Texto

Descripción generada automáticamente

Aprendizaje Reforzado Profundo para la Administración de Portafolios de Renta Fija

« » Deep Reinforcement Learning for Automated Fixed Income Portfolio Management

Estudiante

David Mejía Estrada

[dmejiae3@eafit.edu.co](mailto:dmejiae3@eafit.edu.co)

Asesora

Paula María Almonacid Hurtado

[palmona1@eafit.edu.co](mailto:palmona1@eafit.edu.co)

UNIVERSIDAD EAFIT

Escuela de Administración

Maestría en Ciencias de los Datos y Analítica

Medellín

2023

**Resumen**

En este trabajo se aplican técnicas de aprendizaje reforzado profundo en la administración de portafolios de inversión de renta fija, específicamente títulos soberanos emitidos por el gobierno colombiano. El periodo de análisis comprende siete años, desde enero de 2015 hasta diciembre de 2022. Encontramos que es posible generar rentabilidad y lograr una eficiente gestión del riesgo como resultado de las estrategias de “trading” que los modelos de aprendizaje reforzado profundo prevén más convenientes dadas ciertas condiciones de mercado y de cada uno de los títulos, como su riesgo implícito en métricas como DV01, Duración y Convexidad. Finalmente, este estudio contribuye al muy estudiado campo de las aplicaciones de aprendizaje de máquina e inteligencia artificial sobre predicción del mercado de valores y administración de carteras de inversión.

**Abstract**

This paper applies deep reinforced learning techniques to the management of fixed income investment portfolios, specifically sovereign securities issued by the Colombian government. The period of analysis covers seven years, from January 2015 to December 2022. We find that it is possible to generate profitability and achieve efficient risk management because of the trading strategies that deep reinforced learning models foresee more convenient given certain market conditions and of each of the securities, such as their implied risk in metrics like DV01, Duration and Convexity. Finally, this study contributes to the much-studied field of machine learning and artificial intelligence applications on stock market prediction and portfolio management.

*Palabras Clave:* Yield curve; Machine Learning; Trading strategy; Deep Reinforcement Learning; Fixed Income; Risk Management; Portfolio Management.

**1.** **INTRODUCCIÓN**

**1.1. Planteamiento del problema**

Las estrategias de trading sobre portafolios de renta fija desempeñan un papel fundamental en la gestión de activos y pasivos de las entidades financieras y de cualquiera que gestione este tipo de portafolios de manera independiente[[1]](#footnote-1). En todo momento, estos agentes intentan rentabilizar sus portafolios al tiempo que intentan minimizar el riesgo de tasa de interés asumido, utilizando para ello la información que puede entregar la curva de rendimientos y sus hipotéticos movimientos futuros -como su aplanamiento o empinamiento- así como la posible variación de las tasas de interés y precios de mercado de los títulos que componen la curva vistos de manera individual. Todos estos agentes se encuentran en los mercados financieros de deuda, con objetivos diferentes respecto a la gestión de sus activos y pasivos, siendo los mercados de deuda soberana -este es, donde se negocian los títulos de deuda emitidos por el gobierno de un país para financiar sus necesidades de gasto público- uno de los elegidos por estos agentes para esa labor. En este trabajo, se plantea trabajar con el mercado de deuda soberana de Colombia, lugar donde confluyen diversos agentes con objetivos diferentes.

El mercado de bonos soberanos es un mercado financiero importante, amplio y líquido en Colombia, donde bancos, instituciones e inversionistas extranjeros y privados negocian todos los días grandes cantidades de esos bonos emitidos por el gobierno colombiano[[2]](#footnote-2). Usualmente, esas instituciones no sólo invierten en un título o tenor específico, sino que también tienen una gama de posibilidades para negociar en ese mercado, como, por ejemplo, apalancar ventas en corto con operaciones repo, y operaciones de corretaje, entre otras, que juegan un papel crítico en la gestión de activos y pasivos para esas empresas. Para ello, las instituciones financieras y los operadores independientes constituyen portafolios de inversión que tienden a valorizarse con el tiempo debido a los intereses que nocionalmente devengan estos títulos. Sin embargo, los gestores de carteras pueden aumentar su rentabilidad negociando los bonos antes de su fecha de vencimiento, lo que implica que los portafolios de Renta Fija tendrán una exposición significativa al riesgo de mercado, en este caso, al riesgo de tasas de interés.

Los gestores de portafolios no sólo toman posiciones en un nodo concreto de la curva de rendimientos, sino que también lo hacen a lo largo de toda la curva. La estrategia depende de las expectativas sobre los movimientos de la curva. Por ejemplo, si el gestor espera que la curva de rendimientos se aplane -lo que significa que las tasas de corto plazo subirán mientras que las tasas de largo plazo caerán-, entonces tomará posiciones cortas en bonos a corto plazo y posiciones largas en bonos a largo plazo, y dicha estrategia tendrá un riesgo de mercado asociado.

Esta es una tarea compleja, porque tanto el movimiento de los precios y tasas de interés de los bonos vistos de manera individual como los movimientos de la curva de rendimientos completa son completamente erráticos, volátiles y no lineales, lo que supone una dificultad importante para hacer predicciones (Henrique y otros, 2019), y, por tanto, para tomar posiciones de trading a lo largo de la curva mientras también se busca minimizar el riesgo de mercado asumido.

**1.2. Justificación**

Zenios y Zeimba (2007), explican las carteras o portafolios de renta fija desempeñan un papel importante en la gestión de activos y pasivos, tanto para las instituciones financieras como para las no financieras, e incluso para otros fines y contextos. Sin embargo, el mercado de renta fija presenta cierta complejidad, especialmente en la gestión del riesgo de mercado. Los administradores de portafolios suelen tener que jugar con muchas variables dentro de su actuación, como la inflación, la política monetaria y la liquidez del mercado, entre otras. Esas variables cambian a lo largo del tiempo en patrones no lineales, y afectan a la formación de los precios de los títulos de renta fija, y, por tanto, de la curva de rendimientos.

Entonces, es demasiado importante para la industria financiera encontrar nuevas alternativas eficientes para la gestión de carteras de renta fija, utilizando tanto estrategias de curva como posiciones direccionales individuales. Los gestores de carteras buscan aumentar su rentabilidad a la vez que reducen su riesgo de mercado, por lo que el objetivo principal es maximizar la relación entre la rentabilidad y el riesgo asumido.

En la actualidad, el Machine Learning ha ayudado a resolver algunos problemas de optimización en otras áreas de conocimiento e incluso en la gestión de carteras de valores[[3]](#footnote-3), especialmente en los mercados accionarios y de divisas, donde existe mucha literatura al respecto. En cambio, es bastante importante el potencial para aplicar diferentes modelos de Machine Learning en carteras de renta fija, donde dichos modelos podrían ayudar a aumentar la precisión de las predicciones y mejorar el rendimiento de las carteras.

**1.3. Objetivos**

El propósito de este trabajo es utilizar algoritmos de Aprendizaje por Refuerzo para correr una estrategia, o un ensamble de estrategias, de gestión sobre un portafolio de renta fija, específicamente deuda soberana colombiana, y hacer una comparación de su desempeño, medido por la relación entre el rendimiento total del portafolio y el riesgo de mercado asumido, con otro tipo de metodologías.

**Objetivos específicos**

Este trabajo tiene como objetivos específicos los siguientes:

* Realizar un ciclo completo de ingeniería de datos, lo que implica recolectar los datos del mercado público de deuda soberana colombiana, realizar limpiezas generales y de outliers en caso de aplicar, hacer una descripción completa de las variables y los datasets, transformar los datos según las necesidades y características del mercado, basado en criterio experto profesional.
* Entrenar, probar y ensamblar modelos de Aprendizaje por Refuerzo para administrar una cartera de inversiones en el mercado de bonos soberanos de Colombia, utilizando para ello los datasets recolectados y las transformaciones a las que estos fueren sometidos, buscando alcanzar objetivos de rentabilidad y riesgo asumido, ajustando hiperparámetros según las necesidades.
* Evaluar el desempeño de los modelos, con base en diferentes métricas financieras y de ciencia de datos. Es importante en este objetivo tener en cuenta la necesidad de comparar el desempeño de las estrategias de Aprendizaje por Refuerzo con respecto a estrategias convencionales y frente a referencias -benchmarks- de mercado.

**2. REVISIÓN DE LITERATURA**

**2.1 Aprendizaje de Máquina para Predicción del Mercado de Valores**

El aprendizaje automático ha sido una solución común para los problemas financieros en los mercados de capitales (López de Prado, 2020), donde es ampliamente conocido, como explica Hull, que los precios de los instrumentos financieros que transan en mercados financieros siguen un movimiento browniano geométrico, que clasifica dentro del espectro de los procesos aleatorios markovianos (Hull, 2022), concepto que ha moldeado la forma en la que los agentes del mercado realizan valoraciones sobre instrumentos financieros derivados y de deuda desde mediados de los años setenta con la aparición de la teoría de valoración de opciones (Black & Scholes, 1973) y los aportes de Merton (1973) sobre la racionalidad de dichas valoraciones.

En ese orden de ideas, las aplicaciones del aprendizaje automático en los mercados financieros han sido un tema importante y siguen siendo una rama de investigación relevante para los mercados financieros que está en continuo desarrollo (Henrique y otros, 2019), en específico por la dificultad que se encuentra en la naturaleza aleatoria no estacionaria de las series de tiempo financieras (Zhang y otros, 2017). El inicio del aprendizaje automático en los mercados financieros está ligado con el auge de las redes neuronales a finales del siglo pasado (Refenes y otros, 1997), donde se trataron aplicaciones sobre predicción del mercado accionario con resultados iniciales satisfactorios mejores que resultados obtenidos por algunos análisis más tradicionales (Yoon y otros, 1993). Comenzando la siguiente década, otro tipo de algoritmos comenzaron a ser utilizados para la predicción del comportamiento de los mercados financieros, como Máquinas de Soporte Vectorial (Fernández-Rodrı́guez y otros, 2000).

Algunos precedentes a la estimación de nodos de la curva desde perspectivas de aprendizaje de máquina tuvieron que ver con un objetivo académico proveniente de la década de los ochenta, que pretendía establecer si una estrategia activa de administración de portafolios de renta fija en la que primase el tomar posiciones compradoras y vendedoras a lo largo de la curva de rendimientos era o no más rentable sobre una estrategia más conservadora en la que solo se comprasen y mantuviesen los títulos emitidos hasta el vencimiento. Inicialmente, Dyl y Joehnk (1981) concluyeron que esta estrategia fue más rentable que las letras del tesoro americano, es decir, que las tasas de interés libres de riesgo de más corto plazo, entre los años 1970 y 1975. En otros estudios como Grieves y Marcus (1992) y Peláez (1997) se encontró evidencia empírica de que la estrategia activa fue superior al tradicional «comprar y mantener» en otros marcos temporales. En cambio, en Ang y otros (1998) y en Chua y otros (2005), la evidencia encontrada es mixta y no concluyente. Finalmente, Galvani y Landon (2013) sugieren que la estrategia activa es inefectiva desde un punto de vista de gestión de riesgos de mercado a través del concepto de mínima varianza cuando dicha estrategia incluye compras y ventas de títulos de largo plazo, situación atribuible a la mayor cantidad de convexidad y duración modificada que entonces recaería sobre el portafolio, incrementando el riesgo de mercado final (Fabozzi, 2021).

Dada la evidencia encontrada resumida en el párrafo anterior, otros autores comenzaron a explorar diversas técnicas de aprendizaje de máquina y su aplicación específica a la predicción de tasas de interés y de la curva de rendimientos, así como la optimización de estrategias activas sobre la curva, como, por ejemplo, la descrita en Zimmermann y otros (2000), en donde los autores encontraron que las técnicas convencionales para la predicción de diez nodos elegidos de la curva de rendimientos alemana son superados por una arquitectura de redes neuronales ajustadas por error de modelo. O en Gogas y otros (2015), quienes usaron variables macroeconómicas para modelar, con uso de Máquinas de Soporte Vectorial, la dirección de las tasas de interés y la ocurrencia de recesiones económicas, obteniendo resultados positivos en cuanto a la predicción de estos dos objetivos, además de superar modelos estadísticos convencionales estándar logit y probit.

**2.2 Aprendizaje por Refuerzo: Teoría y Aplicación**

El aprendizaje por refuerzo es un campo de estudio del aprendizaje de máquina, con características de aprendizaje no supervisado, y con un pasado que se remonta a los estudios de Bellman (1952) (1966), en donde el autor plantea los cimientos de lo que llamó «programación dinámica», cuyo principal propósito era crear algoritmos de optimización con capacidad de adaptarse a nuevos estados dentro de un espacio de posibilidades de esos estados, haciendo un símil con la naturaleza humana del constante aprendizaje.

