

**ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**



**FACULTAD DE CIENCIAS NATURALES Y MATEMÁTICAS  
DEPARTAMENTO DE MATEMÁTICAS**

**PROYECTO DE TITULACIÓN**

PREVIO A LA OBTENCIÓN DEL TÍTULO DE:

**“MÁGISTER EN ESTADÍSTICA CON MENCIÓN EN GESTIÓN DE  
LA CALIDAD Y PRODUCTIVIDAD”**

**TEMA:**

Pronóstico de demanda mediante series temporales y redes neuronales aplicado a una empresa del sector ferretero.

**AUTOR:**

**NÉSTOR RAFAEL MONTAÑO PULZARA**

Guayaquil - Ecuador

2020

## RESUMEN

El pronóstico de series de tiempo sigue siendo uno de los campos con constante investigación y desarrollo, desde el boom del llamado deep learning a este campo se han incluido el uso de redes neuronales como uno de los caminos para mejorar los resultados de los modelos actuales.

El presente proyecto genera 10 modelos diferentes usando modelos Box-Jenkins, Modelos estado-espacio para suavización exponencial (ETS) y una arquitectura de red neuronal con capas convolucionales, long short time memory and dense, los cuales se evalúan sobre una muestra de series mensuales de la M4 Competition para luego escoger un modelo a ser aplicado a los productos de una empresa del sector ferretero.

En esta muestra los mejores resultados, en términos del sMAPE, se obtuvieron con el modelo de estado-espacio ETS y con un modelo que es el resultado de la combinación aritmética entre los modelos de suavización exponencial y ARIMA con ciertos parámetros predefinidos.

Palabras clave: Series de tiempo, Pronóstico, Redes Neuronales, Redes LSTM

## **ABSTRACT**

The time series forecast's continues to be one of the fields with constant research and development, since the boom of the so-called deep learning, to this field the use of neural networks has been included as one of the ways to improve the results of models.

The present project generates 10 different models using Box-Jenkins models, Exponential Smoothing State-Space (ETS) models and a neural network architecture with convolutional layers, long short time memory and dense layers which are evaluated on a sample of monthly series from the M4 Competition to then choose a model to be applied to the products of a company in the hardware sector.

In the sample using the sMAPE the best results are obtained with the state-space model ETS and with a model resulting of the arithmetic combination between the models of exponential smoothing and ARIMA with predetermined parameters.

Keywords: Time series, Forecast, Neural Networks, LSTM Networks

## **DEDICATORIA**

A mi madre, qepd.

## **AGRADECIMIENTO**

A la vida.

## DECLARACIÓN EXPRESA

La responsabilidad por los hechos y doctrinas expuestas en este Proyecto de Titulación, me corresponde exclusivamente y ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias o bibliografías. Consecuentemente este trabajo es de mi total autoría. El patrimonio intelectual del mismo, corresponde exclusivamente a la ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL.

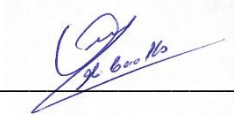
En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance del Trabajo de Titulación referido.



---

Néstor Rafael Montaña Pulzara

# TRIBUNAL DE GRADUACIÓN



---

Holger Cevallos Valdiviezo, Ph.D.

PRESIDENTE



---

Sandra García Bustos, Ph.D.

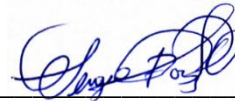
DIRECTOR



---

Francisco Moreira Villegas, M.Sc.

VOCAL



---

Sergio Bauz Olvera. Mgtr.

VOCAL

## ABREVIATURAS O SIGLAS

AIC: Criterio de información de Akaike, *Akaike information criterion*

ARIMA: Modelo Autoregresivo Media Móvil Integrado, *Autoregressive Integrated Moving Average*

CNN: Redes Neuronales Convolucionales, *Convolutional Neural Network*

EDA: Análisis Exploratorio de Datos, *Exploratory Data Analysis*

ETS: Modelo de estado-espacio para suavización exponencial, ETS corresponde a Error, Tendencia y eStacionalidad

FMI: Fondo Monetario Internacional

FFORMA: Promedio de modelos de pronóstico basado en características de la serie *Feature-based Forecast Model Averaging*

GMROI: Margen bruto del retorno de la inversión de inventario, *Gross margin return on inventory investment*

LSTM: Red Neuronal de memoria a corto y largo plazo, *Long-Short Term Memory*

MAE: Media de los errores absolutos, *Mean Absolute Error*

MAPE: Promedio del error porcentual absoluto, *Mean Absolute Percentage Error*

M4 Competition: Cuarta Competición "Makridakis", *Makridakis Competitions*

NARNN: Red Neuronal Autoregresiva No Lineal, *Nonlinear Autoregressive Neural Network*

PIB: Producto Interno Bruto

RMSE: Raíz de los errores cuadráticos, *Root mean square error*

RNN: Red Neuronal Recurrente, *Recurrent Neural Network*

sMAPE: promedio simétrico del error porcentual absoluto, *Symmetric Mean Absolute Percentage Error*



# TABLA DE CONTENIDO

ABSTRACT .....	III
DEDICATORIA .....	IV
AGRADECIMIENTO .....	V
DECLARACIÓN EXPRESA.....	VI
ABREVIATURAS O SIGLAS .....	VIII
LISTADO DE ILUSTRACIONES .....	XII
LISTADO DE TABLAS .....	XIII
CAPÍTULO 1 .....	1
1. INTRODUCCIÓN .....	1
1.1. Antecedentes.....	1
1.2. Descripción del problema .....	3
1.3. Objetivos.....	4
1.3.1. Objetivo General.....	4
1.3.2. Objetivos Específicos .....	4
1.4. Hipótesis .....	4
1.5. Alcance .....	5
CAPÍTULO 2 .....	6
2. MARCO TEÓRICO.....	6
2.1. Bases Teóricas .....	6
2.1.1. Definición de Series de tiempo .....	6
2.1.2. Patrones en Series de Tiempo: Tendencia .....	7
2.1.3. Patrones en Series de Tiempo: Estacionalidad ( <i>seasonality</i> ).....	7
2.1.4. Patrones en Series de Tiempo: Estacionariedad ( <i>stationarity</i> ).....	7
2.1.4.1. Prueba de hipótesis de Estacionariedad KPSS .....	8
2.1.5. Patrones en Series de Tiempo: Ciclicidad .....	8
2.1.6. Autocorrelación.....	8
2.2. Análisis de series de tiempo .....	9
2.2.1. Análisis automático de series de tiempo .....	10
2.2.2. Estimación de Máxima Verosimilitud .....	10
2.2.3. Criterio de información de Akaike AIC.....	11
2.3. Modelos para pronóstico de series de tiempo .....	11
2.3.1. Box-Jenkins.....	11
2.3.2. Suavización exponencial .....	12
2.3.3. Modelos de estado-espacio para suavización exponencial (ETS).....	13

2.3.4.	Redes Neuronales.....	15
2.3.5.	Redes Neuronales Convolucionales (CNN) .....	16
2.3.6.	Redes Neuronales: <i>Long-Short Term Memory</i> (LSTM).....	17
2.3.7.	Combinación de pronósticos .....	17
CAPÍTULO 3 .....		19
3.	METODOLOGÍA.....	19
3.1.	Metodología .....	19
3.1.1.	Modelamiento sobre <i>M4 Competition</i> .....	19
3.1.2.	Combinación .....	20
3.1.2.1.	Combinación usando Media Aritmética.....	20
3.1.2.2.	Combinación Mediante Redes Neuronales .....	21
3.1.3.	Selección de mejor método .....	22
3.1.4.	Implementación en Caso de estudio .....	22
3.1.5.	Software .....	22
3.2.	EDA M4-Competition .....	22
3.2.1.	M4-Competition .....	22
3.2.2.	Cantidad de años de historia .....	23
3.2.3.	Cantidad de años de historia por tipo de serie .....	24
3.2.4.	Cantidad de series por tipo.....	25
3.2.5.	Dispersión: Relación entre media, mediana y rango intercuartil.....	26
3.2.6.	Dispersión por tipo de serie .....	28
3.2.7.	Vistazo a algunas series.....	28
3.3.	Descriptivas Caso de estudio .....	33
3.3.1.	Caso de estudio: Artículos.....	33
CAPÍTULO 4 .....		35
4.	RESULTADOS.....	35
4.1.	Afinamiento de hiperparámetros <i>Tuning</i> .....	35
4.1.1.	Arquitectura de Red Neuronal .....	36
4.1.2.	Detalle del proceso de Afinamiento .....	38
4.1.3.	<i>Tuning</i> : Algunos resultados .....	38
4.2.	Resultado de pronósticos sobre las series analizadas .....	44
4.3.	Modelamiento en la data real.....	57
4.3.1.	Modelo a utilizar .....	57
4.3.2.	Resultados en series de la empresa analizada .....	57
CAPÍTULO 5 .....		62
5.	CONCLUSIONES Y RECOMENDACIONES .....	62

5.1. Conclusiones .....	62
5.2. Recomendaciones .....	63
<b>6. Referencias.....</b>	<b>65</b>

## LISTADO DE ILUSTRACIONES

Ilustración 1 Redes neuronales.....	15
Ilustración 2 Redes neuronales según órdenes por capas .....	16
Ilustración 3 Redes neuronales LSTM y RNN.....	17
Ilustración 4 Histograma y Boxplot de años de historia.....	23
Ilustración 5 Histograma de años de historia según tipo de serie .....	24
Ilustración 6 Cantidad de series según tipo de serie.....	25
Ilustración 7 Gráfico de puntos entre mediana versus rango intercuartil.....	26
Ilustración 8 Gráfico de puntos entre media acotada versus rango intercuartil .....	27
Ilustración 9 Gráfico de puntos entre media acotada versus rango intercuartil .....	27
Ilustración 10 Rango intercuartil para mediana Histograma y Boxplot .....	28
Ilustración 11 Algunas de las Series pertenecientes a la M4 Competition .....	29
Ilustración 12 Venta en unidades, 10 productos escogidos .....	33
Ilustración 13 Gráfico de estacionalidad .....	34
Ilustración 14 Arquitectura en keras.....	37
Ilustración 15 Entrenamiento en etapa de ajuste de hiperparámetros .....	39
Ilustración 16 Gráficos de series de tiempo y pronósticos generados .....	47
Ilustración 17 Caso de Estudio: Series de tiempo y pronósticos con modelo ETS58	
Ilustración 18 Caso de Estudio: Series de tiempo y pronósticos con modelo Comb2 .....	60

## LISTADO DE TABLAS

Tabla 1 Ecuaciones asociadas a los diferentes modelos ETS.....	14
Tabla 2 Cantidad series según la combinación que brinda mejor resultado .....	43
Tabla 3 Cantidad de veces que el modelo es el que tiene menor error .....	45
Tabla 4 Top 5 modelos ganadores.....	46
Tabla 5 Modelos ajustados para los productos del caso de estudio .....	57

# 1. CAPÍTULO 1

## 1. INTRODUCCIÓN

Las decisiones que cada uno de nosotros tomamos en nuestro día a día tienen un gran porcentaje de supuestos sobre el futuro, por ejemplo, nuestra decisión de vestimenta diaria se basa en nuestra estimación del clima, es decir, si creemos que será soleado, nublado, entre otras posibilidades, así, si es temporada de lluvia hacemos una predicción de que va a seguir lloviendo y por tanto cargamos un impermeable o paraguas a la mano.

En las empresas el comportamiento es similar, se emplean múltiples decisiones basadas en una estimación de lo que sucederá en el futuro, por ejemplo, la empresa resuelve abrir un nuevo local cuando espera que en el futuro ese local venda bien; desarrolla y lanza un nuevo producto, debido a que se cree que luego va a tener muchos compradores; o, por el contrario, deja de producir un producto puesto que presume que no va a tener compradores a futuro. En particular, hay una gran cantidad de decisiones de negocio que depende de la estimación de la cantidad futura que se venderá, se puede citar el número de vendedores, las metas de venta, la cantidad a comprar, el tamaño de la bodega, entre otras.

Con base a lo descrito, se puede inferir que el proceso de estimar la venta futura tiene un impacto directo en el correcto desarrollo de los negocios. El presente estudio aplica una combinación de técnicas de pronóstico a este problema aplicado en una empresa de venta de productos de ferretería del país.

### 1.1. Antecedentes

El análisis de las series de tiempo tiene sus inicios en los años 20s con los tres artículos seminales en los que Yule (1921) estudió el efecto de la correlación

dependiente del tiempo, la correlación espuria (Yule, 1926) y el último en el que generó y aplicó de un modelo autoregresivo al números de manchas solares de Wolfer (Yule, 1927). Más tarde otro gran aporte fue dado por Wold (1938) quien se centró en analizar la estacionariedad separándola en una componente determinística y una aleatoria a través de medias móviles. Sin embargo, posteriormente se empezó a tener un proceso definido para el análisis de series de tiempo, en el que se propuso un esquema que consiste en especificación, estimación, diagnóstico y pronóstico para series de tiempo individuales, esquema que seguiremos en el presente trabajo. (Box, Jenkins, Reinsel, Ljung (2016)

Por otro lado, Muth (1960) empezó a usar suavización exponencial para problemas de series de tiempo, una idea que fue desarrollándose con el tiempo, hasta que Snyder (1985) demostró que la suavización exponencial podría considerarse como un modelo de estado-espacio con innovación (*Innovation State Space Model*); luego Hyndman, Koehler, Ord y Snyder (2008) generaron su clásico libro sobre la suavización exponencial que representa una base de muchos de los avances que hoy podemos ver en este campo.

Ahora, ¿si se combinan métodos de pronóstico? La idea de combinar modelos de pronóstico de demanda no es nueva, Bates & Granger (1969) proponían mejorar el pronóstico de demanda mediante la aplicación de un promedio aritmético y promedio ponderado de dos métodos de pronóstico. Winkler & Makridakis (1983) presentaron algunos resultados empíricos correspondientes a aplicar promedio simple a varios métodos de pronóstico, concluyendo que la precisión mejora y la variabilidad entre las diferentes combinaciones disminuye a medida que aumenta el número de métodos usados en el promedio. Armstrong (1989) plantea un desafío para encontrar reglas que indiquen cómo combinar los modelos, mientras que Terui & Van Dijk (2002) extienden la investigación al uso de modelos no lineales además de proponer una técnica de coeficiente variable con respecto al tiempo para realizar las combinaciones.

En la actualidad, adicional a los métodos clásicos hay una gran popularidad y curiosidad concerniente a los métodos de aprendizaje automático. Makridakis,

Spiliotis y Assimakopoulus (2018) evalúan y comparan el rendimiento de varios métodos clásicos y de aprendizaje automático (*machine learning*) para múltiples horizontes de pronóstico. Nótese que este trabajo no combina modelos, los evalúa por separado.

Así también, la idea de combinar los pronósticos sigue siendo motivo de constante investigación. Hassani, Silva, Gupta y Das (2018) proponen una combinación de 12 modelos para pronosticar la temperatura global. Por otro lado, se combinan redes neuronales profundas con métodos clásicos de regresión para estimar el flujo de pacientes en Hong Kong. (Jiang et al., 2019)

Mientras que, en el campo del pronóstico de demanda Li, Ji y Liu (2018) trabajaron sobre las ventas online de China usando una combinación de Modelo Autoregresivo Media Móvil Integrado (ARIMA), junto con Red Neuronal Autoregresiva No Lineal (NARNN) el cual demostró ser mejor que el uso de los mismos modelos por separado.

## **1.2. Descripción del problema**

Para poder vender un producto, usualmente se requiere disponer del mismo en nuestra bodega; de igual modo, para producir o fabricar se precisa tener *stock* de materia prima. En base a esto, es lógico que, tanto para las fábricas como para empresas industriales o comercializadoras (sean de compra local o importadoras) el macro proceso correspondiente al abastecimiento sea fundamental para el funcionamiento de la institución.

Si se posee poco inventario en bodega se corre el riesgo de agotar el *stock* y, en consecuencia, dejar de vender o producir, por lo que se podría pensar entonces que, para vender o producir, es mejor tener un volumen alto de inventario y así disponer del producto en todo momento. Sin embargo, ese razonamiento nos lleva a una situación conocida como *sobrestock*, en donde, por tener un alto inventario o exceso de producto, se afecta directamente la logística del almacenamiento, pudiendo colapsar la bodega o requiriendo una bodega de mayores dimensiones



cuando no es necesario; además, se perjudican indicadores financieros, como rotación o el margen bruto del retorno de la inversión de inventario (GMROI) lo cual causa un impacto en el flujo de efectivo.

Un correcto pronóstico de demanda permitiría ajustar el abastecimiento de los productos a un balance que considere, tanto la disponibilidad de inventario, como los indicadores financieros de la empresa, evitando condiciones de *stock* cero como de *sobrestock*. Además, si este pronóstico es “temprano”, permitiría, de igual forma, detectar anomalías y tomar acciones como aumentar las compras futuras (a productos en crecimiento, mayor al esperado) o incentivar la venta (en productos de menor venta a la pronosticada anteriormente).

### **1.3. Objetivos**

#### **1.3.1. Objetivo General**

Desarrollar un algoritmo que combine metodologías de pronósticos *Box-Jenkins*, suavización exponencial (ETS por sus siglas en inglés) y redes neuronales a fin de estimar la demanda de los productos de una empresa comercial del sector ferretero.

#### **1.3.2. Objetivos Específicos**

- Obtener pronósticos de demanda usando cada familia de modelos por separado.
- Diseñar una metodología que permita combinar los pronósticos obtenidos con cada tipo de modelo.
- Comparar la metodología propuesta en este proyecto contra otros existentes para así determinar su rendimiento.

### **1.4. Hipótesis**

Un modelo de pronóstico de demanda que combine la metodología de *Box-Jenkins*, modelos de suavización exponencial de estado-espacio (*exponential smoothing*

*state space models*) y redes neuronales recurrentes o de memoria a corto y largo plazo (*long short-term memory neural network*) mejorará la precisión de los pronósticos puntuales, medidos usando el promedio del error porcentual absoluto (MAPE o *Mean Absolute Percentage Error*) en la empresa ferretera estudiada.

### **1.5. Alcance**

Para el presente proyecto, se va a trabajar con datos autocorrelacionados y estacionales correspondientes a los años 2014 al 2019 de una empresa del sector ferretero del Ecuador, además para comparar la metodología propuesta, se usará la data correspondiente a la “*M4 Competition*” (Makridakis et al., 2018) la cual es un conjunto de series que se han vuelto un estándar en cuanto a comparación de propuestas de modelos de series temporales.

# CAPÍTULO 2

## 2. MARCO TEÓRICO

La variable a analizar es la venta en unidades mensuales para cada producto, este valor es discreto en la mayoría de los productos, sin embargo, por su magnitud podemos tratarla como una variable continua; se tiene entonces, una variable numérica medida a través del tiempo en intervalos regulares (mensuales en este caso), el objetivo es pronosticar los valores futuros de la variable. Esto configura una serie temporal, en el presente capítulo se intenta brindar base teórica o facilitar el entendimiento e intuitividad detrás de los modelos a utilizar para resolver el problema detallado. (Hyndman & Athanasopoulos, 2019) y (Brockwell & Davis, 2016)

### 2.1. Bases Teóricas

#### 2.1.1. Definición de Series de tiempo

Una serie de tiempo es un conjunto de datos observados durante el tiempo, generalmente son datos regularmente espaciados, mediciones diarias, semanales, mensuales, trimestrales, por ejemplo:

- La venta para las empresas *retail* o las mayoristas,
- Las compañías de energía requieren pronosticar la reserva, producción, y demanda de electricidad,
- Las instituciones educativas analizan cantidad de alumnos a inscribirse,
- Los gobiernos pronostican ingresos y gastos tributarios.
- Los organismos financieros internacionales como el Banco Mundial y el Fondo Monetario Internacional (FMI) analizan la inflación y la actividad económica.

- Las empresas de transporte de pasajeros requieren pronosticar futuros viajes.

### **2.1.2. Patrones en Series de Tiempo: Tendencia**

Se puede entender como tendencia, el comportamiento a largo plazo de nuestra serie; la tendencia puede ser creciente (si el promedio de la serie constantemente aumenta), decreciente (si el promedio de la serie constantemente disminuye) o incluso tener una combinación de ambas.

Un ejemplo usual de este comportamiento es, por un lado, los productos cuya tecnología es nueva y que por consecuencia tienen una tendencia creciente conforme el consumidor adopta la nueva tecnología; por el contrario, los productos de tecnologías que van quedando obsoletas presentarían un comportamiento decreciente.

### **2.1.3. Patrones en Series de Tiempo: Estacionalidad (*seasonality*)**

Supongamos que se analiza juguetes para niños, a priori se puede suponer que los meses que más venta habrá serían junio (por el día del niño) y diciembre (por navidad), o si se analiza la venta de paraguas, entonces se puede inferir que la mayor venta se produzca en invierno. Este comportamiento se conoce como “estacionalidad” y se lo puede explicar como que el promedio de nuestra serie se ve afectado por la estación, mes o cualquier otro factor dependiente de la fecha.

### **2.1.4. Patrones en Series de Tiempo: Estacionariedad (*stationarity*)**

Una serie estacionaria es aquella cuyo valor de promedio, varianza y covarianza se mantiene constante en secciones del mismo ancho, es decir, sus propiedades no dependen de la fecha en que la serie ha sido observada; se puede inferir que una serie con tendencia o estacionalidad no puede ser estacionaria.

Quizá el caso más simple de serie estacionaria sería el conocido como ruido blanco, que no es más que una serie cuyas observaciones son independientes, idénticamente distribuidas y con media cero.

#### **2.1.4.1. Prueba de hipótesis de Estacionariedad KPSS**

La prueba Kwiatkowski-Phillips-Schmidt-Shin KPSS permite verificar la estacionariedad de una serie en torno a un valor determinístico.

En nuestro caso, es deseable que la serie sea estacionaria, al realizar esta prueba se espera no rechazar la hipótesis nula, dado que la prueba define sus hipótesis así:

- La hipótesis nula para la prueba es que los datos son estacionarios.
- La hipótesis alternativa para la prueba es que los datos no son estacionarios.

#### **2.1.5. Patrones en Series de Tiempo: Ciclicidad**

La ciclicidad es similar a la estacionalidad, la serie de tiempo alcanza valores máximos y mínimos, la diferencia es que el ciclo no depende de la fecha o temporada en que es medida la serie (mientras que la estacionalidad depende de la fecha en que se observa: invierno, diciembre, entre otros), además, estos no poseen duración fija dificultando la predictibilidad de los mismos. Un ejemplo conocido de la ciclicidad lo vemos cuando se analiza Producto Interno Bruto (PIB) en el cual aparece el llamado “ciclo económico de los países”.

#### **2.1.6. Autocorrelación**

El coeficiente de correlación de Pearson entre  $x$  y  $y$  mide el grado de relación lineal que existe entre estas dos variables, de tal manera que si  $\rho_{xy} = 1$ , se dice que existe una relación perfecta positiva; mientras que,  $\rho_{xy} = 0$  se dice que no existe una relación entre  $x$  y  $y$  y con  $\rho_{xy} = -1$  se dice que existe una relación perfecta negativa.

La autocorrelación sigue la misma lógica, más en lugar de observar dos variables  $x$  y  $y$ , se analiza  $x_t$  versus  $x_{t-l}$ , donde  $l$  son los “retardos”, con  $l = 1$  se estaría comparando cada observación con su inmediata anterior, con  $l = 2$  se estaría comparando cada observación con la que antecede a su inmediata anterior y así sucesivamente.

## 2.2. Análisis de series de tiempo

Al analizar series de tiempo se puede separar los siguientes procesos claves:

- Entender del problema: quizás el paso más importante y menos valorado, el analista debe interiorizar muy bien el problema para tomar las decisiones correctas antes, durante y después de modelar la serie temporal. Por ejemplo, se puede tener mediciones diarias, pero ¿si para el problema a resolver solo se necesita el total mensual?; o quizás, al entender el problema surge la hipótesis que la serie temporal que está relacionada con otra como la temperatura diaria promedio o los nacimientos mensuales. Conocer el problema te permite determinar el intervalo de tiempo que se requiere para modelar, información necesaria, entre otras cosas;
- Obtener y transformar datos: dependiendo del problema analizado se tiene toda la información disponible o no, así también es posible que los datos requieran algún proceso adicional antes de ser analizados;
- Análisis descriptivo y exploratorio de datos: involucra la visualización de la serie de tiempo tanto a) completa para detectar ciclos o tendencias, como b) con los gráficos específicos que permiten evaluar estacionalidad o autocorrelación,
- Modelamiento: existen diversas técnicas para modelar las series temporales, cada una con sus supuestos y características donde funcionan mejor; en esta parte, el objetivo es encontrar o construir el modelo que nos permita explicar o predecir con la mayor precisión;
- Mejora continua: implementación, control y perfeccionamiento del modelo

### 2.2.1. Análisis automático de series de tiempo

En la práctica es común la situación en las que se tiene muchas series de tiempo a analizar, en este caso, realizar el análisis una a una, es imposible o por tiempo o por recurso humano; se suele tomar una de dos opciones:

- 1) Se ejecuta un algoritmo para modelamiento automático a todas las series de tiempo analizadas, o
- 2) Se escoge las series más importantes para analizar manualmente y el resto se ejecutan con un algoritmo de modelamiento automático.

