

# Técnicas estadísticas para el pronóstico de la demanda mediante Matlab

Jiram Izanami Campos Romero, Luis Joaquín Hernández Núñez, N. González-Cabrera  
División de Ingeniería Eléctrica  
Universidad Nacional Autónoma de México  
Ciudad de México, México  
jirizanamicr@gmail.com, luis.hernandez5325@gmail.com

**Resumen— Existen muchas técnicas estadísticas para realizar la gestión y pronóstico de datos en la actualidad, además de presentar respuestas muy cercanas a los datos de estudio de nuestro interés. En este trabajo, realizamos el pronóstico de la demanda para 24 horas en adelante con respecto al tiempo, por medio del software MATLAB®□, específicamente a través de “Statistics and Machine Learning Toolbox™” y “Econometrics Toolbox™”, haciendo uso de las funciones de modelado y técnicas estadísticas que ofrecen las mismas. Comparando las respuestas, con base en el error y comportamiento gráfico que cada uno obtiene, como forma de validación con respecto al método obtenido.**

*Palabras clave— Matlab, técnicas estadísticas, pronóstico, regresión, comparación.*

## I. INTRODUCCIÓN

La revolución digital está produciendo profundos cambios en los hábitos de consumo de los clientes; entre otros motivos, por un mayor acceso a datos, y un creciente desarrollo de nuevas tecnologías. Todo lo mencionado anteriormente se logró a partir de la Ciencia de los datos (Data Science), que se basa en el uso combinado de técnicas de aprendizaje automático, inteligencia artificial, matemáticas, estadística, bases de datos y optimización. [1]

Las técnicas de aprendizaje automático (o "Machine Learning") están experimentando un auge sin precedentes en diversos ámbitos, tanto en el mundo académico como en el empresarial, y constituye un importante camino hacia la transformación. Las técnicas de aprendizaje automático pueden definirse como un conjunto de métodos capaces de detectar automáticamente patrones en los datos. Bajo este concepto el aprendizaje automático lleva aproximadamente 50 años, periodo en el que se descubrieron y redefinieron diversos métodos estadísticos y se aplicaron al aprendizaje automático a través de algoritmos simples y complejos, aunque circunscritos casi exclusivamente al ámbito académico. [2]

El concepto de aprendizaje automático incluye entonces el uso de patrones detectados para realizar predicciones, o para tomar otro tipo de decisiones en entornos de incertidumbre. Frente a las técnicas de estadística clásicas, la introducción de técnicas Máquina de Aprendizaje (Machine Learning - ML) permiten mejorar el proceso de estimación de modelos, no solo con relación al aumento del poder predictivo a través de nuevas metodologías y técnicas de selección de variables, sino

también en la mejora de la eficiencia de los procesos a través de la automatización. [3]

Entonces en resumen podemos decir que ML aplica técnicas y algoritmos capaces de aprender a partir de distintas y nuevas fuentes de información, construyendo algoritmos que mejoren de forma autónoma con la experiencia. Actualmente existen múltiples técnicas de aprendizaje automático, dependiendo del tipo de información (información estructurada o desestructurada) y del paradigma de aprendizaje que se utilicen. La selección de la técnica que hay que aplicar dependerá, entre otros, del objetivo del modelo que se quiere construir, así como del tipo de información disponible.

El desarrollo de estos componentes supone una evolución con el enfoque que tradicionalmente ha tenido la modelización. Ello implica, entre otros elementos, el uso de un mayor número y tipos de fuente de información, la capacidad de detección de patrones ocultos en los datos a partir de métodos inductivos, el mantenimiento del poder predictivo durante más tiempo, así como la necesidad de una mayor capacidad de almacenamiento y procesamiento de datos. [4]

El uso de nuevas técnicas como ML para la caracterización del comportamiento de la demanda eléctrica ha generado mayor confianza en la gestión de datos adquiridos diariamente. El sistema eléctrico se diseñó inicialmente de forma jerárquica para dar respuesta a las necesidades del consumo eléctrico, estando dimensionadas la generación y las redes eléctricas para poder suministrar el máximo consumo demandado. En este contexto, el aumento de los costos de energía, el incremento de las energías renovables, junto a los nuevos esquemas de generación distribuida, así como la necesidad de mejorar el aprovechamiento de las infraestructuras de la red de transmisión y distribución, marcan la necesidad de nuevas propuestas para la operación y control de los sistemas eléctricos. [5]