Este objetivo de optimización dentro de un espacio amplio de posibilidades de estados implica la necesidad de la existencia de un agente dentro del algoritmo con capacidad de llevar a cabo dicha optimización, lo que implica que los algoritmos de aprendizaje por refuerzo entreguen premios al agente por ejecutar una secuencia de decisiones basadas en probabilidad, que, correctas o incorrectas, lleven al agente a obtener la mayor cantidad de dicho premio. A esta secuencia probabilística de decisiones se le llama «política», mientras que el conjunto de estados posibles y de acciones, en conjunto, dada su aleatoriedad, permiten describir al aprendizaje por refuerzo como un Proceso de Decisión de Markov (Arulkumaran y otros, 2017).

La naturaleza aleatoria de los Procesos de Decisión de Markov -en adelante PDM-, representa un reto para el agente, puesto que se según lo mostrado en (Sutton, Temporal credit assignment in reinforcement learning, 1984), el número de acciones consecutivas que el agente puede ejecutar en cada marco temporal es limitado por las propias limitaciones existentes en el espacio de posibilidades. A manera de ejemplo, si en cierto PDM solo pueden darse un número determinado de estados diferentes, y, suponiendo que conocemos la combinación de acciones consecutivas que maximiza el premio para el agente, entonces cualquier acción incorrecta que el agente tome le impedirá alcanzar el premio óptimo en el futuro. Este problema para el agente se conoce como el problema de asignación temporal de crédito, y fue ampliamente abordado por Watkins (1989) en su tesis doctoral.

La solución de Watkins, bautizada como «Q-Learning», fue un hito que marcó una década de los noventa con valiosos aportes, estudios y variantes del aprendizaje por refuerzo. Dicha solución, basada en la ecuación de Bellman (1952), consiste en utilizar simulaciones basadas en métodos Monte Carlo para realizar un mapeo repetitivo y completo de todas las posibles políticas que el agente puede tomar dado cierto estado, así como sus potenciales premios, y utilizando un factor de descuento definido que permite darle más importancia a los premios de corto plazo (Arulkumaran y otros, 2017). Finalmente, el agente elige ejecutar la acción que tiene un potencial mayor premio. Posteriormente, en (Watkins & Dayan, 1992), se muestra como las decisiones tomadas por el agente convergen a las acciones óptimas cuando se realiza el mapeo completo de los estados y premios posibles al utilizar Q-Learning.

Dentro del amplio espectro de trabajos sobre aprendizaje por refuerzo en los años noventa, se destacan estudios como (Sutton y otros, 1999), quienes encontraron que gracias al enfoque probabilístico de Q-Learning, es posible llegar a mayores niveles de abstracción en los problemas de aprendizaje por refuerzo no solo en PDM, sino también en PDM parciales, es decir, en donde no es posible para el agente observar todos los posibles estados dado un estado actual.

En la década siguiente surgieron de manera primitiva algunos de los algoritmos más utilizados de aprendizaje por refuerzo en la actualidad. En (Busoniu y otros, 2008) se cita una importante cantidad de estudios sobre aprendizaje reforzado multiagente, es decir, en cuyos algoritmos existe más de un agente capaz de ejecutar una política bajo diferentes incentivos o premios. También en esta década surgieron algoritmos y aplicaciones para aprendizaje por refuerzo con «gradiente de política», inspirados en el proceso de «backpropagation» para entrenamiento de redes neuronales (Kakade, 2001). Este tipo de algoritmos llegó a tener buenos resultados en aplicaciones relacionadas con la robótica, como por ejemplo en el trabajo de (Stone & Kohl, 2004), quienes utilizando este tipo de aprendizaje con gradiente de política como optimizador obtuvieron mejores resultados para entrenar a un robot cuadrúpedo a desplazarse, que con otro tipo de métodos. Otro tipo de algoritmos que también fueron ampliamente estudiados durante la época fueron los «actor-crítico», que consiste en incorporar otro agente cuyo objetivo es evaluar el desempeño del agente que busca maximizar el premio, con una clara inspiración en el surgimiento de las redes neuronales adversarias (Arulkumaran y otros, 2017). En (Konda & Tsitsiklis, 2003), los autores muestran como los «agentes críticos» ayudan a converger a los «agentes actores» a la solución óptima, guiándolos con su crítica hacia la dirección del gradiente dentro del espacio de posibles estados.

Con la llegada del aprendizaje profundo, el impacto en el aprendizaje por refuerzo fue significativo, igual que en otras ramas del aprendizaje de máquina. El incremento de la capacidad de procesar altas dimensionalidades por parte de redes neuronales cada vez más complejas, con más capas y diferentes tipos de funciones activadoras contribuyó, igualmente, a la experimentación y resolución de problemas más complejos, y con cada vez mejor nivel de abstracción (Koutník y otros, 2013).

En el caso del aprendizaje por refuerzo, la llegada del aprendizaje profundo implicó una mejora sustancial de los algoritmos mencionados anteriormente, mientras que la popularización de la nube permitió incrementar la velocidad de cómputo e impulsar los trabajos con grandes volúmenes de información (LeCun y otros, 2015).

Trabajos como los de van Hasselt y otros (2015) y Gu y otros (2016) muestran como el uso de aprendizaje profundo mejora la convergencia de los algoritmos de Q-Learning tradicionales, además de una mejora importante en velocidad de cómputo que permite resolver problemas con mayor dimensionalidad, conociéndose así los nuevos algoritmos de Q-Learning profundo. De la misma manera sucede con otros algoritmos antes mencionados, mientras que al tiempo nuevos aportes, como el transfer-learning, entre otros, continúan acelerando la ola del aprendizaje profundo, y, en específico, el aprendizaje por refuerzo profundo (Wang y otros, 2022).

**2.3 Aprendizaje por Refuerzo en Mercados Financieros**

Los algoritmos Aprendizaje Reforzado comenzaron a ser utilizados para aplicaciones de negociación de los mercados financieros alrededor de la década de los noventa junto con toda la ola del aprendizaje de máquina de aquella época. Moody y Saffel (1998) muestran cómo un algoritmo Aprendizaje por Refuerzo Recurrente puede ser entrenado para el comercio de carteras de acciones, mientras que la optimización de la ratio de Sharpe, que es una medida de la relación de la rentabilidad total obtenida por asumir una cantidad de riesgo de mercado.

En la última década, con la aparición y auge del aprendizaje profundo, aparecieron también nuevos enfoques aprendizaje por refuerzo profundo (Sutton & Barto, 2018), con nuevas aplicaciones en diferentes mercados, desde el mercado mayorista energético en (Tao & Wencong, 2018), el mercado de divisas en (Carapuco y otros, 2018), o el mercado accionario, en donde los autores escribieron especialmente sobre aplicaciones bursátiles con diferentes enfoques -como de Q-Learning profundo en (Carta y otros, 2021)-, y estrategias adaptativas de negociación de acciones fueron el centro estudios como (Wu y otros, 2020). En todos los casos el rendimiento mejoró con respecto a los métodos de aprendizaje por refuerzo y redes neuronales básicas y los métodos convencionales de comprar y mantener.

Yang y otros (2020), específicamente propusieron un conjunto de diferentes algoritmos de aprendizaje por refuerzo basados en la arquitectura «actor-crítico», como Advantage Actor Critic A2C, Proximal Policy Optimization PPO y Deep Deterministic Policy Gradient DDPG, para la negociación de acciones y la administración de una cartera de renta variable. La estrategia ensamblada funcionó y obtuvo buenos resultados incluso durante la crisis del covid-19.

Por otra parte, el aprendizaje automático para aplicaciones de renta fija nunca se ha centrado en el comercio o la gestión y optimización de carteras (Dixon y otros, 2020). La atención se centró en el modelado de la curva de rendimiento, la predicción de su forma, sus movimientos y, en algunos casos, la próxima crisis financiera, utilizando máquinas de soporte vectorial -SVM- y análisis de componentes principales -PCA-, entre otros métodos (Gogas y otros, 2015). El aprendizaje por refuerzo no ha sido estudiado en profundidad bajo la óptica de la Renta Fija, por ello, en su tesis doctoral Nunes (2022) realiza un diagnóstico de dicho vacío en la literatura y se dispone a proponer diferentes algoritmos, encontrando que los algoritmos DDPG tenían un mejor rendimiento en la negociación de ETFs de renta fija. Sin embargo, los ETF, pese a poder tener como subyacente uno o varios instrumentos de renta fija, pueden entenderse como instrumentos de renta variable, por lo que valdría la pena revisar si estos algoritmos de aprendizaje por refuerzo profundo pueden utilizarse directamente sobre los subyacentes de renta fija.

**3. MARCO TEÓRICO**

**3.1 Aprendizaje por Refuerzo**

El Aprendizaje por Refuerzo es una rama del aprendizaje automático donde un agente inteligente aprende cómo actuar dentro de un entorno, buscando maximizar las recompensas a largo plazo dadas por un intérprete en función de ciertos objetivos de rendimiento definidos previamente (Wang y otros, 2022).

En la Figura 1 se muestra un esquema general de cómo funciona el aprendizaje por refuerzo. En la primera iteración, el agente inteligente realiza algunas acciones aleatorias dentro del entorno. Estos entornos aleatorios, incluyendo el comportamiento de los mercados financieros, por lo general pueden ser modelados como Procesos de Decisión de Markov (PDM) (Sutton & Barto, 2018). Los resultados de esa iteración serán una observación del intérprete, quien, dependiendo de los objetivos de optimización definidos previamente de esta observación, otorgará una recompensa al agente por el buen o mal desempeño que obtuvo tras tomar esas acciones. El agente aprenderá de las observaciones y recompensas anteriores y aplicará ese conocimiento en futuras iteraciones dentro del entorno, buscando maximizar la cantidad de recompensa que recibe del intérprete.

Figura 1. Diagrama de flujo típico de un algoritmo de aprendizaje por refuerzo



Los primeros algoritmos de RL fueron entrenados para resolver problemas en entornos de baja dimensión (Sutton & Barto, 2018). Sin embargo, con los años aparecieron problemas de mayor envergadura, y con la aparición de las redes neuronales profundas, los algoritmos RL comienzan a ser más complejos, eficientes y útiles para resolver los problemas más complejos y de mayor envergadura, dando cabida a los algoritmos de Aprendizaje por Refuerzo Profundo (Arulkumaran y otros, 2017).

Los siguientes conceptos son importantes para una buena comprensión del aprendizaje por refuerzo:

**A. Procesos de Decisión de Markov (PDM):**

Es un marco común para resolver problemas de aprendizaje por refuerzo que consiste en algunos supuestos, como, por ejemplo, que el entorno es markoviano y observable (Sutton & Barto, 2018) o parcialmente observable (Sutton y otros, 1999). Bajo esta premisa, el agente tendría que ser capaz de observar el entorno y luego, tomar decisiones dentro de este.

Un algoritmo aprendizaje por refuerzo dentro de un PDM intenta encontrar las trayectorias para el agente dentro del entorno markoviano que maximizan la recompensa utilizando los siguientes parámetros (Yang y otros, 2020) (Wang y otros, 2022):

* Un estado s en el que se encuentra el agente, y que pertenece a un set de posibles estados S. El estado inicial es .
* Una acción a que el agente toma en determinado estado s, y que pertenece a un set de posibles acciones A.
* La recompensa inmediata ρ que el agente recibe por tomar una acción a en determinado estado s, llegando así a un estado nuevo s'.
* Una política π, que resulta de la distribución de probabilidad de tomar las acciones A encontrándose en determinado estado s.
* Una recompensa esperada Q de tomar acciones en un estado específico s y siguiendo una política π. Este concepto proviene del Q-Learning (Watkins C. J., 1989).
* Una función de transición de estado f, dada por la probabilidad de llegar al estado s' a partir del estado s por el hecho de tomar una acción a.
* Un factor de descuento γ que reduce el impacto de acciones futuras en el presente A.

**B. Ecuación de Bellman y Q-Learning:**

Dado el número de trayectorias, la política y los diferentes estados a los que se puede enfrentar el agente dentro del entorno, entonces es necesario calcular la recompensa esperada del agente por encontrarse en cierto estado, por lo que Bellman (1966) propone una Función de Valor del Estado **,** mejorada en (Sutton, 1984) que a través de recursividad permite encontrar una valor de recompensa para el estado actual teniendo en cuenta los posibles estados futuros traídos a valor presente con el factor de descuento **γ** planteado, tal como se muestra en la ecuación (1).

*.* (1)

En la misma línea, Watkins (1989) posteriormente argumenta que el valor de la recompensa para el agente no debe obedecer únicamente al estado actual y estados futuros, sino que, además, debe depender de las acciones que el agente toma para llegar a diferentes estados, de manera que se plantea mapear todas estas posibles acciones basadas en la política que sigue el agente. Es entonces así, como la recompensa media ponderada de todas las posibles trayectorias individuales de una acción ***a*** partiendo de un estado ***s*** es la Función de Valor Estado-Acción , ecuación (2) (Watkins & Dayan, 1992).