Los algoritmos que escogen los modelos automáticamente necesitan una forma de medir cuán buenas son las estimaciones de cada modelo y con ello escoger “el mejor”, en el presente trabajo esto se realizará en función del Criterio de Akaike AIC, el mismo que depende de la idea de la estimación de máxima verosimilitud, por lo que se procede a definir ambos conceptos.

### 2.2.2. Estimación de Máxima Verosimilitud

La idea detrás de la estimación de máxima verosimilitud es encontrar los valores de cada parámetro de tal manera que la probabilidad de que haya ocurrido la muestra observada sea la más alta posible.

Suponga que se tiene  $(x_1, x_2, \dots, x_n)$  que son los valores observados de las variables aleatorias  $(X_1, X_2, \dots, X_n)$  distribuidas según la función de densidad  $f$  con parámetros  $\theta = (\theta_1, \theta_2, \dots, \theta_k)$ , denotada  $f(x|\theta)$ ; la función de verosimilitud se define como:

$$L(\theta | x_1, x_2, \dots, x_n) = f_{X_1 X_2 \dots X_n}(x_1, x_2, \dots, x_n | \theta)$$

La expresión se leería como la verosimilitud de que  $\theta$  tome determinado valor habiendo observado los datos  $(x_1, x_2, \dots, x_n)$ , el objetivo entonces es encontrar los valores del vector de parámetros  $\theta$  que maximizan este valor.

### 2.2.3. Criterio de información de Akaike AIC

El criterio de información de Akaike (AIC) permite evaluar qué tan bien un modelo se ajusta al conjunto de datos sin tener sobreajuste; su principal utilidad es comparar modelos; la idea es que, entre dos modelos comparables el modelo que tenga menor AIC es el mejor.

El AIC recompensa los modelos en función de la bondad de ajuste a la estimación de máxima verosimilitud mientras que penaliza a los modelos dependiendo de la complejidad que tienen.

Se define al criterio de información de Akaike (AIC) como:

$$AIC = 2k - \ln ( L(\theta | x_1, x_2, \dots, x_n) ), \text{ donde } \theta = (\theta_1, \theta_2, \dots, \theta_k)$$

## 2.3. Modelos para pronóstico de series de tiempo

### 2.3.1. Box-Jenkins

La metodología *Box-Jenkins* para el modelamiento de series temporales aparece en el libro: *Time Series Analysis Forecasting and Control* (Box et al., 2016), uno de los libros más influyentes en la historia de la estadística; en este libro se desarrolla el análisis de series temporales usando los modelos ARIMA y bajo un esquema iterativo de selección del modelo, estimación de parámetros y comprobación del modelo

La familia de modelos ARIMA (p, d, q) se pueden resumir en:

$$y'_t = c + \phi_1 y'_{t-1} + \dots + \phi_p y'_{t-p} + \theta_1 \varepsilon_{t-1} + \dots + \theta_q \varepsilon_{t-q} + \varepsilon_t,$$

donde:

- $\phi_l$  es el coeficiente de la parte autoregresiva de orden  $l$ , es decir la parte que intenta predecir  $y_t$  en función de los  $p$  retardos  $y_l$ .



- $\theta_l$  es el coeficiente de la parte media móvil de orden  $l$ , es decir la parte que intenta predecir  $y_t$  en función de los  $q$  errores  $\varepsilon_l$ .
- $y'_t$  es la serie de tiempo diferenciada  $d$  veces, la diferenciación es el proceso mediante el cual uno trata de convertir una serie “no estacionaria” en “estacionaria” mediante la resta de valores sucesivos (similar a tomar los cambios diarios, semanales o mensuales en lugar de la serie diaria, semanal o mensual respectivamente).

### 2.3.2. Suavización exponencial

En la componente autoregresiva de los modelos ARIMA,  $y_t$  depende de  $p$  retardos asignándole a cada uno un coeficiente diferente, el modelo más básico de los modelos ETS es similar:

$$\hat{y}_{T+1|T} = \alpha y_T + \alpha(1 - \alpha)y_{T-1} + \alpha(1 - \alpha)^2 y_{T-2} + \dots$$

Donde,  $0 \leq \alpha \leq 1$  es el parámetro de suavización.

Nótese dos diferencias significativas:

1. En este modelo el coeficiente que acompaña a los valores de los retardos  $y_{T-l}$  depende de un sólo parámetro:  $\alpha$ , el cual va decayendo a razón de  $(1 - \alpha)^l$  elevado al índice del retardo; esto implica que las observaciones más antiguas tienen menor peso,
2. En ARIMA ( $p, d, q$ ), los parámetros “ $p$ ” y “ $q$ ” controlaban la cantidad de retardos a usar, en suavización exponencial se usan todos los datos anteriores  $y_1, y_2, \dots, y_T$ .

Con un poco de matemáticas la ecuación anterior puede transformarse en:

$$\hat{y}_{T+1|T} = \sum_{j=0}^{T-1} \alpha (1 - \alpha)^j y_{T-j} + (1 - \alpha)^T \ell_0.$$

Con lo que se tiene como parámetros  $\alpha$  y  $\ell_0$ , donde  $\ell_0$  se entiende como el nivel inicial de la serie. Esto a su vez se puede reescribir como:

Forecast equation:  $\hat{y}_{t+h|t} = \ell_t$

Smoothing equation:  $\ell_t = \alpha y_t + (1 - \alpha)\ell_{t-1}$

Esta misma lógica se aplica a la tendencia y estacionalidad, teniendo esta última dos formas, aditiva y multiplicativa.

### **2.3.3. Modelos de estado-espacio para suavización exponencial (ETS)**

Los modelos de estado-espacio se basan en una visión estructural en la que se tiene por un lado una variable o proceso escondido que define el estado y por otro las observaciones mismas. Existen varias implementaciones de modelos estado-espacio para suavización exponencial, nos centraremos en el trabajo de Hyndman, Koehler, Snyder y Grose (2002) el cual considera una sola fuente de error para el proceso, este error puede ser multiplicativo o aditivo lo cual afecta principalmente a la estimación de la varianza, no al pronóstico puntual.

Los modelos de estado-espacio para suavización exponencial establecen una combinación de Error, Tendencia, Estacionalidad donde:

- El error puede ser Aditivo o Multiplicativo {A,M },
- La tendencia puede ser Nula, Aditiva o Aditiva amortiguada {N,A,A d}, y
- La estacionalidad puede ser Nula, Aditiva o Multiplicativa {N,A,M }

Quedando finalmente 18 posibles modelos detallados en el gráfico a continuación:

**Tabla 1 Ecuaciones asociadas a los diferentes modelos ETS**

*ADDITIVE ERROR MODELS*

Trend	Seasonal		
	N	A	M
<b>N</b>	$y_t = \ell_{t-1} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \alpha\varepsilon_t$	$y_t = \ell_{t-1} + s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \alpha\varepsilon_t$ $s_t = s_{t-m} + \gamma\varepsilon_t$	$y_t = \ell_{t-1}s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \alpha\varepsilon_t/s_{t-m}$ $s_t = s_{t-m} + \gamma\varepsilon_t/\ell_{t-1}$
<b>A</b>	$y_t = \ell_{t-1} + b_{t-1} + \varepsilon_t$ $\ell_t = \ell_{t-1} + b_{t-1} + \alpha\varepsilon_t$ $b_t = b_{t-1} + \beta\varepsilon_t$	$y_t = \ell_{t-1} + b_{t-1} + s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + b_{t-1} + \alpha\varepsilon_t$ $b_t = b_{t-1} + \beta\varepsilon_t$ $s_t = s_{t-m} + \gamma\varepsilon_t$	$y_t = (\ell_{t-1} + b_{t-1})s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + b_{t-1} + \alpha\varepsilon_t/s_{t-m}$ $b_t = b_{t-1} + \beta\varepsilon_t/s_{t-m}$ $s_t = s_{t-m} + \gamma\varepsilon_t/(\ell_{t-1} + b_{t-1})$
<b>A<sub>d</sub></b>	$y_t = \ell_{t-1} + \phi b_{t-1} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha\varepsilon_t$ $b_t = \phi b_{t-1} + \beta\varepsilon_t$	$y_t = \ell_{t-1} + \phi b_{t-1} + s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha\varepsilon_t$ $b_t = \phi b_{t-1} + \beta\varepsilon_t$ $s_t = s_{t-m} + \gamma\varepsilon_t$	$y_t = (\ell_{t-1} + \phi b_{t-1})s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha\varepsilon_t/s_{t-m}$ $b_t = \phi b_{t-1} + \beta\varepsilon_t/s_{t-m}$ $s_t = s_{t-m} + \gamma\varepsilon_t/(\ell_{t-1} + \phi b_{t-1})$

*MULTIPLICATIVE ERROR MODELS*

Trend	Seasonal		
	N	A	M
<b>N</b>	$y_t = \ell_{t-1}(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1}(1 + \alpha\varepsilon_t)$	$y_t = (\ell_{t-1} + s_{t-m})(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1} + \alpha(\ell_{t-1} + s_{t-m})\varepsilon_t$ $s_t = s_{t-m} + \gamma(\ell_{t-1} + s_{t-m})\varepsilon_t$	$y_t = \ell_{t-1}s_{t-m}(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1}(1 + \alpha\varepsilon_t)$ $s_t = s_{t-m}(1 + \gamma\varepsilon_t)$
<b>A</b>	$y_t = (\ell_{t-1} + b_{t-1})(1 + \varepsilon_t)$ $\ell_t = (\ell_{t-1} + b_{t-1})(1 + \alpha\varepsilon_t)$ $b_t = b_{t-1} + \beta(\ell_{t-1} + b_{t-1})\varepsilon_t$	$y_t = (\ell_{t-1} + b_{t-1} + s_{t-m})(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1} + b_{t-1} + \alpha(\ell_{t-1} + b_{t-1} + s_{t-m})\varepsilon_t$ $b_t = b_{t-1} + \beta(\ell_{t-1} + b_{t-1} + s_{t-m})\varepsilon_t$ $s_t = s_{t-m} + \gamma(\ell_{t-1} + b_{t-1} + s_{t-m})\varepsilon_t$	$y_t = (\ell_{t-1} + b_{t-1})s_{t-m}(1 + \varepsilon_t)$ $\ell_t = (\ell_{t-1} + b_{t-1})(1 + \alpha\varepsilon_t)$ $b_t = b_{t-1} + \beta(\ell_{t-1} + b_{t-1})\varepsilon_t$ $s_t = s_{t-m}(1 + \gamma\varepsilon_t)$
<b>A<sub>d</sub></b>	$y_t = (\ell_{t-1} + \phi b_{t-1})(1 + \varepsilon_t)$ $\ell_t = (\ell_{t-1} + \phi b_{t-1})(1 + \alpha\varepsilon_t)$ $b_t = \phi b_{t-1} + \beta(\ell_{t-1} + \phi b_{t-1})\varepsilon_t$	$y_t = (\ell_{t-1} + \phi b_{t-1} + s_{t-m})(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha(\ell_{t-1} + \phi b_{t-1} + s_{t-m})\varepsilon_t$ $b_t = \phi b_{t-1} + \beta(\ell_{t-1} + \phi b_{t-1} + s_{t-m})\varepsilon_t$ $s_t = s_{t-m} + \gamma(\ell_{t-1} + \phi b_{t-1} + s_{t-m})\varepsilon_t$	$y_t = (\ell_{t-1} + \phi b_{t-1})s_{t-m}(1 + \varepsilon_t)$ $\ell_t = (\ell_{t-1} + \phi b_{t-1})(1 + \alpha\varepsilon_t)$ $b_t = \phi b_{t-1} + \beta(\ell_{t-1} + \phi b_{t-1})\varepsilon_t$ $s_t = s_{t-m}(1 + \gamma\varepsilon_t)$

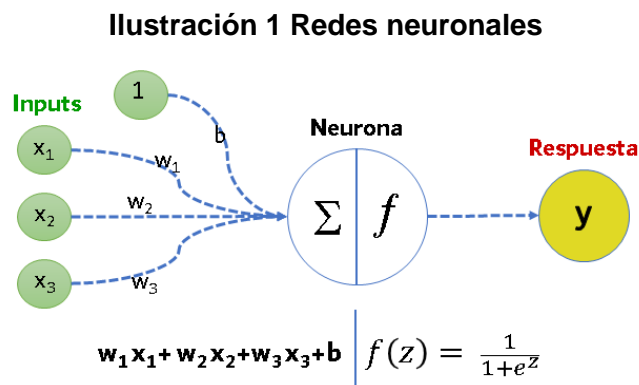
Fuente: Hyndman & Athanasopoulos (2019)

### 2.3.4. Redes Neuronales

Siguiendo las ideas expuestas por Ghatak (2017) y también por Ramasubramanian & Singh (2019); las redes neuronales son un método que intenta emular el funcionamiento del cerebro humano, empezamos entendiendo el criterio bajo el que se pensó este modelo. El cerebro es una agrupación de neuronas que interactúan unas con otras a través de la sinapsis transmitida por las dendritas, la forma de interactuar es tal que una neurona emite una señal que luego es recibida por otra, la neurona receptora en función de la señal que recibe puede aumentarla o disminuirla para a su vez pasarla a otras neuronas.

La base para este tipo de modelos son las neuronas, estas constan de 3 partes, resumidas en la ilustración 1:

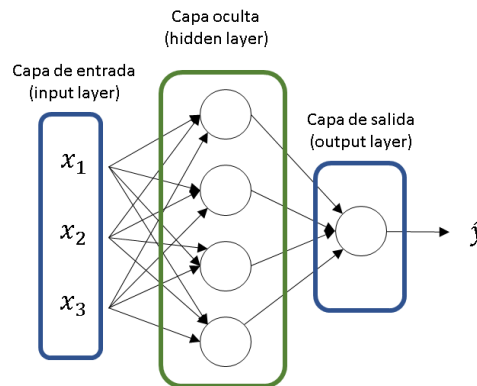
- Conexiones de entrada: son las que se encargan de recibir el input y de asignarle un peso
- El cuerpo de la neurona que consta a su vez de dos elementos:
  - La combinación de los pesos y entradas; y
  - Una función de activación, en la imagen la función de activación es un sigmoideo, pero existe un sinnúmero de funciones que se puede aplicar.
- La salida: no es más que el resultado de aplicar la función de activación a la combinación lineal de pesos y entradas de la neurona.



*Fuente: Elaboración propia*

Las redes neuronales (ilustración 2) corresponden a una agrupación de neuronas ordenadas por capas, donde la primera capa cumple la función de input (variables de entrada) y la última da la salida (predicción o pronóstico) del modelo, la imagen muestra un ejemplo de red neuronal con una capa oculta y totalmente conectada.

**Ilustración 2 Redes neuronales según órdenes por capas**



*Fuente: Elaboración propia*

### 2.3.5. Redes Neuronales Convolucionales (CNN)

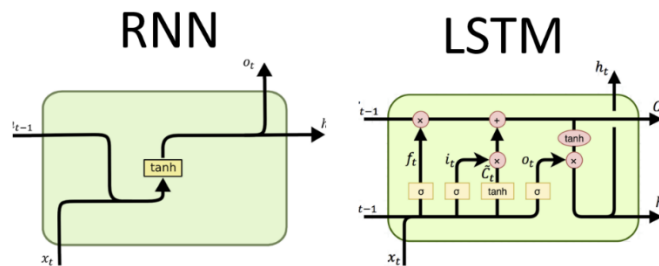
Una convolución es una operación matemática cuyo objetivo es detectar características x-dimensionales de las observaciones, las redes neuronales convolucionales son las que usan esta operación en una o varias de sus capas. El uso más conocido de las redes convoluciones (y responsable de su popularización) es la detección de objetos en imágenes, en esos casos se trata a cada canal de la imagen (RGB, Red, Green, Blue) como una matriz y se opera sobre estas matrices usando varias capas convolucionales de tal manera que las primeras capas permiten detectar líneas rectas, inclinadas, curvas y al agregar más capas se puede detectar figuras geométricas y formas; este caso se conoce como capas Convolucionales en 2D.

La misma lógica, pero 1-dimensional se puede aplicar a las series de tiempo de tal manera que se reconozcan patrones temporales como la estacionalidad. Este tipo de arquitectura para redes neuronales es nuevo, un ejemplo de su uso para predicción de series temporales puede verse en Harbola & Coors (2019.).

### 2.3.6. Redes Neuronales: *Long-Short Term Memory* (LSTM)

Las redes neuronales LSTM son un tipo de red neuronal recurrente, RNN (ilustración 3); la idea detrás de las RNN es construir modelos los que para cada punto consideren la información previa al hacer su predicción, algo que se ajusta a lo que requerimos al analizar series temporales. El tipo de red neuronal más aplicada para este tipo de análisis son las redes de memoria a corto y largo plazo (LSTM) las cuales fueron desarrolladas pensando en que hay ciertas características que deben ser “guardadas” para ser utilizadas luego (de ahí la parte de memoria a largo plazo), con esto se resuelve un problema de las RNN que tienen “memoria corta”.

Ilustración 3 Redes neuronales LSTM y RNN



Fuente: Pytrick (2019)

### 2.3.7. Combinación de pronósticos

Desde que Bates & Granger (1969) escribieron su paper: *The Combination of Forecasts*, se ha venido probando que, combinar el resultado de modelos de pronóstico de demanda suele dar mejores resultados que los modelos independientemente, esto se puede ver en los resultados de las competencias M3 (Makridakis & Hibon, 2000) así como en M4, el cual, posee frases tan contundentes como: “La combinación de métodos fue el rey de la M4. De los 17 métodos más precisos, 12 eran ‘combinaciones’ de enfoques mayormente estadísticos” (Makridakis et al., 2018)

La combinación de métodos para realizar pronósticos se puede realizar mediante un simple proceso de promediar las estimaciones puntuales, enfoque que se puede intentar mejorar usando un promedio ponderado en lugar de un promedio simple, en cuyo caso el desafío es determinar los pesos para la ponderación. Así también, se puede complicar aún más la forma de combinar llegando a métodos como el método *Feature-based Forecast Model Averaging* (FFORMA) (Montero-Manso, Athanasopoulus, Hyndmam y Talagala (2020). Por otra parte, Atiya (2019) discute sobre porqué combinar pronósticos es buena estrategia.

# CAPÍTULO 3

## 3. METODOLOGÍA

En el presente capítulo se explica la metodología a utilizar para el proyecto, así como se explora las series de la M4 (Makridakis et al., 2018) y los productos a analizar de la empresa estudiada.

### 3.1. Metodología

Para el presente estudio se va a seguir un proceso dividido en dos partes

- 1) Modelos simples: Ajustar los métodos de series de tiempo (aprendizaje estadístico) sobre una muestra aleatoria de tamaño 100 de las series con frecuencia mensual de la *M4 Competition*
- 2) Combinar modelos:
  - a. Calcular el pronóstico combinando alguno de los modelos de pronóstico con base estadística, y
  - b. Entrenar un modelo de Red neuronal que tome como entrada los valores ajustados de los modelos ajustados anteriormente.
- 3) Realizar un análisis comparativo del pronóstico de 18 meses de los modelos ajustados previamente versus lo que en realidad debieron pronosticar y adicionalmente comparar estos resultados con el top 5 de modelos con menor error planteados en la *M4 Competition*.
- 4) Aplicar el mejor modelo a los productos estudiados.

#### 3.1.1. Modelamiento sobre *M4 Competition*

Como primera parte del proceso, se definen varios métodos que usarán como base los métodos descritos en el marco teórico, incluyendo el método propuesto para combinarlos, a saber:



- Un modelo de estado-espacio para suavización exponencial cuyas componentes de error, tendencia, estacionalidad, ajustado en función del AIC y máxima verosimilitud.
- Un modelo ARIMA(p, d, q)(P, D, Q) ajustado automáticamente ajustado según el algoritmo Hyndman-Khandakar (Hyndman & Yeasmin, 2008).
- Un modelo ARIMA (p, d, q)(P, D, Q) que se ha definido de tal manera que se le obligue tener una componente estacional, es decir (P, D, Q) tal que P + Q debe ser mayor a cero.
- Un modelo ARIMA(1, d, 0)(P, D, Q), es decir se exige al modelo tener una componente autoregresiva.
- Un modelo ARIMA(2, d, 0)(P, D, Q), es decir se exige al modelo tener dos componentes autoregresivos.
- Un modelo ARIMA(0, d, 1)(P, D, Q), es decir se exige al modelo tener una componente media móvil.
- Un modelo ARIMA(0, d, 2)(P, D, Q), es decir se exige al modelo tener dos componentes media móvil

Los últimos modelos se plantean basados en la experiencia previa del investigador usando el algoritmo Hyndman-Khandakar.

### **3.1.2. Combinación**

#### **3.1.2.1. Combinación usando Media Aritmética**

Se plantea dos modelos en los que simplemente se obtenga la media aritmética de los modelos

- a) 1, 2 y 3
- b) 1, 4, 5, 6, 7

### 3.1.2.2. Combinación Mediante Redes Neuronales

Para este proyecto se ha construido una arquitectura de red neuronal que tiene múltiples entradas y adicionalmente combina capas Convolucionales, Long Short Time Memory LSTM y Dense, en detalle:

- a) Entradas
  - Input A: Serie de tiempo univariante
  - Input B: Serie de tiempo multivariante: Valor analizado en conjunto con los valores ajustados de los modelos de pronósticos ejecutados en el primer literal.
- b) Detectar características mediante filtros convolucionales
  - Capa Convolutiva con ventana convolutiva de tamaño 3 que toma como entrada la serie el input A
  - Capa Convolutiva con ventana convolutiva de tamaño 6 que toma como entrada la serie el input A
  - Capa Convolutiva con ventana convolutiva de tamaño 12 que toma como entrada la serie el input A
- c) Transformar a problema multivariante
  - 3 Capas, la primera de “aplanamiento” *Flatten* conectadas a una capa completa *Dense* y conectadas a una capa de redimensionamiento *Reshape*; esto aplicado a cada layer convolutiva.
  - Capa concatenada *Concatenate* que toma el resultado de las tres capas convolucionales transformadas más el Input B (este input B impide que esta red se construya de forma secuencial, este *input* entraría como *residual connection* o *skip connection*)
- d) Detectar patrones recurrentes con LSTM
  - Capa LSTM multivariante que recibe la capa concatenada
  - Capa Dense que estará encargada de ser el primer filtro para interpretar la salida de la capa LSTM
  - Capa Dense que recibe la salida de la capa Dense anterior y es la que nos proporciona el output del modelo

Para este modelo, se va a ajustar “*tunning*”, el parámetro de cantidad de filtros convolucionales a utilizar, para esto se seguirá una estrategia de remuestreo de tipo ventana rodante, en el que se tendrá siempre 5 cortes y 40 *epochs*; para cada muestra se calcula medidas de ajuste, y con estos valores a su vez se calcula la efectividad del modelo con cada hiperparámetro y se escogerá la combinación a usar para modelar sobre la data total.

### **3.1.3. Selección de mejor método**

Con los resultados de aplicar los modelos en las series *M4 Competition*, se procede a la evaluación de resultados y comparación de los métodos, para ello se usarán una medida de precisión que se usa en la *M4 Competition*, la media absoluta del error porcentual simétrico sMAPE. (Makridakis, 1993)

### **3.1.4. Implementación en Caso de estudio**

Finalmente se aplicará el método con mejor precisión a las series de la empresa analizada.

### **3.1.5. Software**

Para el presente proyecto se usará como base el software estadístico R (Team, 2013) mediante el cual se accederá a keras versión 2.3.1 (Chollet, 2016) usando *Tensorflow* versión 2.0.0 (Girija, 2016) como *backend* corriendo esto sobre Python 3.6. (Van Rossum & Drake, 2011).

## **3.2. EDA M4-Competition**

### **3.2.1. M4-Competition**

Las *M-Competitions* son una serie de concursos abiertos al público cuyo objetivo es encontrar el método de pronóstico más preciso, la más reciente fue la *M4-Competition* citada por Makridakis, Spiliotis y Assimakopoulos (2020), en la cual

tuvo 100.000 series entre anuales, trimestrales, mensuales, semanales, diarias y con observaciones a cada hora; las mensuales son 48.000 divididas en 6 grupos: series microempresariales, industriales, macroeconómicas, financieras, demográficas y otras.