Es imprescindible conocer en buena medida el funcionamiento del software con el que se está trabajando. Matlab®□ combina un entorno de escritorio perfeccionado para el análisis iterativo y los procesos de diseño con un lenguaje de programación que expresa las matemáticas de matrices y vectores directamente. Son diversas los proyectos que pueden apoyarse en este tipo de software. Algunas de las herramientas que nos ofrece son:

- Análisis de datos
- Comunicaciones inalámbricas

- Deep Learning
- Visión artificial
- Procesamiento de señales
- Finanzas cuantitativas y Gestión de riesgos
- Robótica
- Sistemas de control

Este artículo tiene como objetivo realizar la predicción de la demanda, por medio de “Análítica de datos” la cual se apoya en el contexto de “*data science*” por medio de herramientas para acceder y preprocesar datos, crear modelos predictivos y de ML, siendo este último el enfoque de modelado de predicción de la demanda. De igual forma, se pretende comparar los nuevos métodos propuestos contra el método ARIMA, así como, analizar la diferencia entre los métodos, su predicción y el error que presenta cada uno de ellos, todos basadas en los toolbox e Matlab.

## II. TÉCNICAS ESTADÍSTICAS

Matlab® ofrece la herramienta “Statistics and Machine Learning Toolbox”, el cual nos proporciona funciones y apps, para describir, analizar y modelar datos. Puede utilizar estadística descriptiva y gráficos para el análisis exploratorio de datos, ajustar distribuciones de probabilidad a datos, generar números aleatorios para simulaciones, y realizar pruebas de hipótesis. Los algoritmos de regresión y clasificación permiten extraer inferencias de los datos y crear modelos predictivos. [6].

### A. Modelo ARIMA

Box y Jenkins desarrollo modelos estadísticos para series temporales, que tienen en cuenta la dependencia existente entre los datos, esto es, caso observación en un momento dado es modelado en función de los valores anteriores. Los análisis se basan en un modelo explícito. Los modelos se conocen con el nombre genérico de ARIMA (*Autoregressive Integrated Moving Average*), que deriva de sus tres componentes AR (Autorregresivo), (Integrado), MA (Medias Móviles).

El modelo ARIMA permite describir un valor como una función lineal de datos anteriores y errores debidos al azar, además, puede incluir un componente cíclico o estacional. Es decir, debe contener todos los elementos necesarios para describir el fenómeno. Box y Jenkins recomiendan como mínimo 50 observaciones en la serie temporal. [7]

### B. Regresión lineal generalizada

En un modelo lineal generalizado (MLG), se supone que cada resultado  $Y$  de las variables dependientes se genera a partir de una distribución particular en la familia exponencial, un amplio rango de distribuciones de probabilidad que incluye las distribuciones normales, binomial, Poisson y gamma, entre otras. La media,  $\mu$ , de la distribución depende de las variables independientes,  $X$ , por medio de

$$E(Y) = \mu = g^{-1}(X\beta) \quad (1)$$

Donde,  $E(Y)$  es el valor esperado de  $Y$ ;  $X\beta$  es el predictor lineal;  $g$  es la función de enlace. Con esta notación, la varianza es típicamente una función  $V$  de la media:

$$Var(Y) = V(\mu) = V(g^{-1}(X\beta)) \quad (2)$$

Es conveniente si  $V$  sigue la distribución de la familia exponencial, pero puede ser simplemente que la varianza sea una función del valor predicho. El MLG consta de tres elementos:

1) Una función de distribución  $f$ : una generalización de la familia exponencial y el modelo de dispersión exponencial de distribuciones e incluye esas distribuciones de probabilidad, parametrizadas por  $\theta$  y  $\tau$ , cuyas funciones de densidad  $f$  (o función de masa de probabilidad, para el caso de una distribución discreta) se pueden expresar en la forma:

$$f_y = (y | \theta, \tau) = h(y, \tau) * \exp\left(\frac{b(\theta)^T T^T(y) - A(\theta)}{d(\tau)}\right) \quad (3)$$

El parámetro de dispersión  $\tau$ , en general se conoce y generalmente está relacionado con la varianza de la distribución. Las funciones  $h(y, \tau)$ ,  $b(\theta)$ ,  $T(y)$ ,  $A(\theta)$  y  $d(\tau)$  son conocidas. Muchas distribuciones comunes están en esta familia, incluyendo binomial, multinomial y binomial normal, exponencial, gamma, Poisson, Bernoulli (para un número fijo de ensayos).

