



Facultad de empresariales

# **MODELO PREDICTIVO DE RESULTADOS JUDICIALES.**

## **EL CASO DE LA IA Y BIG DATA EN SENTENCIAS DEL TRIBUNAL DE JUSTICIA EUROPEO.**

Clave: 201905347

# ÍNDICE

|   |    |
|---|----|
| <b>CAPÍTULO I. INTRODUCCIÓN</b> .....                                 | 4  |
| 1. ANTECEDENTES .....   | 4  |
| 2. OBJETIVOS .....  | 6  |
| 3. METODOLOGÍA .....  | 6  |
| <b>CAPÍTULO II. LA IA Y BIG DATA EN LOS PROCESOS JUDICIALES</b> ..... | 8  |
| 1. CONCEPTO.....  | 8  |
| <b>1.1. Aprendizaje automático o Machine Learning</b> .....           | 8  |
| <b>1.2. Modelos predictivos</b> .....                                 | 9  |
| <b>1.3. Algoritmos</b> .....  | 10 |
| 2. HITOS IMPORTANTES .....  | 11 |
| 3. PAISAJE ACTUAL .....   | 12 |
| <b>3.1. En el mundo</b> .....   | 13 |
| <b>3.2. En la UE</b> .....  | 14 |
| 4. PERSPECTIVAS .....   | 14 |
| <b>CAPÍTULO III. MODELO PREDICTIVO DE SENTENCIAS</b> .....            | 16 |
| 1. METODOLOGÍA .....  | 16 |
| <b>1.1. Fuente de datos</b> .....                                     | 16 |
| <b>1.2. Recopilación y limpieza de datos</b> .....                    | 16 |
| <b>1.3. Descripción de datos del modelo</b> .....                     | 17 |
| <b>1.4. Procesamiento de datos</b> .....                              | 22 |
| 2. DESARROLLO DEL MODELO PREDICTIVO .....                             | 23 |
| <b>2.1. Diseño del modelo</b> .....                                   | 23 |
| <b>2.2. Entrenamiento del modelo</b> .....                            | 25 |
| <b>2.3. Validación del modelo</b> .....                               | 26 |
| 3. ANÁLISIS DE RESULTADOS .....                                       | 27 |
| <b>3.1. Interpretación de resultados</b> .....                        | 27 |
| <b>3.2. Relevancia de las variables</b> .....                         | 29 |
| 4. GENERACIÓN DEL MODELO .....  | 31 |
| <b>CAPÍTULO IV. DESAFÍOS ÉTICOS Y LEGALES</b> .....                   | 32 |
| 1. PRIVACIDAD Y PROTECCIÓN DE DATOS.....                              | 32 |
| 2. EL SESGO EN LOS ALGORITMOS .....                                   | 34 |
| 3. TRANSPARENCIA.....   | 35 |
| 4. MARCO LEGAL Y REGULACIONES ACTUALES .....                          | 36 |
| <b>CAPÍTULO V. CONCLUSIONES</b> .....                                 | 37 |

|  |    |
|--|----|
| <b>DECLARACIÓN DE USO DE HERRAMIENTAS DE INTELIGENCIA ARTIFICIAL<br/>GENERATIVA EN TRABAJOS FIN DE GRADO</b> ..... | 39 |
| <b>BIBLIOGRAFÍA</b> .....  | 40 |
| <b>ANEXOS</b> .....  | 44 |

## ÍNDICE DE FIGURAS

|  |    |
|--|----|
| Figura 1. Visualización de valores nulos .....                                       | 16 |
| Figura 2. Descripción estadística de las variables .....                             | 20 |
| Figura 3. Descripción estadística de las variables .....                             | 20 |
| Figura 4. Distribución de la variable “has_judgement” .....                          | 21 |
| Figura 5. Relación entre las variables “has_judgement” y “count_decisions”.....      | 21 |
| Figura 6. Relación entre las variables “has_judgement” y “count_on_appeal” .....     | 22 |
| Figura 7. Entrenamiento y ajuste de los pipelines rf y gb .....                      | 26 |
| Figura 8. Validación cruzada y ajuste .....  | 26 |
| Figure 9. Matriz de confusión de rf.....   | 27 |
| Figura 10. Matriz de confusión de gb.....  | 28 |
| Figura 11. Relevancia de las variables en rf.....                                    | 29 |
| Figura 12. Relevancia de las variables en gb.....                                    | 30 |
| Figura 13. Disponibilidad en archivos.....   | 32 |
| Figuae 14. Ejemplo de tratamiento de datos relativo a la selección de personal ..... | 33 |