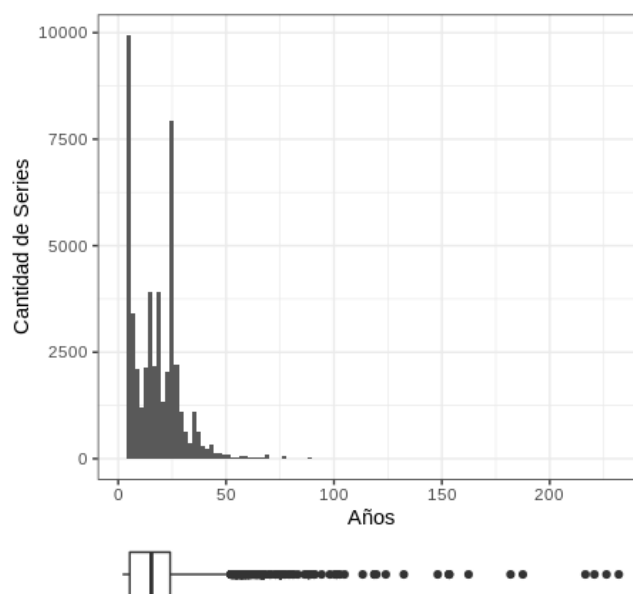
Las *M-Competitions* están enfocadas en series continuas, se excluyen series intermitentes o de valores pequeños; enfoque que va acorde al tipo de serie que necesitamos evaluar en este proyecto.

### 3.2.2. Cantidad de años de historia

Las 48.000 series que constituyen el grupo de series mensuales del *M4-Competition* tienen una duración promedio de 18,02 años con un mínimo y máximo de 3,5 y 232,83 años respectivamente; además un 50% de series dura menos de 16,83 años y el 25% dura más de 25,5 años.

En la ilustración 4 se aprecia que la distribución de años se acumula hacia valores menores, así como se percibe que el *boxplot* detecta una gran cantidad de *outliers* univariante.

**Ilustración 4 Histograma y Boxplot de años de historia**

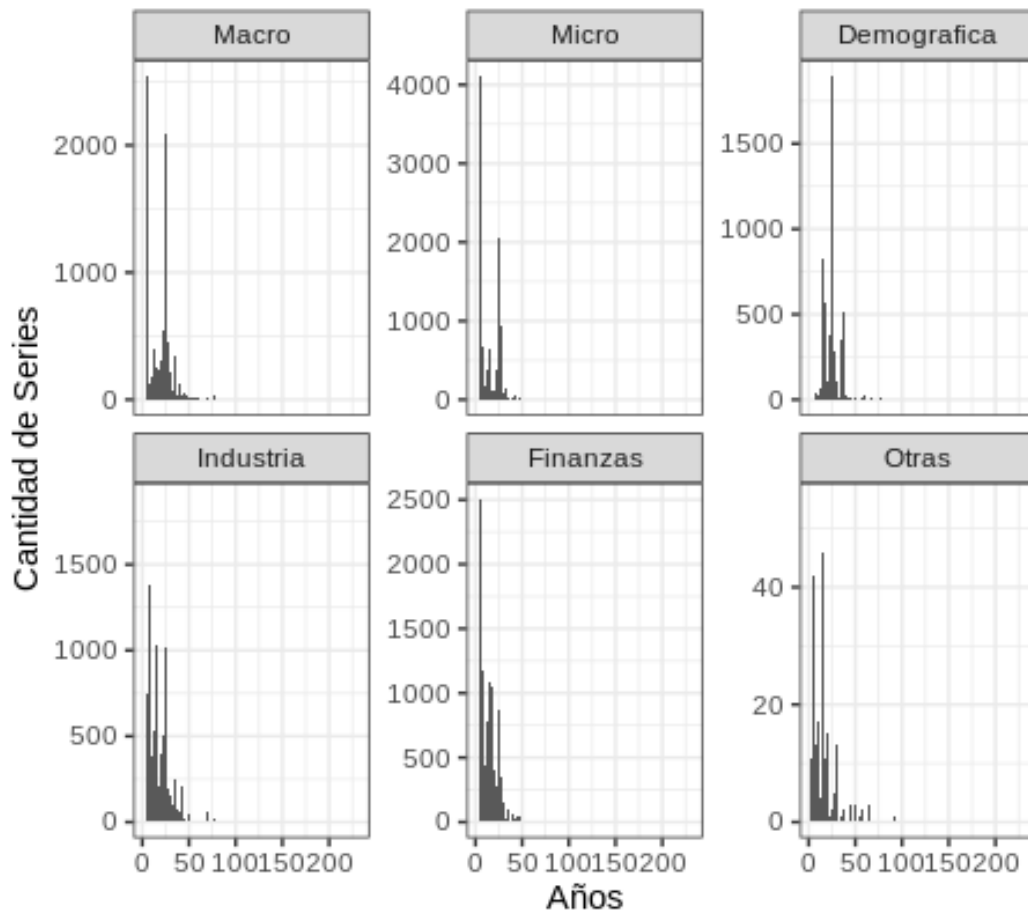


Fuente: Elaboración propia

### 3.2.3. Cantidad de años de historia por tipo de serie

Al inicio del capítulo, se menciona que las 48.000 series están divididas en 6 grupos de series: microempresariales, industriales, macroeconómicas, financieras, demográficas y otras. En la ilustración 5, se muestra la distribución de la cantidad de años de historia según le tipo de serie analizada, se puede visualizar que se mantiene el comportamiento descrito anteriormente, se tiene que las distribuciones son sesgadas a la derecha.

**Ilustración 5 Histograma de años de historia según tipo de serie**

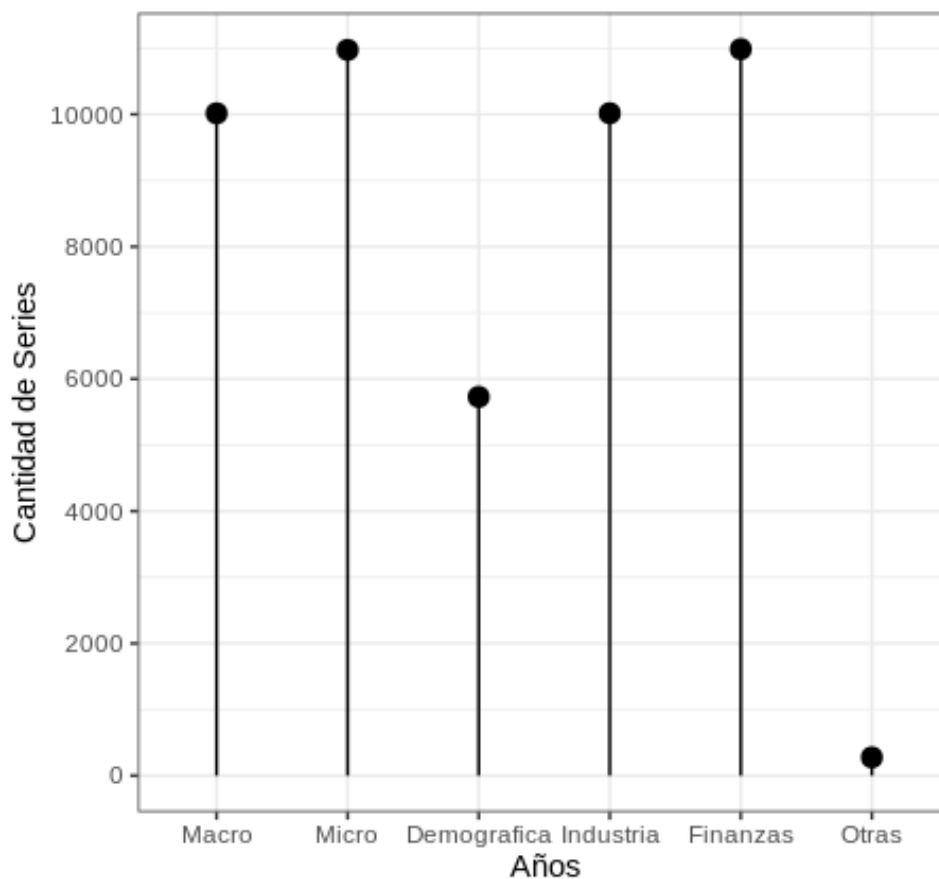


*Fuente: Elaboración propia*

### 3.2.4. Cantidad de series por tipo

Siguiendo el ejemplo mencionado al inicio del capítulo, en la ilustración 6 se muestra la cantidad de series analizadas por tipo, se puede ver que las series macro, micro, industria y financieras tienen una cantidad parecida, mientras que las demográficas son poco más de la mitad de estas.

**Ilustración 6 Cantidad de series según tipo de serie**

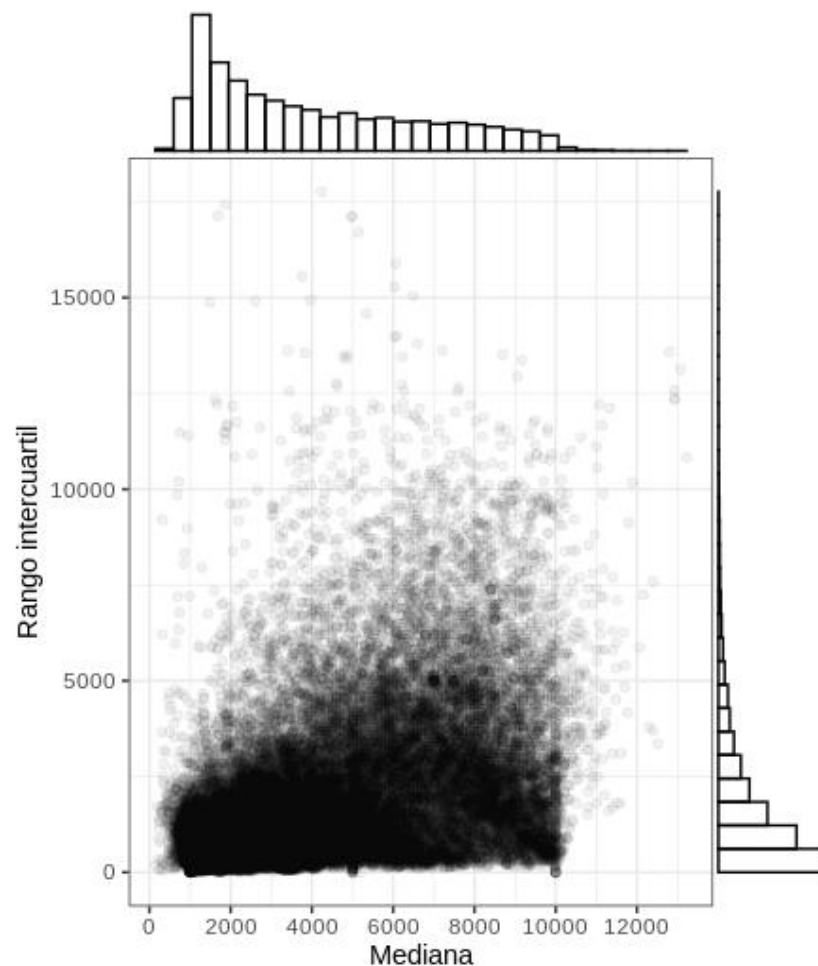


*Fuente: Elaboración propia*

### 3.2.5. Dispersión: Relación entre media, mediana y rango intercuartil

Además de la cantidad de meses o años de historia, es interesante saber cuán dispersas están los valores de nuestras series de tiempo; para ello, vamos a comparar el rango intercuartil para la mediana, se puede ver que los valores se concentran debajo de la diagonal identidad lo que indica que en su mayoría el rango intercuartil es menor a la mediana, mostrada en la ilustración 7.

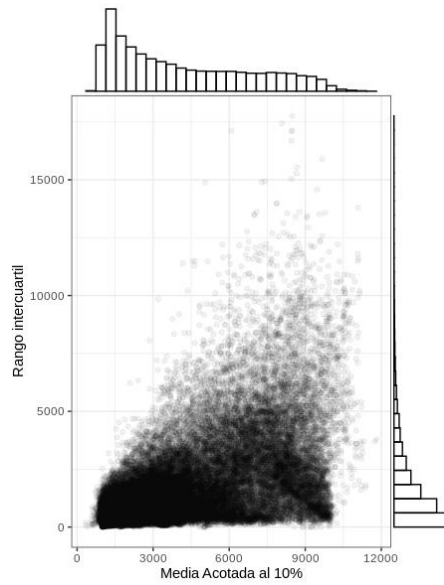
Ilustración 7 Gráfico de puntos entre mediana versus rango intercuartil



Fuente: Elaboración propia

Si realizamos el mismo ejercicio para la media da un resultado parecido, según muestra la ilustración 8:

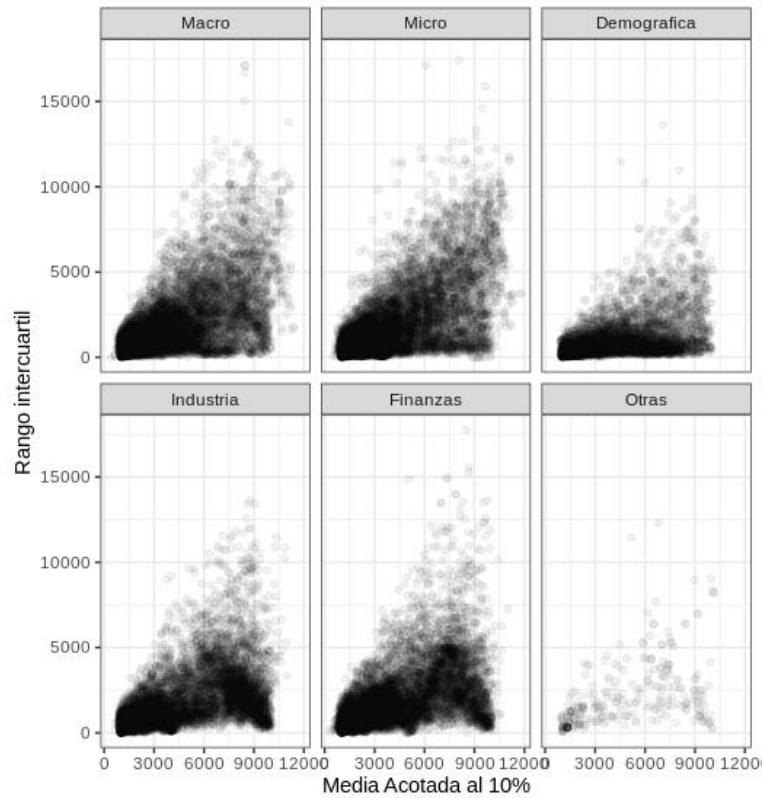
**Ilustración 8 Gráfico de puntos entre media acotada versus rango intercuartil**



*Fuente: Elaboración propia*

Comportamiento que se mantiene por tipo de serie (ilustración 9)

**Ilustración 9 Gráfico de puntos entre media acotada versus rango intercuartil**



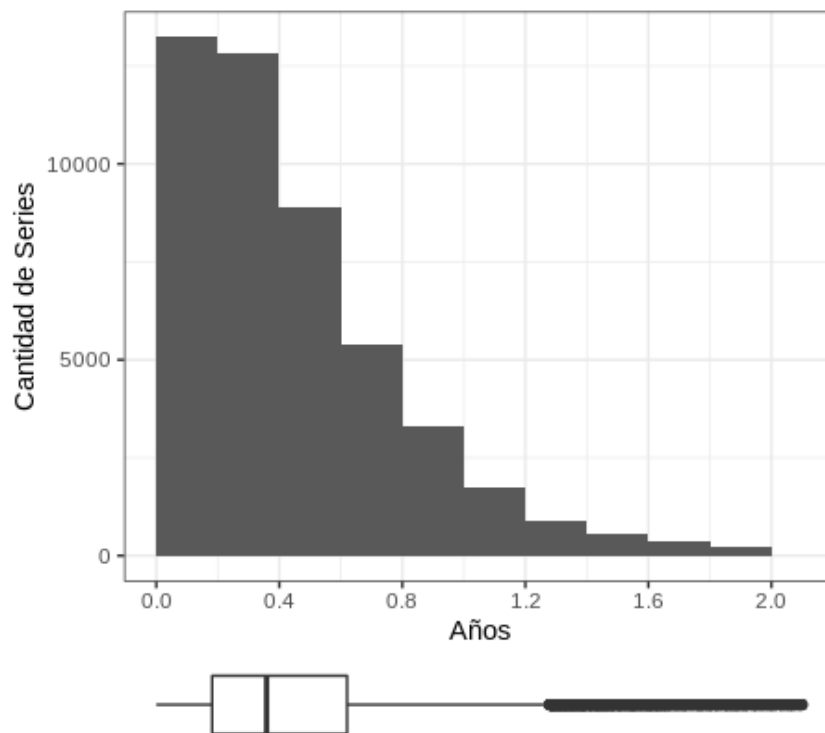
*Fuente: Elaboración propia*



### 3.2.6. Dispersión por tipo de serie

Para seguir explorando la variabilidad de los valores en nuestras series, se escoge como indicador de dispersión estandarizado la razón entre el rango intercuartil versus la mediana, la ilustración 10, muestra un histograma acotado (se deja de lado 547 series cuya ratio es mayor a 2), se puede ver que una gran parte de nuestras series tienen un ratio  $iqr/mediana$  menor a 0,4, lo que nos indica una variabilidad contenida.

**Ilustración 10** Rango intercuartil para mediana Histograma y Boxplot

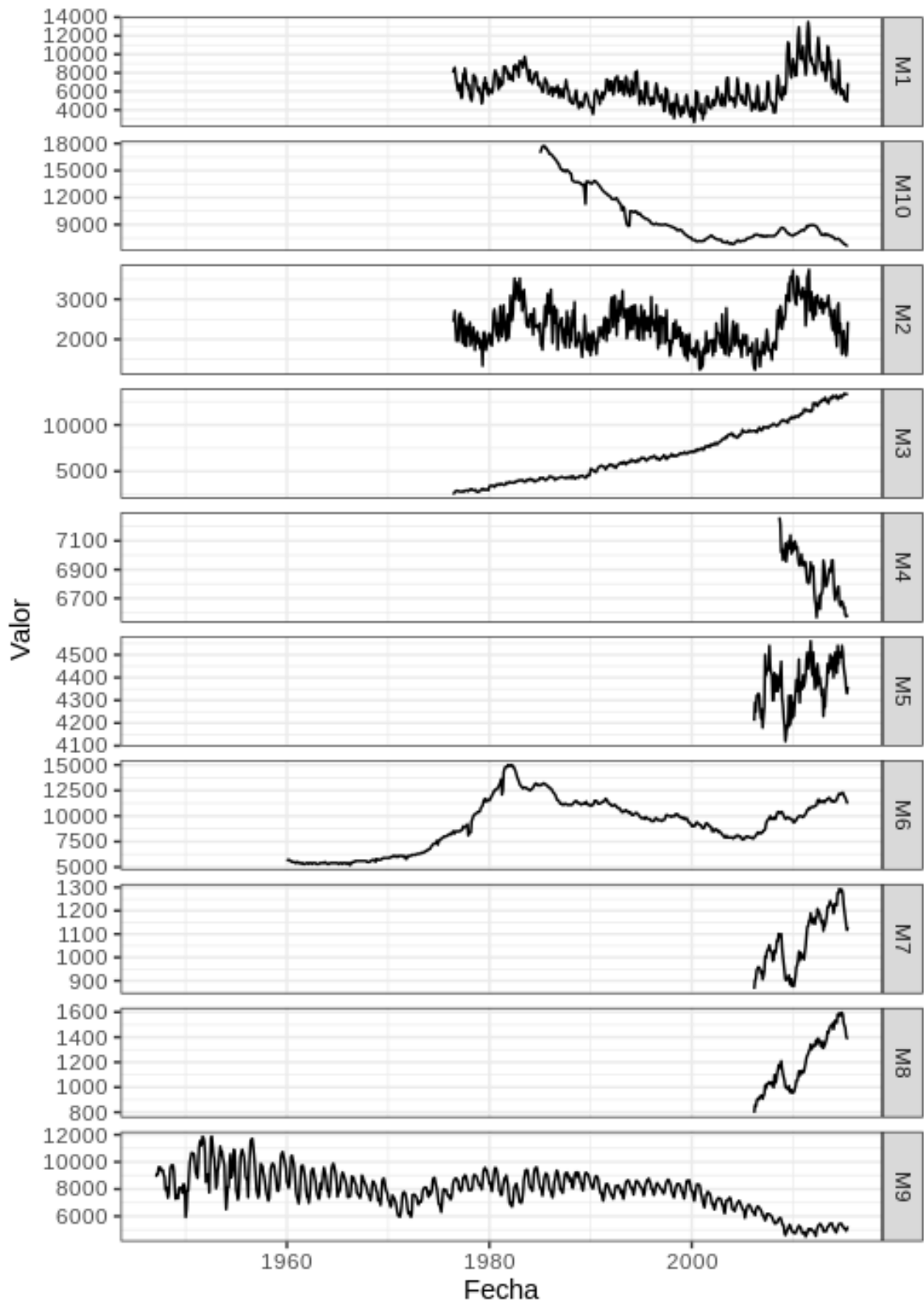


*Fuente: Elaboración propia*

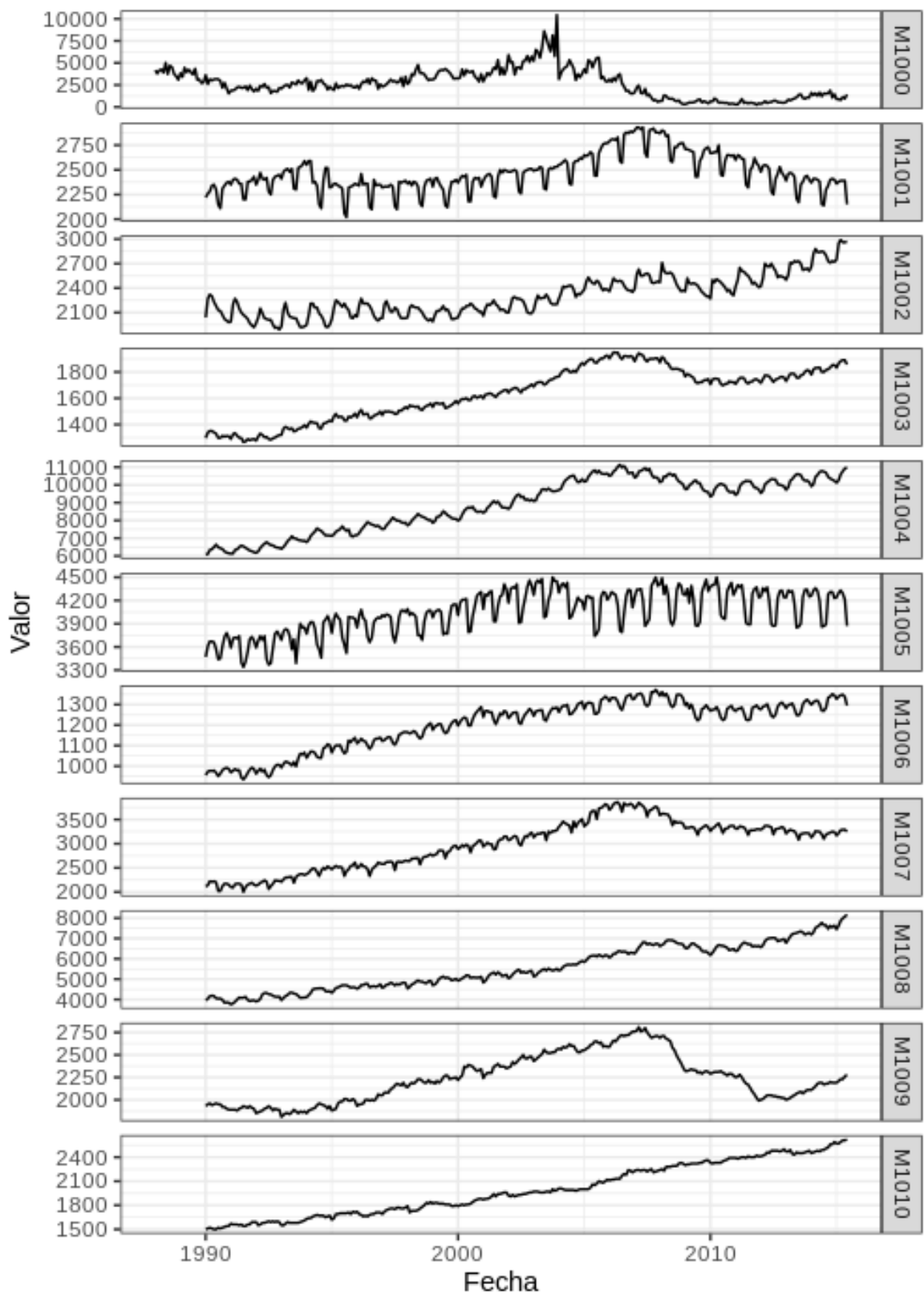
### 3.2.7. Vistazo a algunas series

Finalizamos esta sección con un vistazo de algunas de las series analizadas, según como se puede apreciar en la ilustración 11:

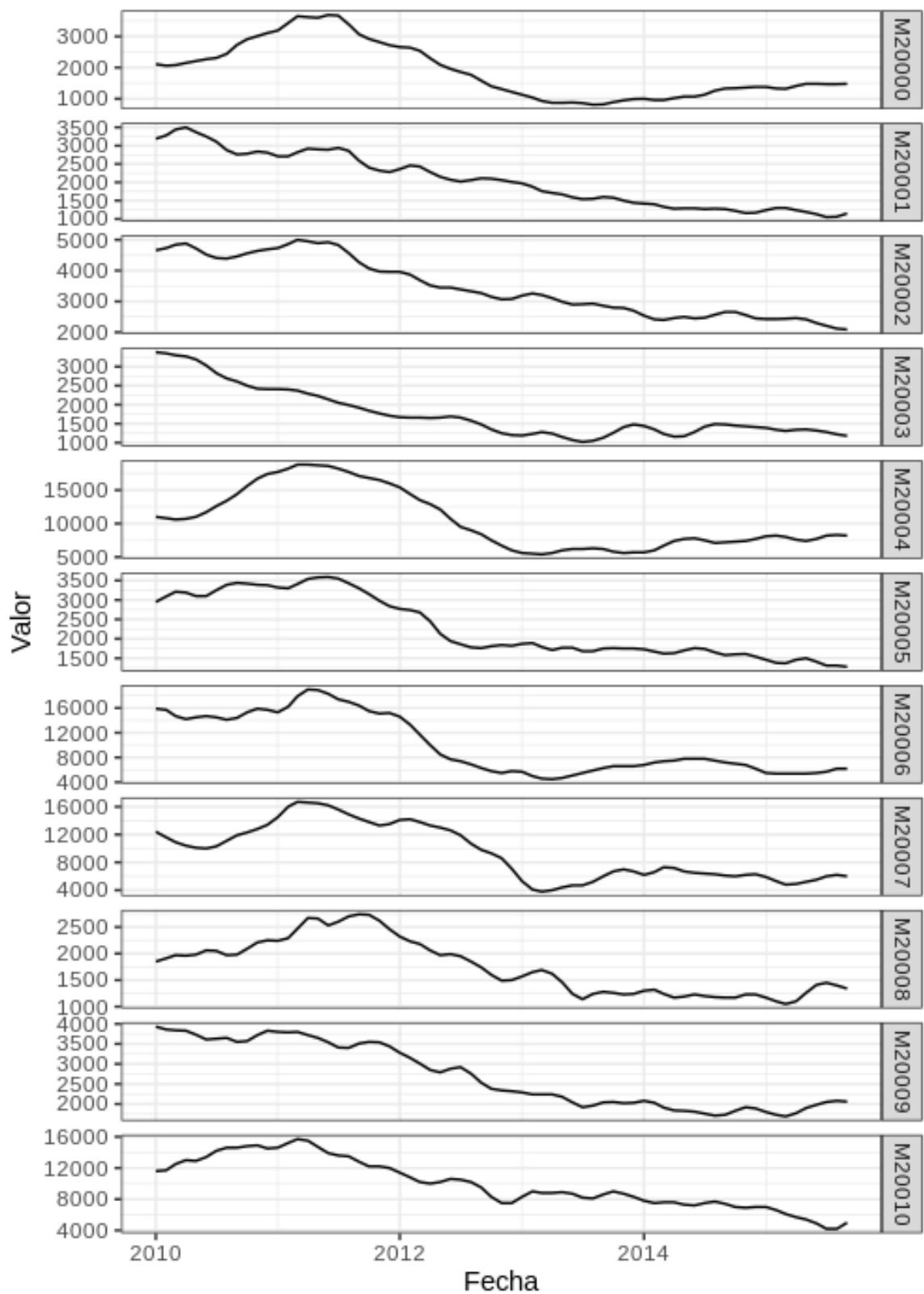
**Ilustración 11 Algunas de las Series pertenecientes a la M4 Competition**



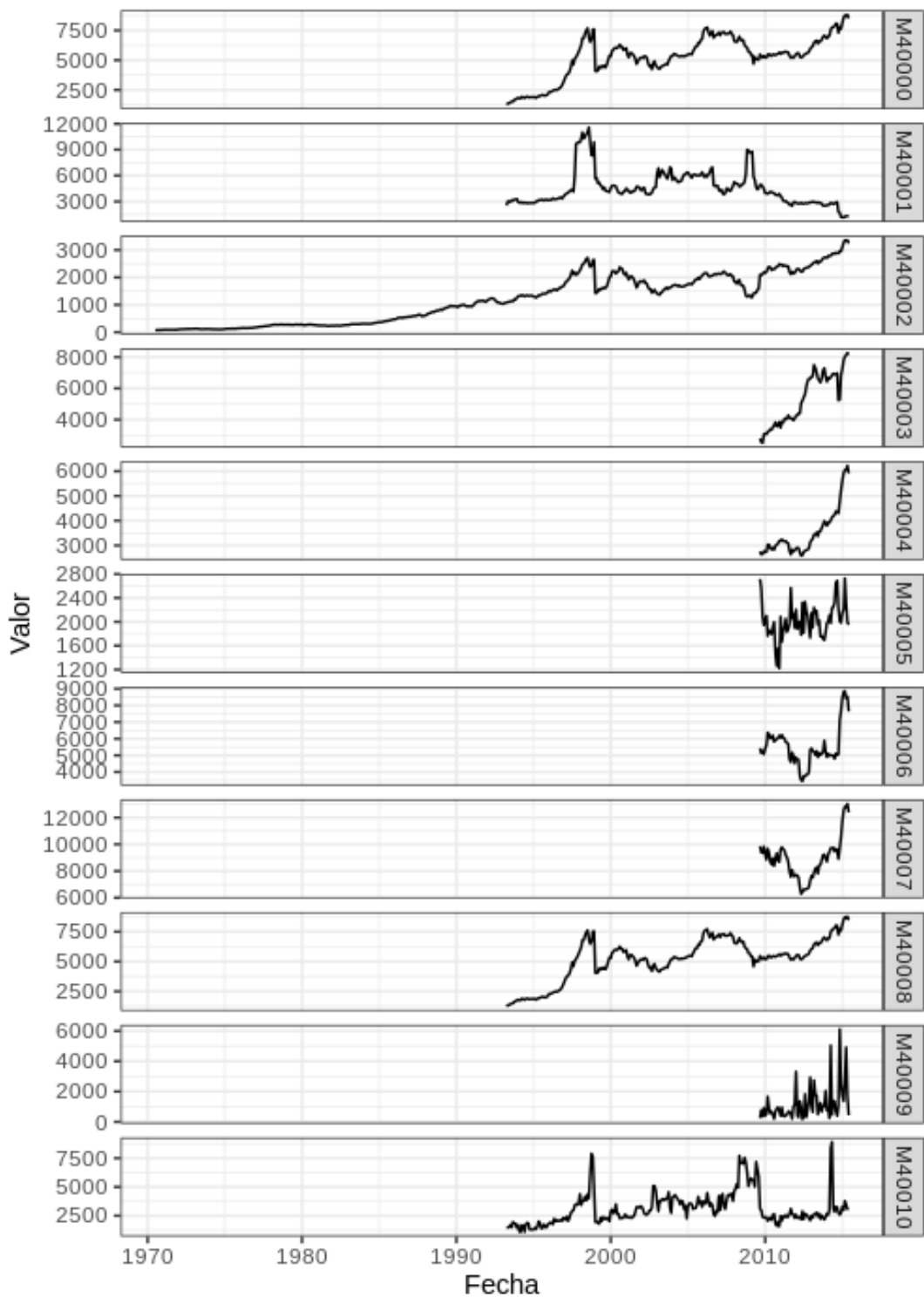
*Fuente: Elaboración propia*



Fuente: Elaboración propia



Fuente: Elaboración propia



Fuente: Elaboración propia

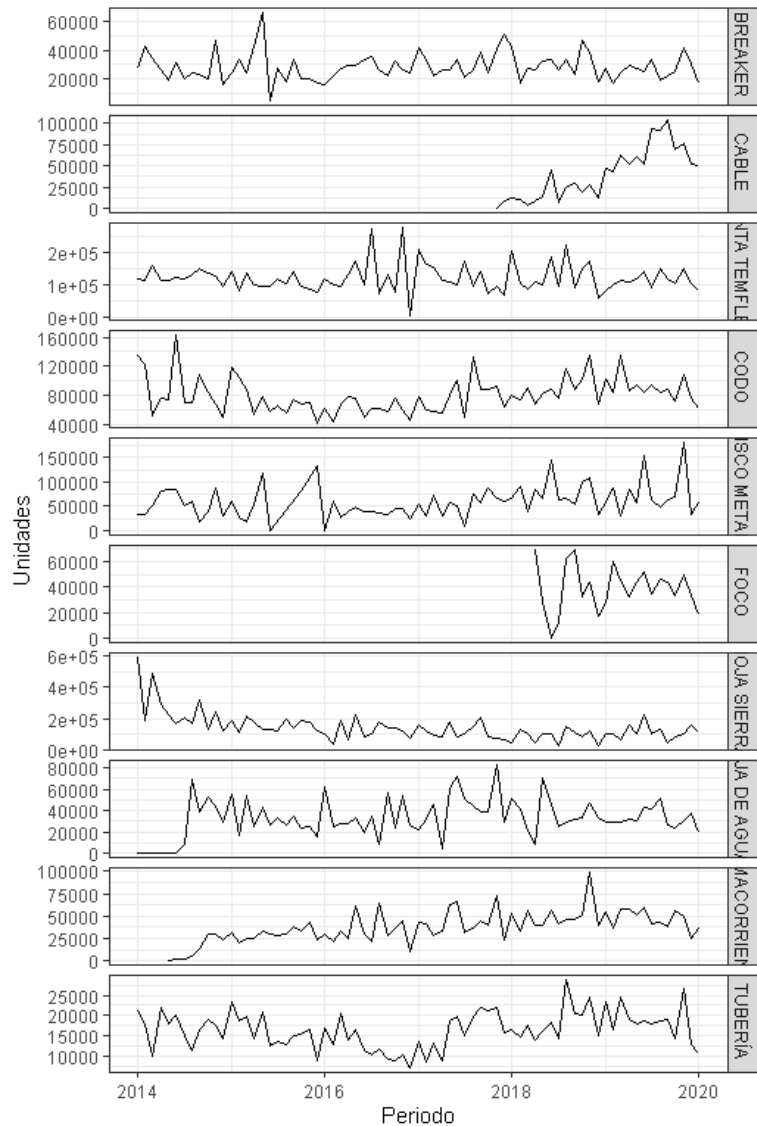
Se puede apreciar que las series tienen un comportamiento bastante diverso, lo que beneficia esta búsqueda del modelo que mejor se comporta en todo tipo de escenarios.

### 3.3. Descriptivas Caso de estudio

#### 3.3.1. Caso de estudio: Artículos

La empresa analizada es una importadora y comercializadora de materiales de ferretería, cuenta con clientes mayoristas y tiendas de *retail* a nivel nacional. De esta empresa se han seleccionado 10 *sku* importantes y variados, entre ellos tenemos productos de plomería, electricidad, herramientas manuales, herramientas eléctricas e iluminación.

**Ilustración 12** Venta en unidades, 10 productos escogidos

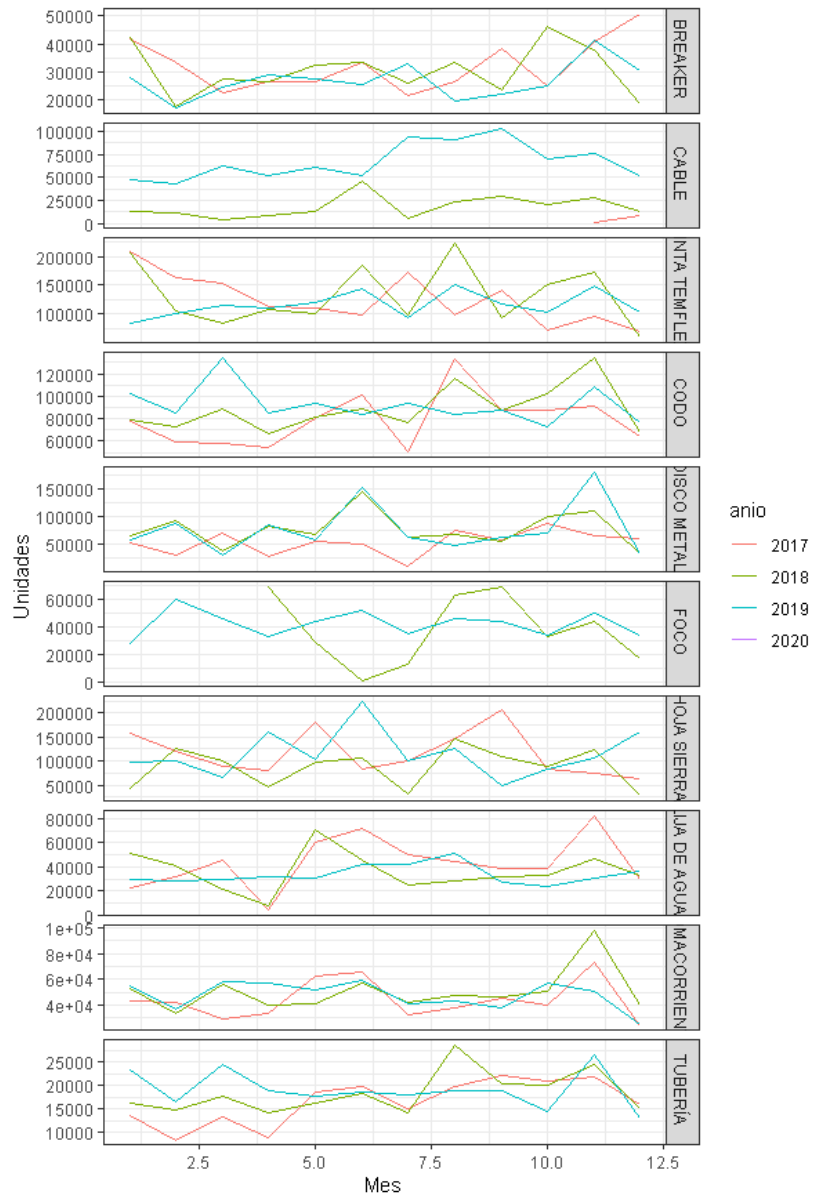


Fuente: Elaboración propia

En la ilustración 12, se puede visualizar que las series (transformadas por cuestiones de privacidad) son bastante diferentes, la empresa a estudiar cuenta con productos que tienen menos años de historia dado que fueron creados recientemente.

Para finalizar, se explora la estacionalidad de los productos analizados, para esto se realiza la ilustración 13 que permite evaluar la forma de la serie separándola por año.

**Ilustración 13 Gráfico de estacionalidad**



Fuente: Elaboración propia

# CAPÍTULO 4

## 4. RESULTADOS

Para el proceso de modelamiento cronológicamente se ha seguido los siguientes pasos:

- Obtención de una muestra aleatoria de tamaño 100 de entre las series Mensuales de la *M4 Competition*.
- Modelamiento y pronóstico con modelos de series de tiempo detallados en el capítulo 3.1.1 del presente documento.
- Afinamiento *tuning* de hiperparámetros para el modelo combinatorio explicado en la sección 3.2.2, cabe recordar que este modelo se alimenta del pronóstico dado en el punto anterior.
- Modelamiento y pronóstico con el modelo combinatorio detallado en el capítulo 3.1.1, esto usando la mejor combinación de hiperparámetros encontrada en el punto anterior.
- Comparación de métodos de pronóstico, tanto los explicados en la sección 3.1.1 como 3.2.1 y 3.2.2; de acuerdo a esta comparación se escogerá el mejor método y con este se procederá a pronosticar la venta mensual en unidades de los 10 productos de la empresa del sector ferretero estudiada.

El presente capítulo está dividido en 3 etapas, la primera etapa corresponde al tuneo de los hiperparámetros del modelo de red neuronal,

### 4.1. Afinamiento de hiperparámetros *Tuning*

Antes de presentar los hiperparámetros evaluados, vamos a recordar la arquitectura propuesta para el presente proyecto



#### 4.1.1. Arquitectura de Red Neuronal

Para este proyecto se ha construido una arquitectura de red neuronal que tiene múltiples entradas y adicionalmente combina capas Convolucionales, LSTM y *Dense*, en detalle:

- a) Entradas
  - Input A: Serie de tiempo univariante
  - Input B: Serie de tiempo multivariante: Valor analizado en conjunto con los valores ajustados de los modelos de pronósticos ejecutados en el primer literal.
  - Capa Convolutiva con ventana de tamaño 3 que toma como entrada la serie el input A (*hiperparámetro 1: cantidad de filtros convolucionales*)
  - Capa Convolutiva con ventana de tamaño 6 que toma como entrada la serie el input A (*hiperparámetro 1: cantidad de filtros convolucionales*)
  - Capa Convolutiva con ventana de tamaño 12 que toma como entrada la serie el input A (*hiperparámetro 1: cantidad de filtros convolucionales*)
  - 3 Capas, la primera de “aplanamiento” *Flatten* conectadas a una capa completa *Dense* y conectadas a una capa de redimensionamiento *Reshape*; esto aplicado a cada layer convolutiva.
  - Capa concatenada *Concatenate* que toma el resultado de las tres capas convolucionales transformadas más el Input B (este input B impide que esta red se construya de forma secuencial, este *input* entraría como *residual connection* o *skip connection*)
  - Capa Long Short Time Memory LSTM multivariante la cual recibe la capa concatenada del punto anterior (*hiperparámetro 2: número de unidades lstm*)
  - Capa *Dense* que estará encargada de ser el primer filtro para interpretar la salida de la capa LSTM.
  - Capa *Dense* que recibe la salida de la capa *Dense* anterior y es la que nos proporciona el output del modelo.

Finalmente, el tercer y último hiperparámetro a afinar será la cantidad de epochs.

Al programar esta arquitectura en keras, se obtiene:

### Ilustración 14 Arquitectura en keras

```

Model
Model: "model"

```

Layer (type)	Output Shape	Param #	Connected to
TimeSeries (InputLayer)	[(None, 48, 1)]	0	
conv1d (Conv1D)	(None, 48, 32)	128	TimeSeries[0][0]
conv1d_1 (Conv1D)	(None, 48, 32)	224	TimeSeries[0][0]
conv1d_2 (Conv1D)	(None, 48, 32)	416	TimeSeries[0][0]
flatten (Flatten)	(None, 1536)	0	conv1d[0][0]
flatten_1 (Flatten)	(None, 1536)	0	conv1d_1[0][0]
flatten_2 (Flatten)	(None, 1536)	0	conv1d_2[0][0]
dense (Dense)	(None, 48)	73776	flatten[0][0]
dense_1 (Dense)	(None, 48)	73776	flatten_1[0][0]
dense_2 (Dense)	(None, 48)	73776	flatten_2[0][0]
reshape (Reshape)	(None, 48, 1)	0	dense[0][0]
reshape_1 (Reshape)	(None, 48, 1)	0	dense_1[0][0]
reshape_2 (Reshape)	(None, 48, 1)	0	dense_2[0][0]
TimeSeriesMulti (InputLayer)	[(None, 48, 8)]	0	
concatenate (Concatenate)	(None, 48, 11)	0	reshape[0][0] reshape_1[0][0] reshape_2[0][0] TimeSeriesMulti[0][0]
lstm (LSTM)	(None, 70)	22960	concatenate[0][0]
dense_3 (Dense)	(None, 60)	4260	lstm[0][0]
forecast (Dense)	(None, 18)	1098	dense_3[0][0]

```

Total params: 250,414
Trainable params: 250,414
Non-trainable params: 0

```

Fuente: Elaboración propia

#### 4.1.2. Detalle del proceso de Afinamiento

El procedimiento de tuneo sigue una búsqueda con cuadrícula que sigue los siguientes pasos:

1. Modelo con 32 filtros convolucionales, 100 unidades LSTM y 40 epochs
2. Modelo con 64 filtros convolucionales, 100 unidades LSTM y 40 epochs
3. Modelo con 32 filtros convolucionales, 70 unidades LSTM y 40 epochs
4. Escoger para cada serie la mejor combinación de los hiperparámetros anteriores, pero aumentar los epochs a 50.

Las 4 combinaciones de hiperparámetros va a ser probada en cada serie; para evaluar la combinación, se sigue una estrategia de remuestreo de tipo ventana rodante de tal manera que se tiene 5 cortes o remuestras, cabe mencionar que en el entrenamiento se puede detener antes de llegar a los 40 o 50 epochs según el criterio de parada definido en “si no mejora luego de 5 epochs”.

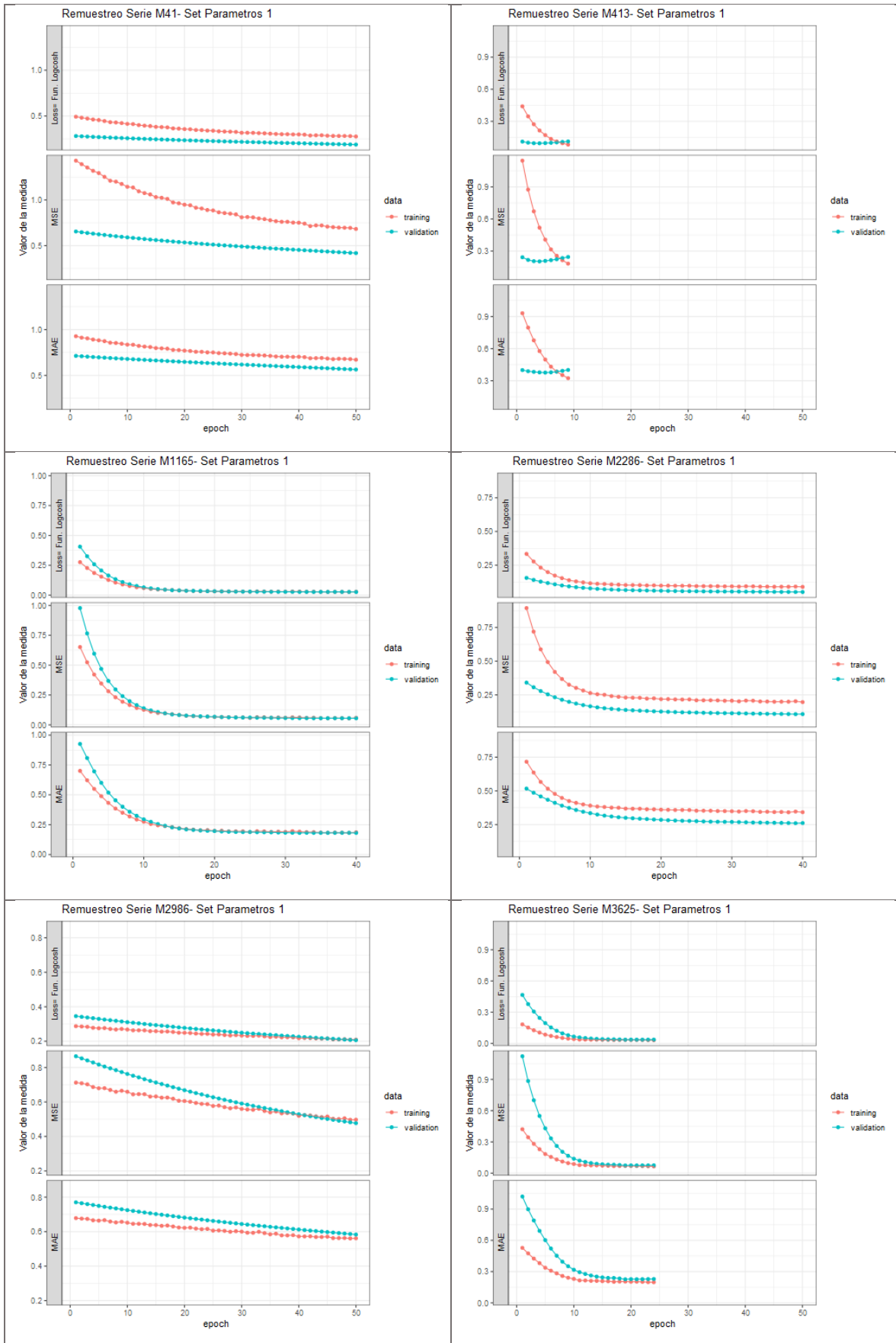
Cada remuestra consta de una porción de la serie de tiempo dividida a su vez en dos partes, *Train* para ajustar el modelo y otra *Test* para evaluarlo, de esta forma se califica el modelo con data que “no conoce”; para esta medición se usa la Raiz de los errores cuadráticos RMSE, es decir, la combinación tendría 5 modelos entrenados. Cada modelo tendría su RMSE, luego para tener la evaluación final de la combinación se parte de estos 5 valores de RMSE y se obtiene la media aritmética y con ellos el RMSE promedio de la combinación.