2) Un predictor lineal  $\eta = X\beta$ : la cantidad que incorpora la información sobre las variables independientes en el modelo. El símbolo  $\eta$  (“eta” griego) denota un predictor lineal. Está relacionado con el valor esperado de los datos a través de la función de enlace. Donde  $\eta$  se expresa como combinaciones lineales (por lo tanto, “lineales”) de parámetros desconocidos  $\beta$ . Los coeficientes de la combinación lineal se representan como la matriz de las variables independientes  $X$ .

3) Función de enlace: proporciona la relación entre el predictor lineal y la media de la función de distribución. Existen muchas funciones de enlace de uso común, y su elección se basa en varias consideraciones. Siempre hay una función de enlace canónica bien definida que se deriva del exponencial de la función de densidad de la respuesta. Sin embargo, en algunos casos tiene sentido tratar de hacer coincidir el dominio de la función de enlace con el rango de la media de la función de distribución, o usar una función de enlace no canónica con fines algorítmico. [8]

### C. Clasificación de máquinas vectoriales de soporte

Una máquina de vectores de soporte (SVM) es un algoritmo de aprendizaje supervisado que se puede emplear para clasificación binaria o regresión. Las máquinas de vectores de soporte son muy populares en aplicaciones como el procesamiento del lenguaje natural, el habla, el reconocimiento de imágenes y la visión artificial.

Una máquina de vectores de soporte construye un hiperplano óptimo en forma de superficie de decisión, de modo que el margen de separación entre las dos clases en los datos se amplía al máximo. Los vectores de soporte hacen referencia a un pequeño subconjunto de las observaciones de entrenamiento que se utilizan como soporte para la ubicación óptima de la superficie de decisión. [9]

El entrenamiento de una máquina de vectores de soporte consta de dos fases:

1) Transformar los predictores (datos de entrada) en un espacio de características altamente dimensional. En esta fase es suficiente con especificar el kernel; los datos nunca se

transforman explícitamente al espacio de características. Este proceso se conoce comúnmente como el truco kernel.

2) Resolver un problema de optimización cuadrática que se ajuste a un hiperplano óptimo para clasificar las características transformadas en dos clases. El número de características transformadas está determinado por el número de vectores de soporte. Para construir la superficie de decisión solo se requieren los vectores de soporte seleccionados de los datos de entrenamiento. Una vez entrenados, el resto de los datos de entrenamiento son irrelevantes. [9]

#### D. Árboles de decisión

Se utiliza un árbol de decisión como un modelo predictivo que mapea observaciones sobre un artículo a conclusiones sobre el valor objetivo del artículo. Es uno de los enfoques de modelado predictivo utilizadas en estadísticas, minería de datos y aprendizaje automático. Los modelos de árbol, donde la variable de destino puede tomar un conjunto finito de valores se denominan árboles de clasificación. En estas estructuras de árbol, las hojas representan etiquetas de clase y las ramas representan las conjunciones de características que conducen a esas etiquetas de clase. Los árboles de decisión, donde la variable de destino puede tomar valores continuos (por lo general números reales) se llaman árboles de regresión. [10]

Los algoritmos para la construcción de árboles de decisión suelen trabajar de manera top-down, escogiendo en cada paso la variable que mejor divide el conjunto de elementos. Diferentes algoritmos utilizan diferentes métricas para medir el "mejor". Estos miden generalmente la homogeneidad de la variable de destino dentro de los subconjuntos. Algunos ejemplos se dan a continuación. Estas métricas se aplican a cada subconjunto candidato, y los valores resultantes se combinan (por ejemplo, un promedio) para proporcionar una medida de la calidad de la división. [11]

Utilizado por el algoritmo de ACR (Árboles de Clasificación y Regresión), la impureza de Gini es una medida de cuán a menudo un elemento elegido aleatoriamente del conjunto sería etiquetado incorrectamente si fue etiquetado de manera aleatoria de acuerdo con la distribución de las etiquetas en el subconjunto. La impureza de Gini se puede calcular sumando la probabilidad de cada elemento siendo elegido multiplicado por la probabilidad de un error en la categorización de ese elemento. Alcanza su mínimo (cero) cuando todos los casos del nodo corresponden a una sola categoría de destino. Para calcular la impureza de Gini de un conjunto de elementos

$$I_G(f) = 1 - \sum_{i=1}^m f_i^2 \quad (4)$$

#### E. Método de regresión del proceso gaussiano

Los procesos gaussianos (GPR) se definen como una distribución de probabilidad sobre funciones aleatorias. Son sobre colecciones infinitas de variables (funciones), tal que cualquier subconjunto de variables aleatorias finitas tiene una distribución gaussiana multivariable. Estos modelos probabilísticos son basados en kernel, esto quiere decir que en el aprendizaje se espera que los puntos con valores predictivos similares, naturalmente, tengan valores de respuesta cercana