*.* (2)

La Función de Valor Estado-Acción ayuda a mapear acciones futuras ligadas a un estado futuro relacionado, permitiendo así una convergencia más rápida y precisa en cantidad de instancias, pero un poco más exigente respecto a la capacidad de cómputo (Sutton & Barto, 2018).

**C. Métodos de Aprendizaje por Refuerzo Profundo:**

En el marco de la aparición de las redes neuronales profundas como solución para problemas de alta dimensionalidad, los estudios de aprendizaje por refuerzo se volcaron a la utilización de redes neuronales convolucionales como componentes de los agentes (Arulkumaran y otros, 2017), y otras soluciones comenzaron a ser planteadas con el objetivo de lograr eficiencias necesarias en este tipo de entornos de grandes dimensiones, la gran mayoría siendo PDM parcialmente observables. En el desarrollo de este trabajo se utilizarán tres métodos específicos a describir a continuación y métodos de ensamble que involucran estos métodos.

* Advantage Actor-Critic (A2C)

Como los métodos Actor-Crítico típicos (Konda & Tsitsiklis, 1999), A2C utiliza un agente adicional, el crítico, que se encarga de revisar si la política seguida por el agente que la ejecuta, el actor. Ambos agentes actúan de manera independiente.

Particularmente, un algoritmo de A2C entrega al agente crítico el resultado de la función de ventaja, ecuación (3), dada por la diferencia entre la función de Valor Estado-Acción, y la función de Valor del Estado (Yang y otros, 2020), lo que se interpreta como la ventaja de usar una política **π** sobre la Función de Valor del Estado por si sola.

(3)

* Deep Deterministic Policy Gradient (DDPG)

Los algoritmos de gradiente de política son ampliamente utilizados en aprendizaje por refuerzo para problemas en donde las acciones que puede tomar el agente actor se pueden describir con una función de probabilidad continua. A partir de allí, estos algoritmos mapean la función , y a través de la optimización con gradiente, buscan minimizar el error total o maximizar la recompensa esperada (Silver y otros, 2014).

DDPG, es un algoritmo de gradiente de política que no utiliza funciones de probabilidad sobre variables continuas. En cambio, también permite optimizar el gradiente de política, ecuación (4), teniendo en cuenta un número de acciones posibles finitas y determinísticas para una función objetivo de retorno ***J***, ecuación (5) (Li y otros, 2019), lo que representa una ventaja, específicamente, para los problemas relacionados con trading, debido a que las acciones que el agente actor puede tomar cumplen esa condición determinística: Compra, venta, mantener la posición, entre otras que puedan plantearse (Yang y otros, 2020).

* Proximal Policy Optimization (PPO):

(4)

(5)

(6)

**3. Models and Methodology**

If the liquidity preference theory holds, the expected value of the spot interest rate for a term , negotiated in , will be less than the forward rate negotiated in time for the same periods of time. That is

. (1)

For the Colombian case, [Rey (](#_bookmark23)2005[)](#_bookmark23) and Rueda [and Arango (](#_bookmark24)2008[)](#_bookmark24) found evidence in favor of the liquidity preference theory in this market. This means that the yield curve will have a positive slope and the forward rate is not an unbiased estimate of the future real interest rate, and therefore, under the liquidity preference theory it is possible to execute arbitrage strategies (see e.g., Ang, 1998; Cox & Felton, 1994; Galvani & Landon, 2013; Grieves & Marcus, 1992; Mercer et al., 2009; Pantalone & Platt, 1994; Pelaez, 1997). A simple strategy is to trade a short position at , a long position at , then sell the long position at , and finally close the short position in the period .[[4]](#footnote-4) Therefore, the profit and loss (P&L) of the long position will be represented by , and the short position by . Thus, the final P&L of the strategy will be given by the result of the following expression.

. (2)

Since

, (3)

the equation (2) can be rewritten as

. (4)

By simplifying Equation (4), and assuming that the interest rates are continuous compounding, the P&L in terms of relative change can be expressed as

. (5)

Using the latter equation, we calculate the relative change of the profit and losses, or simply the returns, of the strategy for the periods {90,90; 180,180; 360,360; 720,360; 1080,360; 1440,360; 1800,360; 2160,360; 2520,360; 2880,360; 3240,360}, for the analyzed 04/28/2006 - 02/22/2019 period. For instance, if a strategy is executed for the periods {1080,360}, assuming , a short position is traded today at 1080 days with rate *y*0*,*1080, a long position with *y*0*,*2160, then sell the long position with *y*1080*,*360, and close the short position at time *t* = 1080. Table 1 shows the main descriptive statistics for the strategy returns, which is expressed in annual continuous compounding.

### 

### Table 1. Main descriptive statistics of strategy returns

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 90 - 90 | 180 - 180 | 360 - 360 | 720 – 360 | 1080 - 360 | 1440 - 360 | 1800 - 360 | 2160 - 360 | 2520 - 360 | 2880 - 360 | 3240 - 360 |
| Min | -2.76% | -2.45% | -2.19% | -1.13% | -0.85% | -0.14% | -9.62% | -5.75% | 0.07% | -0.04% | -0.02% |
| Max | 3.44% | 4.95% | 9.35% | 7.63% | 7.69% | 7.21% | 4.97% | 9.93% | 9.42% | 3.25% | 4.28% |
| Average | 0.08% | 0.45% | 1.30% | 1.48% | 1.94% | 1.75% | 0.76% | 0.07% | 2.47% | 1.17% | 1.18% |
| P10 | -0.72% | -0.73% | -0.78% | -0.29% | 0.59% | 0.42% | -0.18% | -1.40% | 1.55% | 0.76% | 0.56% |
| P20 | -0.43% | -0.28% | -0.13% | 0.07% | 1.00% | 1.04% | -0.06% | -1.00% | 1.75% | 0.94% | 0.65% |
| P30 | -0.21% | -0.08% | 0.40% | 0.57% | 1.15% | 1.33% | 0.03% | -0.52% | 1.89% | 1.03% | 0.81% |
| P40 | -0.06% | 0.12% | 0.70% | 0.97% | 1.52% | 1.55% | 0.17% | -0.33% | 2.06% | 1.11% | 0.94% |
| P50 | 0.07% | 0.35% | 1.03% | 1.35% | 1.97% | 1.73% | 0.50% | -0.17% | 2.24% | 1.17% | 1.06% |
| P60 | 0.18% | 0.55% | 1.29% | 1.62% | 2.42% | 1.89% | 0.94% | 0.12% | 2.51% | 1.23% | 1.27% |
| P70 | 0.31% | 0.74% | 1.62% | 1.94% | 2.71% | 2.15% | 1.24% | 0.77% | 2.86% | 1.29% | 1.42% |
| P80 | 0.45% | 1.02% | 2.27% | 2.42% | 2.94% | 2.53% | 1.58% | 1.31% | 3.30% | 1.37% | 1.54% |
| P90 | 0.75% | 1.45% | 3.61% | 3.73% | 3.33% | 2.92% | 2.10% | 1.69% | 3.65% | 1.55% | 1.95% |
| N | 3343 | 3278 | 3147 | 2886 | 2625 | 2363 | 2103 | 1843 | 1582 | 1320 | 1059 |

Min = minimum, Max = maximum, P means percentile (e.g., P10 is the 10th percentile), and N is the count of the strategies in each period. The number of datapoints (N) change depending of the period of the strategy. For instance, the period {360, 360} uses the data from a specific date to a year later, while period {90,90} uses the data 90 days later. The difference between the number of datapoints between a year and a quarter, corresponding to period {360, 360} and {90, 90} is 196 (=3346−3150). Thus, as increases, the number of variables available for prediction decreases.

As observed in Table 1, the median (i.e., the 50th percentile) of the strategy returns is positive for all periods (except the period {2160, 360}), but the average return is positive for all cases. This means that the robust central tendency measure of the returns is positive for most of the periods. It is worth mentioning that for the {2520, 360} period, it is the only case where the minimum return is positive (0.07%). In addition, the second highest maximum (9.42%) is presented in this period, whereas the highest maximum (9.93%) is exhibited in the {2160, 360} period.

# Prediction of the returns for each strategy

In this paper, we use different forecast approaches to predict the profitability of the proposed strategies. In the first place, we apply supervised regression methods. We use classics models for time series such as AR and ARIMA and after that we employ nonparametric Machine Learning methods such as random forest algorithms to compare the model assessment of these models, considering the nature of the data of interest. Subsequently, we implement classification algorithms to expand the performance of the proposed models. Finally, we compare the model assessment of all models by the means of various metrics of goodness of fit and the Diebold-Mariano test.

* + 1. **Data description and exploration**

The data source is the spot rates of Colombian government bonds daily between 26/7/2006 to 22/2/20019. It is important to note that we also explored the contribution of other variables such as implicit inflation, market variables such as the stock market (Colcap index), and the COP-USD exchange rate. All these variables are available on the Bloomberg platform.

**Gráfico, Gráfico de líneas

Descripción generada automáticamenteFigure 1.** Yield Curves’ shapes for the period of analysis

To analyze and describe the statistical and economic properties of the yield curves, we plotted them for different points in time after cleaning and pre-processing the dataset.

Figures 1 and 2 show the pattern in time followed by the curves from 2006 until 2018, as well as the yields per maturity and the yield spreads to three months, respectively. From these figures, we can conclude that the sovereign yield curves present a decline in level for the time of analysis, as well as a gain steepness, especially for the case of the curves of 2012 and 2014. In addition, it is important to note that the yields of maturities of 5 years are higher than the rest consistently, during the whole period, and the yields for the maturities of 7 years were lower. Finally, regarding the yield spreads it can be observed a correlation between the periods of higher spread with the international and national financial crises of 2008, 2012, and 2016.

**Figure 2.** Yields per Maturity and yield spreads to 3M

Interfaz de usuario gráfica, Gráfico, Histograma

Descripción generada automáticamente

To complement our analysis of the drivers of each point on the curve, we include a graphic of the volatility over 60 days. Accordingly, from the last figure, it can be observed that the periods of more volatility correspond to the international crises of 2008, 2014, and 2016.

**Figure** Interfaz de usuario gráfica, Gráfico, Histograma

Descripción generada automáticamente**3.** Rolling volatility of the yield curves

**Reduction of dimensionality by applying Principal Component Analysis (PCA)**

Considering the objective of prediction to maximize the profit of the trading strategies explained in the methodology section, we decide to reduce the dimensionality of our variables by employing principal component analysis techniques. This type of dimensionality reduction contributes to filtering and reducing data's noise, consequently improving the model’s performance.

On the other hand, as suggested by the literature, applying the yield curve decomposition into its main drivers, we could also delve into the “underlying dynamics” of the yield curve under analysis.

To implement the PCA analysis, we construct our characteristics matrix X with features corresponding to the daily yields for every given maturity. Then, we derived the eigenvectors from the covariance matrix of X by minimizing the distances generated by the projections onto the vector itself. This process guarantees that we can capture the maximum variability of all maturities. Since we are only interested in variance, we centered each of the variables in X to have a mean zero.

Even though the process of the eigenvalue decomposition obtains the same number of vectors as the initial characteristic matrix, we only retained the three most important ones, as these represent 99.72% of the variance explained. As a consequence of this significant quantity of explained variance, the yield curve movements can be approximated by linear combinations of the first three loadings with small relative error, as we will demonstrate in the following sections.

Next, we present in figure 4 the contribution in terms of percentage to the explained variance of the eigenvectors obtained by the PCA process, and in figure 5 the patrons followed by the first three eigenvectors through the maturities. According to the economic meaning proposed by the literature, the first component represents the shifts of the yield curve (level), the second component constitutes tilting of the yield curve (slope), and the third component acts for the curvature of the yield curve.

**Figure 4.** Percentage ofvariance explained by the Eigenvalues

Gráfico

Descripción generada automáticamente

**Figure 5.** Behavior of the three principal components obtained by PCA

Gráfico, Gráfico de líneas

Descripción generada automáticamente

Lastly, regarding the interpretability of the PCA results, we plot the Eigen-scores, which also can be compared to the traditional factors: “level”, “slope” and “curvature”. The higher scores for the first component are those associated with years 2006, 2007, 2008, and 2009, and the higher scores for the second component are those with relatively recent years such as 2015, 2014, 2013, and 2016.

