's and GRU's: A step by step explanation"; <https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21>
- [5] "Understanding LSTM Networks"; <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>
- [6] Eligijus Bujokas, "Energy consumption time series forecasting with python and LSTM deep learning model"; <https://towardsdatascience.com/energy-consumption-time-series-forecasting-with-python-and-lstm-deep-learning-model-7952e2f9a796>
- [7] "How to use SARIMAX"; <https://www.kaggle.com/poiupoiu/how-to-use-sarimax>