(objetivo). En los procesos gaussianos la función de covarianza (kernel) expresa esta similitud. Considere un conjunto de entrenamiento:  $[(\mathbf{x}_i, y_i); i = 1, 2, \dots, n]$ , donde:  $\mathbf{x}_i \in \mathbb{R}^d, y_i \in \mathbb{R}$  procedente de una distribución desconocida. Un modelo GPR aborda la cuestión de predecir el valor de una variable de respuesta  $y$  dado el nuevo vector de entrada  $\mathbf{x}$  y los datos de entrenamiento. Un modelo de regresión lineal tiene la siguiente forma:

$$y = \mathbf{x}^T \cdot \boldsymbol{\beta} + \varepsilon \quad (5)$$

Donde,  $\boldsymbol{\beta}$  son los datos de entrada,  $\mathbf{x}^T$  vector de entrada transpuesta.

Un modelo GPR explica la respuesta introduciendo variables latentes,  $[f(\mathbf{x}_i), i = 2, 3, \dots, n]$  de un proceso gaussiano (GP) y funciones de bases explícitas  $h$ . La función de covarianza de las variables latentes captura la suavidad de las funciones de respuesta y base proyectan entradas  $\mathbf{x}$  en un espacio de características dimensional  $p$ . Un GP es un conjunto de variables aleatorias, de modo que cualquier número finito de ellas tiene una distribución gaussiana conjunta. Si  $[f(\mathbf{x}), \mathbf{x} \in \mathbb{R}^d]$  es un GP, se le dan observaciones  $[\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n]$  la distribución conjunta de las variables aleatorias  $[f(\mathbf{x}_1), f(\mathbf{x}_2), \dots, f(\mathbf{x}_n)]$  es gaussiano, además un GP se define por su función media  $m(\mathbf{x})$  y su función de covarianza  $[k(\mathbf{x}, \mathbf{x}')]$ . Es decir, si  $[f(\mathbf{x}), \mathbf{x} \in \mathbb{R}^d]$  es un proceso gaussiano [12], entonces:

$$E(f(\mathbf{x})) = m(\mathbf{x}) \quad (6)$$

$$\text{Cov}[f(\mathbf{x}), f(\mathbf{x}')] = E[\{(f(\mathbf{x})-m(\mathbf{x}))\} \{(f(\mathbf{x}')-m(\mathbf{x}'))\}] = k(\mathbf{x}, \mathbf{x}') \quad (7)$$

#### F. Construcción y evaluación de modelos.

La primera y más crítica tarea que hay que realizar es encontrar cuáles son las entradas y salidas esperadas. Habría que responder a las siguientes cuestiones:

- Objetivo principal, ¿Que vamos a tratar de predecir?
- Características objetivo
- Mejora esperada
- Estado actual de la variable objetivo
- Medición de la variable objetivo

Entonces para realizar nuestra construcción de nuestro modelo es necesario tomar en cuenta definir adecuadamente nuestro problema, a continuación, realizar la recopilación de nuestros datos con los que vamos a tratar. Es importante establecer el protocolo de evaluación y cuáles son los que están disponibles para establecer en nuestro estudio. Una vez que recopilados nuestros datos deben tratarse correctamente y dividirse. No todos los problemas pueden resolverse, sólo podemos hacer cierta hipótesis hasta que tengamos un modelo funcional. Al crear un modelo de clasificación predictiva de alta calidad, es importante seleccionar las entidades (o predictores) correctas y ajustar los hiper parámetros (parámetros de modelo que no se estiman). [13]

#### G. Clasificador vecino más cercano.

El método de los  $k$  vecinos más cercanos es un método de clasificación supervisada (Aprendizaje, estimación basada en un conjunto de entrenamiento y prototipos) que sirve para estimar la función de densidad  $F(x/C_j)$  de las predictoras  $\mathbf{x}$  por cada clase  $C_j$ .

