



UNIVERSIDAD DE CHILE  
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS  
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA

**ANÁLISIS DE PROCESOS DE PLANTAS QUÍMICAS CON MODELOS DE  
REDES ALEATORIAS Y DE ESPACIO ESTADO**

MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL ELÉCTRICO

**ERWIN NICOLÁS PAILLACÁN HUAITRO**

PROFESOR GUÍA:  
ALBERTO CASTRO ROJAS

MIEMBROS DE LA COMISIÓN:  
CESAR A. AZURDIA MEZA  
PABLO PALACIOS JÁTIVA

ESTE TRABAJO HA SIDO PARCIALMENTE FINANCIADO POR:  
SOCIEDAD QUÍMICA Y MINERA DE CHILE S.A.

SANTIAGO DE CHILE  
2021

RESUMEN DE LA MEMORIA PARA OPTAR  
AL TÍTULO DE INGENIERO CIVIL ELÉCTRICO  
POR: **ERWIN NICOLÁS PAILLACÁN HUAITRO**  
FECHA: 23 de septiembre de 2021  
PROF. GUÍA: ALBERTO CASTRO ROJAS

## **ANÁLISIS DE PROCESOS DE PLANTAS QUÍMICAS CON MODELOS DE REDES ALEATORIAS Y DE ESPACIO ESTADO**

SQM produce carbonato de litio en su planta Salar del Carmen, cerca de Antofagasta, Chile, a partir de soluciones con altas concentraciones de litio provenientes del Salar de Atacama. El carbonato de litio debido a su potencial electroquímico y su liviano peso es principalmente usado en la construcción de cátodos para baterías ion litio, las cuales han tenido una creciente demanda debido al uso en vehículos eléctricos, computadores portátiles, etc. En este trabajo se modela la etapa de micronizado - empaquetado de la producción de este elemento mediante modelos redes aleatorias y de espacio estado.

La metodología propuesta consta de una etapa de preprocesamiento de los datos, luego se genera la modelación del proceso mediante redes bayesianas gaussianas dinámicas (*DGBN*) y modelos de espacio estado, donde se ponen a prueba el uso de conocimiento experto y ensamblajes para las variables de control, con lo que se generan 23 modelos distintos tanto multivariantes como univariantes para las variables objetivos. Posteriormente, se continúa con el desarrollo de las herramientas para su despliegue en línea, llámese código fuente y aplicaciones web. Finalmente, las herramientas desarrolladas se actualizan de forma automatizada con lo cual los modelos y sus resultados se supervisan diariamente.

Con respecto a los resultados obtenidos basados en el modelamiento del proceso se verifica que los modelos de tipo espacio estado y su ajuste mediante filtro Kalman no resultan apropiados para el proceso estudiado, debido a que se cuenta con una baja frecuencia en la actualización de la base de datos, ocasionando que se requiera un horizonte de *forecasting* demasiado amplio. El modelo que obtuvo los mejores resultados fue *DGBN* sin restricciones, sin embargo, se opta por su versión con uso de arcos obligatorios para el estudio de la influencia de las variables de control. Los análisis muestran que la variación de las variables de control ocasiona comportamientos opuestos sobre las variables objetivo.

# Tabla de Contenido

<b>1. Introducción</b>	<b>1</b>
1.1. Objetivos . . . . .	2
1.1.1. Objetivo general . . . . .	2
1.1.2. Objetivos específicos . . . . .	2
1.2. Alcances . . . . .	3
1.3. Estructura de la memoria . . . . .	3
<b>2. Antecedentes</b>	<b>4</b>
2.1. Sociedad Química y Minera . . . . .	5
2.1.1. Producción Carbonato de Litio . . . . .	6
2.2. Estado del arte . . . . .	9
<b>3. Marco teórico</b>	<b>10</b>
3.1. Definiciones . . . . .	10
3.2. Modelos espacio estado . . . . .	12
3.3. Modelos dinámicos lineales . . . . .	14
3.3.1. Modelo local lineal . . . . .	14
3.3.2. Modelo local lineal con tendencia . . . . .	14
3.3.3. Modelo autorregresivo de media móvil . . . . .	15
3.4. Filtro de Kalman . . . . .	16
3.5. Modelos de volatilidad de tipo GARCH . . . . .	17
3.6. Red bayesiana . . . . .	18
3.7. Red bayesiana gaussiana dinámica . . . . .	19
3.7.1. Aprendizaje de la estructura . . . . .	20
3.7.2. Aprendizaje de parámetros . . . . .	21
3.7.2.1. MLE en Gaussiana multivariable . . . . .	21
3.7.3. Inferencia . . . . .	22
3.8. Métodos de imputación . . . . .	23
3.9. Métricas de error . . . . .	25
3.10. Verificación de modelos . . . . .	25
3.10.1. QQ-Plot . . . . .	26
3.10.2. Gráfico de Autocorrelación . . . . .	27
3.10.3. Prueba de Ljung-Box . . . . .	27
3.10.4. Prueba de Kolmogorov-Smirnov . . . . .	28
<b>4. Metodología</b>	<b>29</b>
4.1. Flujo de trabajo . . . . .	30
4.2. Preprocesamiento de datos . . . . .	31

4.2.1.	Análisis datos nulos . . . . .	32
4.2.2.	Imputación de datos . . . . .	34
4.2.3.	Suavizado de datos . . . . .	35
4.2.4.	Detección y eliminación de anomalías . . . . .	36
4.3.	Modelado y <i>forecasting</i> . . . . .	37
4.3.1.	Modelos univariados . . . . .	37
4.3.2.	Modelos multivariados . . . . .	38
4.3.2.1.	Restricciones . . . . .	39
4.3.3.	Ensamble de modelos . . . . .	39
4.3.4.	Experimentación modelos . . . . .	40
4.4.	Dashboard web . . . . .	41
4.4.1.	Selección de hiper-parámetros . . . . .	41
4.4.2.	Visualización redes entrenadas . . . . .	41
4.4.3.	Visualización de parámetros de redes . . . . .	42
4.4.4.	Visualización de métricas de error . . . . .	43
4.4.5.	Visualización <i>forecasting Out of sample</i> . . . . .	44
<b>5.</b>	<b>Resultados</b>	<b>45</b>
5.1.	Periodo particular . . . . .	45
5.1.1.	Diagnóstico de modelos . . . . .	45
5.1.1.1.	Modelos ARMA . . . . .	45
5.1.1.2.	Modelos LLT . . . . .	48
5.1.1.3.	Modelo GARCH . . . . .	51
5.1.2.	<i>Forecasting out sample</i> . . . . .	53
5.1.3.	Estructuras de red . . . . .	56
5.2.	Periodo general . . . . .	60
<b>6.</b>	<b>Discusión de resultados</b>	<b>64</b>
6.1.	Uso de red como simulador . . . . .	65
<b>7.</b>	<b>Conclusiones</b>	<b>68</b>
	Trabajos futuros . . . . .	68
	Inferencia bayesiana y modelos jerárquicos . . . . .	69
	Modelos de ensamble . . . . .	70
	<b>Glosario</b>	<b>72</b>
	<b>Bibliografía</b>	<b>73</b>
	<b>Anexo A. Figuras</b>	<b>76</b>
A.1.	Resultados . . . . .	76

# Índice de Tablas

2.1.	Variables proceso micronizado y envasado. . . . .	8
4.1.	Porcentaje de nulos sobre datos agregados a 2H. Fecha final 2021-05-10. . . . .	33
4.2.	Arcos prohibidos. . . . .	39
4.3.	Arcos obligatorios. . . . .	39
4.4.	Lista de combinaciones de modelos para tasas de envasado. . . . .	40
5.1.	Métricas de error: Tasa envasado MIC 4870 . . . . .	54
5.2.	Métricas error: Tasa envasado MIC 4880. . . . .	55
5.3.	DGBN sin restricciones: nodos $t_0$ y sus padres . . . . .	57
5.4.	DGBN con restricciones: nodos $t_0$ y sus padres . . . . .	59
5.5.	MAPE Tasa envasado MIC 4870. . . . .	60
5.6.	RMSPE Tasa envasado MIC 4870. . . . .	60
5.7.	MAPE Tasa envasado MIC 4880. . . . .	62
5.8.	RMSPE Tasa envasado MIC 4880. . . . .	62

# Índice de Ilustraciones

2.1.	Ubicación del Salar del Carmen. . . . .	5
2.2.	Producción de carbonato de litio. Fuente: SQM. . . . .	6
2.3.	Etapas de producción Carbonato de Litio. Fuente: SQM. . . . .	6
2.4.	Proceso micronizado y envasado. . . . .	8
3.1.	Grafo dirigido y ponderado. . . . .	10
3.2.	Representación estructura de dependencia espacio estado. . . . .	12
3.3.	Ejemplo Red Bayesiana 4 Nodos. Imagen obtenida desde [22] . . . . .	18
3.4.	Red bayesiana dinámica de primer orden. Fuente: pyAgrum. . . . .	19
3.5.	QQ-Plot: Simulación distribución normal. . . . .	27
3.6.	ACF Plot. . . . .	27
4.1.	Flujo de trabajo general. . . . .	30
4.2.	Bloque de preprocesamiento de datos. . . . .	31
4.3.	Datos en Bruto. Agregación original 1H. Periodo: 2020-04-01 a 20-05-10. . . . .	32
4.4.	Datos Imputados. Periodo: 2021-01-01 a 20-05-10. . . . .	34
4.5.	Datos suavizados. Periodo 2021-01-01 a 2021-05-10. . . . .	35
4.6.	Anomalías tasa de envasado. . . . .	36
4.7.	Esquema de generación de <i>forecasting</i> . Modelos univariados. . . . .	38
4.8.	Inferencia sobre modelo DGBN. Modelo multivariado. . . . .	38
4.9.	Selección de hiper-parámetros. . . . .	41
4.10.	Visualización de redes. . . . .	42
4.11.	Visualización de parámetros. . . . .	42
4.12.	Métricas de error sobre datos de validación. . . . .	43
4.13.	Visualización de <i>forecasting Out of sample</i> . . . . .	44
5.1.	Residuos modelo ARMA: Flujo de aire soplador. . . . .	46
5.2.	Residuos modelo ARMA: Velocidad separador. . . . .	47
5.3.	Residuos modelo ARMA: Tasa envasado MIC 4870. . . . .	47
5.4.	Residuos modelo ARMA: Tasa envasado MIC 4870. . . . .	48
5.5.	Residuos modelo LLT: Flujo de aire soplador. . . . .	49
5.6.	Residuos modelo LLT: Velocidad separador. . . . .	49
5.7.	Residuos modelo LLT: Tasa envasado MIC 4870. . . . .	50
5.8.	Residuos modelo LLT: Tasa envasado MIC 4870. . . . .	50
5.9.	Residuos modelo apARCH: Flujo de aire soplador. . . . .	51
5.10.	Residuos modelo apARCH: Velocidad separador. . . . .	51
5.11.	Residuos modelo apARCH: Tasa envasado MIC 4870. . . . .	52
5.12.	Residuos modelo apARCH: Tasa envasado MIC 4880. . . . .	52
5.13.	<i>Forecasting out of sample</i> sobre validación: Tasa envasado MIC 4870. . . . .	53
5.14.	<i>Forecasting</i> sobre validación: Tasa envasado MIC 4880. . . . .	54
5.15.	Estructura DGBN. . . . .	56

5.16.	Estructura DGBN + Restricciones. . . . .	58
5.17.	Distribución RMSPE: Tasa envasado MIC 4870. . . . .	61
5.18.	Prueba Kolmogorov-Smirnov <i>p-values</i> : Tasa envasado MIC 4870. . . . .	61
5.19.	Distribución RMSPE: Tasa envasado MIC 4880. . . . .	62
5.20.	Prueba Kolmogorov-Smirnov <i>p-values</i> : Tasa envasado MIC 4880. . . . .	63
6.1.	Simulación: Tasa envasado MIC 4870. . . . .	66
6.2.	Simulación: Tasa envasado MIC 4880. . . . .	66
6.3.	Simulación: Diferencia tasas de envasado. . . . .	67
A.1.	Validación <i>Forecasting in sample</i> a un paso. . . . .	76

# Capítulo 1

## Introducción

Actualmente la industria se encuentra en proceso de adopción de la cuarta revolución industrial, la cual ha conllevado la tendencia de incorporar sistemas físicos y digitales en ambientes de producción. Tal integración permite recolectar grandes cantidades de datos, que en este contexto guardan información sobre equipos, procesos, eventos, alarmas, entre otros. Como consecuencia de la creciente disponibilidad de datos, su análisis se ha convertido en una importante herramienta para el desarrollo de diversas áreas, tales como: decisiones estratégicas, reducción de costos de mantención, detección de fallas de las maquinarias, incremento de vida útil, maximización de producción, entre otras.

La maximización de producción, en cuanto a cantidad y calidad, manteniendo los costos de producción lo más bajo posible, implica la necesidad de mantener el funcionamiento de la planta con una alta disponibilidad, lo cual significa reducir al mínimo tanto los tiempos de detención como de operación en puntos no óptimos. Por lo tanto, una buena estrategia de mantenimiento apunta a mejorar la condición de operación, reducir la tasa de fallas y minimizar los costos operacionales. Para lidiar con las tareas de mantenimiento, existen variadas estrategias que pueden ser resumidas en cuatro categorías [1]-[2]:

- **Mantenimiento *run-to-failure***: se realizan reparaciones después de la ocurrencia de un fallo. Este es el enfoque más sencillo de la gestión del mantenimiento y, por lo general, el más costoso debido al gran número de productos defectuosos que se obtienen como consecuencia del fallo.
- **Mantenimiento preventivo (o mantenimiento programado)**: hace alusión al mantenimiento que se lleva a cabo periódicamente bajo un calendario preestablecido con el fin de anticiparse a los fallos del proceso. En este enfoque, los fallos se suelen compensar, y, a veces, se realizan mantenimientos innecesarios.
- **Mantenimiento basado en la condición (MBC)**: alude a cuando el mantenimiento es relativo al proceso de verificación de una o más condiciones que indican degradación en la maquinaria. Este enfoque se basa en la vigilancia continua de la salud de la máquina o del proceso y permite realizar mantenimientos solo cuando son realmente necesarios. El inconveniente de la gestión de la MBC es que los mantenimientos no pueden planificarse de antemano.
- **Mantenimiento predictivo (PdM)**: al igual que en la MBC, las medidas de mantenimiento se adoptan solo cuando son necesarias. Sin embargo, se utilizan instrumentos

de predicción para evaluar cuándo es probable que se necesiten esas medidas, con lo que se aplican algunos proyectos de planificación y programación. Los sistemas de PdM pueden emplear factores de salud definidos *ad hoc* o, en general, métodos de inferencia estadística.

En el último tiempo, debido a la mayor disponibilidad de datos, se ha incrementado el interés por crear soluciones basadas en el mantenimiento predictivo sobre la base que se propone estimar el futuro de un ítem a partir de su presente y pasado. Por supuesto, esta tarea contiene un grado de incertidumbre, dado que no se puede predecir el futuro con seguridad. Por lo tanto, bajo este enfoque se tiene dos escenarios posibles:

- Si la predicción de un sistema es de carácter seguro o satisfactorio, entonces ningún tipo de acción debe ser planeada o ejecutada.
- Si la predicción es un estado no aceptable o subóptimo, entonces se deben estudiar las acciones correspondientes para efectuar tareas de mantenimiento.

Aun cuando la popularidad de PdM está continuamente creciendo en el mundo académico, no muchas industrias lo han implementado en sus procesos. De acuerdo con un estudio [3] de casos, se concluyó que existe una brecha entre los beneficios potenciales y los beneficios realizados de PdM. Del mismo modo, en una encuesta [4] realizada a 250 empresas de países Bajos, Alemania y Bélgica, se mostró que solo el 11 % ha implementado algún tipo solución PdM, mientras que el 36 % de ellos todavía depende principalmente de inspección visual (MBC).

En el presente trabajo, se considera la planta Salar del Carmen para su modelación estadística como un sistema complejo encargado de la producción de carbonato de litio. Se proponen y comparan modelos basados en espacio estado y redes bayesianas gaussianas dinámicas para el descubrimiento de dependencias entre variables sensadas en el proceso.

## 1.1. Objetivos

### 1.1.1. Objetivo general

Modelar la etapa correspondiente al micronizado - envasado de carbonato de litio de la planta Salar del Carmen mediante redes aleatorias y espacio estado. Además de generar una herramienta de monitorización del o los modelos propuestos.

### 1.1.2. Objetivos específicos

- Desarrollar una metodología de preprocesamiento conformada por tres etapas secuenciales para la generación de bases de datos apropiadas para el entrenamiento de modelos. Dicha secuencia comienza con el imputado de datos faltantes, prosigue con una etapa de suavizado que busca la remoción de ruido de las mediciones y se finaliza con el filtrado de puntos anómalos o fuera de operación.
- Generar un modelo multivariable de la etapa micronizado - envasado mediante redes bayesianas gaussianas dinámicas que permitan realizar inferencias y pronósticos de las variables de interés.

- Generar modelos univariados para la variable de interés mediante modelos espacio estado: ARMA y Modelo local lineal con tendencia. Además, un modelo basado en estadísticas de segundo orden: GARCH.
- Comparar y validar los modelos propuestos con base en métricas de error y pruebas de bondad de ajuste.
- Programar y desplegar una aplicación web que permita la interacción y visualización de los distintos modelos propuestos.

## 1.2. Alcances

El presente trabajo se enmarca como un aporte a la empresa SOCIEDAD QUÍMICA Y MINERA DE CHILE S.A. (SQM). Se tiene como objetivo contribuir en la elaboración de un modelo del proceso micronizado - empaquetado que pueda ser usado por la empresa, lo cual implica la generación de herramientas necesarias para su despliegue.

## 1.3. Estructura de la memoria

La memoria comienza con el Capítulo 2, donde se describe brevemente el proceso industrial de la producción de carbonato de litio y se presenta el estado del arte de modelación de sistemas complejos. Para luego en el capítulo 3 presentar el marco teórico relacionado con los modelos propuestos, técnicas de imputación, métricas de error, entre otros. El capítulo 4 repasa la metodología desarrollada. El capítulo 5 presenta los resultados obtenidos de la aplicación de la metodología propuesta, mientras que en el capítulo 6 los resultados son discutidos. Finalmente, en el capítulo 7 se presentan las conclusiones y el trabajo futuro propuesto.

# Capítulo 2

## Antecedentes

El litio es un elemento químico y mineral que se encuentra típicamente en minerales y sales, ya que debido a sus propiedades reactivas no es posible de encontrar en forma metálica. Chile posee aproximadamente el 48 % de las reservas mundiales, las cuales geográficamente se encuentran en su mayor parte en depósitos salinos en la alta cordillera.

El litio y su uso, de acuerdo con el estudio hecho por Cochilco [5], puede ser clasificado principalmente en dos categorías:

1. Baterías recargables
  - a) **Electro-movilidad:** vehículos livianos, pesados, bicicletas, escúter, entre otros.
  - b) **Artículos electrónicos:** tablets, computadores, teléfonos, reloj inteligente, entre otros.
  - c) **Almacenamiento de energía:** mediante el uso de sistemas predictivos, se conserva energía para uso futuro.
2. Uso tradicional
  - a) **Vidrios y cerámicas:** la adición de litio a estos materiales les permite mejorar propiedad tales como la disminución de la expansión térmica o de la temperatura de fuego.
  - b) **Grasas y lubricantes:** permite el uso a temperaturas variables.
  - c) **Otros:** farmacéuticos, plásticos y polímeros.

En Chile, una de las formas comunes de encontrar litio es en salmueras, las cuales están conformadas por acumulaciones de aguas subterráneas salinas que poseen una alta concentración de litio disuelto. Para su extracción, la salmuera debe ser bombeada hacia la superficie, y su contenido es almacenado en piscinas, donde con ayuda de evaporación solar se genera un concentrado de salmuera de litio con valores que varían entre 3 % y 6 % [5]. Este producto es tratado posteriormente para la obtención de productos más refinados.

La demanda de litio se ha incrementado fuertemente en los últimos años, principalmente debido a la electro-movilidad. De acuerdo con Cochilco, la demanda pasó de 204 kilotoneladas de Carbonato de Litio Equivalente (kt. LCE) en 2016 a más de 323 kt. LCE en 2019. Si bien

como efecto de la pandemia COVID-19 se esperan bajas en las ventas de litio para automóviles eléctricos de pasajeros, se espera que su uso supere el 32 % de 2019, y alcance cerca de un 80 % en 2039.

## 2.1. Sociedad Química y Minera

La Sociedad Química y Minera de Chile (SQM) es una empresa minera privada que se dedica a la explotación, procesamiento y comercialización de productos químicos. En particular, SQM cuenta con cinco líneas de negocio: Nutrición Vegetal de Especialidad (NVE); Yodo y derivados; Litio y derivados; Potasio y Químicos Industriales.



Figura 2.1: Ubicación del Salar del Carmen.

En la planta Salar del Carmen, ubicada en las cercanías de Antofagasta<sup>1</sup> (ver Figura 2.1) se producen dos productos principalmente:

- **Carbonato de Litio:** se utiliza en diversas industrias, como la electrónica, en cerámicas, vidrios, etc. Debido a su elevado potencial electroquímico y a que es el elemento sólido más liviano, su principal uso es como material de cátodos en las baterías de ion litio. Estas baterías han tenido una fuerte y creciente demanda en el último tiempo, principalmente a causa del gran aumento del mercado de vehículos eléctricos, pero también por su uso en computadores portátiles, teléfonos móviles, cámaras digitales, etc. [6]
- **Hidróxido de Litio:** se utiliza principalmente en la producción de grasas lubricantes que pueden trabajar en condiciones extremas de temperatura y carga. Aproximadamente el 70 % de aquellas producidas en el mundo contienen litio. El hidróxido de litio se utiliza también en baterías y colorantes.

En 2020, SQM daba cuenta de una capacidad de producción de 70 kt de carbonato de litio con el objetivo de ampliación para llegar hasta los 150 kt/año para fines de 2020.

<sup>1</sup> Para más información, revisar en: <https://www.sqm.com/productos/litio-y-derivados/>

## 2.1.1. Producción Carbonato de Litio

A continuación se muestra en la Figura 2.2 el proceso de producción de carbonato de litio ( $Li_2CO_3$ ) en términos generales.

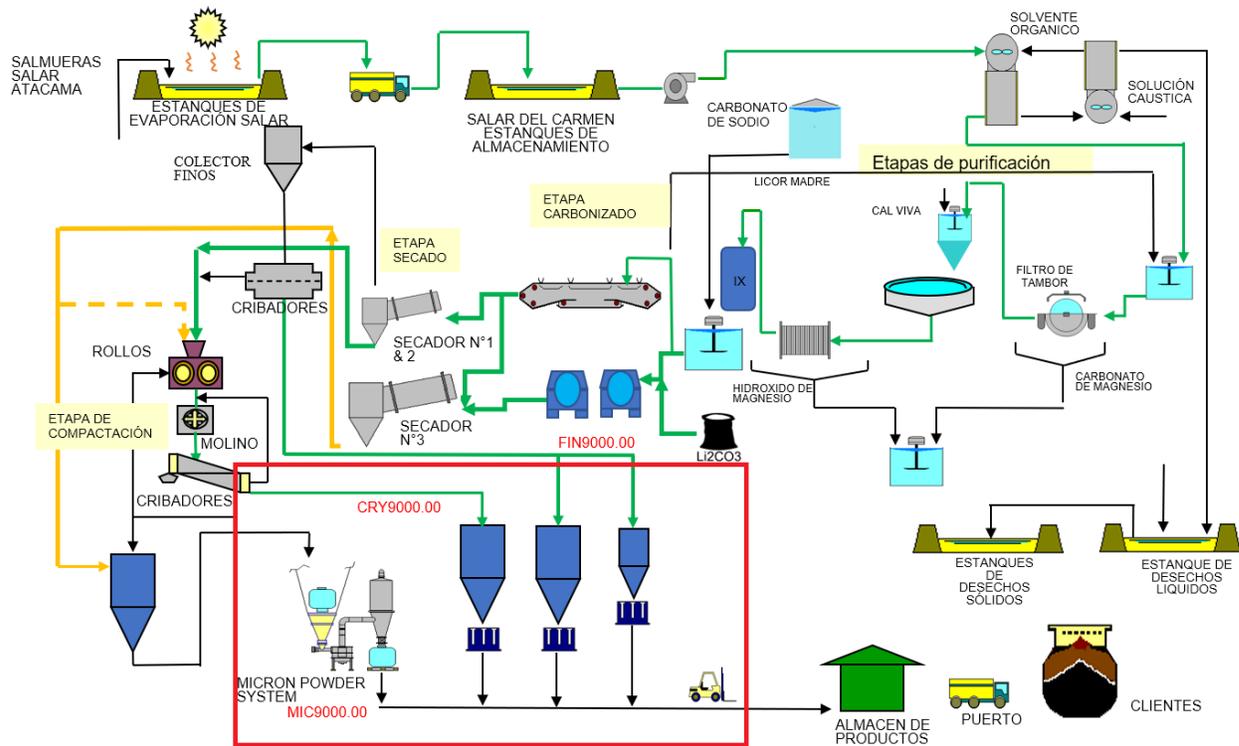


Figura 2.2: Producción de carbonato de litio. Fuente: SQM.

Las diferentes etapas secuenciales consideradas en el proceso de producción son resumidas en la Figura 2.3, y serán detalladas en lo que sigue.

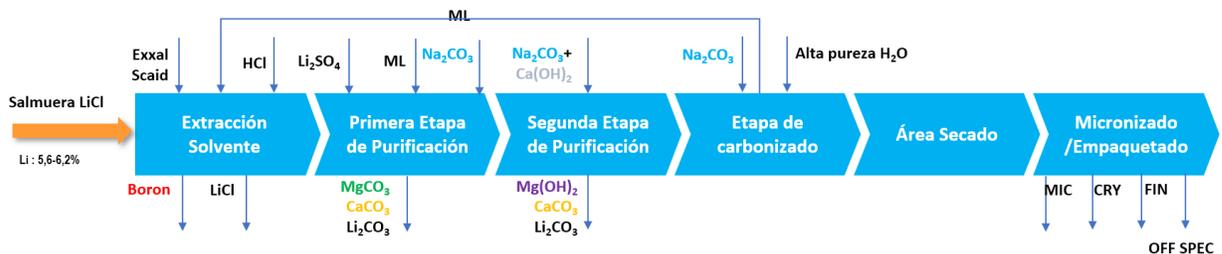


Figura 2.3: Etapas de producción Carbonato de Litio. Fuente: SQM.

- Extracción de solvente:** se procede a la eliminación del 99,5% del boro presente en la salmuera. Para ello, se utiliza una mezcla de compuestos orgánicos. Posteriormente, el licor madre<sup>2</sup> obtenido del proceso se emplea para reextraer el boro del extractador orgánico para poder reutilizarlo.

<sup>2</sup> El licor madre es la solución que queda después de que un componente se ha eliminado mediante algún proceso, como la filtración o, más comúnmente, la cristalización.

- **Primera etapa de purificación:** en esta etapa, se utiliza la salmuera refinada y el licor madre, el cual es obtenido de la mezcla en los reactores para precipitar carbonato de magnesio y carbonato de calcio. La solución de licor madre proporciona el ion  $\text{CO}_3$  necesario para precipitar estos compuestos. Luego, la pulpa se purifica en los filtros de tambor donde la cantidad de magnesio se reduce al 20 % de la concentración inicial.
- **Segunda etapa de purificación:** se procede a mezclar la lechada de cal<sup>3</sup> en los reactores con la salmuera filtrada obtenida en la primera etapa de purificación, con el fin de aumentar el pH de la solución y luego precipitar el resto de magnesio en forma de hidróxido de magnesio. Se utilizan dos decantadores para facilitar la separación de sólido - líquido. Posteriormente, la solución se vierte en los filtros prensa, para obtener una salmuera libre de impurezas, como magnesio y calcio.
- **Etapa de carbonizado:** la salmuera purificada se alimenta en los reactores con una solución de carbonato de sodio a 70-80 °C para precipitar el carbonato de litio. Luego, utilizando hidrociclones<sup>4</sup> y el espesante de esta etapa, se alimenta la pulpa con alta concentración de sólidos al equipo de filtrado y lavado, donde en la etapa de lavado, se elimina el 99,9 % de cloruro.
- **Área de secado:** el producto húmedo pasa a las secadoras que eliminan la humedad.
- **Área de envasado - micronizado (MIC):** posteriormente, dependiendo de los requerimientos comerciales que existan en el momento, el producto puede ser llevado a líneas de compactación en el área seca o a las plantas micronizadas para producir los productos de diferentes grados como MIC9000, CRY9000, entre otros.

En el presente trabajo se modela la última etapa descrita, la cual consta del proceso de micronizado - envasado (ver cuadro rojo en Figura 2.2), que en particular, ocurre en la planta número 4. A continuación se describe con un poco más de detalle el proceso de la Figura 2.4.

El producto de las compactadoras llamado pre-micronizado (PRE-MIC) es almacenado en silos para un posterior uso o pasa a las plantas de micronizado directamente. El material PRE-MIC se introduce en el molino gracias a la ayuda de la gravedad, luego se produce la reducción de tamaño gracias al impacto contra los elementos de molienda giratorios. Gracias a su sistema interno de separación, a medida que se logran partículas finas estas pasan a través de la rueda de clasificación y salen al exterior del molino, mientras que el material grueso regresa a la zona de molienda para una mayor trituración. Las partículas finas son arrastradas por la corriente hacia envasado, donde se tiene una válvula rotatoria que alimentan los maxisacos<sup>5</sup> y se detiene una vez alcanzado el peso requerido. El aire contaminado sigue su camino hacia el colector de polvo donde son filtradas las partículas contaminantes, tras lo cual el aire limpio es expulsado hacia la atmósfera.

<sup>3</sup> La lechada de cal es un fluido que está compuesto por una suspensión de hidróxido cálcico en agua.

