Facultad de Ciencias Económicas y Empresariales
ICADE

PREDICCIÓN Y GENERACIÓN DE ESCENARIOS DE CURVAS DE FUTUROS DEL BRENT USANDO TÉCNICAS DE MACHINE LEARNING

Autor: Isabel Lantero Hernández
Director: José Portela González

MADRID Junio, 2024

Índice

Índice de tablas y figuras 3
Resumen 6
Abstract 7
1. Introducción 8
1.1. Motivación y justificación 8
1.2. Objetivos 8
1.3. Metodología 9
2. Marco teórico 11
2.1. Los derivados financieros 11
2.2. Tipos de contratos de derivados 14
2.3. Los futuros 16
2.4. Los futuros del Brent 17
2.5. Trading con futuros del Brent: técnicas de Machine Learning 19
3. Metodología 21
3.1. Conjunto de datos utilizado 21
3.2. Análisis exploratorio del conjunto de datos 22
3.3. Técnicas de Machine Learning empleadas 25
4. Resultados 31
4.1. Objetivo I: Predecir las curvas de futuros del Brent 40
4.2. Objetivo II: Generar escenarios para las curvas en base a errores del modelo 51
4.3. Objetivo III: Generar escenarios para las curvas usando el precio del Brent como variable explicativa 55
5. Conclusiones 62
6. Declaración de Uso de Herramientas de Inteligencia Artificial Generativa en Trabajos Fin de Grado 66
7. Bibliografía 68
8. Anexo I: Código 71

Índice de tablas y figuras

Figura 1: Principales mercados de derivados en el mundo (2022)
Figura 2: Generación de beneficios de la posición larga y de la corta de un futuro
Figura 3: Evolución del volumen de futuros negociados globalmente
Figura 4: Variables que forman las curvas de futuros
Figura 5: Dispersión de las posiciones
Figura 6: Curvas de futuros originales
Figura 7: Correlación entre las posiciones de las curvas
Figura 8: Necesidad de estandarización de las variables
Figura 9: Loading vectors
Figura 10: Gráfico de contribución de cada variable a las componentes principales
Figura 11: Scores de las componentes principales
Figura 12: Correlación entre las componentes principales
Figura 13: Proporción de varianza explicada
Figura 14: Evolución de las dos primeras componentes principales
Figura 15: Biplot con las dos primeras componentes principales
Figura 16: Carga de las variables en cada componente principal
Figura 17: Valores de las componentes principales según la fecha
Figura 18: PCA - Serie temporal original vs. reconstruida
Figura 19: PCA - Serie temporal original vs. reconstruida – posiciones 1 y 11
Figura 20: PCA - Curvas de fututos originales vs. reconstruidas
Figura 21: Objetivo I - ACF y PACF de las dos primeras componentes principales
Figura 22: Objetivo I - Evolución de las dos primeras componentes principales antes y después de diferenciar
Figura 23: Objetivo I - ACF y PACF de las series diferenciadas
Figura 24: Objetivo I - Test Z de coeficientes significativos del modelo ARIMA(1,1,1) para la primera componente principal
Figura 25: Objetivo I - Test Z de coeficientes significativos del modelo ARIMA(1,1,0) para la primera componente principal
Figura 26: Objetivo I - ACF y PACF de los residuales de los modelos
Figura 27: Objetivo I - Prueba de Ljung-Box
Figura 28: Objetivo I - Métricas de error del dataset de entrenamiento de la primera componente principal usando ARIMA(1,1,0)
Figura 29: Objetivo I - Métricas de error del dataset de entrenamiento de la segunda componente principal usando ARIMA(0,1,0)
Figura 30: Objetivo I - Predicción de la primera componente principal con forecast()

Figura 31: Objetivo I - Predicción de la segunda componente principal con forecast()
Figura 32: Objetivo I - Predicción media de la primera componente principal con forecast()
Figura 33: Objetivo I - Predicción media de la segunda componente principal con forecast()
Figura 34: Objetivo I - Métricas de error de la primera componente principal usando ARIMA(1,1,0) aplicado a los datos de test.
Figura 35: Objetivo I - Métricas de error de la segunda componente principal usando ARIMA(0,1,0) aplicado a los datos de test
Figura 36: Objetivo I - Predicción media de la primera componente principal con fitted()
Figura 37: Objetivo I - Predicción media de la segunda componente principal con fitted()
Figura 38: Objetivo I - Serie temporal original vs. reconstruida
Figura 39: Objetivo I - Serie temporal original vs. reconstruida – posiciones 1 y 11
Figura 40: Objetivo I - Curvas de fututos originales vs. reconstruidas
Figura 41: Objetivo I – Error medio (ME) de cada posición
Figura 42: Objetivo I – Error medio absoluto (MAE) de cada posición
Figura 43: Objetivo II - Escenarios de la primera componente principal
Figura 44: Objetivo II - Escenarios de la segunda componente principal
Figura 45: Objetivo II - Serie temporal reconstruida
Figura 46: Objetivo II - Serie temporal original vs. reconstruida – posiciones 1 y 11
Figura 47: Objetivo II – Curvas de originales vs. reconstruidas
Figura 48: Objetivo III – ACF y PACF de las dos primeras componentes principales
Figura 49: Objetivo III – ACF y PACF de las dos primeras componentes principales diferenciadas
Figura 50: Objetivo III - Test Z de coeficientes significativos del modelo ARIMA(1,1,0) para la primera componente principal
Figura 51: Objetivo III - Test Z de coeficientes significativos del modelo ARIMA(0,1,0) para la segunda componente principal
Figura 52: Objetivo III - ACF y PACF de los residuales de los modelos
Figura 53: Objetivo III: Predicciones de las componentes principales
Figura 54: Objetivo III: Comparación de errores de predicción (MSE)
Figura 55: Objetivo III - Serie temporal original vs. reconstruida
Figura 56: Objetivo III - Serie temporal original vs. reconstruida – posiciones 1 y 11
Figura 57: Objetivo III - Curvas de fututos originales vs. reconstruidas
Figura 58: Objetivo III - Error medio (ME) de cada posición
Figura 59: Objetivo III - Error medio absoluto (MAE) de cada posición
Figura 60: Comparación entre modelos: Error medio (ME) de cada posición
Figura 61: Comparación entre modelos: Error medio absoluto (MAE) de cada posición

Tabla 1: Comparativa entre los distintos tipos de derivados

Tabla 2: Características de los futuros del Brent negociados en el ICE

Tabla 3: Objetivo I - Errores de predicción (MSE) con forecast()

Tabla 4: Objetivo I - Comparación de errores de predicción (MSE) con forecast() y con fitted()

Tabla 5: Objetivo III: Comparación de errores de predicción (MSE)

Resumen

La importancia estratégica de los futuros del Brent en todos los sectores de la economía global, junto con su creciente volumen de negociación y su elevada liquidez, los convierte en objeto de estudio para un número creciente de inversores. Este aumento de interés ha venido acompañado por una profundización en la investigación de técnicas de Machine Learning empleadas para poder predecir sus precios. Por lo tanto, el objetivo de este trabajo es evaluar distintas técnicas para predecir y generar escenarios de la curva de futuros del Brent que permitan determinar mejores estrategias de trading.

El análisis de componentes principales determina que las dos primeras componentes explican el 99,8% de la varianza y que éstas deben usarse en la implementación de los modelos de predicción y de generación de escenarios.

Para predecir las curvas de futuros del Brent se evalúan dos opciones: un modelo ARIMA(1,1,0), y un modelo de regresión dinámica ARIMA(1,1,0) que usa el precio actual del Brent como variable explicativa. Aunque las predicciones de ambos modelos son adecuadas, el modelo de regresión dinámica tiene una precisión más alta en la mayoría de los puntos de las curvas.

Por otro lado, la técnica de generación de escenarios mediante la simulación de errores históricos de un modelo ARIMA no es particularmente efectiva para simular las curvas de futuros del Brent. Los escenarios no contribuyen a entender la variabilidad y la incertidumbre de las predicciones.

Palabras clave: futuros, Brent, predicciones, PCA, ARIMA, generación de escenarios.

Abstract

The strategic importance of Brent futures to all sectors of the global economy, together with their growing trading volume and high liquidity, has made them a major target for many investors. This high level of interest has been accompanied by increased research on machine learning techniques used to predict their prices. Therefore, this study aims to apply different techniques to predict and generate scenarios of the Brent futures curve to determine better trading strategies.

Principal Component Analysis shows that the first two components explain 99.8% of the variance and that they should be used in implementing forecasting and scenario generation models.

Two options are evaluated for predicting the Brent futures curves: an ARIMA(1,1,0) model and an ARIMA(1,1,0) dynamic regression model using the current Brent price as the explanatory variable. Although the predictions of both models are reasonable, the dynamic regression model has a higher accuracy at most points on the curves.

On the other hand, the generation of price scenarios that simulate the historical errors of an ARIMA model did not prove to be particularly useful in simulating Brent futures curves. The scenarios did not help to understand the variability and uncertainty of the forecasts.

Key words: futures, Brent, forecasting, PCA, ARIMA, scenario generation.

1. Introducción

1.1. Motivación y justificación

Los derivados financieros, la categoría en la que se engloban los futuros, han evolucionado significativamente a lo largo del tiempo, adaptándose a nuevas tecnologías y marcos normativos (Sánchez Navarro, 2015). Por su parte, el volumen de futuros negociados globalmente se ha duplicado en los últimos años, pasando de 12,1 miles de millones en 2012 a 29,3 en 2022 (Futures Industry Association, 2023).

En particular, dentro del mercado de futuros, se encuentran los futuros del petróleo Brent. La relevancia del Brent en todos los sectores de la economía global, especialmente dado el contexto actual de inflación y el conflicto entre Rusia y Ucrania, provoca que el precio de los futuros del Brent sea objeto de estudio por un número creciente de inversores que también buscan la liquidez de estos contratos.

El interés por parte de los inversores en los futuros del Brent ha venido acompañado por un aumento en la investigación de técnicas empleadas de Machine Learning para poder predecir sus precios. Algunas de las técnicas incluyen: el análisis de componentes principales, la predicción de precios empleando modelos ARIMA, la generación de escenarios de precios en base a errores históricos, y la predicción de precios integrando un modelo ARIMA a una variable explicativa.

Por lo tanto, resulta necesario realizar un análisis del mercado de los futuros de Brent con el objetivo de predecir y generar escenarios de sus curvas usando estas técnicas de Machine Learning para poder colaborar en la determinación de estrategias de trading.

1.2. Objetivos

Por ello, se han establecido tres objetivos que, de manera ordenada y secuencial, pretenderán implementar técnicas de Machine Learning para predecir y generar escenarios de las curvas de futuros del Brent:

- Objetivo I: Predecir las curvas de futuros del Brent.
- Objetivo II: Generar escenarios para las curvas en base a errores históricos del modelo.
- Objetivo III: Predecir las curvas usando el precio del Brent como variable explicativa.

1.3. Metodología

Antes de exponer los resultados obtenidos en cada uno de los tres objetivos, es importante explicar el conjunto de datos del que se parte, y realizar una revisión teórica de las técnicas de Machine Learning que se usarán para el cumplimiento de cada objetivo.

El conjunto de datos del que parte este trabajo está formado por series temporales de los precios de cierre de posiciones de contratos de futuros del Brent con diferentes vencimientos. El dataset está compuesto por 763 filas, que corresponden a los días de negociación (desde el 1 de febrero de 2019, hasta el 14 de enero de 2022), y 12 variables, que corresponden a la variable fecha y a las 11 posiciones de futuros del Brent, representando distintos vencimientos. A partir de esta información, se pueden construir 763 curvas de futuros, una por cada día de negociación, formadas por 11 puntos, uno por cada posición de contrato de futuros de distintos vencimientos.

Debido al elevado número de variables y su alta correlación, en primer lugar, se llevará a cabo un análisis de componentes principales (*Principal Component Analysis*, PCA), que se trata de una técnica utilizada para reducir la dimensionalidad de un conjunto de datos mientras se retiene la mayor cantidad posible de información. Esta técnica permitirá estudiar los factores que influyen en la curva de futuros para poder determinar el número de componentes principales que se usarán para predecir y generar escenarios de las curvas. Una vez se tengan las componentes principales más explicativas, se podrá proceder a la ejecución de los objetivos modelando las curvas de futuros usando técnicas distintas para cada uno.

- Objetivo I: Predecir las curvas de futuros del Brent.

Para la consecución del primer objetivo, se utilizará un modelo ARIMA (Autoregressive Integrated Moving Average). El modelo ARIMA(p,d,q) es una técnica de series temporales de Machine Learning que combina componentes autorregresivos (p), de diferenciación (d) y de media móvil (q) para predecir futuros valores basados en datos históricos (Hyndman & Athanasopoulus, 2018).

- Objetivo II: Generar escenarios para las curvas en base a errores históricos del modelo.

Para ello, se usará la función “simulate” de R, que permite generar distintas trayectorias futuras de una serie temporal mediante simulaciones con un componente aleatorio. De

esta manera, cada escenario se basará en el modelo ARIMA previamente ajustado integrado con los errores simulados aleatoriamente según la distribución histórica del modelo. Así, se podrá entender en mayor profundidad la incertidumbre de las predicciones.

- Objetivo III: Predecir las curvas usando el precio del Brent como variable explicativa.

Para cumplir el último objetivo, se usará un modelo de regresión dinámica que permita combinar un modelo ARIMA con la regresión de una variable explicativa, en este caso el precio *spot* del Brent, que corresponde a la primera posición del *dataset* y que se refiere al precio de mercado actual del Brent. De esta manera, no solo tendrán en cuenta las series temporales para realizar las predicciones, sino que, además, se tendrán en cuenta factores explicativos que puedan afectar a las curvas en horizontes posteriores (como eventos geopolíticos, cambios en políticas energéticas…).

2. Marco teórico

Antes de abordar los principales objetivos, es esencial contextualizar el pilar sobre el que se sustenta el trabajo: las técnicas de Machine Learning para predecir y generar escenarios de las curvas de futuros del Brent. De esta manera, se acotará progresivamente el marco hasta llegar específicamente a la teoría de este pilar. Primero, se ofrecerá una visión general de los derivados, luego se contextualizarán los futuros como tipo de contrato de derivado. Posteriormente, se definirán específicamente los futuros del Brent y, finalmente, se presentará la teoría de las técnicas de Machine Learning utilizadas para predecir las curvas de estos contratos.

Aunque el análisis se centra en las técnicas de Machine Learning, es crucial abordar los temas previos para asegurar una comprensión completa. Posteriormente, en la sección de resultados, se aplicará a la práctica la teoría expuesta para pronosticar y generar escenarios de los precios de los futuros del Brent.

2.1. Los derivados financieros

a. Definición de derivados

Los futuros son un tipo de contrato de derivados financieros. Según la CNMV (2024), los derivados son instrumentos financieros "cuyo valor deriva de la evolución de los precios de otro activo, denominado activo subyacente". Dicho de otra manera, se trata de un acuerdo de compraventa en el que se especifican los detalles en el momento del pacto, pero el intercambio se produce en un momento futuro. Una de las características más relevantes de los productos derivados es que están sujetos al efecto apalancamiento: la inversión inicial para entrar un contrato de este tipo es reducida en comparación con la exposición total al activo subyacente. De esta manera, los resultados se pueden amplificar significativamente, ya sea en ganancias o pérdidas, en relación al dinero invertido, por lo que a menudo se consideran productos de alto riesgo (Giraldo-Prieto, et al., 2017).

Según se explicará más adelante, existen distintos tipos de derivados, como futuros u opciones, según el mercado en el que se negocien (organizado o no organizado) y las obligaciones de cada parte del contrato. Por otro lado, también existen distintos tipos de activos subyacentes sobre los que puede depender un derivado (acciones, tipos de interés, materias primas…).

b. ¿Por qué se contratan?

Los motivos para la contratación de derivados financieros dependen de la necesidad y del fin que busca cumplir el inversor. Existen tres principales motivos por los que un agente de mercado entra en un contrato de futuros: la especulación, la cobertura y el arbitraje (Elvira & Larranga, 2008; Sánchez Navarro, 2015).

En primer lugar, la especulación hace referencia a la intención de anticipar la dirección del mercado y obtener beneficios basados en predicciones. Los inversores especuladores tienen el fin de maximizar sus ganancias y minimizar su capital invertido mediante productos financieros apalancados.

Por otro lado, la cobertura se refiere a la capacidad de los derivados para reducir el riesgo de cambios variables de mercado, fijando el precio de compraventa de un activo con anterioridad. De esta manera, aunque se pierde la posibilidad de generar ganancias por los movimientos de precios, también se elimina la incertidumbre de que existan fluctuaciones de precios que afecten de manera negativa a la posición del inversor.

Por último, la cualidad de arbitraje permite que los contratos de derivados se aprovechen de imperfecciones en los mercados financieros para obtener beneficios sin riesgo, tomando posiciones simultáneas en distintos mercados o instrumentos. Un ejemplo es, ante condiciones de ineficiencia de mercado, tomar una posición larga en un activo subyacente y una posición corta en su futuro correspondiente. Además de eliminar el riesgo de una operación, el arbitraje mejora la eficiencia del mercado al corregir las desviaciones de precio.

c. Historia de los derivados y principales mercados de negociación

Contradiciendo la percepción moderna de que los derivados son productos recientes y complejos destinados solo para grandes corporaciones e inversores, este tipo de productos financiero ha sido una herramienta vital en el comercio durante siglos, adaptándose y evolucionando para cumplir con las necesidades de cobertura y especulación de diferentes épocas y mercados (Sánchez Navarro, 2015).

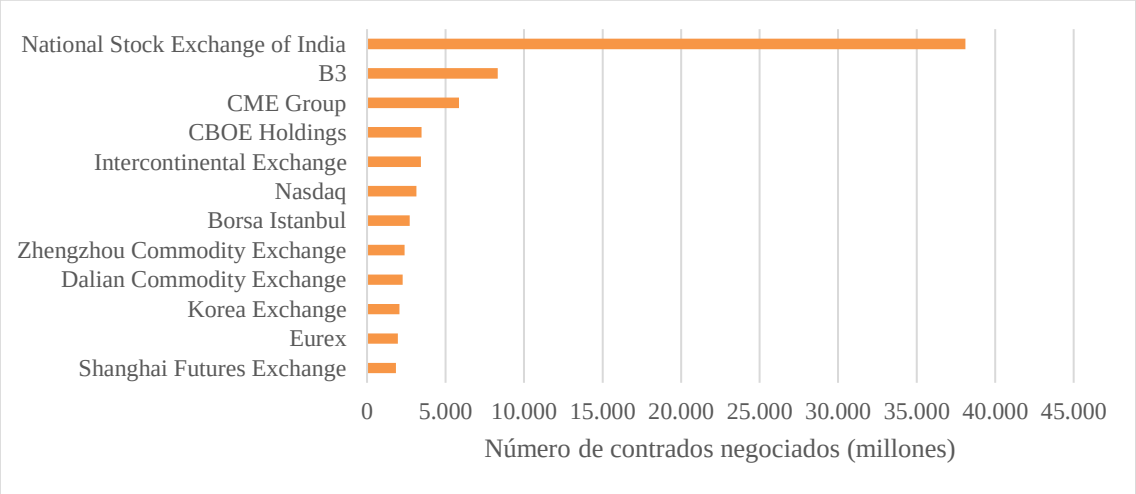
La primera referencia a un instrumento similar a un contrato de derivados se encuentra en la “Política” de Aristóteles, donde se menciona cómo Tales de Mileto se enriqueció mediante un contrato derivado gracias a sus predicciones sobre la cosecha de aceitunas (Castellanos, 2017). Este tipo de contratos también fueron comunes entre los fenicios, griegos y romanos, quienes establecían acuerdos a futuro para asegurar precios y

suministros (Kummer & Pauletto 2012; Lamothe, 1993). Sin embargo, el primer contrato de derivados tal y como lo conocemos hoy en día, surgió en el siglo XVII en Osaka, donde se estableció un mercado organizado sobre el arroz, sentando así las bases de los mercados de derivados modernos (Castellanos, 2017).

El primer mercado formal de derivados fue el Chicago Board of Trade (CBOT) en 1848, centrado inicialmente en productos agrícolas. Para abordar la necesidad de los inversores de cubrirse de la volatilidad de precios, en las décadas siguientes, este mercado introdujo contratos a futuro estandarizados en términos de calidad, cantidad, fecha y lugar de entrega (Bañón González, 2018). La expansión de los mercados de derivados continuó en Europa en 1978 con la European Options Exchange en Ámsterdam y posteriormente con la creación de LIFFE en Londres y MATIF en París. En España, los mercados de derivados se establecieron en los años 80, con la creación de OM Ibérica en Madrid y MEFF en Barcelona, que actualmente están integrados en Bolsas y Mercados Españoles (Bañón González, 2018; Castellanos, 2017).

Los mercados de derivados han evolucionado significativamente a lo largo de los años, adaptándose a nuevas tecnologías y marcos normativos, pasando de negociarse en subastas de viva voz a negociarse electrónicamente sin necesidad de un intermediario financiero con presencia física (Sánchez Navarro, 2015). La amplia y creciente acogida de los derivados justifica la necesidad de estudio de un tema como el que presenta este trabajo. Es importante destacar que, hoy en día, los mercados de derivados en Asia y Norteamérica representan una parte muy notable del mercado global de derivados, según se muestra en el siguiente gráfico (Futures Industry Association, 2022).

Figura 1: Principales mercados de derivados en el mundo (2022)



Fuente: Elaboración propia a partir de datos de Futures Industry Association (2022)

2.2. Tipos de contratos de derivados

Como se anticipó en el apartado anterior, existen diversos tipos de derivados según el nivel de organización del mercado en el que se negocian y el grado de obligación de cada una de las partes del contrato.

Antes de introducir las clases de productos derivados, es conveniente describir brevemente los mercados en los que se pueden negociar: mercados organizados o mercados no organizados, comúnmente conocidos como O*ver The Counter* (OTC). La diferencia entre ambos reside principalmente en tres dimensiones: la regulación, el riesgo de contrapartida, y la estandarización de los contratos (Feelcapital, 2017). En primer lugar, los mercados organizados están regulados por una institución financiera o gubernamental (por ejemplo, el MEFF en España), mientras que los contratos en mercados OTC no tienen a ningún regulador que supervise las operaciones. Además, los mercados organizados funcionan a través de una cámara de compensación, exigiendo un depósito de garantías que aumenta los requerimientos de liquidez y que reduce el riesgo de contrapartida; sin embargo, los mercados OTC se negocian directamente entre las partes, por lo que no hay ningún intermediario que limite este tipo de riesgo. Ambas dimensiones previamente descritas llevan a la tercera diferencia: la estandarización. Para cumplir con los requisitos de los reguladores y de liquidez, los derivados en mercados organizados son estandarizados, mientras que los contratos OTC se hacen a medida entre las partes (Feelcapital, 2017; Magnier Villamil, 2014).

Principalmente se pueden encontrar los siguientes tipos de productos derivados (Elvira & Larraga, 2008; Figueroa, 2008; Gray & Place, 2003; Santander, 2024):

- Futuros: se trata de contratos negociados en mercados organizados en los que el comprador y el vendedor pactan intercambiar una cantidad determinada de un activo subyacente en un momento futuro, a un precio (denominado precio de ejercicio) y según unas condiciones (cantidad, lugar de entrega, forma pago, fecha de vencimiento…) especificadas en el momento inicial del acuerdo. Al negociarse en mercados organizados, cuentan con un alto grado de estandarización, con garantías, y con liquidación diaria. Será en esta clase de derivados en la que se centrará el presente trabajo, por lo que se profundizará más sobre sus características en los siguientes apartados.

- Forwards: este tipo de productos funcionan de la misma manera que un futuro, a excepción del tipo de mercado en el que se negocian y las implicaciones que ello conlleva. A diferencia de los futuros, los forwards son acuerdos negociados directamente entre las partes en mercados OTC, por lo que son altamente personalizables, limitando su estandarización. No tienen liquidación diaria y el riesgo de la contraparte es mayor.
- Opciones: contrato negociado en mercados organizados que otorga al comprador el derecho, pero no la obligación, de intercambiar una cantidad determinada de un activo subyacente en un momento futuro, a un precio y según unas condiciones especificadas previamente. A cambio de obtener este derecho, el comprador de la opción debe pagar una prima. De esta manera, la exposición a pérdidas del comprador es limitada ya que, llegada la fecha de vencimiento, puede decidir si ejercitar la opción dependiendo la diferencia entre el precio de ejercicio y el precio actual del activo subyacente.
- Warrants: este producto financiero presenta el mismo funcionamiento que la excepción, con la diferencia de que se negocian OTC entre empresas, por lo que tienen un nivel de estandarización y liquidez menor.
- *Swaps*: contratos negociados en mercados no organizados, en los que las distintas partes acuerdan el intercambio de flujos monetarios en un momento futuro y según ciertas reglas establecidas previamente.

Tabla 1: Comparativa entre los distintos tipos de derivados

	Futuros	**Forwards**	**Opciones**
Mercado	Organizado	OTC	Organizado
Términos del contrato	Estandarizado	Ajustado a las medidas de las partes	Estandarizado
Relación partes	Anónima (a través de una cámara de compensación)	Directa	Anónima (a través de una cámara de compensación)
Liquidación diaria	Sí	No	No
Derecho / obligación del comprador	Obligación	Obligación	Derecho
Limite pérdidas	No	No	Sí, a cambio del pago de una prima

Fuente: Elaboración propia

2.3. Los futuros

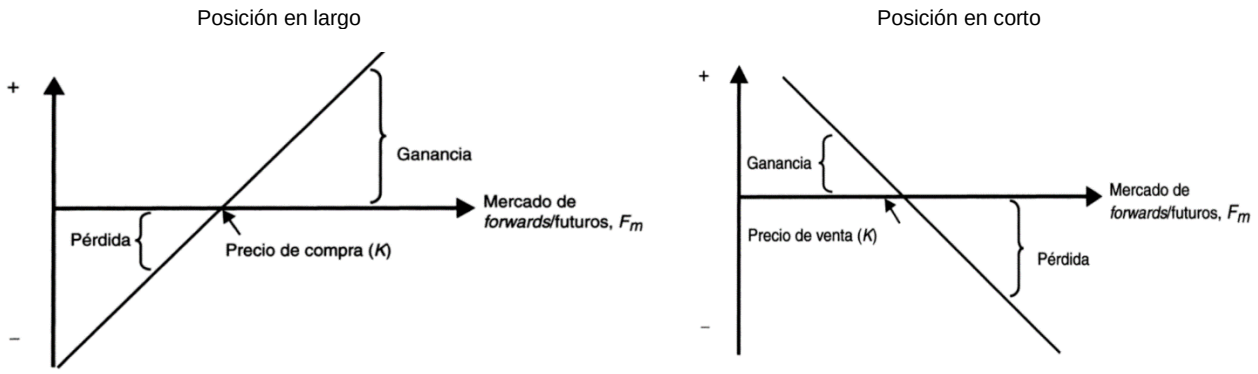
Como se ha mencionado previamente, los futuros son un tipo de contratos de derivados negociados en mercados organizados a través de cámaras de compensación. En estos contratos, el comprador y el vendedor acuerdan unas condiciones y un precio en un momento inicial (precio de ejercicio), para intercambiar una cantidad determinada de un activo subyacente en un momento futuro (a vencimiento del contrato).

Como el precio de un futuro se especifica en el momento inicial del acuerdo, las partes implicadas tienen pérdidas o ganancias según su posición (comprador o vendedor) y según sea el precio del activo subyacente en el momento de vencimiento del contrato (precio de liquidación).

De esta manera, la parte que se compromete a comprar, que se dice que "tiene una posición larga", genera beneficios cuando el precio del activo subyacente crece respecto al momento inicial del acuerdo. Así, el precio de ejercicio (K) sería menor que el precio de liquidación (F_m), por lo que, en el momento del intercambio del subyacente se habría pagado menos por él que si se pagara el precio de cotización actual. De la misma manera, si el precio del activo subyacente cae, la posición en largo tendría pérdidas (Sánchez Navarro, 2015).