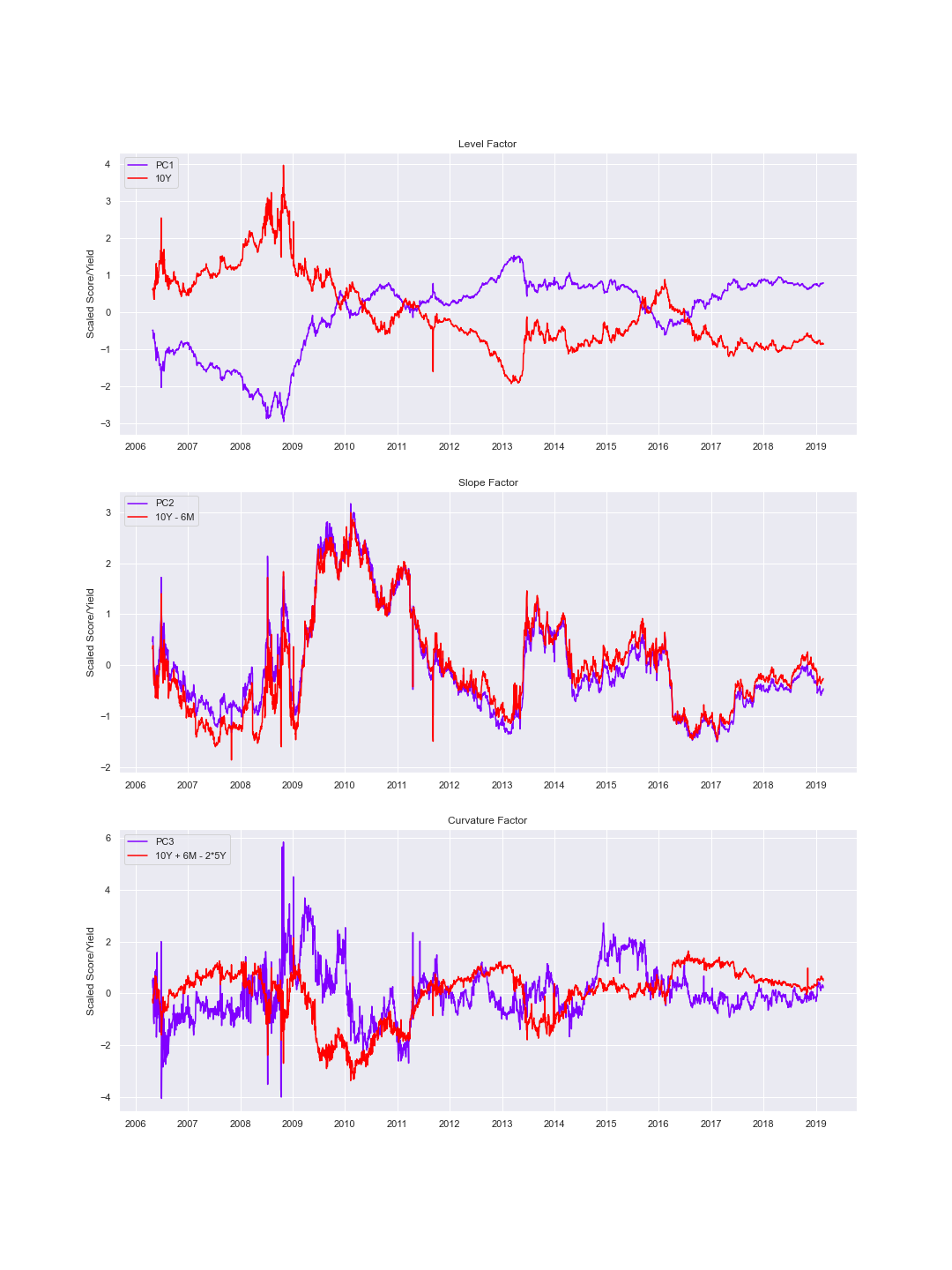
**Figure 6.** Visual representation of the PCA’s scores

Gráfico, Gráfico de dispersión

Descripción generada automáticamente

In the literature, these factors are usually understood as a proxy of the spread between maturities, and that is why we examine the tendencies followed by the 10Y-6M spread as a proxy for steepness.

**Figure 7.** Comparation of the patrons followed by each of the three components vs. its references



Now, for the sake of the evaluation of the goodness of fit of the model, we will compare the yield curve obtained by the reverse transformation of the derived scores of the PCA process to the realized curves. The next figures present the comparison mentioned for a specific day, helping to evaluate the goodness of fit of the PCA process applied.

Gráfico, Gráfico de líneas

Descripción generada automáticamente**Figure 8.** Yield curve comparison for a specific day

Next, we also assess the predictive power out-of-sample of the underlying components by computing the metric of RMSE for the whole-time horizon. According to the figure, it exhibits very good performance.

**Figure 9.** Goodness of fit in sample and out-of-sample of the PCA method

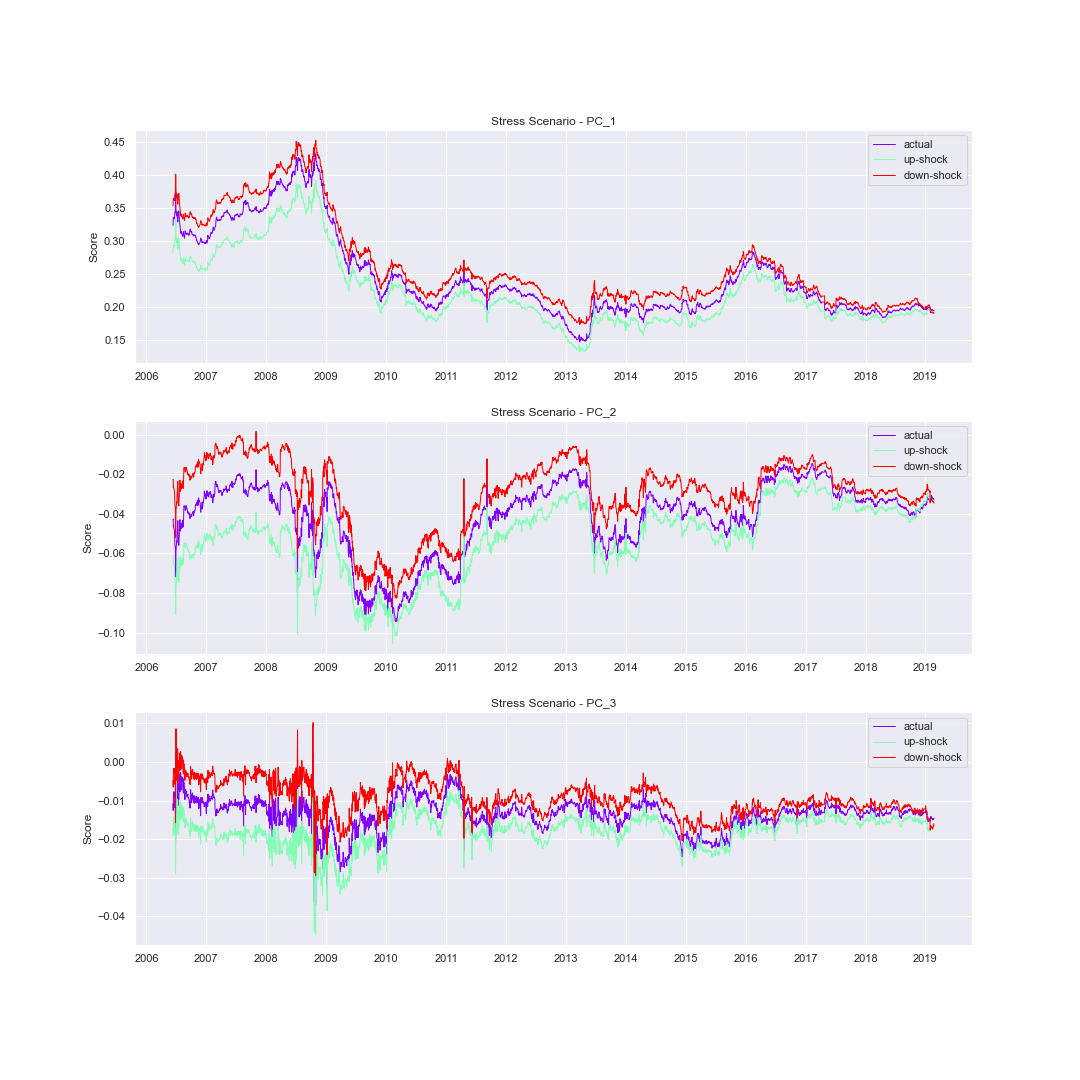
Gráfico, Histograma

Descripción generada automáticamente

**Testing in stressing scenarios**

So far, we have demonstrated the three components extracted from the spot sovereign yield curve seize its variability solidly and consistently. Nevertheless, it is also necessary to test this technique in realistic non-linear stress scenarios. As is often done in financial analysis, we can generate parallel upward and downward shocks by applying the concept of historical Value at Risk. Specifically, we will construct a 95% confidence interval considering the 5% largest deviations within a rolling time window.

**Figure 10.** Behavior of each of the components in stress scenarios



The pattern followed by the actual yield curve never crosses out of the confidence interval of 95% for each case (PC1, PC2, and PC3), indicating a consistency in the economic interpretation of each of the factors derived for stressed scenarios according to a VaR of 5%.

**Figure 11.** Behavior of the yield curve reconstructed from the main components in stress scenarios.

Gráfico, Gráfico de líneas

Descripción generada automáticamente

**Regression models for prediction**

As the task of prediction is not only our main concern in this work but also interpretation (considering the interest on formulate trading strategies), we propose to apply for the sake of comparing different types of models from parametric such as Naïve, AR, ARIMA to more complex and non-parametric like Decision trees and Random Forest models. It is important to note that in all of these cases, we take as an input the three principal component scores, which grant us, as we demonstrated in the last section, to fit our models with filtered and informative data.

In the following sections, we present each of the notice models, its specification, their results, and the metrics of the goodness of fit. Finally, we compare the performance of all these models using the test Diebold-Mariano, to establish if there is a statistical difference between them.

Before the stages of modeling and prediction, we also present the process of time series analysis to investigate the presence of the most common time series components, which could prevent them to be stationary in the weak sense. In that regard, we check visually for tendencies, seasonalities, cycles, and irregularities, and after that, we utilize the Augmented Dickey fuller test to verify the stationarity for each of the series analyzed.

**Stationarity analysis of the series**

As we stated before, to perform our tasks of modeling and prediction for time series data these must be stationary in mean, variance, and covariance. In that sense, we visualize the patterns of the PCA scores and afterward we check for stationarity as can be seen in the following figures:

**Figure 12.** Visualization of the time series for each of the components

Gráfico, Histograma

Descripción generada automáticamente

Stationarity-Test: PC\_1

{'adf\_stat': -1.549, 'p\_val': 0.5092, 'threshold': -2.8624, 'stationary': 'no'}

Stationarity-Test: PC\_2

{'adf\_stat': -2.556, 'p\_val': 0.1024, 'threshold': -2.8624, 'stationary': 'no'}

Stationarity-Test: PC\_3

{'adf\_stat': -3.5859, 'p\_val': 0.006, 'threshold': -2.8624, 'stationary': 'yes'}

According to the pattern presented in the graph, it can be concluded that the first component presents more marked trends and irregularities than the rest of the components. On the other hand, the third component seems, at first glance, to behave in a stationary manner in mean, variance, and autocorrelation. These observations are corroborated by the results obtained with the "Augmented Dickey-Fuller" test, for which the null hypothesis of unit roots is not rejected in the case of the first two components, while it is rejected in the case of the third component.

Considering the analysis of the results obtained, we proceed to carry out the respective transformations. Following the recommendations of the literature, we apply differences, and calculate again the Augmented Dickey-Fuller test, as can be seen below.

**Figure 13.** Visualization of the transformed time series for each of the components

Gráfico

Descripción generada automáticamente

Stationarity-Test: PC\_1\_diff

{'adf\_stat': -9.9636, 'p\_val': 0.0, 'threshold': -2.8624, 'stationary': 'yes'}

Stationarity-Test: PC\_2\_diff

{'adf\_stat': -13.1447, 'p\_val': 0.0, 'threshold': -2.8624, 'stationary': 'yes'}

Stationarity-Test: PC\_3\_diff

{'adf\_stat': -11.2339, 'p\_val': 0.0, 'threshold': -2.8624, 'stationary': 'yes'}

Both the graph and the test show that the applied transformations were effective in stationary all the series.

**AR Model**

After stationarizing the series, we proceed, first, to run the linear models AR(p) and ARIMA(p,d,q) for each of the vectors of the scores corresponding to the first three components found using the PCA method.

The general specification for AR(p) models is as follows:

Through this model, we assume that the current value of each of the series can be explained by their lags. Specifically, in our specific case, we will verify the predictive capacity of the model utilizing the first 5 lags (based on the lags indicated in the autocorrelation graphs, ACF, and the partial autocorrelation PACF). After that, we will select the best model through a more general specification such as the ARIMA model. The results obtained through the AR models can be seen below:

**Table 2.** Estimation of an AR model for each of the components

Imagen de la pantalla de un celular con letras

Descripción generada automáticamente con confianza media

Tabla

Descripción generada automáticamente

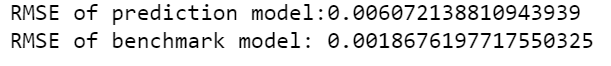
Tabla

Descripción generada automáticamente

The modeling results show a good fit in general for the three variables (the first differentiated component PC1\_diff, the second differentiated component PC2\_diff, and the third differentiated component), identifying practically all the lags proposed in each case as significant.