<sup>4</sup> La función principal del Hidrociclón es separar los sólidos suspendidos en un determinado flujo de alimentación, en dos fracciones, una gruesa que acompaña al flujo llamado descarga (*underflow*) y otra fina que acompaña al flujo denominado rebose (*overflow*)

<sup>5</sup> El maxisaco es un contenedor flexible, que se utiliza para el envasado de diversos productos tales como: Minerales, cementos, harina, arroz, resinas, etc. En dosis que van desde 500 a 2.000 kilogramos.

Dependiendo la distribución de tamaño de partícula requerida, el molino puede ser ajustado para obtener dos tipos de carbonato de litio:

- **Estándar:** granulometría de 50 % > 6 $\mu$ m
- **Nichia:** granulometría de 50 % < 5.5 $\mu$ m

El producto resultante es envasado en maxisacos de tamaño relativo a requerimientos del cliente, tras lo cual los maxisacos son paletizados, envueltos y almacenados.

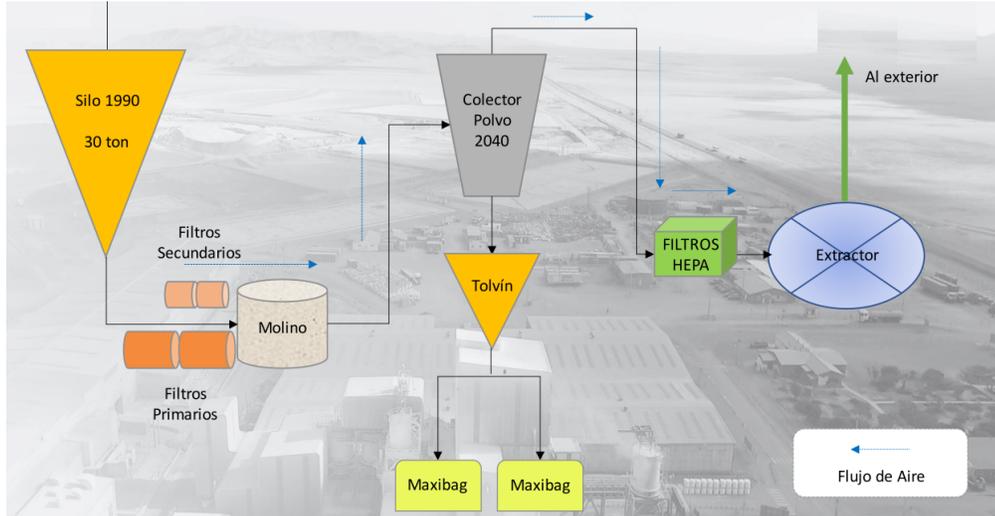


Figura 2.4: Proceso micronizado y envasado.

Las variables involucradas en el proceso se muestran en la tabla 2.1.

Tabla 2.1: Variables proceso micronizado y envasado.

Variable observación	Variable ajustable	Variable objetivo
Corriente separador [A]	Flujo aire soplador [SCFM]	Tasa envasado MIC 4870 [ton]
Corriente rotor [A]	Velocidad separador [RPM]	Tasa envasado MIC 4880 [ton]
Corriente soplador [A]		
Nivel silo 1990 [ton]		
Presión diferencial colector [mBar]		
Presión secundaria [mBar]		
Presión primaria [psi]		

## 2.2. Estado del arte

En las últimas décadas, la complejidad de los sistemas de ingeniería ha aumentado y se han requerido técnicas más sofisticadas para evaluar su confiabilidad e identificar con anticipación fallas para poder evitarlas. En consecuencia, el diagnóstico de fallos ha recibido una atención cada vez mayor [7]. Sin embargo, para interpretar correctamente los datos disponibles y construir modelos predictivos, el conocimiento de la composición del sistema y las interrelaciones entre los componentes es crucial para el desarrollo de modelos fácilmente interpretables.

A pesar de la proliferación de métodos de diagnóstico de fallas en las últimas décadas, se necesita un método capaz de analizar relaciones causales complejas entre componentes, y aún más importante, que pueda actualizarse durante la vida útil del sistema de acuerdo con la continua colección de datos.

Con el fin de explorar soluciones existentes para la modelación de sistemas por medio de redes bayesianas, se ha revisado el trabajo de Muller [8], el cual remarca la dificultad de controlar globalmente un sistema complejo de ingeniería, en cuanto a performance, debido a que están en continuo cambio y evolución por su carácter dinámico. Considerando estos antecedentes, Muller propuso un nuevo paradigma de modelación que permite anticiparse a desvíos de performance, el cual denominó Sistema Integrado de Mantenimiento Proactivo (ISPM). El autor da cuenta de un enfoque basado en el monitoreo, diagnosis, prognosis y módulos de decisión. En particular, propone el uso de redes bayesianas dinámicas (DBN) para la modelación de sistemas dinámicos complejos, por sus propiedades de explicabilidad y la posibilidad de adición de conocimiento experto.

También se ha considerado el trabajo de Zhang [9], que propone el uso de una red bayesiana dinámica con una mezcla de gaussianas como un identificador de fallas. El autor remarca con gran énfasis que las relaciones temporales entre estados en un proceso dinámico no pueden ser capturadas a través de redes bayesianas estáticas.

Por último, se ha tenido en cuenta el trabajo de Lewin [10] que hace uso de un modelo DBN para diagnósticos y predicción del estado de un sistema complejo de ingeniería, a través del uso de datos provenientes de sensores. Estudia experimentalmente el uso del estado actual y su pronóstico como ayuda para la toma de mejores decisiones en tareas de operación y mantenimiento.

Considerando los trabajos previos, se destaca como una idea válida la modelación de sistemas mediante redes bayesianas. En particular, para el objetivo de este trabajo, la modelación mediante DBN permite la representación de relaciones temporales complejas a través del uso de probabilidades condicionales y modelos de gráficos acíclicos dirigidos. Además, permiten la inferencia hacia adelante y hacia atrás de los estados del sistema, diagnosticando su salud actual, y pronosticando estados futuros dentro del mismo marco de modelado.

# Capítulo 3

## Marco teórico

### 3.1. Definiciones

A continuación se definen conceptos útiles para la lectura de este trabajo [11].

**Definición 3.1** *Un grafo  $G$  se define como un par ordenado  $\langle V, E \rangle$ , donde  $V$  es un set finito - no vacío de nodos y  $E$  es un set de enlaces entre nodos donde  $E \subseteq \{(u, v) \mid u, v \in V\}$ .*

**Definición 3.2** *Se dice que un grafo  $G$  es dirigido cuando la relación  $E$  satisface:  $\exists (u, v) \in E \mid (v, u) \notin E$ .*

**Definición 3.3** *Un grafo  $G$  ponderado o weighed se define como una tripleta  $G = \langle V, E, W \rangle$ , donde  $V$  y  $E$  son sets de nodos y enlaces, respectivamente, y  $W$  es una matriz que contiene los pesos de los enlaces.*

**Definición 3.4** *Un grafo  $G$  se dice dirigido acíclico si no contiene ciclos dirigidos, es decir, si para cada vértice  $v$ , no hay un camino de enlaces dirigidos que comience y termine en  $v$ .*

A modo de ilustración de las definiciones anteriores, en la Figura 3.1 se muestra un grafo dirigido y ponderado. Dependiendo del contexto, los pesos pueden representar distancia, flujo de tráfico, entre otros. En particular, información tabular de  $n$  variables puede ser visualizada como un grafo ponderado con  $n$  nodos, donde los pesos están dados por las correlaciones entre las variables.

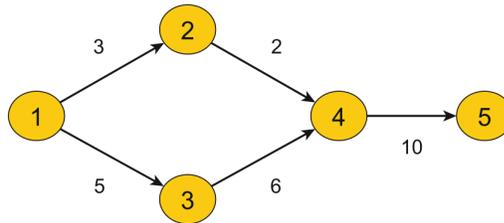


Figura 3.1: Grafo dirigido y ponderado.

**Definición 3.5** *Sean  $X, Y$  variables aleatorias, si la probabilidad de  $Y$  es diferente de 0, la probabilidad condicional de  $X$  se define como:*

$$P(X|Y) = \frac{P(X,Y)}{P(Y)} \quad (3.1)$$

**Definición 3.6** Sean  $X, Y$  variables aleatorias, el teorema de Bayes determina la probabilidad condicional de  $X$  dado  $Y$  basado en las probabilidades de  $X, Y$  e  $Y$  dado  $X$ :

$$P(X|Y) = \frac{P(Y|X)P(X)}{P(Y)} \quad (3.2)$$

**Definición 3.7** Sean  $X, Y$  variables aleatorias, se dice que  $X$  e  $Y$  son independientes si y solo si  $P(X|Y) = P(X)$

**Definición 3.8** Sean  $X, Y$  y  $Z$  variables aleatorias, se dice que  $X$  e  $Y$  son condicionalmente independientes dado  $Z$  si y solo si  $P(X|Y, Z) = P(X|Z)$

## 3.2. Modelos espacio estado

Dada una serie de tiempo  $(Y_t)_{t \geq 1}$ , es necesario definir la distribución de probabilidad conjunta para  $(Y_1, \dots, Y_t)$  para cualquier  $t \geq 1$ , lo cual no es trivial. Si bien asumir independencia implica declarar que la evolución temporal es irrelevante, en general se asume que existe una dependencia de carácter markoviano para  $Y$  en el cual el tiempo tiene influencia sobre la evolución de un proceso, pero de manera limitada. Se dice que  $(Y_t)_{t \geq 1}$  es una cadena de Markov si para cualquier  $t > 1$  se cumple que [12]:

$$\pi(y_t|y_{t-1}) = \pi(y_t|y_{t-1}) \quad (3.3)$$

Lo cual implica que la información acerca de  $Y_t$  que está contenida por todas las observaciones hasta  $t - 1$  es exactamente la misma que contiene  $y_{t-1}$  por sí misma. Dicho de otra manera,  $Y_t$  y  $Y_{1:t-2}$  son condicionalmente independientes. Una vez que se asume la propiedad markoviana, es posible definir la distribución conjunta como sigue:

$$\pi(y_{1:t}) = \pi(y_1) \cdot \prod_{j=2}^t \pi(y_j|y_{j-1}) \quad (3.4)$$

Un modelo espacio estado asume que existe una cadena de Markov  $(\theta_t)$  llamada proceso de estado, y que  $Y_t$  es una medida imprecisa de  $\theta_t$ . En el contexto ingenieril,  $\theta_t$  hace referencia al estado físico de un sistema observable que produce como salida  $Y_t$ .

De manera formal, un modelo espacio estado consiste en una serie de tiempo  $\theta_t : t = 0, 1, \dots, p$  y otra  $Y_t : t = 1, 2, \dots, m$  que satisface los siguientes supuestos:

- $\theta_t$  es una cadena de Markov
- Condicionalmente en  $\theta$ , los  $Y_t$ 's son independientes e  $Y_t$  depende solo de  $\theta_t$

Como consecuencia de estos dos supuestos, el modelo de espacio estado queda completamente definido por su distribución inicial  $\pi(\theta_0)$  y las densidades de probabilidad condicional  $\pi(\theta_t|\theta_{t-1})$  y  $\pi(y_t|\theta_t)$ .  $t > 1$ :

$$\pi(\theta_{0:t}, y_{1:t}) = \pi(\theta_0) \cdot \prod_{j=1}^t \pi(\theta_j|\theta_{j-1})\pi(y_j|\theta_j) \quad (3.5)$$

Una manera gráfica de entender el flujo de información asumido por un modelo de espacio estado puede ser observado en la Figura 3.2, la cual representa un grafo acíclico dirigido, y deja de manifiesto las propiedades de independencia condicional de las variables aleatorias involucradas.

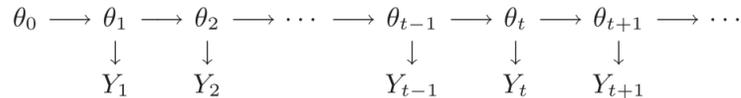


Figura 3.2: Representación estructura de dependencia espacio estado.

Otra forma de definir el modelo espacio estado, de acuerdo a Murphy [13], requiere un prior

$P(X_1)$ , una función de estado-transición  $P(X_t|X_{t-1})$  y una función de observación  $P(Y_t|X_t)$ . Además, en un ambiente controlado, la última expresión se convierte en  $P(X_t|X_{t-1}, U_t)$  y  $P(Y_t|X_t, U_t)$ , donde  $U$  representa las variables de entrada/control. Ejemplos típicos de modelos espacio estado son: cadenas ocultas de Markov (HMM), filtro de Kalman (KFM) y redes bayesianas dinámicas (DBN).

A continuación, se mencionan brevemente algunos de los tipos de inferencia que se realizan sobre modelos de espacio estado, bajo la definición de Murphy.

- **Filtrado:** tarea que tiene como objetivo filtrar el ruido observacional a través de una estimación recursiva de los estados por medio de la regla de Bayes:

$$P(X_t|y_{1:t}) \propto P(y_t|X_t, y_{1:t-1})P(X_t|y_{1:t-1}) \quad (3.6)$$

$$= P(y_t|X_t) \left[ \sum_{x_{t-1}} P(X_t|x_{t-1})P(x_{t-1}|y_{1:t-1}) \right] \quad (3.7)$$

Ahora bien, si se asume propiedad markoviana de primer orden:  $P(y_t|X_t, y_{1:t-1}) = P(y_t|X_t)$ , el proceso recursivo de estimación de dos etapas queda como sigue:

1. Predecir: computar  $P(X_t|y_{1:t-1})$
2. Actualizar: computar  $P(X_t|y_{1:t})$

- **Suavizado:** tarea también conocida como *smoothing*, estima los estados del pasado, usando como evidencia todos los datos hasta el tiempo actual. Por lo tanto, el objetivo consiste en estimar  $P(X_{t-l}|y_{1:t})$ , donde  $l > 0$  es el *lag* a considerar.
- **Predicción:** tarea que busca estimar futuras observaciones en función del presente e instantes anteriores. Si se denota  $h > 0$  como el largo o pasos de predicción, la estimación del estado futuro queda denotada como:  $P(X_{t+h}|y_{1:t})$ . Tras la estimación del estado  $X_{t+h}$ , es posible obtener la predicción para el valor observado marginalizando  $X_{t+h}$ :

$$P(Y_{t+h} = y|y_{1:t}) = \sum_x P(Y_{t+h} = y|X_{t+h} = x)P(X_{t+h} = x|y_{1:t}) \quad (3.8)$$

- **Decoding:** el problema de decodificación consiste en el cálculo de la secuencia de estado más probable a partir de la observación:

$$x_{1:t}^* = \operatorname{argmax}_{x_{1:t}} P(x_{1:t}|y_{1:t}) \quad (3.9)$$

### 3.3. Modelos dinámicos lineales

Un modelo dinámico lineal (DLM) es un modelo espacio estado de carácter lineal gaussiano [14]. De manera formal, se define por una distribución prior de clase normal dado por un vector de estados  $p$ -dimensional a tiempo  $t = 0$ ,

$$\theta_0 \sim \mathcal{N}_p(m_0, C_0) \quad (3.10)$$

en conjunto con el par de ecuaciones para  $t > 1$ :

$$Y_t = F_t \theta_t + v_t, \quad v_t \sim \mathcal{N}_m(0, V_t) \quad (3.11)$$

$$\theta_t = G_t \theta_{t-1} + w_t, \quad w_t \sim \mathcal{N}_p(0, W_t) \quad (3.12)$$

donde  $G_t$  es de orden  $p \times p$  y  $F_t$  es orden  $m \times p$ , ambas conocidas. Además,  $(v_t)_{t \geq 1}$  y  $(w_t)_{t \geq 1}$  son dos vectores aleatorios independientes gaussianos con mediana 0 y matrices de varianza conocidas, las cuales se definen por  $(V_t)_{t \geq 1}$  y  $(W_t)_{t \geq 1}$  respectivamente. La ecuación (3.11) es llamada ecuación de observación, mientras que (3.12) se conoce como ecuación de estado.

#### 3.3.1. Modelo local lineal

Uno de los casos más simples de sistemas lineales dinámicos es el caso de una serie de tiempo univariada  $Y_t$  modelada a través de *random walk* [15] sumado a ruido, definido por:

$$\begin{aligned} Y_t &= \mu_t + v_t, \quad v_t \sim \mathcal{N}_p(0, V) \\ \mu_t &= \mu_{t-1} + w_t, \quad w_t \sim \mathcal{N}_p(0, W) \end{aligned} \quad (3.13)$$

donde  $v_t$  y  $w_t$  son los errores independientes. El sistema queda definido de forma matricial con  $m = p = 1$ ,  $\theta_t = \mu_t$  y  $F_t = G_t = 1$ , en el que *random walk*  $\mu_t$  no es estacionario.

Cabe destacar que  $W = 0$  tiene un modelo constante con valor igual a la media aritmética. De esta manera, las observaciones son modeladas como observaciones ruidosas con *level*  $\mu_t$  sujeta a cambios aleatorios en el tiempo, modeladas por *random walk*<sup>1</sup>. Intuitivamente este modelo resulta apropiado para series de tiempo sin una tendencia o estacionalidad clara.

#### 3.3.2. Modelo local lineal con tendencia

Un modelo local lineal con tendencia (LLT) considera una pendiente variante en el tiempo para  $\mu_t$ , el modelo se define [16]:

$$\begin{aligned} Y_t &= \mu_t + v_t, \quad v_t \sim \mathcal{N}_p(0, V) \\ \mu_t &= \mu_{t-1} + \beta_{t-1} + w_{t,1}, \quad w_{t,1} \sim \mathcal{N}_p(0, \sigma_\mu^2) \\ \beta_t &= \beta_{t-1} + w_{t,2}, \quad w_{t,2} \sim \mathcal{N}_p(0, \sigma_\beta^2) \end{aligned} \quad (3.14)$$

con errores no correlaciones  $v_t$ ,  $w_{t,1}$  y  $w_{t,2}$ . Lo que puede ser definido mediante las matrices:

<sup>1</sup> fenómeno estadístico en el que una variable no sigue una tendencia discernible y se mueve aparentemente de forma aleatoria.

$$\theta_t = \begin{bmatrix} \mu_t \\ \beta_t \end{bmatrix}$$

$$G = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$

$$W = \begin{bmatrix} \sigma_\mu^2 & 0 \\ 0 & \sigma_\beta^2 \end{bmatrix}$$

$$F = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

Donde las varianzas del sistema  $\sigma_\mu^2$  y  $\sigma_\beta^2$  pueden ser nulas. Se dice que el modelo es invariante en el tiempo en el caso de que las matrices  $G_t$ ,  $F_t$ ,  $V_t$  y  $W_t$  sean constantes.

### 3.3.3. Modelo autorregresivo de media móvil

Un modelo ARMA es definido por un par de enteros no negativos  $p$  y  $q$ , en particular, y en el caso univariado, un modelo ARMA  $(p, q)$  se define por la siguiente relación [17]:

$$Y_t = \mu + \sum_{j=1}^p (\phi_j Y_{t-j} - \mu) + \sum_{j=1}^q \psi_j \epsilon_{t-j} + \epsilon_t \quad (3.15)$$

donde  $\epsilon_t$  corresponde a ruido blanco gaussiano con varianza  $\sigma_e^2$  y los parámetros  $\phi_1, \dots, \phi_p$  satisfacen condición de estacionalidad. En el caso de que los datos no cumplan con esta última condición, se procede a diferenciarlos hasta conseguirlo y luego se ajusta un modelo ARMA. De esta forma, se define un modelo autorregresivo integrado de media móvil ARIMA  $(p, d, q)$ , donde  $d$  indica el orden de diferenciación.

La manera de elegir el par  $(p, q)$  típicamente se lleva a cabo bajo el criterio de Información Akaike (AIC), el cual estima la cantidad mínima de bits necesaria para describir las secuencias [18]. Por lo tanto, aquel modelo elegido es el que menor AIC obtiene. Matemáticamente, AIC se calcula mediante la siguiente expresión:

$$\text{AIC} = 2k - 2 \ln \hat{L} \quad (3.16)$$

donde  $k$  indica el número total de parámetros estimados y  $\hat{L}$  es el valor máximo de la función verosimilitud del modelo.

Un modelo ARMA puede ser representado como un modelo espacio estado, tal como se describe a continuación. Se considera un proceso ARMA  $(p, q)$  con media  $\mu = 0$ , por simplicidad, definido por:

$$Y_t = \sum_{j=1}^r \phi_j Y_{t-j} + \sum_{j=1}^{r-1} \psi_j \epsilon_{t-j} + \epsilon_t \quad (3.17)$$

con  $r = \max\{p, q + 1\}$ ,  $\phi_j = 0$  para  $j > p$  y  $\psi_j = 0$  para  $j > q$ , se definen las siguientes matrices:

$$\begin{aligned}
F &= \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix} \\
G &= \begin{bmatrix} \phi_1 & 1 & 0 & \dots & \dots & \dots & 0 \\ \phi_2 & 0 & 1 & \dots & \dots & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \phi_{r-1} & 0 & 0 & \dots & 0 & 1 \\ \phi_r & 0 & 0 & \dots & \dots & \dots & 0 \end{bmatrix} \\
R &= \begin{bmatrix} 1 & \psi_1, \dots, \psi_{r-2} & \psi_{r-1} \end{bmatrix}'
\end{aligned} \tag{3.18}$$

Si se define un vector estado  $r$ -dimensional  $\theta = (\theta_{1,t}, \dots, \theta_{r,t})'$ , entonces el modelo equivalente dinámico lineal, equivalente a ARMA, queda del siguiente modo:

$$\begin{aligned}
Y_t &= F\theta_t \\
\theta_{t+1} &= G\theta_t + R\epsilon_t
\end{aligned} \tag{3.19}$$

Donde  $V = 0$  y  $W = RR'\sigma^2$ , con  $\sigma^2$  como la varianza de la secuencia de errores  $\epsilon_t$ .

Tal como se plantea en [12], si bien es posible encontrar una representación de modelos ARMA como espacio estado, en realidad son de naturalezas distintas. Es así como un modelo ARMA proporciona un enfoque de caja negra para el análisis de datos, ofreciendo la posibilidad de pronosticar observaciones futuras, pero con una interpretación limitada del modelo ajustado. Mientras que en el marco DLM se alienta al analista a pensar en términos de procesos fácilmente interpretables, aunque no observables, tales como tendencia y componentes estacionales, que explican la serie de tiempo observada.

### 3.4. Filtro de Kalman

Dado un modelo espacio estado, las principales tareas radican en realizar inferencia sobre estados no observables, o predecir futuras observaciones basadas en una secuencia de observaciones previas.

Para la estimación del vector de estados, se deben computar las densidades condicionales  $\pi(\theta_s|y_{1:t})$ , tarea en la cual se tiene un problema de filtrado cuando  $s = t$ , predicción de estado  $s > t$  y suavizado  $s < t$ . Vale la pena clarificar que el problema de filtrado asume que los datos son recibidos de manera secuencial, y por lo tanto, la tarea radica en estimar el valor actual del vector de estado con observaciones hasta tiempo  $t$ , actualizar la inferencia actual del vector de estados y hacer un pronóstico para  $t + 1$ . De esto se traduce que a partir de densidad  $\pi(\theta_t|y_{1:t})$  se desee obtener  $\pi(\theta_{t+1}|y_{1:t+1})$ .

Debido a las propiedades markovianas de los modelos espacio estado y de las independencias de las observaciones, el problema de filtrado puede ser computado a través de un algoritmo recursivo, y en general de la siguiente manera [12]:

1. Se calcula la predicción a un paso de la distribución para  $\theta_t$  dado  $y_{1:t-1}$  basado en la densidad  $\pi(\theta_t|y_{1:t-1})$  y la distribución condicional de  $\theta_t$  dado  $\theta_{t-1}$ .
2. Se computa la predicción a un paso de la distribución para la siguiente observación.

3. Se computa la distribución de filtrado  $\pi(\theta_t|y_{1:t})$  usando la regla de Bayes con  $\pi(\theta_t|y_{1:t-1})$  como distribución prior y la verosimilitud  $\pi(y_t|\theta_t)$

Ahora bien, considerando el sistema lineal dinámico descrito por las ecuaciones (3.10), (3.11) y (3.12), y dado:

$$\theta_{t-1}|y_{1:t-1} \sim \mathcal{N}(m_{t-1}, C_{t-1}) \quad (3.20)$$

El algoritmo de filtro de Kalman, bajos los supuestos considerados, permite computar la distribución predictiva y filtrada recursivamente partiendo desde 3.20, luego computando  $\pi(\theta_1|y_1)$  y así sucesivamente a medida que más data esté disponible. A continuación, se describen de manera detallada los pasos del algoritmo.

1. Computar la predicción a un paso de la distribución de  $\theta_t$  dado  $y_{1:t-1}$ , la cual es gaussiana con parámetros:

$$\begin{aligned} a_t &= E(\theta_t|y_{1:t-1}) = G_t m_{t-1} \\ R_t &= Var(\theta_t|y_{1:t-1}) = G_t C_{t-1} G_t' + W_t \end{aligned} \quad (3.21)$$

2. Computar la predicción a un paso de la distribución de  $Y_t$  dado  $y_{1:t-1}$ , la cual es gaussiana con parámetros:

$$\begin{aligned} f_t &= E(Y_t|y_{1:t-1}) = F_t a_t \\ Q_t &= Var(Y_t|y_{1:t-1}) = F_t R_t F_t' + V_t \end{aligned} \quad (3.22)$$

3. Computar la distribución filtrada de  $\theta_t$  dado  $y_t$ , la cual es gaussiana con parámetros:

$$\begin{aligned} m_t &= E(\theta_t|y_{1:t}) = a_t + R_t F_t' Q_t^{-1} e_t \\ C_t &= Var(\theta_t|y_{1:t}) = R_t - R_t F_t' Q_t^{-1} F_t R_t \end{aligned} \quad (3.23)$$

Se define  $e_t = Y_t - f_t$  como el error de predicción o *forecasting*. Además, típicamente la expresión  $K_t = R_t F_t' Q_t^{-1}$  recibe el nombre de matriz de ganancia y determina el factor de corrección.

## 3.5. Modelos de volatilidad de tipo GARCH

Un modelo GARCH, o modelo autorregresivo generalizado condicional heterocedástico, es clasificado como un modelo de volatilidad condicional, y se basa en la optimización exponencial de pesos de valores previos, para la posterior predicción de la volatilidad [18].

Considerar  $r_t = \mu + \epsilon_t$ , donde  $\mu$  es el retorno esperado y  $\epsilon_t$  representa el error respecto a la media  $\mu$  con  $\epsilon_t = \sigma_t z_t$ , con  $z_t$  una sucesión de variables aleatorias independientes e idénticamente distribuidas con media 0 y varianza 1, ante lo cual la varianza queda determinada por [19]:

$$\sigma_t^2 = \omega + \sum_{i=1}^p \beta_i \sigma_{t-i}^2 + \sum_{j=1}^q \alpha_j \epsilon_{t-j}^2 \quad (3.24)$$

Ahora bien, de forma alternativa, se puede definir  $\sigma_t^2 = \epsilon_t^2 - \eta_t$  y por consiguiente  $\sigma_{t-1}^2 = \epsilon_{t-1}^2 - \eta_t$ , con lo cual reemplazando en (3.24) se obtiene la siguiente expresión:

$$\sigma_t^2 = \omega - \sum_{i=1}^p \beta_i \eta_{t-i}^2 + \sum_{j=1}^{\max(p,q)} (\alpha_j + \beta_j) \epsilon_{t-j}^2 + \eta_t \quad (3.25)$$

la ecuación resultante tiene la forma de un modelo ARMA para  $\sigma_t^2$ . Por lo que se dice que un modelo GARCH puede ser interpretado como un modelo ARMA para la serie  $\sigma_t^2$ .

En el caso de que  $e_t$  no sea independiente, se puede definir un modelo más generalizado llamado apARCH, el cual asume una forma paramétrica de la potencia de la condición heterocedástica para considerar más *lags* que el caso de solo volatilidad condicional [20]. De esta manera, un modelo apARCH (p, q) se define como:

$$\sigma_t^\delta = \omega + \sum_{i=1}^p \alpha_i (|e_{t-i}| - \gamma_i \epsilon_{t-i})^\delta + \sum_{j=1}^q \beta_j \sigma_{t-j}^\delta \quad (3.26)$$

## 3.6. Red bayesiana

Una red bayesiana ofrece una intuitiva representación gráfica de relaciones entre variables aleatorias en el dominio de sus dependencias condicionales, a través de un grafo acíclico dirigido. Formalmente, de acuerdo con [21] una red bayesiana para  $U$  es un par  $B = \langle G, \Theta \rangle$ . Aquí, la primera componente  $G$  es un grafo dirigido acíclico (DAG), donde los enlaces representan las dependencias directas entre las variables.  $G$  asume que cada variable  $X_i$  es independiente de sus nodos no descendientes dados sus padres en  $G$ . La segunda componente  $\Theta$  representa el set de parámetros que cuantifica la red, contiene parámetros de la forma  $\theta_{x_i|\Pi_{x_i}} = P_B(x_i|\Pi_{x_i})$  para cada posible valor  $x_i$  de  $X_i$ , y  $\Pi_{x_i}$  de  $\Pi_{X_i}$ , donde  $\Pi_{X_i}$  denota los padres de  $X_i$  en  $G$ . De esta manera se define una única distribución de probabilidad conjunta sobre  $B$ :

$$P_B(X_1, \dots, X_n) = \prod_i^n P_B(X_i|\Pi_{X_i}) = \prod_i^n \theta_{X_i|\Pi_{X_i}} \quad (3.27)$$

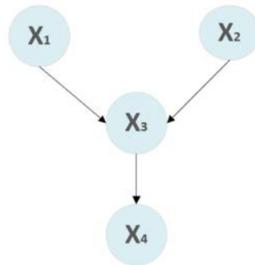


Figura 3.3: Ejemplo Red Bayesiana 4 Nodos. Imagen obtenida desde [22]

A modo de ilustración, en la Figura 3.3 se puede observar que la variable  $X_4$  depende de  $X_3$ , que a su vez obedece a  $X_1$  y  $X_2$ . Usando la regla de la cadena es posible obtener la distribución conjunta como:

$$P(X_1, X_2, X_3, X_4) = P(X_4|X_3)P(X_3|X_1, X_2)P(X_2)P(X_1) \quad (3.28)$$

Esencialmente, el proceso de aprendizaje de una red bayesiana consta de dos etapas: aprendizaje de estructura y estimación de parámetros.