Por otro lado, la parte que se compromete a vender, que se dice que "tiene una posición en corto", recibe beneficios si el precio del activo subyacente cae. De esta manera, el precio de ejercicio (K) estaría por encima del precio de liquidación en el momento de vencimiento (F_m), por lo que, la posición en corto cobraría más que si vendiera al precio actual (Sánchez Navarro, 2015).

Figura 2: Generación de beneficios de la posición larga y de la corta de un futuro

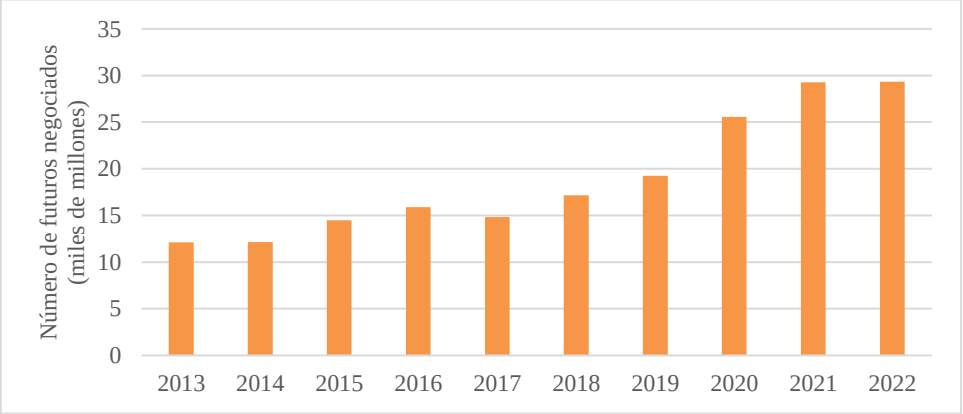


Fuente: De Lara (2005)

De esta manera, al contrario que en el caso de las opciones, el hecho de que ambas partes tengan la obligación de ejecutar el intercambio en la fecha de vencimiento, tanto los beneficios como las pérdidas potenciales, son ilimitadas.

En conclusión, al negociarse en mercados organizados, los futuros son instrumentos con un alto grado de estandarización y, por lo tanto, liquidez. Además, su riesgo crediticio es limitado ya que una cámara de compensación actúa de intermediario. Por último, al ser un derivado, cumple los requisitos de tres distintos tipos de inversores: de cobertura, especuladores y arbitrajistas. Estas ventajas hacen la negociación de este tipo de contratos útil en muchos casos. De hecho, el volumen de futuros negociados globalmente se ha multiplicado por dos en los diez últimos años, pasando de 12,1 miles de millones en 2012 a 29,3 en 2022 (Futures Industry Association, 2023).

Figura 3: Evolución del volumen de futuros negociados globalmente



Fuente: Elaboración propia a partir de datos de Futures Industry Association (2023)

Como se puede observar, los futuros son un instrumento financiero en pleno crecimiento gracias a las ventajas que ofrece. Por lo tanto, su estudio es cada vez más relevante y esencial, justificando así la necesidad de llevar un análisis que permita a los inversores tomar mejores decisiones sobre su contratación y estrategias de trading. En el caso del presente trabajo, dicho análisis consistirá en la predicción y simulación de sus precios.

2.4. Los futuros del Brent

Según lo explicado previamente, existen distintos tipos de activos subyacentes sobre los que puede derivar el precio de un futuro. Según Sánchez Navarro (2015), existen dos principales categorías de futuros según su activo subyacente: futuros sobre instrumentos financieros (divisas, tipos de interés, acciones, índices bursátiles…) y futuros sobre materias primas (productos agrícolas, energía, metales…).

Dentro del subgrupo de energías perteneciente a la categoría de materias primas, se encuentran los futuros del petróleo, los cuales representan la mayor parte del subgrupo en términos de volumen de negociación. Sin embargo, el petróleo no solo es un recurso para la generación de energía, sino que, además, sus derivados son componentes esenciales en todo tipo de procesos industriales: fabricación de plásticos, asfalto, pinturas, detergentes, etc. (Gamero, 2018). En el contexto geopolítico actual de inflación y de conflicto entre Rusia y Ucrania, el petróleo determina en gran medida las relaciones de comercio internacional y la consecución de las actividades industriales. Por lo tanto, la relevancia del petróleo en todos los sectores de la economía global provoca que su precio sea objeto de estudio por un número creciente de inversores.

A su vez, dentro de los futuros del petróleo se encuentran los futuros del Brent, los cuales sirven como referencia de precios en Europa y representan cerca del 66% de los contratos físicos de todo el mundo, demostrando así su elevado volumen de negociación (Moreno-Torres Gálvez, 2020).

El Brent es un tipo de petróleo crudo que se extrae principal del Mar del Norte. Es dulce y ligero, lo cual lo hace óptimo para la producción de gasolina y diésel, dos de los derivados del petróleo más demandados mundialmente (Gonzálvez, 2023).

En particular, hoy en día, los futuros del Brent cobran especial importancia dada la situación de conflicto entre Rusia y Ucrania, y el contexto de inflación, que afecta en gran medida a los precios y a la volatilidad de esta materia prima (Díaz-Pinzón, 2023).

Actualmente, los futuros del Brent se negocian en el Intercontinental Exchange (ICE) y en el New York Mercantile Exchange (NYMEX), dos mercados regulados que implican la estandarización de este tipo de contratos, promoviendo así su liquidez.

Tabla 2: Características de los futuros del Brent negociados en el ICE

Símbolo	B
Tamaño de contrato	1.000 barriles
Múltiplo	1.000 barriles
Divisa	USD
Cámara de compensación	ICE Clear Europe
Liquidación	Efectivo, por diferencias, o en especie

Fuente: Intercontinental Exchange (2024); Gonzálvez (2023)

Por lo tanto, la importancia estratégica del Brent en todos los sectores de la economía global, su elevado y creciente volumen de negociación, su liquidez, y el contexto geopolítico de conflicto e inflación que lleva a la volatilidad de precios de este tipo de materia prima, determinan la importancia de realizar un análisis al mercado de los futuros de Brent y modelar sus curvas.

2.5. Trading con futuros del Brent: técnicas de Machine Learning

El creciente interés por parte de los inversores en los futuros del Brent ha venido acompañado por un aumento en la investigación de técnicas empleadas para poder predecir sus precios. Por lo tanto, se ha realizado una revisión literaria de las técnicas de Machine Learning que se implementarán en secciones posteriores.

En cuanto al análisis de componentes principales, Jacobson (2015) estableció en su publicación la capacidad de predicción de los precios de futuros usando esta técnica para poder evaluar estrategias de inversión. Además, en otro estudio el uso de PCA determinó que las curvas de futuros del petróleo se explican principalmente por dos componentes principales: la primera está relacionada con el patrón temporal de la serie original, y la segunda sirve para ajustar ligeramente la curva según convenga (Mohamad, 2019).

Por otro lado, Box y Jenkins presentaron en 1970 el modelo ARIMA, el cual se emplea para identificar, estimar, evaluar, y predecir series temporales (Stellwagen y Tashman, 2013). Desde entonces, el modelo se ha empleado en distintas publicaciones para predecir todo tipo de valores, entre ellos el precio del petróleo. Aplicando la metodología de Box y Jenkins, Mensah (2015) utilizó datos históricos de dos décadas del Brent para modelar sus precios a corto plazo y llegó a la conclusión de que ARIMA (1,1,1) era el modelo más preciso. Por su parte, Shah y Kurthiga (2020) intentaron predecir los precios a cuatro años del petróleo WTI, determinando que el modelo ARIMA (0,1,4) era la mejor opción. Otros autores como Zhao Wang (2014) también utilizaron esta técnica de Machine Learning para predecir y generar escenarios de los precios del petróleo crudo (Gasper & Wbwambo, 2023). Aunque los modelos recomendados variaran en su orden, probablemente debido a la discrepancia de fechas utilizadas para entrenar el modelo y al hecho de que los precios eran de distintos tipos de petróleo, todos ellos demostraron ser suficientemente precisos en la generación de pronósticos, demostrando así la utilidad de los modelos ARIMA.

Además, con el fin de crear los pronósticos más precisos, existen publicaciones que han llevado el modelo ARIMA un paso más allá añadiendo una variable explicativa. Es el

caso de Rahmayanti y Andreas (2021), que aplicaron una variable exógena a un modelo ARIMA para analizar el impacto de la guerra comercial entre Estados Unidos y China en los precios de futuros del Brent. Utilizaron una variable dicotómica que indicaba la presencia o ausencia de la guerra comercial como variable exógena, y el modelo reveló que la guerra comercial tenía un impacto significativo en el precio del Brent.

3. Metodología

En este apartado, se describirá el conjunto de datos utilizado y se expondrán los pasos que se han seguido en la sección "Resultados" para abordar los tres objetivos de investigación. Para ello, primero, se presentarán las características del conjunto de datos. Luego, se ofrecerá un repaso teórico de las técnicas de Machine Learning empleadas, para facilitar la comprensión de su implementación posterior.

De esta manera, más adelante, en la sección de "Resultados" se aplicarán estas técnicas de Machine Learning para abordar los tres objetivos de investigación, y poder predecir y generar escenarios de los precios de los futuros del Brent.

3.1. Conjunto de datos utilizado

En esta sección, se describen las fuentes y el proceso de construcción del conjunto de datos de las curvas de futuros de Brent del que parte el presente trabajo. Este *dataset* constituye la base para el análisis de las estrategias de trading mediante técnicas de Machine Learning.

El conjunto de datos parte de un *dataset* inicial construido en el contexto de la Research Community de la Universidad Pontificia Comillas, que contiene los precios de varios contratos de futuros del Brent de distintos vencimientos. A su vez, en un Trabajo de Fin de Grado anterior realizado por Ángel Gonzálvez (2023), se adaptó el *dataset* para solventar los problemas de liquidez y vencimiento que suponía tener distintos contratos de futuros, construyendo así el conjunto de datos del que partimos. Para ello, se seleccionaron los contratos con mayor liquidez y volumen de negociación (específicamente los correspondientes a los meses de marzo, junio, septiembre y diciembre), y se acudió a un análisis por posiciones. Un análisis por posiciones implica organizar los datos de tal manera que cada punto de la curva de futuros representa una posición específica en la serie temporal de contratos, independientemente de los contratos específicos. Esto permite comparar directamente las mismas posiciones a lo largo del tiempo, algo que no sería posible si se utilizaran los contratos individuales ya que a medida que fueran venciendo, faltarían datos. Al organizar los datos por posiciones, se asegura la continuidad en la información, facilitando así el análisis de patrones y dinámicas en las curvas de futuros.

De esta manera, el conjunto de datos del que parte este trabajo está formado por series temporales de los precios de cierre de posiciones de contratos de futuros del Brent con

diferentes vencimientos. Está compuesto por 763 filas, que corresponden a los días de negociación (desde el 1 de febrero de 2019, hasta el 14 de enero de 2022), y 12 variables, que corresponden a la variable fecha y a las 11 posiciones de futuros del Brent, representando distintos vencimientos. Así, se pueden construir 763 curvas de futuros, una por cada día de negociación, formadas por 11 puntos, uno por cada posición de contrato de futuros de distintos vencimientos.

3.2. Análisis exploratorio del conjunto de datos

Antes de profundizar en las técnicas de Machine Learning empleadas para determinar las estrategias óptimas de trading con futuros del Brent, es esencial realizar un análisis exploratorio del *dataset* para saber cómo se comportan este tipo de derivados financieros, y cómo ajustar adecuadamente las técnicas y garantizar su efectividad.

Como se ha mencionado anteriormente, el conjunto de datos está compuesto por 12 variables: once de ellas representan el precio de cada posición en futuros del Brent, y la otra es la fecha. A continuación, se muestra la serie temporal de las 11 posiciones que forma la curva de futuros.

Figura 4: Variables que forman las curvas de futuros

Fuente: Elaboración propia

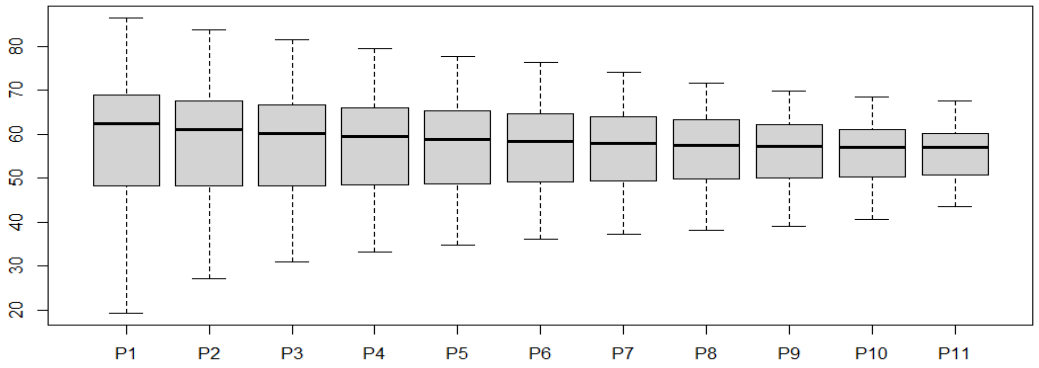
Según muestra la evolución del gráfico, los precios de los contratos permanecieron relativamente contantes, entre 60 y 70 euros, hasta la llegada del COVID-19 a principios de 2020, cuando los precios cayeron notablemente, situándose por debajo de los 50 euros. Antes de la pandemia, las posiciones con un vencimiento más cercano tenían un precio más elevado que las últimas posiciones; es decir, la curva de futuros estaba en *backwardation*. Sin embargo, desde marzo de 2020 hasta diciembre del mismo año, la

curva de futuros estaba en contango ya que los precios de vencimientos más alejados eran más altos. La caída de precios duró hasta finales de abril, pero desde entonces los precios han subido de manera bastante consistente hasta enero de 2022 (última fecha disponible), alcanzando precios por encima de los 70 euros en 2022, y la curva de futuros ha vuelto a estar en *backwardation*.

En cuanto a la comparación entre las distintas variables, el gráfico indica que todas las posiciones siguen un patrón similar, demostrando así una elevada correlación. Sin embargo, las primeras posiciones muestran movimientos más notables que las últimas posiciones: ante subidas generalizadas de los precios de los contratos, el precio de las primeras posiciones sube en mayor proporción, y ante bajadas generalizadas de precios, las primeras posiciones caen más que las últimas. Por lo tanto, se podría concluir que cuanto más cercano sea el vencimiento de los contratos, mayor es su varianza y fluctuación, por lo que son más sensibles. Se podría decir, que la primera posición es la que lidera al resto.

Por otro lado, la mayor volatilidad de las primeras posiciones se confirma con el siguiente gráfico, que muestra cómo la varianza de los precios de los contratos va decayendo a medida que se alejan los vencimientos.

Figura 5: Dispersión de las posiciones

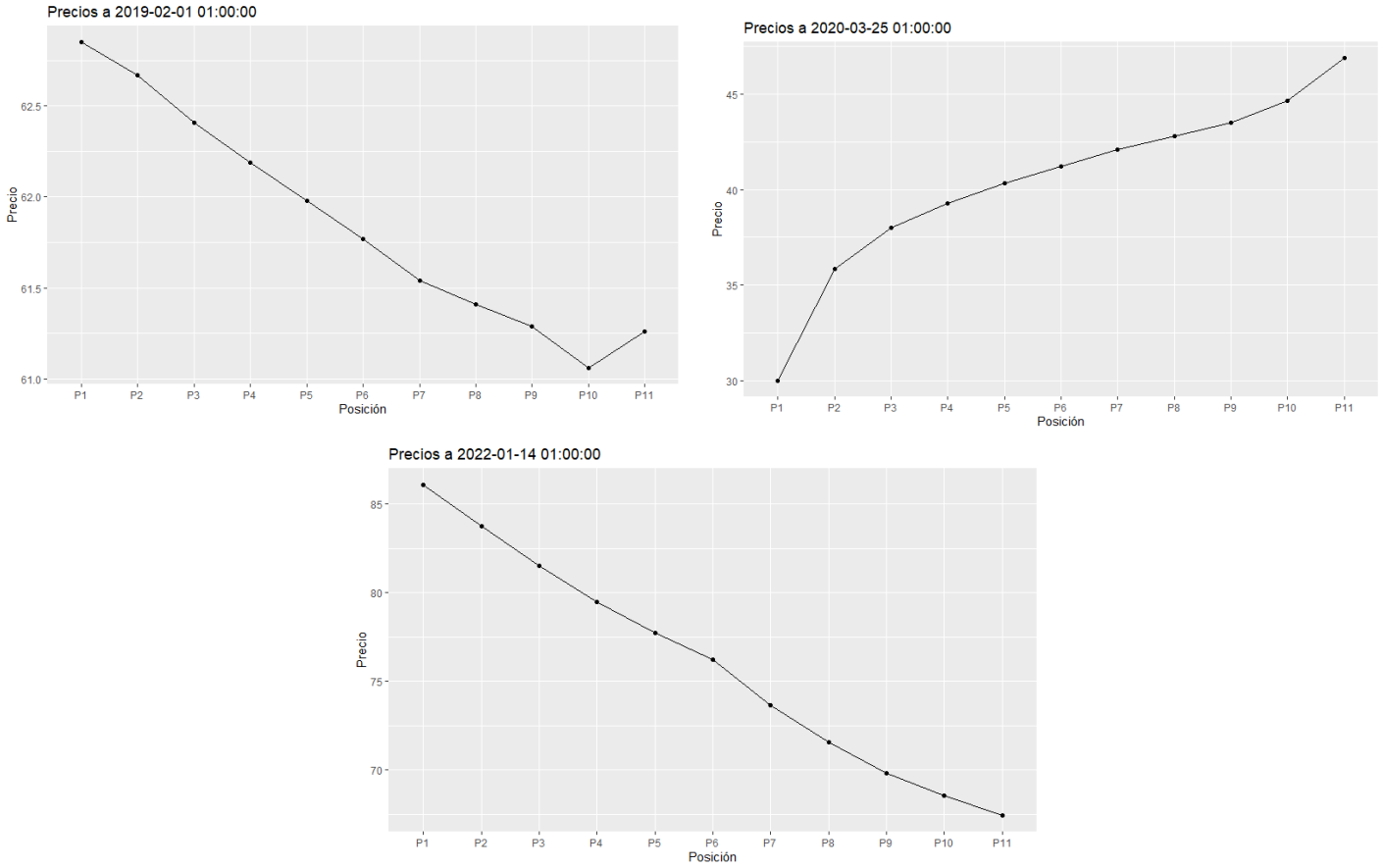


Fuente: Elaboración propia

Esta mayor volatilidad de los precios de posiciones con vencimientos más próximos podría deberse a una mayor sensibilidad a noticias y a eventos inmediatos, como tensiones geopolíticas, desastres naturales, o eventos inesperados como lo fue el COVID-19. Esto podría deberse a que este tipo de situaciones pueden provocar la interrupción inesperada de petróleo, afectando rápidamente a los contratos a corto plazo debido a la inmediatez de la entrega.

Una vez se han introducido las variables del *dataset*, es conveniente introducir los gráficos de las curvas de futuros, los cuales, para un determinado día de negociación, muestran los precios de cada posición según si vencimiento. A continuación, se exponen los gráficos de curvas de futuros de 3 días distintos. El primero muestra la curva del primer día disponible del *dataset*, el segundo la curva durante la pandemia, y el último la curva del último día disponible.

Figura 6: Curvas de futuros originales



Fuente: Elaboración propia

Como se puede observar, se confirma lo expuesto previamente en lo que se refiere al sentido de las curvas. Antes y después de los efectos de la pandemia, la curva de futuros estaba en *backwardation*: a medida que el vencimiento de los contratos se aleja, menor es el precio. Sin embargo, en plena crisis de COVID-19, la curva estaba en contango.

La situación de mercado en contango implica que los productores y otros actores del mercado del petróleo creen que el precio del petróleo subirá en el futuro. Durante la pandemia, la curva de futuros del Brent estuvo en contango debido a la significativa caída de la demanda global del petróleo y sus elevados costes de almacenamiento. De hecho,

según Reuters, el "desplome del petróleo Brent creó el contango más acentuado en 11 años" (2020). El petróleo incluso cotizó en negativo y los operadores del mercado esperaban una recuperación de los precios en el futuro cuando se recobrara la normalidad, lo que llevó a una acumulación de inventarios para aprovechar la diferencia de precios entre contratos a corto y a largo plazo.

Aunque haya movimientos acentuados en la evolución de los precios de los futuros, el presente trabajo tendrá como objetivo captar las dinámicas y los patrones de las series usando técnicas de Machine Learning para predecir las curvas de futuros y determinar las mejores estrategias de trading.

3.3. Técnicas de Machine Learning empleadas

Según lo expuesto previamente, el uso de técnicas de Machine Learning para modelar los precios de futuros del Brent y determinar estrategias de trading es cada vez más común y efectivo. Por ello, este trabajo implementa tres modelos distintos de Machine Learning para predecir las curvas, evalúa su precisión, y las compara para determinar cuál de todas es mejor: ARIMA, simulación de escenarios, y modelos de regresión dinámica.

Con el fin de ajustar los modelos y predecir los precios, se ha empleado el programa R Studio debido a su gran utilidad para programar de manera eficiente. Su amplia comunidad facilita la consulta y resolución de dudas, y ofrece una amplia variedad de librerías para realizar visualizaciones.

Debido al elevado número de variables y su alta correlación, en primer lugar, se llevará a cabo un análisis de componentes principales (PCA), con el que se estudiarán los factores que influyen en la curva de futuros y se determinarán el número de componentes principales que se usarán para predecir las curvas. Una vez se tengan las componentes principales más explicativas, se podrán modelar las curvas de futuros. Para ello, se dividirá el conjunto de datos de componentes principales en datos de entrenamiento, y datos de prueba (las dos últimas semanas disponibles). El proceso será el mismo para los tres modelos empleados:

- Ajuste de un modelo para cada componente principal usando los datos de entrenamiento correspondientes.
- Predicción o simulación de cada una de las componentes principales usando el modelo previamente ajustado.

- Reconstrucción de las curvas de futuros a partir de las componentes principales predichas y comparación con las curvas originales.
- Análisis de las predicciones.

En los siguientes apartados se explicarán desde un punto de vista teórico cada una de las técnicas de Machine Learning empleadas, y se expondrá cómo contribuyen al cumplimiento de cada objetivo del presente trabajo. En la sección "Resultados", se llevarán a la práctica y se podrán ver los resultados.

a. Análisis de componentes principales

El análisis de componentes principales (*Principal Component Analysis*, PCA) es una técnica estadística utilizada para reducir la dimensionalidad y evitar la multicolinealidad de un conjunto de datos mientras se retiene la mayor cantidad posible de información (Gonzálvez, 2023; Joliffe, 2002). En este proceso, se crea un nuevo conjunto de variables, las componentes principales, que surgen como combinación lineal de las variables originales. Las componentes principales se pueden ordenar según su varianza explicada; en otras palabras, su información explicada sobre las variables objetivas. Si se seleccionan todas las componentes principales, se puede reconstruir el conjunto de datos original sin perder nada de información.

El primer paso a la hora de realizar este análisis consiste en estandarizar los datos originales para que sean comparables entre sí, de manera que cada variable tenga media de 0 y desviación típica de 1. Después, a través del cálculo de la matriz de covarianza de los datos estandarizados, se obtienen los *loading vectors* y los *scores*. Los *loading vectors* definen la dirección de los nuevos ejes en el espacio de las variables originales, y se representan en un gráfico llamado *biplot*. Los scores hacen referencia la proyección de los datos en el eje definido por los *loading vectors*.

Cada componente principal (P_i) se calcula como una combinación lineal de las variables originales ($X_1, X_2, \ldots, X_p$) usando los loading vectors (ϕ_{ip}) como coeficientes, de tal manera que la primera componente principal se podría calcular de esta manera:

$$P_1 = \phi_{11} X_1 + \phi_{12} X_2 + \cdots + \phi_{1p} X_p$$

Es importante destacar que todas las componentes principales deben ser ortogonales entre sí, y que, por lo tanto, están no correlacionadas. Por ello, para el cálculo de resto componentes principales se añaden restricciones a la fórmula para asegurar que la nueva

componente principal defina una dirección ortogonal a las anteriores.

Una vez generadas todas las componentes principales, se ordenan de acuerdo a su varianza explicada y se seleccionan las que tienen los valores más de esta métrica, ya que son las que retienen mayor variabilidad en los datos. A partir de las componentes principales seleccionadas se puede reconstruir una estimación del conjunto de datos original.

El análisis de componentes principales permite reducir el número de variables manteniendo la mayor cantidad de información posible. Esta característica es especialmente útil para los modelos de Machine Learning que se explicarán a continuación, ya que en lugar de predecir cada una de las 11 variables originales, solo será necesario predecir un par de componentes principales. A partir de estas componentes principales, se podrán reconstruir las curvas de futuros.

b. Modelos ARIMA

Para cumplir el primer objetivo, que consiste en predecir las curvas de futuros del Brent, se empleará un modelo ARIMA (Autoregressive Integrated Moving Average). El modelo ARIMA(p,d,q) es una técnica de series temporales de Machine Learning que combina componentes autorregresivos (p), de diferenciación (d) y de media móvil (q) para predecir futuros valores basados en datos históricos (Hyndman & Athanasopoulus, 2018).

Para ajustar un modelo ARIMA es necesario conocer su orden. En primer lugar, para conocer el valor del parámetro *d*, que hace referencia al componente de diferenciación, es esencial verificar si la serie temporal es estacionaria. Si su media y su varianza varían a lo largo del tiempo, la serie no es estacionaria y hace falta diferenciar. En el caso de este trabajo, dicha comprobación se hará visualizando la función de autocorrelación de la serie (ACF). Si su ACF decae progresivamente, la serie no es estacionaria y es necesario diferenciar, por lo que el parámetro *d* tendría el valor 1; en caso contrario, tendría valor 0. La diferenciación transforma una serie temporal no estacionaria en estacionaria calculando diferencias consecutivas (Hyndman & Athanasopoulus, 2018):

$$\Delta y_t = y_t' = y_t - y_{t-1}$$

Una vez se haya determinado si hay que diferenciar o no, se puede proceder a la identificación de los otros dos parámetros de ARIMA: *p* (orden del componente autorregresivo), y *q* (orden del componente de media móvil). La función de autocorrelación (ACF) de la serie estacionaria permite identificar el parámetro *q* (número

de términos de media móvil), y la función de autocorrelación parcial (PACF) de la serie estacionaria ayuda a determinar el parámetro q (número de términos de autorregresión).

Una vez se ha determinado el posible orden del modelo ARIMA(p,d,q), se puede proceder a ajustar el modelo y estimar las series temporales. Como se ha mencionado anteriormente, un modelo ARIMA combina los componentes de autorregresión y de media móvil, aplicando diferenciación para hacer la serie estacionaria. Durante el proceso de ajuste del modelo, se determinan los coeficientes ϕ (correspondientes al modelo AR) y θ (correspondientes al modelo MA) (Hyndman & Athanasopoulus, 2018):

$$\text{Modelo AR (p): } y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \ldots + \phi_p y_{t-p} + \epsilon_t$$

$$\text{Modelo MA (q): } y_t = \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \ldots + \theta_q \epsilon_{t-q}$$

$$\text{Modelo ARIMA (p,d,q): } Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \ldots + \phi_p Z_{t-p} + \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \ldots + \theta_q \epsilon_{t-q}$$

Donde, si $d = 0 \rightarrow Z_t = y_t$

Donde, si $d = 1 \rightarrow Z_t = y_t - y_{t-1}$

Tras ajustar el modelo, se debe verificar que sus residuos sean ruido blanco, es decir, que muestren un patrón aleatorio, tengan media cero y varianza constante. Esta comprobación se puede realizar usando los gráficos ACF y PACF, asegurándose de que no haya autocorrelación en ningún horizonte temporal. Por otro lado, se debe comprobar que los coeficientes de autocorrelación (ϕ) y de media móvil (θ) son significativos mediante la computación de su p-valor. Si dicho estadístico es suficientemente bajo, se concluye que hay una baja probabilidad de que los coeficientes sean cero, por lo que son significativos.