Este es un método de clasificación no paramétrico, que estima el valor de la función de densidad de probabilidad o directamente la probabilidad a posteriori de que un elemento  $x$  pertenece a la clase  $C_j$  a partir de la información proporcionada por el conjunto de prototipos. En el proceso de aprendizaje no se hace ninguna suposición acerca de la distribución de las variables predictoras. En el reconocimiento de patrones, el algoritmo k-nn es usado como método de clasificación de objetos (elementos) basado en un entrenamiento mediante ejemplos cercanos en el espacio de los elementos. k-nn es un tipo de aprendizaje vago (lazy learning), donde la función se aproxima solo localmente y todo el cómputo es diferido a la clasificación. Los ejemplos de entrenamiento son vectores en un espacio característico multidimensional, cada ejemplo está descrito en términos de  $p$  atributos considerando  $q$  clases para la clasificación. Los valores de los atributos del  $i$ -ésimo ejemplo (donde  $1 < i < n$ ) se representan por el vector  $p$ -dimensional  $x_i = (x_{i1}, x_{i2}, \dots, x_{ip_i}) \in X$ .

El espacio es particionado en regiones por localizaciones y etiquetas de los ejemplos de entrenamiento. Un punto en el espacio es asignado a la clase  $C$  si esta es la clase más frecuente entre los  $k$  ejemplos de entrenamiento más cercano. Generalmente se usa la distancia euclidiana.

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^p (x_{ri} - x_{rj})^2} \quad (8)$$

Donde,  $x_i$  es el vector  $p$  dimensional,  $x_j$  es un vector  $q$  de clases,  $p$  es el número de atributos y  $(x_{ri} - x_{rj})^2$  es la distancia euclidiana.

La fase de entrenamiento del algoritmo consiste en almacenar los vectores característicos y las etiquetas de las clases de los ejemplos de entrenamiento. En la fase de clasificación, la evaluación del ejemplo (del que no se conoce su clase) es representada por un vector en el espacio característico. Se calcula la distancia entre los vectores almacenados y el nuevo vector, y se seleccionan los  $k$  ejemplos más cercanos. El nuevo ejemplo es clasificado con la clase que más se repite en los vectores seleccionados. Este método supone que los vecinos más cercanos nos dan la mejor clasificación y esto se hace utilizando todos los atributos; el problema de dicha suposición es que es posible que se tengan muchos atributos irrelevantes que dominen sobre la clasificación: dos atributos relevantes perderían peso entre otros veinte irrelevantes. [14]

#### H. Clasificador naive Bayes.

En teoría de la probabilidad y minería de datos, un clasificador Bayesiano ingenuo es un clasificador probabilístico fundamentado en el teorema de Bayes y algunas hipótesis simplificadoras adicionales. Es a causa de estas simplificaciones, que se suelen resumir en la hipótesis de independencia entre las variables predictoras, que recibe el apelativo de ingenuo.

En términos simples, un clasificador de Bayes ingenuo asume que la presencia o ausencia de una característica particular no está relacionada con la presencia o ausencia de cualquier otra característica, dada la clase variable. Por

ejemplo, una fruta puede ser considerada como una manzana si es roja, redonda y de alrededor de 7 cm de diámetro. Un clasificador de Bayes ingenuo considera que cada una de estas características contribuye de manera independiente a la probabilidad de que esta fruta sea una manzana, independientemente de la presencia o ausencia de las otras características.

Para otros modelos de probabilidad, los clasificadores de Bayes ingenuo se pueden entrenar de manera muy eficiente en un entorno de aprendizaje supervisado. En muchas aplicaciones prácticas, la estimación de parámetros para los modelos Bayes ingenuo utiliza el método de máxima verosimilitud, en otras palabras, se puede trabajar con el modelo ingenuo de Bayes sin aceptar probabilidad bayesiana o cualquiera de los métodos bayesianos. Una ventaja del clasificador de Bayes ingenuo es que solo se requiere una pequeña cantidad de datos de entrenamiento para estimar los parámetros (las medias y las varianzas de las variables) necesarias para la clasificación. Como las variables independientes se asumen, solo es necesario determinar las varianzas de las variables de cada clase y no toda la matriz de covarianza. Dado un ejemplo  $x$  representado por  $k$  valores el clasificador de naive Bayes se basa en encontrar la hipótesis más probable que describa a ese ejemplo. Si la descripción de este ejemplo viene dada por los valores  $\langle a_1, a_2, \dots, a_n \rangle$ , la hipótesis más probable será aquella que cumpla:

$$v_{MAP} = \operatorname{argmax}_{v_j \in V} P(a_1, \dots, a_n | v_j) \quad (9)$$

es decir, la probabilidad de que conocidos los valores que describen a ese ejemplo, éste pertenezcan a la clase  $v_j$  (donde  $v_j$  es el valor de la función de clasificación  $f(x)$  en el conjunto finito  $V$ ). Por el teorema de Bayes.