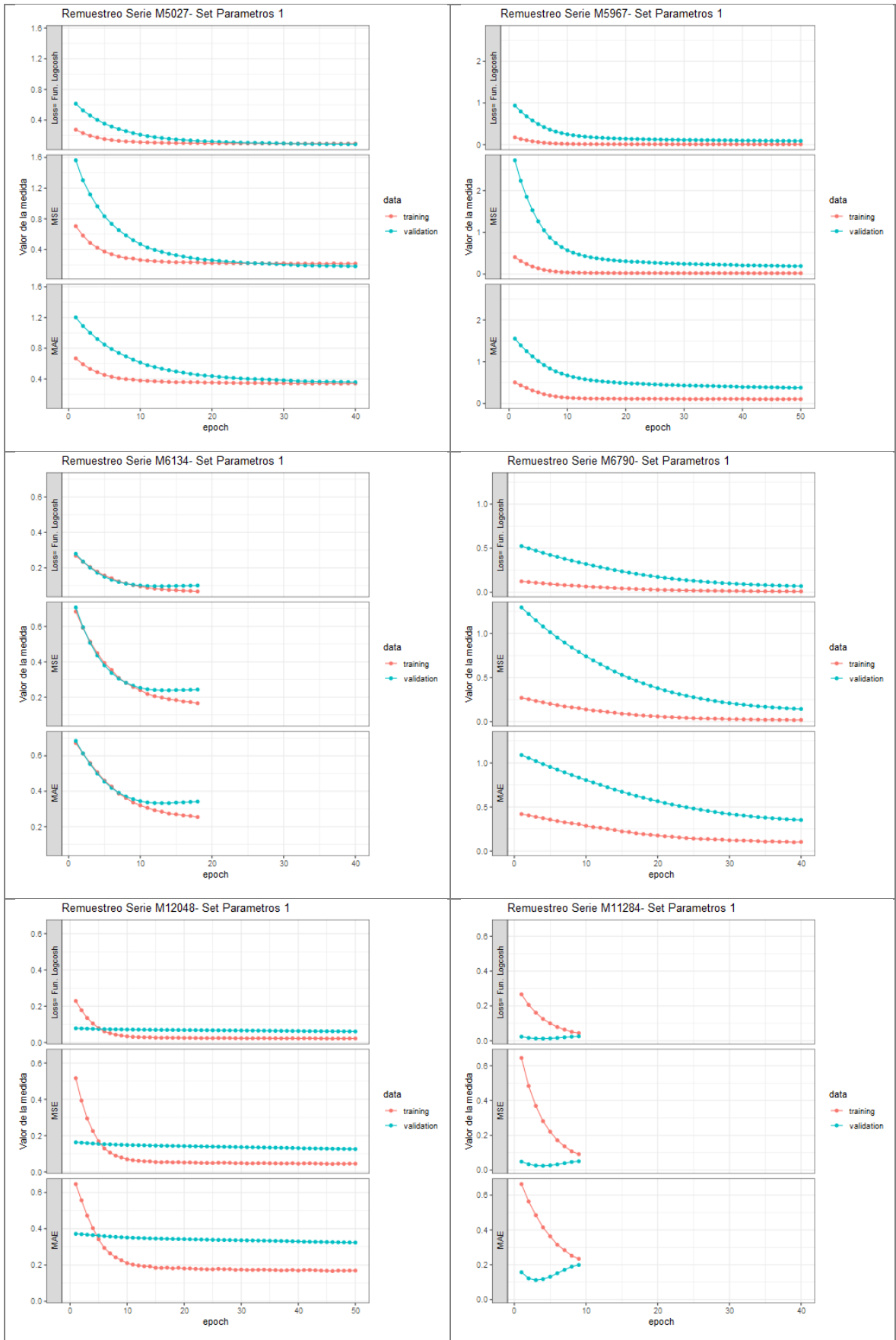
Al final, en la etapa de entrenamiento tendremos: 100 series x 4 Combinaciones x 5 Remuestras = 2000 modelos ajustados.

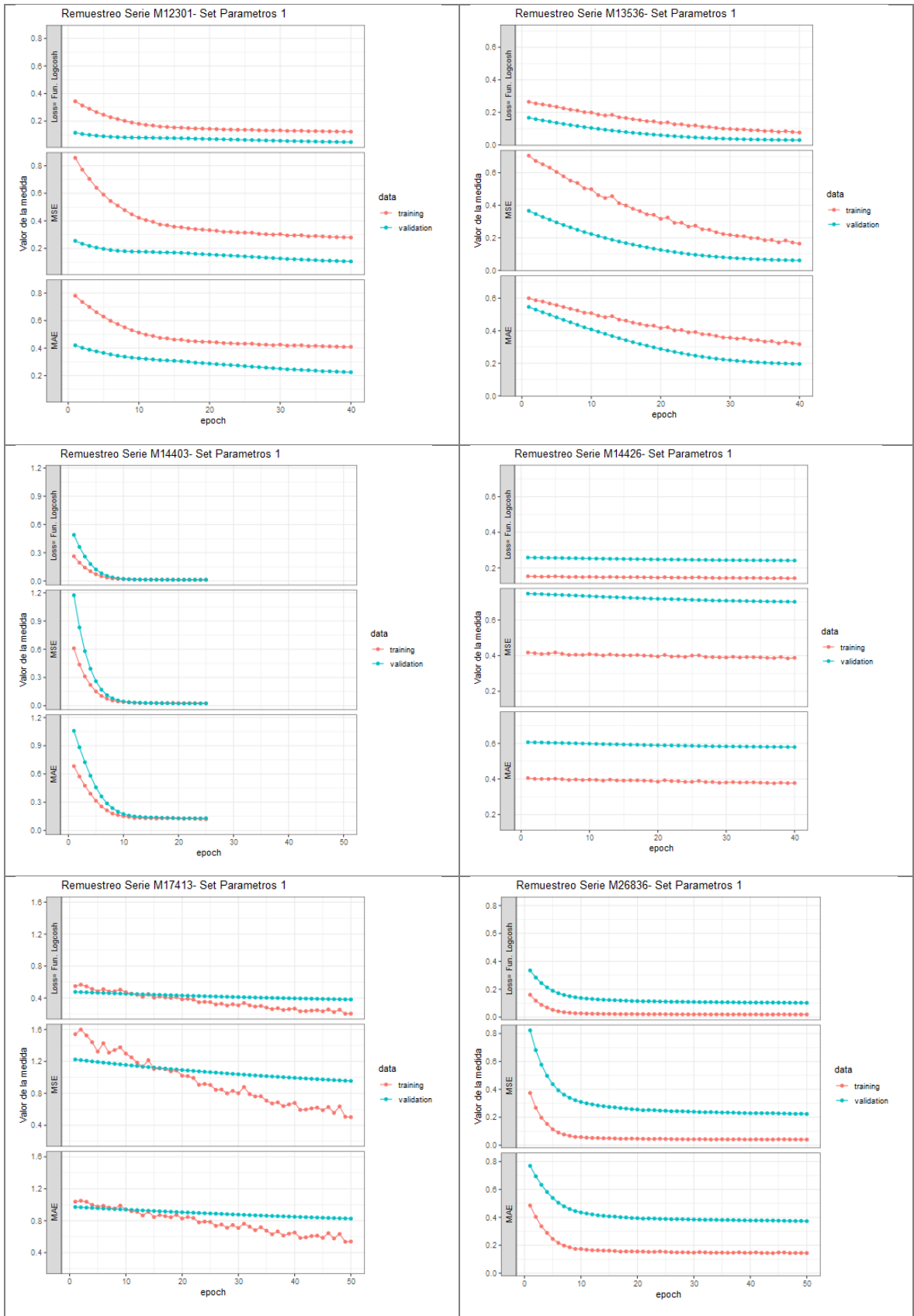
#### 4.1.3. *Tuning*: Algunos resultados

Durante el entrenamiento del modelo en cada remuestreo se tiene comportamientos diversos, para iniciar, se va a mostrar algunos gráficos que permitan entender este proceso.

## Ilustración 15 Entrenamiento en etapa de ajuste de hiperparámetros







Fuente: Elaboración propia

Los subgráficos mostrados representan, en orden de arriba hacia abajo: el valor de la función de pérdida LogCosh, RMSE y MAE calculados, tanto para el conjunto de datos de entrenamiento, como el de validación para una remuestra correspondiente a la evaluación de una combinación de parámetros para la serie analizada.

Durante esta etapa de modelamiento sobre las remuestras se puede apreciar el comportamiento bastante variado, convengamos para empezar que lo que se espera es lo siguiente:

- Valor en el set de validación que empieza mucho más elevado que el valor en el set de entrenamiento, luego a medida que se completan epochs ambos valores disminuyen (aceleradamente al inicio) sin embargo el valor en el set de validación decrece a mayor velocidad llegando a estar muy cerca del valor del set de entrenamiento.

Sin embargo, en los gráficos mostrados se puede ver:

- Si bien el valor en el conjunto de validación decrece, este queda aún lejos del valor del conjunto de entrenamiento, lo que puede significar que se debería probar con más epochs.
- Valor en el set de validación puede empezar debajo que el del valor en el set de entrenamiento y mantenerse así durante todo el entrenamiento
- No se ejecutan todas las epochs, esto debido a que se configuró un criterio de parada el cual definía que si luego de 5 epochs no había mejoría en el conjunto de validación se dejaba de entrenar.
- Hay ocasiones en que el decrecimiento no se presenta aceleradamente, incluso en el quincuagésimo subgráfico se puede ver que la pendiente del decrecimiento es casi inexistente.
- Valores en el set de validación y entrenamiento no tienen el mismo comportamiento, es decir, en algún momento el uno crece y el otro decrece, visualmente se ve un cruce entre ambos valores, esto puede indicar problemas de sobreajuste. También se ve un cruce cuándo el decrecimiento de un conjunto es mucho más acelerado que el del otro conjunto de datos.

A continuación, se presenta las combinaciones que genera los mejores resultados de acuerdo a nuestras opciones evaluadas:

**Tabla 2 Cantidad series según la combinación que brinda mejor resultado**

Número Filtros Convolucionales	Cantidad de unidades LSTM	Número de epochs	Frecuencia (cantidad de series)
64	100	50	34
32	100	40	26
32	100	50	23
64	100	40	12
32	70	40	4
32	70	50	1

*Fuente: Elaboración propia*

Nótese como primer punto a resaltar que a pesar de tener una estrategia de 4 pasos para el afinamiento de hiperparámetros, al final tenemos 6 combinaciones distintas en nuestra tabla.

El 34% de las series presentaron un menor promedio del RMSE utilizando 64 filtros convolucionales, 100 unidades LSTM y definiendo 50 épocas epochs, la segunda mejor combinación corresponde a 32 filtros convolucionales, 100 unidades LSTM y 40 epochs, esta se presenta el 26% de series analizadas.

Se puede ver que la cantidad de unidades LSTM parece ser un hiperparámetro importante, en el 95% de las series tienen en su mejor combinación este parámetro definido como igual a 100, sólo un 5% tiene este hiperparámetro con valor de 70.

Adicional, se puede pensar que sí es importante subir el número de epochs, en las tres mejores combinaciones dos involucran 50 epochs, sumando 57% de las series evaluadas entre estas dos combinaciones.



## 4.2. Resultado de pronósticos sobre las series analizadas

Luego de haber determinado la mejor combinación de hiperparámetros, se procede a modelar sobre la serie temporal completa para luego realizar la comparación de todos los métodos de pronóstico explicados en las secciones 3.1.1, 3.2.1 y 3.2.2, en detalle para lo que resta del capítulo los modelos se van a abreviar siguiendo la siguiente nomenclatura:

- **ar\_01**.- Un modelo ARIMA(1, d, 0)(P, D, Q), es decir se exige al modelo tener una componente Autoregresiva.
- **ar\_02**.- Un modelo ARIMA(2, d, 0)(P, D, Q), es decir se exige al modelo tener dos componentes autoregresivas.
- **ma\_01**.- Un modelo ARIMA(0, d, 1)(P, D, Q), es decir se exige al modelo tener una componente media móvil.
- **ma\_02**.- Un modelo ARIMA(0, d, 2)(P, D, Q), es decir se exige al modelo tener dos componentes media móvil.
- **autoarima**.- Un modelo ARIMA(p, d, q)(P, D, Q) ajustado automáticamente ajustado según el algoritmo Hyndman-Khandakar (Hyndman y Yeasmin, 2008).
- **sarima**.- Un modelo ARIMA(p, d, q)(P, D, Q) que se ha definido de tal manera que se le obligue tener una componente estacional, es decir (P, D, Q) tal que  $P + Q$  debe ser mayor a cero.
- **ets**.- Un modelo de estado-espacio ETS cuyas componentes de error, tendencia, estacionalidad ajustado en función del AIC y máxima verosimilitud.
- **Comb1**.- Combinación entre los modelos *ets*, *autoarima* y *sarima*, dándole a los 3 el mismo peso.
- **Comb2**.- Combinación aritmética entre los modelos *ets*, *ar\_01*, *ar\_02*, *ma\_01* y *ma\_02*.
- **LSTMComb**.- Combinación usando una arquitectura de red neuronal con múltiples entradas y con capas Convolucionales de una dimensión, capas LSTM y *Dense*.

A continuación, se muestra la cantidad de ocasiones en que cada modelo ha tenido el menor sMAPE para las series analizadas, se puede ver que el modelo ETS es el que en un 24% de ocasiones presentó el menor sMAPE, seguido por el modelo ar\_01 que fue el mejor en 16% de series y luego el autoarima y LSTMComb con 10% cada uno.

**Tabla 3 Cantidad de veces que el modelo es el que tiene menor error**

<b>Modelo</b>	<b>Número de Series</b>
ets	24
ar_01	16
autoarima	10
LSTMComb	10
ar_02	9
ma_01	8
sarima	8
Comb1	7
Comb2	4
ma_02	4

*Fuente: Elaboración propia*

Ahora se muestran las estadísticas de los modelos usados en conjunto con los 5 mejores modelos presentados en el M4 Competition, se analiza la Media absoluta del error porcentual simétrico sMAPE para todos los modelos en cada serie, de esto se obtienen el promedio del indicador, el promedio acotado cortando 5% de datos a cada extremo de la distribución y el mínimo y máximo.

Se puede visualizar que, de los modelos desarrollados, el modelo ETS es el que menor promedio presenta con un valor de 8.42, seguidos de Comb2 y Comb1 con 8.51 y 8.54 respectivamente; por el contrario, al analizar el promedio acotado se mantienen los 3 modelos como los mejores pero siendo el Comb2 el modelo que presenta mejores indicadores.

Por otro lado, se puede ver que para la muestra tomada sea que se analice el promedio o se analice el promedio acotado al menos uno de los modelos generados estarían entre el top 5 de los modelos ganadores del M4.

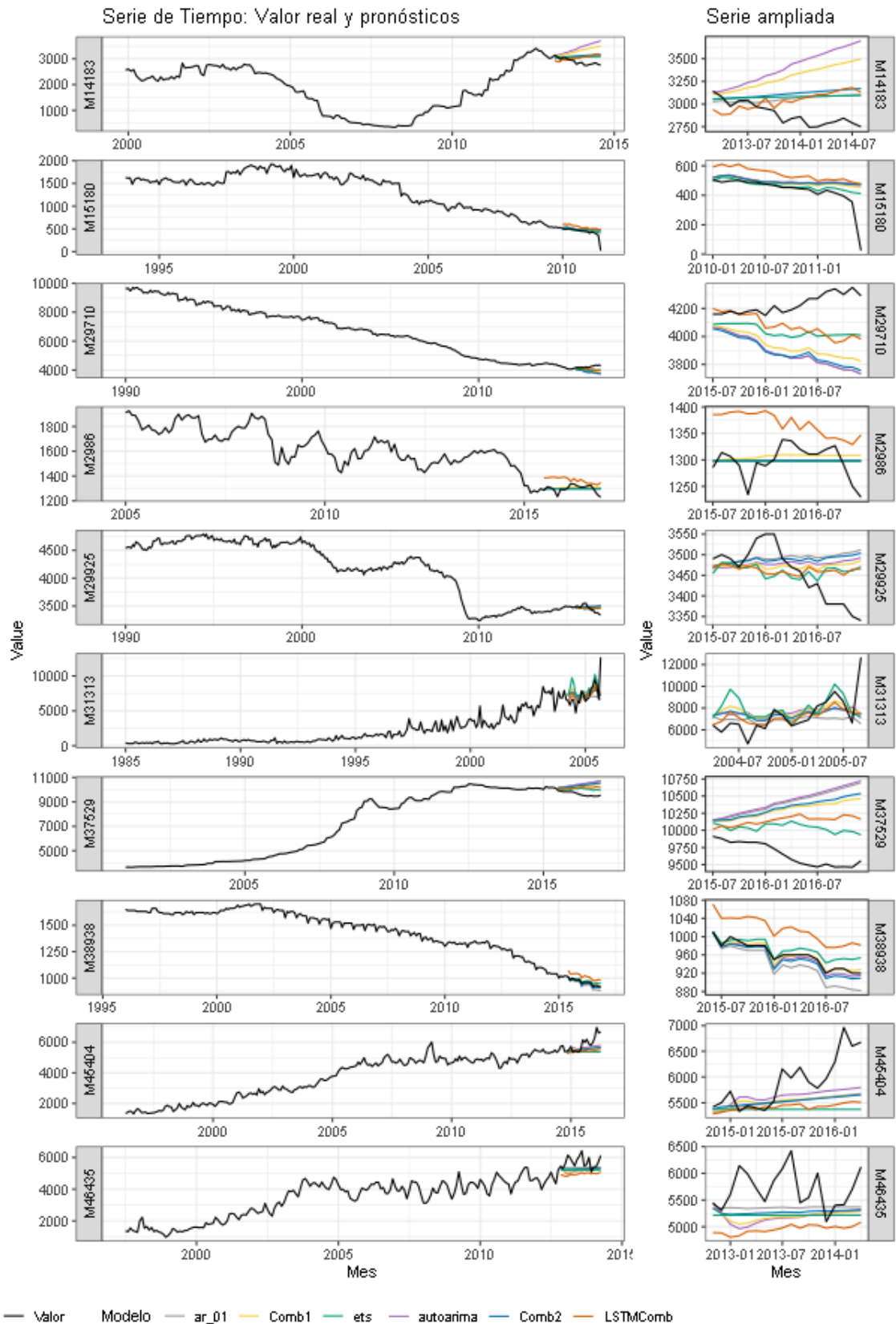
**Tabla 4 Top 5 modelos ganadores**

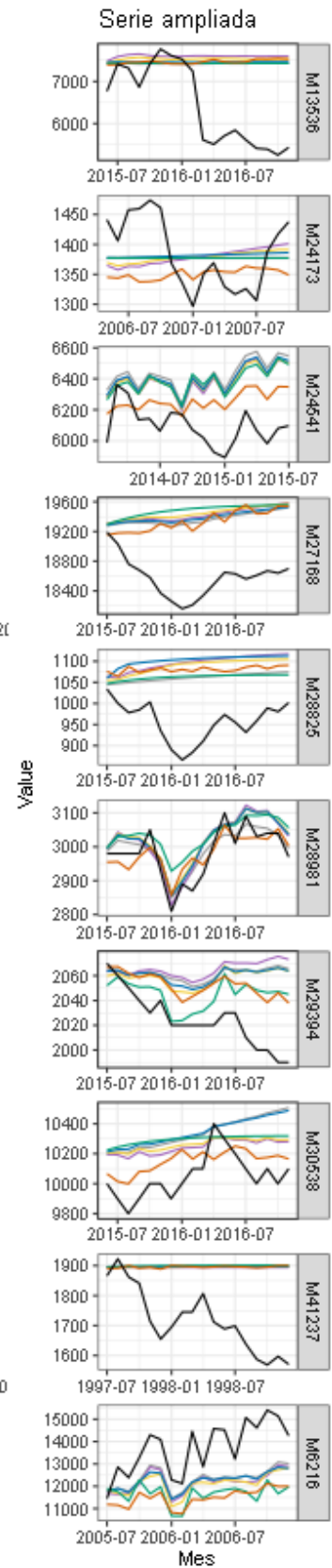
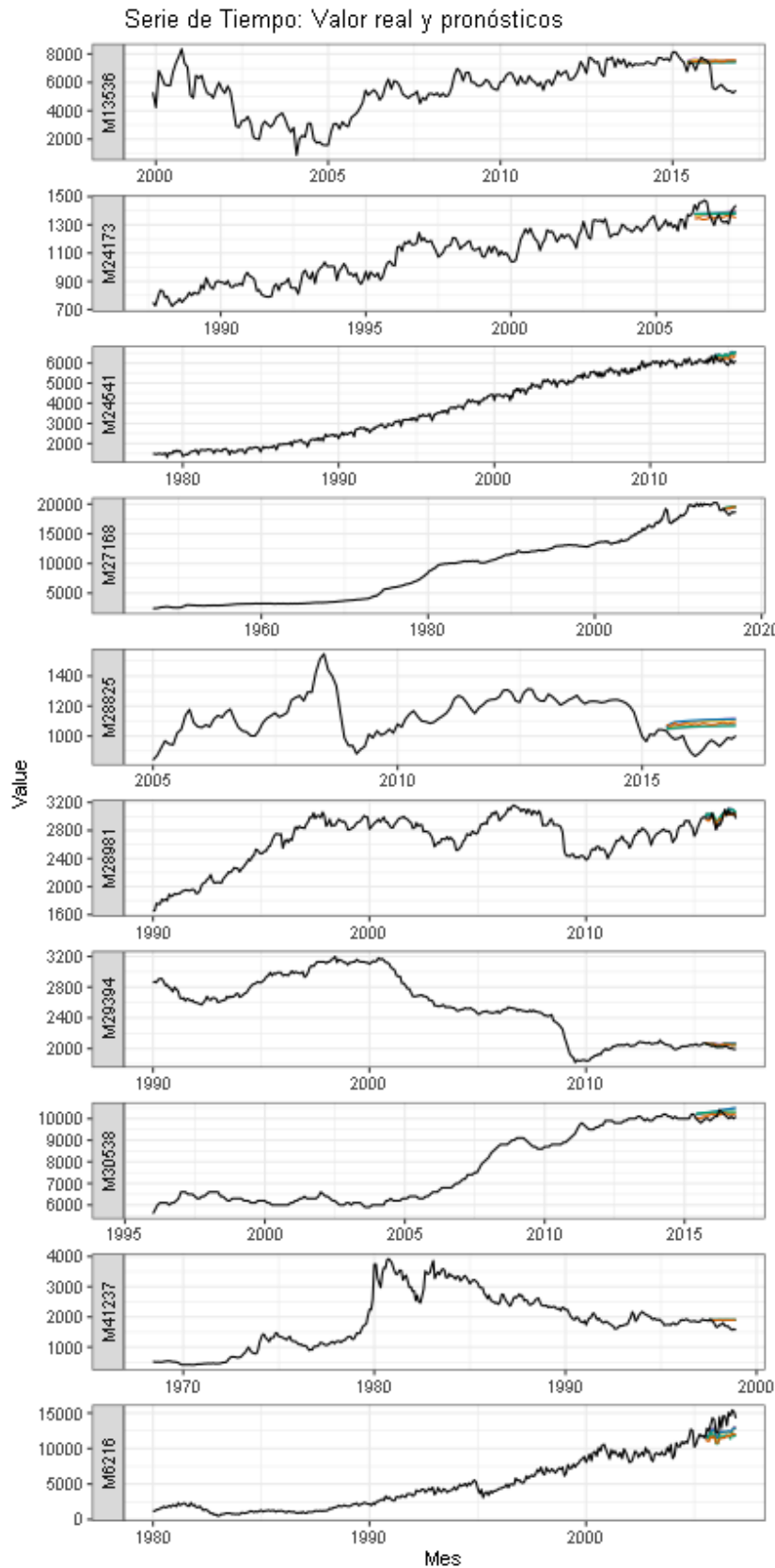
Modelo	Promedio	SMAPE		
		Prom Acotado	Mínimo	Máximo
ar_01	8.86	6.49	0.33	84.23
ar_02	8.77	6.40	0.34	81.88
ma_01	8.67	6.45	0.41	95.52
ma_02	8.94	6.61	0.34	95.10
autoarima	8.77	6.44	0.33	81.83
sarima	8.80	6.44	0.19	81.83
ets	8.42	6.38	0.39	57.65
Comb1	8.54	6.28	0.40	73.79
Comb2	8.51	6.20	0.41	83.55
LSTMComb	9.32	7.06	0.52	66.93
Best1	8.52	6.25	0.56	69.38
Best2	8.41	6.20	0.34	67.89
Best3	8.37	6.14	0.44	65.77
Best4	8.21	6.12	0.40	59.70
Best5	8.40	6.20	0.44	61.07

*Fuente: Elaboración propia*

En los siguientes gráficos se muestra cada una de las series que integran la muestra analizada tanto la porción utilizada para modelar como la utilizada para evaluar los modelos, esto junto con el pronóstico que cada modelo ha dado como resultado. Para mejor visualización de los resultados de cada modelo, se amplía la porción del gráfico que corresponde a los periodos pronosticados.