Finalmente, cuando cumpla con todos los requisitos, el modelo se podrá usar para predecir valores futuros de una serie temporal, los precios de distintas posiciones de futuros del Brent en el caso de este trabajo.

c. Generación de escenarios futuros usando simulaciones de modelos ARIMA

Hasta este punto, el modelo ARIMA generado anteriormente proporciona una única trayectoria futura. Por lo tanto, es muy relevante simular escenarios futuros que permitan explorar otros posibles caminos que la serie temporal podría seguir en el futuro para entender en mayor profundidad la variabilidad y la incertidumbre de las predicciones, y conocer el rango de posibles futuros para preparase para cada uno de ellos.

Para ello, se usa la función “simulate” de R, que permite generar distintas trayectorias futuras de una serie temporal mediante simulaciones con un componente aleatorio. Al introducir un factor aleatorio en el proceso predictivo, se simulan nuevos errores que

siguen la distribución de los errores históricos del modelo. Estos errores no se habían predicho con el modelo ARIMA previamente explicado. De esta manera, cada escenario se basa en el modelo ARIMA previamente ajustado y en los errores simulados aleatoriamente según la distribución histórica del modelo.

Por lo tanto, al emplear esta técnica de Machine Learning la función "simulate" genera valores futuros de la serie utilizando los componentes autorregresivos, de diferenciación y de media móvil, además de introducir nuevos errores aleatorios (ϵ_t) basados en la distribución histórica de los errores del modelo:

Simulate(Modelo ARIMA (p,d,q)): $Z_t=\phi_1 Z_{t-1}+\phi_2 Z_{t-2}+\ldots+\phi_p Z_{t-p}+\underline{\boldsymbol{\epsilon_t}}+\theta_1\epsilon_{t-1}+\theta_2\epsilon_{t-2}+\ldots+\theta_q\epsilon_{t-q}$

Donde, si $d \rightarrow 0$, $Z_t = y_t$

Donde, si $d \rightarrow 1$, $Z_t = y_t - y_{t-1}$

d. Modelos de regresión dinámica con ARIMA

Hasta este punto se han explicado técnicas de predicción en base a series temporales históricas, pero no se han tenido en cuenta otras variables que también pueden ser relevantes. En esta sección, se extiende el modelo ARIMA incorporando una variable explicativa, en este caso el precio spot del Brent (la primera posición del *dataset*). Esta técnica se conoce como modelo de regresión dinámica con ARIMA, y se usará para predecir las curvas usando el precio spot del Brent como variable explicativa.

En el caso de los futuros del Brent, incluir la variable explicativa permite que el modelo tenga en cuenta factores externos que se crea que puedan tener lugar en el futuro, aumentando así la efectividad del modelo. Estos factores explicativos pueden ser eventos geopolíticos, instinto, fluctuaciones de la demanda global del Brent, incremento en el almacenamiento del petróleo, cambios en políticas energéticas…

De esta manera, un modelo de regresión dinámico con ARIMA (W_t) depende de dos variables: la variable explicativa (X_t), la cual hay que diferenciar si no es estacionaria (X_t'), y la variable temporal, que se refiere al modelo ARIMA (Z_t). Lo que hace ARIMA es modelar la dependencia temporal del error de la regresión (Harris y Sollis, 2003; Hyndman & Athanasopoulus, 2018):

Modelo de regresión dinámica con ARIMA: $W_t = \beta X_t' + Z_t$

ARIMA (p,d,q): $Z_t = \phi_1 Z_{t-1}+\phi_2 Z_{t-2}+\ldots+\phi_p Z_{t-p}+\epsilon_t+\theta_1\epsilon_{t-1}+\theta_2\epsilon_{t-2}+\ldots+\theta_q\epsilon_{t-q}$

Donde, si $d = 0 \rightarrow Z_t = y_t$

Donde, si $d = 1 \rightarrow Z_t = y_t - y_{t-1}$

Donde, si X es estacionaria $\rightarrow X_t' = X_t$

Donde, si X no es estacionaria $\rightarrow X_t' = X_t - X_{t-1}$

De esta manera, el modelo ARIMA original y sus simulaciones sólo consideran la estructura interna de la serie temporal, y al incorporar una variable explicativa, se puede modelar el impacto del precio spot del Brent en las componentes principales para mejorar la precisión de las predicciones de las curvas de futuros.

4. Resultados

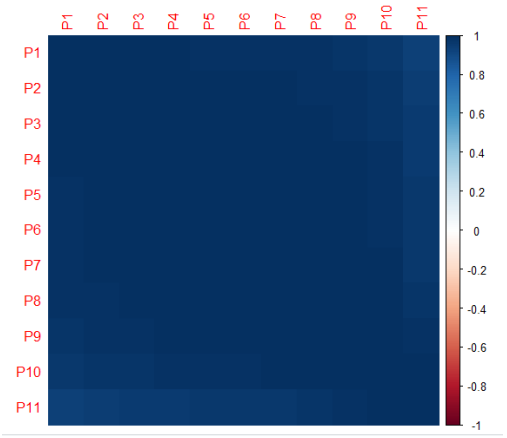
Como se ha comentado anteriormente, el Brent es un recurso de gran importancia a nivel global debido a su uso tan extendido en distintos sectores, desde la movilidad a través de producción de combustible, hasta la alimentación y el consumo a través de la producción de plásticos como envases. Por lo tanto, su volumen de negociación y liquidez es muy significativo, y surge la necesidad de modelar el comportamiento de los futuros del petróleo. Según se ha expuesto previamente, este trabajo implementará tres distintas técnicas de Machine Learning, una correspondiente a cada objetivo, para predecir o simular las curvas de futuros de este tipo de petróleo, lo que permitirá determinar las estrategias de trading.

En cuanto al primer objetivo, con el fin de predecir las curvas de futuros del Brent, se usará el modelo ARIMA. Más adelante, el segundo objetivo pretenderá generar escenarios posibles que predigan las curvas de futuros a partir de errores históricos, empleando la función "simulate" de R, que genera escenarios futuros usando simulaciones del modelo ARIMA. Por último, para predecir las curvas usando el precio del Brent como variable explicativa y cumplir con el tercer objetivo, se usará un modelo de regresión dinámico que permita combinar el modelo ARIMA con la regresión del precio spot del Brent como variable explicativa.

Según se ha mencionado en apartados anteriores, antes de empezar a modelar las curvas, es necesario llevar a cabo un análisis de componentes principales (PCA). El PCA no solo permite reducir la dimensionalidad y evitar la multicolinealidad del conjunto de datos para que sea factible llevar a cabo las predicciones de curvas de futuros, sino que también es útil para explorar las dinámicas de las posiciones de futuro según su vencimiento. El modelado de las curvas se llevará a cabo a partir de las componentes principales seleccionadas, y, tras ajustar el modelo y realizar las predicciones, se procederá a la reconstrucción de las curvas.

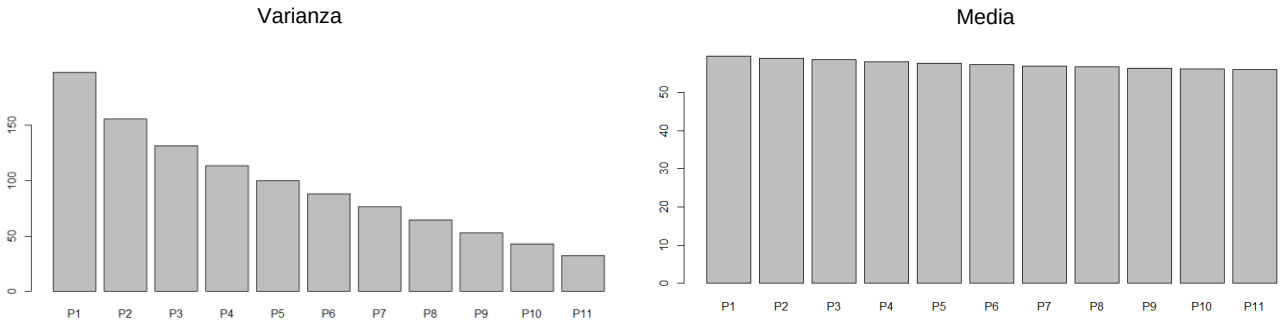
El PCA es especialmente útil cuando se implementa sobre un *dataset* con variables altamente correlacionadas. Además, dicha correlación es un síntoma de multicolinealidad, que se debe evitar a la hora de hacer predicciones. Según se muestra en el siguiente gráfico, el PCA será muy útil en este conjunto de datos dada la elevada correlación entre las posiciones de las curvas.

Figura 7: Correlación entre las posiciones de las curvas



Por otro lado, con el fin de ver si los datos son comparables entre sí, la siguiente ilustración muestra la varianza y la media de cada variable. Como se vio en el análisis exploratorio, la varianza disminuye a medida que el vencimiento de contrato se aleja, por lo que haría falta escalar las variables. Por otro lado, aunque muy ligeramente, la media disminuye a medida que vencimiento de contrato se aleja, por lo que centrar las variables tampoco vendría mal.

Figura 8: Necesidad de estandarización de las variables



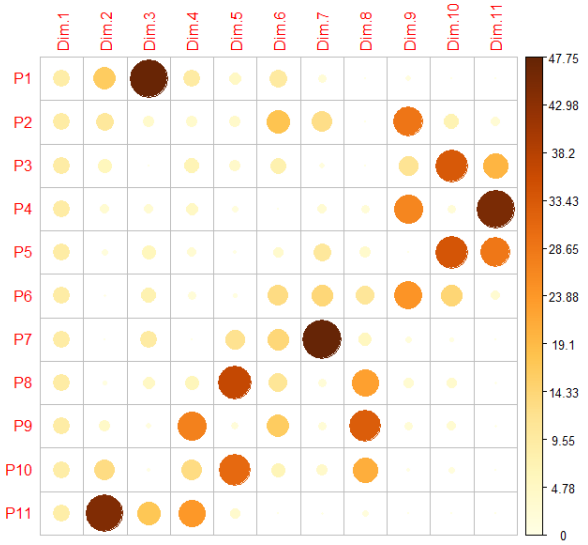
Una vez estandarizadas las variables, se pueden obtener los *loading vectors* para cada una de las componentes principales para definir los nuevos ejes en el espacio de las variables originales. Como se puede observar en la siguiente figura, existen tantas variables originales como componentes principales, las cuales están ordenadas según el grado de varianza explicada. En este caso, la primera componente principal se tiene una relación negativa con todas las variables originales, y la segunda componente principal pasa de tener asociaciones negativas a positivas según el vencimiento de las posiciones se aleja.

Figura 9: Loading vectors

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10	PC11
P1	-0.2995101	-0.404137041	0.68766392	-0.30577778	-0.21931785	-0.31057747	0.1506565	0.0399862971	-0.102504140	0.02729619	0.021865063
P2	-0.3014176	-0.322532496	0.19394517	0.19492214	0.19669564	0.42126077	-0.3590659	-0.0131295260	0.537559567	-0.26496188	-0.158383356
P3	-0.3022556	-0.243549515	-0.03474589	0.26191168	0.19779472	0.27732769	-0.1063746	-0.0008878006	-0.341527325	0.57941187	0.447002711
P4	-0.3027424	-0.171537121	-0.16475699	0.21881044	0.11951268	0.02339568	0.1671644	-0.1442858505	-0.516658119	-0.14622132	-0.672301947
P5	-0.3029858	-0.111498199	-0.24217171	0.16667182	0.05931973	-0.18526379	0.3225056	-0.1949939483	-0.005881312	-0.58465565	0.536293118
P6	-0.3031154	-0.052129799	-0.27350974	0.14127323	0.08937432	-0.36418382	0.3774288	0.3305724157	0.494642893	0.37879069	-0.173596074
P7	-0.3031216	-0.006351473	-0.30656275	-0.03004515	-0.34558028	-0.37562201	-0.6910369	0.2316841293	-0.121446408	-0.07992702	0.018131784
P8	-0.3030295	0.088300296	-0.21557276	-0.24471346	-0.60456403	0.33263038	0.1459331	-0.4775137336	0.175478364	0.18883851	-0.045943415
P9	-0.3024070	0.203999694	-0.10049566	-0.52176044	0.13522858	0.40429230	0.1493803	0.5722812427	-0.146919368	-0.17237110	0.041765824
P10	-0.3009043	0.364879758	0.05968191	-0.36302224	0.55691153	-0.25584732	-0.1992437	-0.4561132635	0.062100895	0.11473635	-0.024609975
P11	-0.2950381	0.667955784	0.41663306	0.48937252	-0.18691513	0.03202331	0.0425603	0.1134315805	-0.035442887	-0.04149909	0.009990019

Por otro lado, el siguiente gráfico muestra cuánto contribuye cada variable a la información explicada por cada componente principal, independientemente de su relación en términos de dirección. Mientras que para la primera componente el peso de cada una de las variables originales está muy repartido, para el resto, suele existir una variable original que afecta más que el resto a la componente principal.

Figura 10: Gráfico de contribución de cada variable a las componentes principales



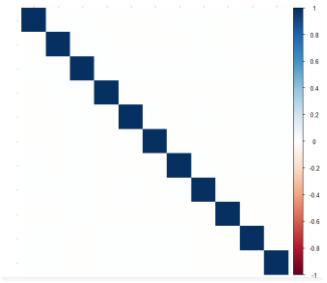
Además, al disponer de los *loading vectors*, ya se conoce el eje sobre el que se proyectan las componentes principales, por lo que se pueden calcular los *scores*, que se muestran a continuación. Serán estos datos los que se predigan en cada uno de los modelos que se implementarán más tarde. Más adelante, se podrán reconstruir las curvas originales a partir de los *scores* usando los *loading vectors*

Figura 11: Scores de las componentes principales

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10	PC11
[1,]	-1.695520	0.6559860	-0.01523010	-0.03927536	-0.010779535	0.012349552	-0.0076484883	0.007905436	0.004611151	-0.001701798	0.001487062
[2,]	-1.634497	0.6189109	-0.03805732	-0.05445377	-0.010609081	0.009837310	-0.0056788247	0.008456389	0.006939819	-0.001699154	0.002164229
[3,]	-1.512661	0.5888059	-0.06015868	-0.06335519	-0.008439416	0.004776578	-0.0044398644	0.006932570	0.006629865	-0.001842671	0.001835181
[4,]	-1.661173	0.5474290	-0.09539322	-0.08773888	-0.008021598	0.003398198	-0.0029943175	0.007028435	0.008492823	-0.001886128	0.002229815
[5,]	-1.308451	0.5237887	-0.06808990	-0.07014260	-0.014547974	0.001177758	-0.0014959746	0.008835319	0.008897851	-0.003318668	0.001966608
[6,]	-1.335118	0.4822001	-0.06632248	-0.07068536	-0.013848108	0.002952177	-0.0009360299	0.010097908	0.008817411	-0.003686918	0.001865815

Antes de seleccionar el número de componentes principales que se usarán para los posteriores análisis, es necesario realizar una serie de comprobaciones para asegurarse que el PCA se ha llevado a cabo correctamente. Por un lado, la siguiente figura demuestra que no existe correlación entre las componentes principales; por otro lado, el producto escalar de los *loading vectors* es igual a cero, confirmando así que son ortogonales. De esta manera, se confirma que el análisis PCA es adecuado.

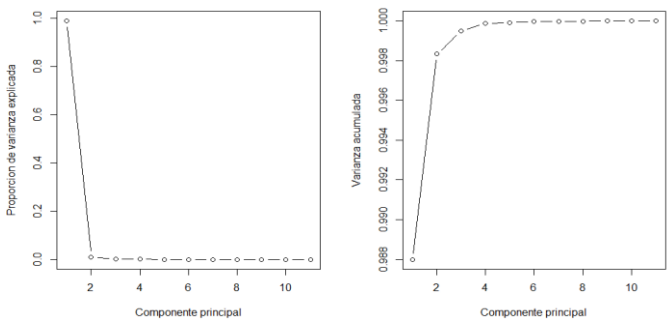
Figura 12: Correlación entre las componentes principales



Una vez analizadas todas las componentes principales y comprobado que se han calculado correctamente, es hora de seleccionar aquellas que retengan más información. Como se puede observar, la primera componente principal es capaz de representar el 98,8% de la varianza de las variables originales. A partir de ahí, la capacidad del resto de componentes principales es mucho menor, situándose por debajo del 5%. Usando el método del codo, se puede observar cómo, aunque la curva de varianza acumulada se tuerce notablemente entre la primera y la segunda componente principal, también hay un ligero punto de inflexión ente la segunda y la tercera. Más allá de ese punto, añadir componentes adicionales no mejora significativamente la variabilidad explicada. Por lo tanto, se emplearán las dos primeras componentes principales, que juntas representan un 99,8% de la información para implementar las técnicas de Machine Learning posteriormente.

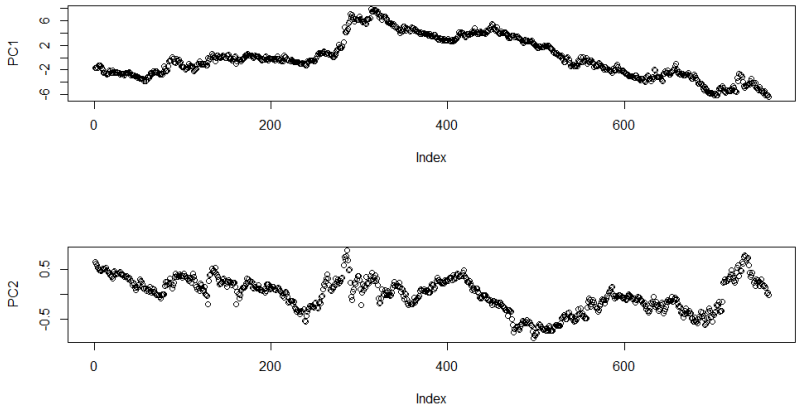
Figura 13: Proporción de varianza explicada

```
Importance of components:
                          PC1     PC2     PC3     PC4     PC5     PC6     PC7     PC8     PC9    PC10    PC11
Standard deviation     3.297 0.33724 0.11291 0.06445 0.02436 0.02323 0.01159 0.009634 0.007036 0.0026 0.00165
Proportion of Variance 0.988 0.01034 0.00116 0.00038 0.00005 0.00005 0.00001 0.000010 0.000000 0.0000 0.00000
Cumulative Proportion  0.988 0.99833 0.99949 0.99987 0.99992 0.99997 0.99999 0.999990 1.000000 1.0000 1.00000
```



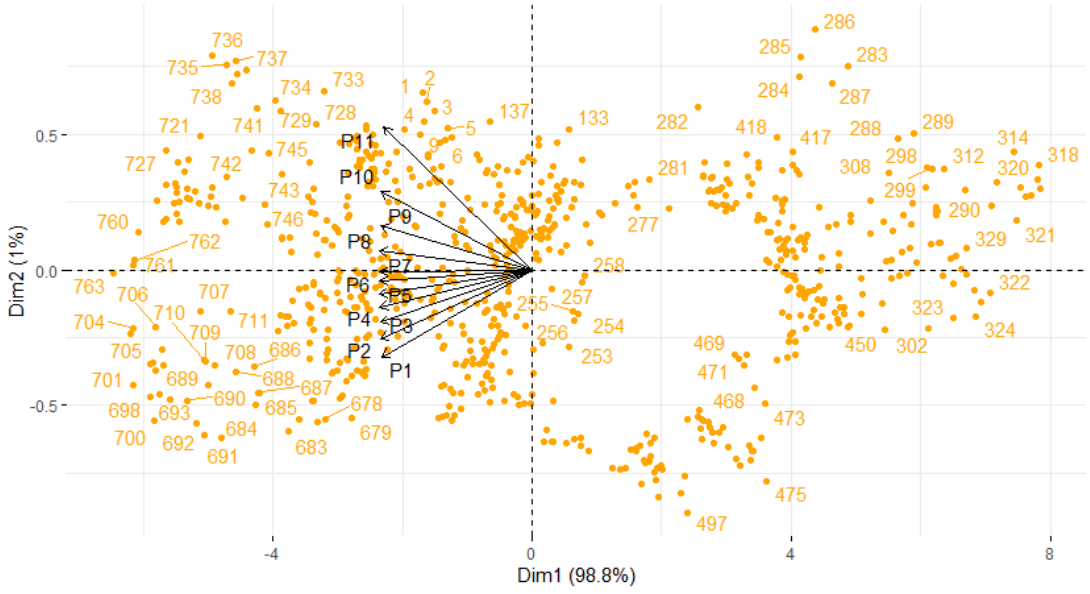
A modo ilustrativo, a continuación, se muestra la evolución de primeras componentes principales. Aunque a primera vista, poco tienen que ver con la serie original de los precios de futuros que se expone en la figura 4, en realidad la primera componente en realidad parece seguir un patrón opuesto a la serie original, por lo que en realidad sí están más relacionadas de lo que pueda parecer. Por otro lado, aunque ahora no sea especialmente relevante, cabe destacar que la media y la varianza de estas series cambia con el paso del tiempo, lo cual será imprescindible saber a la hora de ajustar el modelo ARIMA.

Figura 14: Evolución de las dos primeras componentes principales



El siguiente gráfico es un *biplot* con las dos primeras componentes principales, que permite visualizar la relación entre variables originales y la distribución de las observaciones en el espacio definido por las dos primeras componentes principales.

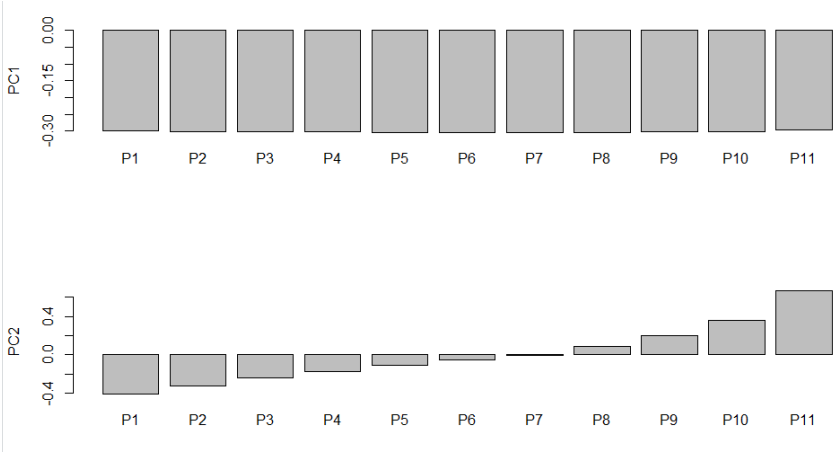
Figura 15: Biplot con las dos primeras componentes principales



Las flechas negras hacen referencia a los *loading vectors* y los puntos naranjas a los *scores*. Como se puede observar, la primera componente principal tiene una relación negativa con todas las variables originales. Por otro lado, las posiciones con vencimientos más cercanos tienen un impacto negativo sobre la segunda componente principal, mientras que las posiciones con vencimientos más lejanos tienen un impacto positivo.

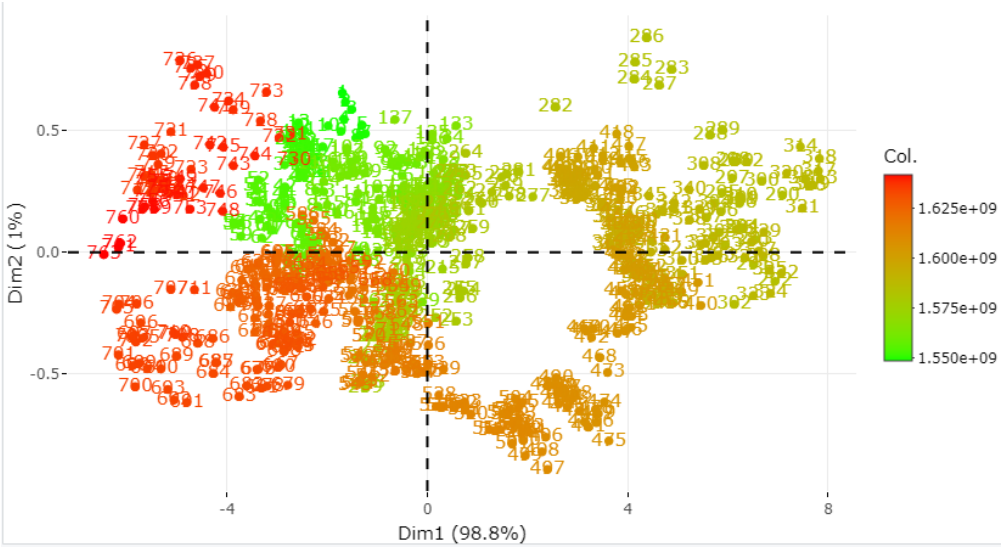
El *biplot* permite saber la relación direccional entre las variables originales y las componentes principales. Sin embargo, no cuantifica el peso de dicha relación direccional. El siguiente gráfico muestra el peso de cada variable en cada componente en términos de dirección y carga. Es otra manera de visualizar los *loading vectors*, Como se ha demostrado anteriormente, todas las variables originales contribuyen con la misma carga de información a la primera componente principal, y la relación entre ambas siempre es negativa. Sin embargo, en cuanto a la segunda componente principal, las variables P11, P1, P10 y P2 son las que más información aportan, mientras que la variable P7 apenas contribuye. Además, a medida que se aleja el vencimiento de los contratos de futuros, la relación entre las variables originales y la segunda componente principal se vuelve más positiva.

Figura 16: Carga de las variables en cada componente principal



Por otro lado, hasta este punto no se ha contemplado el efecto del paso del tiempo en los valores de las componentes principales seleccionadas. El siguiente gráfico muestra la proyección de los scores en los ejes de las dos primeras componentes principales según la fecha, permitiendo así la formación de grupos de información de manera sencilla. Los puntos verdes hacen referencia a los *scores* de las fechas más cercanas (empezando el 1 de febrero de 2019), y los puntos rojos a los de las fechas más tardías (acabando el 14 de enero de 2022).

Figura 17: Valores de las componentes principales según la fecha



La incorporación de un factor temporal en la representación de las componentes principales seleccionadas permite formar grupos de datos según su fecha de ocurrencia. Además, si se compara con los precios de los futuros del *dataset* original, se podría llegar a conclusiones más sólidas.

Durante 2019, cuando la curva de futuros estaba en *backwardation*, las observaciones se sitúan en el cuadrante superior izquierdo, por lo que la primera componente principal tiene fuerza negativa y la segunda positiva. Sin embargo, a medida que las noticias de la inminente llegada del COVID-19 a nivel global se acentuaban (entre finales de 2019 y principios de 2020), la segunda componente empieza a cobrar fuerza negativa.

Con la llegada definitiva del COVID-19, los precios de los futuros caen notablemente y la curva se sitúa en contango, y la primera componente principal comienza a tomar fuerza positiva. En esto punto, las observaciones empiezan situándose en el cuadrante superior derecho, aunque en el último trimestre de 2020 la segunda componente principal empieza a cobrar fuerza negativa, y las observaciones pasan a reflejarse en el cuadrante inferior derecho.

A medida que las restricciones de la pandemia se iban retirando a principios de 2021, los precios de los futuros aumentan y la curva vuelve a estar en *backwardation*, la primera componente principal vuelve a tomar fuerza negativa y las observaciones se representan en el cuadrante inferior izquierdo. Por último, durante los últimos meses disponibles, la segunda componente pasa a tomar fuerza positiva y las observaciones vuelven al cuadrante inicial: el superior izquierdo.