### 3.7. Red bayesiana gaussiana dinámica

Una red bayesiana, tal como fue descrita en la sección 3.6, representa el estado de un fenómeno en un instante puntual de tiempo. Sin embargo, en el ámbito de sistemas de ingeniería resulta mucho más interesante representar la evolución temporal de procesos, en otras palabras, el estudio de la evolución de las variables de interés en el tiempo. Una manera de representar esta evolución temporal es por medio de modelos espacio estado a intervalos de tiempo discreto; en particular, mediante la red bayesiana dinámica, que contiene ventanas de tiempo y modela el valor de cada variable para todo instante tiempo  $t$ .

Una red bayesiana dinámica (DBN) es una extensión de una BN que permite el modelamiento de procesos temporales [23]. A su vez, una DGBN es una DBN donde se asumen los siguientes supuestos:

- Cada nodo o variable sigue una distribución normal.
- Un nodo raíz es descrito por su respectiva distribución marginal.
- Cada nodo tiene asociada una varianza específica, es decir, no depende de las varianzas de los padres. El efecto condicional solo está dado por la aditividad lineal de las medias.
- La distribución local de cada nodo puede ser equivalentemente expresada como un modelo lineal gaussiano donde los padres proveen las variables explicativas.

Típicamente  $n$  instantes de tiempo son representados mediante  $n$  columnas o *time slices*, donde  $n - 1$  indica el orden markoviano del proceso. Además, dentro de cada *time slice* existen nodos que representan las variables, tal como es mostrado en la Figura 3.4.

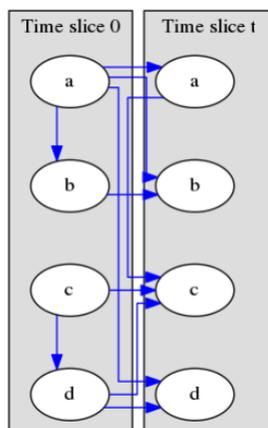


Figura 3.4: Red bayesiana dinámica de primer orden. Fuente: pyAgrum.

De manera similar a una BN, una DBN es un par  $B = (G, \Theta)$ , con  $G$  como red de transición, donde los arcos conectados denotan dependencia condicional directa, mientras que la ausencia representa independencia condicional. Cada nodo en la segunda columna  $X_i (i =$

$1, \dots, n$ ) contiene  $\theta_{X_i|pa(X_i)} = P(X_i|pa(X_i))$  que denota la distribución condicional de  $X_i$  dados sus nodos padres  $pa(X_i)$ . De esta manera, las independencias y dependencias condicionales codificadas en la DBN, son representadas como la distribución conjunta  $P_B(X_1, X_2, \dots, X_n)$ . Sin embargo, en el caso de una red bayesiana gaussiana dinámica (DGBN) se asume que cada variable aleatoria sigue una distribución gaussiana condicional que es linealmente dependiente de sus padres [24]:

$$P(X|Pa(X)) \sim N\left(\mu_X + \sum_{\Pi \in Pa(X)} \beta_{X,\Pi}(\pi - \mu_\Pi); \sigma_X^2\right) \quad (3.29)$$

Donde  $\mu_X$  y  $\mu_\Pi$  son las medias no condicionales de  $X$  y  $\Pi$ ,  $\pi$  es la realización de  $\Pi$ ,  $\sigma_X^2$  es la varianza de  $X$  y  $\beta_{X,\Pi}$  representa la influencia de los padres  $\Pi$  sobre sus hijos  $X$ .

Considerando que una DBN contiene  $N$  variables en cada *time slice* y su estructura puede ser arbitraria, es posible mencionar algunos casos particulares:

- **Cadenas de Markov:** es el caso más simple de una DBN. Consiste en una única variable por *time slice*:  $X_t$  influenciada solo por  $X_{t-1}$ .
- **Cadenas ocultas de Markov:** contiene dos variables por cada *time slice*: variable de estado  $S$ , y la variable de observación  $Y$ . Donde  $S_t$  solo depende de  $S_{t-1}$  e  $Y_t$  solo depende de  $S_t$ .
- **Filtro de Kalman:** cada variable tiene una observación y un estado, ambos continuos.

### 3.7.1. Aprendizaje de la estructura

El aprendizaje de estructura de una red bayesiana consiste en encontrar la mejor DAG que se ajuste a los datos. Existen variados métodos, que se resumen en dos clases:

1. **Métodos globales:** consiste en la búsqueda sobre un espacio de estructuras. Se comienza de una estructura inicial y se generan variaciones paso a paso. La mejor estructura es elegida con base en alguna función *score*, por ejemplo: *Bayesian information criterion* (BIC) o *Minimum Description Length* (MDL) [25].
2. **Métodos locales:** la estructura es descubierta secuencialmente por medio de tests de independencia condicional.

Adicionalmente, en el caso de contar con conocimiento experto, este puede ser plasmado como arcos prohibidos u obligatorios, que den cuenta de dependencia condicional [26]. En el caso de una DBN se deben aprender dos DAG:

- **Conectividad intra-temporal:** debe ser representada mediante un grafo acíclico dirigido.
- **Conectividad inter-temporal:** para cada nodo en el sector  $t$ , se debe elegir sus padres de la sección  $t-1$ .

A continuación se describen brevemente dos métodos de aprendizaje:

1. **Dynamic Max Min Hill Climbing (DMMHC)**: propuesto por Trabelsi en [27], hace uso de aprendizaje de estructura con métodos de búsqueda local, adaptando el algoritmo MMHC a DBN. MMHC usa el test de independencia condicional para encontrar candidatos de arcos padres-hijos y así reducir el espacio de búsqueda, tras lo cual se usa *Hill-Climbing* para puntuar cada una de las estructuras posibles.
2. **Particle swarm optimization algorithm for higher-order DBN**: Santos [28] propone una técnica de optimización basada en el enjambre de partículas para aprender las estructuras de transición de ho-DBN, tratando de explotar sus características intrínsecas para sugerir representaciones y simplificaciones eficientes en su aprendizaje.

### 3.7.2. Aprendizaje de parámetros

Como fue mencionado anteriormente, un modelo espacio estado tiene parámetros libres  $\Theta$  que definen el modelo de transición  $P(X_t|X_{t-1})$  y el modelo de observación  $P(Y_t|X_t)$ . El aprendizaje o estimación de estos parámetros  $\Theta$  deben ser aprendidos desde los datos, típicamente se usa el criterio *maximum-likelihood* (MLE) para este propósito. Por lo cual, en términos generales, el objetivo es resolver el siguiente problema de maximización:

$$\theta_{ML}^* = \underset{\theta}{\operatorname{argmax}} \log P(Y|\theta) \quad (3.30)$$

En otras palabras, se deben encontrar los mejores parámetros  $\theta$  dadas las observaciones  $Y$ . Ya que en las DGBN se tiene que su distribución global es normal multivariante, y las distribuciones locales son variables aleatorias normales unidas por restricciones lineales, se explicará el proceso de estimación en la siguiente subsección.

#### 3.7.2.1. MLE en Gaussiana multivariable

Dada una distribución gaussiana multivariable de un vector  $x \in \mathbb{R}^d$ , se define su función densidad (PDF) [29]:

$$p(X, \mu, \Sigma) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp \left\{ -\frac{1}{2} (X - \mu)^T \Sigma^{-1} (X - \mu) \right\} \quad (3.31)$$

El *loglikelihood* de un set  $X$  de  $n$  vectores i.d.d está dado por:

$$l(X, \mu, \Sigma) = -\frac{n}{2} \log |\Sigma| - \frac{1}{2} \sum_{i=1}^n (x_i - \mu)^T \Sigma^{-1} (x_i - \mu) \quad (3.32)$$

Para encontrar MLE de  $\mu$ , se debe derivar con respecto a  $\mu$  e igualar a 0:

$$\frac{\partial}{\partial \mu} = \left( -\frac{1}{2} \sum_{i=1}^n (-2\mu^T \Sigma^{-1} x_i + \mu^T \Sigma^{-1} \mu) \right) = 0 \quad (3.33)$$

Tras lo cual se obtiene que la estimación  $\mu_{MLE}$  está dada por:

$$\hat{\mu}_{MLE} = \frac{1}{n} \sum_{i=1}^n x_i \quad (3.34)$$

Mientras tanto, la estimación MLE de  $\Sigma$  se obtiene de manera similar al procedimiento mencionado y queda como sigue:

$$\hat{\Sigma}_{MLE} = \frac{1}{n} \sum_{i=1}^n (x_i - \mu)(x_i - \mu)^T \quad (3.35)$$

En esta última ecuación  $\mu$  hace referencia a  $\hat{\mu}_{MLE}$ , donde se observa el proceso general de estimación de parámetros de cada nodo, en que la media y la varianza son estimadas con base en los datos observados.

### 3.7.3. Inferencia

La inferencia probabilística se define como la propagación de evidencia en una red bayesiana y la estimación de sus efectos en variables desconocidas. En otras palabras, usando un subset de valores se obtienen las probabilidades posteriores de las variables restantes. En el caso de una red bayesiana gaussiana, la inferencia es posible de realizar bajo el procedimiento descrito en [30], el cual se describe a continuación.

Se sabe que existe una relación lineal entre cada variable y sus nodos padres:

$$X = b_1 U_1 + b_2 U_2 + \dots + b_n U_n + W_X \quad (3.36)$$

Donde  $U_i$  son los padres de la variable  $X$ ,  $b_i$  son coeficientes constantes y  $W_x$  representa ruidos gaussianos como media 0.

Ahora bien, el proceso de inferencia consiste en la propagación de medias y desviaciones estándares. Dado que en el caso de distribuciones gaussianas la distribución marginal de todas las variables también son gaussianas, se tiene que la probabilidad posterior de una variable puede ser expresada como sigue:

$$P(X|E) = N(\mu_X, \sigma_X) \quad (3.37)$$

Donde  $\mu_X$  y  $\sigma_X$  son el promedio y la desviación estándar de  $X$  dada la evidencia  $E$

A continuación se describe el algoritmo de propagación. Cada variable o nodo envía a sus padres  $i$ :

$$\mu_i^- = \frac{1}{b_i} \left[ \mu_\lambda - \sum_{k \neq i} b_k \mu_k^+ \right] \quad (3.38)$$

$$\sigma_i^- = \frac{1}{b_i^2} \left[ \sigma_\lambda - \sum_{k \neq i} b_k^2 \sigma_k^+ \right] \quad (3.39)$$

A su vez, cada variable envía a sus hijos  $j$ :

$$\mu_j^+ = \frac{\sum_{k \neq j} [\mu_k^- / \sigma_k + \mu_\pi / \sigma_\pi]}{\sum_{k \neq j} 1 / \sigma_k^- + \mu_\pi / \sigma_\pi} \quad (3.40)$$

$$\sigma_j^+ = \left[ \sum_{k \neq j} 1 / \sigma_k^- + 1 / \sigma_\pi \right]^{-1} \quad (3.41)$$

De esta forma, cada variable integra los mensajes tanto de sus padres como de sus hijos a través de las siguientes ecuaciones:

$$\mu_{pi} = \sum_i b_i \mu_i^+ \quad (3.42)$$

$$\sigma_{pi} = \sum_i b_i^2 \sigma_i^+ \quad (3.43)$$

$$\mu_\lambda = \sigma_\lambda \sum_j \mu_j^- / \sigma_j^- \quad (3.44)$$

$$\sigma_\lambda = \left[ \sum_j 1 / \sigma_j^- \right]^{-1} \quad (3.45)$$

Finalmente, cada variable combina la información desde sus padres e hijos para obtener el promedio y desviación estándar mediante las siguientes ecuaciones:

$$\mu_X = \frac{\sigma_\pi \mu_\lambda + \sigma_\lambda \mu_\pi}{\sigma_\pi + \sigma_\lambda} \quad (3.46)$$

$$\sigma_X = \frac{\sigma_\pi \sigma_\lambda}{\sigma_\pi + \sigma_\lambda} \quad (3.47)$$

### 3.8. Métodos de imputación

La ausencia de datos ocasionada por fallas en sensores, errores humanos o data corrupta es una de las mayores dificultades a enfrentar previamente a un problema de modelación predictiva. En cualquier caso, existen diferentes técnicas para lidiar con este problema, tales como: interpolación, reemplazo por media aritmética, reemplazo aleatorio, entre otros. Sin embargo, en el caso de las series de tiempo se debe considerar el dominio temporal, de lo contrario, se pueden corromper estructuras tales como autocorrelación, tendencia y estacionariedad. Algunos métodos usados en la literatura se listan a continuación:

- **List-wise:** este método deja fuera de todo análisis a todas las observaciones que contengan valores nulos. Por lo tanto, el tamaño del set de datos es reducido y en el caso de las series temporales puede lidiar a intervalos de tiempo irregulares.
- **Reemplazo media aritmética:** método que reemplaza los valores nulos por el promedio del conjunto de datos. Este procedimiento mantiene el tamaño de los datos, sin embargo, ocasiona que su variabilidad sea menor y, por lo tanto, la desviación estándar puede ser subestimada. [31]
- **Media móvil exponencial:** los valores faltantes son imputados calculando la media móvil ponderada exponencial (EWMA) [32]. En primer lugar, se establece un tamaño de ventana. En segundo, a partir de la obtención de la media desde ambos lados de un dato faltante, este es imputado. Considerar que hay un valor nulo  $y_i$  y la ventana es fijada a  $w = 2$ . Finalmente, usando los valores  $y_{i-2}, y_{i-1}, y_{i+1}, y_{i+2}$  se calcula el promedio, que será el nuevo valor para  $y_i$ . Además, los pesos decrecen exponencialmente a medida que más lejos se está del valor central, donde el valor de EWMA a periodo  $t$ ,  $w$  está dado por  $S_t$ :

$$S_i = \lambda(y_{i-1} + (1 - \lambda)y_{i-2} + (1 - \lambda)^2 y_{i-3} + \dots + (1 - \lambda)^w y_{i-(w+1)}) S_{i-(w+1)} \quad (3.48)$$

$\lambda$  es un coeficiente que representa el grado de decaimiento.

- **Interpolación polinomial:** consiste en ajustar una curva de grado  $n$  a los datos discretos. Luego, usando la función ajustada, se estiman los puntos donde no existen registros.
- **Interpolación *spline*:** dado que ajustar solo una curva a datos puede producir oscilaciones, *spline* ajusta curvas de bajo orden en  $k$  segmentos de datos. De esta manera se evitan oscilaciones o valores divergentes.[33]
- **Imputación por método ARIMA:** un modelo autorregresivo integrado de media móvil modela una serie de tiempo mediante una tripleta  $(p, d, q)$ . Una observación cualquiera es modelada como una combinación lineal de observaciones anteriores, lo cual es denominado como componente autorregresiva  $p$ ;  $q$  se refiere a la componente de media móvil; y  $d$  el número de diferenciaciones de las series para cumplir estacionalidad. En este proyecto, se usa la librería `imputeTS` [34] que usa Kalman Smoothing sobre una representación en el espacio de estados de un modelo ARIMA obtenido por `auto.arima`.
- **Imputación de series de tiempo multivariable:** en el caso de múltiples series un enfoque *naive* de imputación, es necesario tratar cada serie por separado, lo cual no considera una eventual correlación inter-series.

Con el fin de capturar tanto la correlación entre series de tiempo como la estructura temporal de cada una, en [35] se propone como base el algoritmo esperanza-maximización para estimar un vector de promedio y la matriz de covarianza entre series temporales. Por otro lado, se modelan las relaciones temporales de cada serie a través de filtros tales como *spline* y ARIMA, que son usados para calcular el vector promedio para el paso del algoritmo EM.

Considerar el vector  $p$ -variado a tiempo  $t$ ,  $x_t$ , ( $t = 1, \dots, n$ ) con  $m$  datos no observados. El vector  $x_t$  puede ser reordenado de forma de que los  $m$  datos desconocidos queden al principio:  $x_t = (x_{t1}, \dots, x_{tm}, x_{t(m+1)}, \dots, x_{tp})^T$ , lo cual es representado como  $x_t = (x_{t1}, x_{t2})^T$ . Adicionalmente, se puede considerar que la data está dividida en  $B$  ventanas de tiempo con índices  $b$ , ( $b = 1, \dots, B$ ). De esta manera, el vector promedio a tiempo  $t$  y la matriz de covarianza en la ventana  $b$  puede ser representada como:

$$\tilde{\mu}_t = \begin{bmatrix} \tilde{\mu}_{t1} \\ \tilde{\mu}_{t2} \end{bmatrix} \quad (3.49)$$

$$\tilde{\Sigma}_t = \begin{bmatrix} \tilde{\Sigma}_{b11} & \tilde{\Sigma}_{b12} \\ \tilde{\Sigma}_{b21} & \tilde{\Sigma}_{b22} \end{bmatrix} \quad (3.50)$$

Los pasos a seguir para la estimación de datos faltantes consta de los siguientes pasos:

1. Reemplazar los datos faltantes por estimados. Típicamente se comienza con las medias muestrales y matriz de covarianza de los datos observados.

2. Estimar  $\mu$  y  $\Sigma$ .
3. Estimar el *level* de cada una de las series de tiempo.
4. Reestimar los valores faltantes usando los valores actualizados de los parámetros y el *level* de las series de tiempo.

### 3.9. Métricas de error

Un error de *forecasting* [36] se define como la diferencia entre el valor pronosticado con el valor real de validación:

$$e_{t+h} = y_{T+h} - \hat{y}_{T+h|T} \quad (3.51)$$

donde los datos de entrenamiento y validación están dados por  $\{y_1, \dots, y_T\}$  y  $\{y_{T+1}, y_{T+2}, \dots\}$ , respectivamente.

Existen variadas formas de estimar la precisión de predicciones en tareas de *forecasting*. Una de las más típicas es conocida como raíz del error cuadrático medio (RMSE), la cual considera los errores en la misma escala que los datos, y por lo tanto, no es útil para comparar dos series de tiempo en diferentes escalas. Notar que una minimización de RMSE conlleva a predecir el promedio de la serie.

$$\text{RMSE} = \sqrt{\text{mean}(e_t^2)} = \sqrt{\frac{\sum_{t=1}^T (\hat{y}_t - y_t)^2}{T}} \quad (3.52)$$

En el caso de que se quiera evaluar un modelo en diferentes conjuntos de datos con escalas no necesariamente iguales, se debe optar por usar errores porcentuales como referencia. Sin embargo, se debe considerar el problema de  $y_t = 0$  dado que el valor porcentual se indefine. A continuación se presentan los métodos de cálculo para RMSPE y MAPE, ambos de carácter porcentual. El primero considera los residuales en su forma cuadrática y el segundo en su forma absoluta.

$$\text{RMSPE} = \sqrt{\text{mean}((e_t/y_t)^2)} = \sqrt{\frac{\sum_{t=1}^T ((\hat{y}_t - y_t)/y_t)^2}{T}} \quad (3.53)$$

Una forma de interpretar MAPE consiste en que no hace más que medir qué tan lejos están las predicciones del promedio.

$$\text{MAPE} = \sqrt{\text{mean}(100 \cdot e_t/y_t)} = \frac{100}{n} \sum_{t=1}^T \left| \frac{\hat{y}_t - y_t}{y_t} \right| \quad (3.54)$$

### 3.10. Verificación de modelos

Un modelo dinámico lineal puede computar la predicción a un paso  $f_t = E(Y_t|y_{1:t-1})$ , por lo que el error de forecasting se define como:

$$e_t = Y_t - E(Y_t|y_{1:t-1}) = Y_t - f_t \quad (3.55)$$

Otra forma de escribir el error de forecasting es en términos de error de estimación a un paso:

$$e_t = Y_t - F_t a_t = F_t \theta_t + v_t - F_t a_t = F_t(\theta_t - a_t) + v_t \quad (3.56)$$

La secuencia de errores de forecasting  $(e_t)_{t \geq 1}$  de un modelo DLM cumple las siguientes propiedades [12]:

- (i) El valor esperado de  $e_t$  es cero.
- (ii) El vector aleatorio  $e_t$  no está correlacionado con ninguna función de  $Y_1, \dots, Y_{t-1}$ .
- (iii) Para cualquier  $s \leq t$ ,  $e_t$  e  $Y_s$  no están correlacionados.
- (iv) Para cualquier  $s \leq t$ ,  $e_t$  y  $e_s$  no están correlacionados.
- (v)  $e_t$  es una función lineal de  $Y_1, \dots, Y_t$ .
- (vi)  $(e_t)_{t \geq 1}$  es un proceso gaussiano

En el caso de observaciones univariadas la secuencia de errores o innovaciones estandarizadas, definidas por  $\tilde{e}_t = e_t / \sqrt{Q_t}$ , es ruido gaussiano blanco, es decir, una secuencia de variables aleatorias independientes e idénticamente distribuidas de media 0. Esta propiedad puede aprovecharse para comprobar los supuestos del modelo: si el modelo es correcto, la secuencia  $\tilde{e}_1, \dots, \tilde{e}_t$  calculada a partir de los datos debería parecerse a una muestra de tamaño  $t$  de una distribución normal estándar. Para llevar a cabo la verificación hay pruebas que se dividen en dos grandes categorías: las destinadas a comprobar si la distribución de las  $\tilde{e}_t$ 's es normal estándar, y las destinadas a comprobar si las  $\tilde{e}_t$ 's no están correlacionadas.

Los más útiles son, en opinión de los autores de [12], un gráfico QQ y un gráfico de la función de autocorrelación empírica de las innovaciones estandarizadas. El primero puede utilizarse para evaluar la normalidad, mientras que el segundo revela las desviaciones de la falta de correlación.

### 3.10.1. QQ-Plot

El propósito de la gráfica cuantil-cuantil (QQ) es mostrar si dos conjuntos de datos provienen de la misma distribución [37], trazando los cuantiles del primer conjunto de datos a lo largo del eje x y trazando los cuantiles del segundo conjunto de datos a lo largo del eje y. En la práctica, muchos conjuntos de datos se comparan con la distribución normal. La distribución normal es la distribución base y sus cuantiles se trazan a lo largo del eje x como cuantiles teóricos, mientras que los cuantiles de la muestra se trazan a lo largo del eje y como cuantiles de la muestra. En la figura 3.5 se observa el caso simulado de una distribución normal, donde las líneas calzan en su mayor parte lo que indica que ambos cuantiles calzan.

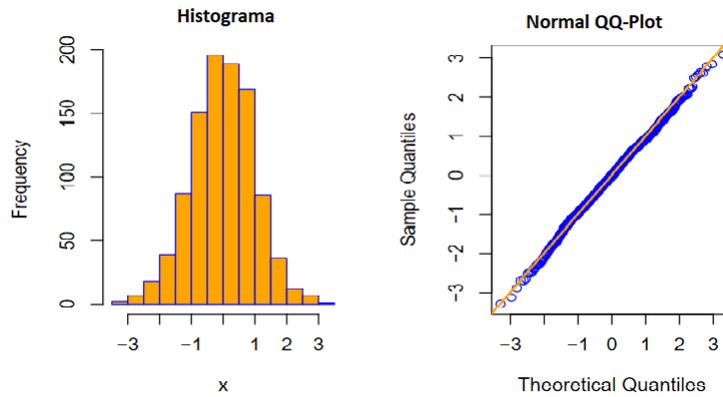


Figura 3.5: QQ-Plot: Simulación distribución normal.

### 3.10.2. Gráfico de Autocorrelación

Un correlograma o ACF es una forma visual de mostrar la correlación serial en datos que cambian con el tiempo [38]. La correlación serial (también llamada autocorrelación) es cuando un error en un punto en el tiempo se traslada a un punto posterior en el tiempo. Con los datos de series temporales, es muy probable que el valor de una variable observada en el período actual sea similar a su valor en el período anterior, y así sucesivamente. Por lo tanto, al ajustar un modelo de regresión a los datos de las series temporales, es habitual encontrar autocorrelación en los residuos. En este caso, el modelo estimado incumple el supuesto de ausencia de autocorrelación en los errores, y las previsiones pueden ser ineficientes, ya que queda algo de información que debería tenerse en cuenta en el modelo para obtener mejores previsiones. En la figura 3.6 se muestra un ejemplo donde los residuos de un modelo de regresión no muestran correlación significativa, por lo que se concluye que el modelo captura los patrones presentes en los datos.

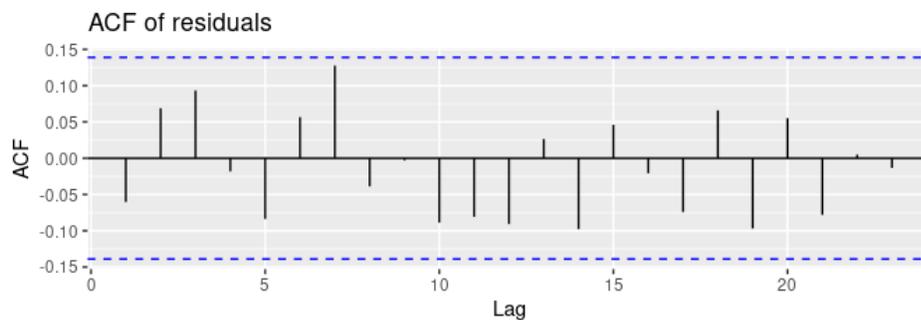


Figura 3.6: ACF Plot.

### 3.10.3. Prueba de Ljung-Box

La prueba de Ljung-Box en el caso de análisis de residuos puede ser definida como [39]:

- **H0**: Los residuos se distribuyen de forma independiente.
- **H1**: Los residuos no se distribuyen de forma independiente; exhiben correlación serial.

La hipótesis nula en el caso de los residuos de un modelo ajustado afirma un ajuste apropiado. Mientras que la hipótesis alternativa indica una falta de ajuste.

Se puede rechazar la hipótesis nula, es decir, rechazar que el modelo ha sido perfectamente ajustado cuando  $Q > \chi^2_{1-\alpha, h}$ . Donde esta última expresión, hace referencia el valor encontrado en la tabla de distribución chi-cuadrado para el nivel de significación  $\alpha$  y  $h$  grados de libertad.

### 3.10.4. Prueba de Kolmogorov-Smirnov

La prueba de Kolmogorov-Smirnov para dos muestras de datos mide la similitud de estas distribuciones, es decir, prueba si ambas muestras provienen de la misma distribución. Formalmente, considerar  $x_1, \dots, x_m$  y  $y_1, \dots, y_n$  dos muestras de observaciones independientes de poblaciones con funciones de distribución continuas  $F$  y  $G$ , respectivamente. [18]-[40]. Se evalúa

- **H0:**  $F = G$
- **H1:**  $F \neq G$

Siendo las distribuciones empíricas  $F_m$  y  $G_n$ , el estadístico de prueba mide la distancia entre dos funciones empíricas está dado por:

$$D_{mn} = \sqrt{\frac{mn}{m+n}} \sup_x |F_m(x) - \hat{G}_n(x)| \quad (3.57)$$

Las pruebas de Kolmogorov-Smirnov se aplican para las series de validación y de *forecasting*, para analizar si un modelo de interés genera una distribución similar a la distribución empírica de los datos reales o validación.

# Capítulo 4

## Metodología

En este apartado se presenta la metodología de desarrollo de la memoria. En primer lugar, se comienza por entender el proceso de producción de carbonato de Litio en la planta Salar del Carmen. Tras lo cual se definen los objetivos y el alcance de la memoria en función de la disponibilidad de data.

En segundo lugar, se analiza la base de datos que lleva el registro del proceso. Se procede a estudiar la tasa de agregación de cada una de las variables, su nivel de ruido, cantidad de valores nulos y se define un rango de datos que permita una correcta imputación. Tras esto último, se procede a eliminar la componente ruidosa de las señales dado un proceso de sensado en un ambiente hostil, para lo cual se aplica un filtro de Kalman en su tarea de suavizado o *smoothing*. Como paso final, se detectan y filtran anomalías de las tasas de envasado de acuerdo con el criterio estadístico del rango inter-cuartil.

Una vez finalizado el procesamiento de datos, se definen y combinan los modelos propuestos: redes bayesiana gaussianas dinámicas, espacio estado (ARMA - LLT) y de segundo orden (apARCH). Tras lo cual se estiman los parámetros de cada modelo; y dadas las propiedades de las redes bayesianas, se entrenan diferentes versiones para estas con el fin de estudiar la influencia del uso de conocimiento experto y vectores de evidencia. Para esto último, se usan modelos univariados en las variables ajustables para la generación de vectores que son usadas como entrada en la inferencia de los modelos DGBN. Tras la definición de todos los modelos se estudia su desempeño mediante métricas de error y se pone a prueba la distribución generada por estos con respecto a la distribución de validación.

Finalmente, se crea una herramienta web que permite la interacción y comparación de los modelos a través de métricas de error. Además, la aplicación es integrada a la base de datos del proceso tanto para lectura como escritura de datos, con el fin de la creación de historial para uso futuro.

## 4.1. Flujo de trabajo

La modelación del proceso debido a los datos disponibles se basa en las siguientes condiciones:

- **Se dispone de datos agregados por hora:** no se tiene acceso a los datos sensados en tiempo real, sino más bien, la lectura se realiza mediante una base de datos en línea donde las mediciones ya se encuentran agregadas mediante la media aritmética.
- **Son usadas todas las variables disponibles:** para ello, se requiere elegir un periodo de tiempo mínimo donde los porcentajes de nulos sean inferiores a 30 % para una correcta imputación de datos. Lo cual implica una reducción de la disponibilidad original. La razón para considerar todas las variables, radica en el interés por detectar las relaciones de dependencia condicional que existen entre ellas.
- **Los sensores están inmersos en un ambiente hostil:** por lo tanto se tiene una importante componente de ruido en las mediciones, la cual será tratada mediante el suavizado de datos.
- **Se modela la planta en puntos de operación normal:** lo que requiere descartar puntos de operación anómalos.