Now, to carry out and evaluate the prediction of the proposed models, it is necessary to calculate the inverse transformation of the scores of the components obtained (by multiplying the Eigen scores and the inverse of the eigenvectors). In such a way we obtain a matrix of the original units that represent the yield curves generated from the 3 main components.

Additionally, we compare the prediction obtained through the AR models with the prediction obtained through a model called Naive, which consists of predicting the t+1 value through the immediately previous value.



When comparing the RMSE out-of-sample values for both models, it is clear that the Naive model presents better performance. However, it is also important to establish if there is a significant difference between both models, for this we will implement the Diebold-Mariano test, not only for the mentioned models but also for the rest of the proposed models.

**ARIMA model**

Next, an ARIMA model was estimated for each of the components obtained. The selection of the model specification was carried out using the pmdarima.arima.auto\_arima package. This Python package allows an optimized search of the parameters (p,d,q), based on different criteria such as Kwiatkowski–Phillips–Schmidt–Shin, Augmented Dickey-Fuller or Phillips–Perron.

The specification and estimation results of the models implemented for components one, two and three differentiated (PC\_1\_diff, PC\_2\_diff, PC\_3\_diff) can be seen in the following tables.

The optimal specification for the data corresponding to the first component turned out to be ARIMA(3,0,4), for the second component differentiated ARIMA(2,0,1), and for the third component differentiated ARIMA(2,0,2).

**Table 3.** Estimation of an ARIMA model for the first component differentiated

Tabla

Descripción generada automáticamente

**Table 4.** Estimation of an ARIMA model for the second component differentiated

Tabla

Descripción generada automáticamente

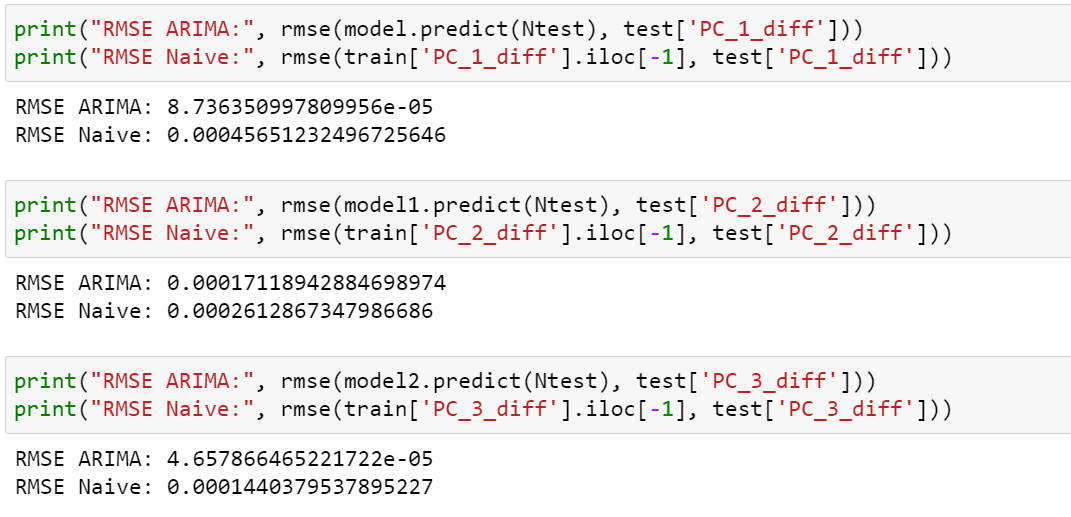
**Table 4.** Estimation of an ARIMA model for the third component differentiated

Tabla

Descripción generada automáticamente

The adjustment and comparison of the prediction for each of these models can be seen in figure 14, as well as in table 4. According to the patterns presented in the graphs, where the predicted values are very close to the real values, and with the RSME values, it is clear that the prediction capacity of the ARIMA models is much higher than that of the Naive models, confirming once plus the applicability of liquidity theory.

**Table 5.** Comparison of the root mean square error metric between the estimated models and the naive model



**Figure 14.** Adjustment level of predictions for the last five days for each of components for ARIMA models

**Gráfico, Histograma

Descripción generada automáticamente**

**Gráfico, Gráfico de líneas

Descripción generada automáticamente**

**Gráfico, Histograma

Descripción generada automáticamente**

**Random Forest Model**

Finally, in order to once again improve the prediction, trying not to overfit the model, a Random Forest was estimated. This method is characterized by using a combination of weak predictors (decision trees), sampled by bootstrap, averaging them and obtaining more accurate out-of-sample predictions than they would have using a single tree, considering their ability to reduce variance (by averaging the variance of the trees that constitute it).

Although according to the results obtained so far, everything seems to indicate that less flexible models such as linear models fit the data very well, we still wanted to experiment with non-parametric, non-linear models for comparison purposes. The results obtained can be seen in the following table.

After an exhaustive exploration of the different possibilities in terms of hyperparameters using the grid search method, the following model specification has arrived: lags =1, max\_deep= 3, and n\_estimators=100. The figures corresponding to the predictions of the last 5 days are shown below.

**Figure 15.** Adjustment level of predictions for the last five days for each of components for Random Forest models

Gráfico, Gráfico de líneas

Descripción generada automáticamente

Gráfico, Gráfico de líneas

Descripción generada automáticamente

Gráfico, Gráfico de líneas

Descripción generada automáticamente

The results show a decrease in the quality of the predictions, perhaps due to overfitting of the model. These results can be corroborated by the goodness-of-fit metrics, whose values for the first, second, and third components, respectively are shown below.





**Method Comparison for the regression models**

To finish the modeling and prediction stage, we calculate the Diebold-Mariano test in order to determine if there is a significant difference in terms of prediction between all the implemented methods. Additionally, we include the graphs corresponding to the prediction of the yield curves for the 5 days following the last date on which information is available.

Tabla

Descripción generada automáticamente con confianza baja

Texto

Descripción generada automáticamente con confianza baja

Imagen que contiene Texto

Descripción generada automáticamente

According to the results obtained there is no statistically significant difference between the estimated models, the null hypothesis is that there is no statistically significant difference between the predictions and the p-value located on the right side.

## Classification models

### 

### Toy example

As an example, we use a part of our data to obtain a deeper understanding of our method. To this end, we use the profits and losses for the {90,90} period ranging between 07/26/2006 and 08/02/2006 as our dependent variables (also known as labels). As is usual in decision tree applications, a simple transformation is performed. That is, a negative return is labeled with zero and a positive return is categorized with one. Finally, the zero-coupon yield curve rates for nodes 180 and 360 days are employed as the features. Table 6 represents the values for the toy example.

### Table 6. Values for the toy example

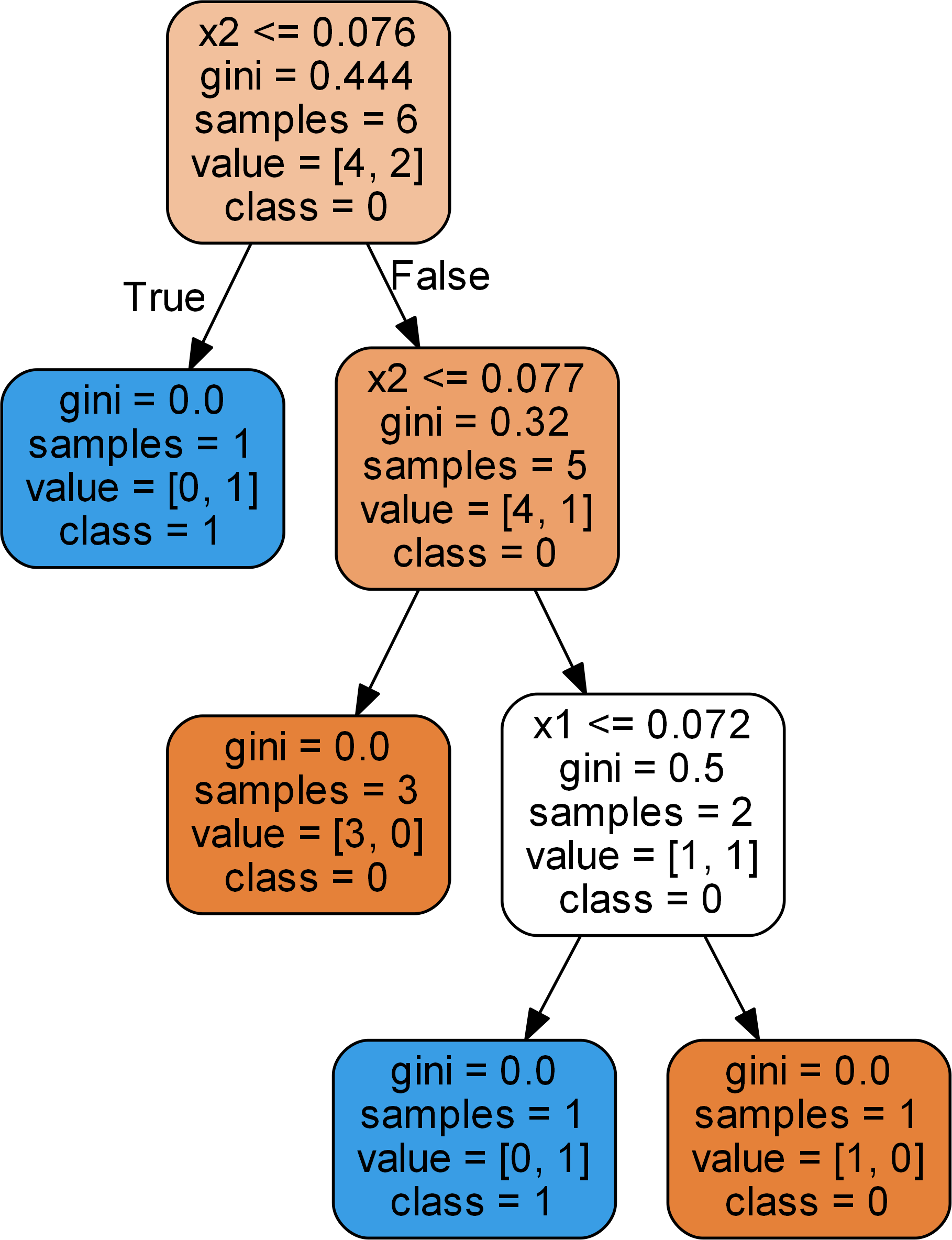
|  |  |  |  |
| --- | --- | --- | --- |
| **Return {90,90}** | **Transformation {90,90}** | **node 180** | **node 360** |
| -0.34% | 0 | 0.0733 | 0.0770 |
| -0.11% | 0 | 0.0717 | 0.0759 |
| 0.01% | 1 | 0.0712 | 0.0757 |
| -0.26% | 0 | 0.0710 | 0.0767 |
| 0.16% | 1 | 0.0710 | 0.0765 |
| 0.30% | 1 | 0.0711 | 0.0766 |

As can be inferred from Table 2, this is a linearly inseparable problem, where a simple line would not be able to separate the datapoints. Therefore, the linear regression approach or support vector machine (with linear kernel) technique would not be suitable for this type of problems. The (classification) decision tree can be used to “completely” separate these datapoints since this tool essentially draws many repeated linear limits between the datapoints.

## The Single Decision Tree (the toy model)

In the training set, the (decision) tree will learn how to separate the datapoints by “building” a diagram of questions based on the values of the features. At each stage, the decision tree is split to minimize the Gini impurity index.[[5]](#footnote-5) Usually, the bigger the tree (i.e., large number of nodes), the lesser the Gini impurity. However, a problem of data overfitting in the training set may occur when minimizing the Gini impurity index, resulting in poor modeling for the test set. In fact, an overfitting drawback can be identified using a model with “very” high accuracy rates in the training set, and low levels of accuracy in the test set. In practice, it is common to limit the depth of the tree via cross-validation or “pruning the tree” techniques. For the case of the toy example, the tree has seven nodes and a depth level of three (see Figure 1). Moreover, the accuracy is equal to 1 and the Gini impurity index is equal to zero in the last two nodes of the tree. That is, from the six observations of the sample, all of them were perfectly classified.