Por lo tanto, aunque no existan conclusiones claras sobre la relación de la segunda componente principal con los precios de futuros del Brent, la primera componente principal es capaz de capturar si la curva está en contango (cobra fuerza positiva) o en *backwardation* (cobra fuerza negativa).

Por último, para comprobar la utilidad y la fiabilidad del análisis de componentes principales, se ha llevado a cabo una reconstrucción de las curvas usando las dos primeras componentes principales, recogiendo de esta manera un 99,8% de la varianza explicada. Para realizar la reconstrucción se ha llevado a cabo el proceso inverso al de la creación de las componentes principales: multiplicando los *scores* por los *loading vectors*, y desestandarizando los resultados. A continuación, se muestra el gráfico original de series temporales de precios las posiciones de futuros, y su reconstrucción. Ambos son prácticamente idénticos, por lo que se demuestra la utilidad de emplear las dos primeras componentes principales.

Figura 18: PCA - Serie temporal original vs. reconstruida

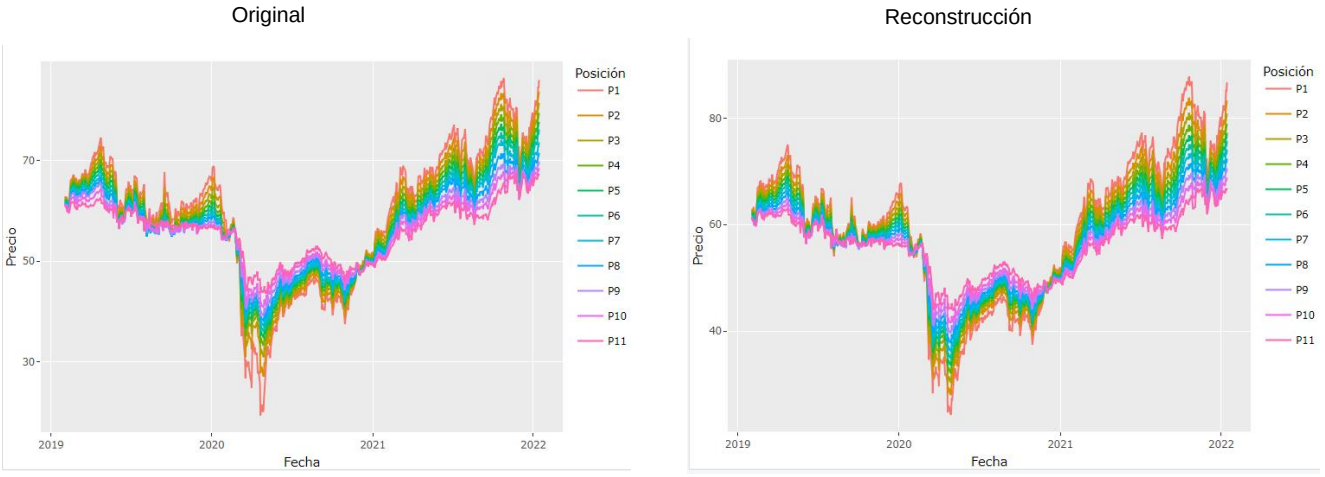
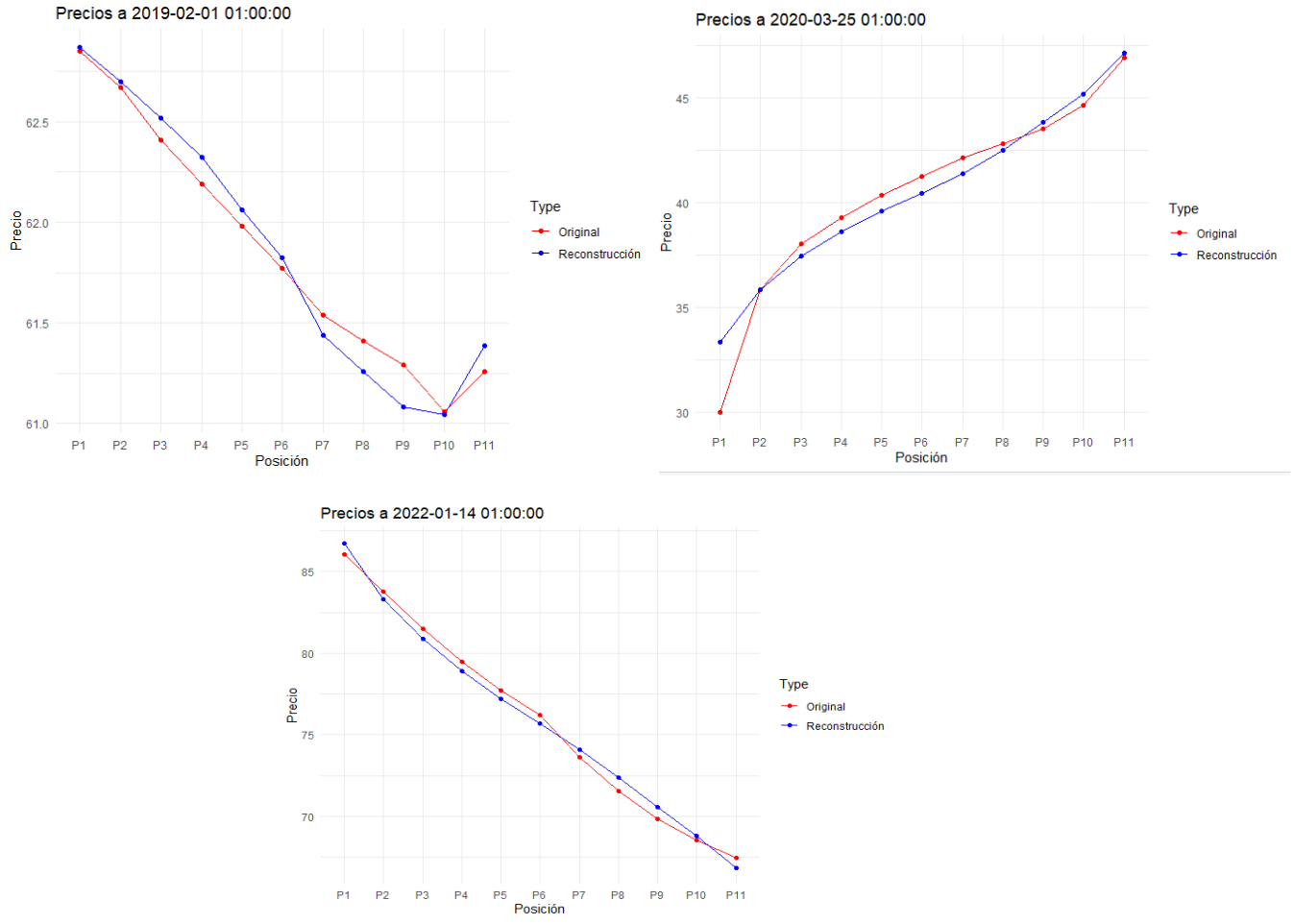


Figura 19: PCA - Serie temporal original vs. reconstruida – posiciones 1 y 11

Además, también se pueden reconstruir las curvas de futuros en sí. Como se puede observar en la siguiente figura, las reconstrucciones son decentes, aunque hay puntos de la curva que no se han logrado reconstuir de manera ideal. Por ejemplo, para el 1 de febrero de 2019 el precio reconstruido de las posiciones P1-P6 es más alto que el original, pero el precio reconstruido de las posiciones P7-P9 es menor que el original. Existen diferencias similares en el resto de días graficados, aunque las diferencias no son sigificativas, por lo que se puede concluir que la precisión de usar dos componentes principales es adecuada.

Figura 20: PCA - Curvas de fututos originales vs. reconstruidas



En conclusión, se ha establecido que se pueden usar las dos primeras componentes principales como base para modelar las curvas de futuros usando técnicas de Machine Learning en los siguientes apartados. Como recordatorio, para ello, se dividirá el conjunto de datos de componentes principales en datos de entrenamiento, y datos de prueba (las dos últimas semanas disponibles). El proceso será el mismo para los tres modelos empleados:

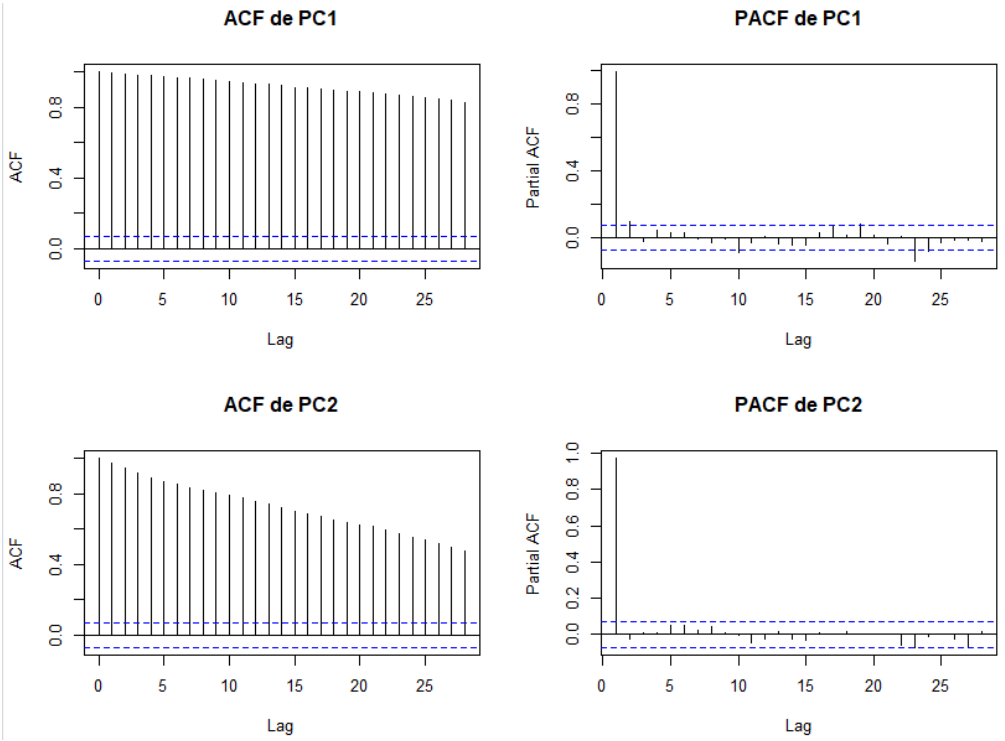
- Ajuste de un modelo para cada componente principal usando los datos de entrenamiento correspondientes.
- Predicción o simulación de cada una de las componentes principales usando el modelo previamente ajustado.
- Reconstrucción de las curvas de futuros a partir de las componentes principales predichas y comparación con las curvas originales.
- Análisis de las predicciones.

4.1. Objetivo I: Predecir las curvas de futuros del Brent

Con el objetivo de predecir las curvas de futuros de Brent, se usará el modelo ARIMA siguiendo los pasos descritos en el apartado "Metodología". El modelo ARIMA(p,d,q) es una técnica de series temporales que combina componentes de autorregresión (p), de diferenciación (d) y de media móvil (q) para predecir valores futuros basados en datos históricos (Hyndman & Athanasopoulus, 2018).

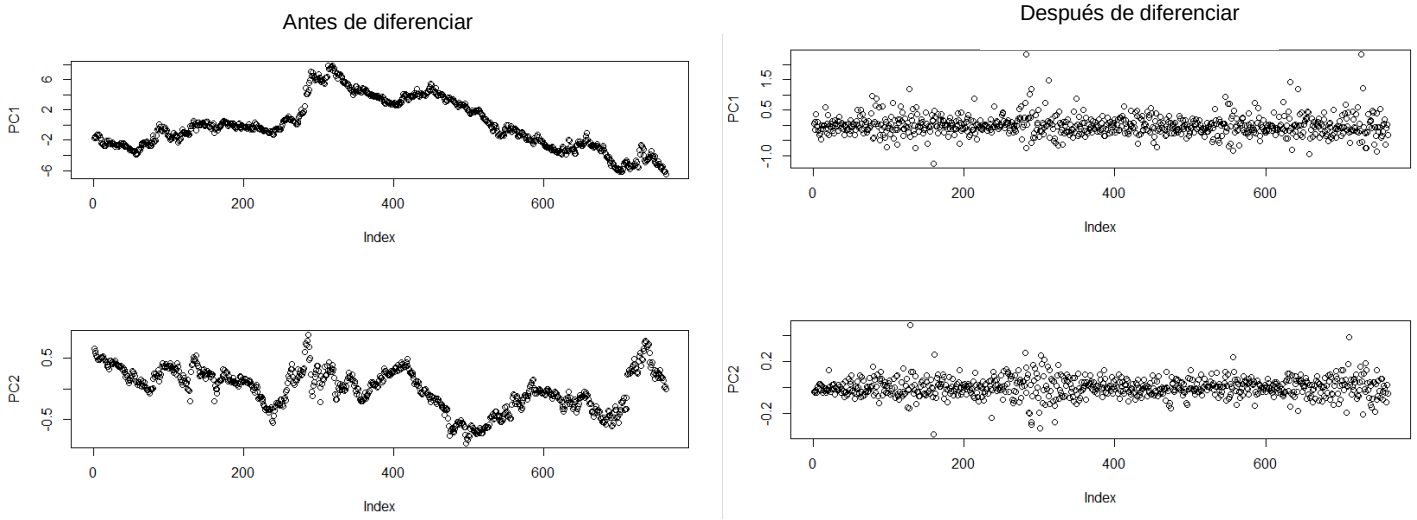
Según lo explicado, el primer paso consiste en determinar el orden del modelo, en particular el parámetro *d*. Para ello, hay que explorar si la serie estacionaria o no. Si no lo es, se ha de diferenciar y el parámetro *d* será 1.

Figura 21: Objetivo I - ACF y PACF de las dos primeras componentes principales



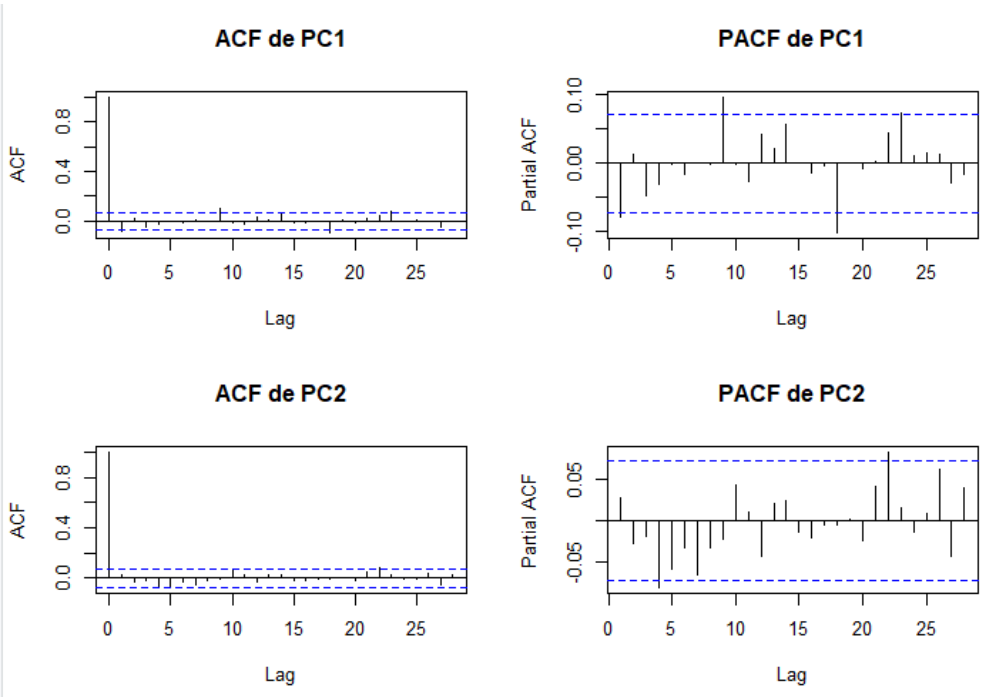
Como se puede observar, el ACF de ambas componentes decae progresivamente, lo que significa que son series no estacionarias y que se ha de diferenciar. Además, el siguiente el gráfico confirma que las series no son estacionarias porque sus medias y sus varianzas cambian con el tiempo. Además, se muestra cómo, al diferenciar, las series se transforman en estacionarias.

Figura 22: Objetivo I - Evolución de las dos primeras componentes principales antes y después de diferenciar



Para determinar el orden del componente autorregresivo (p) y del de media móvil (q), hay que visualizar el número de rezagos significativos en el PACF y el ACF respectivamente,

Figura 23: Objetivo I - ACF y PACF de las series diferenciadas



En el caso de la primera serie temporal, tanto el ACF como el PACF muestran un rezago significativo, por lo que habrá que implementar un ARIMA (1,1,1). Por otro lado, tanto el ACF como el PACF de la segunda componente principal parecen ser ruido blanco, por lo que su modelo será un ARIMA (0,1,0). Aunque no existan componentes autorregresivos ni de media móvil en el modelo de la segunda componente principal, se seguirá modelando ya que será el modelo base que se use a la hora de implementar el segundo objetivo. Además, con el simple hecho de diferenciar a través del parámetro *q* se consigue que las predicciones sean iguales al último valor disponible, en vez de igual a cero.

Una vez ajustado el modelo, se pueden realizar las comprobaciones necesarias. En primer lugar, hay que verificar si los coeficientes del modelo ARIMA(1,1,1) de la primera componente principal son significativos.

Figura 24: Objetivo I - Test Z de coeficientes significativos del modelo ARIMA(1,1,1) para la primera componente principal

```
z test of coefficients:

    Estimate Std. Error z value Pr(>|z|)
ar1 -0.41307    0.36488 -1.1321   0.2576
ma1  0.33709    0.37633  0.8957   0.3704
```

En esto caso, según el Test Z de coeficientes significativos, el p-valor es alto en ambos coeficientes. En particular, el de media móvil indica que hay un 37% de probabilidades de que el coeficiente en realidad sea igual a cero. Por lo tanto, se volverá a ajustar el modelo para que sea un ARIMA (1,1,0), y se volverá a estudiar si esta vez, el coeficiente autorregresivo es significativo.

Figura 25: Objetivo I - Test Z de coeficientes significativos del modelo ARIMA(1,1,0) para la primera componente principal

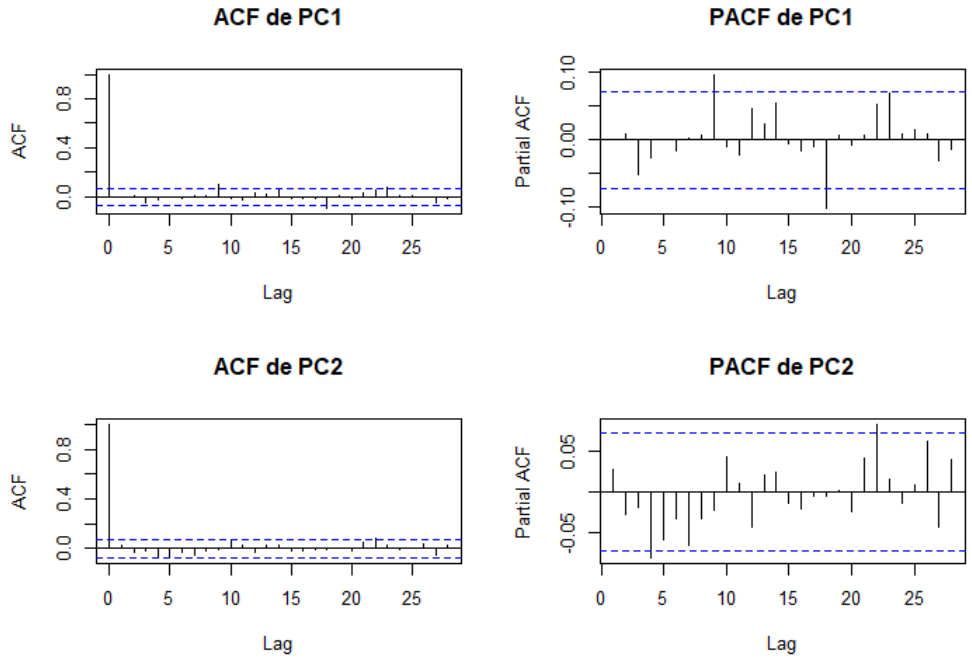
```
z test of coefficients:

     Estimate Std. Error z value Pr(>|z|)
ar1 -0.079395   0.036590 -2.1698  0.03002 *
```

El modelo ARIMA (1,1,0) para la primera componente asigna un coeficiente significativo al componente autorregresivo. Solo hay un 3% de probabilidades de que el coeficiente sea igual a 0. Por lo tanto, se llevará a cabo la segunda fase de verificación para comprobar si este modelo es válido.

El segundo paso para verificar que el orden del modelo es el adecuado consiste en el

supuesto de ruido blanco. Es decir, que los residuales del modelo no se pueden predecir porque son aleatorios, y se asume que siguen una distribución normal. Para ello, se ha de graficar el PACF y el PACF de los residuales del modelo.

Figura 26: Objetivo I - ACF y PACF de los residuales de los modelos



Como se puede observar, los residuales de ambos modelos son ruido blanco así que no se pueden predecir. Otra manera de llevar a cabo esta comprobación es computando el p-valor de la prueba de Ljung-Box.

Figura 27: Objetivo I - Prueba de Ljung-Box

```
> Box.test(model_PC1$residuals, type = "Ljung-Box")

        Box-Ljung test

data:  model_PC1$residuals
X-squared = 0.00048383, df = 1, p-value = 0.9825

> Box.test(model_PC2$residuals, type = "Ljung-Box")

        Box-Ljung test

data:  model_PC2$residuals
X-squared = 0.57128, df = 1, p-value = 0.4498
```

En este caso, como ambos p-valores son suficientemente altos, se confirma que los residuales de ambos modelos son ruido blanco. Por lo tanto, se concluye que se predecirá la primera componente principal con un ARIMA (1,1,0) y la segunda con un ARIMA (0,1,0).

ARIMA (1,1,0): $Z_t = \phi_1 Z_{t-1} + \epsilon_t$
ARIMA (0,1,0): $Z_t = \epsilon_t$
Donde, $Z_t = y_t - y_{t-1}$

Se ha comprobado que los modelos son significativos y que sus residuales son aleatorios, pero no se conoce cómo de precisos son. Por lo tanto, a continuación, se presentan las métricas de error del conjunto de datos de entrenamiento. Por el momento, la mayoría de los indicadores están en valores absolutos y aún no se han generado otros modelos de las mismas series temporales para poder compararlos. Sin embargo, a medida que se vayan generando más modelos, se podrán comparar estas métricas para evaluar la efectividad de cada modelo.

Figura 28: Objetivo I - Métricas de error del dataset de entrenamiento de la primera componente principal usando ARIMA(1,1,0)

```
Coefficients:
          ar1
      -0.0794
s.e.   0.0366

sigma^2 estimated as 0.1075:  log likelihood = -227.29,  aic = 458.57

Training set error measures:
                     ME      RMSE       MAE      MPE     MAPE      MASE          ACF1
Training set -0.004576995 0.3276684 0.2244461 52.62698 88.26406 0.9948403 0.0008021137
```

Figura 29: Objetivo I - Métricas de error del dataset de entrenamiento de la segunda componente principal usando ARIMA(0,1,0)

```
sigma^2 estimated as 0.005687:  log likelihood = 872.06,  aic = -1742.12

Training set error measures:
                      ME       RMSE        MAE      MPE     MAPE     MASE        ACF1
Training set -0.0004706877 0.07536136 0.05430462 207.7504 285.6842 0.998681 0.02756223
```

Una vez ajustado el modelo a los datos de entrenamiento, se utilizará para predecir los valores futuros de cada componente principal. Se realizarán predicciones para un horizonte de 14 días y luego se compararán con el conjunto de datos de prueba para evaluar la precisión del modelo.

En primer lugar, se usa la función *forecast()* para predecir los valores futuros de cada componente principal, y éstos se comparan con el conjunto de datos de prueba. Dicha función genera un rango de predicciones para distintos intervalos de confianza.

Figura 30: Objetivo I - Predicción de la primera componente principal con forecast()

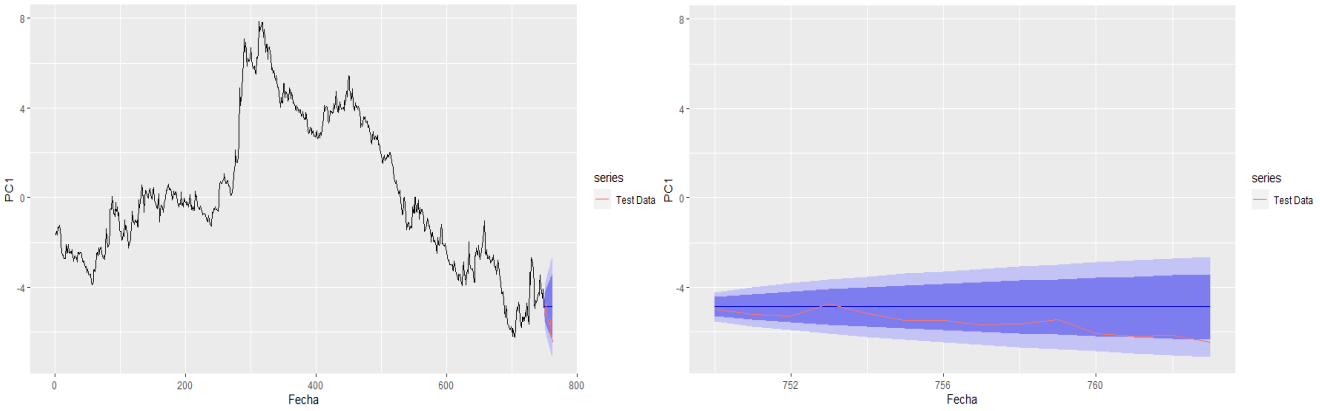
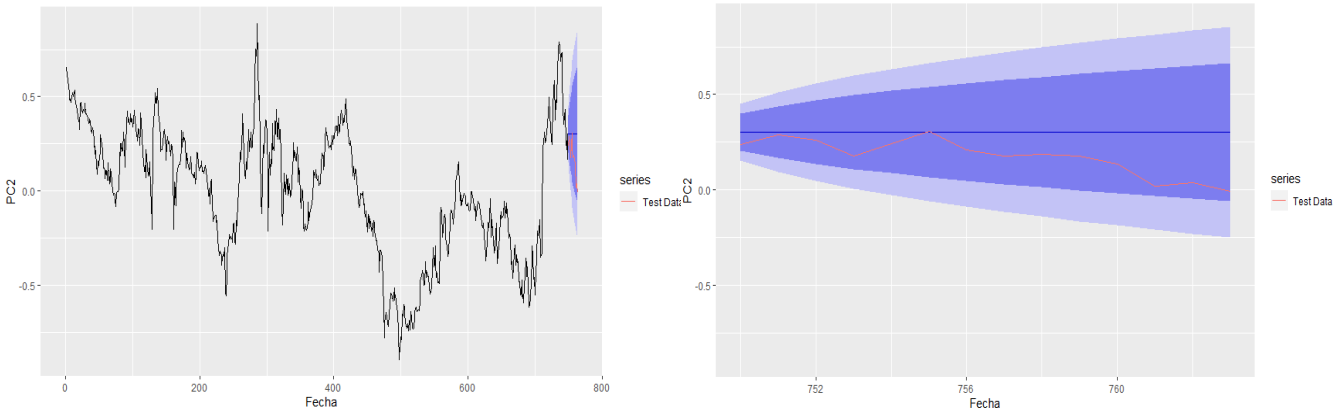


Figura 31: Objetivo I - Predicción de la segunda componente principal con forecast()



Como se puede observar para ambas componentes principales, los datos de prueba reales están dentro del rango de predicciones basados en distintos intervalos de confianza. De esta manera, se justifica que tiene sentido generar escenarios simulados para observar cuál se puede acercar más a la realidad, que es lo que se investigará en el segundo objetivo.