La metodología llevada a cabo para este trabajo puede ser resumida, a grandes rasgos, en cuatro bloques que son mostrados en la Figura 4.1.

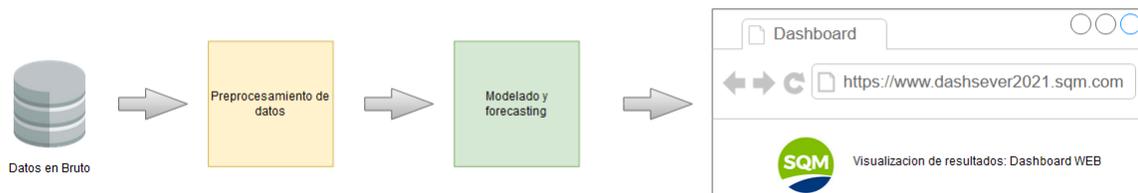


Figura 4.1: Flujo de trabajo general.

Como se puede observar, se comienza por la consulta de datos en bruto, los cuales son actualizados diariamente y están disponibles en una base de datos de tipo relacional almacenada en Azure. En la Tabla 2.1 se observan las variables disponibles; cabe destacar que se tiene interés de *forecasting* no solo en las variables de objetivo, sino también en aquellas ajustables que serán usadas como evidencia durante la etapa de modelado.

Una vez obtenidos los datos en bruto, estos son preprocesados para controlar la cantidad de ruido, porcentaje de nulos y puntos de operación no deseados. Por consiguiente, se genera una nueva base de datos para el modelado posterior. Luego, se continúa con el entrenamiento de múltiples modelos univariados y multivariados para realizar *forecasting* de  $n$  pasos y así comparar su desempeño en una ventana de tiempo. Un punto importante a destacar, es

que en el contexto de este trabajo el modelo multivariable corresponde a la red DGBN que permite hacer labores como: *forecasting* con y sin evidencia, adición de conocimiento experto para el aprendizaje de estructuras y detección de relaciones de dependencia condicional entre variables. Estas características convierten a este tipo de modelos en una poderosa herramienta de simulación.

Como etapa final del trabajo, se crea una aplicación web sobre lenguaje Python que contiene la lógica embebida de las etapas anteriores. De esta manera, el usuario final puede fácilmente interactuar con los modelos propuestos y visualizar los últimos resultados disponibles.

## 4.2. Preprocesamiento de datos

Con el fin de generar una base de datos de calidad para etapas posteriores, los datos en bruto son tratados mediante tres subtareas que se muestran en la Figura 4.2.

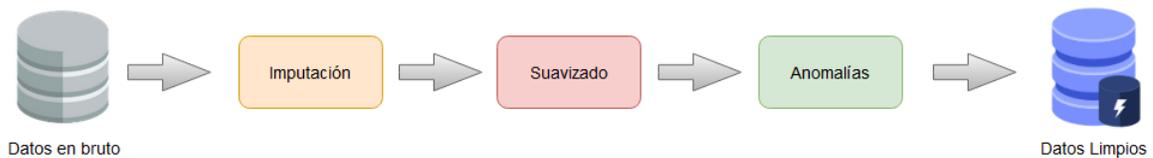


Figura 4.2: Bloque de preprocesamiento de datos.

Resulta importante destacar que no se tiene acceso a los datos en tiempo real de sensado, sino que a la agregación por hora de cada una de las variables. Es decir, se tiene una fila nueva por cada hora del día con los datos promediados en ese intervalo de tiempo, que por lo general no siguen un número constante de muestras por hora. En la Figura 4.3 se observan los datos disponibles para el proceso de interés con fecha de inicio en abril de 2020. Allí se cuenta con once variables, que poseen distintas características según composición de nulos y niveles de ruido.

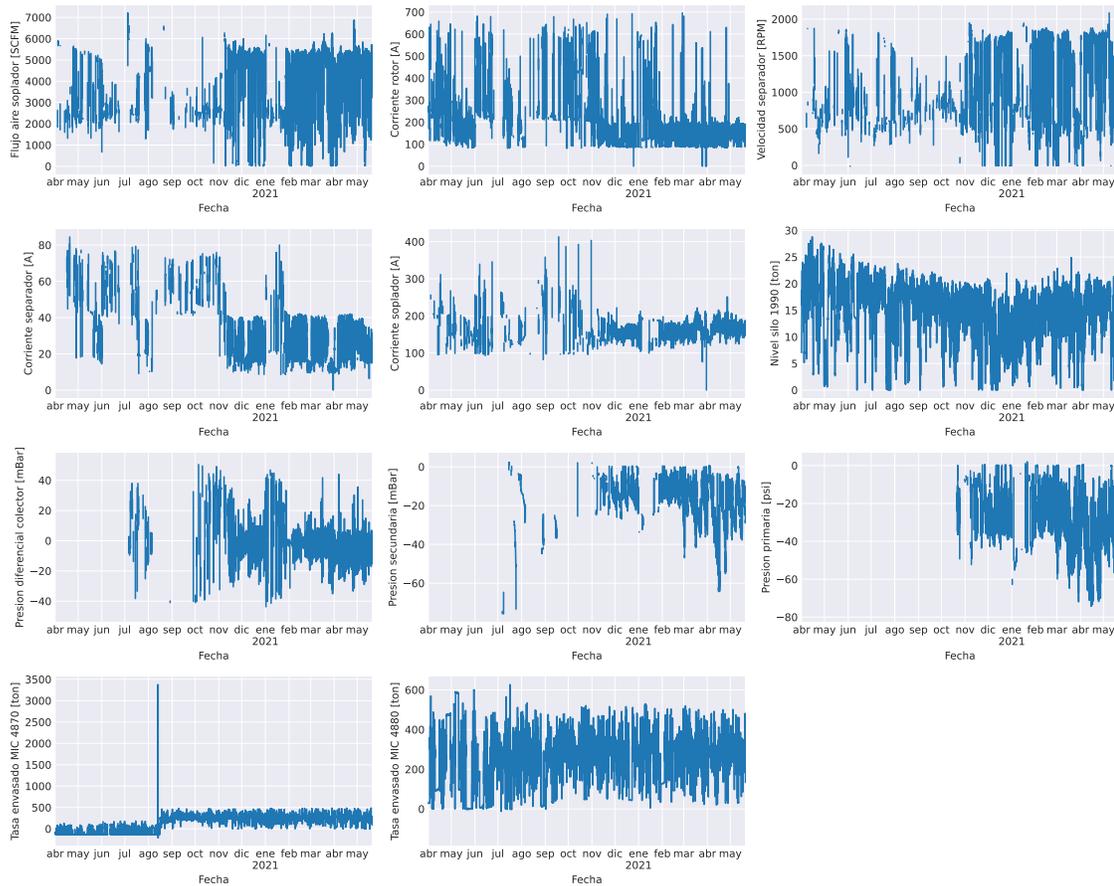


Figura 4.3: Datos en Bruto. Agregación original 1H. Periodo: 2020-04-01 a 20-05-10.

### 4.2.1. Análisis datos nulos

Como fue definido en la sección anterior, es necesario usar todas las variables disponibles, lo cual implica no superar un 30 % de valores nulos. Esto se debe a que (como ya fue descrito en la sección 3.8) se obtiene un mejor desempeño al momento de aplicar algoritmos de imputación. Si bien los datos originales están agregados a una tasa de periodo de 1H, se decide añadirlos nuevamente a una tasa de 2H con el fin de reducir nulos y evitar *forecasting* de demasiados pasos para alcanzar 1/2 días. De esta manera, el *forecasting* solo necesita 12 pasos por día y no 24 como se esperaría.

Ahora bien, en la Tabla 4.1 se observa la cantidad de nulos en términos de porcentaje a la fecha 10 de mayo de 2021, con un intervalo de agregación de 2H. Es importante mencionar que a medida que el periodo se acerca a la fecha actual, existe una mejora sustancial en la disponibilidad de datos, pasando de valores de 63 % a menos de 30 % como es el caso de la variable **Presion\_primaria\_psi**. Por lo cual, se concluye que la empresa está continuamente mejorando la calidad de colección, lo que permitirá mejores análisis futuros.

Tabla 4.1: Porcentaje de nulos sobre datos agregados a 2H. Fecha final 2021-05-10.

Variable	2020-04-01	2020-05-01	2020-06-01	2020-07-01	2020-08-01	2020-09-01	2020-10-01	2020-11-01	2020-12-01	2021-01-01	2021-02-01
Flujo_aire_soplador_SCFM	49.02%	47.18%	45.71%	42.88%	39.26%	33.84%	26.48%	18.47%	15.16%	16.94%	3.49%
Corriente_rotor_A	31.88%	31.33%	32.3%	30.09%	27.73%	22.78%	18.11%	12.9%	12.77%	12.02%	5.45%
Velocidad_separador_RPM	51.62%	50.12%	48.21%	46.47%	44.63%	40.39%	34.85%	28.3%	26.26%	27.02%	18.13%
Corriente_separador_A	53.35%	50.75%	49.48%	47.22%	44.37%	39.19%	33.5%	26.33%	22.67%	22.95%	11.57%
Corriente_soplador_A	52.51%	50.61%	48.51%	45.94%	43.3%	38.53%	32.06%	24.66%	20.22%	21.91%	8.68%
Nivel_silo_1990_ton	0.23%	0.25%	0.27%	0.29%	0.33%	0.37%	0.41%	0.48%	0.05%	0.06%	0.09%
Presion_diferencial_colector_mBar	60.55%	57.39%	53.54%	49.08%	46.23%	40.88%	33.8%	26.42%	22.41%	23.92%	11.06%
Presion_secundaria_mBar	63.9%	61.0%	57.47%	53.4%	50.58%	46.4%	40.14%	30.72%	25.74%	28.05%	13.19%
Presion_primaria_psi	61.19%	58.08%	54.29%	49.91%	44.4%	37.53%	29.05%	19.7%	15.63%	17.32%	3.74%
Tasa_envasado_MIC_4870_ton	3.16%	3.01%	2.53%	1.23%	0.68%	0.6%	0.68%	0.79%	0.26%	0.32%	0.43%
Tasa_envasado_MIC_4880_ton	0.27%	0.29%	0.32%	0.35%	0.38%	0.43%	0.49%	0.57%	0.0%	0.0%	0.0%

Tras el análisis de los datos se concluye que a partir de diciembre 2020/enero 2021 los datos son bien comportados, en el sentido descrito anteriormente. Por lo tanto, para efectos de modelamiento se decide usar enero 2021 como fecha de inicio en la nueva base de datos de modelado.

## 4.2.2. Imputación de datos

Tras la elección de la fecha de inicio para efectos de modelado, se prosigue a imputar los valores nulos presentes para la generación de datos completos. En esta tarea se experimenta con el uso de dos algoritmos multivariantes: `mtsdi`<sup>1</sup> y `MICE`<sup>2</sup>. El primero de ellos fue diseñado específicamente para series de tiempo, mientras que el segundo, si bien es una técnica multivariable, no hace uso de eventuales componentes temporales.

En la Figura 4.4 se muestran de forma gráfica los resultados de la aplicación de los algoritmos mencionados. Es posible notar que en el caso `mtsdi` se observa un rango de valores más amplios en los datos imputados en comparación a su contraparte, `MICE`, que tiende a ser más conservador en este sentido.

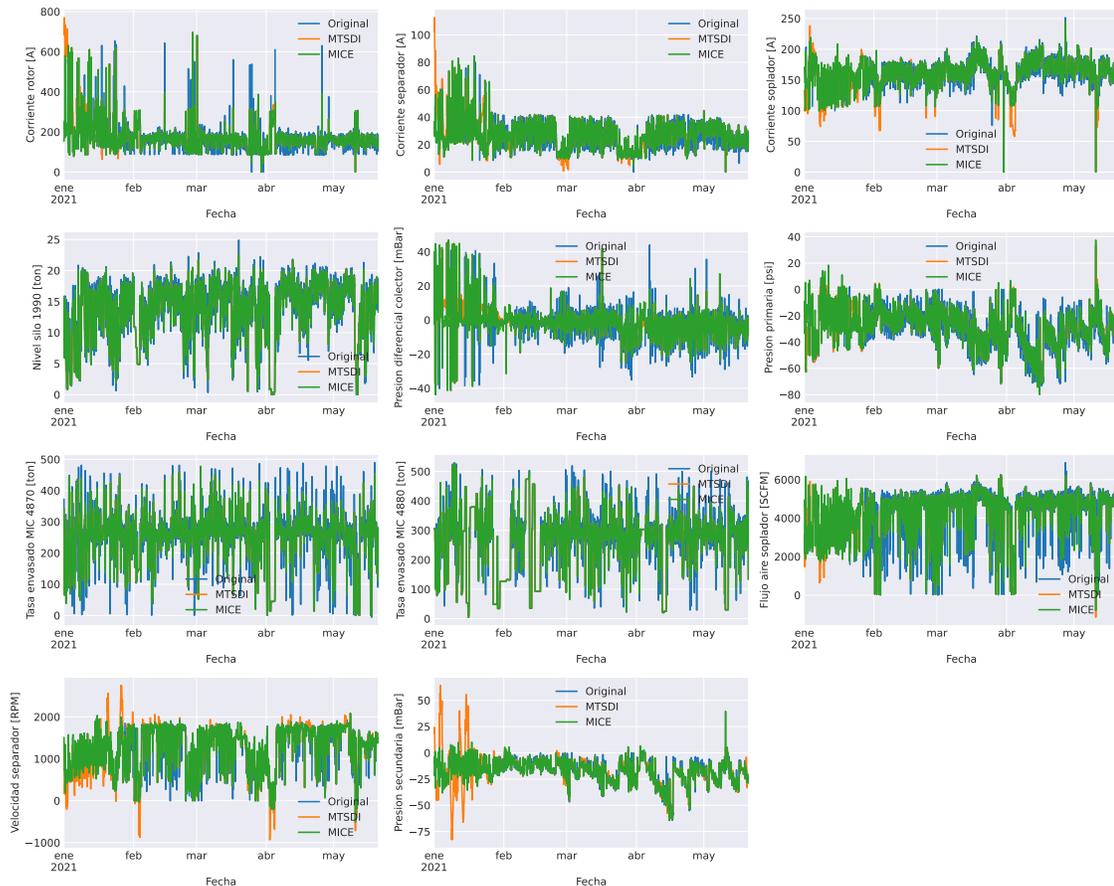


Figura 4.4: Datos Imputados. Periodo: 2021-01-01 a 20-05-10.

Debido a que ambos métodos se comportan de manera relativamente parecida, se decanta por usar `mtsdi` dada su consideración de correlaciones inter-series y temporalidad intra-serie. De todas formas, se implementan todos los algoritmos descritos en la sección 4.4 para un eventual uso futuro.

<sup>1</sup> `mtsdi`: Multivariate Time Series Data Imputation

<sup>2</sup> Multivariate Imputation by Chained Equations

### 4.2.3. Suavizado de datos

Considerando el ambiente hostil en el cual están inmersos los sensores, se asume que las mediciones y su posterior agregación contiene una importante componente de ruido. Se decide aplicar el algoritmo *Kalman Smoother* descrito en la sección 3.4, que busca remover el ruido observacional y conservar la tendencia o *level* para la detección de patrones de forma más clara.

Los resultados de la aplicación del algoritmo se muestran en la figura 4.5, donde los nuevos datos tienden a variar en rangos más pequeños, y además, algunos puntos pasan a ser ignorados cuando se tiene alta variabilidad.

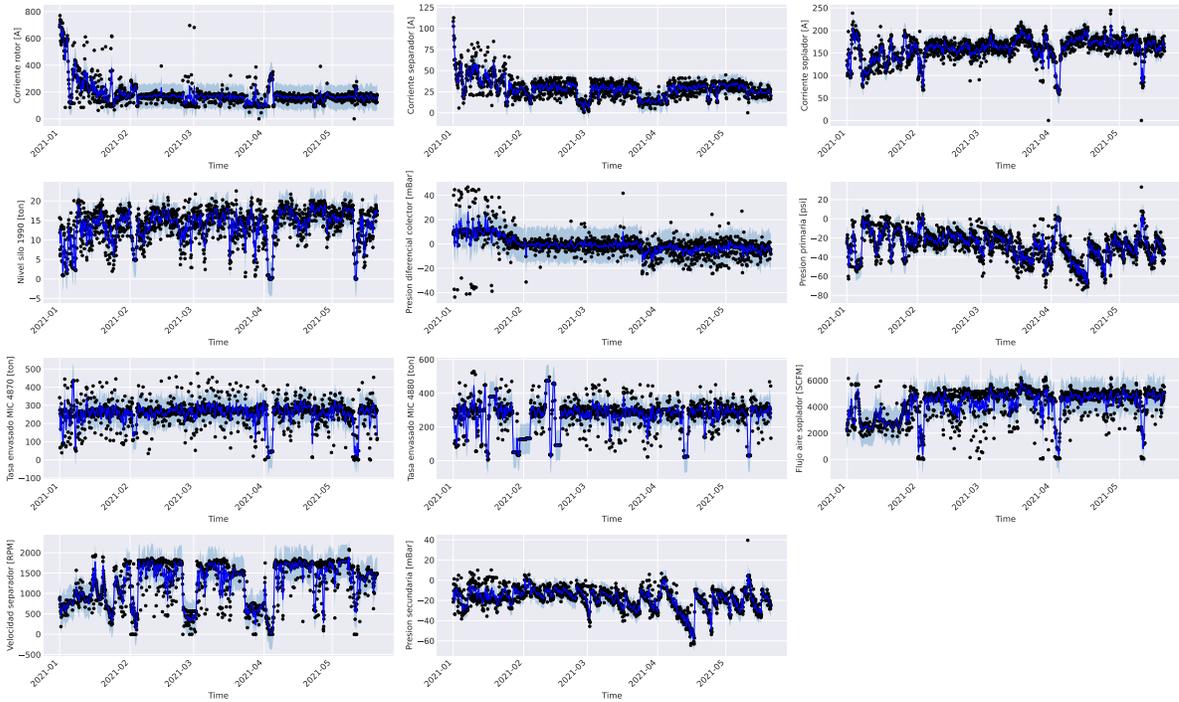


Figura 4.5: Datos suavizados. Periodo 2021-01-01 a 2021-05-10.

#### 4.2.4. Detección y eliminación de anomalías

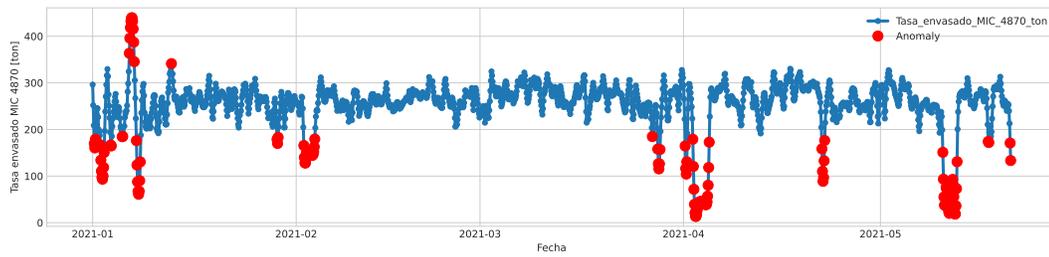
En principio, en la actual base de datos no se tiene ningún tipo de etiqueta con respecto a detenciones, por lo tanto no se puede saber de antemano si los datos corresponden efectivamente a la planta en un periodo de producción. Ahora bien, considerando que se tiene como objetivo modelar el funcionamiento solo en condiciones normales se deben filtrar de forma automatizada situaciones anómalas tales como: detención, fallas u otros.

Dado los antecedentes planteados, se opta por eliminar estos puntos en función de la técnica estadística de rango inter-cuartílico [41] con  $c = 1.5$ , el cual es un detector común para este tipo de casos que se basa en la estadística histórica de los mismos datos. Un valor es descartado o etiquetado como de “no funcionamiento” cuando está fuera del rango definido por la siguiente expresión:

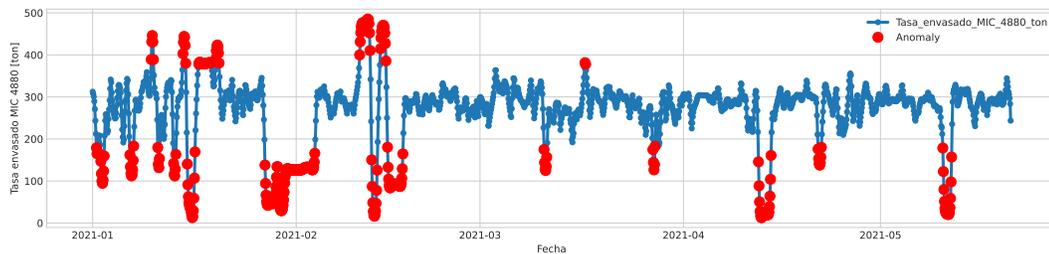
$$[Q1 - c \times IQR, Q3 + c \times IQR] \quad (4.1)$$

aquí,  $IQR = Q3 - Q1$  corresponde a la diferencia entre el 25 % y el 75 % de los cuantiles.

A modo de ejemplo, se muestran los puntos excluidos para el periodo estudiado en la figura 4.6. Es necesario destacar que no solo se descartan puntos de operación bajos, sino también puntos altos pero carácter atípico en la producción histórica.



(a) Tasa de Envasado MIC 4870.



(b) Tasa de Envasado MIC 4880.

Figura 4.6: Anomalías tasa de envasado.

### 4.3. Modelado y *forecasting*

A partir de los nuevos datos de modelado, se procede a entrenar tanto modelos univariados como multivariados con el fin de generar un vector de *forecasting* de  $n$  pasos. Cada uno de los modelos generados son evaluados en un set de validación, con el objetivo de compararlos a base de varias métricas de error, y con esto decidir las opciones más apropiadas.

Se recalca que los modelos son automatizados en la elección de sus parámetros dado el dinamismo de la planta, es decir, con cada nuevo día de datos los modelos son reentrenados sobre una ventana de tiempo, por lo que los parámetros pueden continuamente cambiar.

#### 4.3.1. Modelos univariados

Los modelos univariados son usados para dos propósitos mediante el esquema mostrado en la Figura 4.7:

- **Generación evidencia:** se usarán los vectores de *forecasting* sobre las variables ajustables para ser usadas como evidencia, es decir, la DGBN no deberá hacer predicciones en estos nodos. Se buscará estudiar el efecto de ensamble de modelos sobre los resultados finales.
- ***Forecasting* de variables objetivos:** se pretende contrastar el desempeño con los modelos multivariados para las tasas de envasado. *A priori* no se puede afirmar que existe una relación clara entre las variables sensadas y las variables objetivo.

En particular, los modelos propuestos son:

- **ARMA:** se hace uso del paquete de `auto.arima` para generar una definición de modelo inicial automatizada basada en el criterio AIC, que luego es transformada a su representación espacio estado. Tras esto, se genera *forecasting* mediante el uso de filtro de Kalman.
- **LLT:** se inicializa el modelo espacio-estado y se estiman sus parámetros mediante MLE, para luego generar *forecasting* mediante filtro de Kalman.
- **GARCH:** a modo de referencia, se usa un modelo basado en estadística de segundo orden: `apARCH(1,1)`. Dada la imposibilidad de automatizar la elección de sus parámetros, se usa en su forma más básica, tras lo cual se genera *forecasting* mediante *bootstrapping*.

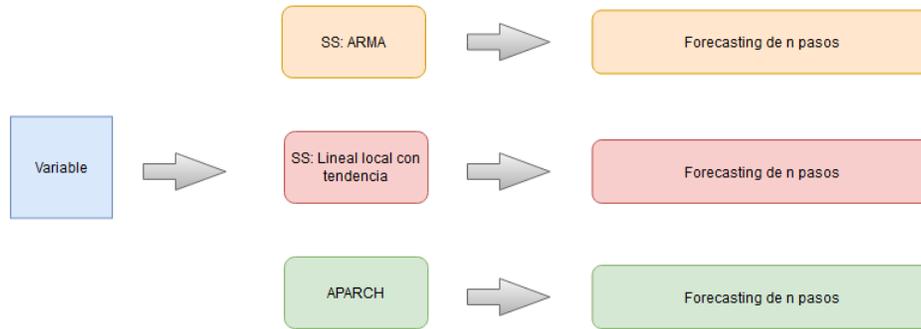


Figura 4.7: Esquema de generación de *forecasting*. Modelos univariados.

### 4.3.2. Modelos multivariados

En este apartado son consideradas todas las variables disponibles para la modelación mediante DGBN. Se entrenan dos tipos de redes: con y sin adición de restricciones en el aprendizaje de estructura. Para el caso de la inferencia, puede realizarse con y sin evidencia<sup>3</sup>. Esta característica puede ser usada para estudiar los efectos de variaciones sobre variables, sirviendo como un simulador de sistemas complejos.

Ahora bien, en este trabajo se generan predicciones sobre las variables ajustables y se usa como evidencia para la inferencia de las redes DGBN, tal como se menciona en la sección anterior. Los tipos de redes obtenidas y a su vez las inferencias planteadas se observan en la Figura 4.8.

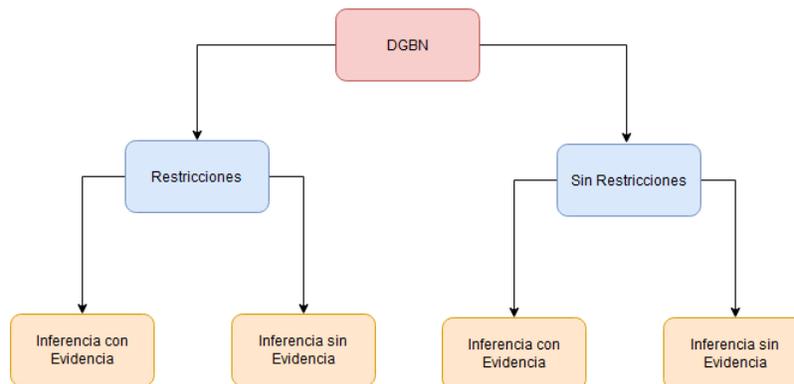


Figura 4.8: Inferencia sobre modelo DGBN. Modelo multivariado.

<sup>3</sup> En este contexto, evidencia se entiende como el conocimiento previo de los valores futuros de uno o más nodos de la red.

### 4.3.2.1. Restricciones

En el caso del aprendizaje de estructura del modelo DGBN se proponen condiciones restrictivas para el algoritmo DMMHC. A continuación, en la Tabla 4.2, se muestran los arcos no permitidos en la estructura a aprender. Estos arcos se basan en la siguiente premisa: las tasas de envasado son un producto final, por lo tanto no hay relaciones de dependencia condicional que partan desde sus nodos.

Tabla 4.2: Arcos prohibidos.

from	to
Tasa_ensvasado_MIC_4870_ton_t_0	Corriente_rotor_A_t_0
Tasa_ensvasado_MIC_4870_ton_t_0	Presion_secundaria_mBar_t_0
Tasa_ensvasado_MIC_4880_ton_t_0	Presion_secundaria_mBar_t_0
Tasa_ensvasado_MIC_4870_ton_t_0	Presion_primaria_psi_t_0
Tasa_ensvasado_MIC_4880_ton_t_0	Presion_primaria_psi_t_0
Tasa_ensvasado_MIC_4880_ton_t_0	Presion_diferencial_colector_mBar_t_0
Tasa_ensvasado_MIC_4870_ton_t_0	Velocidad_separador_RPM_t_0

De forma similar, en la Tabla 4.3 se listan las dependencias condicionales impuestas, que relacionan las variables ajustables con las variables de interés.

Tabla 4.3: Arcos obligatorios.

from	to
Velocidad_separador_RPM_t_0	Tasa_ensvasado_MIC_4880_ton_t_0
Velocidad_separador_RPM_t_0	Tasa_ensvasado_MIC_4870_ton_t_0
Flujo_aire_soplador_SCFM_t_0	Tasa_ensvasado_MIC_4870_ton_t_0
Flujo_aire_soplador_SCFM_t_0	Tasa_ensvasado_MIC_4880_ton_t_0

### 4.3.3. Ensamble de modelos

Tras la descripción de los distintos modelos posibles para la planta, se procede a generar la combinación de cada una de las opciones, en las que se generan modelos univariados y multivariados para la predicción de las variables de interés. A modo de ejemplo, se explica la composición de algunos modelos en particular mostrados en la Tabla 4.4.

- **DGBN + flujo.aparch\_sep.localtrend:** hace referencia al modelo DGBN sin restricciones, que usa como evidencia el modelo APARCH para la variable Flujo\_aire\_soplador\_SCFM y el modelo lineal local con tendencia para Velocidad\_separador\_RPM.
- **ARMA:** las tasas de envasado son modeladas solo al considerar sus valores pasados.

Tabla 4.4: Lista de combinaciones de modelos para tasas de envasado.