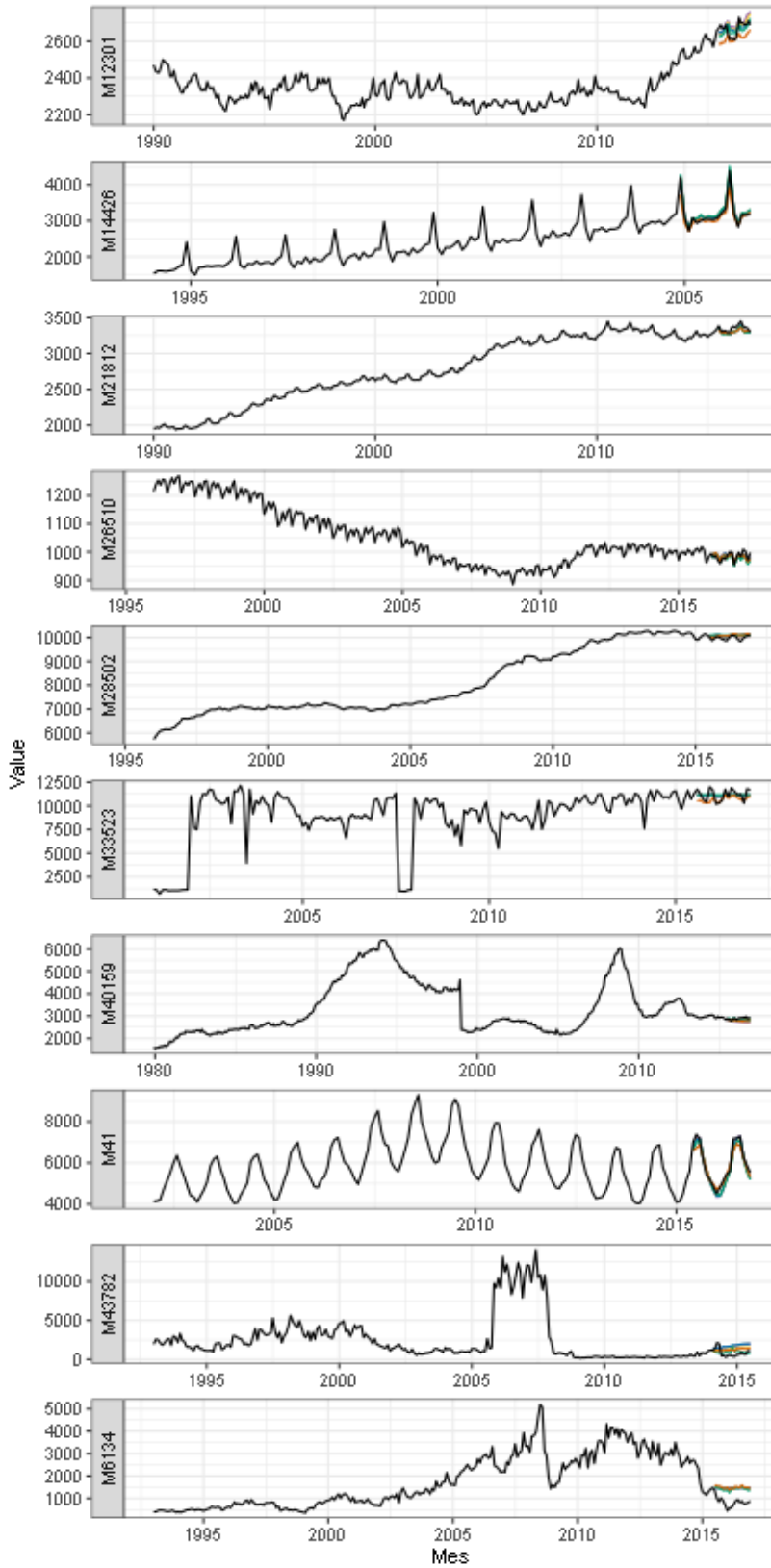
## Ilustración 16 Gráficos de series de tiempo y pronósticos generados



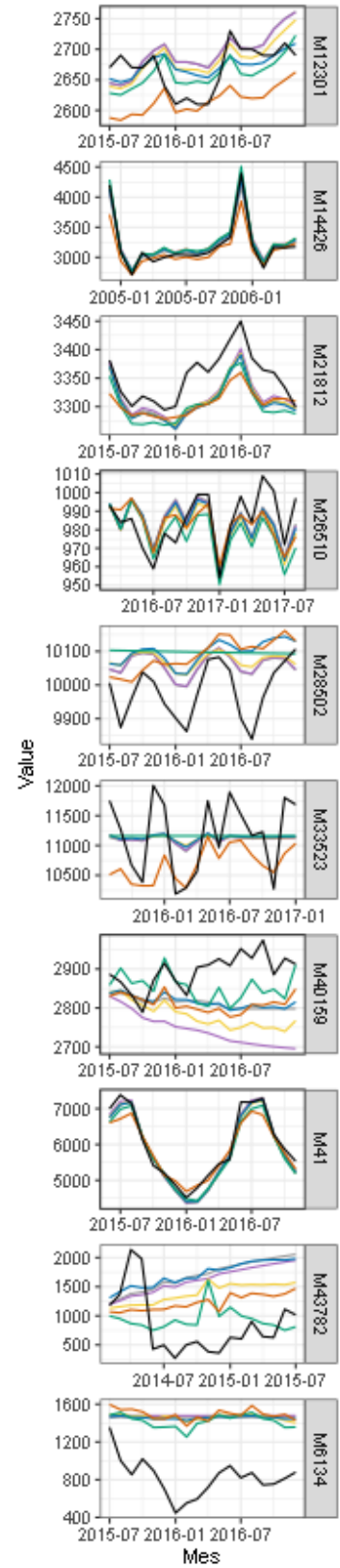


— Valor    Modelo    — ar\_01    — Comb1    — ets    — autoarima    — Comb2    — LSTMComb

Serie de Tiempo: Valor real y pronósticos

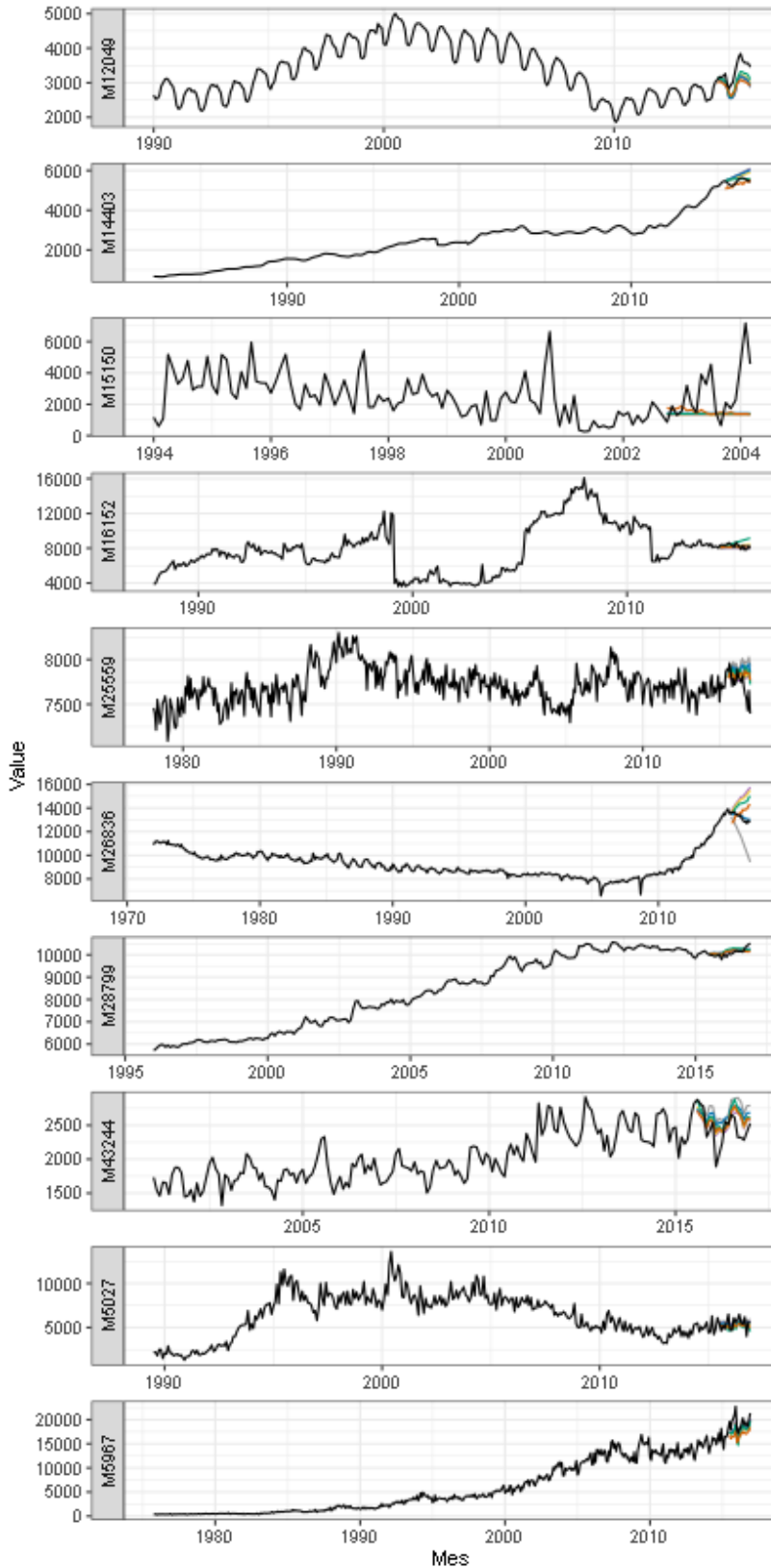


Serie ampliada

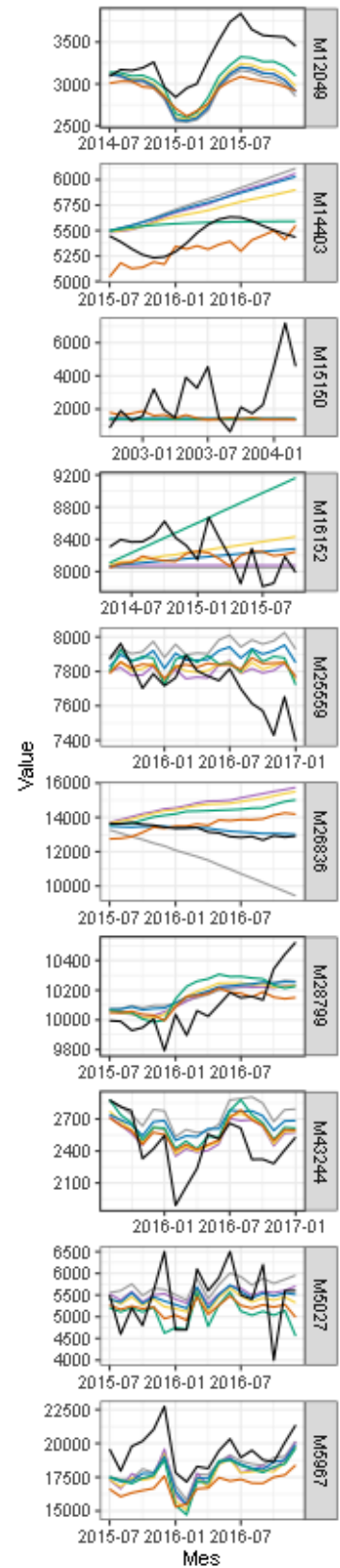


— Valor    Modelo    — ar\_01    — Comb1    — ets    — autoarima    — Comb2    — LSTMComb

Serie de Tiempo: Valor real y pronósticos

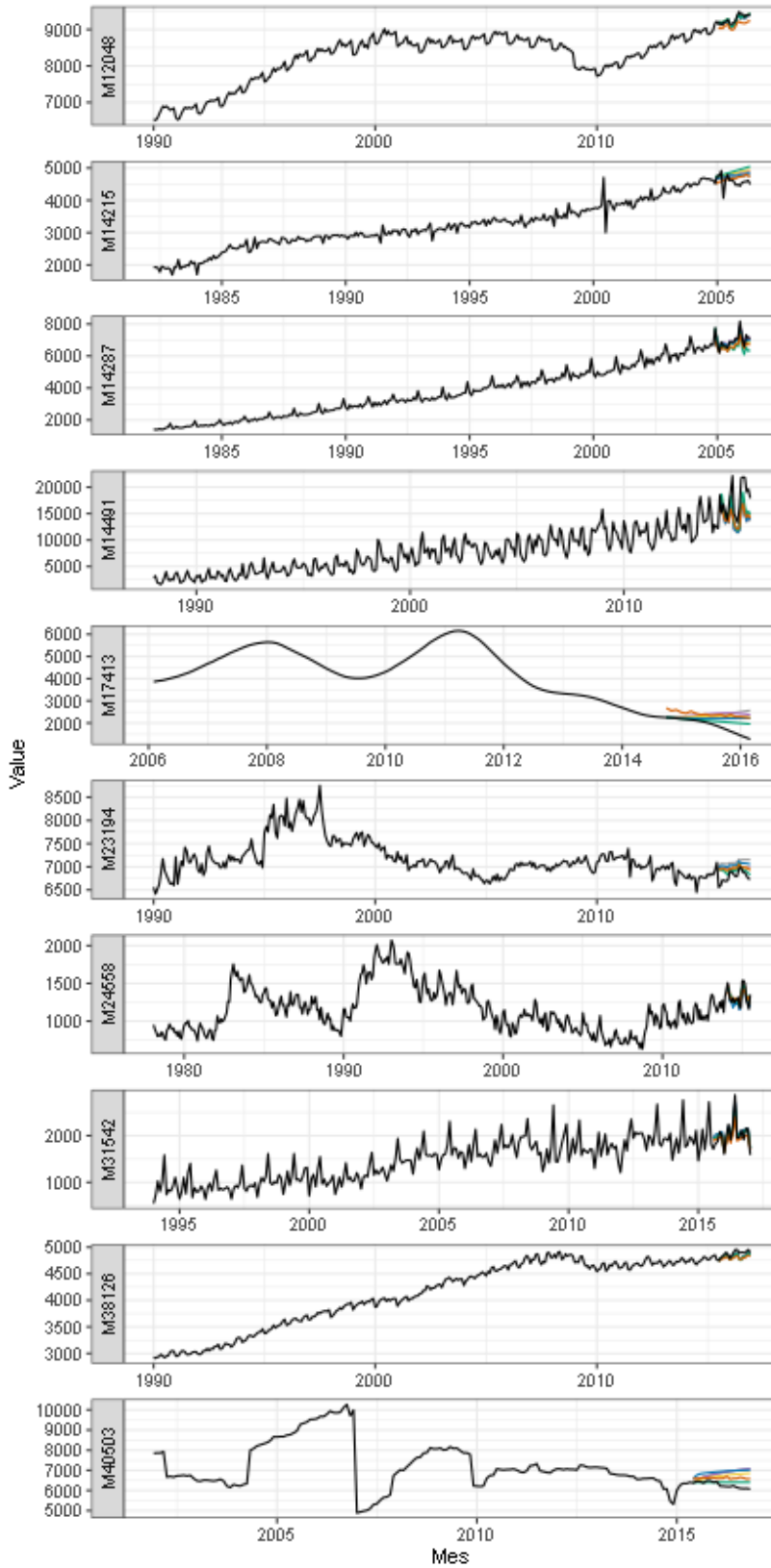


Serie ampliada

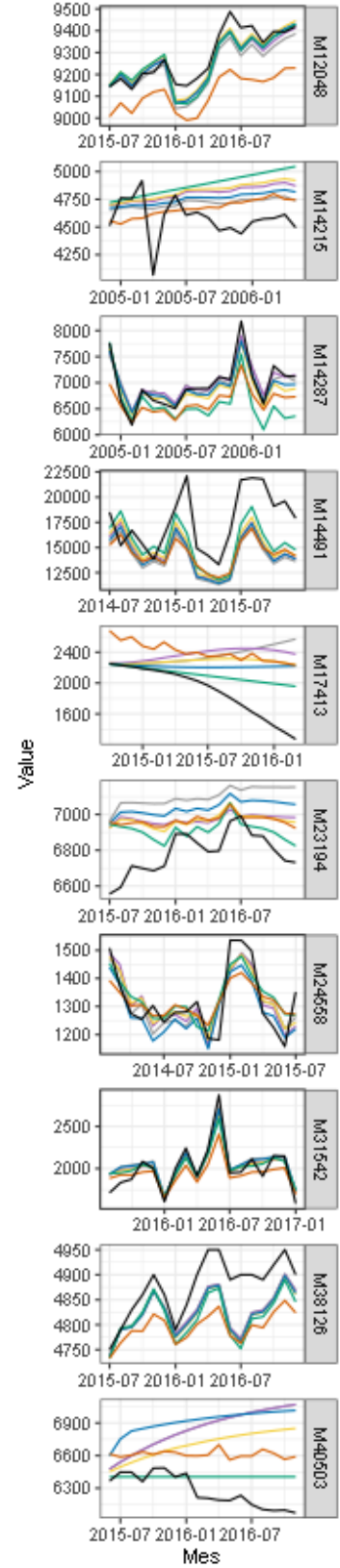


— Valor    Modelo    — ar\_01    — Comb1    — ets    — autoarima    — Comb2    — LSTMComb

Serie de Tiempo: Valor real y pronósticos



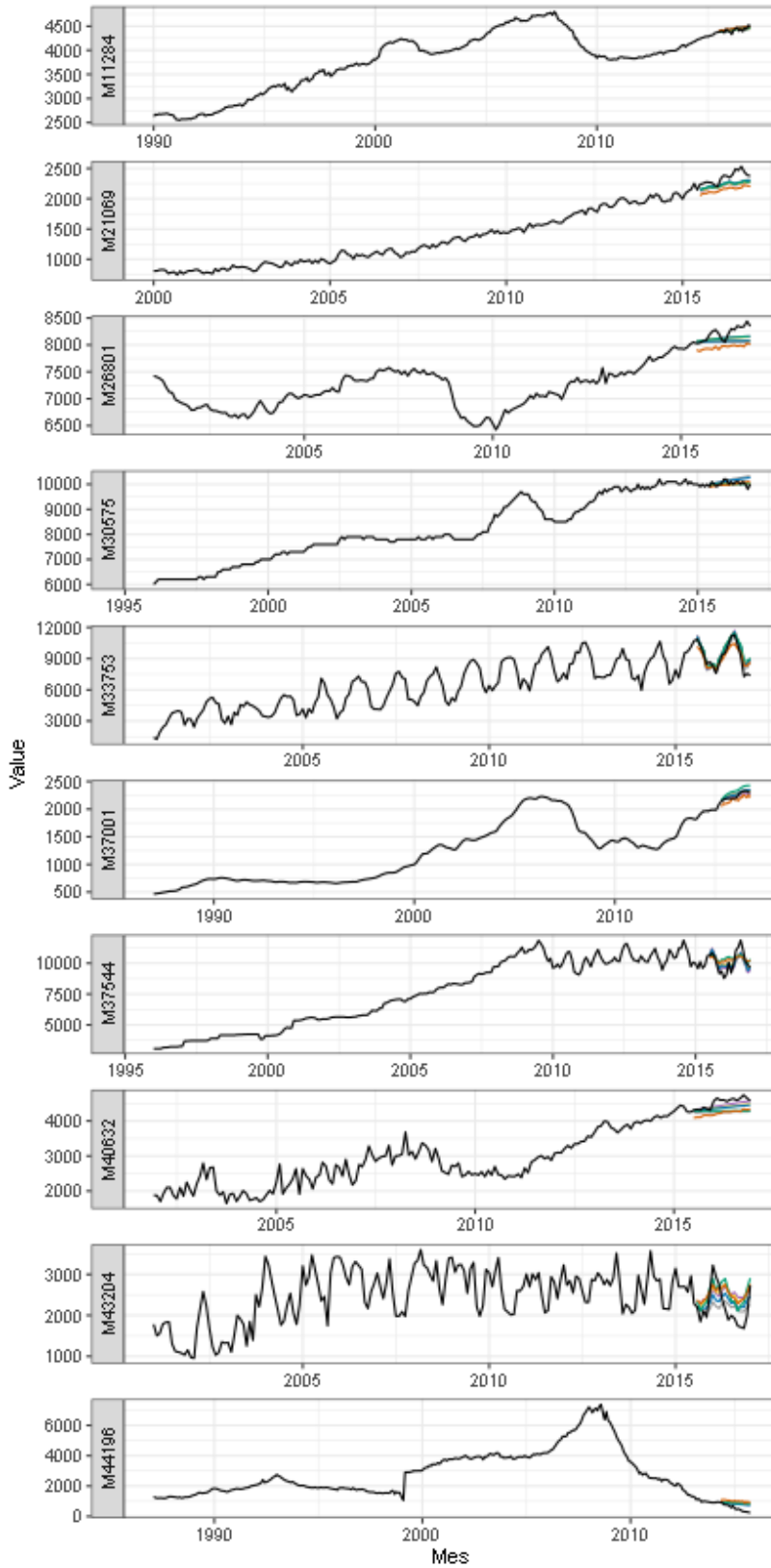
Serie ampliada



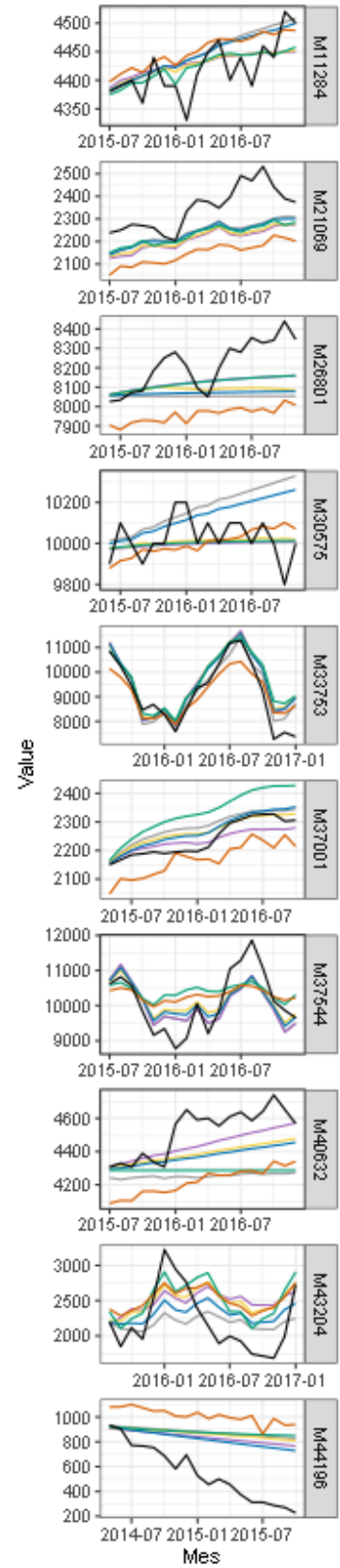
— Valor    Modelo    — ar\_01    — Comb1    — ets    — autoarima    — Comb2    — LSTMComb