Por el momento, se usará la media del rango de predicciones de los modelos como valores futuros, cuyos resultados se muestran a continuación:

Figura 32 Objetivo I - Predicción media de la primera componente principal con forecast()

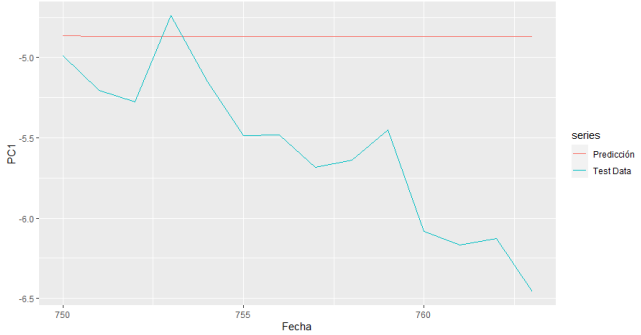


Figura 33: Objetivo I - Predicción media de la segunda componente principal con forecast()

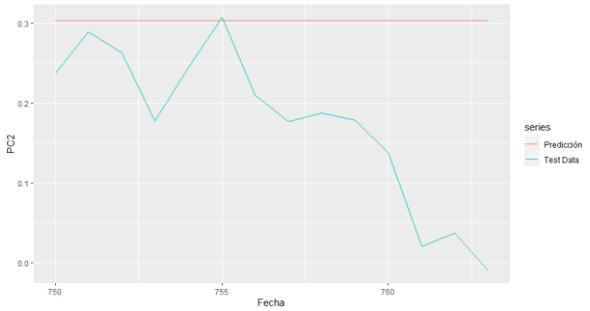


Tabla 3: Objetivo I - Errores de predicción (MSE) con forecast()

Primera componente principal	0,715
Segunda componente principal	0,025

Al usar la media del rango de predicciones de cada modelo ARIMA, es evidente que las predicciones son idénticas o muy parecidas al último dato de entrenamiento disponible, y que no captan los patrones de movimientos adecuadamente. En el caso de la primera componente principal, los cambios de los valores predichos apenas cambian porque el coeficiente autorregresivo es muy pequeño (0,079) y no hay ningún coeficiente de media móvil. En el caso de la segunda componente principal, no hay ningún coeficiente, por lo que las predicciones medias son iguales al último valor disponible. Esto se debe a que en horizontes posteriores a t+1, la función *forecast()* genera predicciones basadas en predicciones anteriores, por lo que si el coeficiente es demasiado pequeño o nulo, como es el caso, las predicciones apenas variarán.

Lo más común es entrenar un modelo con los datos de entrenamiento y luego evaluar su efectividad con los datos de prueba, como se ha hecho hasta ahora con forecast(). Sin embargo, esto provoca el problema de generar pronósticos basados en valores predichos, por lo que el error de predicción se va acumulando.

Para solventar este problema, se puede aplicar el modelo a los datos de prueba sin acumular los errores en cada predicción. Esto significa que se usan datos de entrenamiento para estimar los parámetros, pero que para calcular las predicciones, se pueden usar las observaciones anteriores de los datos de prueba (Hyndman & Athanasopoulus, 2018). Así, en lugar de predecir sobre valores previamente predichas, se usan valores reales de periodos anteriores del conjunto de prueba para generar predicciones. Además, como sólo

se han usado los datos de entrenamiento para estimar los parámetros, se puede decir que la predicción es justa. Cabe mencionar que, en el caso de los modelos aquí implementado, ARIMA(1,1,0) y ARIMA(0,1,0), los datos empleados en cada predicción son todos los datos de entrenamiento y el dato de prueba inmediatamente anterior (no todos). Esto se debe a que la diferenciación es de primer nivel y a que el componente autorregresivo tiene un rezago de un periodo, considerando sólo la observación del periodo justo anterior.

Figura 34: Objetivo I - Métricas de error de la primera componente principal usando ARIMA(1,1,0) aplicado a los datos de test.

```
Coefficients:
          ar1
      -0.0794
s.e.   0.0000

sigma^2 = 0.1075:  log likelihood = -2.92
AIC=7.84   AICc=8.2   BIC=8.4

Training set error measures:
                     ME      RMSE       MAE      MPE     MAPE      MASE       ACF1
Training set -0.1115256 0.2918147 0.2234415 1.807786 4.058112 0.9384134 -0.2690597
```

Como se puede observar, el coeficiente del componente autorregresivo es el mismo que el estimado previamente, ya que solo se ha calculado en base a los datos de entrenamiento. Sin embargo, las métricas de error cambian respecto a la versión anterior (figura 28). El modelo aplicado a los datos de prueba muestra menores valores en cuanto a las medidas de error, RMSE, MAE, MPE, MAPE y MASE, lo cual indica una mejor precisión en las predicciones. Además, la autocorrelación de los errores también es menor en este caso, indicando que los errores de predicción son más independientes que en la primera versión, probablemente porque en este escenario no se han ido acumulando.

Figura 35: Objetivo I - Métricas de error de la segunda componente principal usando ARIMA(0,1,0) aplicado a los datos de test

```
sigma^2 = 0.005687:  log likelihood = 18.01
AIC=-34.01   AICc=-33.65   BIC=-33.45

Training set error measures:
                      ME       RMSE        MAE       MPE     MAPE      MASE       ACF1
Training set -0.01763169 0.05835667 0.04754075 -11.07487 94.01864 0.9289022 -0.1235572
```

Algo similar ocurre con la segunda componente principal. Si se compara con la figura 29, las métricas de errores suelen ser menores, indicando una mejor precisión.

Una vez se ha ajustado el modelo para que pueda tener en cuenta todas las observaciones de entrenamiento y la observación anterior del conjunto de prueba, se

pueden realizar las predicciones con la fusión *fitted()*.

Figura 36: Objetivo I - Predicción media de la primera componente principal con fitted()

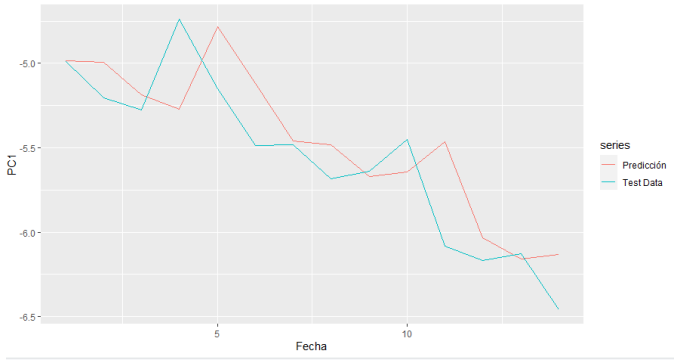


Figura 37: Objetivo I - Predicción media de la segunda componente principal con fitted()

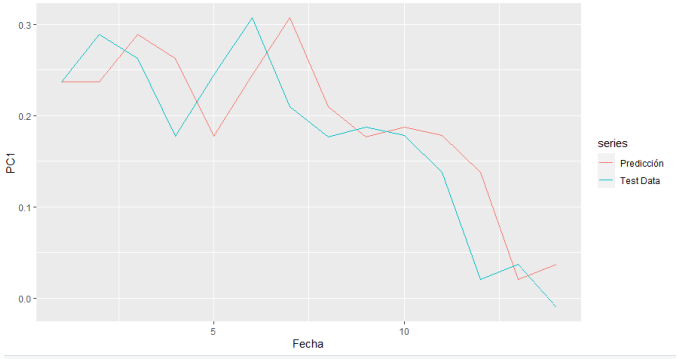


Tabla 4: Objetivo I - Comparación de errores de predicción (MSE) con forecast() y con fitted()

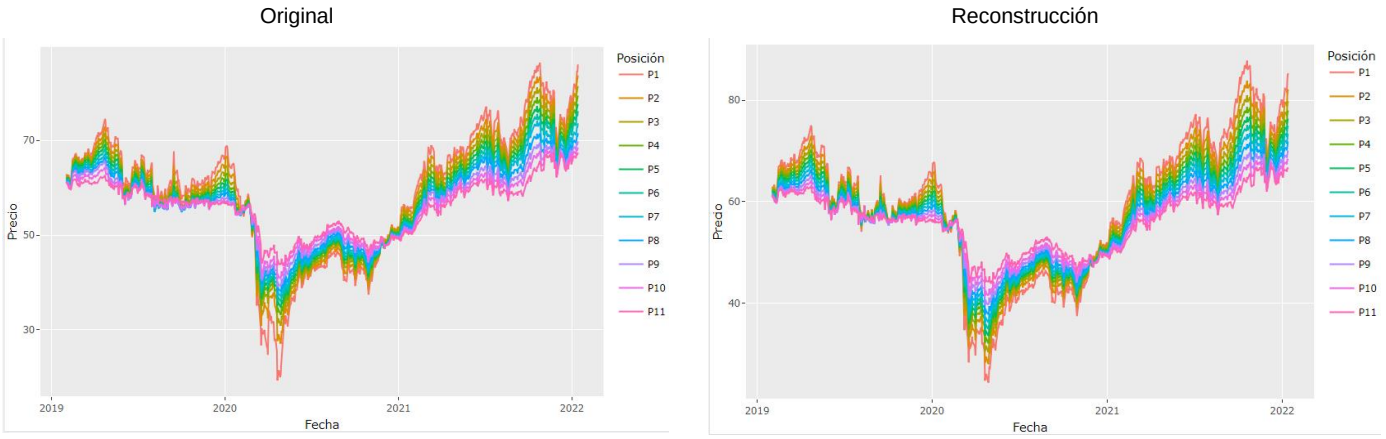
Errores de predicción (MSE)	*ARIMA forecast()*	*ARIMA fitted()*
Primera componente principal	0,715	0,085
Segunda componente principal	0,025	0,003

Según se observa en los gráficos, las predicciones se ajustan mucho mejor a la realidad, llevando a unos errores notablemente más bajos. Mientras que el modelo de la segunda componente principal hace que las predicciones sean las mismas a la observación real del día inmediatamente anterior, el modelo de la primera componente principal es capaz de generar pronósticos que difieren ligeramente de la observación real del día anterior por tener un componente autorregresivo.

Teniendo en cuenta que los pronósticos de los modelos aplicados al conjunto de prueba son significativamente mejores, serán estas predicciones las que se usen para reconstruir las curvas y proyectar los precios de posiciones de derivados del Brent. A modo de recordatorio, la reconstrucción se llevará a cabo en base a las siguientes predicciones generadas por los siguientes modelos:

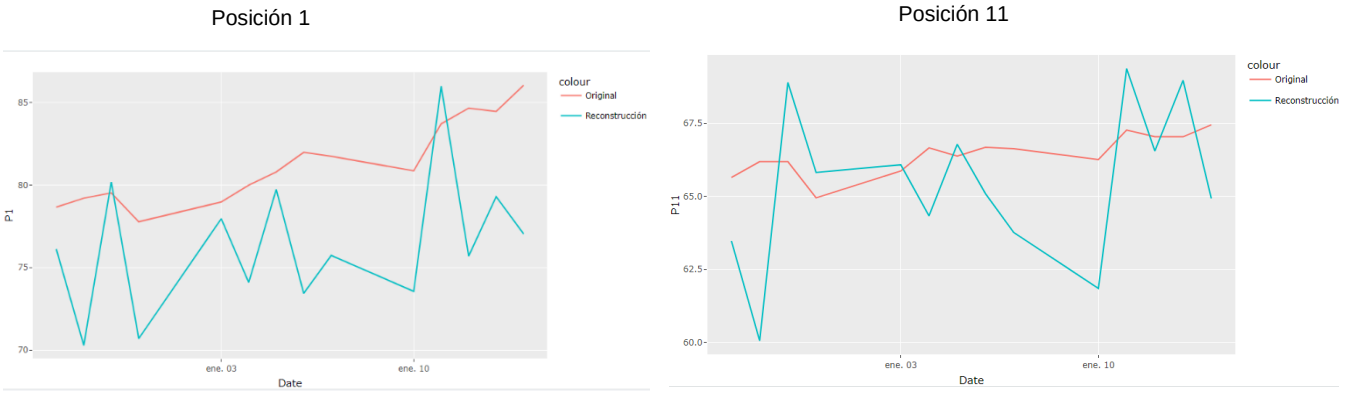
- Primera componente principal: ARIMA(1,1,0) aplicado a los datos de prueba.
- Segunda componente principal: ARIMA(0,1,0) aplicado a los datos de prueba.

Figura 38: Objetivo I - Serie temporal original vs. reconstruida



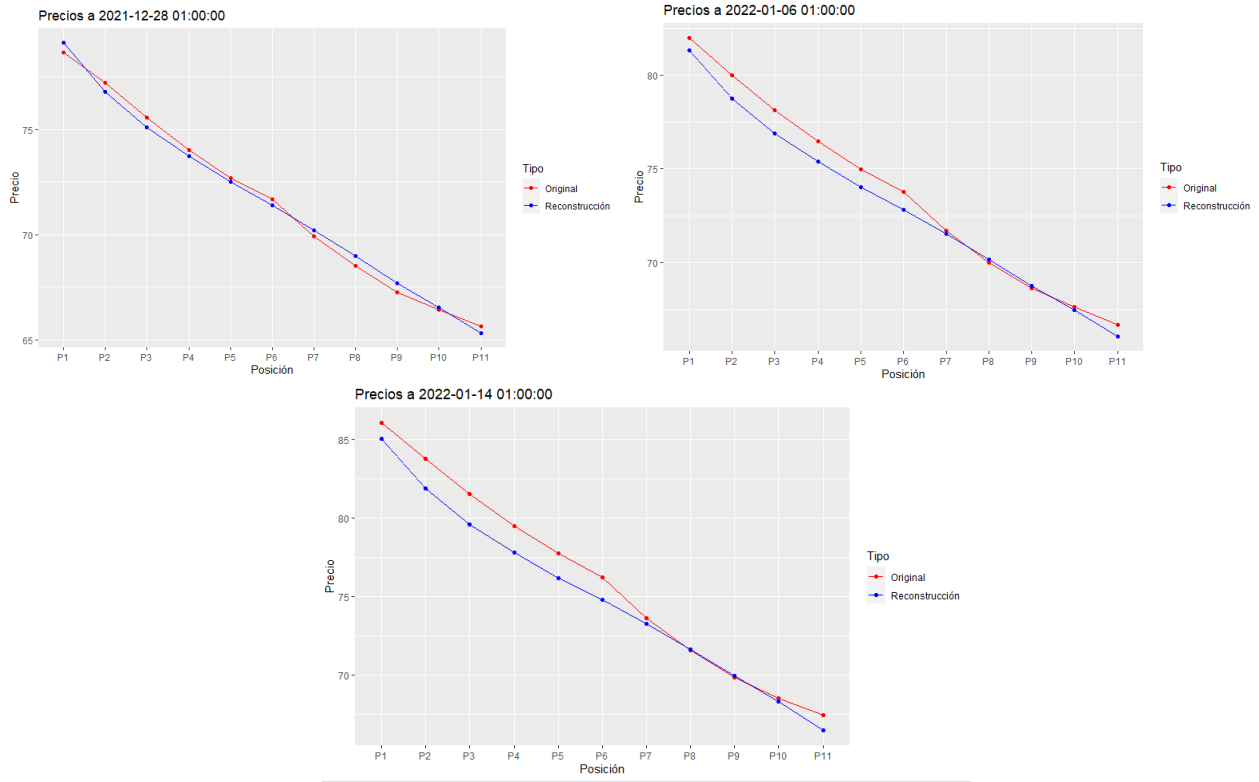
Según muestran los gráficos anteriores, las predicciones parecen muy parecidas a la realidad, y se consigue predecir el precio de las distintas posiciones de futuros con el paso del tiempo.

Figura 39: Objetivo I - Serie temporal original vs. reconstruida – posiciones 1 y 11



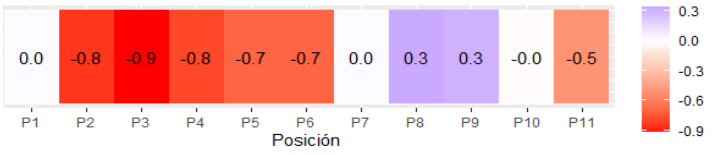
Sin embargo, si uno observa el detalle de la figura 39, ve que los precios predichos son más volátiles que los originales y el ajuste no es perfecto.

Figura 40: Objetivo I - Curvas de fututos originales vs. reconstruidas



Como se puede observar, en general, se suele predecir un precio menor al real en las primeras posiciones (desde la segunda posición hasta la sexta), cuya diferencia se acentúa a medida que se proyectan fechas más lejanas. Sin embargo, mientras que el 28 de diciembre 2021, las posiciones más lejanas (de la séptima a la décima) muestran predicciones más altas que la realidad, el resto de los días generan pronósticos muy precisos para estas mismas posiciones. Por último, las predicciones de la undécima posición suelen estar por debajo de la realidad. Con el fin de simplificar las conclusiones, se presenta el siguiente gráfico que muestra el error medio para cada posición:

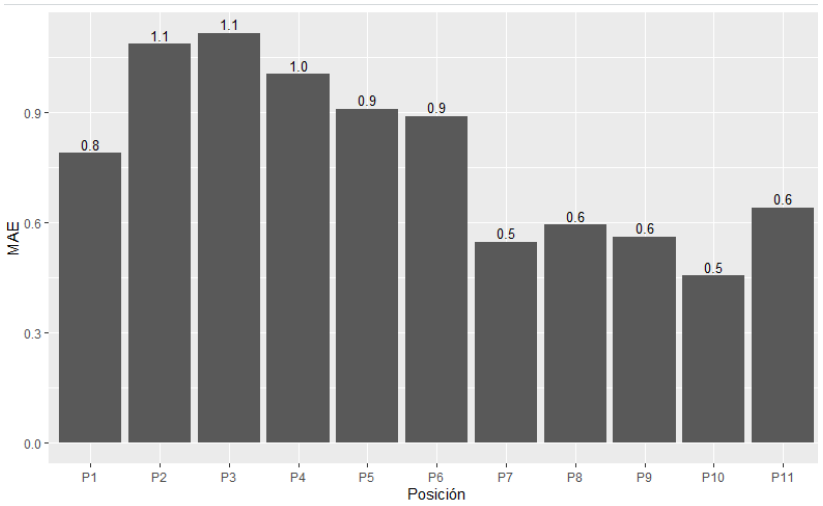
Figura 41: Objetivo I – Error medio (ME) de cada posición



En el gráfico superior se ve claramente como las predicciones de las primeras posiciones suelen ser estar por debajo de la media. Por ejemplo, las predicciones de la posición dos están de media -0,8$ por debajo del precio real. Sin embargo, para las posiciones 8 y 9 las predicciones suelen estar 0,3$ por encima de la realidad.

Con la explicación anterior se han mostrado en qué dirección se desvían las predicciones respecto a la realidad. Sin embargo, el siguiente gráfico muestra de una manera más clara el tamaño del error de cada vencimiento.

Figura 42: Objetivo I – Error medio absoluto (MAE) de cada posición



Como se puede observar, las predicciones tienen error más elevado en las primeras posiciones, que va decayendo a medida que aumentan los vencimientos. Esto podría ser porque, como se vio en la sección de análisis exploratorio, las primeras posiciones son las que lideran al resto y son más volátiles, por lo que debería ser más normal que fueran más difíciles de predecir.

En conclusión, los modelos ARIMA son capaces de predecir las curvas del Brent, cumpliendo así con el primer objetivo. Como se acaba de explicar, aun así, la precisión de dichas predicciones varía según la posición. De todas formar, el error en las predicciones no es demasiado elevado, por lo que el modelo es adecuado.

4.2. Objetivo II: Generar escenarios para las curvas en base a errores del modelo

Para la consecución del segundo objetivo, se generarán distintos escenarios futuros usando simulaciones de los errores históricos de los modelos ARIMA ajustados en el desarrollo del primer objetivo. Es importante destacar que los modelos ARIMA que se usarán como base serán los que no se aplican a los datos de prueba, sino los que únicamente se aplican al conjunto de entrenamiento.

Hasta este punto, el modelo ARIMA generado proporciona una única trayectoria futura: la media dentro del rango de intervalos de confianza. Por lo tanto, es muy relevante simular escenarios futuros que permitan explorar otros posibles caminos que la serie

temporal podría seguir en el futuro para entender en mejor la incertidumbre de las predicciones, y conocer el rango de posibles futuros para preparase para cada uno de ellos.

Como se ha explicado anteriormente y según se demuestra en la figura 30, los datos de prueba reales están dentro del rango de predicciones basados en distintos intervalos de confianza de los modelos ARIMA creados en el Objetivo I. De esta manera, se justifica que tiene sentido generar escenarios simulados para observar cuál se puede acercar más a la realidad.

Para ello, se usa la función "simulate" de R, que permite generar distintas trayectorias futuras de una serie temporal mediante simulaciones con un componente aleatorio: el término del error (ϵ_t). De esta manera, cada escenario se basa en el modelo ARIMA previamente ajustado y en los errores simulados aleatoriamente según la distribución histórica del modelo. Por lo tanto, se generan valores futuros de la serie utilizando los componentes autorregresivos, de diferenciación y de media móvil, además de introducir nuevos errores aleatorios (ϵt) basados en la distribución histórica de los errores del modelo. En el caso de este trabajo, se generarán 10 escenarios distintos.

Figura 43: Objetivo II - Escenarios de la primera componente principal

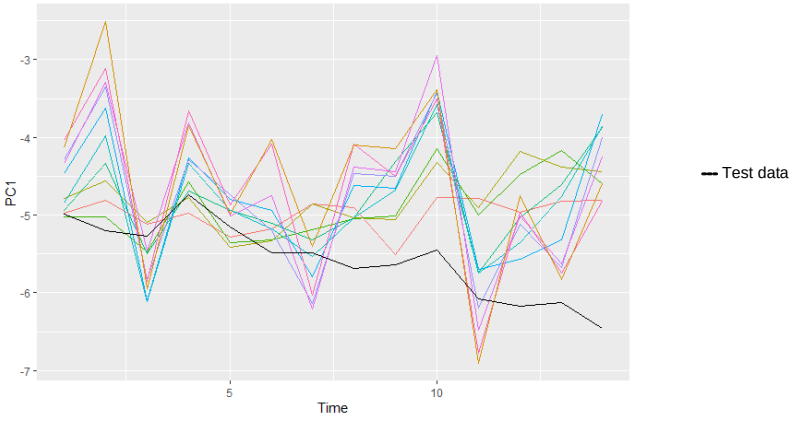
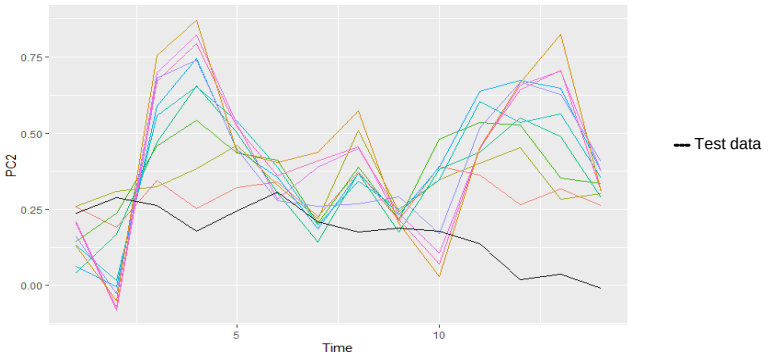


Figura 44: Objetivo II - Escenarios de la segunda componente principal



Como se puede observar, los escenarios de ambas componentes principales muestran valores mucho más volátiles que los datos reales. Además, suelen situarse por encima de la realidad. Por otro lado, el hecho de que los coeficientes del ARIMA sean o muy bajos o nulos, implica que el patrón que siguen los escenarios está mayoritariamente basado en el error simulado aleatoriamente en base a su distribución histórica.

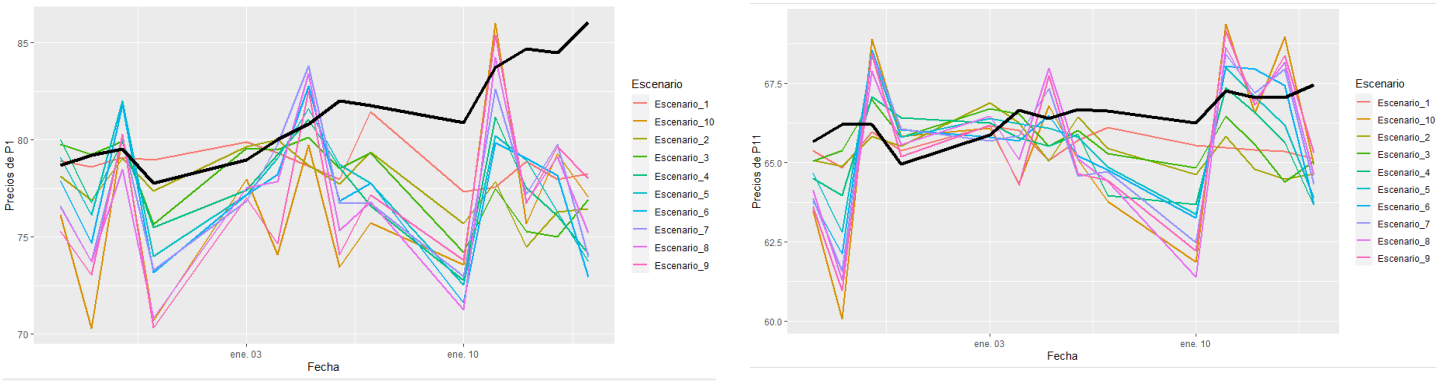
Una vez se han simulado los posibles escenarios de cada componente principal en base a los errores, se puede continuar con la reconstrucción de las curvas. A modo de repaso, la reconstrucción se realizará en base a las predicciones generadas de la siguiente manera:

- Primera componente principal: simulación de 10 escenarios de predicciones según ARIMA(1,1,0) únicamente aplicado a los datos de entrenamiento, basados en errores históricos.
- Segunda componente principal: simulación de 10 escenarios de predicciones ARIMA(0,1,0) únicamente aplicado a los datos de entrenamiento, basados en errores históricos.