$$v_{MAP} = \operatorname{argmax}_{v_j \in V} \frac{P(a_1, \dots, a_n | v_j) P(v_j)}{P(a_1, \dots, a_n)} \quad (10)$$

Podemos estimar  $P(v_j)$  contando las veces que aparece el ejemplo  $v_j$  en el conjunto de entrenamiento y dividiéndolo por el número total de ejemplos que forman este conjunto. Para estimar el término  $P(a_1, \dots, a_n | v_j)$ , es decir, las veces en que para cada categoría aparecen los valores del ejemplo  $x$ , debo recorrer todo el conjunto de entrenamiento. Este cálculo resulta impracticable para un número suficientemente grande de ejemplos por lo que se hace necesario simplificar la expresión. Para ellos se recurre a la hipótesis de la independencia condicional con el objeto de poder factorizar la probabilidad. [15]

En resumen, esta hipótesis nos dice que “Los valores  $a_j$  que describen un atributo de un ejemplo cualquiera  $x$  son independientes entre sí conocido el valor de la categoría a la que pertenecen” [9]. Así la probabilidad de observar la conjunción de atributos  $a_j$  dada una categoría a la que pertenecen es justamente el producto de las probabilidades de cada valor por separado:

$$P(a_1, \dots, a_n | v_j) = \prod_i P(a_i | v_j) \quad (11)$$

### III. METODOLOGÍA

La actualización de las librerías (Toolbox) expande las capacidades de Matlab para realizar determinados procesos en el área de ingeniería. En este trabajo, se consideran siete



métodos implementados en “Statistics and Machine Learning Toolbox™”. De acuerdo con las siguientes estructuras:

Paso 1. Cargar datos a un archivo tipo .m de Matlab® (script).

Paso 2. Crear un nuevo script y asignar una variable por cada vector de datos (datos muestra y datos a validar), considerando que el tamaño de ambas matrices son iguales.

Paso 3. Introducir la función estadística para crear un objeto de acuerdo con el método que se desea utilizar, colocando como argumento las variables transpuestas mencionadas en el paso anterior.

TABLA 1. NOMBRE DE LAS FUNCIONES USADAS POR MÉTODO

Técnica estadística	Función
Regresión lineal generalizada (RLG)	fitglm
Clasificación de máquinas vectoriales de soporte (MVS)	fitrsvm
Árboles de regresión (TREE)	fitrtree
Regresión de proceso gaussiano (RPG)	fitrgp
Construcción y evaluación de modelos (CEM)	fscnca
Clasificador vecino más cercano (KNN)	fitcknn
Clasificador de naive Bayes (NBC)	fitcnb

Paso 4. Utilizar la función compactar ‘compact’ para reducir el tamaño del objeto de estudio con la finalidad de observar los datos más relevantes en la ventana de comandos y asignarle una variable.

Paso 5. Asignar una variable a la función predictora ‘predict’ o ‘feval’ y colocar dentro de la función como argumentos la variable de objeto compacto y su vector de datos para validar.

Paso 6. Obtener el error cuadrático medio (MSE) con las funciones ‘loss’ o ‘immse’ para determinar la diferencia entre los datos a validar y la predicción.

Paso 7. Haciendo uso de la función ‘plot’, se obtiene en un mismo gráfico la respuesta de los datos predecidos y los datos a validar en un periodo de 24 h.

#### IV. CASOS DE ESTUDIO

Para comprobar la eficiencia de los métodos, tomamos datos reales de la demanda (4 días previos) del mercado PJM como se observa en la Fig.1, y se pronostican las siguientes 24 horas mediante las técnicas revisadas.

De la Fig. 1, podemos observar que la predicción se acerca a los datos reales, pero no se aprecia con claridad la diferencia entre sus respuestas. Por lo tanto, generamos un gráfico que compara las respuestas de los métodos y los datos reales en un periodo de 24 horas.

En Fig. 2, se presentan los datos de los pronósticos estimados utilizando las diversas técnicas descritas. Al observar el conjunto de curvas podemos notar desde un punto de vista de aproximación que la curva simulada por la técnica ARIMA es la que menos se asemeja a la curva de comportamiento real, debido a la diferencia que existe en la muestra de datos.