Serie de Tiempo: Valor real y pronósticos

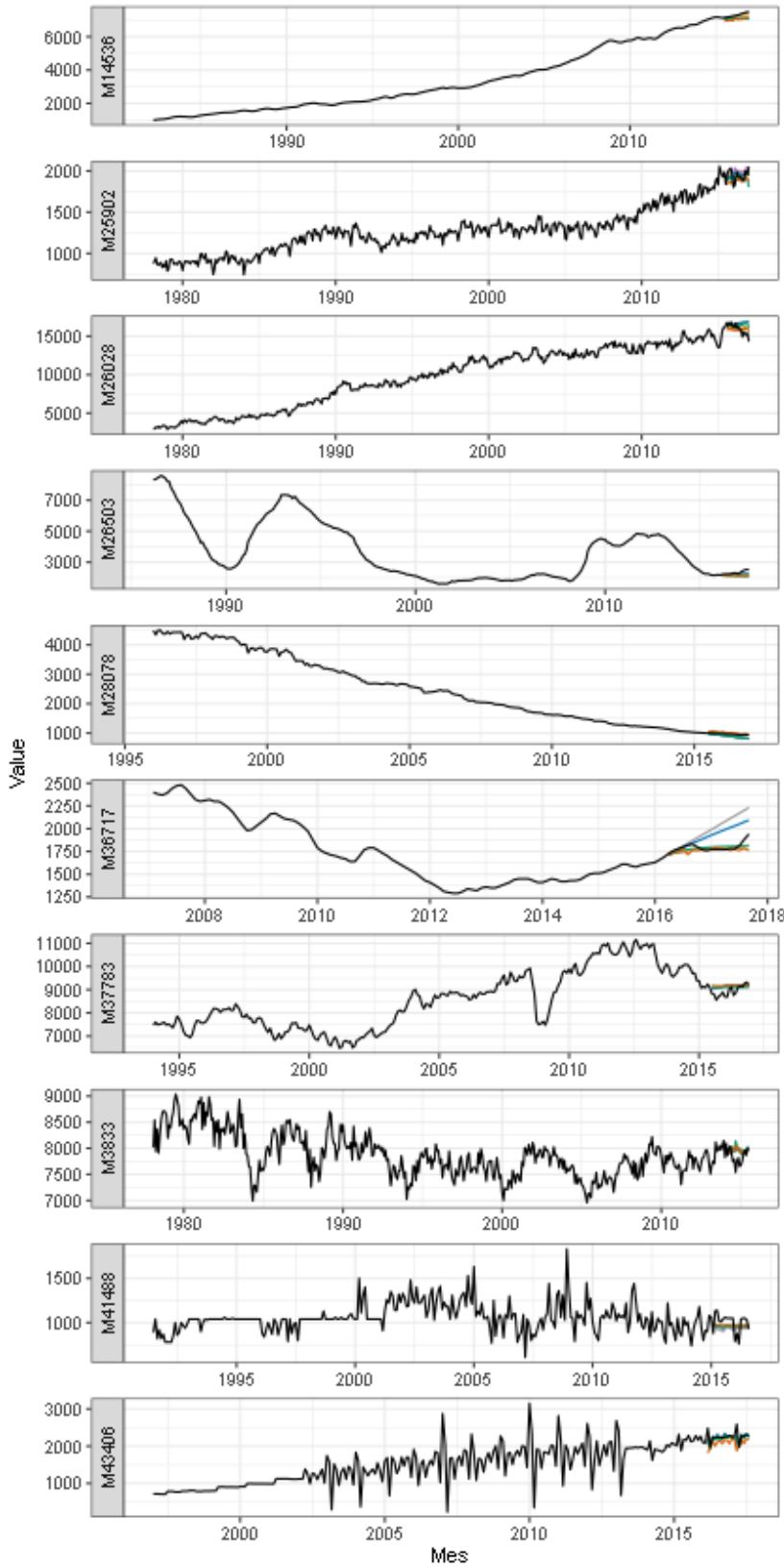


Serie ampliada

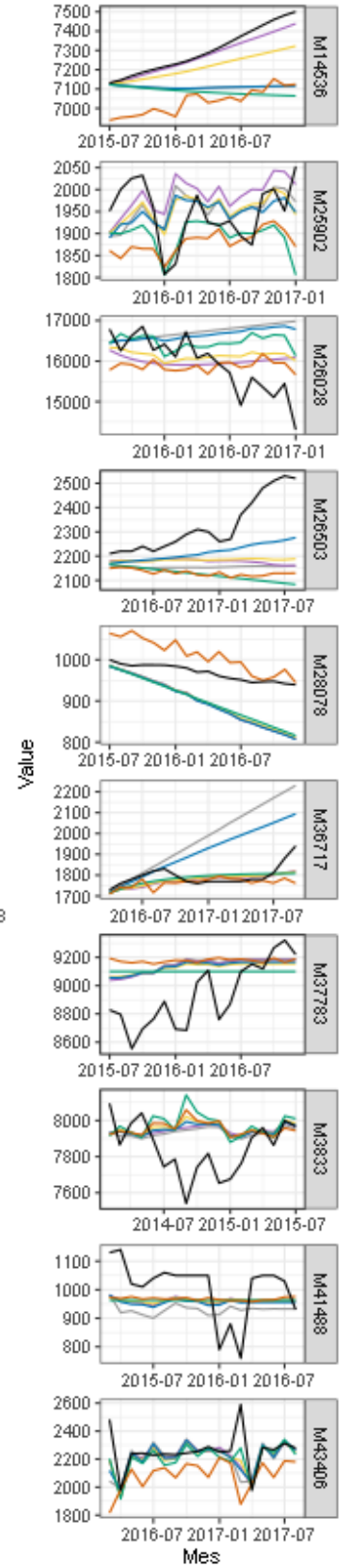


— Valor    Modelo    — ar\_01    — Comb1    — ets    — autoarima    — Comb2    — LSTMComb

Serie de Tiempo: Valor real y pronósticos

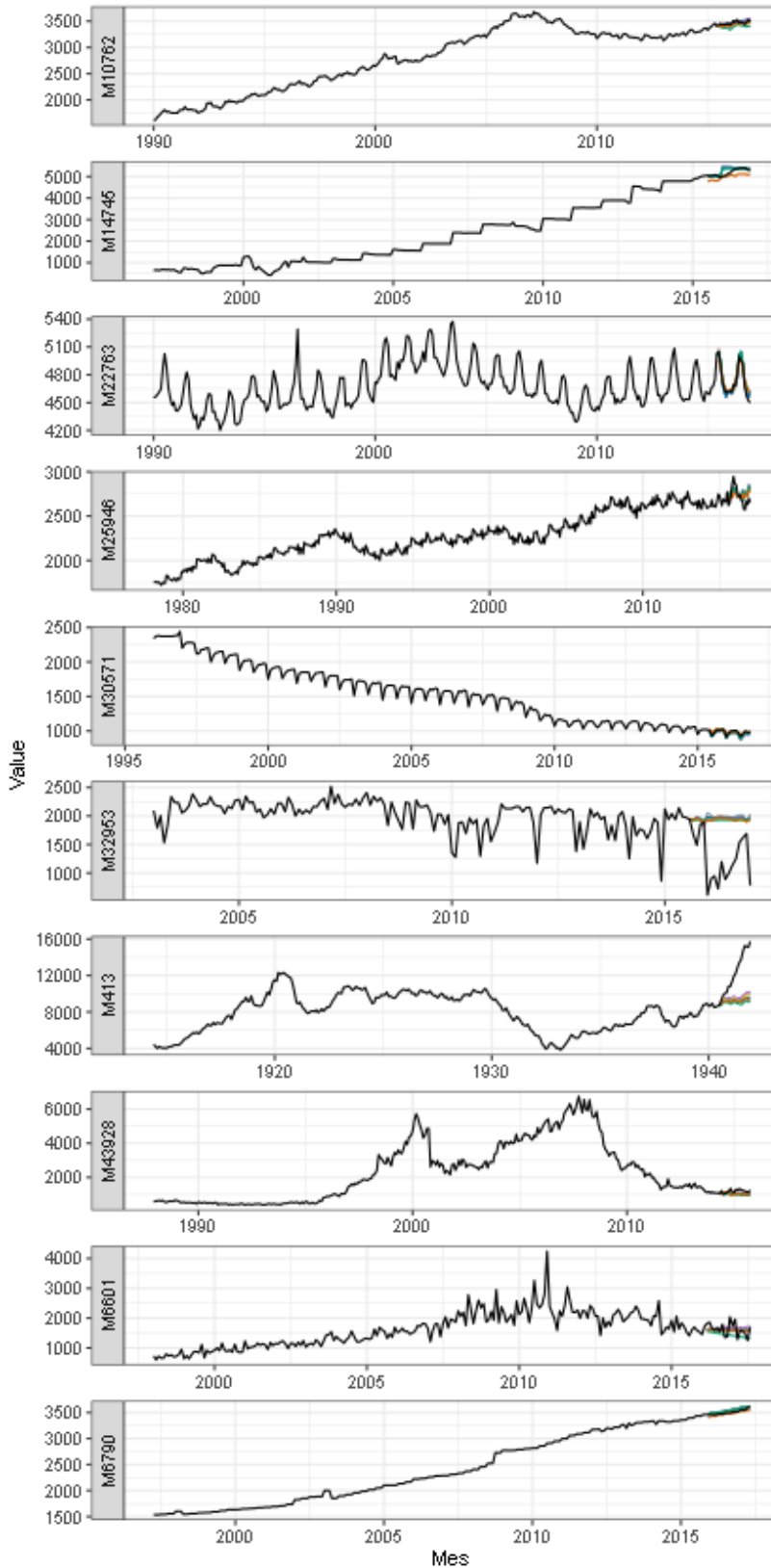


Serie ampliada

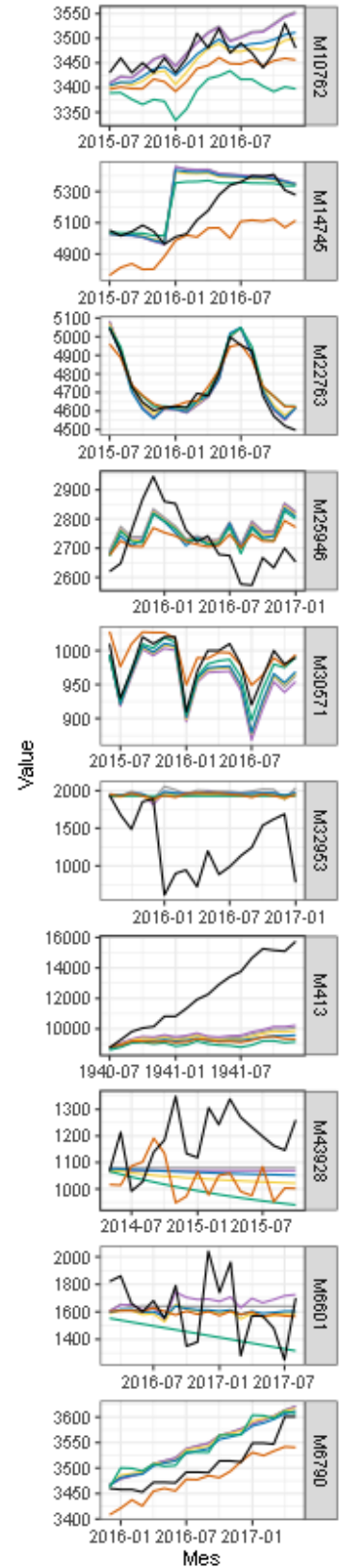


— Valor    Modelo    — ar\_01    — Comb1    — ets    — autoarima    — Comb2    — LSTMComb

Serie de Tiempo: Valor real y pronósticos

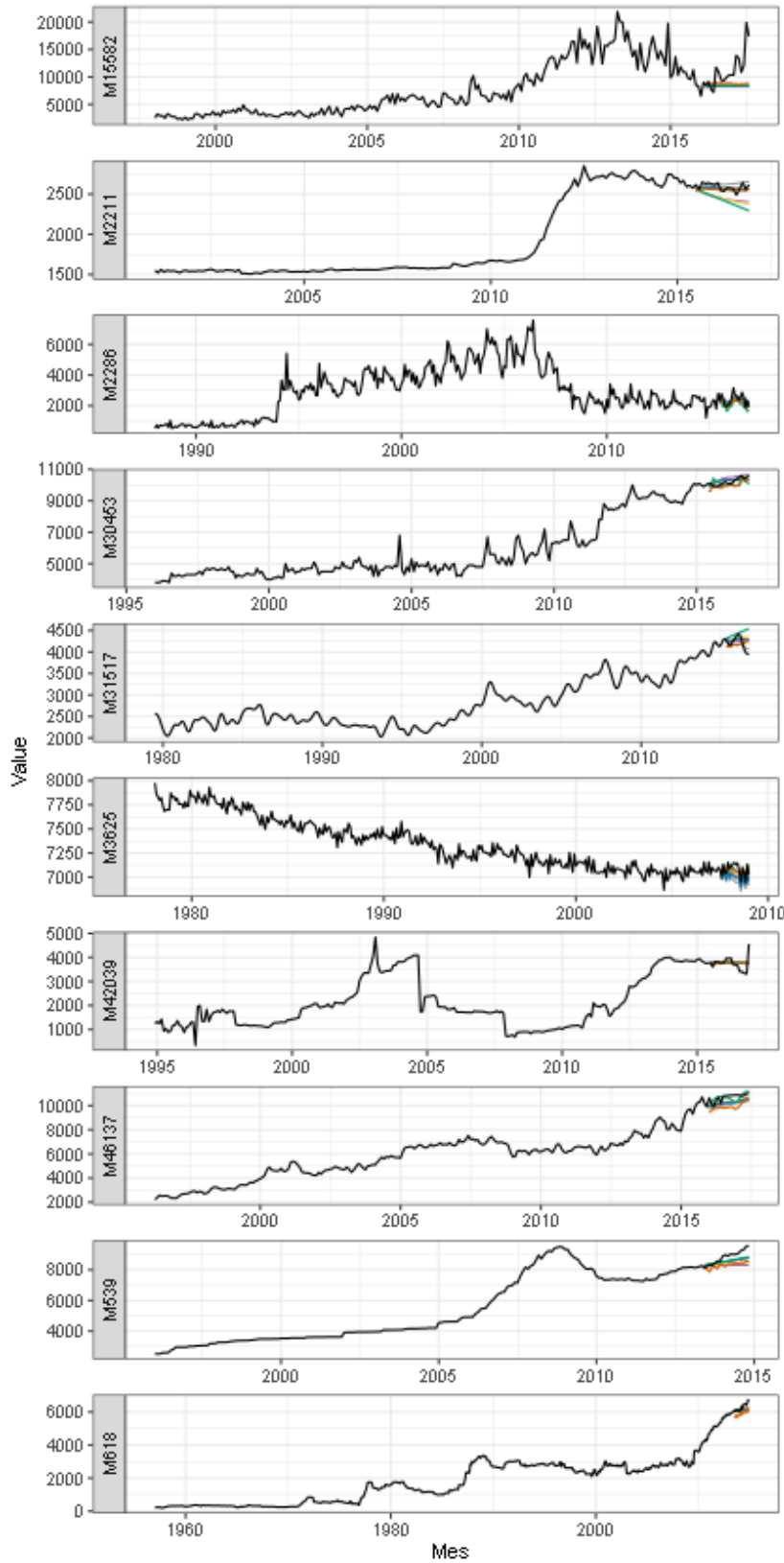


Serie ampliada

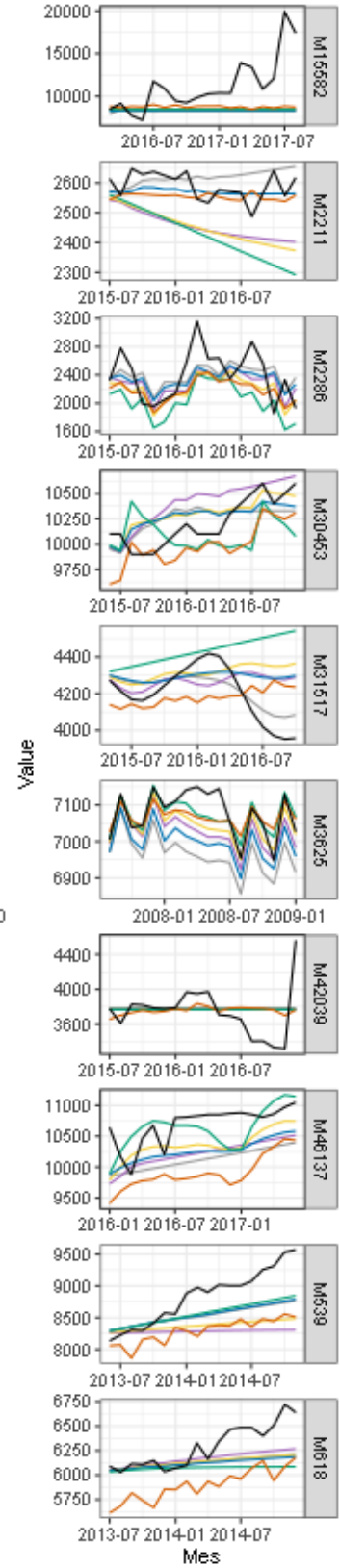


— Valor    Modelo    — ar\_01    — Comb1    — ets    — autoarima    — Comb2    — LSTMComb

Serie de Tiempo: Valor real y pronósticos

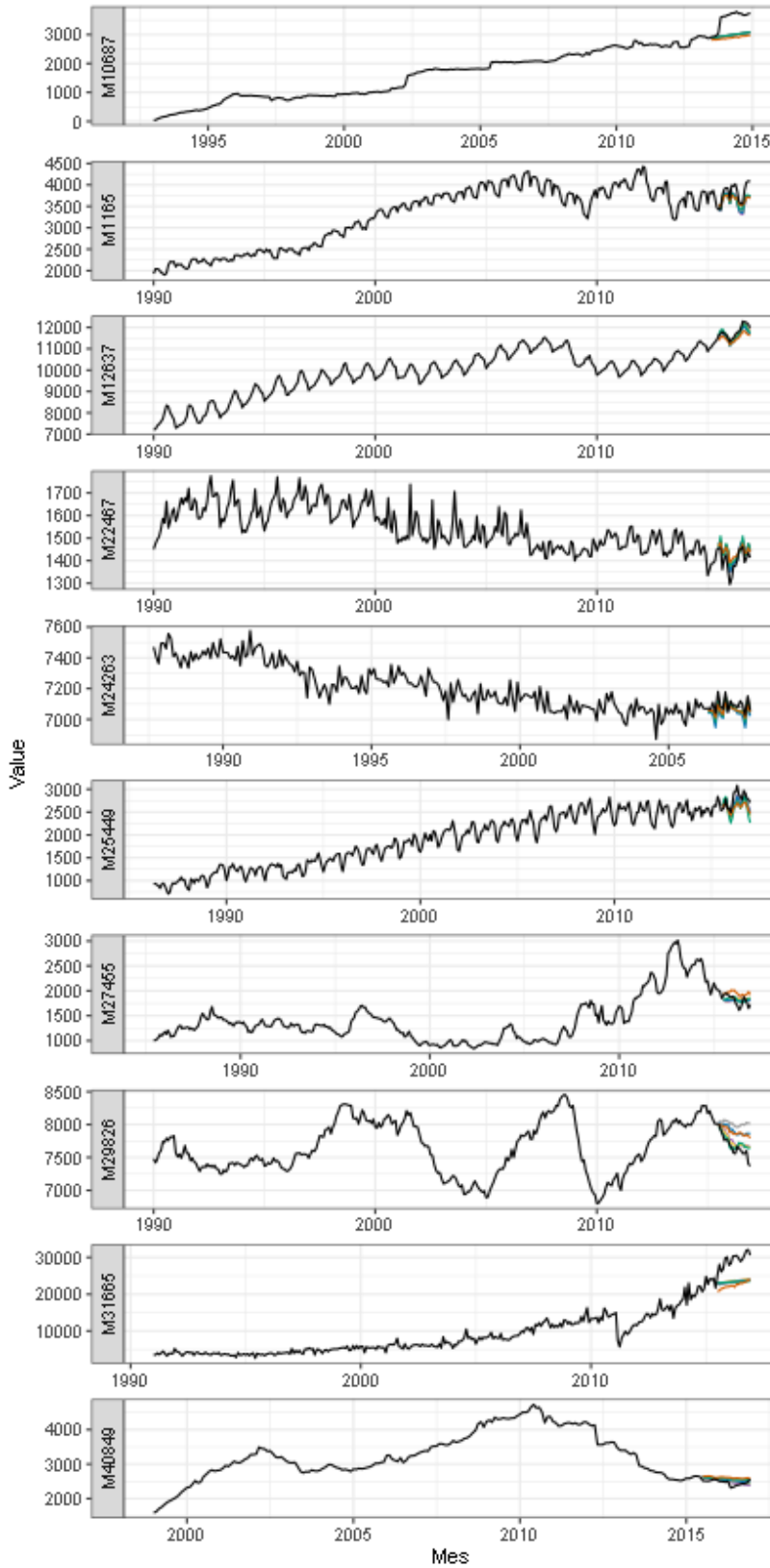


Serie ampliada

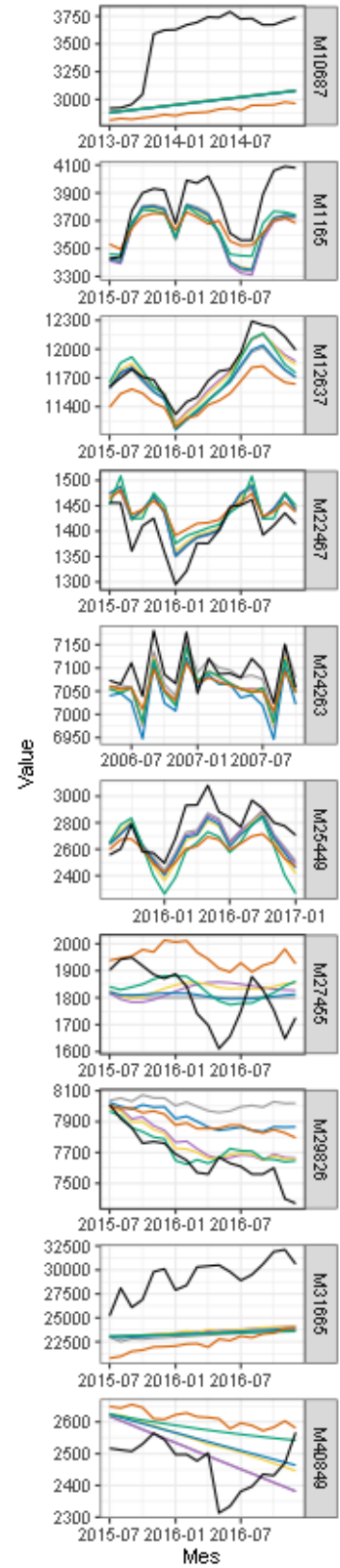


— Valor    Modelo    — ar\_01    — Comb1    — ets    — autoarima    — Comb2    — LSTMComb

Serie de Tiempo: Valor real y pronósticos



Serie ampliada



— Valor    Modelo    — ar\_01    — Comb1    — ets    — autoarima    — Comb2    — LSTMComb

Fuente: Elaboración propia

### 4.3. Modelamiento en la data real

#### 4.3.1. Modelo a utilizar

Considerando que fue el mejor en casi la cuarta parte de las series analizadas y tuvo un buen resultado al evaluar el promedio del sMAPE, se ha escogido en primera instancia al modelo Modelo de ETS cuyas componentes de Error, Tendencia, Estacionalidad se ajustan en función del AIC y máxima verosimilitud; adicionalmente también se ha escogido el modelo Comb2 debido a que fue el de mejor desempeño al evaluar el promedio acotado del sMAPE.