Figura 45: Objetivo II - Serie temporal reconstruida

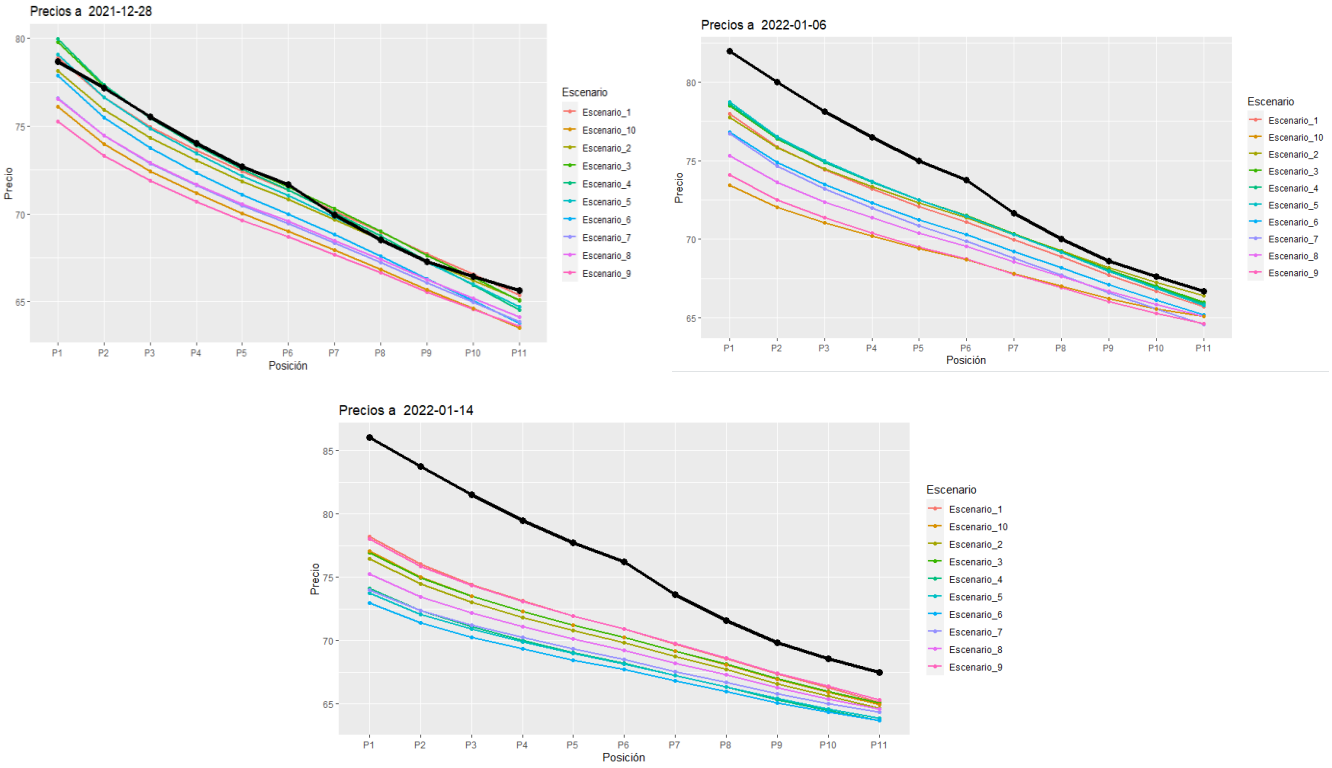
Como se ve en la figura 45, en las dos últimas semanas, que corresponden a los datos de prueba, se han creado 10 escenarios distintos para cada posición. Si se compara con el gráfico original de esta misma serie temporal (figura 4), se observa como para cada fecha, la varianza de los precios es mayor, ya que lo que se está intentando hacer en este objetivo es la incertidumbre de las predicciones.

Figura 46: Objetivo II - Serie temporal original vs. reconstruida – posiciones 1 y 11



Una vez se observa la serie temporal de cada posición desde cerca, uno se da cuenta de que los escenarios no simulan correctamente el precio real (línea negra). No solo suelen generar precios menores, sino que además, los precios son mucho más volátiles que los originales. Esto quiere decir que las simulaciones no se ajustan bien a la invertidumbre.

Figura 47: Objetivo II – Curvas de originales vs. reconstruidas



Se pueden sacar varias conclusiones de las visualizaciones de las curvas de futuros para diferentes días. Por un lado, es importante destacar que se suelen generar escenarios por

debajo de los precios reales. Además, a medida que se aumenta el horizonte temporal de proyección, los errores cada vez son más significativos y la diferencia con los precios reales se acentúa.

Es muy importante mencionar que la función *simulate* de R se ha aplicado sobre el modelo del primer objetivo basado únicamente en base a los datos de entrenamiento. De esta manera, cada proyección de cada escenario se basa en un valor predicho del día anterior, por lo que los errores se van acumulando. Por lo tanto, las conclusiones de este objetivo no son comparables con las del primero ya que en el primero se usaron predicciones basadas en los datos de prueba para no predecir sobre valores predichos, evitando así la acumulación de errores.

En conclusión, los resultados de este objetivo demuestran que, aunque se han podido generar escenarios para las curvas en base a errores del modelo, éstos no se ajustan a la incertidumbre. Es decir, se concluye que el uso de técnicas de simulación de errores históricos de modelos ARIMA, no es adecuado para generar escenarios de las curvas de futuros, por lo que no contribuyen a entender la incertidumbre de una predicción.

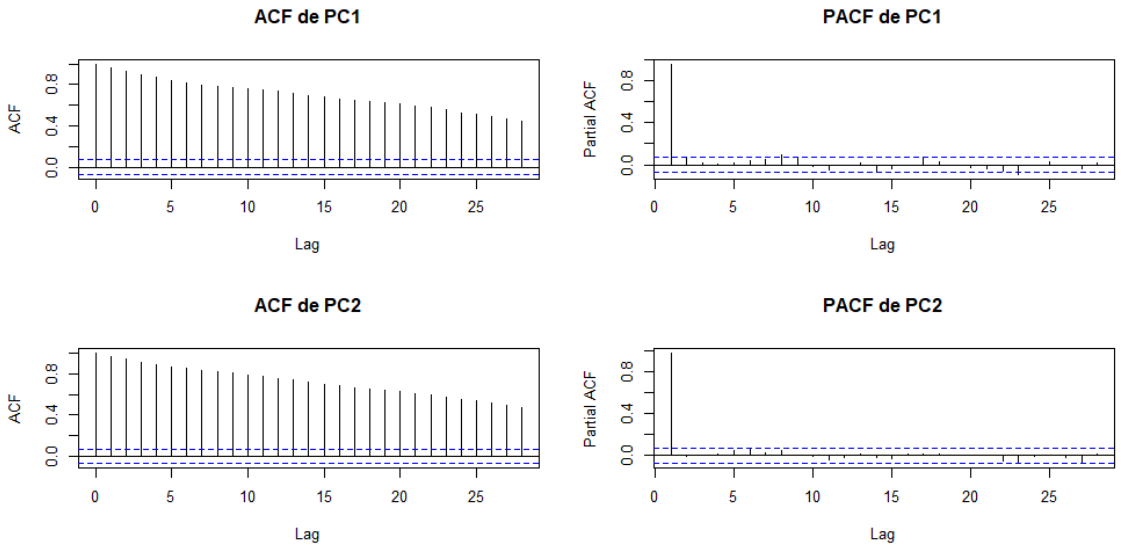
4.3. Objetivo III: Generar escenarios para las curvas usando el precio del Brent como variable explicativa

Hasta este punto se han explicado técnicas de predicción en base a series temporales históricas, pero no se han tenido en cuenta otras variables que también pueden ser relevantes. En ese objetivo, se aplica al modelo ARIMA una variable explicativa, en este caso el precio *spot* del Brent, que corresponde a la primera posición del dataset y que se refiere al precio de mercado actual del Brent. Este tipo de modelos se denominan modelos de regresión dinámica con ARIMA, y dependen de dos factores: la variable explicativa y la variable temporal representada por el modelo ARIMA.

De esta manera, no sólo se tienen en cuenta las series temporales para realizar las predicciones, sino que, además, se contabilizan factores explicativos reflejados en el precio *spot* que puedan afectar a las curvas en horizontes posteriores (como eventos geopolíticos, costes de almacenamiento, cambios en políticas energéticas, etc).

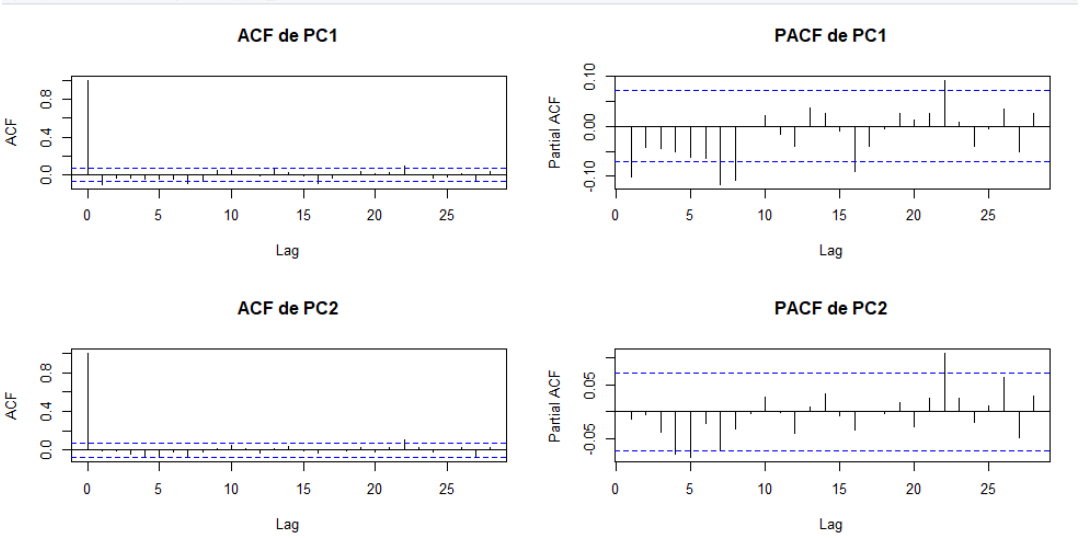
Como se ha explicado en la sección de metodología, el primer paso para determinar el orden del modelo consiste en determinar si se deben diferenciar las variables o no.

Figura 48: Objetivo III – ACF y PACF de las dos primeras componentes principales



Como se puede observar abajo, el ACF de ambas componentes disminuye gradualmente, indicando que las series no son estacionarias y que es necesario diferenciarlas. El siguiente paso es determinar el orden de los parámetros autorregresivos y de media móvil, visualizando el ACF y el PACF del modelo diferenciado.

Figura 49: Objetivo III – ACF y PACF de las dos primeras componentes principales diferenciadas



En cuanto a la primera componente principal, el ACF demuestra que la serie no tiene ningún componente de media móvil, mientras que el PACF tiene un rezago significativo en el primer horizonte temporal. Por lo tanto, se aplicará un modelo ARIMA(1,1,0) para variable temporal de la primera componente principal. Por otro lado, la segunda

componente parece ser ruido blanco, por lo que sus valores temporales tienen un carácter aleatorio y su variable temporal será un ARIMA(0,1,0). Sin embargo, es importante mencionar que, según se vio en el análisis PCA, esta componente explica únicamente un 1% de la varianza (mientras que la primera componente principal explica el 98,8%), por lo que el hecho de que no se pueda modelar la parte temporal no será especialmente significativo para las proyecciones.

Una vez ajustado el modelo, que tiene en cuenta la variable temporal y la explicativa, es necesario llevar a cabo varias comprobaciones para asegurar que se ha estimado correctamente.

En primer lugar, todos los coeficientes son significativos, tanto en el modelo de la primera como en el de la segunda componente principal. Es decir, como el p-valor es bajo, hay pocas posibilidades de que el coeficiente en realidad sea igual a 0.

Figura 50: Objetivo III - Test Z de coeficientes significativos del modelo ARIMA(1,1,0) para la primera componente principal

```
z test of coefficients:

                      Estimate Std. Error  z value  Pr(>|z|)
ar1                  -0.1052065  0.0370176  -2.8421  0.004482 **
curvas_train_data$P1 -0.2128741  0.0036915 -57.6656 < 2.2e-16 ***
```

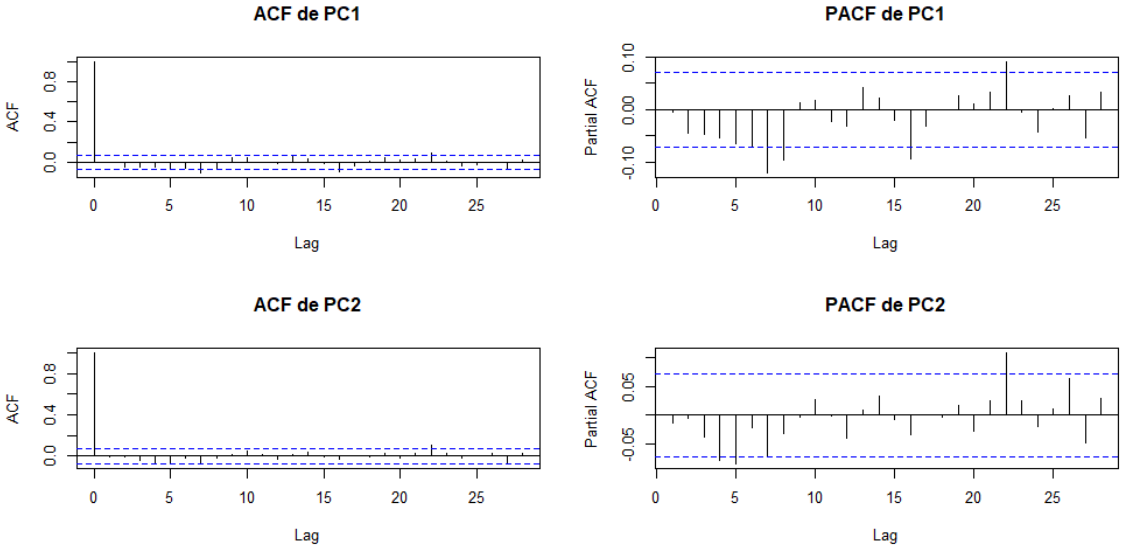
Figura 51: Objetivo III - Test Z de coeficientes significativos del modelo ARIMA(0,1,0) para la segunda componente principal

```
z test of coefficients:

                      Estimate Std. Error z value  Pr(>|z|)
curvas_train_data$P1 -0.0107698  0.0019487 -5.5265 3.266e-08 ***
```

Además, según muestran las figuras superiores, la variable explicativa (“curvas_train_data$P1) también es significativa, por lo que se demuestra la utilidad aplicar un modelo de regresión dinámica para predecir curvas de futuros.

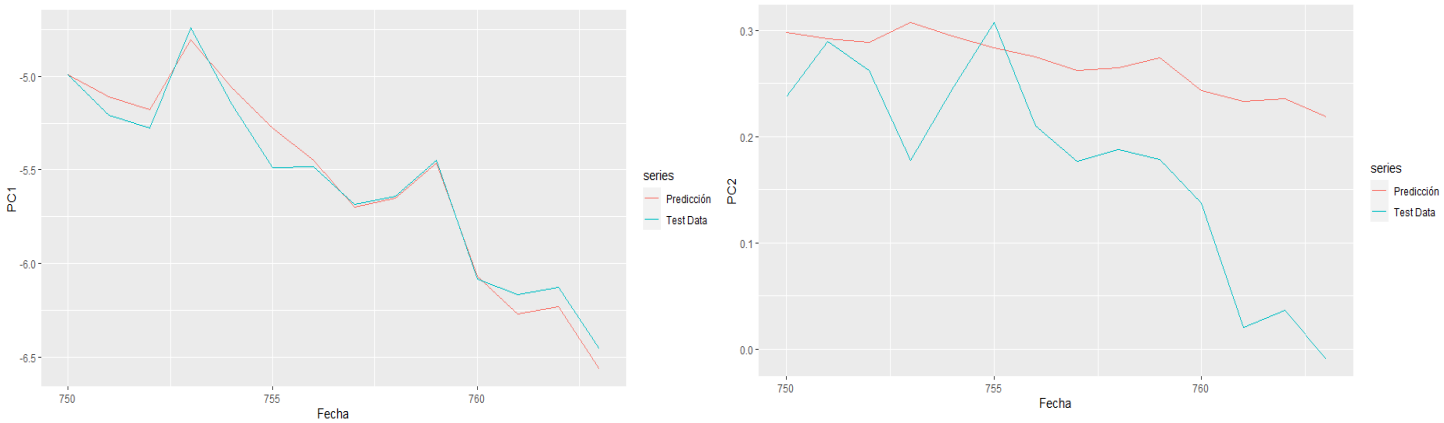
El siguiente paso en el proceso de verificaciones consiste en visualizar el ACF y el PACF de los residuales para comprobar que son ruido blanco y que, por lo tanto, no se pueden modelar. Como se muestra en la figura 52, los residuales son efectivamente ruido blanco en ambas componentes. Además, aunque no se ha incluido, la prueba de Ljung-Box corrobora el hecho de que los residuales son ruido blanco y que, en consecuencia, el modelo está bien ajustado.

Figura 52: Objetivo III - ACF y PACF de los residuales de los modelos



Llegados a este punto, se ha comprobado que los coeficientes son significativos y que el modelo está bien ajustado, por lo que se puede proceder a realizar las predicciones empleando los modelos de regresión dinámica de ambas componentes principales.

Figura 53: Objetivo III: Predicciones de las componentes principales



Como se puede observar, las predicciones de la primera componente principal se ajustan muy bien a la realidad. Además, aunque la segunda componente principal suele estimar valores demasiado altos que no se ajustan demasiado a la realidad, sí capta la tendencia de decrecimiento.

Tabla 5: Objetivo III: Comparación de errores de predicción (MSE)

Errores de predicción (MSE)	*ARIMA forecast()*	*ARIMA fitted()*	*ARIMA con variable explicativa*
Primera componente principal	0,715	0,085	0,008
Segunda componente principal	0,025	0,003	0,014

Es a la hora de comparar entre los distintos modelos entrenado cuando realmente se demuestra la efectividad de estos modelos. Como se puede observar, el modelo de ARIMA con la variable explicativa se ajusta notablemente mejor que el resto a los valores reales de la primera componente principal. Incluso es más preciso que el segundo modelo de ARIMA basado en los datos de prueba. Si bien el error de las actuales predicciones para la segunda componente principal no es el más bajo, es importante destacar que está variable es más bien aleatoria y no es realmente significativa a la hora de reconstruir las curvas.

Como recordatorio, la reconstrucción de las curvas se llevará a cabo en base a las siguientes predicciones generadas por los siguientes modelos:

- Primera componente principal: ARIMA(1,1,0) usando el precio del Brent como variable explicativa.
- Segunda componente principal: ARIMA(0,1,0) usando el precio del Brent como variable explicativa.

Figura 55: Objetivo III - Serie temporal original vs. reconstruida

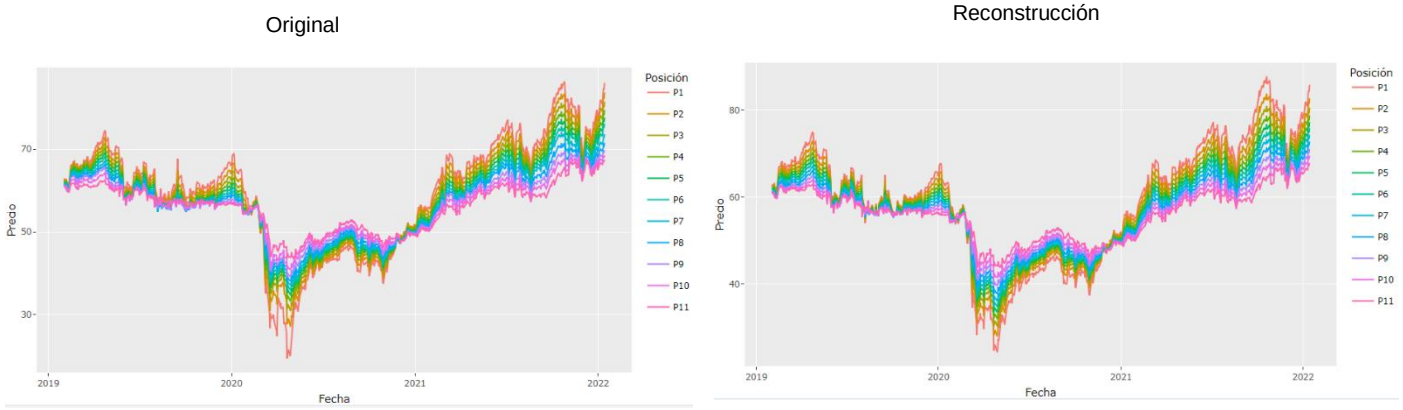
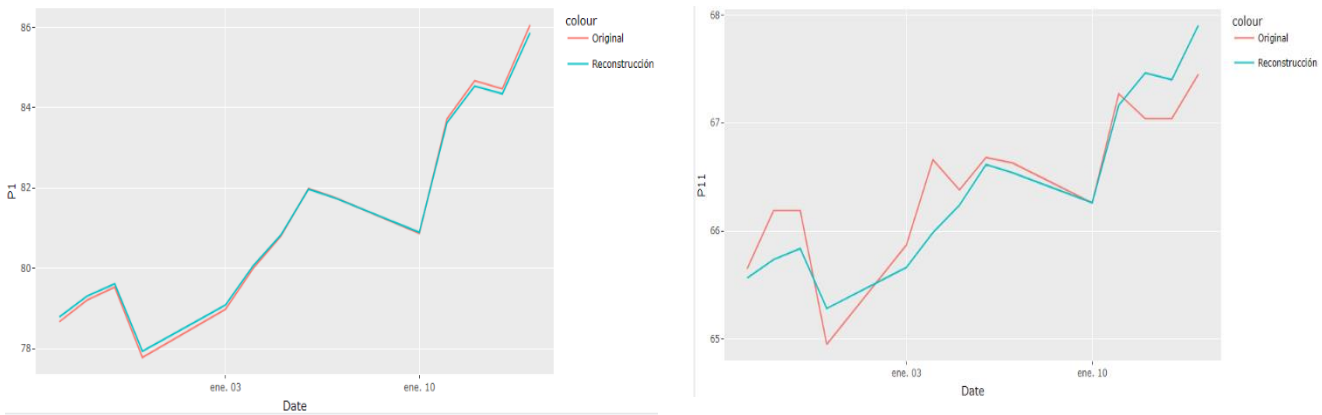
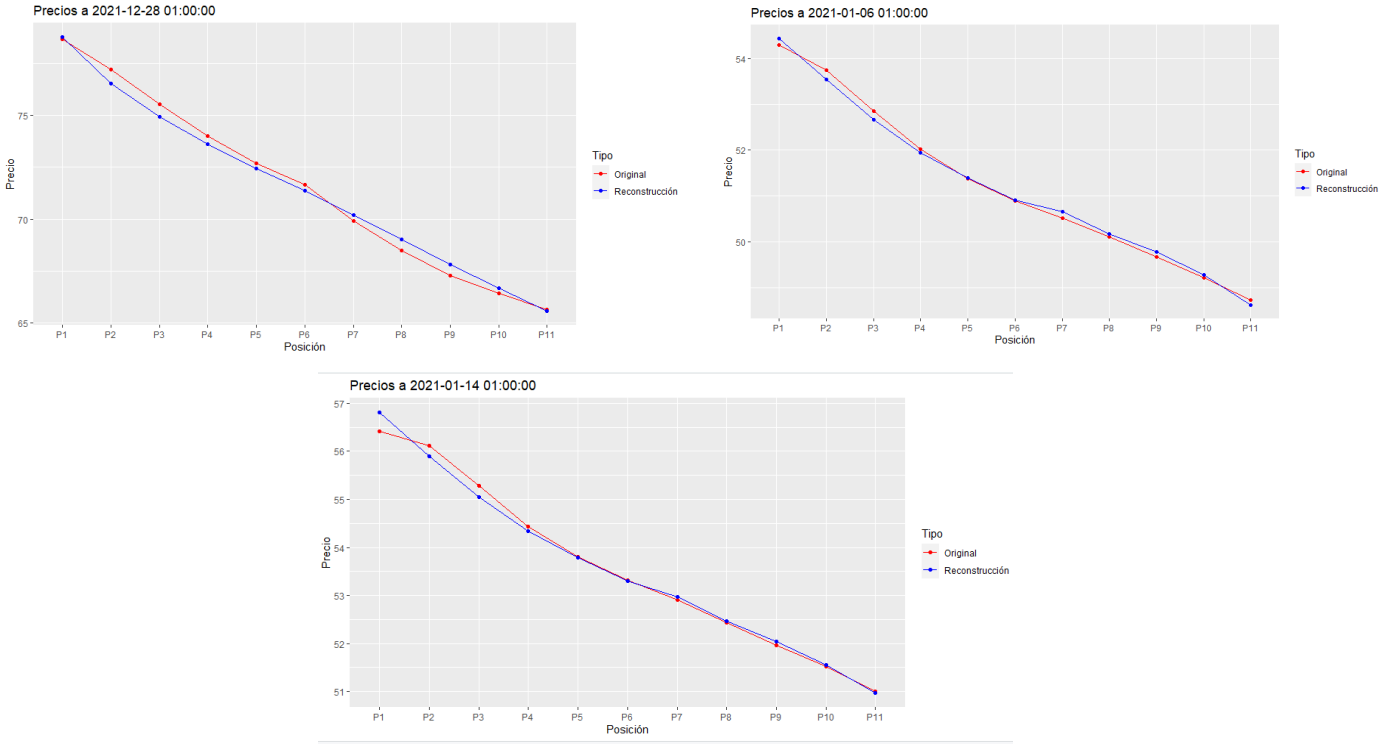


Figura 56: Objetivo III - Serie temporal original vs. reconstruida – posiciones 1 y 11



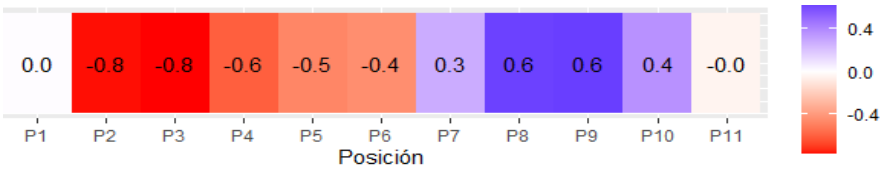
Como se puede observar, las predicciones se ajustan muy bien a la realidad. Si bien hay una ligera diferencia entre los valores predichos y originales en la serie temporal de la posición 11, la posición 1 muestra una predicción muy precisa. Esto se debe al hecho de que es la misma variable que se ha usado como variable explicativa en el ajuste del modelo, por lo que tiene sentido que se capten las proyecciones tan bien.

Figura 57: Objetivo III - Curvas de fututos originales vs. reconstruidas



En cuanto a las predicciones de las curvas de futuros, se ajustan muy bien a la realidad, si bien es cierto que el primer día predicho, por ejemplo, los pronósticos de las primeras posiciones son más bajos que los precios reales, mientras que en las últimas posiciones ocurre lo contrario.

Figura 58: Objetivo III - Error medio (ME) de cada posición

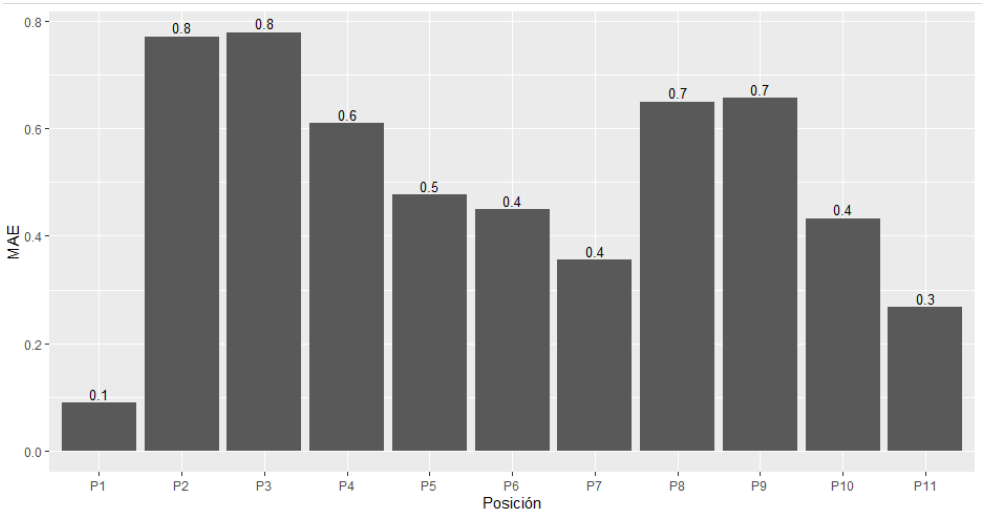


La figura 58 muestra el hecho de que se suelen predecir precios más bajos que los reales en las primeras posiciones, mientras que las últimas posiciones suelen tener predicciones

por encima de la realidad. En los vencimientos del medio y de los extremos, las predicciones suelen ser más precisas.

Hasta este punto se han analizado en qué dirección se debían las predicciones respecto de los precios reales. Sin embargo, el error medio absoluto permite identificar aquellas posiciones que mejor se predicen, independientemente de la dirección que tome el error de las predicciones (por encima o por debajo de la realidad).