Nombre modelo	Tipo	Restricciones	Evidencia
APARCH(1,1)	Univariable	-	-
ARMA	Univariable	-	-
Local Lineal con Tendencia (LLT)	Univariable	-	-
DGBN	Multivariable	No	No
DGBN + flujo.aparch_sep.localtrend	Multivariable	No	Si
DGBN + flujo.aparch_sep.arma	Multivariable	No	Si
DGBN + flujo.aparch_sep.aparch	Multivariable	No	Si
DGBN + flujo.localtrend_sep.arma	Multivariable	No	Si
DGBN + flujo.localtrend_sep.aparch	Multivariable	No	Si
DGBN + flujo.localtrend_sep.localtrend	Multivariable	No	Si
DGBN + flujo.arma_sep.aparch	Multivariable	No	Si
DGBN + flujo.arma_sep.arma	Multivariable	No	Si
DGBN + flujo.arma_sep.localtrend	Multivariable	No	Si
DGBN + restricciones	Multivariable	Si	No
DGBN + restricciones + flujo.aparch_sep.localtrend	Multivariable	Si	Si
DGBN + restricciones + flujo.aparch_sep.arma	Multivariable	Si	Si
DGBN + restricciones + flujo.aparch_sep.aparch	Multivariable	Si	Si
DGBN + restricciones + flujo.localtrend_sep.loc	Multivariable	Si	Si
DGBN + restricciones + flujo.localtrend_sep.arma	Multivariable	Si	Si
DGBN + restricciones + flujo.localtrend_sep.aparch	Multivariable	Si	Si
DGBN + restricciones + flujo.arma_sep.localtrend	Multivariable	Si	Si
DGBN + restricciones + flujo.arma_sep.arma	Multivariable	Si	Si
DGBN + restricciones + flujo.arma_sep.aparch	Multivariable	Si	Si

#### 4.3.4. Experimentación modelos

Tras la definición de los 23 modelos a considerar, se comienza a diseñar los experimentos que pongan a prueba la capacidad de predicción y detección de dependencias condicionales. De esta forma, se ejecuta el flujo de trabajo sobre el conjunto de datos determinado por las siguientes características:

- **Periodo particular:** 2021-03-26 al 2021-04-08. Para estudiar el ajuste de modelos en detalle.
- **Periodo general:** 2021-03-18 al 2021-04-30. Se repite el experimento 30 veces con una ventana deslizante, para identificar modelos estables.
- **Ventana de datos:** 12 días de entrenamiento y 2 días de validación.
- **Validación:** *forecasting out of sample* de 2 días con pasos de 4 horas.

En el caso del periodo general resultan más de 600 modelos entrenados, que, para su posterior análisis, se estudiará la distribución de las métricas de error: RMSPE y MAPE.

## 4.4. Dashboard web

Con el fin de disponibilizar el trabajo desarrollado, se crea una página web con características útiles para la experimentación de los modelos. Esta es almacenada en Azure a través de máquinas virtuales y es conectada de forma directa a la base de datos relacional disponible por la empresa. Se implementa mediante la librería Dash<sup>4</sup> en su versión para Python y la lógica de cálculo en realizada en R. Cada uno de los módulos disponibles se describen a continuación.

### 4.4.1. Selección de hiper-parámetros

El usuario puede incorporar conocimiento experto para el aprendizaje de la estructura de las redes mediante la adición o prohibición de arcos obligatorios. Además, puede elegir diferentes órdenes markovianos, además de tener la posibilidad de ajustar la ventana de entrenamiento de los modelos. El diseño se muestra en la Figura 4.9, donde además se incluye una breve explicación de los parámetros ajustables e información sobre el último trabajo computado.

Figura 4.9: Selección de hiper-parámetros.

### 4.4.2. Visualización redes entrenadas

Tras el entrenamiento de los modelos, se pueden observar dos estructuras en la Figura 4.10, que corresponden a una red sin restricciones y a otra red considerando el conocimiento experto adicionado por el usuario. En el ejemplo mostrado por la Figura 4.10, se observa que las redes son de orden markoviano 2, y es fácil notar las dependencias condicionales al observar las flechas que unen los nodos intra e inter-temporalmente. De esta manera, se ofrece una forma intuitiva de detección de dependencias condicionales.

<sup>4</sup> <https://plotly.com/dash/>

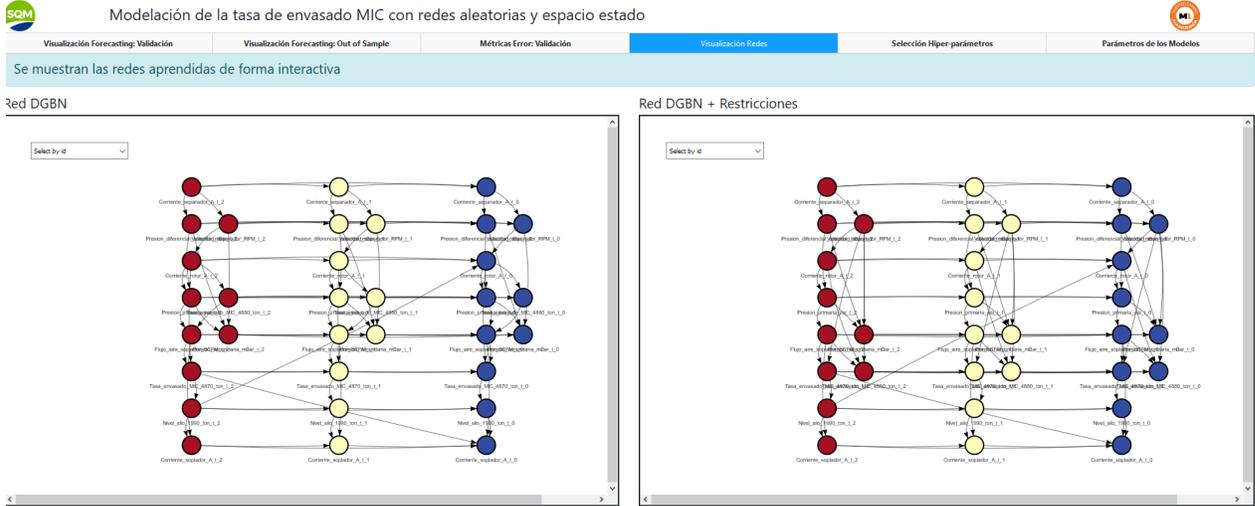


Figura 4.10: Visualización de redes.

### 4.4.3. Visualización de parámetros de redes

No solo se ofrece la posibilidad de analizar la red de forma gráfica, además sus parámetros pueden ser explorados en la web o ser exportados a una planilla de cálculo. Los parámetros que definen la red son: el vector de medias de los nodos y la matriz de covarianzas entre estos. En la Figura 4.11 se observan ambos conjuntos de parámetros para el caso de la red sin restricciones.

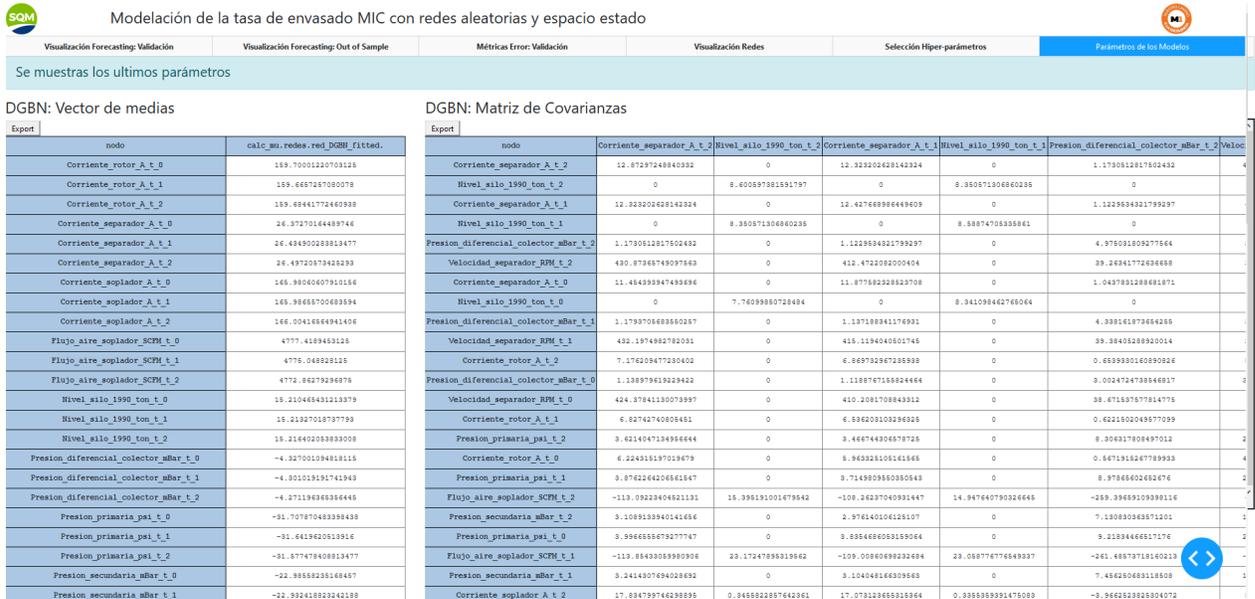


Figura 4.11: Visualización de parámetros.

#### 4.4.4. Visualización de métricas de error

Además de la visualización de los modelos entrenados, estos son puestos a prueba en un conjunto de validación de 2 días. El usuario puede comparar cada uno de los modelos e identificar cuál es el mejor para el presente periodo basándose en las métricas disponibles: error absoluto medio (MAE), error porcentual absoluto medio (MAPE), error cuadrado promedio (MSE), raíz del error cuadrático medio (RMSE), raíz del error cuadrático medio. porcentual (RMSPE) y error medio escalado absoluto (MASE)

Modelación de la tasa de envasado MIC con redes aleatorias y espacio estado

Se muestran las métricas de error sobre los datos de validación.

Métricas error Tasa de envasado 4880

Modelo	MAE	MAPE	MSE	RMSE	RMSPE	MASE	Rank	RHSE
DGBN	9.3799	0.039	180.3894	13.4309	0.0473	0.7801	1	
DGBN + flujo.aparch_sep.localtrend	9.9573	0.0849	183.0472	13.5302	0.0477	0.8248	2	
DGBN + flujo.aparch_sep.aparch	9.8937	0.0848	183.479	13.5454	0.0477	0.8229	3	
DGBN + flujo.aparch_sep.arma	9.8821	0.0348	183.8551	13.5593	0.0478	0.8219	4	
DGBN + flujo.arma_sep.aparch	12.2144	0.0432	193.7901	13.9209	0.0489	1.0159	5	
DGBN + flujo.arma_sep.arma	12.2023	0.0431	193.8436	13.9226	0.0489	1.0149	6	
DGBN + flujo.arma_sep.localtrend	12.238	0.0432	193.8742	13.9239	0.0489	1.0179	7	
DGBN + restricciones + flujo.aparch_sep.arma	11.6566	0.0413	200.9388	14.1753	0.0509	0.9695	8	
DGBN + restricciones	11.0444	0.039	209.4903	14.4757	0.0516	0.9186	9	
DGBN + restricciones + flujo.aparch_sep.aparch	11.0164	0.0389	209.846	14.4757	0.0515	0.9163	10	
DGBN + flujo.localtrend_sep.arma	13.90	0.0496	221.9659	14.8985	0.0539	1.1428	11	
DGBN + flujo.localtrend_sep.aparch	13.9316	0.0497	222.2969	14.9096	0.054	1.1697	12	
DGBN + flujo.localtrend_sep.localtrend	14.0182	0.0497	222.9529	14.9316	0.054	1.1657	13	
DGBN + restricciones + flujo.aparch_sep.localtrend	10.4181	0.0366	227.2235	15.0739	0.0528	0.8665	14	
DGBN + restricciones + flujo.arma_sep.localtrend	12.8486	0.0454	227.8339	15.0942	0.0541	1.0687	15	
DGBN + restricciones + flujo.arma_sep.aparch	14.0002	0.0496	236.672	15.3841	0.0566	1.1644	16	
DGBN + restricciones + flujo.arma_sep.arma	14.6403	0.052	246.0688	15.6866	0.0589	1.2177	17	
DGBN + restricciones + flujo.localtrend_sep.localtrend	13.1419	0.0539	258.8544	16.089	0.0584	1.2411	18	
DGBN + restricciones + flujo.localtrend_sep.aparch	14.3134	0.0581	298.1076	17.2658	0.0626	1.3648	19	
APARCH(1,1)	11.9346	0.0427	323.4644	17.9851	0.0639	0.9926	20	
DGBN + restricciones + flujo.localtrend_sep.arma	16.9636	0.0604	328.7445	18.1313	0.0656	1.4101	21	
ARMA	13.9492	0.05	335.33	18.312	0.0678	1.1602	22	
Local Lineal con Tendencia	17.6125	0.0633	420.8259	20.514	0.0765	1.4649	23	

Métricas error Tasa de envasado 4870

Modelo	MAE	MAPE	MSE	RMSE	RMSPE	MASE	Rank	RHSE
ARMA	22.2658	0.0814	526.5389	24.9844	0.107	0.908	1	
APARCH(1,1)	26.1823	0.0898	877.8202	29.623	0.1016	1.0682	2	
DGBN + restricciones + flujo.localtrend_sep.aparch	29.839	0.1024	1181.8958	34.3743	0.1214	1.2049	3	
DGBN + restricciones + flujo.localtrend_sep.arma	29.0816	0.1017	1181.6516	34.3752	0.1237	1.1856	4	
DGBN + restricciones + flujo.arma_sep.arma	31.0799	0.107	1227.9044	35.0418	0.1214	1.268	5	
DGBN + restricciones + flujo.localtrend_sep.localtrend	31.4463	0.1081	1239.5214	35.2065	0.1213	1.283	6	
DGBN + restricciones + flujo.arma_sep.aparch	32.1629	0.1101	1266.2144	35.6399	0.1213	1.3122	7	
DGBN + flujo.localtrend_sep.localtrend	32.9019	0.1112	1309.8692	36.1921	0.1251	1.3179	8	
DGBN + flujo.localtrend_sep.aparch	32.3617	0.1114	1311.6932	36.2179	0.1251	1.3203	9	
DGBN + flujo.localtrend_sep.arma	32.8921	0.1115	1312.2847	36.2255	0.1251	1.3215	10	
DGBN + restricciones + flujo.arma_sep.localtrend	34.0762	0.1157	1376.1287	37.0962	0.1242	1.3903	11	
DGBN + flujo.arma_sep.localtrend	34.095	0.1164	1388.2934	37.2598	0.1241	1.3908	12	
DGBN + flujo.arma_sep.aparch	34.3488	0.1165	1391.4652	37.3023	0.1241	1.3932	13	
DGBN + flujo.arma_sep.arma	34.3792	0.1166	1392.9728	37.3226	0.1242	1.3945	14	
DGBN + restricciones + flujo.aparch_sep.aparch	34.8838	0.1174	1401.991	37.443	0.1266	1.403	15	
DGBN + restricciones	35.0732	0.1192	1436.0844	37.8957	0.127	1.4309	16	
DGBN + restricciones + flujo.aparch_sep.aparch	35.4723	0.1204	1469.9954	38.3405	0.1282	1.4472	17	
DGBN + flujo.aparch_sep.localtrend	36.3796	0.1238	1538.8898	39.2287	0.1311	1.4842	18	
DGBN + flujo.aparch_sep.aparch	36.4394	0.1237	1543.2022	39.2836	0.1312	1.4867	19	
DGBN + flujo.aparch_sep.arma	36.4697	0.1238	1546.4875	39.3123	0.1313	1.4879	20	
DGBN	37.012	0.1266	1579.6165	39.7464	0.1327	1.5177	21	
DGBN + restricciones + flujo.aparch_sep.localtrend	37.3857	0.1261	1628.8684	40.3222	0.1333	1.5233	22	
Local Lineal con Tendencia	104.8487	0.368	12924.5062	113.686	0.4016	4.2777	23	

Figura 4.12: Métricas de error sobre datos de validación.

### 4.4.5. Visualización forecasting Out of sample

Finalmente, el usuario puede ver las proyecciones para los próximos 2 días. Donde los modelos son reentrenados incluyendo el periodo de validación.

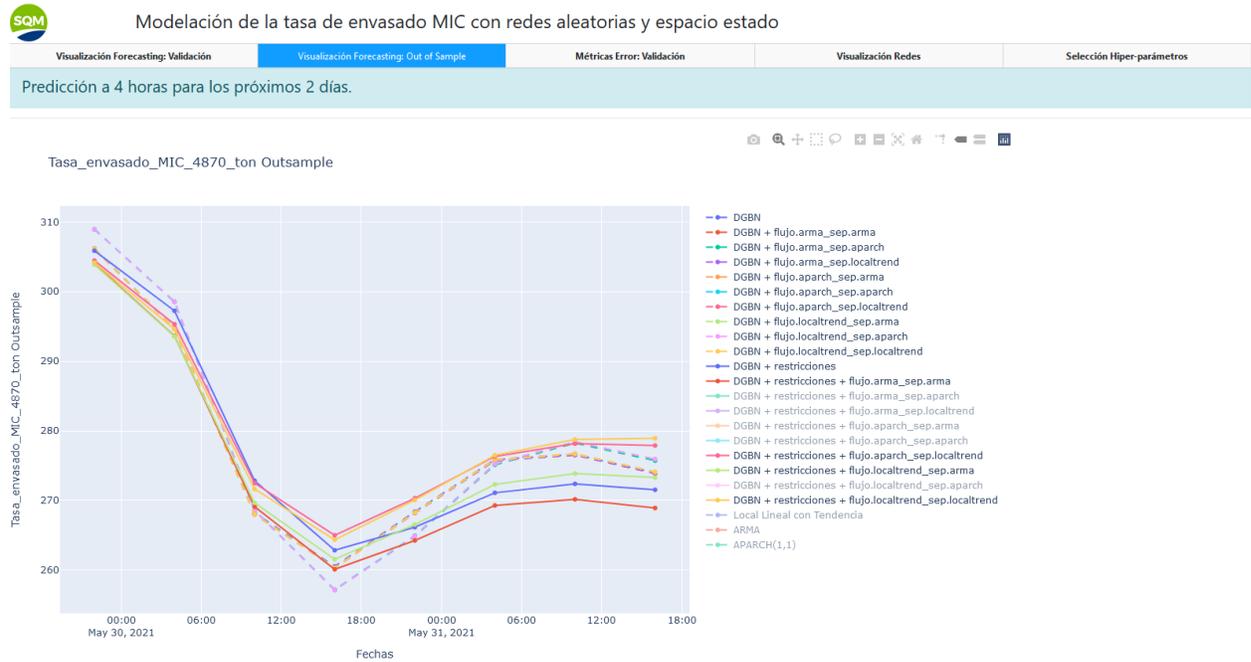


Figura 4.13: Visualización de forecasting Out of sample.

# Capítulo 5

## Resultados

En lo que sigue se da cuenta de los resultados obtenidos al usar la metodología propuesta en el Capítulo 4, aplicada en dos modalidades que se explican a continuación:

- **Periodo particular:** se estudia en detalle el desempeño de los distintos modelos sobre un periodo de tiempo en específico.
- **Periodo general:** se ejecuta el modelado durante un mes, es decir, 30 iteraciones. De lo cual, se obtiene una distribución de métricas de error y *p-values* para la prueba de Kolmogorov-Smirnov.

### 5.1. Periodo particular

A continuación se presentan los resultados de los modelos propuestos para el periodo 2021-03-26 al 2021-04-08. Cabe destacar que en este periodo las redes bayesianas gaussianas dinámicas son entrenadas con una propiedad markoviana de segundo orden, y que cada uno de los modelos resultantes son validados sobre los últimos dos días del periodo antes mencionado.

En primer lugar, se expone el estudio gráfico de residuos sobre el conjunto de entrenamiento para luego, en segundo lugar, proceder a revisar los resultados de *forecasting* sobre el conjunto de validación.

#### 5.1.1. Diagnóstico de modelos

En este diagnóstico se describe brevemente la calidad de ajuste tanto para las variables objetivo como las de ajuste mediante el uso de las herramientas introducidas en la Sección 3.10 sobre la base de análisis de residuos.

##### 5.1.1.1. Modelos ARMA

A continuación se estudian los residuos de los modelos ARMA en su representación espacio estado, ajustados por filtro de Kalman, descritos en la Sección 3.3.3.

## Flujo de aire soplador

En la Figura 5.1 se observa, por un lado, la existencia de una autocorrelación significativa en los residuos sobre el *lag* 6, por lo que se concluye que el modelo podría ser mejorado dada la existencia de información remanente no capturada. Por otro lado, los *p-values* para la prueba de Ljung-Box denotan que es posible rechazar la hipótesis nula (ver Sección 3.10.3), y por tanto, no es posible afirmar que el modelo ha sido perfectamente ajustado.

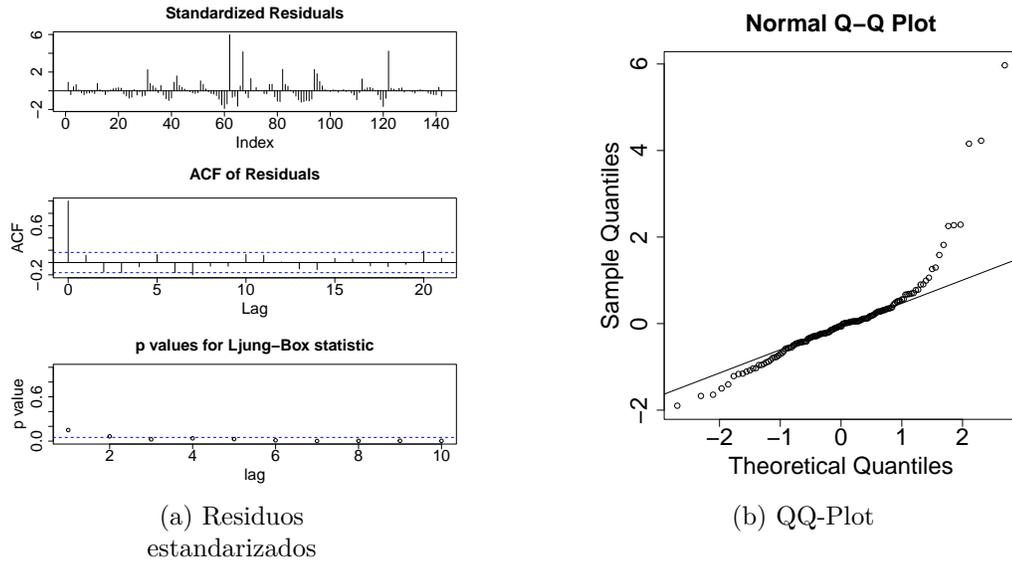


Figura 5.1: Residuos modelo ARMA: Flujo de aire soplador.

## Velocidad separador

En el caso del modelo ARMA para la variable Velocidad separador se tienen resultados satisfactorios. En general, como se observa en la Figura 5.2, no existe significancia importante en el gráfico de autocorrelación de residuos y, además, se tiene la suficiente significancia para aceptar la hipótesis nula de la prueba de Ljung-Box.

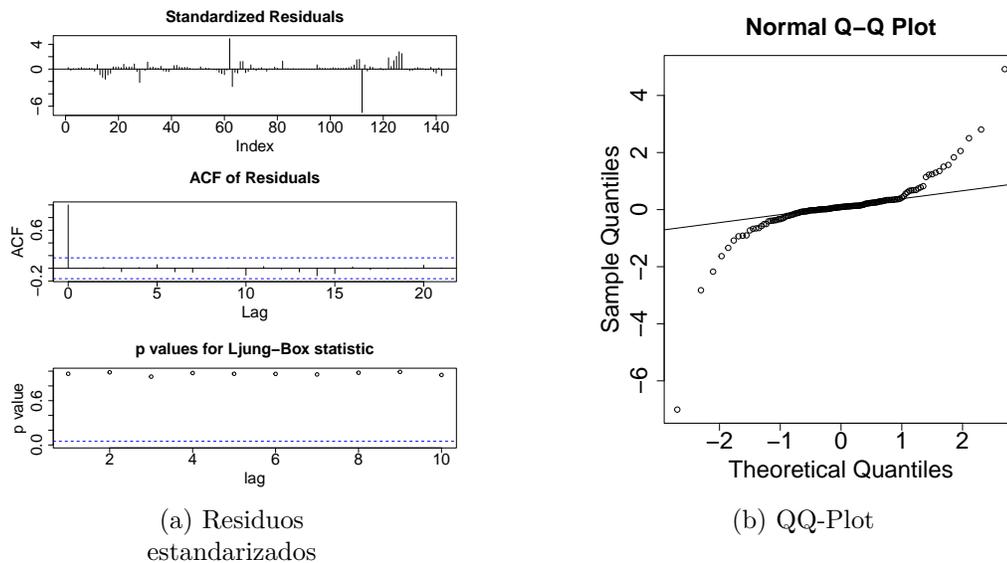


Figura 5.2: Residuos modelo ARMA: Velocidad separador.

### Tasa envasado MIC 4870

El ajuste para Tasa envasado MIC 4870 no logra un modelado satisfactorio, ya que se obtienen pobres resultados en el test de Ljung-Box a partir del *lag* 6 y autocorrelación importante en los *lags* 3, 6 y 10.

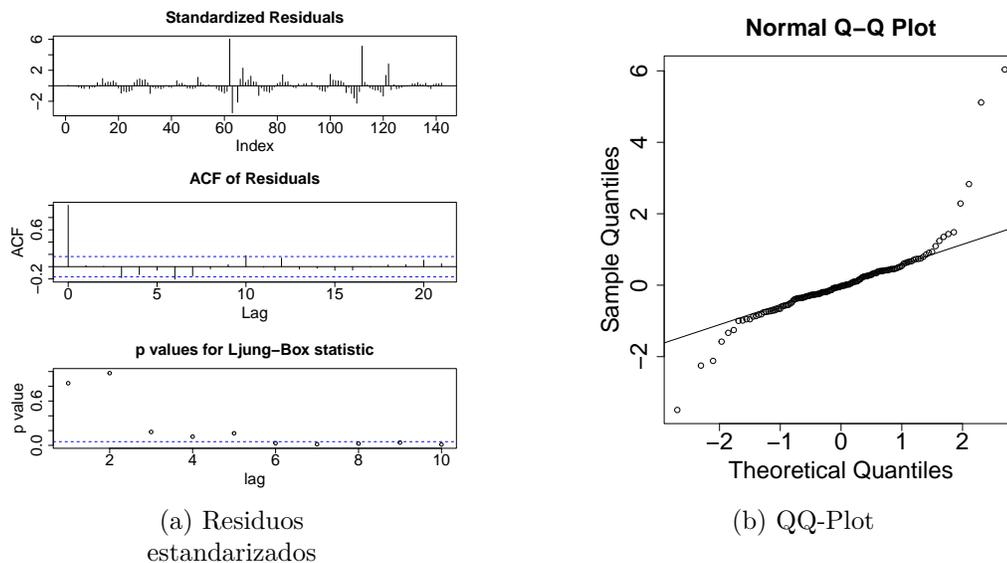


Figura 5.3: Residuos modelo ARMA: Tasa envasado MIC 4870.

### Tasa envasado MIC 4880

El análisis para Tasa envasado MIC 4880 da cuenta de un modelo significativamente bien ajustado, tal como reflejan los *p-values*. Además, en el gráfico QQ-plot de la Figura 5.4 se

observa la presencia solo de unos pocos *outliers* apartados de la recta de normalidad, luego se considera que los residuos distribuyen de forma normal.

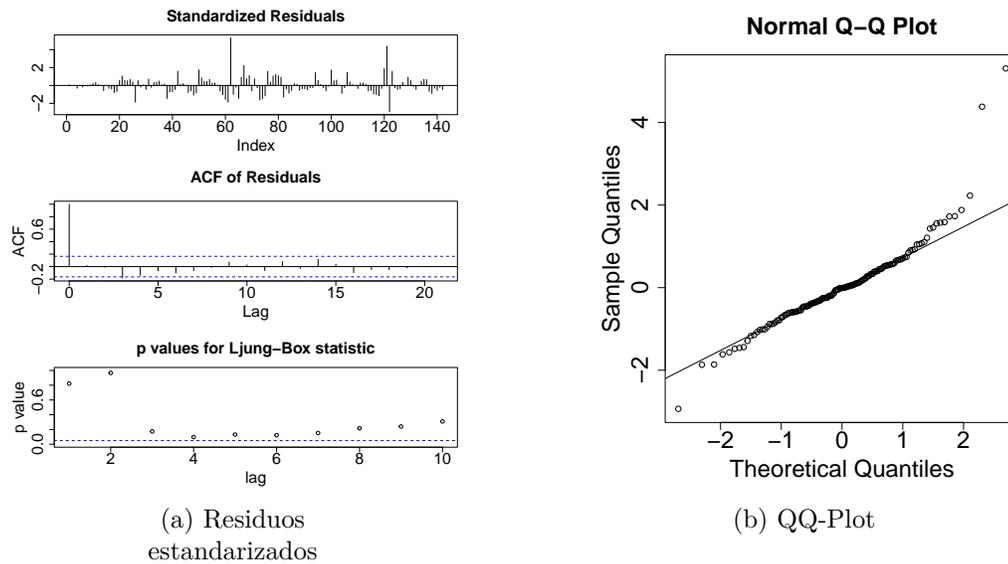


Figura 5.4: Residuos modelo ARMA: Tasa envasado MIC 4870.

### 5.1.1.2. Modelos LLT

Se describe la calidad de ajuste sobre el modelo local lineal con tendencia descrito en la Sección 3.3.2.

### Flujo de aire soplador

Los resultados mostrados en la Figura 5.5 denotan un pobre ajuste tanto en ACF (ver *lags* 3, 4, 5), Ljung-Box y por sobre todo, en el gráfico Q-Q Plot (ver Sección 3.10.1) donde se observan colas bastante alejadas de la recta, pues señalan la no normalidad de los residuos.