**Figure 16.** The single classification tree for the toy example



The five rows in each node (except the final nodes) represents the following:

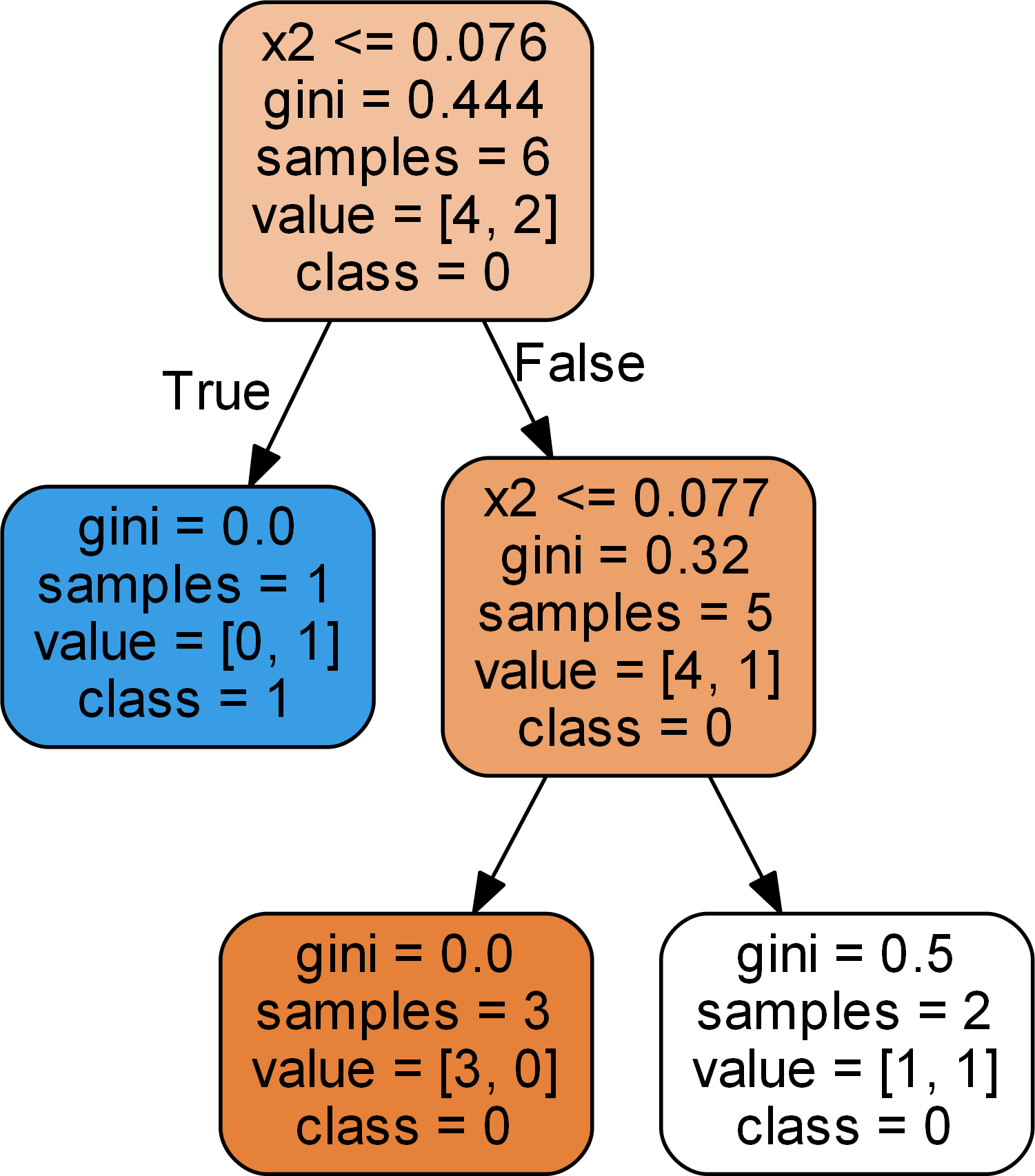
* Question asked about the features, in our case (the toy example) about the value of the interest rate in nodes 180 and 360.
* Gini: the node’s Gini Impurity index.
* Samples: the number of training observations in the node.
* Value: the number of samples in each class [0,1].
* Class: the predicted class for the entire sample of the node.

Except for the final nodes, the tree makes decisions about the class of the sample.

## Maximum Depth Limit (the toy example)

In our application, we will employ a strategy to set the size of the tree. For the toy case, we limit the maximum depth of the tree to two. In this case, the accuracy is equal to 0.67 since four of the six observations were adequately classified. Moreover, the Gini impurity index for the last (right) node is not equal to zero (it is 0.5). See Figure 2 for the results of this action.

**Figure 17.** The toy example with maximum depth equivalent to two of the classification tree



As observed in Figure 2, the tree is no more accurate in terms of the training data than it is in Figure 1, but it probably performs better on the testing set. This toy example highlights the bias − variance tradeoff, which is present in most of the machine learning models. A model with high variance learns the training data very well but it cannot perform the same with the observations on the testing set. On the other hand, a model with high bias has not learned the training data very well because of the lack of complexity, but it can generalize the results very well to the datapoints in the testing set. Our simple example shows that the bias can be improved by limiting the depth of the tree. However, another alternative is to employ an entire forest of trees. The idea is to train each tree with different subsamples of the training data and finally “averaging” each individual tree to obtain a classification or numeric prediction. In the next section, we present the results with the whole sample.

1. **Empirical Results**

This section presents the results of the machine learning techniques (decision trees and random forest) applied to the {90,90} period, which is the more balanced case for positive and negative returns. In our application, we employ three different measures: recall, precision, and AUC. The recall (also known as true positive rate) is calculated as

, (6)

where TP stands for true positives and FN the false negatives. Whereas precision is assessed as

, (7)

where FP stands for false positives. Finally, AUC is the area under the ROC curve, where ROC stands for receiver operating characteristic and it is a graph where the y-axis is the true positive rate and the x-axis is the false positive rate. Thus, the closer the ROC curve and AUC to one, the better. This is also true for the recall and precision measures.

These measures are calculated for the training set (74% of the data), test set (26% of the data), and the baseline case. The baseline case acts as the benchmark case and it represents the 45° diagonal in the ROC curve (see Figure 3), thus, the AUC for the baseline is always 0.5, and a curve above this diagonal is considered to be a good model.

* 1. **Decision tree**

This section presents the riding the yield curve strategy for period {90,90}. In the first place, we implemented the algorithm without restricting the depth of the tree (there were 299 nodes with a maximum depth of 15). Subsequently, we tuned the hyperparameters, including the maximum depth, obtaining the best model in terms of prediction. In order to validate the performance, we calculated the metrics shown in Table 3 and Figure 3.

### Table 7. Results for the strategy for period {90,90} using the decision tree algorithm optimized with cross-validation

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Baseline** | **Test** | **Train** |
| **Recall** | 1 | 0.89 | 1 |
| **Precision** | 0.6 | 0.92 | 1 |
| **AUC** | 0.5 | 0.88 | 1 |

**Figure 3.** ROC Curve for results presented in Table 3

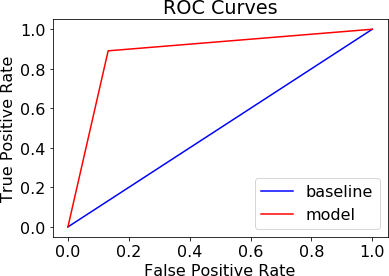
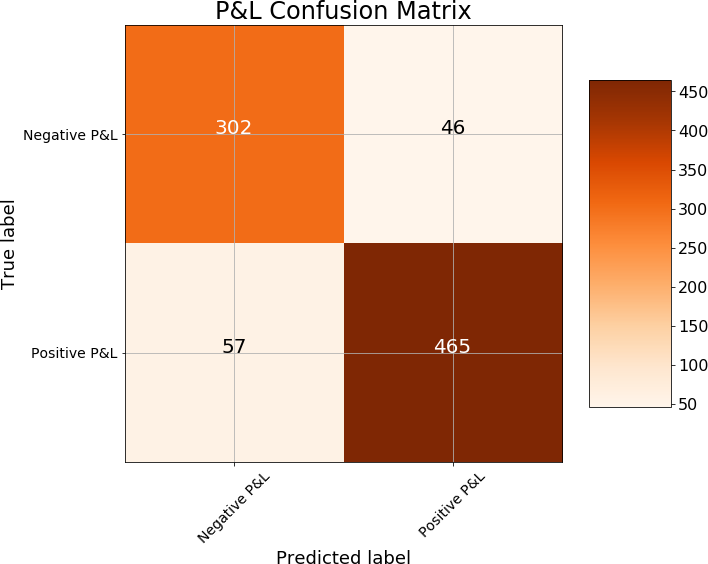


Figure 4 depicts the confusion matrix, where other measures can be calculated (namely, accuracy rate, error rate, specificity, among other indicators).

**Figure 18.** Confusion matrix for period {90,90}



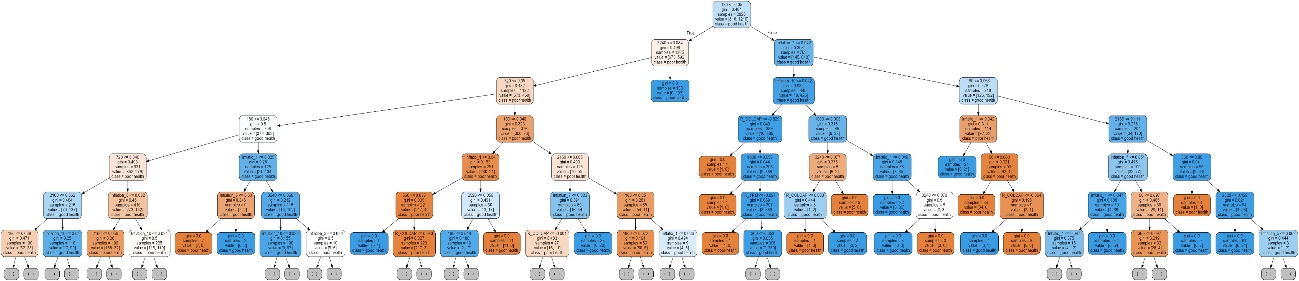
As observed in Figure 4, the number of errors in classifying observations that were positive when in fact they were negative is 46 (false negative). On the other hand, the number of errors in classifying datapoints that were negatives when in fact they were positive is 57 (false positive). Table 4 shows the most important features for the tree. The importance value is determined by the reduction in Gini impurity over all the nodes in which the feature is used.

### Table 8. Most important features for period {90,90} with classification tree

|  |  |
| --- | --- |
| **Feature** | **Importance** |
| **Node 180** | 0.289819 |
| **Node 3240** | 0.104953 |
| **1-year Inflation** | 0.093712 |
| **7-year Inflation** | 0.091384 |
| **Node 720** | 0.081656 |

The most important feature of this tree is the interest rate of node 180 in the zero-coupon yield curve since it is the variable that contributes the most to reducing Gini impurity. Figure 5 (the graph is attached as tree3.png for better visualization) depicts the classification tree for the analyzed case.

**Figure 19.** The classification tree for period {90,90}



## Random Forest

The random forest is a method that employs a certain number of individual trees. Each tree uses a random set of observations and a subset of the features are used for prediction. Table 5 and Figure 6 present the results of the random forest with 100 individual trees. With this setting, there are 274 nodes and a maximum depth of 15, on average.

### Table 9. Results of the random forest

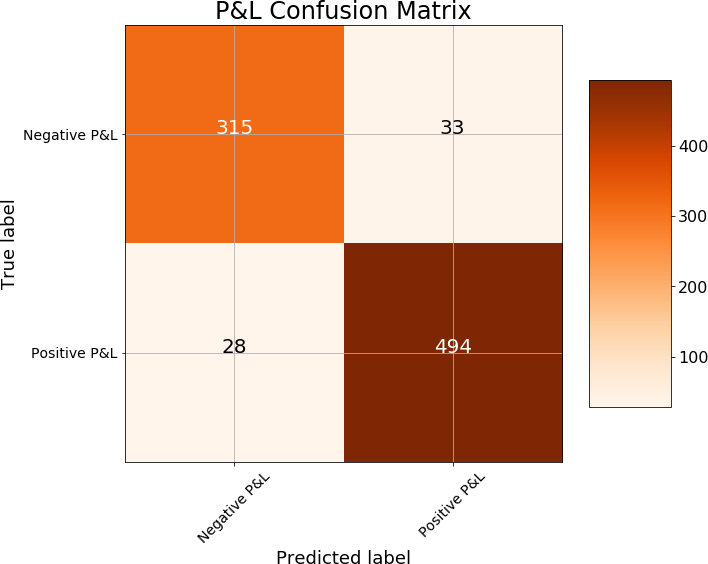
|  |  |  |  |
| --- | --- | --- | --- |
|  | **Baseline** | **Test** | **Train** |
| **Recall** | 1 | 0.95 | 1 |
| **Precision** | 0.6 | 0.94 | 1 |
| **AUC** | 0.5 | 0.98 | 1 |

**Figure 20.** ROC Curve for period {90,90} with 100 random trees



The measures for the test set in the random forest case are 0.95, 0.94 and 0.98 for precision, recall, and AUC, respectively. As observed, these results are better than those of previous case (the decision tree) for the strategy for period {90,90}. It is important to note that we selected the {90,90} period strategy considering that this period presents the more balanced classes of negative and positive returns. Figure 7 presents the confusion matrix for the analyzed period.

**Figure 21.** Confusion matrix for period {90,90}



Comparing the cases of misclassification reveals that the random forest does a better job with respect to the (individual) classification tree. The false negative is now 33 cases (before it was 46), and the false positive is 28 cases (previously it was 57). Though there are several changes in the importance feature, node 180 is still the most important variable as shown in Table 6.