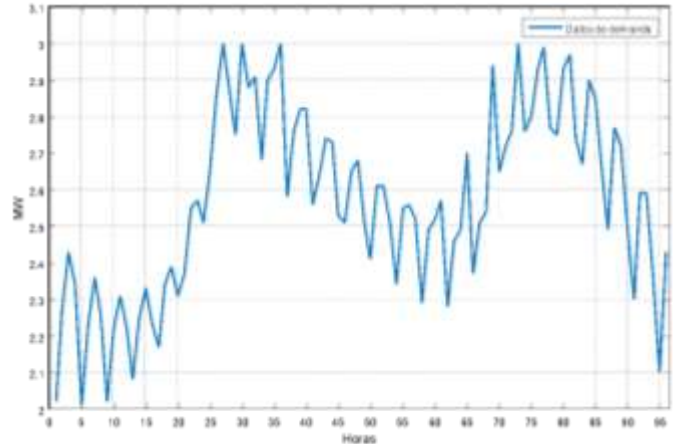


Fig. 1. Serie Temporal de la demanda para 96 horas previas.

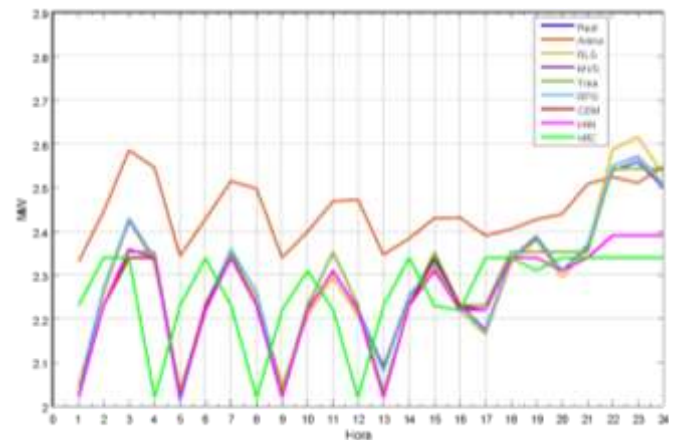


Fig. 2. Resultados de los pronósticos para 24 horas.

Notamos que las curvas tienden a un comportamiento similar, establecido en el intervalo de las 13 horas hasta las 17 horas como se aprecia en la Fig. 3.

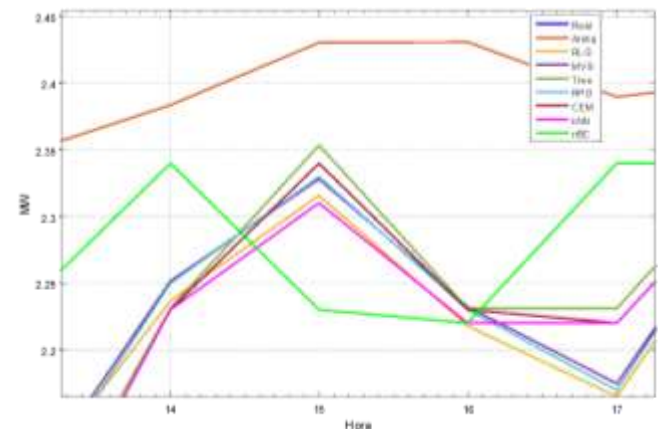


Fig. 3. Acercamiento de la hora 13 a 17.

Sin embargo, la mayor parte del tiempo las curvas presentan distintos comportamientos que nos dan la pauta para confirmar que existen algunos métodos mejores que otros para la pronosticación de datos. Consideramos que la variación entre las respuestas puede deberse a la composición matemática de los métodos, como se demostró en la sección II de este documento

Posteriormente a partir de la hora 21 se puede apreciar que las curvas obtenidas con las técnicas “Clasificador Vecino más Cercano” y “Clasificador naive Bayes” pierden totalmente la forma de la curva real, por lo tanto, no las podemos considerar como las mejores técnicas predictivas. Enseguida observamos que para la hora 22, el método que mejor modeló el comportamiento de la curva real es el método de “Regresión de Proceso Gaussiano”. Finalmente, para completar el análisis anterior, se considera el error cuadrático medio que arroja cada método. Al obtener este dato, podemos justificar el uso de una técnica que realice una predicción más precisa sobre los datos con los que trabajamos.