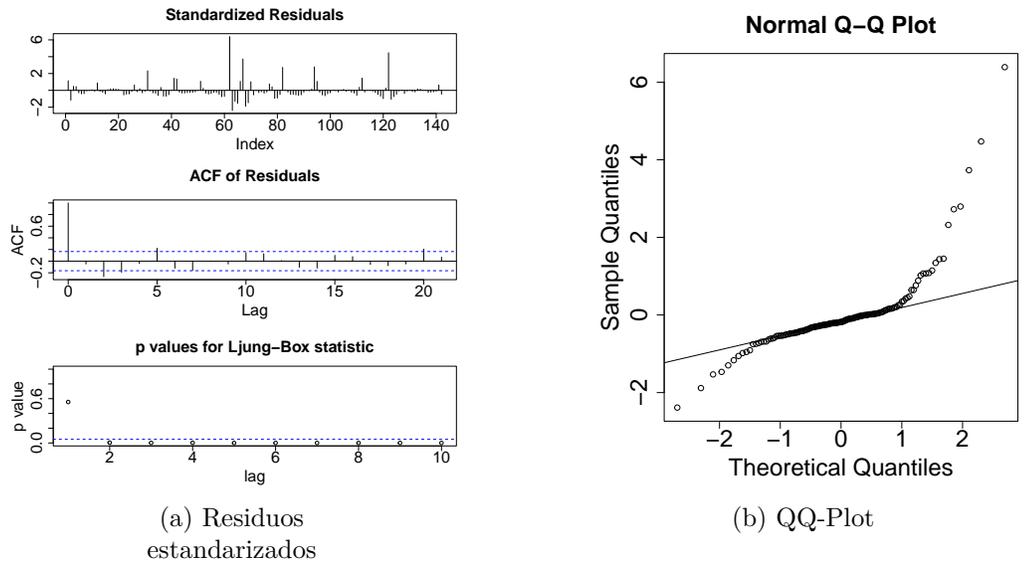


Figura 5.5: Residuos modelo LLT: Flujo de aire soplador.

### Velocidad separador

El ajuste en el caso de la variable Velocidad separador es bastante bueno, sin indicios de autocorrelaciones importantes en el gráfico ACF y con los 10 primeros *p-values* con suficiente significancia para rechazar la hipótesis nula de Ljung-Box.

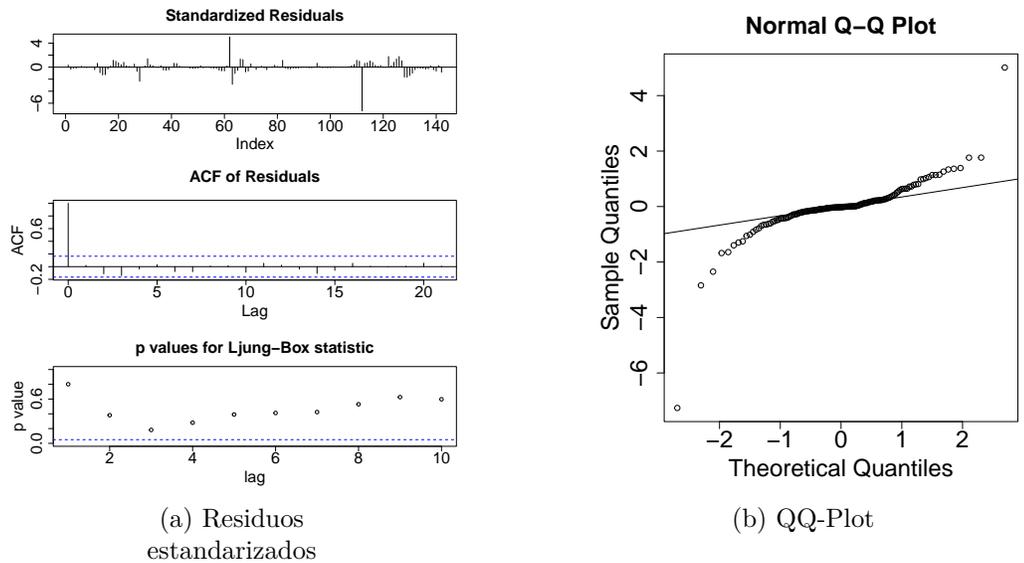
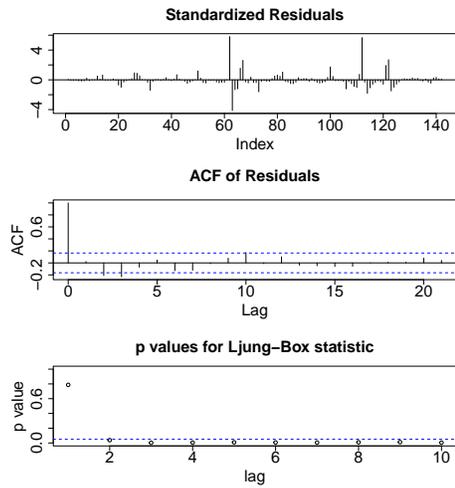


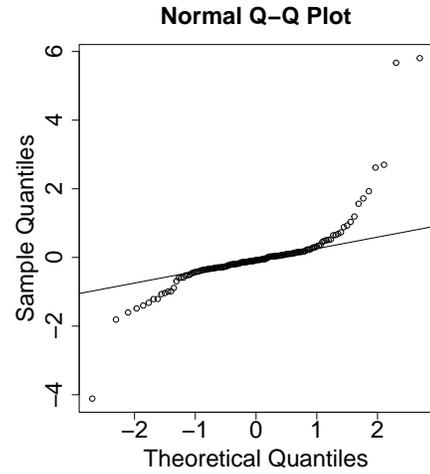
Figura 5.6: Residuos modelo LLT: Velocidad separador.

### Tasa envasado MIC 4870

Los resultados mostrados en la Figura 5.7 denotan un pobre ajuste, por lo que el modelo no captura con fidelidad eventuales relaciones existentes sobre los datos.



(a) Residuos estandarizados

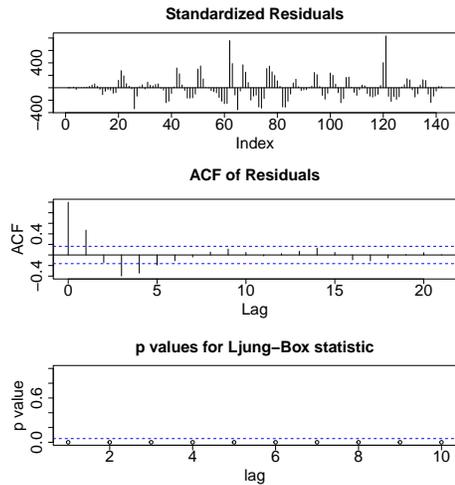


(b) QQ-Plot

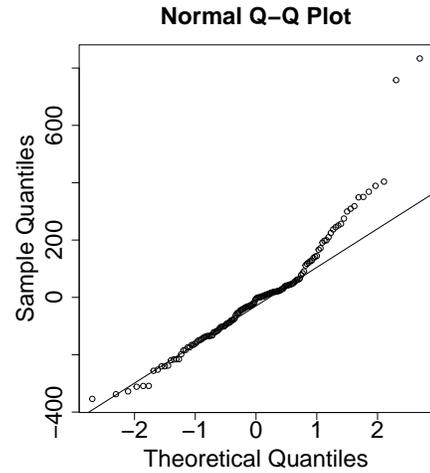
Figura 5.7: Residuos modelo LLT: Tasa envasado MIC 4870.

### Tasa envasado MIC 4880

De manera similar al caso anterior, la Figura 5.8 deja en evidencia un ajuste no adecuado.



(a) Residuos estandarizados



(b) QQ-Plot

Figura 5.8: Residuos modelo LLT: Tasa envasado MIC 4870.

### 5.1.1.3. Modelo GARCH

En lo que sigue se describen los resultados de ajuste para el modelo apARCH(1,1) descrito en la sección 3.5.

#### Flujo de aire soplador

Tras el ajuste apARCH para la variable Flujo de aire soplador, se obtienen los gráficos ACF (ver Figura 5.9) de los residuos estandarizados y estandarizados al cuadrado. En el primero de ellos se observa la presencia de autocorrelaciones importantes en los *lags* 4, 6 y 7. Por consiguiente, no se puede concluir que los residuos estandarizados no están correlacionados y son homocedásticos.

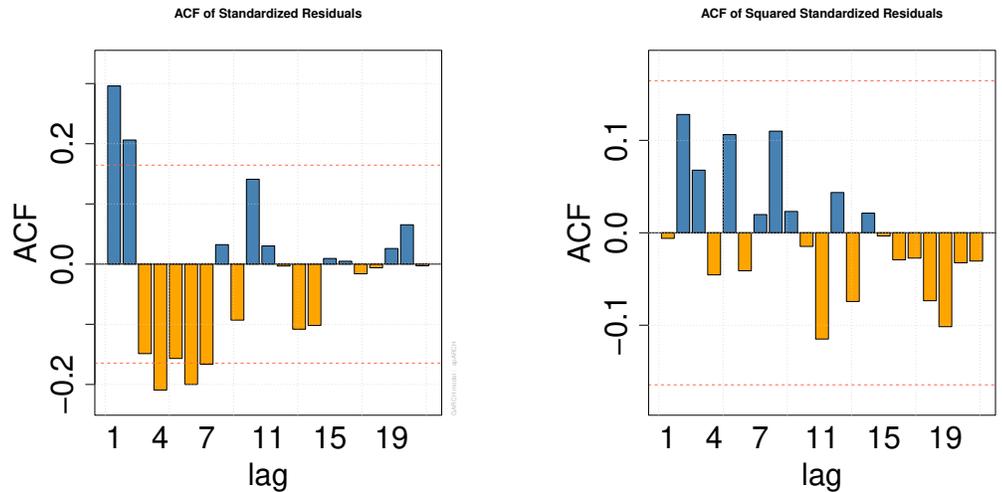


Figura 5.9: Residuos modelo apARCH: Flujo de aire soplador.

#### Velocidad separador

Para el caso de la segunda variable ajustable, Velocidad separador, es claro que existe autocorrelación importante sobre los residuos estandarizados, lo que se evidencia en la Figura 5.10. Luego, el modelo no ajusta correctamente los datos.

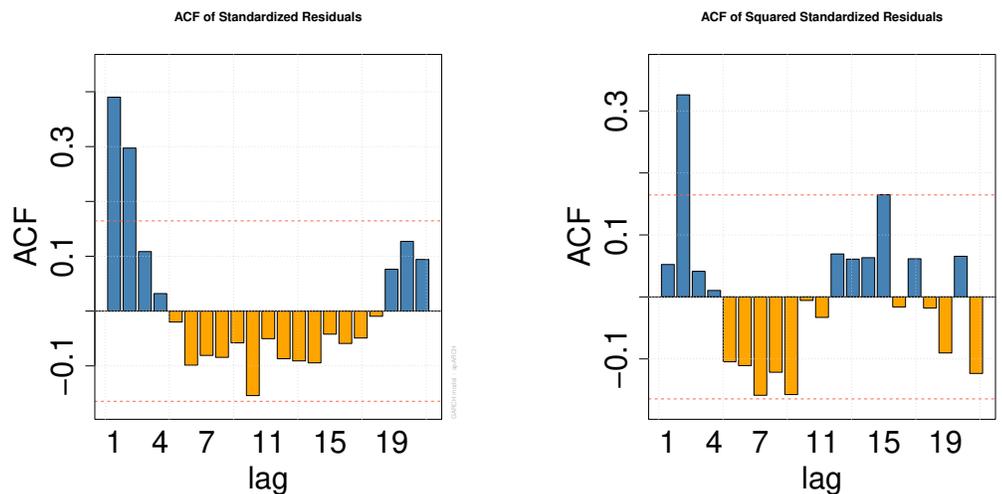


Figura 5.10: Residuos modelo apARCH: Velocidad separador.

## Tasa envasado MIC 4870

En el caso de la variable Tasa envasado MIC 4870 se tiene presencia importante de auto-correlaciones, por lo tanto, el modelado no es satisfactorio.

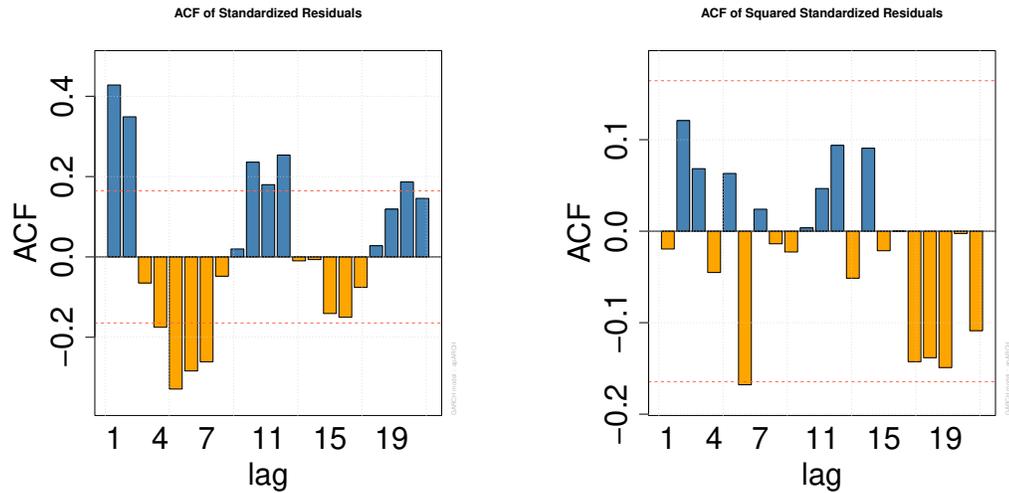


Figura 5.11: Residuos modelo apARCH: Tasa envasado MIC 4870.

## Tasa envasado MIC 4880

Finalmente, para la Tasa de envasado MIC 4880 no se obtienen buenos resultados dada una evidente autocorrelación remanente no capturada por el modelo, lo cual es reflejado en la Figura 5.12.

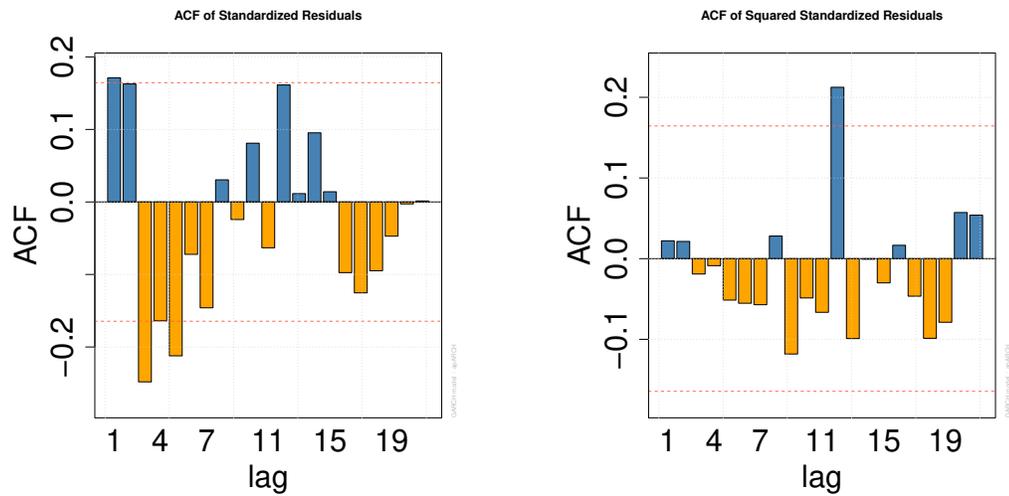


Figura 5.12: Residuos modelo apARCH: Tasa envasado MIC 4880.

### 5.1.2. *Forecasting out sample*

Una buena manera de probar las suposiciones de un modelo y comparar con otros su desempeño de pronóstico de manera realista, es realizar una validación fuera de la muestra. Esto significa retener algunos datos del proceso de identificación y estimación del modelo, y luego usar este último para predecir los datos retenidos con el objetivo de verificar su precisión y determinar si las estadísticas de sus errores son similares a las que hizo el modelo dentro de la muestra de datos que se ajustó.

#### Tasa envasado MIC 4870

En la Figura 5.13 se observa el contraste de los resultados de *forecasting* respecto al conjunto de validación. El modelo LLT es el que evidencia los peores resultados, pues muestra una recta con pendiente positiva que claramente solo aumenta con el horizonte de tiempo.

De la figura se observa que los peores modelos basados en DGBN son DGBN + restricciones + flujo.localtrend\_sep.localtrend, DGBN + restricciones + flujo.arma\_sep.localtrend y DGBN + restricciones + flujo.aparch\_sep.localtrend, lo cual indica que el *forecasting* realizado por el modelo LLT para la variable Velocidad separador tuvo un error considerable, que resultó en una atenuación del poder predictivo de las redes.

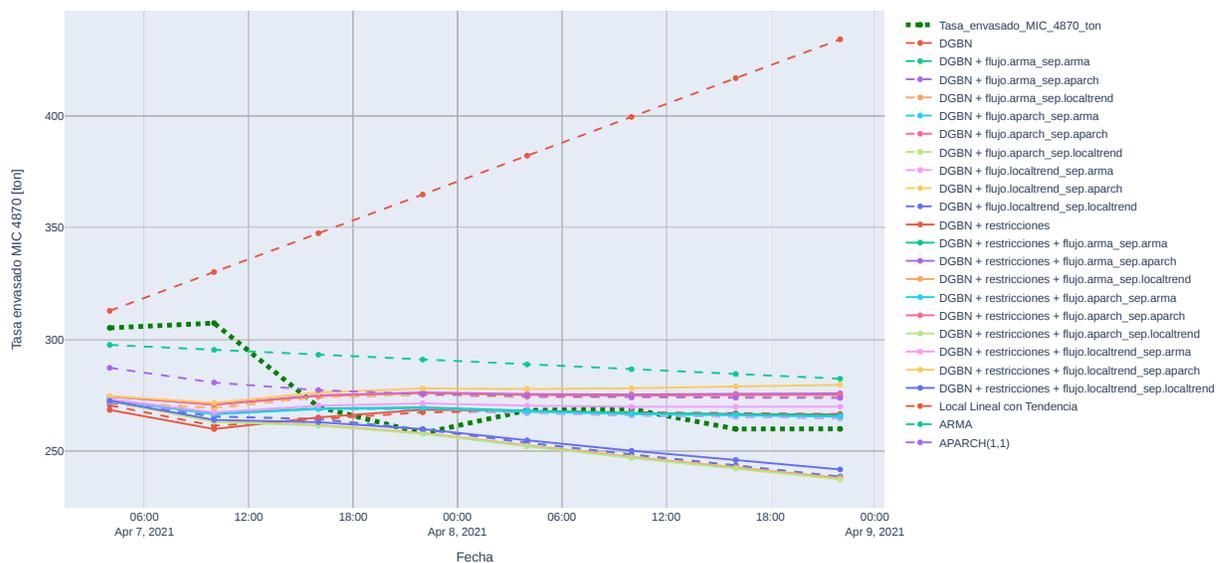


Figura 5.13: *Forecasting out of sample* sobre validación: Tasa envasado MIC 4870.

En la Tabla 5.1 se resumen las métricas obtenidas para el periodo estudiado. El modelo apARCH(1,1) obtiene los mejores resultados en cuanto a RMSPE con un 5%, lo cual es elocuente dado que proyecta una recta que sigue la tendencia. Sin embargo, la naturaleza univariable del modelo no permite un estudio de interacción de variables, por lo que, pese a sus buenos resultados, no es muy interesante para estudiar posibles correlaciones. También es claro que los primeros tres lugares en cuanto a performance están dominados por redes bayesianas con uso de conocimiento experto (o restricciones).

Tabla 5.1: Métricas de error: Tasa envasado MIC 4870

Modelo	MAE	MAPE	MSE	RMSE	RMSPE	MASE
APARCH(1,1)	13,63762	0,048966	231,0227	15,19943	0,053641	1,34014
DGBN + restricciones + flujo.aparch_sep.arma	12,43002	0,042488	361,31	19,00816	0,062872	1,221472
DGBN + restricciones + flujo.arma_sep.arma	12,45024	0,0426	361,6411	19,01686	0,062984	1,223458
DGBN + flujo.aparch_sep.arma	12,43645	0,042435	365,4219	19,11601	0,06314	1,222104
DGBN + flujo.localtrend_sep.arma	12,43645	0,042435	365,4219	19,11601	0,06314	1,222104
DGBN + flujo.arma_sep.arma	12,43645	0,042435	365,4219	19,11601	0,06314	1,222104
DGBN + restricciones + flujo.localtrend_sep.arma	13,59517	0,047057	373,9022	19,33655	0,064678	1,335969
DGBN + restricciones + flujo.aparch_sep.aparch	16,6842	0,05906	396,4809	19,91183	0,068419	1,639521
DGBN + restricciones + flujo.arma_sep.aparch	16,9864	0,060221	402,4208	20,06043	0,069105	1,669218
DGBN	13,88221	0,047436	440,583	20,99007	0,069269	1,364175
DGBN + flujo.aparch_sep.aparch	16,61131	0,058529	417,4503	20,4316	0,069694	1,632358
DGBN + flujo.localtrend_sep.aparch	16,61131	0,058529	417,4503	20,4316	0,069694	1,632358
DGBN + flujo.arma_sep.aparch	16,61131	0,058529	417,4503	20,4316	0,069694	1,632358
DGBN + restricciones	14,3468	0,048978	477,217	21,8453	0,072077	1,40983
DGBN + restricciones + flujo.localtrend_sep.aparch	18,78645	0,067112	447,1026	21,1448	0,073875	1,846105
DGBN + restricciones + flujo.localtrend_sep.localtrend	18,57657	0,065318	504,0583	22,45124	0,076588	1,82548
DGBN + flujo.aparch_sep.localtrend	19,35225	0,068333	529,6904	23,015	0,079216	1,901705
DGBN + flujo.arma_sep.localtrend	19,35225	0,068333	529,6904	23,015	0,079216	1,901705
DGBN + flujo.localtrend_sep.localtrend	19,35225	0,068333	529,6904	23,015	0,079216	1,901705
ARMA	20,15884	0,075231	460,0187	21,44805	0,081079	1,980966
DGBN + restricciones + flujo.arma_sep.localtrend	20,13969	0,071162	568,836	23,85028	0,082245	1,979085
DGBN + restricciones + flujo.aparch_sep.localtrend	20,46673	0,072386	581,6148	24,11669	0,083304	2,011223
Local Lineal con Tendencia	98,84638	0,373213	12884,34	113,5092	0,431641	9,713425

### Tasa envasado MIC 4880

Un patrón similar al caso anterior es en el cual el modelo LLT o localtrend atenúa el poder predictivo cuando es usado para *forecasting*, ya sea tanto para las variables objetivo como las de control, para su posterior uso como evidencia. De esto se concluye que no es un modelo apropiado.

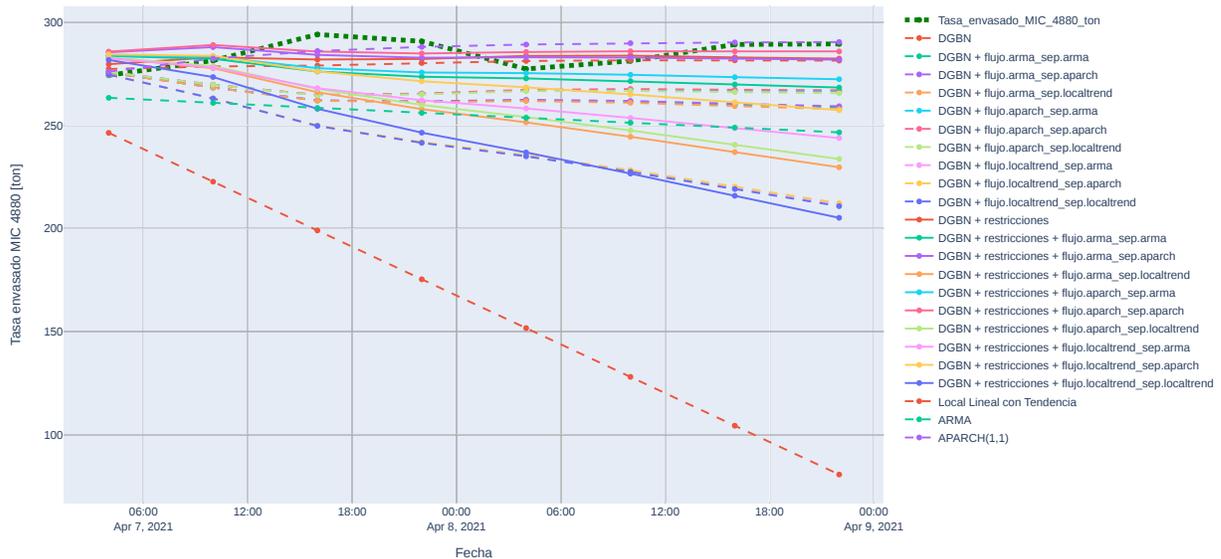


Figura 5.14: *Forecasting* sobre validación: Tasa envasado MIC 4880.

A pesar de ser univariado, el modelo apARCH(1,1) nuevamente resulta ganador en relación a la minimización de los errores. En cuanto a las redes bayesianas, se obtiene un mejor

comportamiento con restricciones y sin el uso de evidencia con RMSPE cercanos al 2%.

Tabla 5.2: Métricas error: Tasa envasado MIC 4880.

Modelo	MAE	MAPE	MSE	RMSE	RMSPE	MASE
APARCH(1,1)	4,477106	0,015796	36,48429	6,040223	0,021408	0,644725255
DGBN + restricciones	6,236198	0,02175	48,51356	6,965168	0,024134	0,898043119
DGBN + restricciones + flujo.aparch_sep.aparch	6,625309	0,023406	50,59041	7,112694	0,025269	0,954077021
DGBN + restricciones + flujo.arma_sep.aparch	7,165359	0,025152	58,53654	7,650917	0,026886	1,031846928
DGBN	6,4325	0,022291	61,83957	7,863814	0,027032	0,926311688
DGBN + restricciones + flujo.aparch_sep.arma	10,58101	0,036801	148,0467	12,16744	0,042109	1,523716871
DGBN + restricciones + flujo.arma_sep.arma	12,52117	0,043558	205,5089	14,33558	0,049613	1,803109752
DGBN + flujo.aparch_sep.aparch	16,99981	0,059044	363,736	19,07186	0,065861	2,448056794
DGBN + flujo.aparch_sep.arma	17,21572	0,059798	372,1922	19,29228	0,066632	2,479148129
DGBN + restricciones + flujo.localtrend_sep.aparch	16,87914	0,058815	370,8432	19,25729	0,06679	2,430679877
DGBN + flujo.aparch_sep.localtrend	17,59633	0,061126	387,5819	19,6871	0,068013	2,533958514
DGBN + flujo.arma_sep.aparch	21,07333	0,073268	546,6168	23,37984	0,080915	3,03466368
DGBN + flujo.arma_sep.arma	21,28924	0,074023	557,6071	23,6137	0,081733	3,065755015
DGBN + flujo.arma_sep.localtrend	21,66985	0,075351	577,7863	24,03718	0,083213	3,1205654
DGBN + restricciones + flujo.localtrend_sep.arma	24,91306	0,086813	803,9015	28,35316	0,098426	3,587602536
ARMA	29,8256	0,104098	991,741	31,49192	0,109451	4,295033405
DGBN + restricciones + flujo.aparch_sep.localtrend	28,75049	0,100223	1110,369	33,32221	0,11577	4,14021208
DGBN + restricciones + flujo.arma_sep.localtrend	30,94845	0,107896	1283,305	35,82324	0,124489	4,456729383
DGBN + flujo.localtrend_sep.aparch	44,03993	0,153484	2489,51	49,89499	0,173544	6,341966381
DGBN + flujo.localtrend_sep.arma	44,25583	0,154238	2515,201	50,15178	0,174441	6,373057716
DGBN + restricciones + flujo.localtrend_sep.localtrend	43,55268	0,151905	2559,884	50,5953	0,176018	6,27179954
DGBN + flujo.localtrend_sep.localtrend	44,63645	0,155567	2563,473	50,63076	0,176111	6,427868102
Local Lineal con Tendencia	121,1502	0,423483	17936,09	133,9257	0,467129	17,44621822

### 5.1.3. Estructuras de red

Se describen a continuación las estructuras DGBN encontradas en los datos de este periodo en particular. Se expone dos casos: el primero es aquel en que la red ha sido aprendida solo a partir de los datos disponibles, mientras que el segundo es el caso donde, además de los datos, se consideran restricciones impuestas (o conocimiento experto). Cabe notar que el sufijo  $t_n$  indica el retraso temporal respecto al punto actual. De esta forma  $t_{-1}$  indica un retraso temporal ( $t - 1$ ) de una unidad con respecto a  $t_0$ . El *forecasting in sample* a un paso para cada uno de los nodos de las redes es mostrado en la Figura A.1 del Anexo A.1.

#### Sin adición de conocimiento experto

En la Figura 5.15 se muestra la red markoviana de segundo orden<sup>1</sup>. De las relaciones de dependencia condicionales encontradas en este periodo, se destaca que la Tasa envasado MIC 4870 tiene una naturaleza puramente autorregresiva en su red intertemporal, es decir, su valor actual solo depende de sus valores anteriores. Mientras que intratemporalmente, tiene una dependencia con Tasa envasado MIC 4880.

En el caso de Tasa envasado MIC 4880 se detecta una influencia intratemporal de la variable Presión diferencial colector, y en cuanto a sus dependencias intertemporales también se tiene componente solo de carácter autorregresivo.

Las variables Corriente rotor y Corriente soplador también se comportan con patrones autorregresivos, denotando la falta de correlaciones significativas. El resto de dependencias se muestra de forma más clara en la Tabla 5.3.