### Table 10. The most important features for period {90,90} with random forest

|  |  |
| --- | --- |
| **Feature** | **Importance** |
| **Node 180** | 0.116424 |
| **Node 360** | 0.088857 |
| **Node 90** | 0.077711 |
| **Node 720** | 0.074946 |
| **Node 3240** | 0.066314 |
| **Node 2880** | 0.056533 |
| **Node 2160** | 0.052062 |
| **10-year Inflation** | 0.051956 |
| **Node 1800** | 0.051123 |
| **5-year Inflation** | 0.045169 |

Since more nodes of the zero-coupon yield curve are incorporated in the feature importance indicator in the random forest technique, this methodology employs the information in a more efficient way than the previous method (single classification tree).

## Random Forest Optimization through Random Search

To improve the performance of the random forest, we performed a random search for better hyperparameter tuning. We then evaluated the random selection of hyperparameter combinations by using cross validation of training data, and then chose the best performing parameter values for the prediction procedure.

* **bootstrap:** whether bootstrap samples are used when building trees.
* **n estimators:** the number of trees in the forest.
* **min sample split:** the minimum number of samples required to split an internal node.
* **max leaf nodes:** limits the number of leaf nodes as long as there is no improvement of impurity.
* **min samples split:** the minimum number of samples required to split an internal node.

For our application, the results of the best parameters are presented in Table 7.

### Table 11. Results of the hyperparameters after random search

|  |  |
| --- | --- |
|  | Best Parameters |
| Bootstrap | 16 |
| N estimators | 100 |
| Min sample split | 43 |
| Max leaf nodes | 5 |
| Min sample splits | 29 |

With these parameters for the random forest model, the mean of nodes is 274 and the maximum depth is 15 on average. Table 8 and Figure 8 present the results of the performance measures for the best model.

**Table 12.** Best random forest model according to random search for period {90,90}

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Baseline** | **Test** | **Train** |
| **Recall** | 1 | 0.95 | 0.98 |
| **Precision** | 0.81 | 0.94 | 0.87 |
| **AUC** | 0.50 | 0.98 | 0.88 |

**Figure 22.** ROC Curve for period {90,90} with the best random forest model



Though this model exhibits worse indicators than the previous case for the training set, the performance for the test data is the same as the previous model. In the next section, we present the results for all the cases: periods {90,90}, {180,180}, {360,360}, {720,360}, {1080, 360},{1440,360}, {1800, 360}, and {2160, 360} using the best random forest model with the hyperparameters tuned according to the random search procedure.

## Using Random Forest Optimization

For each period, we implemented the machine learning technique based on the best random forest model in riding the Colombian yield curve. We found the best hyperparameters by random search in the cross-validation procedure as previously explained. Table 9 shows the performance measures for each period in the baseline case, train set, and test set.

### Table 13. Random forest model results for different period strategies

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Baseline** | **Test** | **Train** | **Baseline** | **Test** | **Train** |
|  |  | **90, 90** |  | **180,180** | | |
| **Recall** | 1 | 0.95 | 0.98 | 1 | 0.96 | 0.99 |
| **Precision** | 0.81 | 0.94 | 0.87 | 0.72 | 0.72 | 0.99 |
| **ROC** | 0.50 | 0.98 | 0.88 | 0.50 | 0.96 | 1 |
|  |  | **360, 360** |  |  | **720, 360** |  |
| **Recall** | 1 | 1 | 1 | 1 | 0.99 | 1 |
| **Precision** | 0.86 | 0.99 | 1 | 0.85 | 0.98 | 0.99 |
| **ROC** | 0.50 | 1 | 1 | 0.5 | 1 | 1 |
|  |  | **1080, 360** |  |  | **1440, 360** |  |
| **Recall** | 1 | 1 | 1 | 1 | 0.99 | 1 |
| **Precision** | 0.93 | 0.99 | 1 | 0.98 | 1 | 1 |
| **ROC** | 0.50 | 1 | 1 | 0.5 | 1 | 1 |
|  |  | **1800, 360** |  |  | **2160, 360** |  |
| **Recall** | 1 | 0.94 | 1 | 1 | 0.96 | 1 |
| **Precision** | 0.66 | 0.95 | 0.99 | 0.25 | 0.94 | 1 |
| **ROC** | 0.50 | 0.98 | 1 | 0.5 | 1 | 1 |

As observed in Table 9, the model performs well in terms of classifying the negative and positive returns of the strategies for the test set. The precision measure is lower than 0.95 for only three cases (not shown in Table 9): periods {90,90}, {180,180}, and {2160,360}. In particular, the precision is 0.72 for the period {180, 180}, which is the same result as for the baseline case.

### Table 14. Sample sizes and number of positive and negative returns for different periods’ strategies

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **90 – 90** | **180- 180** | **360 - 360** | **720 - 360** | **1080 - 360** | **1440 - 360** | **1800 - 360** | **2160 - 360** | **2520 - 360** | **2880 - 360** | **3240 - 360** |
| ***r <* 0** | 1484 | 1115 | 692 | 523 | 147 | 30 | 568 | 1058 | 0 | 1 | 1 |
| ***r >* 0** | 1859 | 2163 | 2455 | 2363 | 2478 | 2333 | 1535 | 785 | 1582 | 1319 | 1058 |
| **N** | 3343 | 3278 | 3147 | 2886 | 2625 | 2363 | 2103 | 1843 | 1582 | 1320 | 1059 |

As we can see in Table 10, for the periods in which we do not adjust the classification model there are no or very few negative results. Thus, the problem is no longer a classification problem but one of determining the value of the profit of the strategy, which is why a different method must be applied.

**Discussion and Conclusions**

The term structure of interest rates, represented graphically by the yield curve, has been considered a powerful instrument for predicting financial crises, and the behavior of other important macroeconomic variables (economic activity, inflation, and fiscal and monetary policies, among others). In this sense, it has been and is the focus of theoretical discussions about the factors that alter its form, among which stand out the approaches of the theory of pure expectations, the theory of market segmentation, and that on which it is based the development of this work, liquidity preference.

According to this last theory, investors, in general, require a risk premium to be encouraged to buy financial assets that are valid in the medium and long term. Therefore, financial assets with such maturity periods usually present higher rates of return than those whose maturity period is shorter, hence their "normal" form, in most cases. In this research, we show that this characteristic can be used by the different type of market agents, since it is possible to predict, with a certain degree of precision, and under normal conditions, the movements of the curve.

In this sense, Machine learning techniques have been successfully applied in several financial applications. Since the work of Kondratyev (2018), there has been an increasing interest in the analysis of machine learning in the interest rate field. Our work contributes to this strand of the finance literature with the application of Machine Learning techniques to the so-called ‘riding the yield curve’ strategy.

.

Our empirical evidence shows that classification trees perform well in classifying the negative and positive returns of the strategies in Colombian government bonds for the 2006 – 2019 period. We also found that regression trees present good results for the analyzed period, and that, according to the MSE values. The same as Pelaez (1997), our results show evidence in favor of the liquidity preference theory.

Regarding prediction process using regression methods, we implemented dimensionality reduction methods, decomposing the curve into the factors or components that explain its variation to a greater degree. It is noteworthy that the components obtained through the principal component analysis (PCA) coincide with the factors underlying the shape of the level curve, slope, and curvature, proposed by the theory, and that in the Colombian case they manage to explain more than 99% of the variability of the data. This last fact helps to explain the high predictive capacity of the proposed linear models.

Considering the valuable source of information that the yield curve represents for all market agents, before the modeling process, an exhaustive process of exploration and analysis of the movements of the curve was carried out for the study period (7/26/ 2006 to 2/22/20019). In general, changes in the level and steepening of the curves were observed as time progressed, and different stress scenarios were generated, using VaR as a risk measure, to verify the capacity of the main components. to explain the behavior of the curve, even in extreme cases.

Finally, the modeling and prediction process was given, in which different types of models were tested, from those characterized by less flexibility, and linearity, but greater interpretability to those that are much more flexible and non-linear, but with restrictions in terms of their performance. interpretability. According to the results obtained, linear models are superior of non-linear models in terms of prediction for the case of regression models

Future research could be very practical for the different type of market agents to improve the models in their explanatory and causal capacity. It is also suggested to explore the management of risk and the construction of portfolios taking into account different stress scenarios.

# Bibliografía

Ang, S., Alles, L., & Allen, D. (1998). Riding the yield curve: An analysis of international evidence. *The Journal of Fixed Income, 8*(3), 57-74.

Arulkumaran, K., Deisenroth, M. P., Brundage, M., & Bharath, A. A. (2017). Deep Reinforcement Learning: A Brief Survey. *IEEE Signal Processing Magazine, 34*(6), 26-38. https://doi.org/10.1109/MSP.2017.2743240

Bellman, R. (1952). On the Theory of Dynamic Programming. *Proceedings of the National Academy of Sciences, 38*(8), 716-719. https://doi.org/10.1073/pnas.38.8.716

Bellman, R. (1966). Dynamic programming. *Science, 153*(3731), 34-37. https://doi.org/10.1126/science.153.3731.34

Black, F., & Scholes, M. (1973). The pricing of options and corporate liabilities. *Journal of political economy, 81*(3), 637-654.

Bolsa de Valores de Colombia S.A. (2023). *Mercado Local en Línea*. Bolsa de Valores de Colombia: www.bvc.com.co

Busoniu, L., Babuska, R., & De Schutter, B. (2008). A Comprehensive Survey of Multiagent Reinforcement Learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 38*(2), 156-172. https://doi.org/10.1109/TSMCC.2007.913919

Carapuco, J., Ferreira-Neves, R., & Horta, N. C. (2018). Reinforcement learning applied to forex trading. *Applied Soft Computing, 73*, 783-794. https://doi.org/10.1016/j.asoc.2018.09.017

Carta, S., Rerreira, A., Podda, A. S., Recupero, D. R., & Sannai, A. (2021). Multi-dqn: An ensemble of deep q-learning agents for stock market forecasting. *Expert Systems with Applications, 164*, 113820. https://doi.org/10.1016/j.eswa.2020.113820

Chua, C. T., Koh, W. T., & Ramaswamy, K. (2005). Comparing returns of US treasuries versus equities: implications for market and portfolio efficiency. *Applied Financial Economics, 15*(17), 1213-1218.

Dixon, M., Halperin, I., & Bilokon, P. (2020). *Machine Learning in Finance: From Theory to Practice.* Springer International.

Dyl, E., & Joehnk, M. D. (1981). Riding the yield curve: does it work? *The journal of portfolio management, 7*(3), 13-17.

Fabozzi, F. J. (2021). *The handbook of fixed income securities* (Novena ed.). McGraw-Hill Education.

Fernández-Rodrı́guez, F., González-Martel, C., & Sosvilla-Rivero, S. (2000). On the profitability of technical trading rules based on artificial neural networks: Evidence from the Madrid stock market. *Economics Letters, 69*(1), 89-94. https://doi.org/https://doi.org/10.1016/S0165-1765(00)00270-6

Galvani, V., & Landon, S. (2013). Riding the yield curve: a spanning analysis. *Review of Quantitative Finance and Accounting, 40*, 135-154.

Gogas, P., Papadimitriou, T., Matthaiou, M., & Chrysanthidou, E. (2015). Yield curve and recession forecasting in a machine learning framework. *Computational Economics, 45*, 635-645.

Grieves, R., & Marcus, A. J. (1992). Riding the yield curve: reprise. *Journal of Portfolio Management, 18*(4), 67-76.

Gu, S., Lillicrap, T., Sutskever, I., & Levine, S. (2016). Continuous Deep Q-Learning with Model-based Acceleration. *Proc. Int. Conf. Learning Representations,.* https://arxiv.org/abs/1603.00748

Henrique, B. M., Sobreiro, V. A., & Kimura, H. (2019). Literature review: Machine learning techniques applied to financial market prediction. *Expert Systems with Applications, 124*, 226-251. https://doi.org/https://doi.org/10.1016/j.eswa.2019.01.012

Hull, J. C. (2022). *Options, Futures, and Other Derivatives* (Undécima ed.). Pearson.

Kakade, S. M. (2001). A Natural Policy Gradient. *Advances in Neural Information Processing Systems* (págs. 1531-1538). Cambridge: MIT Press.

Konda, V. R., & Tsitsiklis, J. N. (2003). OnActor-Critic Algorithms. *SIAM Journal on Control and Optimization, 42*(4), 1143-1166. https://doi.org/10.1137/S0363012901385691

Koutník, J., Cuccu, G., Schmidhuber, J., & Gomez, F. J. (2013). Evolving large-scale neural networks for vision-based reinforcement learning. *Annual Conference on Genetic and Evolutionary Computation.* Manno. https://doi.org/10.1145/2463372.2463509

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*, 436-444. https://doi.org/10.1038/nature14539

López de Prado, M. M. (2020). *Machine Learning for Asset Managers.* Cambridge University Press. https://doi.org/10.1017/9781108883658

Merton, R. C. (1973). Theory of Rational Option Pricing. *The Bell Journal of Economics and Management Science, 4*(1), 141-183.