#### 4.3.2. Resultados en series de la empresa analizada

Procedemos a aplicar el modelo de ETS, a los diez productos analizados de la empresa comercializadora de ferretería, los modelos resultantes son:

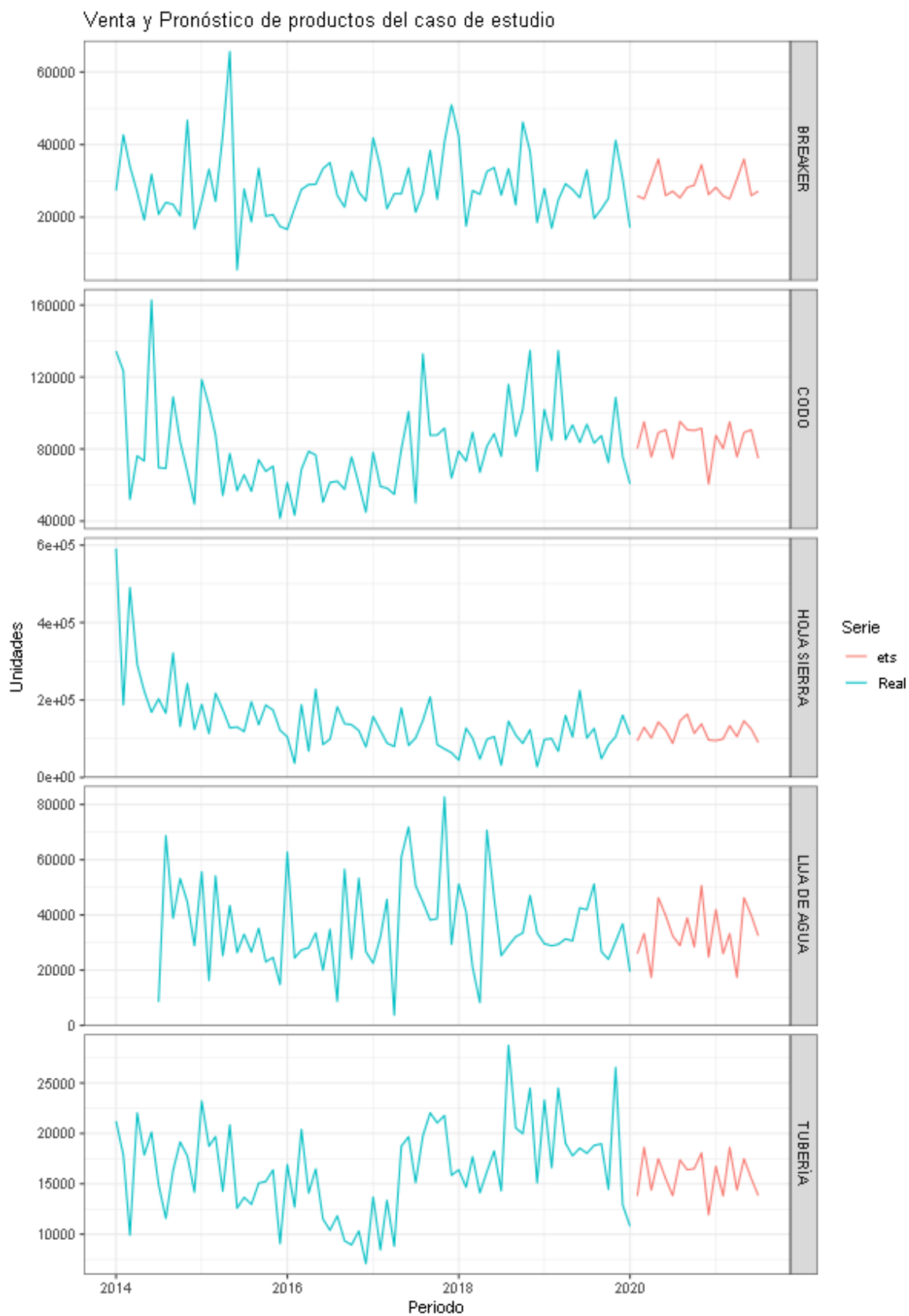
**Tabla 5 Modelos ajustados para los productos del caso de estudio**

SKU	DESCRIPCION	MODELO	DESCRIPCIÓN
00028050	TUBERÍA	ETS(M,N,A)	Error: Multiplicativo, Tendencia: No, Estacionalidad: Aditiva
00075499	BREAKER	ETS(M,N,A)	Error: Multiplicativo, Tendencia: No, Estacionalidad: Aditiva
00271981	CODO	ETS(M,N,A)	Error: Multiplicativo, Tendencia: No, Estacionalidad: Aditiva
00320101	HOJA SIERRA	ETS(M,Ad,A)	Error: Multiplicativo, Tendencia: Aditiva, Estacionalidad: Aditiva
00367557	LIJA DE AGUA	ETS(A,N,A)	Error: Aditiva, Tendencia: No, Estacionalidad: Aditiva
00411793	DISCO METAL	ETS(M,N,A)	Error: Multiplicativo, Tendencia: No, Estacionalidad: Aditiva
00444190	CINTA TEMFLEX	ETS(M,N,A)	Error: Multiplicativo, Tendencia: No, Estacionalidad: Aditiva
00563072	TOMACORRIENTE	ETS(A,Ad,A)	Error: Aditiva, Tendencia: Aditiva, Estacionalidad: Aditiva
00624131	CABLE	ETS(M,N,A)	Error: Multiplicativo, Tendencia: No, Estacionalidad: Aditiva
00624804	FOCO	ETS(A,N,A)	Error: Aditiva, Tendencia: No, Estacionalidad: Aditiva

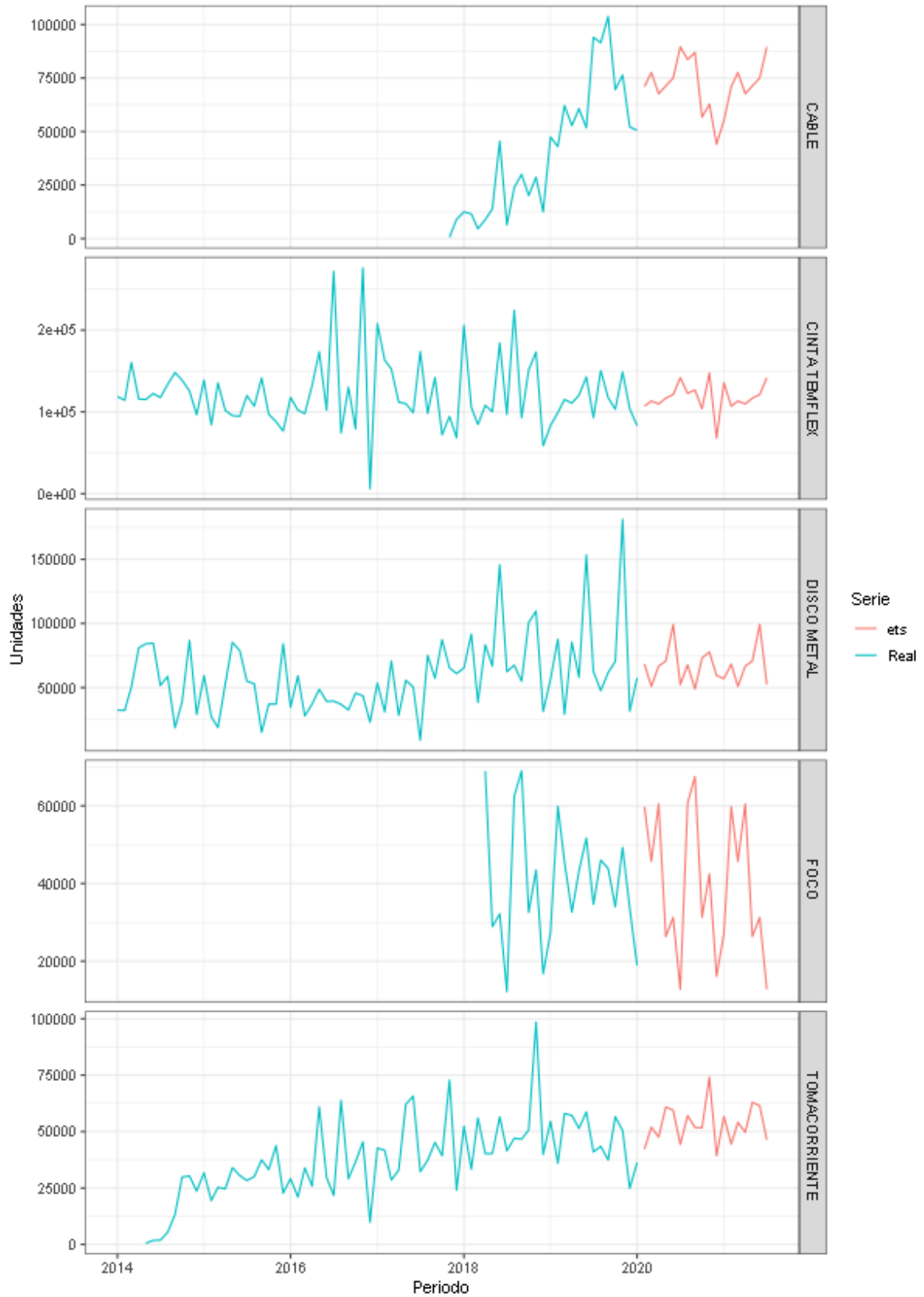
*Fuente: Elaboración propia*

A continuación, se muestran gráficamente los pronósticos generados:

### Ilustración 17 Caso de Estudio: Series de tiempo y pronósticos con modelo ETS



Venta y Pronóstico de productos del caso de estudio

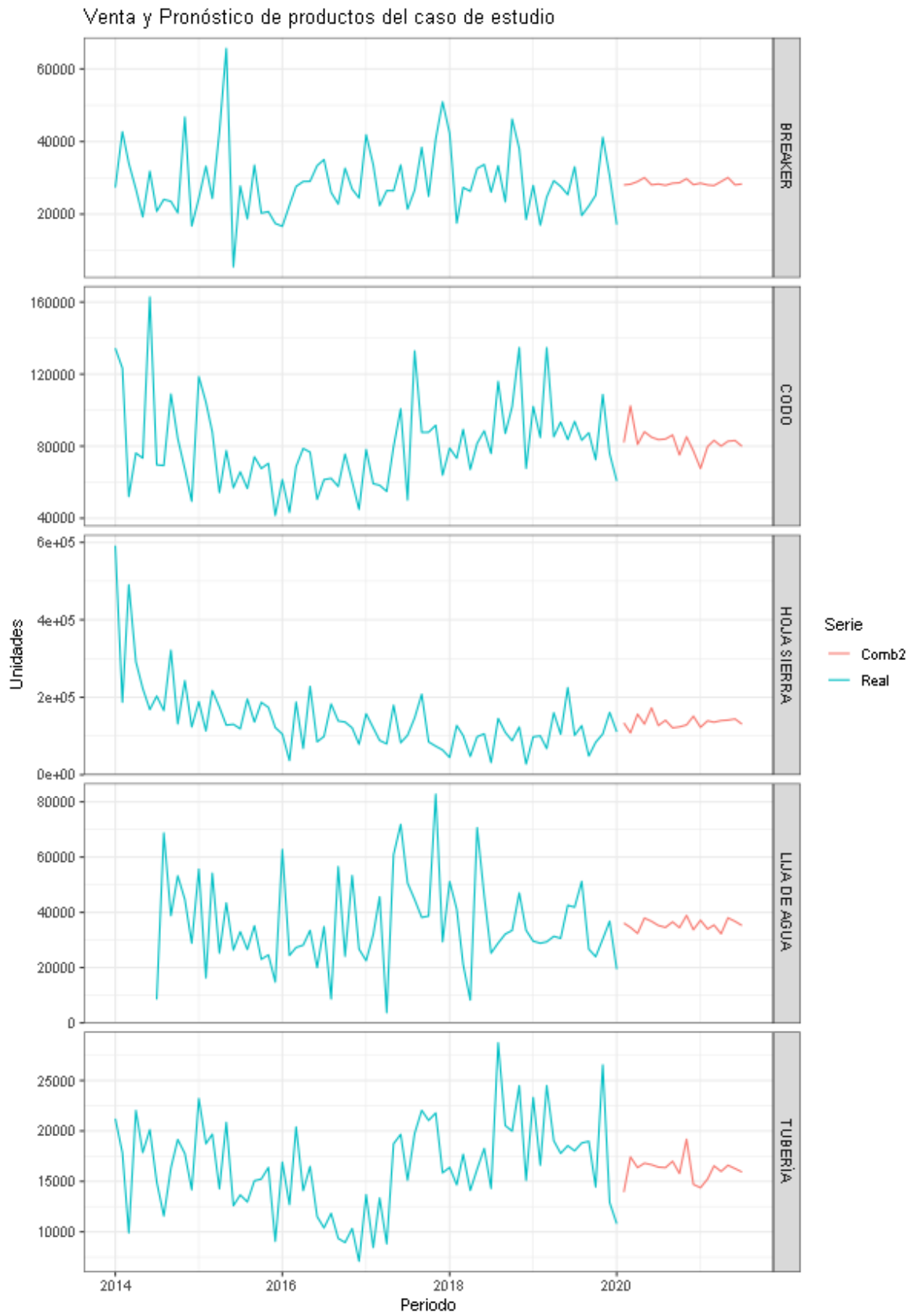


Fuente: Elaboración propia

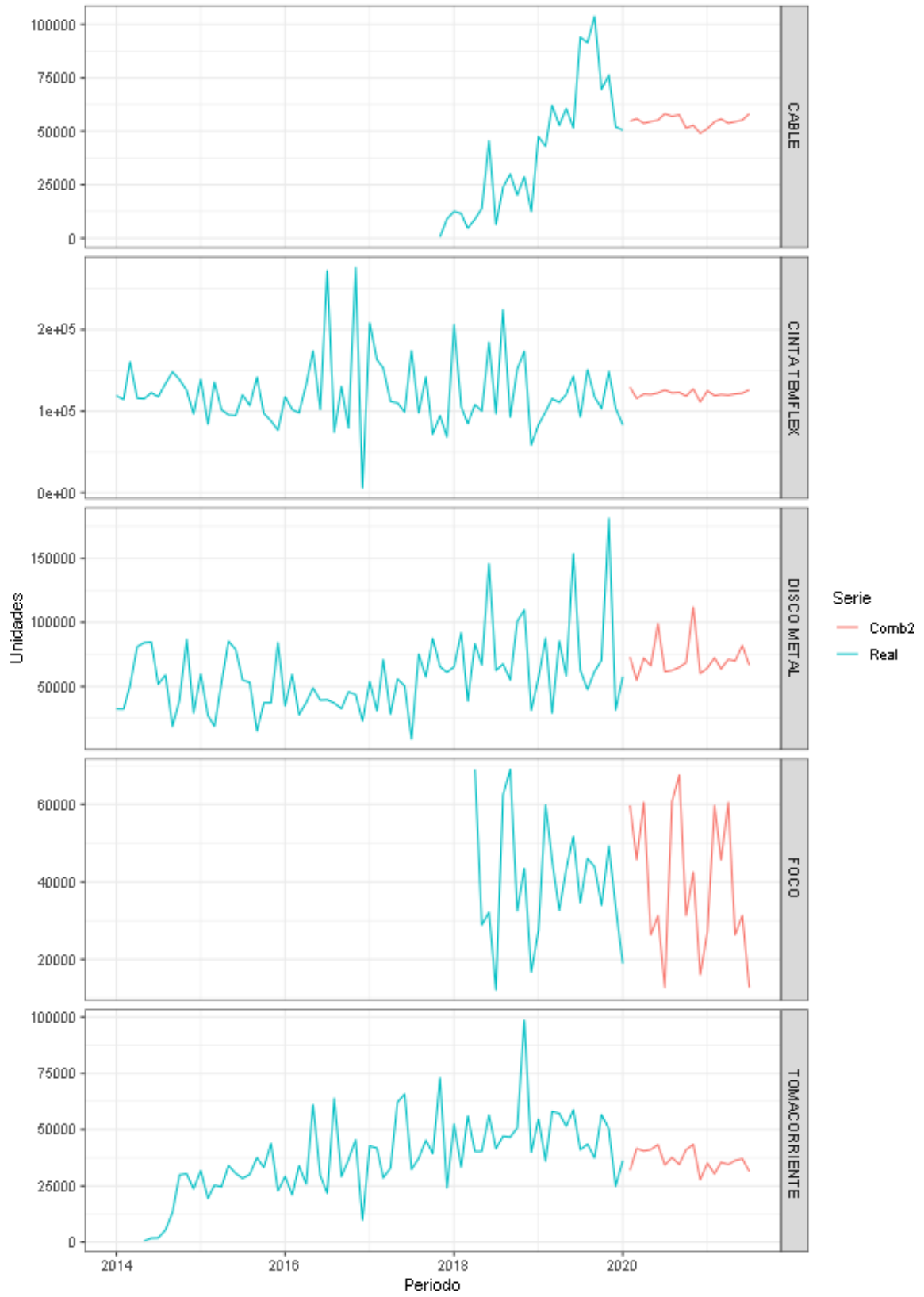
A continuación el resultado de aplicar el modelo Comb2



### Ilustración 18 Caso de Estudio: Series de tiempo y pronósticos con modelo Comb2



Venta y Pronóstico de productos del caso de estudio



Fuente: Elaboración propia

# CAPÍTULO 5

## 5. CONCLUSIONES Y RECOMENDACIONES

### 5.1. Conclusiones

En este proyecto el objetivo es encontrar una metodología para pronóstico de venta en unidades de productos del sector ferretero, para cumplir este objetivo se han evaluado diversas metodologías a una muestra de las series temporales pertenecientes a la M4 competition la cual consta de series de tipo microempresariales, industriales, macroeconómicas, financieras y demográficas.

En la etapa de evaluación se genera un total de 3000 modelos, 2000 para evaluar el modelo combinatorio basado en redes neuronales puesto que se evalúa 4 combinaciones de hiperparámetros en cada una de las 100 Series correspondientes a la muestra de la M4 Competition y a su vez que cada combinación sea entrena en 5 Remuestras. Los otros 1000 modelos más corresponden a los 10 tipos de técnicas (incluyendo la mejor combinación para redes neuronales) en las 100 series correspondientes a la muestra de la M4 Competition.

En la evaluación de los 10 modelos en la muestra de la M4 Competition los mejores resultados, en términos del sMAPE, se obtuvieron con el modelo de estado-espacio ETS cuyas componentes de Error, Tendencia, Estacionalidad (ETS) ajustado en función del AIC y máxima verosimilitud y con el Modelo que es el resultado de la combinación aritmética entre los modelos ets, ar\_01, ar\_02, ma\_01 y ma\_02.

La combinación de modelos no ajustados con AIC ha resultado con mejores evaluaciones que la combinación que utilizaba un modelo arima y sarima ajustado en base al AIC.

Se diseñó un algoritmo que combina metodologías de pronósticos Box-Jenkins, suavización exponencial (ETS por sus siglas en inglés) y una arquitectura de red neuronal con múltiples entradas, capas Convolucionales de una dimensión, capas Long Short Time Memory LSTM y Capas Dense, sin embargo, esta arquitectura sólo fue mejor en el 10% de las series usadas en la evaluación de métodos.

En la muestra tomada de la M4 Competition al analizar promedio del sMAPE el modelo ets quedó en 5to puesto, luego de 4 de los 5 mejores modelos de la competición. Mientras tanto, comparando el promedio acotado del sMAPE el modelo combinatorio Comb2 queda en 4to puesto debajo de 3 de los 5 modelos ganadores; cabe resaltar que el que un modelo combinatorio tenga buenos resultados es consistente con los resultados de la M4 Competition.

En cuanto a los productos de la empresa analizada, se ajustaron modelos ets a cada uno de ellos, resultando en modelos de tipo:

- Error: Multiplicativo, Tendencia: No, Estacionalidad: Aditiva
- Error: Multiplicativo, Tendencia: Aditiva, Estacionalidad: Aditiva
- Error: Aditiva, Tendencia: No, Estacionalidad: Aditiva
- Error: Aditiva, Tendencia: Aditiva, Estacionalidad: Aditiva

Así también se ajustaron las series con el Modelos Comb2, ambos modelos deberán ser usados y evaluados para la toma de decisiones en la empresa.

## **5.2. Recomendaciones**

Al ver que la combinación de los modelos ets, ar\_01, ar\_02, ma\_01 y ma\_02 tuvo muy buenos resultados, se recomienda probar el resultado de agregar otros modelos a dicha combinación.

A la empresa se le propone una metodología donde se utilice el modelo Comb2 y el modelo ETS en los productos, esto sumado a un proceso de evaluación constante de ambos modelos para, en base a dicha evaluación, escoger que opción utilizar en cada producto.

La arquitectura de red neuronal propuesta para el presente proyecto fue la mejor en el 10% de los casos evaluados, se recomienda seguir probando esta idea realizando pequeñas modificaciones a la arquitectura propuesta; por ejemplo que las capas convolucionales tengan una ventana de mayor cantidad de meses.

Siguiendo el punto anterior se recomienda repetir el estudio aumentando el tamaño muestral (o usar todas las series de la Competición M4) tanto para confirmar las conclusiones como para probar nuevas combinaciones o incluir nuevas metodologías de pronóstico.

## 6. Referencias

- Armstrong, S. (1989). Combining Forecasts: The end of the beginning or the beginning of the end? *International Journal of Forecasting*, 5(4), 585–588.  
[https://doi.org/10.1016/0169-2070\(89\)90013-7](https://doi.org/10.1016/0169-2070(89)90013-7)
- Atiya, A. (2019). Why does forecast combination work so well? *International Journal of Forecasting*, 36(1), 197–200. <https://doi.org/10.1016/j.ijforecast.2019.03.010>
- Bates, J., & Granger, C. (1969). The Combination of Forecasts. *Operational Research Quarterly*, 20(4), 451–468.
- Box, G., Jenkins, G., Reinsel, G., & Ljung, G. (2016). *Time Series Analysis: Forecasting and Control* (Fifth, Vol. 3, Issue 2). John Wiley & Sons.  
<http://repositorio.unan.edu.ni/2986/1/5624.pdf>
- Brockwell, P. J., & Davis, R. A. (2016). *Introduction to Time Series and Forecasting* (Third, Vol. 92, Issue 440). Springer Nature. <https://doi.org/10.2307/2965440>
- Chollet, F. (2016). *Construyendo autoencoders en keras*. El Blog de Keras.
- Ghatak, A. (2017). *Machine learning with R*. Springer Singapore.
- Giriya, S. (2016). Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *Software Available from Tensorflow. Org*.
- Harbola, S., & Coors, V. (n.d.). *Convolutional Neural Network Architectures for Wind Analysis*.
- Hassani, H., Silva, E., Gupta, R., & Das, S. (2018). Predicting global temperature anomaly: A definitive investigation using an ensemble of twelve competing forecasting models. *Physica A*, 509, 121–139. <https://doi.org/10.1016/j.physa.2018.05.147>
- Hyndman, R., & Athanasopoulos, G. (2019). *Forecasting: Principles and Practice* (Third).
- Hyndman, R., Koehler, A., Ord, J., & Snyder, R. (2008). Forecasting with exponential smoothing: the state space approach. In *Forecasting with Exponential Smoothing: The State Space Approach*. Springer Science & Business Media.
- Hyndman, R., Koehler, A., Snyder, R., & Grose, S. (2002). A state space framework for automatic forecasting using exponential smoothing methods. *International Journal of Forecasting*, 18(3), 439–454.
- Hyndman, R., & Yeasmin, K. (2008). Automatic time series forecasting: The forecast package for R. *Journal of Statistical Software*, 27(3), 1–22.
- Jiang, S., Xiao, R., Wang, L., Luo, X., Huang, C., Wang, J.-H., Chin, K.-S., & Nie, X.

- (2019). Combining Deep Neural Networks and Classical Time Series Regression Models for Forecasting Patient Flows in Hong Kong. *IEEE Access*, 7, 118965–118974. <https://doi.org/10.1109/access.2019.2936550>
- Li, M., Ji, S., & Liu, G. (2018). Forecasting of Chinese E-Commerce Sales: An Empirical Comparison of ARIMA, Nonlinear Autoregressive Neural Network, and a Combined ARIMA-NARNN Model. *Mathematical Problems in Engineering*, 2018, 1–12. <https://doi.org/10.1155/2018/6924960>
- Makridakis, S. (1993). Accuracy measures: theoretical and practical concerns. *International Journal of Forecasting*, 9(4), 527–529.
- Makridakis, S., & Hibon, M. (2000). The M3-Competition: results, conclusions and implications. *International Journal of Forecasting*, 16(4), 451–476. [https://doi.org/10.1016/S0169-2070\(00\)00057-1](https://doi.org/10.1016/S0169-2070(00)00057-1)
- Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2018). The M4 Competition: Results, findings, conclusion and way forward. *International Journal of Forecasting*, 34(4), 802–808. <https://doi.org/10.1016/j.ijforecast.2018.06.001>
- Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2020). The M4 Competition: 100,000 time series and 61 forecasting methods. *International Journal of Forecasting*, 36(1), 54–74. <https://doi.org/10.1016/j.ijforecast.2019.04.014>
- Montero-Manso, P., Athanasopoulos, G., Hyndman, R., & Talagala, T. (2020). FFORMA: Feature-based forecast model averaging. *International Journal of Forecasting*, 36(1), 86–92.
- Muth, J. (1960). Optimal Properties of Exponentially Weighted Forecasts. *Journal of the American Statistical Association*, 55(1), 299–306.
- Pytrick, L. (2019). *Red neuronal recurrente (RNN), memoria a corto y largo plazo (LSTM) y unidad recurrente cerrada (GRU)*. DProgrammer Lopez. <http://dprogrammer.org/rnn-lstm-gru>
- Ramasubramanian, K., & Singh, A. (2019). *Machine Learning Using R: With Time Series and Industry-Based Use Cases in R* (Second). Apress.
- Snyder, R. (1985). Recursive Estimation of Dynamic Linear Models. *Journal of the Royal Statistical Society*, 47(2), 272–276.
- Team, R. C. (2013). *R: A language and environment for statistical computing*.
- Terui, N., & Van Dijk, H. K. (2002). Combined forecasts from linear and nonlinear time series models. *International Journal of Forecasting*, 18(3), 421–438.

[https://doi.org/10.1016/S0169-2070\(01\)00120-0](https://doi.org/10.1016/S0169-2070(01)00120-0)

- Van Rossum, G., & Drake, F. (2011). *The python language reference manual*. Network Theory Ltd.
- Winkler, R., & Makridakis, S. (1983). The Combination of Forecasts. *Journal of the Royal Statistical Society: Series A*, 146(2), 150–157.
- Wold, H. (1938). On the inversion of moving averages. *Revista Actuarial Escandinava*, 1938(3–4), 208–217.
- Yule, U. (1921). On the Time-Correlation Problem, with Especial Reference to the Variate- Difference Correlation Method. *Journal of the Royal Statistical Society*, 84(4), 497–537.
- Yule, U. (1926). Why do we Sometimes get Nonsense-Correlations between Time-Series ? --A Study in Sampling and the Nature of Time-Series. *Journal of the Royal Statistical Society*, 89(1), 1–63. <http://www.jstor.org/stable/2341482>
- Yule, U. (1927). On a Method of Investigating Periodicities in Disturbed Series, with Special Reference to Wolfer's Sunspot Numbers. *Philosophical Transactions of the Royal Society of London*, 226(636–646), 267–298.  
<https://doi.org/10.1098/rsta.1927.0007>