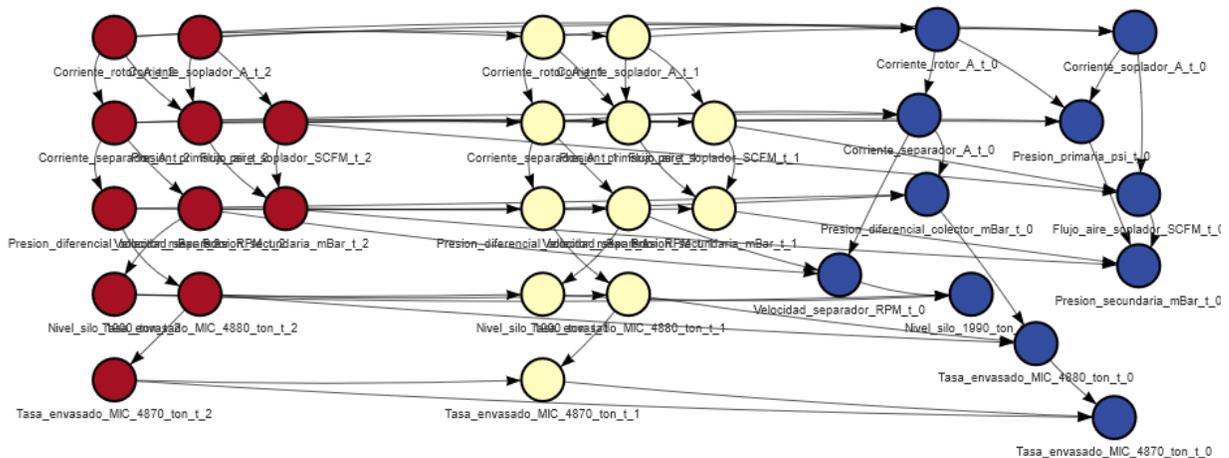


Figura 5.15: Estructura DGBN.

<sup>1</sup>  $t_0$  depende solo de  $t_{-1}$  y  $t_{-2}$

Tabla 5.3: DGBN sin restricciones: nodos t\_0 y sus padres

Padres	Nodo
Corriente_rotor_A_t_1 + Corriente_rotor_A_t_2	Corriente_rotor_A_t_0
Corriente_rotor_A_t_0 + Corriente_separador_A_t_1 + Corriente_separador_A_t_2	Corriente_separador_A_t_0
Corriente_soplador_A_t_1 + Corriente_soplador_A_t_2	Corriente_soplador_A_t_0
Corriente_soplador_A_t_0 + Presion_diferencial_colector_mBar_t_0 + Flujo_aire_soplador_SCFM_t_1 + Flujo_aire_soplador_SCFM_t_2	Flujo_aire_soplador_SCFM_t_0
Corriente_separador_A_t_0 + Nivel_silo_1990_ton_t_0 + Velocidad_separador_RPM_t_1 + Velocidad_separador_RPM_t_2	Velocidad_separador_RPM_t_0
Presion_primaria_psi_t_0 + Presion_secundaria_mBar_t_1 + Presion_secundaria_mBar_t_2 + Flujo_aire_soplador_SCFM_t_0	Presion_secundaria_mBar_t_0
Nivel_silo_1990_ton_t_1 + Nivel_silo_1990_ton_t_2 + Velocidad_separador_RPM_t_0	Nivel_silo_1990_ton_t_0
Corriente_soplador_A_t_0 + Presion_diferencial_colector_mBar_t_1 + Presion_diferencial_colector_mBar_t_2	Presion_diferencial_colector_mBar_t_0
Corriente_rotor_A_t_0 + Corriente_soplador_A_t_0 + Presion_primaria_psi_t_1 + Presion_primaria_psi_t_2	Presion_primaria_psi_t_0
Tasa_envasado_MIC_4870_ton_t_1 + Tasa_envasado_MIC_4870_ton_t_2 + Tasa_envasado_MIC_4880_ton_t_0	Tasa_envasado_MIC_4870_ton_t_0
Presion_diferencial_colector_mBar_t_0 + Tasa_envasado_MIC_4880_ton_t_1 + Tasa_envasado_MIC_4880_ton_t_2	Tasa_envasado_MIC_4880_ton_t_0

## Con adición de conocimiento experto

En la Figura 5.16 se presenta la red para el caso con restricciones y en la Tabla 5.4 se listan las relaciones de manera resumida. De esta última tabla, se extrae que las restricciones impuestas están presentes en la estructura resultante. Por supuesto, de forma similar al caso anterior, se mantiene la propiedad puramente autorregresiva para Corriente rotor y Corriente soplador.

Esto es esperable, dada una independencia condicional (de acuerdo a la red anterior) de estas últimas variables con respecto al conocimiento experto, por lo cual las restricciones no influyen de manera directa sobre esta subred.

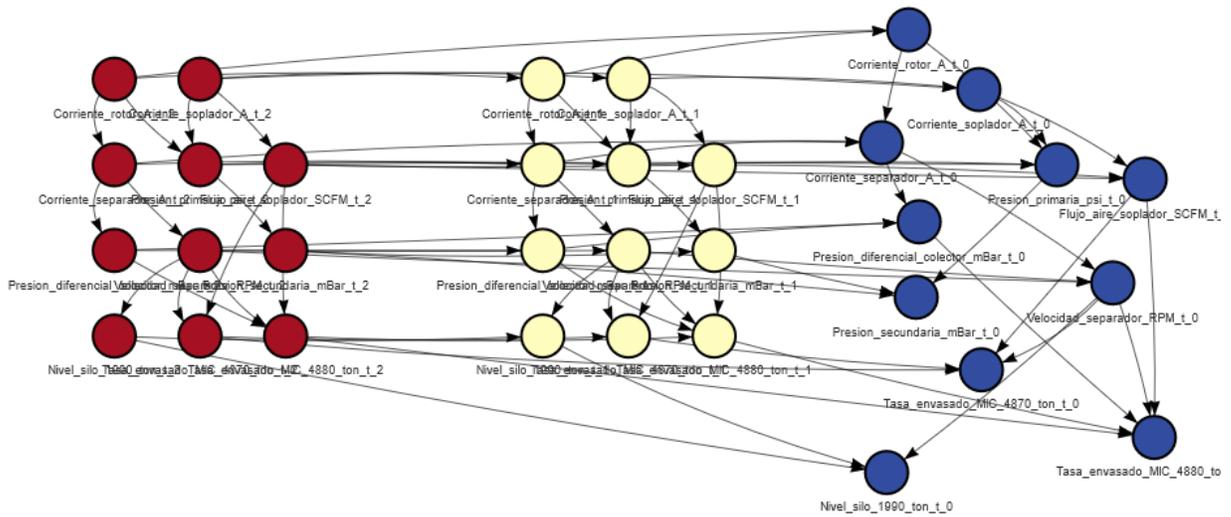


Figura 5.16: Estructura DGBN + Restricciones.

Tabla 5.4: DGBN con restricciones: nodos t\_0 y sus padres

Padres	Nodos
Corriente_rotor_A_t_1 + Corriente_rotor_A_t_2	Corriente_rotor_A_t_0
Corriente_rotor_A_t_0 + Corriente_separador_A_t_1 + Corriente_separador_A_t_2	Corriente_separador_A_t_0
Presion_primaria_psi_t_0 + Corriente_soplador_A_t_1 + Corriente_soplador_A_t_2	Corriente_soplador_A_t_0
Corriente_soplador_A_t_0 + Flujo_aire_soplador_SCFM_t_1 + Flujo_aire_soplador_SCFM_t_2	Flujo_aire_soplador_SCFM_t_0
Corriente_separador_A_t_0 + Velocidad_separador_RPM_t_1 + Velocidad_separador_RPM_t_2	Velocidad_separador_RPM_t_0
Presion_secundaria_mBar_t_1 + Presion_secundaria_mBar_t_2 + Presion_primaria_psi_t_0	Presion_secundaria_mBar_t_0
Nivel_silo_1990_ton_t_1 + Nivel_silo_1990_ton_t_2 + Velocidad_separador_RPM_t_0	Nivel_silo_1990_ton_t_0
Corriente_separador_A_t_0 + Presion_diferencial_colector_mBar_t_1 + Presion_diferencial_colector_mBar_t_2	Presion_diferencial_colector_mBar_t_0
Corriente_soplador_A_t_0 + Corriente_rotor_A_t_0 + Presion_primaria_psi_t_1 + Presion_primaria_psi_t_2	Presion_primaria_psi_t_0
Flujo_aire_soplador_SCFM_t_0 + Velocidad_separador_RPM_t_0 + Tasa_envasado_MIC_4870_ton_t_1 + Tasa_envasado_MIC_4870_ton_t_2	Tasa_envasado_MIC_4870_ton_t_0
Presion_diferencial_colector_mBar_t_0 + Flujo_aire_soplador_SCFM_t_0 + Velocidad_separador_RPM_t_0 + Tasa_envasado_MIC_4880_ton_t_1 + Tasa_envasado_MIC_4880_ton_t_2	Tasa_envasado_MIC_4880_ton_t_0

## 5.2. Periodo general

A continuación se muestran los resultados generales para la iteración de los modelos durante 30 días, es decir, la aplicación del experimento de periodo particular de manera consecutiva. En las Tablas 5.5, 5.6, 5.7 y 5.8 se muestra el cuartil 1 (q1), mediana y cuartil 3 (q3) de las distribuciones de error MAPE y RMSPE. Notar que las tablas son ordenadas de manera descendente de acuerdo con q3, luego los mejores modelos son los que tienen un q3 más bajo en cuanto a su error.

### Tasa envasado MIC 4870

De acuerdo a los resultados generales mostrados en las Tablas 5.5 y 5.6 se observa un mejor comportamiento de los modelos DGBN frente a los univariados. En particular, la red DGBN sin restricciones y sin uso de evidencia es la que tiene el mejor de los desempeños con una mediana de 9.138 % y 12.83 % como tercer cuartil para RMSPE. Esto da cuenta de que la red por sí misma hace mejor inferencia sobre los nodos ajustables que un modelo externo, es decir, ensambles.

Si bien en el periodo particular estudiado en la Sección 5.1.2 el modelo apARCH fue el mejor, este buen comportamiento no resultó ser replicable sobre un gran número de experimentos.

Modelo	q1	mediana	q3	Modelo	q1	mediana	q3
DGBN	6.094 %	7.519 %	10.10 %	DGBN	7.275 %	9.138 %	12.83 %
DGBN + flujo.aparch_sep.arma	5.993 %	7.496 %	10.95 %	DGBN + restricciones	7.608 %	9.437 %	12.98 %
DGBN + restricciones	6.152 %	8.194 %	11.22 %	DGBN + flujo.arma_sep.arma	6.840 %	9.138 %	13.10 %
DGBN + flujo.arma_sep.arma	5.993 %	7.496 %	11.31 %	DGBN + flujo.aparch_sep.arma	6.840 %	9.138 %	13.10 %
DGBN + flujo.localtrend_sep.arma	5.993 %	7.496 %	11.31 %	DGBN + flujo.localtrend_sep.arma	6.840 %	9.138 %	13.10 %
DGBN + flujo.aparch_sep.localtrend	6.678 %	7.776 %	11.40 %	DGBN + restricciones + flujo.aparch_sep.arma	7.040 %	10.05 %	13.46 %
DGBN + restricciones + flujo.arma_sep.arma	6.084 %	8.750 %	11.54 %	DGBN + restricciones + flujo.arma_sep.arma	6.643 %	10.03 %	13.63 %
DGBN + restricciones + flujo.aparch_sep.arma	6.025 %	8.764 %	11.55 %	DGBN + restricciones + flujo.arma_sep.localtrend	7.881 %	10.58 %	13.92 %
DGBN + restricciones + flujo.aparch_sep.localtrend	6.923 %	9.322 %	11.58 %	DGBN + restricciones + flujo.aparch_sep.localtrend	8.158 %	10.60 %	14.03 %
DGBN + flujo.arma_sep.localtrend	6.679 %	7.776 %	11.68 %	DGBN + flujo.arma_sep.localtrend	7.921 %	9.421 %	14.11 %
DGBN + restricciones + flujo.arma_sep.localtrend	6.960 %	9.309 %	11.76 %	DGBN + flujo.aparch_sep.localtrend	7.921 %	9.421 %	14.37 %
DGBN + flujo.localtrend_sep.localtrend	6.713 %	7.776 %	12.38 %	DGBN + flujo.localtrend_sep.localtrend	7.921 %	9.421 %	14.37 %
APARCH(1,1)	5.718 %	8.282 %	12.40 %	APARCH(1,1)	8.169 %	9.664 %	15.27 %
DGBN + restricciones + flujo.localtrend_sep.arma	6.708 %	10.71 %	12.75 %	DGBN + restricciones + flujo.localtrend_sep.arma	8.101 %	12.06 %	15.45 %
ARMA	8.002 %	12.59 %	14.51 %	DGBN + restricciones + flujo.localtrend_sep.localtrend	8.092 %	12.92 %	16.52 %
DGBN + restricciones + flujo.localtrend_sep.localtrend	7.379 %	11.01 %	14.75 %	ARMA	8.497 %	13.65 %	17.37 %
DGBN + flujo.localtrend_sep.aparch	11.06 %	18.01 %	24.92 %	DGBN + restricciones + flujo.aparch_sep.aparch	16.34 %	24.21 %	32.65 %
DGBN + flujo.aparch_sep.aparch	11.06 %	18.01 %	26.89 %	DGBN + restricciones + flujo.arma_sep.aparch	16.45 %	24.92 %	32.66 %
DGBN + flujo.arma_sep.aparch	11.06 %	18.01 %	26.89 %	DGBN + restricciones + flujo.localtrend_sep.aparch	17.18 %	22.54 %	32.71 %
DGBN + restricciones + flujo.arma_sep.aparch	13.66 %	20.23 %	28.52 %	DGBN + flujo.arma_sep.aparch	13.63 %	19.62 %	33.70 %
DGBN + restricciones + flujo.aparch_sep.aparch	13.28 %	19.28 %	29.12 %	DGBN + flujo.aparch_sep.aparch	13.63 %	19.62 %	33.70 %
DGBN + restricciones + flujo.localtrend_sep.aparch	14.42 %	20.51 %	30.94 %	DGBN + flujo.localtrend_sep.aparch	13.63 %	19.62 %	33.70 %
Local Lineal con Tendencia	15.75 %	28.07 %	54.52 %	Local Lineal con Tendencia	18.18 %	37.42 %	58.39 %

Tabla 5.5: MAPE Tasa envasado MIC 4870.

Tabla 5.6: RMSPE Tasa envasado MIC 4870.

Por un lado, en la Figura 5.17 se muestran las distribuciones de error para RMSPE obtenidos en los 30 periodos, para los 6 mejores modelos, donde destacan solo modelos multivariados. Por otro lado, la Figura 5.18, muestra un gráfico *box-plot* que refleja el conjunto de *p-values* obtenidos de la aplicación de prueba de la Kolmogorov-Smirnov, de lo que se concluye que los modelos DGBN y DGBN + restricciones son los más estables, teniendo una mediana inferior a un 5 % y menor cantidad de *outliers*. Luego, no es posible rechazar la hipótesis nula de la prueba introducida en la sección 3.10.4, con lo que se afirma que en la mayoría de las iteraciones los modelos antes mencionados producen un *forecasting* con

distribución similar a la del conjunto de validación.

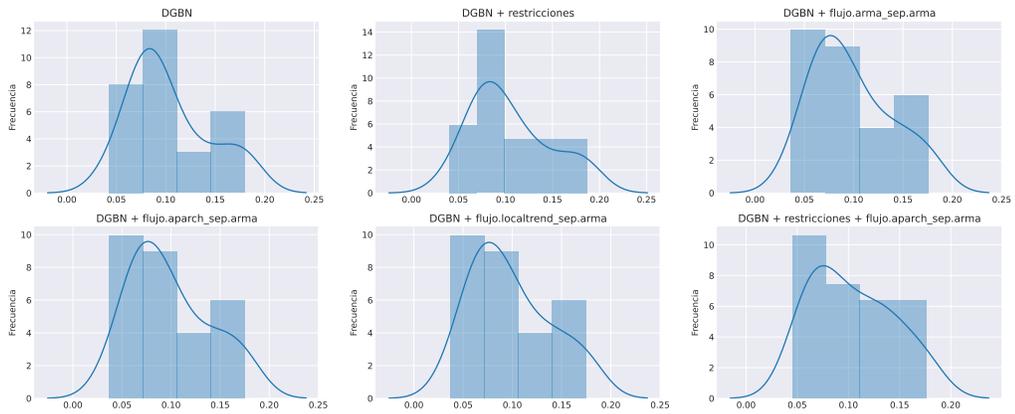


Figura 5.17: Distribución RMSPE: Tasa envasado MIC 4870.

Ahora bien, considerando tanto el criterio de minimización de RMSPE como de distribución de  $p$ -values, el mejor modelo es DGBN, que por lo demás tiene el cuartil 3 de errores más bajo y además el único sin *outliers* en la prueba de Kolmogorov-Smirnov.

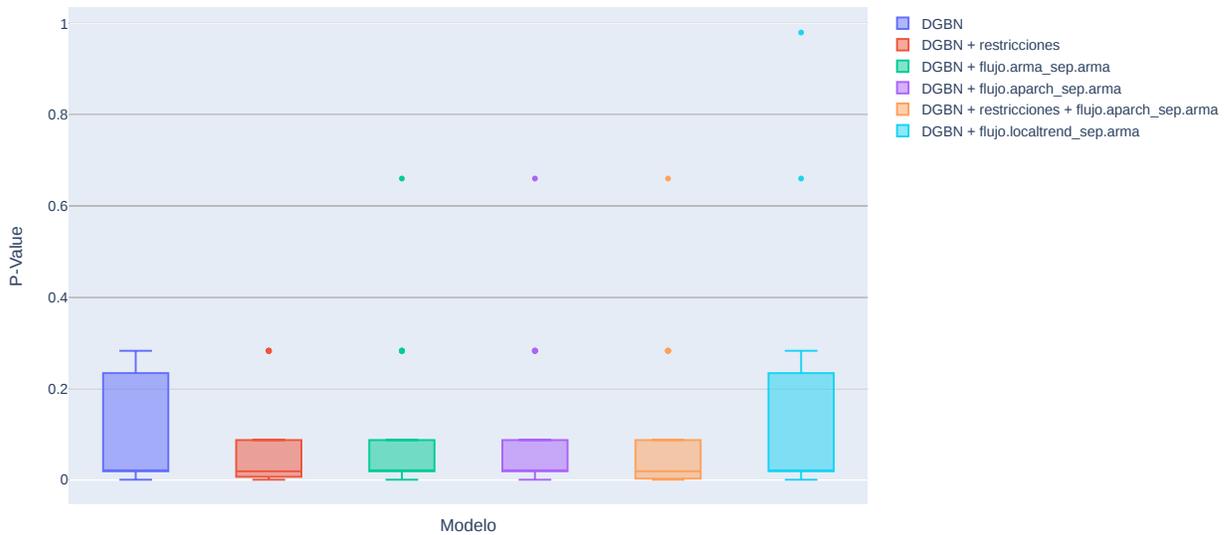


Figura 5.18: Prueba Kolmogorov-Smirnov  $p$ -values: Tasa envasado MIC 4870.

## Tasa envasado MIC 4880

Los resultados generales para Tasa envasada MIC 4880 se muestran en las Tablas 5.7 y 5.8, mientras que en la Figura 5.19 se encuentran las distribuciones de los 6 mejores modelos en RMSPE. De las métricas obtenidas se destaca, por un lado, que nuevamente los modelos multivariados, y en particular los que no usan evidencia externa, dominan los primeros lugares en cuanto a la minimización de error. Por otro lado, al igual que el caso anterior, los modelos ARMA y LLT muestran resultados en general inapropiados.

Modelo	q1	mediana	q3	Modelo	q1	mediana	q3
DGBN + restricciones	6.252 %	8.371 %	9.736 %	DGBN	7.411 %	9.589 %	13.00 %
DGBN + restricciones + flujo.aparch_sep.arma	6.103 %	8.026 %	9.833 %	DGBN + restricciones	6.965 %	9.785 %	13.28 %
DGBN + restricciones + flujo.arma_sep.arma	6.242 %	7.947 %	9.949 %	DGBN + flujo.aparch_sep.localtrend	6.965 %	9.758 %	13.28 %
DGBN	6.285 %	8.331 %	10.00 %	DGBN + flujo.arma_sep.localtrend	7.676 %	9.907 %	13.44 %
DGBN + flujo.aparch_sep.arma	6.282 %	8.355 %	10.10 %	DGBN + flujo.arma_sep.arma	7.676 %	9.936 %	13.44 %
DGBN + flujo.aparch_sep.localtrend	6.282 %	8.331 %	10.26 %	DGBN + restricciones + flujo.aparch_sep.arma	6.815 %	9.150 %	13.61 %
DGBN + flujo.arma_sep.arma	6.541 %	8.457 %	10.28 %	DGBN + restricciones + flujo.arma_sep.arma	6.902 %	9.648 %	13.63 %
DGBN + flujo.arma_sep.localtrend	6.488 %	8.434 %	10.44 %	DGBN + flujo.aparch_sep.arma	7.339 %	9.209 %	13.84 %
DGBN + restricciones + flujo.arma_sep.localtrend	6.325 %	9.483 %	11.15 %	DGBN + restricciones + flujo.aparch_sep.localtrend	7.409 %	10.36 %	14.06 %
DGBN + restricciones + flujo.aparch_sep.localtrend	6.396 %	9.385 %	11.27 %	APARCH(1,1)	6.953 %	9.598 %	14.13 %
APARCH(1,1)	6.218 %	8.811 %	11.70 %	DGBN + restricciones + flujo.arma_sep.localtrend	7.299 %	10.35 %	14.18 %
DGBN + flujo.arma_sep.aparch	6.537 %	8.337 %	13.14 %	DGBN + flujo.arma_sep.aparch	7.443 %	9.469 %	15.77 %
DGBN + flujo.aparch_sep.aparch	6.046 %	8.317 %	13.76 %	DGBN + flujo.aparch_sep.aparch	6.945 %	9.372 %	16.59 %
DGBN + flujo.localtrend_sep.arma	7.208 %	9.943 %	13.95 %	DGBN + flujo.localtrend_sep.arma	8.844 %	13.93 %	17.58 %
DGBN + flujo.localtrend_sep.localtrend	7.208 %	9.943 %	13.95 %	DGBN + flujo.localtrend_sep.localtrend	8.862 %	13.97 %	17.61 %
DGBN + restricciones + flujo.localtrend_sep.arma	6.906 %	9.482 %	14.67 %	DGBN + restricciones + flujo.localtrend_sep.arma	8.499 %	13.33 %	18.45 %
DGBN + restricciones + flujo.localtrend_sep.localtrend	6.958 %	10.48 %	15.19 %	DGBN + restricciones + flujo.localtrend_sep.localtrend	8.013 %	14.52 %	18.79 %
DGBN + flujo.localtrend_sep.aparch	7.090 %	12.45 %	17.68 %	DGBN + flujo.localtrend_sep.aparch	8.718 %	14.25 %	19.73 %
ARMA	7.998 %	11.00 %	20.17 %	ARMA	10.94 %	13.43 %	22.14 %
DGBN + restricciones + flujo.localtrend_sep.aparch	11.09 %	15.83 %	22.11 %	DGBN + restricciones + flujo.localtrend_sep.aparch	13.80 %	18.23 %	23.31 %
DGBN + restricciones + flujo.arma_sep.aparch	8.824 %	14.28 %	23.94 %	DGBN + restricciones + flujo.arma_sep.aparch	11.40 %	16.88 %	24.71 %
DGBN + restricciones + flujo.aparch_sep.aparch	8.792 %	14.11 %	25.01 %	DGBN + restricciones + flujo.aparch_sep.aparch	11.52 %	16.73 %	25.70 %
Local Lineal con Tendencia	16.97 %	39.26 %	55.24 %	Local Lineal con Tendencia	18.67 %	46.49 %	61.51 %

Tabla 5.7: MAPE Tasa envasado MIC 4880.

Tabla 5.8: RMSPE Tasa envasado MIC 4880.

En la Figura 5.20 se muestran los  $p$ -values obtenidos en la aplicación de la prueba de Kolmogorov-Smirnov, donde nuevamente el modelo de mayor estabilidad es DGBN en cuanto a distribución de  $p$ -values.

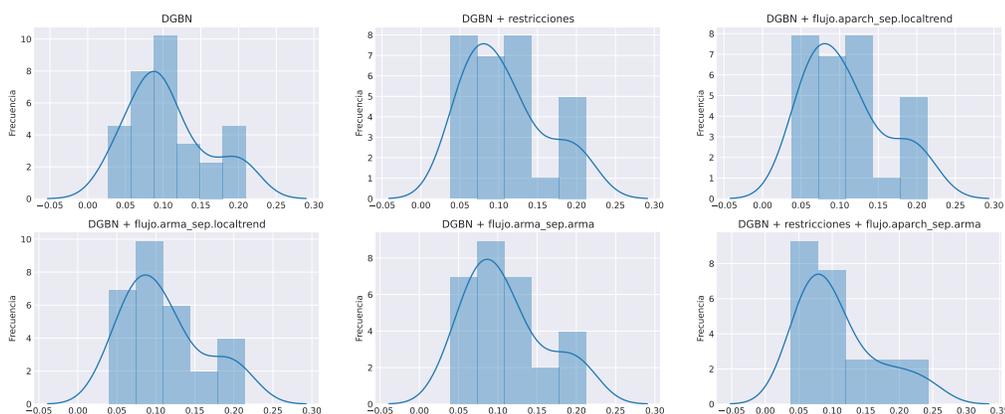


Figura 5.19: Distribución RMSPE: Tasa envasado MIC 4880.

Considerando el criterio de minimización de errores y estabilidad de  $p$ -values para la

prueba de Kolmogorov-Smirnov, se tiene que el mejor modelo es DGBN con una mediana 0.08 de  $p$ -values, que pese a tener un rango más amplio que su versión con restricciones, esta última tiene *outliers* que lo hace menos apropiados para un uso general.

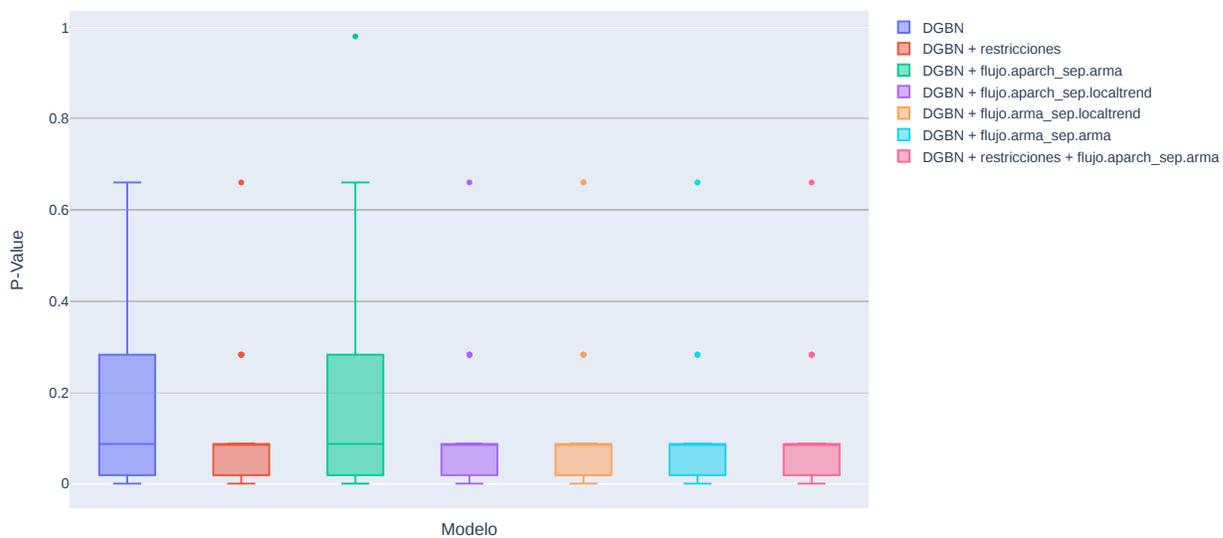


Figura 5.20: Prueba Kolmogorov-Smirnov  $p$ -values: Tasa envasado MIC 4880.

# Capítulo 6

## Discusión de resultados

Por una parte, en el periodo particular los modelos espacio estado no mostraron un ajuste satisfactorio, es decir, no capturaron la totalidad de los patrones contenidos en los datos, lo cual no implica que sean modelos con un pobre poder predictivo, sin embargo, podrían mejorar. En el caso de apARCH, aun cuando el ajuste no es perfecto para las tasas de envasado, se obtienen buenos resultados en las métricas de error. Lo anterior es esperable dadas sus proyecciones casi constantes, algo no muy diferente de lo que haría un predictor *naive*, que pronostica como valor futuro el último valor observado, que aunque pueda resultar en una minimización del error cuadrático, no captura realmente la dinámica del sistema ni las eventuales relaciones entre variables, factor importante debido a la presencia de variables de control. Ahora bien, cabe indicar que el desempeño de los modelos depende directamente de la ventana de datos considerados, por lo que resulta fundamental el estudio consecutivo sobre un periodo extenso para reconocer buenos desempeños en general, más allá de los casos aislados.

Por otra parte, en el periodo general los modelos espacio estado ARMA y LLT mostraron malos resultados sobre las 30 iteraciones, lo cual es razonable dado que 24 pasos de predicción resultan ser excesivos para el algoritmo de Kalman. Es necesario considerar que al no tener observaciones reales durante el *forecasting out of sample*, y por lo tanto, el uso de las mismas predicciones anteriores como entrada, el algoritmo no corrige de manera idónea el paso siguiente (ver factor de corrección en ecuación 3.23), ocasionando una continua adición de errores y, en consecuencia, una atenuación del poder predictivo con el aumento del horizonte *out of sample*. El uso de modelos de espacio estado tiene un muy buen desempeño para el paso siguiente (*in sample*), sin embargo, bajo la condición de obtener datos actualizados hasta el día anterior y pretender predicciones al menos para el día presente, los modelos espacio estado y su ajuste a través de filtro de Kalman resultaron no ser apropiados.

Ahora bien, en el caso de que en el futuro la actualización de datos aumente su frecuencia, se propone reintentar con este tipo de modelos. En el caso de apARCH, que aunque obtuvo buenos resultados en el periodo particular, este no pudo mantener tal rendimiento ante el dinamismo intrínseco del proceso marcado por el continuo deslizamiento de la ventana de entrenamiento. En consecuencia, se concluye que los modelos univariados para la tasa de envasado no son apropiados dado su pobre poder predictivo sostenido sobre la iteración continua y su nula capacidad explicativa en el proceso estudiado.