Moody, J., & Saffell, M. (1998). Reinforcement Learning for Trading. En M. Kearns, S. Solla, & D. Cohn (Ed.), *Advances in Neural Information Processing Systems.* *11.* Boston: MIT Press. https://proceedings.neurips.cc/paper\_files/paper/1998/file/4e6cd95227cb0c280e99a195be5f6615-Paper.pdf

Nunes, M. (2022). *Machine learning in fixed income markets: forecasting and portfolio management.* University of Southampton.

Pelaez, R. F. (1997). Riding the yield curve: Term premiums and excess returns. *Review of Financial Economics, 6*(1), 113-119. https://doi.org/https://doi.org/10.1016/S1058-3300(97)90017-3

Refenes, N. A.-P., Burgess, N. A., & Bentz, Y. (1997). Neural networks in financial engineering: a study in methodology. *IEEE transactions on neural networks, 8*, 1222-1267. https://doi.org/10.1109/72.641449

Stone, P., & Kohl, N. (2004). Policy gradient reinforcement learning for fast quadrupedal locomotion. *IEEE International Conference on Robotics and Automation, 3*, 2619-2624. https://doi.org/10.1109/ROBOT.2004.1307456

Sutton, R. S. (1984). *Temporal credit assignment in reinforcement learning.* University of Massachusetts Amherst.

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement Learning: An Introduction* (Segunda ed.). MIT Press.

Sutton, R. S., Precup, D., & Singh, S. (1999). Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence, 112*(1), 181-211. https://doi.org/https://doi.org/10.1016/S0004-3702(99)00052-1

Tao, C., & Wencong, S. (2018). Local Energy Trading Behavior Modeling With Deep Reinforcement Learning. *IEEE Access, 6*, 62806-62814. https://doi.org/10.1109/ACCESS.2018.2876652

van Hasselt, H., Guez, A., & Silver, D. (2015). Deep Reinforcement Learning with Double Q-Learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, *30.* https://doi.org/10.1609/aaai.v30i1.10295

Wang, X., Wang, S., Liang, X., Zhao, D., Huang, J., Xu, X., . . . Miao, Q. (2022). Deep reinforcement learning: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 1-15. https://doi.org/10.1109/TNNLS.2022.3207346

Watkins, C. J. (1989). *Learning from delayed rewards.* Learning from delayed rewards.

Watkins, C. J., & Dayan, P. (1992). Q-Learning. *Machine Learning, 8*(3), 279-292. https://doi.org/10.1007/BF00992698

Wu, X., Chen, H., Wang, J., Troiano, L., Loia, V., & Fujita, H. (2020). Adaptive stock trading strategies with deep reinforcement learning methods. *Information Sciences, 538*, 142-158.

Yang, H., Lui, X.-Y., Zhong, S., & Walid, A. (2020). Deep Reinforcement Learning for Automated Stock Trading: An Ensemble Strategy. *Proceedings of the First ACM International Conference on AI in Finance.* Nueva York: Association for Computing Machinery. https://doi.org/10.1145/3383455.3422540

Yoon, Y., Swales, G., & Margavio, T. M. (1993). A Comparison of Discriminant Analysis versus Artificial Neural Networks. *Journal of the Operational Research Society, 44*(1), 51-60. https://doi.org/10.1057/jors.1993.6

Zenios, S. A., & Ziemba, W. T. (2007). *Handbook of Asset and Liability Management: Applications and case studies.* Elsevier.

Zhang, N., Lin, A., & Shang, P. (2017). Multidimensional k-nearest neighbor model based on EEMD for financial time series forecasting. *Physica A: Statistical Mechanics and its Applications, 477*, 161-173. https://doi.org/https://doi.org/10.1016/j.physa.2017.02.072

Zimmermann, H., Neuneier, R., & Grothmann, R. (2000). Modeling of the german yield curve by error correction neural networks. *Proceedings of the Second International Conference on Intelligent Data Engineering and Automated Learning, Data Mining, Financial Engineering, and Intelligent Agents* (págs. 262-267). Berlín: Springer-Verlag.

**References**

Ang, S., Alles, L. and Allen, D. (1998): Riding the Yield Curve: An Analysis of International Evidence, Journal of Fixed Income 8, 57–74.

Baele, L., Bekaert, G., and Inghelbrecht, K. (2010). The determinants of stock and bond return comovements. The Review of Financial Studies, 23(6):2374–2428.

Benth, F.E., Detering, N. & Lavagnini, S. (2021). Accuracy of Deep Learning in Calibrating HJM Forward Curves. Digital Finance forthcoming.

Bianchi D, Büchner M, Tamoni A (2021). Bond risk premiums with machine learning. Rev. Financial Stud. 34(2):1046–1089.

Bieri, D.S. and Chincarini, L. (2005): Riding the Yield Curve: A Variety of Strategies, Journal of Fixed Income 15, 6–35.

Campbell, J. Y. and Shiller, R. J. (1991). Yield spreads and interest rate movements: A bird’s eye view. The Review of Economic Studies, 58(3):495–514.

Chandy, P.R. and Hsueh, P. (1995): Riding the Yield Curve: An Empirical Examination of Recent US Data, Singapore Management Review 17, 31–39.

Chataigner, M., Crépey, S. and Pu, J. (2020). Nowcasting Networks. Journal of Computational Finance 24(3), 1-39.

Chua CT, Koh WTH, Ramaswamy K (2005) Comparing returns of US treasuries versus equities: implications for market and portfolio efﬁciency. Appl Financial Econ 15:1213–1218.

Cox, R. A., & Felton, J. M. (1994). (1994). Performance from Riding the Yield Curve: 1980-1992. *Journal of Business and Economic Perspectives*, *20*, 128–132.

Culbertson, J. M. (1957). The term structure of interest rates. The Quarterly Journal of Economics, 71(4):485–517.

Dewachter, H., Iania, L., and Lyrio, M. (2014). Information in the yield curve: A macro- finance approach. Journal of Applied Econometrics, 29(1):42–64. 12

Dyl, E.A., and M.D. Joehnk, 1981, “Riding the Yield Curve: Does it Work?” Journal of Portfolio Management, 7(3), 13–17.

English, W. B., Van den Heuvel, S. J. & Zakrajsek, E. (2018). Interest rate risk and bank equity valuations. Journal of Monetary Economics 98, 80-97.

Fama, E. F. (1984). The information in the term structure. Journal of financial economics, 13(4):509–528.

Fama, E. F. (2006). The behavior of interest rates. The Review of Financial Studies, 19(2):359–379.

Fisher, I. (1896). Appreciation and Interest: A Study of the Influence of Monetary Ap- preciation and Depreciation on the Rate of Interest with Applications to the Bimetallic Controversy and the Theory of Interest, volume 11. American economic association.

Frankel, J. A. and Lown, C. S. (1994). An indicator of future inflation extracted from the steepness of the interest rate yield curve along its entire length. The Quarterly Journal of Economics, 109(2):517–530.

Galvani, V., Landon, S. (2012). Riding the yield curve: A spanning analysis. Review of Quantitative Finance and Accounting 40 (1), 135-154.

Geurts P., Ernst D., and Wehenkel, L. (2006). Extremely randomized trees. Machine Learning, 63(1), 3-42.

Gogas P, Papadimitriou T, Matthaiou M, Chrysanthidou E. 2015. Yield curve and recession forecasting in a machine learning framework. Computational Economics 45: 635–645.

Grieves, R., and A.J. Marcus, (1992): Riding the Yield Curve: A Reprise, Journal of Portfolio Management 18, 67–76.

Grieves, R., Mann, S., Marcus, A.J. and Ramanlal, P. (1999): Riding the Bill Curve, Journal of Portfolio Management 25, 74–82.

Hamburger, M. J. and Platt, E. N. (1975). The expectations hypothesis and the efficiency of the treasury bill market. The Review of Economics and Statistics, pages 190–199. 24

Haubrich, J. G. (2004). Interest Rates, Yield Curves, and the Monetary Regime. Federal Reserve Bank of Cleveland. Economic Commentary; Cleveland (Jun 2004): 1-3.

Hicks, J. R. (1939). Value and capital. London: Oxford University Press.

Jansen, S. (2020). Machine Learning for Algorithmic Trading: Predictive models to extract signals from market and alternative data for systematic trading strategies with Python. Packt Publishing Ltd.

Kane, E. J. and Malkiel, B. G. (1967). The term structure of interest rates: An analysis of a survey of interest-rate expectations. The Review of Economics and Statistics, pages 343–355.

Kanevski, Mikhail F., and Vadim Timonin. 2010. Machine learning analysis and modeling of interest rate curves. Paper presented at the 18th European Symposium on Artificial Neural Networks ESANN, Bruges, Belgium, April 28–30; Available online: <https://www.elen.ucl.ac.be/Proceedings/esann/esannpdf/es2010-17.pdf>

Kim, M. (2021). Adaptive trading system integrating machine learning and back-testing: Korean bond market case. Expert Systems with Applications 176, 114767.

Kondratyev, A. (2018) Curve dynamics with artificial neural networks. Risk 31 (6), 74-79.

Lee, C. F. (2021). Bond Portfolio Management, Swap Strategy, Duration, and Convexity. In Lee, C.F. and Lee, J.C. (Ed). Handbook of Financial Econometrics, Mathematics, Statistics, and Machine Learning (pp. 3059-3098). World Scientific Pub Co Inc.

Martellini, L., Priaulet, P. and S. Priaulet (2003): Fixed-Income Securities: Valuation, Risk Management and Portfolio Strategies, Wiley, West Sussex, England.

Mercer, J., Moore, M.E. and D.B. Winters (2009): Do Traders Benefit from Riding the T-Bill Yield Curve? Journal of Portfolio Management 26, 131–140.

Modigliani, F. and Sutch, R. (1966). Innovations in interest rate policy. The American Economic Review, 56(1/2):178–197. 34

Nunes, M., Gerding, E. McGroarty, F., and Niranjan, M. (2019). A comparison of multitask and single task learning with artificial neural networks for yield curve forecasting. Expert Systems with Applications, 119:362–375.

Pantalone, C. C., & Platt, H. D. (1984). Riding the Yield Curve. *Journal of Financial Education*, 5–9.

Pelaez R.F. (1997). Riding the yield curve: term premiums and excess returns. Rev Financial Econ 6(1):113–119.

Phoa, W. and Shearer, M. (1998). Advanced fixed income analytics, volume 32. John Wiley & Sons.

Rey, M. (2005). La hipótesis de expectativas en la estructura a plazo de las tasas de interés: Una estimación para Colombia. Colegio Mayor Universidad del Rosario.

Rueda, D. A. and Arango, M. A. A. (2008). La curva de rendimientos a plazo y las expecta- tivas de tasas de interés en el mercado de renta fija en Colombia, 2002-2007. Lecturas de Economía, (68):39–66.

Spears, T. Zohren, S. and Roberts, S. (2021). Investment sizing with deep learning prediction uncertainties for high-frequency eurodollar futures trading. J. Financial Data Sci. 3 (1) 57–73.

Yun, J. (2019). Bond risk premia in a small open economy with volatile capital flows; the case of Korea. Journal of International Money and Finance, 93:223–243. 42

1. El concepto de Asset and Liability Management (ALM) para las entidades financieras suele ser bastante relevante. Es a partir de este concepto que el negocio bancario puede ser rentable y sostenible en el tiempo, y cuenta con un alto nivel de complejidad técnico, ampliamente explicado en (Zenios & Ziemba, 2007). [↑](#footnote-ref-1)
2. De acuerdo con cifras oficiales publicadas por la Bolsa de Valores de Colombia en (Bolsa de Valores de Colombia S.A, 2023), el monto diario negociado de deuda pública en el mercado público promedia los 1,8 billones de pesos colombianos, convirtiéndolo así en el mercado de valores más líquido, amplio y profundo que opera en Colombia. Si se desea, se puede comparar con el volumen medio del mercado accionario de tan solo 70 mil millones de pesos colombianos al día. [↑](#footnote-ref-2)
3. (López de Prado, 2020) explica en su libro como el Machine Learning ha impactado positivamente la gestión de los asset managers, es decir, los gestores de carteras de inversión, toda vez que se minimizan tiempos de análisis de las potenciales inversiones, al tiempo que los mismos son, en general, más precisos. Todo esto ayuda al gestor a tomar decisiones más informadas, y, en general, mejores para su gestión específica. [↑](#footnote-ref-3)
4. The rate in the period for a period is . Assuming that the rates are continuous compounding, the forward rate is , where, the rate at time for a period is , and the rate at time for a period is .

   [↑](#footnote-ref-4)
5. The Gini Impurity index is the probability that a randomly selected sample from the node will be incorrectly classified. The idea is to reduce the Gini impurity, which will eventually reach 0 if the tree does not have a depth limit. The entropy index is also employed, but results do not vary substantially. [↑](#footnote-ref-5)