Figura 59: Objetivo III - Error medio absoluto (MAE) de cada posición



Como se puede observar, las predicciones de la primera posición son muy buenas. Sin embargo, a partir de la segunda posición el tamaño del error aumenta significativamente. Si bien es cierto que se va reduciendo gradualmente hasta la séptima posición, el error vuelve a subir en la posición 8. Una vez más, el error vuelve a bajar en posiciones posteriores. En general, cuanto más lejanos sean los vencimientos de las posiciones, mejor serán las predicciones.

En conclusión, el modelo de regresión dinámica con ARIMA ha demostrado generar buenos resultados, y se demuestra, a pesar de los errores, que podría servir para predecir las curvas de futuros del Brent para contribuir a la determinación de mejores estrategias de trading.

5. Conclusiones

Para concluir este estudio, se revisitarán las principales cuestiones tratadas para dar una respuesta concisa a cada uno de los tres objetivos analizados. De esta manera, se podrán comparar los resultados de las distintas técnicas de Machine Learning empleadas, para determinar cuál de ellas es más útil a la hora de predecir y generar escenarios de las curvas de futuros del Brent. Por último, se propondrán futuras líneas de investigación para fomentar el desarrollo de este tipo de técnicas de manera que se puedan utilizar para realizar mejores estrategias de trading con futuros del Brent.

Tras una aproximación al mercado de derivados, con un filtro específico en el mercado de futuros y un enfoque en los contratos de futuros del Brent, se determinó su particular interés como objeto de estudio. Los futuros del Brent sirven como referencia de precios del petróleo. Su importancia estratégica en la economía global, junto con su creciente volumen de negociación y elevada liquidez, los convierte en un foco de atención para cada vez más inversores. A su vez, este elevado interés ha venido acompañado por un aumento en la investigación de técnicas empleadas de Machine Learning para poder predecir sus precios. Sin embargo, los resultados varían de estudio en estudio, determinando así la importancia de realizar un estudio del mercado de los futuros de Brent para predecir y generar escenarios de sus curvas.

Para ello, en primer lugar, se realizó un análisis exploratorio del conjunto de datos de curvas de futuros, en el que se llegaron a dos conclusiones. En primer lugar, las primeras posiciones son las que lideran los precios del Brent, por lo que también son las más volátiles. En segundo lugar, es relevante señalar que, antes y después de la pandemia, la curva de futuros se encontraba en *backwardation*. Sin embargo, en plena crisis de COVID-19, la curva pasó a estar en contango. Durante este período, el petróleo llegó a cotizar en negativo, y los operadores del mercado anticipaban una recuperación de los precios, lo que llevó a la acumulación de inventarios para aprovechar las diferencias entre contratos de distintos vencimientos, provocando así la situación de contango.

Posteriormente, se realizó una revisión teórica de las técnicas de predicción de curvas de futuros del Brent que se implementarían más adelante en el presente estudio: el análisis de componentes principales, la predicción de precios empleando modelos ARIMA, la generación de escenarios de precios en base a errores históricos, y la predicción de precios integrando un modelo ARIMA a una variable explicativa (modelo de regresión dinámica).

Por un lado, el PCA determinó que las dos primeras componentes principales eran capaces de explicar un 99,8% de la varianza explicada. De esta manera, se seleccionaron estas dos variables como base para generar las predicciones y simulaciones de los modelos que se implementaron posteriormente.

Para predecir las curvas de futuros, se utilizó un modelo ARIMA(1,1,0) aplicado a los datos de prueba (objetivo I), y un modelo ARIMA(1,1,0) usando el precio actual del Brent como variable explicativa (modelo de regresión dinámica con ARIMA) (objetivo III). Si bien las proyecciones de ambos modelos se ajustaban a los precios reales, concluyendo así la aptitud de ambos modelos, también existían errores en las predicciones. A continuación, se compararán los errores de ambos modelos para determinar en qué consisten los errores cometidos, y qué modelo es el más preciso.

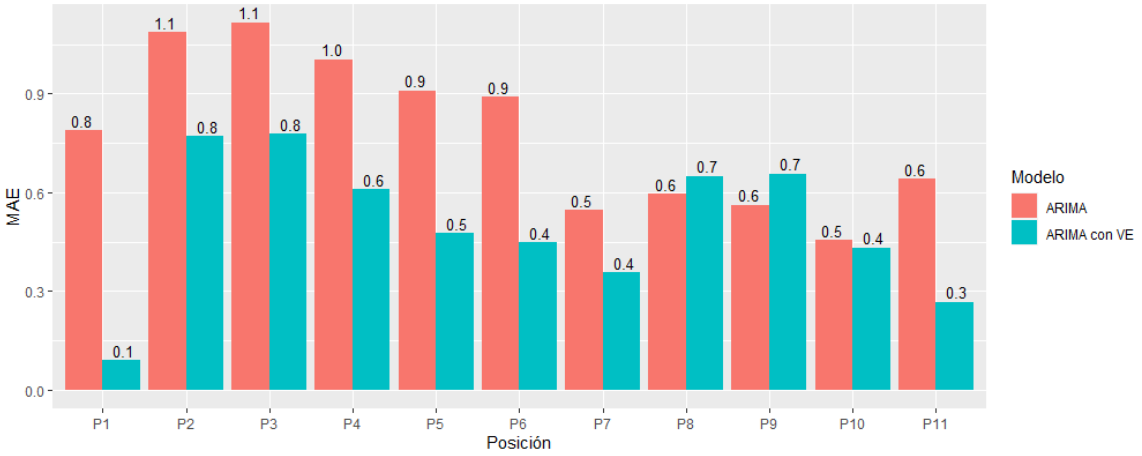
El siguiente gráfico detalla el error medio de las predicciones para cada posición y muestra que las direcciones de los errores de ambos modelos suelen ser las mismas: las predicciones de las primeras posiciones (de la segunda a la sexta) suelen ser menores que los precios reales, mientras que las predicciones de las últimas posiciones, a excepción de la undécima (de la séptima a la décima), suelen ser más altas que los valores reales.

Figura 60: Comparación entre modelos: Error medio (ME) de cada posición

Una vez se han analizado en qué consisten los errores usando como parámetro la dirección que suelen tomar respecto a los valores reales, conviene estudiar de qué tamaño es el error para poder concluir qué modelo predice mejor las curvas de futuros. Para ello, se emplea el error medio absoluto (MAE), representado en la figura 61. La primera conclusión que se puede derivar es que, en general, los errores de ambos modelos decrecen conforme se predicen las últimas posiciones. Esto probablemente se deba al hecho de que las primeras

posiciones son las que lideran el precio de los futuros y son las más volátiles, lo cuál puede dificultar su predicción. La siguiente conclusión es que los errores del modelo de regresión dinámica (ARIMA(1,1,0) con una variable explicativa) suelen ser menores que el modelo ARIMA(1,1,0) simple. Esto se cumple para todas las posiciones excepto para la octava y la novena, en las que los errores del modelo ARIMA simple son ligeramente menores.

Figura 61: Comparación entre modelos: Error medio absoluto (MAE) de cada posición



Por lo tanto, se concluye que el modelo ARIMA(1,1,0) usando el precio actual del Brent como variable explicativa suele generar predicciones más precisas de las curvas de futuros del Brent, por lo que permitirá determinar mejores estrategias de trading.

Por otro lado, la generación de escenarios de precios simulando los errores históricos de un modelo ARIMA(1,1,0) no resultó ser especialmente útil. Los escenarios generados predecían precios inferiores a los reales de manera consistente, por lo que no contribuyeron a entender la incertidumbre de una predicción. Además, dichas diferencias se acentuaban conforme el horizonte temporal se alejaba. Por lo tanto, se sugiere que el uso de técnicas de simulación de errores históricos de modelos ARIMA puede no ser adecuado para generar escenarios de las curvas de futuros del Brent.

Este trabajo ha pretendido utilizar distintas técnicas de Machine Learning, para predecir y generar escenarios de la curva de futuros del Brent. Dada la importancia global y creciente de este tipo de contratos, estos modelos podrían ser empleados por numerosos inversores para determinar mejores estrategias de trading en base a los precios predichos. Sin embargo, el hecho de que las técnicas de modelaje estén en constante evolución y existan una amplia variedad de modelos para predecir precios, deja la puerta abierta para futuras líneas de líneas de investigación. En primer lugar, dado que la generación de

escenarios en base a errores de modelos ARIMA no ha proporcionado resultados demasiado útiles, se recomienda probar otro tipo de modelos para intentar capturar de alguna manera la incertidumbre de una predicción. Por otro lado, se sugiere profundizar más en el modelo de regresión dinámica con ARIMA; por ejemplo, se podría realizar un estudio de generación de escenarios usando los errores históricos de este modelo, en vez de los errores del ARIMA simple.

6. Declaración de Uso de Herramientas de Inteligencia Artificial Generativa en Trabajos Fin de Grado

ADVERTENCIA: Desde la Universidad consideramos que ChatGPT u otras herramientas similares son herramientas muy útiles en la vida académica, aunque su uso queda siempre bajo la responsabilidad del alumno, puesto que las respuestas que proporciona pueden no ser veraces. En este sentido, NO está permitido su uso en la elaboración del Trabajo fin de Grado para generar código porque estas herramientas no son fiables en esa tarea. Aunque el código funcione, no hay garantías de que metodológicamente sea correcto, y es altamente probable que no lo sea.

Por la presente, yo, Isabel Lantero Hernández, estudiante de Doble Grado en Administración y Dirección de Empresas, y Análisis de Negocios (E-2 + Analytics) de la Universidad Pontificia Comillas al presentar mi Trabajo Fin de Grado titulado "Predicción y generación de escenarios de curvas de futuros del Brent usando técnicas de Machine Learning", declaro que he utilizado la herramienta de Inteligencia Artificial Generativa ChatGPT u otras similares de IAG de código sólo en el contexto de las actividades descritas a continuación [el alumno debe mantener solo aquellas en las que se ha usado ChatGPT o similares y borrar el resto. Si no se ha usado ninguna, borrar todas y escribir “no he usado ninguna”]:

1. **Referencias:** Usado conjuntamente con otras herramientas, como Science, para identificar referencias preliminares que luego he contrastado y validado.
2. **Metodólogo:** Para descubrir métodos aplicables a problemas específicos de investigación.
3. **Interpretador de código:** Para realizar análisis de datos preliminares.
4. **Corrector de estilo literario y de lenguaje:** Para mejorar la calidad lingüística y estilística del texto.
5. **Sintetizador y divulgador de libros complicados:** Para resumir y comprender literatura compleja.
6. **Revisor:** Para recibir sugerencias sobre cómo mejorar y perfeccionar el trabajo con diferentes niveles de exigencia.
7. **Traductor:** Para traducir textos de un lenguaje a otro.

Afirmo que toda la información y contenido presentados en este trabajo son producto de mi investigación y esfuerzo individual, excepto donde se ha indicado lo contrario y se han

dado los créditos correspondientes (he incluido las referencias adecuadas en el TFG y he explicitado para que se ha usado ChatGPT u otras herramientas similares). Soy consciente de las implicaciones académicas y éticas de presentar un trabajo no original y acepto las consecuencias de cualquier violación a esta declaración.

Fecha: 21 de junio de 2024

Firma: 

7. Bibliografía

Gonzálvez, Á. (2023). Análisis Fractal, ARFIMA y PCA: aproximación a los mercados de futuros del petróleo.

Bañón González, G. (2018). Mercado de derivados financieros: evolución, análisis y perspectivas de futuro.

Castellanos, E. (2017, septiembre). *Breve historia de los mercados de derivados*. MEFF. https://www.meff.es/docs/newsletter/2017/NEWSLETTER_MEFF-49_Septiembre.pdf

CNMV. (2024). *Productos derivados.* https://www.cnmv.es/portal/inversor/derivados.aspx?lang=es

De Lara, A. (2005). Productos financieros: instrumentos, valuación y cobertura de riesgo. https://books.google.es/books/about/PRODUCTOS_DERIVADOS_FINANCIEROS_INSTRUME.html?hl=es&id=NdmPbFliI9sC&redir_esc=y

Díaz-Pinzón, J. E. (2023). Fluctuación del precio del petróleo Brent debido a la guerra entre Rusia y Ucrania. *Revista Economía y Política*, (37), 104-116.

Elvira, O., & Larraga, P. (2008). *Mercado de productos derivados: futuros, forwards, opciones y productos estructurados* (Vol. 9). Profit Editorial.

Feelcapital. (2017). *Principales diferencias entre Mercados Organizados y Mercados OTC*. Feelcapital. https://blog.feelcapital.com/mercados-organizados-mercados-otc/

Figueroa, V. M. (2008). Los instrumentos financieros derivados: concepto, operación y algunas estrategias de negociación. *Revista de Ciencias Económicas*, *26*(2).

Futures Industry Association. (2023). ETD Volume. https://www.fia.org/fia/articles/etd-volume-december-2022

Gasper, L., & Mbwambo, H. (2023). Forecasting crude oil prices by using ARIMA model: evidence from Tanzania.

Giraldo-Prieto, C. A., González Uribe, G. J., Vesga Bermejo, C., & Ferreira Herrera, D. C. (2017). Coberturas financieras con derivados y su incidencia en el valor de mercado en empresas colombianas que cotizan en Bolsa. *Contaduría y administración*, 62(SPE5), 1553-1571.

Gray, S. T., & Place, J. (2003). *Derivados financieros*. Centro de Estudios Monetarios Latinoamericanos.

Harris, R., & Sollis, R. (2003). *Applied time series modelling and forecasting*. John Wiley & Sons.

Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: principles and practice*. OTexts.

Interncontintal Exchange (2024). *Brent crude futures.* https://www.ice.com/products/219/Brent-Crude-Futures

Jacobsson, M. (2015). Forecasting commodity futures using Principal Component Analysis and Copula.

Jolliffe, I. T. (2002). *Principal component analysis for special types of data* (pp. 338-372). Springer New York.

Gamero, J. R. (2018). *El petróleo: su importancia geopolítica*. https://www.linkedin.com/pulse/el-petr%C3%B3leo-su-importancia-geopol%C3%ADtica-jos%C3%A9-rafael-gamero-lanz/?originalSubdomain=es

Kummer, S., & Pauletto, C. (2012). The history of derivatives: A few milestones. In *EFTA Seminar on Regulation of Derivatives Markets* (pp. 431-466).

Lamothe, P., (1993). *Opciones financieras*. 1ª ed. Madrid: McGraw Hill.

Magnier Villamil, G. (2014). Mercados over-the-counter de productos derivados.

Mensah, E. K. (2015). Box-jenkins modelling and forecasting of brent crude oil price.

Moreno-Torres Gálvez. (2020). *Del super-contango a los precios negativos del petróleo*. Asociación de Ingenieros Industriales del Estado. https://ingenierosindustrialesdelestado.es/2020/05/05/del-super-contango-a-los-precios-negativos-del-petroleo/

Muhamad. (2019). *Análisis WTI de Curva de Futuros con PCA (Parte 1)*. https://support.numxl.com/hc/es/articles/360032850551-An%C3%A1lisis-WTI-de-Curva-de-Futuros-con-PCA-Parte-1

Rahmayanti, I. A., Andreas, C., & Ulyah, S. M. (2021, February). Does US-China trade war affect the Brent crude oil price? An ARIMAX forecasting approach. In *AIP Conference Proceedings* (Vol. 2329, No. 1). AIP Publishing.

Reuters. (2022). *Desplome del petróleo Brent crea el contango más acentuado en 11 años*. Reuters.https://www.reuters.com/article/idUSKBN21D249/

Sánchez Navarro, A. (2015). Aproximación al mercado de futuros sobre materias primas. Desarrollo de un caso práctico de cobertura en el mercado de futuros del latón.

Santander. (2024). *¿Qué son los derivados financieros y qué tipos existen?* Banco Santander. https://www.bancosantander.es/glosario/derivados-financieros

Shah ,J , Kiruthiga, G. (2020). Crude Oil Price Forecasting Using ARIMA model. *International Journal of Advanced Scientific Inovation*, 1(1), 1–11.

Stellwagen, E., & Tashman, L. (2013). ARIMA: The Models of Box and Jenkins. *Foresight: The International Journal of Applied Forecasting*, (30).

Zhao, C. L., & Wang, B. (2014). Forecasting crude oil price with an autoregressive integrated moving average (ARIMA) model. *Fuzzy information & engineering and operations research & management*, 275-286.