TABLA 3. ERROR MEDIO GEOMÉTRICO

Técnica estadística	MSE
ARIMA	$1.5 \times 10^{-3}$
Regresión lineal generalizada	$0.37 \times 10^{-3}$
Clasificación de máquinas vectoriales de soporte	$0.037 \times 10^{-3}$
Árboles de regresión	$0.099 \times 10^{-3}$
Regresión de proceso gaussiano	$5.2 \times 10^{-9}$
Construcción y evaluación de modelos	$6.1 \times 10^{-3}$
Clasificador vecino más cercano	0.66
Clasificador de naive Bayes	0.58

Con base en la Tabla 3. podemos asegurar que la técnica estadística que genera un pronóstico más preciso, a juzgar por el error que arroja, es la de Regresión de proceso Gaussiano. Sin embargo, podemos decir que las técnicas estadísticas con un error por debajo de 0.01 obtienen respuestas similares a la esperada, sin llegar a ser tan precisa como la antes mencionada. Mientras que las técnicas restantes que tienden a un error mayor a 0.01, las podemos considerar como las técnicas de menor uso para el pronóstico de datos, debido a que sus respuestas son las menos precisas para este caso.

## V. CONCLUSIÓN

Debido a las herramientas en las que se apoya la ciencia de los datos (Data Science), ha permitido que esta sea una alternativa para la gestión y predicción de datos, ya que al analizar las respuestas que se obtienen por los mismos, podemos afirmar que son superiores a la técnica ARIMA, con

base en el error cuadrático medio y en el comportamiento de la caracterización de la demanda. Sin embargo, dentro de la misma comparación de técnicas estadísticas podemos seleccionar aquella técnica que obtuvo una mejor respuesta para la predicción de datos, tomando en cuenta las mismas bases de comparación, siendo la técnica de “Regresión de Proceso Gaussiano” la que mostró una respuesta casi idéntica con respecto a nuestros datos de interés (datos reales).

A pesar de que algunas técnicas estadísticas utilizadas en este trabajo tengan inconveniente con la cantidad y calidad de datos recogidos, pueden realizar una predicción adecuada y cumplir de manera eficiente y confiable. Finalmente, el uso de las nuevas técnicas derivadas de “Machine Learning” generarán mayor confianza en la gestión de datos adquiridos diariamente para la futura automatización del sistema eléctrico en México.

## REFERENCIAS

- [1] Dahr, V. (2013). Profesor de Stern School of Business y Director en Center of Digital Economy Research, New York.
- [2] Murphy K. (2012). Machine learning a probabilistic perspective.
- [3] Mitchell, T. (1997). Doctorado en Stanford, profesor de la cátedra E. Fredkin en el departamento de Machine Learning de la escuela de Ciencias de la Computación en Carnegie Mellon University.
- [4] Calvo, Guzmán, Ramos, socios de Management Solutions (2018).
- [5] González, (2019). Proyecto de Investigación, Integración de la respuesta de la demanda a los mercados eléctricos considerando: modelos con aprendizaje y control predictivo de carga.
- [6] The MathWorks, Inc. (2020), Statistics and Machine Learning Toolbox™ Release Notes, [Online]. Available: [https://la.mathworks.com/help/pdf\\_doc/stats/rn.pdf](https://la.mathworks.com/help/pdf_doc/stats/rn.pdf)
- [7] Santiago de la Fuente Fernández, (2011). Departamento de Economía Aplicada, Facultad de Ciencias Económicas y Empresariales, “SERIES TEMPORALES: MODELO ARIMA”
- [8] Estadística para las ciencias del comportamiento. Pagano, R. Ed. Thompson. México, 1999(5a Edición).
- [9] E. Osuna, R. Freund and F. Girosi, “An Improved Training Algorithm for Support Vector Machines,” Proc. 1997 IEEE Workshop on Neural Networks for Signal Processing, pp 276-285, 1997.
- [10] Ankarali, H.; Canan, A.; Akkus, Z.; Bugdayci, R.; Ali Sungur, M.: Comparison of logistic regression model and classification tree: An application to postpartum depression data. En: Expert Systems with Applications 32 (2007), p. 987–994
- [11] Breiman, L.; Friedman, J.H.; Olshen, R.A.; Stone, C.J.: Classification and Regression Trees. Boca Raton : CHAPMAN & HALL/CRC, 1984
- [12] Rasmussen, C. E. and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press. Cambridge, Massachusetts, 2006.
- [13] Aha, D., Kibler, D., Albert, M., Instance-based learning algorithms., Machine Learning, Springer Netherlands, 1991.
- [14] Mora, J., Carrillo, G., Barrera, V., Fault Location in Power Distribution Systems Using a Learning Algorithm for Multivariable Data Analysis., IEEE Transaction on Power Delivery, Vol. 22, No. 3, July 2007, pp. 1715–1721.
- [15] Mitchell, Tom, “Machine Learning”, Ed. McGraw-Hill (1997)