El efecto de la adición de conocimiento experto sobre las redes DGBN en el periodo general muestra resultados similares para ambas variables objetivo, ya que la inclusión de restricciones empeora levemente los resultados, produciendo un incremento en los errores y la aparición de *outliers* en la prueba de Kolmogorov-Smirnov. Sin embargo, en el caso de querer estudiar la influencia de las variables de control se debe añadir el conocimiento experto para forzar la inclusión de los arcos deseados. Por esto, a pesar de que DGBN + restricciones es el segundo mejor modelo, es el que se usa como simulador dada la inclusión de arcos desde las variables control hacia las variables objetivos.

Finalmente, se tiene que el uso de modelos externos para el modelamiento de las variables de control no es mejor que el *forecasting* hecho por las propias DGBN. De aquí se concluye que la mejor opción es no usar modelos ensambles.

## 6.1. Uso de red como simulador

Tras comprobar con un gran número de experimentos que las redes DGBN resultan ser el tipo de modelo más apropiado en la modelación del proceso estudiado, se pretende entender cómo afectan las variables de control de acuerdo con el modelo propuesto. Para lograrlo, se usa la red aprendida con restricciones mostrada en la Figura 5.16. De esta manera, la red permite ser utilizada como un modelo de simulación que estudia la interacción entre las distintas variables representadas por nodos. Se estudia el efecto de la variación de un 5% en las variables de control con respecto a su promedio histórico, que para flujo aire soplador SCFM y velocidad separador RPM son 4713 y 1346, respectivamente.

En la Figura 6.1 se observa que un incremento en la velocidad separador produce un aumento en la tasa de envasado 4870, mientras que un aumento del flujo aire soplador genera el comportamiento opuesto.

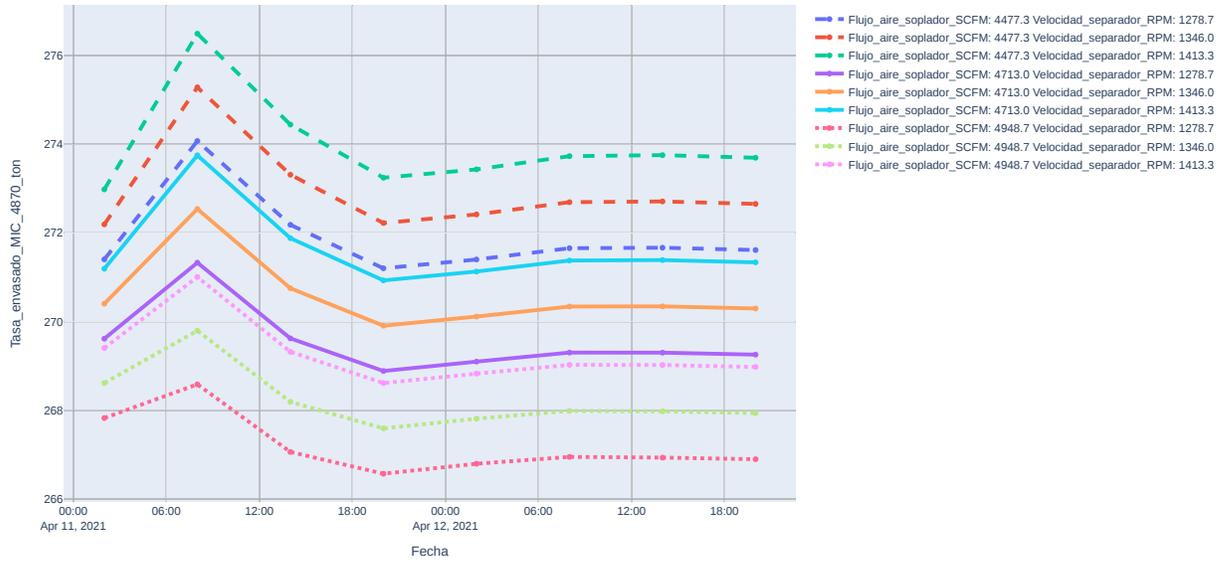


Figura 6.1: Simulación: Tasa envasado MIC 4870.

Ahora bien, en el caso de la Figura 6.2 se observa claramente que un incremento en el flujo de aire genera un incremento de la tasa de envasado MIC 4880. Por otro lado, el efecto del aumento de la velocidad separador es leve, pero con tendencia a la disminución de la variable objetivo.

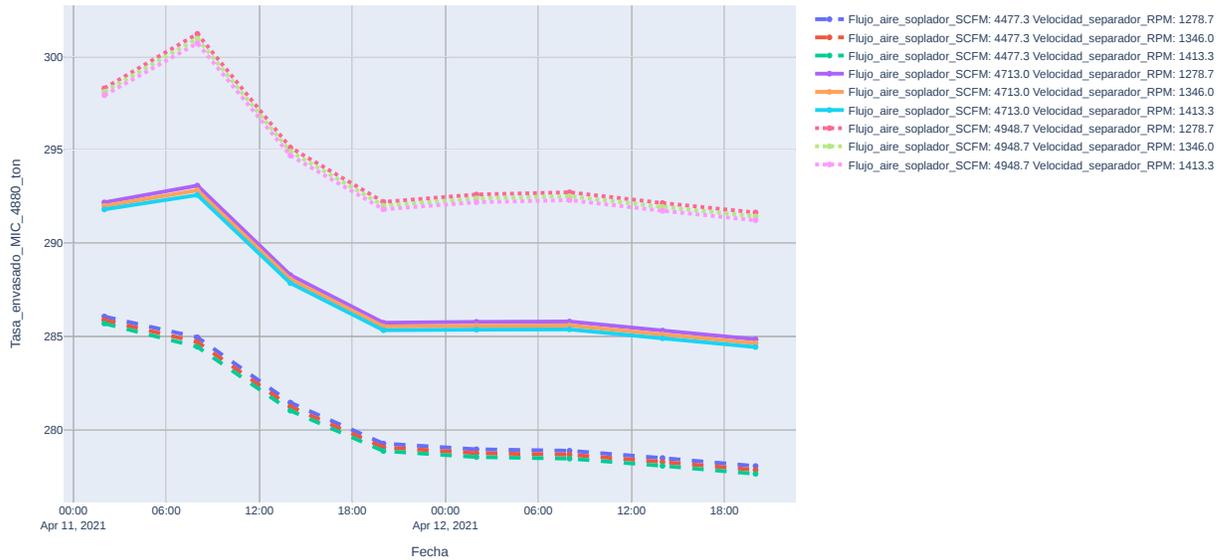


Figura 6.2: Simulación: Tasa envasado MIC 4880.

Del análisis anterior se entiende que una modificación en las variables de control produce comportamientos opuestos en cada tasa de envasado. Luego, considerando que ambas tasas

funcionan de manera simultánea, se procede a estudiar su comportamiento en conjunto con ayuda de la diferencia dada por la siguiente relación:

$$\text{Diferencia} = \text{Tasa\_envasado\_MIC\_4880\_ton} - \text{Tasa\_envasado\_MIC\_4870\_ton} \quad (6.1)$$

En la Figura 6.3 se presenta el comportamiento conjunto donde se observa que el incremento de la velocidad separador produce una reducción de la diferencia, mientras que el aumento del flujo de aire soplador produce un aumento en la diferencia resultante. Luego, se comprueba que las alteraciones en las variables de control generan comportamientos opuestos en las tasas de envasado, por lo que la elección de los parámetros adecuados depende de los requerimientos de producción particulares.

Por consiguiente, del análisis anterior se deducen las siguientes reglas para el aumento de cada variable objetivo:

- **Tasa envasado MIC 4870:** se debe incrementar la velocidad separador y reducir el flujo de aire de separador.
- **Tasa envasado MIC 4880:** se debe reducir la velocidad separador y aumentar el flujo de aire soplador.

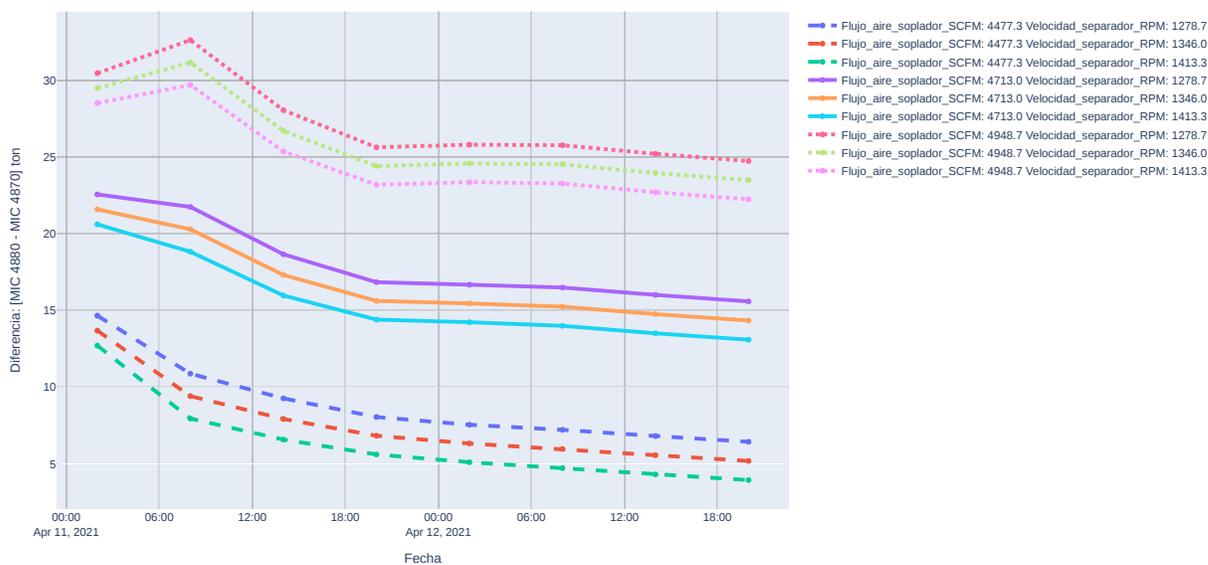


Figura 6.3: Simulación: Diferencia tasas de envasado.

# Capítulo 7

## Conclusiones

El objetivo de este trabajo es la modelación del proceso de micronizado - envasado de carbonato de litio de la planta MIC 4 mediante redes bayesianas y modelos espacio estado. La metodología desarrollada y propuesta presenta como resultado un módulo de preprocesado - *forecasting* ejecutado diariamente para la generación de historial, una primera aplicación web que permite la experimentación de los distintos modelos propuestos sin la necesidad de escribir códigos con parámetros configurables como orden markoviano, ventana de entrenamiento, arcos obligatorios, arcos prohibidos, entre otros. Finalmente, una segunda aplicación web que permite la lectura de resultados generados por el módulo automatizado descrito anteriormente, donde ambas aplicaciones están conectadas a las bases de datos del proceso, con lo cual se tiene acceso a los últimos datos disponibles.

Con respecto a la modelación del proceso se comprueba que los modelos de tipo espacio estado y su ajuste mediante filtro Kalman no resultan apropiados para el proceso estudiado, dado que se requiere un horizonte de *forecasting* demasiado amplio debido a la frecuencia de actualización de la base de datos. El modelo que obtuvo los mejores resultados fue DGBN sin restricciones, sin embargo, se opta por su versión con uso de arcos obligatorios para el estudio de la influencia de las variables de control sobre las de objetivo. Este análisis muestra que la alteración de las variables control ocasiona comportamientos opuestos sobre las tasas de envasado. Luego, el ajuste se debe realizar sujeto a requerimientos específicos del cliente. Ahora bien, el modelo DGBN propuesto puede ser usado tanto para labores de predicción como de control, donde este último es ajustado paso a paso en el caso disponer de datos con frecuencia cercana al tiempo real.

Los objetivos han sido cumplidos, sin embargo, para un mejor desempeño y resultados más cercanos a la operación real se recomienda incrementar la frecuencia de actualización de las bases de datos, y de esta manera reducir el horizonte *out of sample*. Además, con el fin de probar distintas alternativas de modelación se propone como trabajo futuro un modelamiento basado en modelos jerárquicos del tipo bayesiano y el uso de ensamblados de modelos.

## Trabajos futuros

Si bien se ha cumplido con modelar la planta mediante un modelo DGBN, este tipo de modelos asume comportamiento normal de las variables del proceso. Luego, se propone extender el análisis mediante modelación bayesiana jerárquica donde se combina medidas

reales con creencias o *prior* expresadas en forma de probabilidades. Además, se proponen modelos de ensambles para la combinación de múltiples hipótesis, bajo la premisa que la combinación de múltiples modelos entrenados para un mismo problema puede dar mejores resultados que cada uno por si solo. En lo que sigue se profundiza sobre posibles extensiones.

## Inferencia bayesiana y modelos jerárquicos

Un modelo especifica la probabilidad de valores particulares dado la estructura del modelo y valores de los parámetros.

El modelo se estima a partir de los datos y parámetros como:

$$p(\text{datos} \mid \text{valores de parámetros y estructura del modelo})$$

y usamos la regla de Bayes para convertir la expresión anterior a que tanta certidumbre tenemos del modelo condicional a los datos:

$$p(\text{valores de parámetros y estructura del modelo} \mid \text{datos})$$

Entonces el procedimiento de inferencia bayesiana es:

- Cuantificamos la información (o incertidumbre) acerca del parámetro desconocido  $\theta$  mediante las distribuciones de probabilidad.
- Antes de observar datos  $x$ , cuantificamos la información de  $\theta$  externa a  $x$  en una *distribución a priori*:

$$p(\theta)$$

esto es, la distribución a priori resume la información externa acerca del parámetro de los datos, que en el caso del trabajo podría considerar la distribución normal sobre la tasa de envasado con datos históricos. Por otra parte, cuantificamos la información de  $\theta$  asociada a  $x$  mediante la *distribución de verosimilitud*

$$p(x|\theta)$$

Combinamos la información a priori y la información que provee  $x$  mediante el teorema de Bayes obteniendo así la *distribución posterior*

$$p(\theta|x) \propto p(x|\theta)p(\theta)$$

- Las inferencias se obtienen de los resúmenes estadísticos de la distribución posterior, como medias y varianzas.

Vemos que la distribución posterior es proporcional al producto de la verosimilitud y la distribución inicial, el denominador  $p(x)$  no depende de  $\theta$  por lo que es constante (como función de  $\theta$ ) y esta ahí para normalizar la distribución posterior asegurando que tengamos una distribución de probabilidad.

La estimación de los parámetros implica determinar hasta que punto son fiables los posibles valores a priori del parámetro. En estadística bayesiana la estimación se realiza con la

distribución posterior sobre los valores de los parámetros  $\theta$ .

La predicción de valores o la probabilidad de datos futuros, un dato  $y$  (no observado) se determina promediando las probabilidades predictivas de los datos a lo largo de todos los posibles valores de los parámetros y ponderados por la distribución a priori de los valores de los parámetros. Cuando solo contamos con estimaciones iniciales de los parámetros se tiene:

$$p(y) = \int p(y|\theta)p(\theta)d\theta$$

Los modelos jerárquicos involucran varios parámetros de tal manera que las estimaciones de uno de los parámetros dependen de manera significativa de los valores de otros parámetros.

En el caso jerárquico no se puede derivar la distribución posterior de manera analítica, pero si los parámetros e hiperparámetros toman un número finito de valores y no hay muchos de ellos, podemos aproximar la distribución posterior usando distintas metodologías de simulaciones como la familia de métodos de Montecarlo para la aproximación numérica vía resúmenes estadísticos de las simulaciones.

Enumeramos algunas de las ventajas potenciales de usar un modelo jerárquico:

- Modelo unificado, además de elegir entre distintos modelos por ranking de métricas de errores, podemos aunar los modelos a través de modelos de los parámetros de estos modelos de manera conjunta.
- Incertidumbre en los parámetros, estimar una distribución a los hiperparámetros,  $\theta$ , nos permite incorporar nuestra incertidumbre sobre la distribución inicial de los parámetros de los datos.
- Cómputo, la estructura jerárquica facilita los cálculos posteriores, debido a que la distribución posterior se factoriza en distribuciones condicionales más sencillas que facilitan la implementación de por ejemplo el método de Gibbs.

## Modelos de ensamble

Los modelos de ensamble consideran múltiples hipótesis simultáneamente para formar una hipótesis que podría tener un mejor desempeño. Las combinaciones que hacen uso de múltiples hipótesis pertenecientes a una misma familia de modelos se consideran modelos de ensamble mientras que se usa el término más general de sistemas de aprendizaje múltiples cuando las hipótesis que se combinan provienen de diversas familias.

Debido a que los métodos combinados hacen uso de varias hipótesis simultáneas, se produce una mayor exigencia en el cómputo, por lo que se usan algoritmos rápidos como espacio de hipótesis base, como son los árboles de decisión.

Una combinación de algoritmos de aprendizaje supervisado es en sí mismo un algoritmo de aprendizaje supervisado y puede ser entrenado y usado para hacer predicciones. Sin embargo, se debe tener en cuenta que una combinación de hipótesis (algoritmos) de una determinada familia no es necesariamente una hipótesis (algoritmo) de la misma familia, por

lo que podríamos obtener mejores resultados que con los elementos individuales de la familia, aunque también se tiene el riesgo de obtener un modelo sobre ajustado si no se tienen algunas precauciones. En la práctica, la forma en que se seleccionan los modelos individuales que se combinan hacen uso de algunas técnicas que tienden a reducir los problemas relacionados con el exceso de ajuste de los datos de entrenamiento y mejoran la predicción conjunta.

Empíricamente, se ha comprobado que cuando existe una diversidad significativa entre los modelos individuales, las combinaciones tienden a obtener mejores resultados, por lo que muchos de los métodos existentes buscan promover la diversidad entre los modelos que se combinan, y por ello se usan modelos de manera aleatoria que modelos dirigidos y que funcionan mejor individualmente.

# Glosario

**ACF** Autocorrelation Function.

**ARIMA** Autoregressive Integrated Moving Average.

**ARMA** Autoregressive Moving Average.

**BIC** Bayesian information criterion.

**BN** Bayesian Network.

**DAG** Directed Acyclic Graph.

**DBN** Dynamic Bayesian Network.

**DGBN** Dynamic Gaussian Bayesian Network.

**DLM** Dynamic Linear Model.

**DMMHC** Dynamic Max Min Hill Climbing.

**GARCH** Autoregressive Conditional Heteroskedasticity.

**HMM** Hidden Markov Model.

**ISPM** Integrated System Predictive Maintenance.

**KFM** Kalman Filtering Method.

**MAPE** Mean Absolute Percentage Error.

**MBC** Mantenimiento Basado en la Condición.

**MDL** Minimum Description Length.

**MLE** Maximum Likelihood Estimation.

**PDF** Probability Density Function.

**PdM** Predictive Maintenance.

**RMSPE** Root Mean Square Percentage Error.

# Bibliografía

- [1] G. A. Susto, A. Beghi, and C. De Luca, “A predictive maintenance system for epitaxy processes based on filtering and prediction techniques,” *IEEE Transactions on Semiconductor Manufacturing*, vol. 25, no. 4, pp. 638–649, 2012.
- [2] B. Jones, I. Jenkinson, Z. Yang, and J. Wang, “The use of bayesian network modelling for maintenance planning in a manufacturing industry,” *Reliability Engineering & System Safety*, vol. 95, no. 3, pp. 267–277, 2010.
- [3] Z. A. Bukhsh and I. Stipanovic, “Predictive maintenance for infrastructure asset management,” *IT Professional*, vol. 22, no. 5, pp. 40–45, 2020.
- [4] M. Haarman, M. Mulders, and C. Vassiliadis, “Predictive maintenance 4.0-predict the unpredictable,” *PwC and Mainnovation*, 2017.
- [5] C. C. del Cobre, “Mercado internacional del litio y su potencial en chile,” 2019.
- [6] P. Maxwell and M. Mora, “Lithium and chile: looking back and looking forward,” *Mineral Economics*, vol. 33, no. 1, pp. 57–71, 2020.
- [7] M. Lampis and J. Andrews, “Bayesian belief networks for system fault diagnostics,” *Quality and Reliability Engineering International*, vol. 25, no. 4, pp. 409–426, 2009.
- [8] A. Muller, M.-C. Suhner, and B. Iung, “Formalisation of a new prognosis model for supporting proactive maintenance implementation on industrial system,” *Reliability Engineering & System Safety*, vol. 93, no. 2, pp. 234–253, 2008.
- [9] Z. Zhang and F. Dong, “Fault detection and diagnosis for missing data systems with a three time-slice dynamic bayesian network approach,” *Chemometrics and Intelligent Laboratory Systems*, vol. 138, pp. 30–40, 2014.
- [10] A. D. Lewis and K. M. Groth, “A dynamic bayesian network structure for joint diagnostics and prognostics of complex engineering systems,” *Algorithms*, vol. 13, no. 3, p. 64, 2020.
- [11] M. Gondran, M. Minoux, and S. Vajda, *Graphs and algorithms*. John Wiley & Sons, Inc., 1984.
- [12] P. Campagnoli, G. Petris, and S. Petrone, *Dynamic Linear Models with R*. Springer, 2009.
- [13] K. P. Murphy, “Dynamic bayesian networks: representation, inference and learning,” 2002.
- [14] P. Sevestre and A. Trognon, “Dynamic linear models,” in *The Econometrics of Panel Data*, pp. 120–144, Springer, 1996.

- [15] R. B. Litterman, “A random walk, markov model for the distribution of time series,” *Journal of Business & Economic Statistics*, vol. 1, no. 2, pp. 169–173, 1983.
- [16] J. Durbin and S. J. Koopman, *Time series analysis by state space methods*. Oxford university press, 2012.
- [17] P. de Jong and J. Penzer, “The arma model in state space form,” *Statistics & probability letters*, vol. 70, no. 1, pp. 119–125, 2004.
- [18] A. Castro Rojas, *Modelamiento de tráfico en nodos de acceso de banda ancha*. Departamento de Ingeniería Eléctrica. Universidad de Chile., 2012.
- [19] J. Zuñiga, *Comparación de la volatilidad ex-ante entre un modelo APARCH y un modelo de Volatilidad Estocástica con Apalancamiento respecto al estimador de Yang & Zhang*. PhD thesis, 11 2019.
- [20] H. Chen, J. Zhang, Y. Tao, and F. Tan, “Asymmetric garch type models for asymmetric volatility characteristics analysis and wind power forecasting,” *Protection and Control of Modern Power Systems*, vol. 4, no. 1, pp. 1–11, 2019.
- [21] N. Friedman, D. Geiger, and M. Goldszmith, “Bayesian network classifiers machine learning 29, 131-163 (1997) 10.1023,” *A: 1007465528199*, 1997.
- [22] G. Bonaccorso, *Mastering Machine Learning Algorithms: Expert techniques for implementing popular machine learning algorithms, fine-tuning your models, and understanding how they work*. Packt Publishing Ltd, 2020.
- [23] B. Cai, Y. Liu, Z. Liu, Y. Chang, and L. Jiang, *Bayesian networks for reliability engineering*. Springer, 2020.
- [24] M. Hartwig and R. Möller, “How to encode dynamic gaussian bayesian networks as gaussian processes?,” in *Australasian Joint Conference on Artificial Intelligence*, pp. 371–382, Springer, 2020.
- [25] G. F. Cooper and E. Herskovits, “A bayesian method for the induction of probabilistic networks from data,” *Machine learning*, vol. 9, no. 4, pp. 309–347, 1992.
- [26] M. Scutari and J.-B. Denis, *Bayesian networks: with examples in R*. CRC press, 2021.
- [27] G. Trabelsi, P. Leray, M. B. Ayed, and A. M. Alimi, “Dynamic mmhc: A local search algorithm for dynamic bayesian network structure learning,” in *International Symposium on Intelligent Data Analysis*, pp. 392–403, Springer, 2013.
- [28] F. P. Santos and C. D. Maciel, “A pso approach for learning transition structures of higher-order dynamic bayesian networks,” in *5th ISSNIP-IEEE Biosignals and Birobotics Conference (2014): Biosignals and Robotics for Better and Safer Living (BRC)*, pp. 1–6, IEEE, 2014.
- [29] F. Pascal, L. Bombrun, J.-Y. Tourneret, and Y. Berthoumieu, “Parameter estimation for multivariate generalized gaussian distributions,” *IEEE Transactions on Signal Processing*, vol. 61, no. 23, pp. 5960–5971, 2013.
- [30] L. E. Sucar, “Probabilistic graphical models,” *Advances in Computer Vision and Pattern Recognition. London: Springer London. doi*, vol. 10, no. 978, p. 1, 2015.
- [31] K. Rantou, “Missing data in time series and imputation methods,” *University of the*

*Aegean, Samos*, 2017.

- [32] A. Flores, H. Tito, and D. Centty, “Model for time series imputation based on average of historical vectors, fitting and smoothing,” *IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 10, no. 10, pp. 346–352, 2019.
- [33] H. Junninen, H. Niska, K. Tuppurainen, J. Ruuskanen, and M. Kolehmainen, “Methods for imputation of missing values in air quality data sets,” *Atmospheric Environment*, vol. 38, no. 18, pp. 2895–2907.
- [34] S. Moritz and T. Bartz-Beielstein, “imputeTS: Time Series Missing Value Imputation in R,” *The R Journal*, vol. 9, no. 1, pp. 207–218, 2017.
- [35] W. Junger and A. P. De Leon, “Imputation of missing data in time series for air pollutants,” *Atmospheric Environment*, vol. 102, pp. 96–104, 2015.
- [36] J. S. Armstrong and F. Collopy, “Error measures for generalizing about forecasting methods: Empirical comparisons,” *International journal of forecasting*, vol. 8, no. 1, pp. 69–80, 1992.
- [37] R. Gnanadesikan and M. B. Wilk, “Probability plotting methods for the analysis of data,” *Biometrika*, vol. 55, no. 1, pp. 1–17, 1968.
- [38] M. S. Paoletta, *Linear Models and Time-Series Analysis: Regression, ANOVA, ARMA and GARCH*. John Wiley & Sons, 2018.
- [39] P. Burns, “Robustness of the ljung-box test and its rank equivalent,” *Available at SSRN 443560*, 2002.
- [40] F. J. Massey Jr, “The kolmogorov-smirnov test for goodness of fit,” *Journal of the American statistical Association*, vol. 46, no. 253, pp. 68–78, 1951.
- [41] D. Zwillinger and S. Kokoska, *CRC standard probability and statistics tables and formulae*. Crc Press, 1999.

# Anexo A

## Figuras

### A.1. Resultados

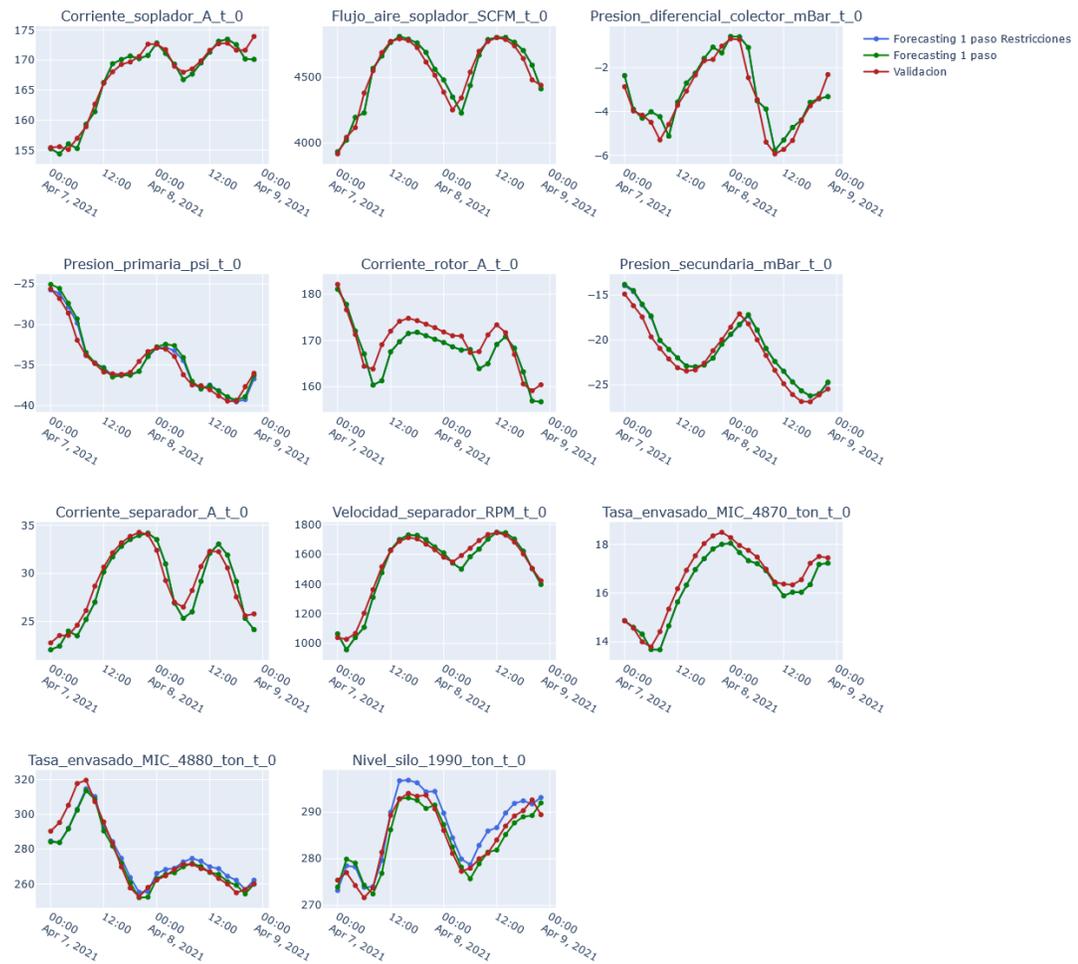


Figura A.1: Validación *Forecasting in sample* a un paso.