8. Anexo I: Código

```
setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
getwd()

########### LIBRERIAS ############
library(readxl)
library(ggplot2)
library(plotly)
library(ggfortify)
library(corrplot)
library(tidyr)
library(factoextra)
library(gganimate)
library(gifski)
library(FactoMineR)
library(dplyr)
library(fable)
library(lmtest)
library(forecast)

############# IMPORTAR DATASET ###########
curvas <- read_excel("Curvas.xlsx",2)

############# ANALISIS EXPLORATORIO #################
anyNA(curvas)
summary(curvas)
str(curvas$Date)
#evolución serie
grafico_curvas <- curvas %>% pivot_longer(cols = -c(Date), names_to = "posicion", values_to =
"precio")
ordenpos <- c("P1", "P2", "P3", "P4", "P5", "P6", "P7", "P8", "P9", "P10", "P11")
grafico_curvas$posicion <- factor(grafico_curvas$posicion, levels = ordenpos)
evol <- ggplot(grafico_curvas, aes(x = Date, y = precio, color = posicion, group = posicion)) +
  geom_line() +
  labs(x = "Fecha", y = "Precio", color = "Posición")
ggplotly(evol)

#Ditribución por Posición
boxplot(curvas[,2:12])

#Curva de un dia, cambiar fecha según la que se qiuiera mostrar
fecha <- "2019-02-01 01:00:00"
datosdia <- curvas %>% filter(Date == fecha) %>%
  pivot_longer(cols = starts_with("P"), names_to = "Posicion", values_to = "Precio")
ordenpos <- c("P1", "P2", "P3", "P4", "P5", "P6", "P7", "P8", "P9", "P10", "P11")
datosdia$Posicion <- factor(datosdia$Posicion, levels = ordenpos)
ggplot(data = datosdia, mapping = aes(x = Posicion, y = Precio)) +
  geom_line(group = 1)+
  geom_point() +
  labs(title = paste("Precios a", fecha), x = "Posición", y = "Precio")


##############ANALISIS PCA##############
```

```
#Determinando necesidad y detalles del PCA
pairs(curvas)
#--> correlacion
cor(curvas[2:12]) #como hay correlacion --> si hay posibilidad de PCA para evitar multicolinealidad.
corrplot(cor(curvas[,2:12]), method = "color")
var(curvas[,2:12])  # matriz de covarianza
#--> varianza
sapply(curvas[,2:12], var)  #varianza de cada una de las variables. Varianza disminuye a medada que vencimiento de Posición se aleja. Haría falta escalar.
barplot(sapply(curvas[,2:12], var)) #Varianza disminuye a midada que vencimiento de Posición se aleja. Haría falta escalar.
#--> media
sapply(curvas[,2:12], mean)  #Media de cada una de las variables. Media disminuye a medada que vencimiento de Posición se aleja, aunque muy similar. Haría falta centrar
barplot(sapply(curvas[,2:12], mean))
#--> Media y varianza: Ditribución por Posicións
boxplot(curvas[,2:12]) #Varianza y media disminuyen a medida que el vencimiento de Posicións se aleja


#PCA variables centradas, por lo que también escalaremos.
pcs <- prcomp(curvas[,2:12], center=TRUE, scale. = TRUE)
names(pcs)
pcs$scale
pcs$center
pcs$rotation
summary(pcs)

#comprobaciones
cor(pcs$x)
corrplot(cor(pcs$x), method = "color", tl.cex = 0.00001) #no correlación entre las componentes principales
sum((pcs$rotation[,1])^2) #loading vectors tienen norma 1
head(as.matrix(scale(curvas[2:12])) %*% pcs$rotation) #lo mismo que pcs$x, porque en realidad los score vectors se obtienen multiplicando la matriz con las variables originales ( centradas y/o escaladas) * matriz de rotación
head(pcs$x)
sum(pcs$rotation[,1]*pcs$rotation[,2]) #los loading vectors son ortogonales (producto escalar igual a cero)
sum(sapply(as.data.frame(pcs$x), var)) #la varianza total de los scores en las 11 componentes es 11, cada variable var=1 pq estan centradas y escaladas
#gráfico de contribución
var <- get_pca_var(pcs)
corrplot(var$contrib, is.corr = FALSE)
par(mfrow=c(2,1))
barplot(pcs$rotation[,1],ylab="PC1")
barplot(pcs$rotation[,2],ylab="PC2")
par(mfrow=c(1,1))

#proporcion varianza explicada
PVE <- summary(pcs)$importance[2,]
PVE_acum <- summary(pcs)$importance[3,]
par(mfrow=c(1,2))
plot(PVE, xlab="Componente principal", ylab="Proporcion de varianza explicada", type="b")
```

```
plot(PVE_acum, xlab="Componente principal", ylab="Varianza acumulada", type="b")
par(mfrow=c(1,1))
#se selccionan las dos primeras componentes principales

#visualizacion 2 primeras componentes principales
par(mfrow=c(2,1))
plot(pcs$x[,1],ylab="PC1")
plot(pcs$x[,2],ylab="PC2")
par(mfrow=c(1,1))
#valor medio cambia con el tiempo
#Nedeisadad de diferenciar en ARIMA. Quedaria así:
par(mfrow=c(2,1))
plot(diff(pcs$x[,1]),ylab="PC1")
plot(diff(pcs$x[,2]),ylab="PC2")
par(mfrow=c(1,1))
#valor no medio cambia con el tiempo

#biplot
biplot(pcs, scale = 0, choices = c(1,2))
fviz_pca_biplot(pcs, repel = TRUE, col.var = "black", col.ind = "orange")

#según fecha
fviz_pca_ind(pcs, col.ind = curvas$Date)+scale_color_gradient(low="green", high = "red")
#grafico animado
par(mfrow=c(1,2))
fviz_pca_ind(pcs, label = "none") + transition_time(curvas$Date)+labs(title = "Año:
{frame_time}")+
  shadow_mark(size=1.2, alpha=1)


############## Reconstrucciónccon 2 componentes principales #########
#Reconstruccion total
head(pcs$x%*%t(pcs$rotation)) #los datos originales estandarizados (centrados y escalados)
#para a los datos originales sin centrar y sin escalar hay que hacer las operaciones inversas
Xoriginal= Xnormalizado*sd + center
xorig<-as.data.frame(t(t(pcs$x%*%t(pcs$rotation))*pcs$scale+pcs$center))
xorig$Date <- curvas$Date

#Reconstruccion usando las dos componentes principales
curvas_rec2pc<-as.data.frame(t(t(pcs$x[,1:2]%*%t(pcs$rotation[,1:2]))*pcs$scale+pcs$center))
curvas_rec2pc$Date <- curvas$Date

#visualizacion
#Graficos curvas
ggplotly(evol) #grafico curvas original
#grafico curvas reconstruido
grafico_curvas <- curvas_rec2pc %>% pivot_longer(cols = -c(Date), names_to = "posicion",
values_to = "precio")
ordenpos <- c("P1", "P2", "P3", "P4", "P5", "P6", "P7", "P8", "P9", "P10", "P11")
grafico_curvas$posicion <- factor(grafico_curvas$posicion, levels = ordenpos)
evol2pc <- ggplot(grafico_curvas, aes(x = Date, y = precio, color = posicion, group = posicion))
+
  geom_line() +
  labs(x = "Fecha", y = "Precio", color = "Posición")
ggplotly(evol2pc)
```

```
#Visualización curvas por separado
#P1
p <- ggplot(data = curvas, mapping = aes(x=Date, y=P1))+
  geom_line(mapping = aes(x=Date, y=P1, color="Original"))+
  geom_line(data = curvas_rec2pc, mapping = aes(x=Date, y=P1, color="Reconstrucción"))
ggplotly(p)
#P2
p <- ggplot(data = curvas, mapping = aes(x=Date, y=P11))+
  geom_line(mapping = aes(x=Date, y=P11, color="Original"))+
  geom_line(data = curvas_rec2pc, mapping = aes(x=Date, y=P11, color="Reconstrucción"))
ggplotly(p)

#Curva de un dia
diaselec <- "2020-03-25 01:00:00"
datosdia <- curvas %>% filter(Date == diaselec) %>% pivot_longer(cols = starts_with("P"),
names_to = "posicion", values_to = "precio")
datosdia2pc <- curvas_rec2pc %>% filter(Date == diaselec) %>% pivot_longer(cols =
starts_with("P"), names_to = "posicion", values_to = "precio")
ordenpos <- c("P1", "P2", "P3", "P4", "P5", "P6", "P7", "P8", "P9", "P10", "P11")
datosdia$posicion <- factor(datosdia$posicion, levels = ordenpos)
datosdia2pc$posicion <- factor(datosdia2pc$posicion, levels = ordenpos)
datosdia$Type <- "Original"
datosdia2pc$Type <- "Reconstrucción"
orig_rec <- bind_rows(datosdia, datosdia2pc)
ggplot(data = orig_rec, mapping = aes(x = posicion, y = precio, color = Type, group = Type)) +
  geom_line() +
  geom_point() +
  labs(title = paste("Precios a", diaselec), x = "Posición", y = "Precio") +
  scale_color_manual(values = c("Original" = "red", "Reconstrucción" = "blue"))

#dataframe con las dos primeras componentes principales
curvas_pca <- as.data.frame(pcs$x[,1:2])
curvas_pca$Date <- as.Date(curvas$Date)
curvas_pca$Date <- as.Date(curvas_pca$Date)
curvas_pca <- as.data.frame(curvas_pca[, c("Date", "PC1", "PC2")])

#train (primeras 749 observaciones) y test sets (2 ultimas semanas)
train_data <- curvas_pca[1:749, ]
test_data <- curvas_pca[750:763, ]

################ OBJETIVO 1: MODELO ARIMA ################

#ACF y PACF para PC1 y PC2 para saber si hay que diferenciar o ya se puede determinar el orden
de ARIMA
par(mfrow = c(2, 2))
acf(train_data$PC1, main="ACF de PC1")
pacf(train_data$PC1, main="PACF de PC1")
acf(train_data$PC2, main="ACF de PC2")
pacf(train_data$PC2, main="PACF de PC2")
#hay que diferenciar

#visualizacion 2 primeras componentes principales
par(mfrow=c(2,1))
plot(pcs$x[,1],ylab="PC1")
```

```
plot(pcs$x[,2],ylab="PC2")
par(mfrow=c(1,1))
#valor medio cambia con el tiempo, por lo que se diferencia y quedaria así:
par(mfrow=c(2,1))
plot(diff(pcs$x[,1]),ylab="PC1")
plot(diff(pcs$x[,2]),ylab="PC2")
par(mfrow=c(1,1))
#valor medio no cambia con el tiempo

par(mfrow = c(2, 2))
acf(diff(train_data$PC1), main="ACF de PC1")
pacf(diff(train_data$PC1), main="PACF de PC1")
acf(diff(train_data$PC2), main="ACF de PC2")
pacf(diff(train_data$PC2), main="PACF de PC2")
#Ya es una serie estacionaria. PC1 (1,1,0). PC2 es ruido blanco (0,1,0)

#ajuste modelos ARIMA para PC1 y PC2
model_PC1 <- arima(train_data$PC1, order = c(1,1,0))
model_PC2 <- arima(train_data$PC2, order = c(0,1,0))
summary(model_PC1)
summary(model_PC2)
#resultados
accuracy(model_PC1)
accuracy(model_PC1)

#comprobaciones
#verificacion de coeficientes significativos
coeftest(model_PC1)
coeftest(model_PC2)
#supuesto de ruido blanco
par(mfrow = c(2, 2))
acf(model_PC1$residuals, main="ACF de PC1")
pacf(model_PC1$residuals, main="PACF de PC1")
acf(model_PC2$residuals, main="ACF de PC2")
pacf(model_PC2$residuals, main="PACF de PC2")
#test de Ljung-Box para los residuos para confirmar ruido blanco
Box.test(model_PC1$residuals, type = "Ljung-Box")
Box.test(model_PC2$residuals, type = "Ljung-Box")

##FROMA 1. funcion forecast() para predecir las observaciones del test dataset
forecast_PC1 <- forecast(model_PC1, h = nrow(test_data))
forecast_PC2 <- forecast(model_PC2, h = nrow(test_data))
summary(forecast_PC1)
summary(forecast_PC2)

#visualización predicciones
#PC1
comienzo <- start(forecast_PC1$mean)
test_data_ts <- ts(test_data$PC1, start=comienzo, frequency=frequency(forecast_PC1$mean))
autoplot(forecast_PC1) +
  autolayer(test_data_ts, series="Test Data") +
  labs( x="Fecha", y="PC1")
autoplot(forecast_PC1) +
  autolayer(test_data_ts, series="Test Data") +
  scale_x_continuous(limits = c(comienzo[1], max(time(forecast_PC1$mean)))) +
```

```
  labs( x="Fecha", y="PC1")
#PC2
comienzo <- start(forecast_PC2$mean)
test_data_ts <- ts(test_data$PC2, start=comienzo, frequency=frequency(forecast_PC2$mean))
autoplot(forecast_PC2) +
  autolayer(test_data_ts, series="Test Data") +
  labs( x="Fecha", y="PC2")
autoplot(forecast_PC2) +
  autolayer(test_data_ts, series="Test Data") +
  scale_x_continuous(limits = c(comienzo[1], max(time(forecast_PC2$mean)))) +
  labs( x="Fecha", y="PC2")

#Comparacion predicciones con observaciones originales usando la media como mejor valor
#PC1
comp_PC1 <- data.frame(
  Date = test_data$Date,
  Original = test_data$PC1,
  Predicho = forecast_PC1$mean)
comp_PC1
#error de predicción
mse_PC1 <- mean((comp_PC1$Original - comp_PC1$Predicho)^2)
mse_PC1
comienzo <- start(forecast_PC1$mean)
test_data_ts <- ts(test_data$PC1, start=comienzo, frequency=frequency(forecast_PC1$mean))
autoplot(forecast_PC1$mean, series="Predicción") +
  autolayer(test_data_ts, series="Test Data") +
  labs( x="Fecha", y="PC1")
#PC2
comp_PC2 <- data.frame(
  Date = test_data$Date,
  Original = test_data$PC2,
  Predicho = forecast_PC2$mean)
comp_PC2
#error de predicción
mse_PC2 <- mean((comp_PC2$Original - comp_PC2$Predicho)^2)
mse_PC2
comienzo <- start(forecast_PC2$mean)
test_data_ts <- ts(test_data$PC2, start=comienzo, frequency=frequency(forecast_PC2$mean))
autoplot(forecast_PC2$mean, series="Predicción") +
  autolayer(test_data_ts, series="Test Data") +
  labs( x="Fecha", y="PC2")

#FORMA 2. Coge test data dia anterior para seguir prediciendo
model2_PC1 <- Arima(test_data$PC1, model=model_PC1)
model2_PC2 <- Arima(test_data$PC2, model=model_PC2)
summary(model2_PC1)
summary(model2_PC2)
accuracy(model2_PC1)
accuracy(model2_PC2)
#predicciones
forecast2_PC1 <- fitted(model2_PC1)
forecast2_PC2 <- fitted(model2_PC2)

#comparacion predicciones con las observaciones originales
#PC1
```

```
comp_PC1 <- data.frame(
  Date = test_data$Date,
  Original = test_data$PC1,
  Predicho = forecast2_PC1
)
comp_PC1
#error de predicción
mse_PC1 <- mean((comp_PC1$Original - comp_PC1$Predicho)^2)
mse_PC1
comienzo <- start(forecast2_PC1)
test_data_ts <- ts(test_data$PC1, start=comienzo, frequency=frequency(forecast2_PC1))
autoplot(forecast2_PC1, series="Predicción") +
  autolayer(test_data_ts, series="Test Data") +
  labs( x="Fecha", y="PC1")
#PC2
comp_PC2 <- data.frame(
  Date = test_data$Date,
  Original = test_data$PC2,
  Predicho = forecast2_PC2
)
print(comp_PC2)
#error de predicción
mse_PC2 <- mean((comp_PC2$Original - comp_PC2$Predicho)^2)
mse_PC2
comienzo <- start(forecast2_PC2)
test_data_ts <- ts(test_data$PC2, start=comienzo, frequency=frequency(forecast2_PC2))
autoplot(forecast2_PC2, series="Predicción") +
  autolayer(test_data_ts, series="Test Data") +
  labs( x="Fecha", y="PC1")

#dataframe con predicciones usando la media como mejor valor
predicciones_df <- data.frame(
  Date = test_data$Date,
  predicciones_PC1 = forecast2_PC1,
  predicciones_PC2 = forecast2_PC2
)
colnames(predicciones_df) <- c("Date", "PC1", "PC2")
curvas_pca_pred <- rbind(curvas_pca[1:749,], predicciones_df)
View(curvas_pca_pred)

############## Reconstrucciónccon 2 componentes principales #########
curvas_pca_pred2 <- as.matrix(curvas_pca_pred[,2:3])
#Reconstruccion usando las dos componentes principales
curvas_pred_rec<-
as.data.frame(t(t(as.matrix(curvas_pca_pred2[,1:2])%*%t(pcs$rotation[,1:2]))*pcs$scale+pcs$c
enter))
curvas_pred_rec$Date <- as.POSIXct(curvas_pca_pred$Date)

#visualizacion
#Graficos curvas
ggplotly(evol) #grafico curvas original
#grafico curvas reconstruido
grafico_curvas <- curvas_pred_rec %>% pivot_longer(cols = -c(Date), names_to = "posicion",
values_to = "precio")
ordenpos <- c("P1", "P2", "P3", "P4", "P5", "P6", "P7", "P8", "P9", "P10", "P11")
```

```
grafico_curvas$posicion <- factor(grafico_curvas$posicion, levels = ordenpos)
evolarima <- ggplot(grafico_curvas, aes(x = Date, y = precio, color = posicion, group = posicion))
+
  geom_line() +
  labs(x = "Fecha", y = "Precio", color = "Posición")
ggplotly(evolarima)

#visualización curvas por separado
p <- ggplot(data = curvas[750:763,], mapping = aes(x=Date, y=P1))+
  geom_line(mapping = aes(x=Date, y=P1, color="Original"))+
  geom_line(data = curvas_pred_rec[750:763,], mapping = aes(x=Date, y=P1,
color="Reconstrucción"))
ggplotly(p)
p <- ggplot(data = curvas[750:763,], mapping = aes(x=Date, y=P11))+
  geom_line(mapping = aes(x=Date, y=P11, color="Original"))+
  geom_line(data = curvas_pred_rec[750:763,], mapping = aes(x=Date, y=P11,
color="Reconstrucción"))
ggplotly(p)

#Curva de un dia
diaselec <- "2022-01-14 01:00:00"
datosdia <- curvas %>% filter(Date == diaselec) %>% pivot_longer(cols = starts_with("P"),
names_to = "posicion", values_to = "precio")
datosdia2pc <- curvas_pred_rec %>% filter(Date == diaselec) %>% pivot_longer(cols =
starts_with("P"), names_to = "posicion", values_to = "precio")
ordenpos <- c("P1", "P2", "P3", "P4", "P5", "P6", "P7", "P8", "P9", "P10", "P11")
datosdia$posicion <- factor(datosdia$posicion, levels = ordenpos)
datosdia2pc$posicion <- factor(datosdia2pc$posicion, levels = ordenpos)
datosdia$Tipo <- "Original"
datosdia2pc$Tipo <- "Reconstrucción"
orig_rec <- bind_rows(datosdia, datosdia2pc)
ggplot(data = orig_rec, mapping = aes(x = posicion, y = precio, color = Tipo, group = Tipo)) +
  geom_line() +
  geom_point() +
  labs(title = paste("Precios a", diaselec), x = "Posición", y = "Precio") +
  scale_color_manual(values = c("Original" = "red", "Reconstrucción" = "blue"))

#Errores por posicion en los datos de test
curvasorig_test <- curvas[750:763,]
curvaspred_test <-curvas_pred_rec[750:763,]
errores <- data.frame()
for (i in 1:11) {
  posicion <- paste0("P", i)
  error_me <- mean(curvasorig_test[[posicion]] - curvaspred_test[[posicion]])
  error_mae <- mean(abs(curvasorig_test[[posicion]] - curvaspred_test[[posicion]]))
  error_mse <- mean((curvasorig_test[[posicion]] - curvaspred_test[[posicion]])^2)
  errores <- rbind(errores, c(posicion,error_me,error_mae, error_mse))
}
colnames(errores) <- c("Posición", "ME", "MAE", "MSE")
errores
ordenpos <- c("P1", "P2", "P3", "P4", "P5", "P6", "P7", "P8", "P9", "P10", "P11")
errores$Posición <- factor(errores$Posición, levels = ordenpos)
#ME
errores$ME <- as.numeric(errores$ME)
ggplot(errores, aes(x = Posición, y = 1, fill = ME)) +
```

```
  geom_tile() +
  geom_text(aes(label = sprintf("%.1f", ME)), color = "black", size = 4) +
  scale_fill_gradient2(low = "red", high = "blue", mid = "white", midpoint = 0, space = "Lab",
na.value = "grey50", guide = "colourbar", aesthetics = "fill")
#MAE
errores$MAE <- as.numeric(errores$MAE)
ggplot(errores, aes(x = Posición, y = MAE)) +
  geom_bar(stat = "identity") +
  geom_text(aes(label = sprintf("%.1f", MAE)), vjust = -0.3, size = 3.5)

############### OBJETIVO 2: GENERACIÓN DE ESCENARIOS CON SIMULACIONES
DE ARIMA EN BASE A ERRORES HISTÓRICOS #############
# se tiene en cuenta el error que no tenia en cuenta arima a la hora de hacer  las predicciones
plot(model_PC1$residual)
nsim <-10
h <- nrow(test_data)

#PC1
escenarios_PC1 <- matrix(0, nrow=h, ncol=nsim)
for(i in seq(h)){
  escenarios_PC1[i,] <- simulate(model_PC1,nsim=nsim, seed=100+i)
}
escenarios_PC1
#viz
comienzo <- start(escenarios_PC1)
escenarios_PC1 <- ts(escenarios_PC1, start=comienzo, frequency=frequency(escenarios_PC1))
test_data_ts_pc1 <- ts(test_data$PC1, start=comienzo, frequency=frequency(escenarios_PC1))
autoplot(test_data_ts_pc1) +
  autolayer(escenarios_PC1, colour=TRUE) +
  autolayer(test_data_ts_pc1, colour=FALSE) +
  ylab("PC1")

#PC2
escenarios_PC2 <- matrix(0, nrow=h, ncol=nsim)
for(i in seq(h)){
  escenarios_PC2[i,] <- simulate(model_PC2,nsim=nsim, seed=10+i)
}
escenarios_PC2
#viz
comienzo <- start(escenarios_PC2)
escenarios_PC2 <- ts(escenarios_PC2, start=comienzo, frequency=frequency(escenarios_PC2))
test_data_ts_pc2 <- ts(test_data$PC2, start=comienzo, frequency=frequency(escenarios_PC2))
autoplot(test_data_ts_pc2) +
  autolayer(escenarios_PC2, colour=TRUE) +
  autolayer(test_data_ts_pc2, colour=FALSE) +
  ylab("PC2")

#dataframe con predicciones
train_data_df <- as.data.frame(train_data)
test_data_df <- as.data.frame(test_data)
escenarios_PC1_df <- as.data.frame(escenarios_PC1)
escenarios_PC1_df$Date <- test_data$Date
escenarios_PC2_df <- as.data.frame(escenarios_PC2)
escenarios_PC2_df$Date <- test_data$Date
```

```
############# Reconstrucciónccon 2 componentes principales #########
nsim <- 10
total_escenarios <- list()
for (i in 1:nsim) {
  escenario_pc1_pc2 <- cbind(escenarios_PC1_df[, c("Date", paste0("Series ", i))],
escenarios_PC2_df[, paste0("Series ", i)])
  names(escenario_pc1_pc2)[2:3] <- c("PC1", "PC2")
  traintest <- bind_rows(train_data_df, escenario_pc1_pc2)
  curvas_pred_rec <- as.data.frame(t(t(as.matrix(traintest[, 2:3]) %*% t(pcs$rotation[, 1:2])) *
pcs$scale + pcs$center))
  curvas_pred_rec$Date <- curvas$Date
  curvas_pred_rec$Escenario <- paste0("Escenario_", i)
  total_escenarios[[i]] <- curvas_pred_rec
}
total_escenarios_df <- bind_rows(total_escenarios)

#visualizacion serie con escenarios
total_escenarios_pivot <- total_escenarios_df %>% pivot_longer(cols = -c(Date, Escenario),
names_to = "posicion", values_to = "precio")
ordenpos <- c("P1", "P2", "P3", "P4", "P5", "P6", "P7", "P8", "P9", "P10", "P11")
total_escenarios_pivot$posicion <- factor(total_escenarios_pivot$posicion, levels = ordenpos)
ggplot(total_escenarios_pivot, aes(x = Date, y = precio, color = posicion, group =
interaction(Escenario, posicion))) +
  geom_line() +
  labs(x = "Fecha", y = "Precio", color = "Posición")

#visualización curvas por separado
#P1
original <- curvas[750:763,]%>% select(Date, P1)
names(original)[2] <- "Original"
total_escenarios_df$Date <- as.POSIXct(total_escenarios_df$Date)
escenarios_p1 <- total_escenarios_df %>% select(Date, P1, Escenario)
fechas_comunes <- intersect(original$Date, escenarios_p1$Date)
escenarios_p1 <- escenarios_p1 %>% filter(Date %in% fechas_comunes)
original <- original %>% rename(P1 = Original) %>% mutate(Escenario = "Original")
datacomb <- bind_rows(original, escenarios_p1)
datacomb_pivot <- datacomb %>% pivot_longer(cols = -c(Date, Escenario), names_to =
"posicion", values_to = "precio")
ggplot(datacomb_pivot, aes(x = Date, y = precio, color = Escenario, group = Escenario)) +
  geom_line(data = datacomb_pivot %>% filter(Escenario != "Original"), size = 1)+
  geom_line(data = datacomb_pivot %>% filter(Escenario == "Original"), color = "black", size =
1.5)+
  labs(x = "Fecha", y = "Precios de P1")
#P11
original <- curvas[750:763,]%>% select(Date, P11)
names(original)[2] <- "Original"
total_escenarios_df$Date <- as.POSIXct(total_escenarios_df$Date)
escenarios_p11 <- total_escenarios_df %>% select(Date, P11, Escenario)
fechas_comunes <- intersect(original$Date, escenarios_p1$Date)
escenarios_p11 <- escenarios_p11 %>% filter(Date %in% fechas_comunes)
original <- original %>% rename(P11 = Original) %>% mutate(Escenario = "Original")
datacomb <- bind_rows(original, escenarios_p11)
datacomb_pivot <- datacomb %>% pivot_longer(cols = -c(Date, Escenario), names_to =
"posicion", values_to = "precio")
```

```
ggplot(datacomb_pivot, aes(x = Date, y = precio, color = Escenario, group = Escenario)) +
  geom_line(data = datacomb_pivot %>% filter(Escenario != "Original"), size = 1)+
  geom_line(data = datacomb_pivot %>% filter(Escenario == "Original"), color = "black", size =
1.5)+
  labs(x = "Fecha", y = "Precios de P11")

#curvas de un dia especifico
diaselec <- as.Date("2022-01-14")
total_escenarios_df$Date <- as.Date(total_escenarios_df$Date)
escenarios_dia <- total_escenarios_df %>% filter(Date == diaselec)
curvas_sim <- curvas
curvas_sim$Date <- as.Date(curvas$Date)
curvas_dia <- curvas_sim %>% filter(Date == diaselec)
curvas_dia$Escenario <- "Original"
datacomb <- bind_rows(curvas_dia, escenarios_dia)
datacomb_pivot <- datacomb %>% pivot_longer(cols = -c(Date, Escenario), names_to =
"posicion", values_to = "precio")
ordenpos <- c("P1", "P2", "P3", "P4", "P5", "P6", "P7", "P8", "P9", "P10", "P11")
datacomb_pivot$posicion <- factor(datacomb_pivot$posicion, levels = ordenpos)
#visualización
ggplot(datacomb_pivot, aes(x = posicion, y = precio, color = Escenario, group = Escenario)) +
  geom_line(data = datacomb_pivot%>% filter(Escenario != "Original"), size = 1)+
  geom_point(data = datacomb_pivot %>% filter(Escenario != "Original"), size = 1.5)+
    geom_line(data = datacomb_pivot%>% filter(Escenario == "Original"), color = "black", size =
1.5)+
  geom_point(data = datacomb_pivot %>% filter(Escenario == "Original"), color = "black", size
= 3)+
  labs(title = paste("Precios a ", diaselec), x = "Posición", y = "Precio")

################## OBJETIVO 3: VARIABLE EXPLICATIVA  ##################
curvas_train_data <- curvas[1:749, ]
curvas_test_data <- curvas[750:763, ]

#ajudete modelos para PC1 y PC2
model_PC1_ve <- arima(train_data$PC1, order = c(0,0,0), xreg=curvas_train_data$P1)
model_PC2_ve <- arima(train_data$PC2, order = c(0,0,0), xreg=curvas_train_data$P1)
#ACF y PACF para PC1 y PC2 para saber si hay que diferenciar o ya se puede determinar el orden
del modelo
par(mfrow = c(2, 2))
acf(model_PC1_ve$residuals, main="ACF de PC1")
pacf(model_PC1_ve$residuals, main="PACF de PC1")
acf(model_PC2_ve$residuals, main="ACF de PC2")
pacf(model_PC2_ve$residuals, main="PACF de PC2")
#hay que diferenciar

#ajuste modelos para PC1 y PC2 para determinar el orden delo modelo
model_PC1_ve <- arima(train_data$PC1, order = c(0,1,0), xreg=curvas_train_data$P1)
model_PC2_ve <- arima(train_data$PC2, order = c(0,1,0), xreg=curvas_train_data$P1)
#ACF y PACF para PC1 y PC2 para determinar el orden del modelo
par(mfrow = c(2, 2))
acf(model_PC1_ve$residuals, main="ACF de PC1")
pacf(model_PC1_ve$residuals, main="PACF de PC1")
acf(model_PC2_ve$residuals, main="ACF de PC2")
pacf(model_PC2_ve$residuals, main="PACF de PC2")
#Ya es una serie estacionaria. PC1 (1,1,0). PC2 es ruido blanco (0,1,0)
```

```
#Ajustar modelos ARIMA para PC1 y PC2
model_PC1_ve <- arima(train_data$PC1, order = c(1,1,0), xreg=curvas_train_data$P1)
model_PC2_ve <- arima(train_data$PC2, order = c(0,1,0), xreg=curvas_train_data$P1)
summary(model_PC1_ve)
summary(model_PC2_ve)
#resultados
accuracy(model_PC1_ve)
accuracy(model_PC2_ve)

#comprobaciones
#verificacion de coeficientes significativos
coeftest(model_PC1_ve)
coeftest(model_PC2_ve)
#supuesto de ruido blanco
par(mfrow = c(2, 2))
acf(model_PC1_ve$residuals, main="ACF de PC1")
pacf(model_PC1_ve$residuals, main="PACF de PC1")
acf(model_PC2_ve$residuals, main="ACF de PC2")
pacf(model_PC2_ve$residuals, main="PACF de PC2")
#test de Ljung-Box para los residuos para confirmar ruido blanco
Box.test(model_PC1_ve$residuals, type = "Ljung-Box")
Box.test(model_PC2_ve$residuals, type = "Ljung-Box")

#prediccion de las observaciones usando una variable explicativa
forecast_PC1_ve <- predict(model_PC1_ve, newxreg=curvas_test_data$P1)
forecast_PC2_ve <- predict(model_PC2_ve, newxreg=curvas_test_data$P1)

#Comparacion predicciones con observaciones originales
#PC1
comp_PC1 <- data.frame(
  Date = test_data$Date,
  Original = test_data$PC1,
  Predicho = forecast_PC1_ve$pred)
comp_PC1
#error de predicción
mse_PC1_ve <- mean((comp_PC1$Original - comp_PC1$Predicho)^2)
mse_PC1_ve
comienzo <- start(forecast_PC1_ve$pred)
test_data_ts <- ts(test_data$PC1, start=comienzo, frequency=frequency(forecast_PC1_ve$pred))
autoplot(forecast_PC1_ve$pred, series="Predicción") +
  autolayer(test_data_ts, series="Test Data") +
  labs( x="Fecha", y="PC1")
#PC2
comp_PC2 <- data.frame(
  Date = test_data$Date,
  Original = test_data$PC2,
  Predicho = forecast_PC2_ve$pred)
comp_PC2
#error de predicción
mse_PC2_ve <- mean((comp_PC2$Original - comp_PC2$Predicho)^2)
mse_PC2_ve
comienzo <- start(forecast_PC2_ve$pred)
test_data_ts <- ts(test_data$PC2, start=comienzo, frequency=frequency(forecast_PC2_ve$pred))
autoplot(forecast_PC2_ve$pred, series="Predicción") +
```

```
  autolayer(test_data_ts, series="Test Data") +
  labs( x="Fecha", y="PC2")

#dataframe con las predicciones
predicciones_df <- data.frame(
  Date = test_data$Date,
  predicciones_PC1 = forecast_PC1_ve$pred,
  predicciones_PC2 = forecast_PC2_ve$pred
)
colnames(predicciones_df) <- c("Date", "PC1", "PC2")
curvas_pca_pred <- rbind(curvas_pca[1:749,], predicciones_df)

############# Reconstrucciónccon 2 componentes principales #########
curvas_pca_pred2 <- as.matrix(curvas_pca_pred[,2:3])
#Reconstruccion usando las dos componentes principales
curvas_pred_rec<-
as.data.frame(t(t(as.matrix(curvas_pca_pred2[,1:2])%*%t(pcs$rotation[,1:2]))*pcs$scale+pcs$c
enter))
curvas_pred_rec$Date <- as.POSIXct(curvas_pca_pred$Date)

#visualizacion
#Graficos curvas
ggplotly(evol) #grafico curvas original
#grafico curvas reconstruido
grafico_curvas <- curvas_pred_rec %>% pivot_longer(cols = -c(Date), names_to = "posicion",
values_to = "precio")
ordenpos <- c("P1", "P2", "P3", "P4", "P5", "P6", "P7", "P8", "P9", "P10", "P11")
grafico_curvas$posicion <- factor(grafico_curvas$posicion, levels = ordenpos)
evolarima <- ggplot(grafico_curvas, aes(x = Date, y = precio, color = posicion, group = posicion))
+
  geom_line() +
  labs(x = "Fecha", y = "Precio", color = "Posición")
ggplotly(evolarima)

#visualización curvas por separado
p <- ggplot(data = curvas[750:763,], mapping = aes(x=Date, y=P1))+
  geom_line(mapping = aes(x=Date, y=P1, color="Original"))+
  geom_line(data = curvas_pred_rec[750:763,], mapping = aes(x=Date, y=P1,
color="Reconstrucción"))
ggplotly(p)
p <- ggplot(data = curvas[750:763,], mapping = aes(x=Date, y=P11))+
  geom_line(mapping = aes(x=Date, y=P11, color="Original"))+
  geom_line(data = curvas_pred_rec[750:763,], mapping = aes(x=Date, y=P11,
color="Reconstrucción"))
ggplotly(p)

#Curva de un dia
diaselec <- "2021-01-14 01:00:00"
datosdia <- curvas %>% filter(Date == diaselec) %>% pivot_longer(cols = starts_with("P"),
names_to = "posicion", values_to = "precio")
datosdia2pc <- curvas_pred_rec %>% filter(Date == diaselec) %>% pivot_longer(cols =
starts_with("P"), names_to = "posicion", values_to = "precio")
ordenpos <- c("P1", "P2", "P3", "P4", "P5", "P6", "P7", "P8", "P9", "P10", "P11")
datosdia$posicion <- factor(datosdia$posicion, levels = ordenpos)
datosdia2pc$posicion <- factor(datosdia2pc$posicion, levels = ordenpos)
```

```
datosdia$Tipo <- "Original"
datosdia2pc$Tipo <- "Reconstrucción"
orig_rec <- bind_rows(datosdia, datosdia2pc)
ggplot(data = orig_rec, mapping = aes(x = posicion, y = precio, color = Tipo, group = Tipo)) +
  geom_line() +
  geom_point() +
  labs(title = paste("Precios a", diaselec), x = "Posición", y = "Precio") +
  scale_color_manual(values = c("Original" = "red", "Reconstrucción" = "blue"))

#Errores por posicion en los datos de test
curvasorig_test <- curvas[750:763,]
curvaspred_test <-curvas_pred_rec[750:763,]
errores_ve <- data.frame()
for (i in 1:11) {
  posicion <- paste0("P", i)
  error_me <- mean(curvasorig_test[[posicion]] - curvaspred_test[[posicion]])
  error_mae <- mean(abs(curvasorig_test[[posicion]] - curvaspred_test[[posicion]]))
  error_mse <- mean((curvasorig_test[[posicion]] - curvaspred_test[[posicion]])^2)
  errores_ve <- rbind(errores_ve, c(posicion,error_me,error_mae, error_mse))
}
colnames(errores_ve) <- c("Posición", "ME", "MAE", "MSE")
errores_ve
ordenpos <- c("P1", "P2", "P3", "P4", "P5", "P6", "P7", "P8", "P9", "P10", "P11")
errores_ve$Posición <- factor(errores_ve$Posición, levels = ordenpos)
#ME
errores_ve$ME <- as.numeric(errores_ve$ME)
ggplot(errores_ve, aes(x = Posición, y = 1, fill = ME)) +
  geom_tile() +
  geom_text(aes(label = sprintf("%.1f", ME)), color = "black", size = 4) +
  scale_fill_gradient2(low = "red", high = "blue", mid = "white", midpoint = 0, space = "Lab",
na.value = "grey50", guide = "colourbar", aesthetics = "fill")
#MAE
errores_ve$MAE <- as.numeric(errores_ve$MAE)
ggplot(errores_ve, aes(x = Posición, y = MAE)) +
  geom_bar(stat = "identity") +
  geom_text(aes(label = sprintf("%.1f", MAE)), vjust = -0.3, size = 3.5)

############ CONCLUSION: COMPARACION MODELO ARIMA (TAREA 1) VS.
MODELO ARIMA CON VARIABLE EXPLICATIVA (V3) #########
# Agregar una columna 'Dataset' para identificar cada dataframe
errores$Modelo <- "ARIMA"
errores_ve$Modelo <- "ARIMA con VE"
comperrores <- rbind(errores, errores_ve)
comperrores$Posición <- factor(comperrores$Posición, levels = errores$Posición)
#ME
ggplot(comperrores, aes(x = Posición, y = ME, fill = Modelo)) +
  geom_bar(stat = "identity", position = "dodge") +
  geom_text(aes(label = sprintf("%.1f", ME)), vjust = -0.3, position = position_dodge(1), size =
3.5) +
  labs( x = "Posición", y = "ME")
#MAE
ggplot(comperrores, aes(x = Posición, y = MAE, fill = Modelo)) +
  geom_bar(stat = "identity", position = "dodge") +
  geom_text(aes(label = sprintf("%.1f", MAE)), vjust = -0.3, position = position_dodge(1), size =
3.5) +  labs( x = "Posición", y = "MAE")
```