## ÍNDICE DE ABREVIATURAS

|          |   |
|----------|---|
| AEPD     | Agencia Española de Protección de Datos                                       |
| CDFUE    | Carta de Derechos Fundamentales de la Unión Europea                           |
| CEDH     | Carta Europea de Derechos Humanos   |
| COMPAS   | Corrección y Perfilado de Gestión de Delincuentes para Sanciones Alternativas |
| DSA      | Servicios de los Mercados Digitales   |
| GB       | “Gradient Boosting”   |
| IA       | Inteligencia Artificial   |
| KNN      | K-Vecinos Más Cercanos  |
| ML       | “Machine Learning”  |
| ODR      | Métodos Alternativos de Resolución de Conflictos                              |
| op. cit. | “opere citato” o en la obra citada anteriormente.                             |
| PPO      | Optimización Proximal de Políticas  |
| RF       | “Random Forest”   |

|      |   |
|------|---|
| RGPD | Reglamento General de Protección de Datos |
| SQRT | Raíz cuadrada (comando de Python 3)       |
| SVM  | Máquina de Vectores de Soporte            |
| TEDH | Tribunal Europeo de Derechos Humanos      |
| TJUE | Tribunal de Justicia de la Unión Europea  |
| UE   | Unión Europea                             |

## CAPÍTULO I. INTRODUCCIÓN

Desde tiempos inmemoriales, la administración de justicia ha sido una piedra angular de las sociedades civilizadas, siendo los tribunales la última instancia para resolver disputas y aplicar la ley. En este contexto, el paradigma de la justicia experimenta una transformación sin precedentes con la creciente integración de la Inteligencia Artificial (IA) y el análisis de Big Data en los procesos judiciales. El presente trabajo se sumerge en esta revolución judicial, centrándose específicamente en el Tribunal de Justicia de la Unión Europea (TJUE) y explorando la intersección entre la tecnología avanzada y la toma de decisiones judiciales.

Hace no muchos años la IA parecía una realidad lejana, mayormente restringida a entornos de laboratorios de investigación y al desarrollo llevado a cabo por empresas tecnológicas. Sin embargo, la IA ya es más inteligente que cualquier humano en el planeta en términos de muchas tareas específicas y aisladas. El campeón mundial de ajedrez es una máquina. El campeón mundial de Jeopardy es el superordenador de IBM, Watson. El campeón mundial de Go<sup>1</sup> es AlphaGo de Google (Gawdat, 2021).

Tras el lanzamiento público de ChatGPT, la IA se ha vuelto el foco de una considerable atención, representando un hito significativo según indican las fuentes (IBM, 2023), y ha destacado el extenso rango de aplicaciones que la IA podría brindar al sector legal. Un reciente sondeo realizado por Thomson Reuters pone de manifiesto que un considerable 82% de los abogados en despachos legales percibe que la IA posee un potencial notable para aplicarse en temas legales (véase el Anexo 1) (Thompson Reuters, 2023).

### 1. ANTECEDENTES

La conceptualización del término "Inteligencia Artificial" data de 1950, momento en el cual Alan Turing publicó su obra "Computing Machinery and Intelligence". En dicho documento, Turing formuló la cuestión fundamental: ¿poseen las máquinas la capacidad de pensar? En este contexto, propuso el célebre "Test de Turing", un experimento conceptual destinado a evaluar si una máquina puede manifestar un comportamiento inteligente que resulte indistinguible del observado en un ser humano. Si la persona que está entablando la conversación con la máquina no era capaz de distinguirla de un humano, la máquina habría pasado el test (Innovación digital 360, 2023).

En la década de 1950 a 1960, se registraron progresos significativos con proyectos como el programa de ajedrez concebido por Claude Shannon y el "Logic Theorist" creado por Newell y Simon (Serna, 2019). Dada la multiplicidad de beneficios de la IA en el ámbito legal, cabe destacar la relevancia de la obra "Lo que pueden hacer las computadoras: Análisis y predicción de decisiones judiciales", publicada en 1963. El autor sostiene que la precisión de los resultados predictivos está intrínsecamente ligada a la comprensión

---

<sup>1</sup> Go es un juego de estrategia abstracto inventado en China hace más de 2.500 años y se sabe que es uno de los juegos de estrategia más complejos debido a su gran número de posibles configuraciones de tablero. Para hacerse una idea de la escala, hay más movimientos posibles en Go que átomos en el universo.

científica de cómo la ley y los hechos inciden en los agentes de toma de decisiones pertinentes, en particular, los jueces (Lawlor, 1963).

Posteriormente, entre las décadas de 1980 y 1990, emergieron nuevas técnicas, entre las cuales destacan las redes neuronales y los sistemas expertos. A partir del año 2000, el rápido crecimiento ha sido impulsado por el incremento en la capacidad de almacenamiento y el acceso a vastos conjuntos de datos.

La mayoría de las computadoras hasta el cambio de siglo XXI no se pueden considerar realmente inteligentes, realizaban lo que sus programadores les indicaban. Obedecían y hacían lo que se les decía, pero lo hacían a grandes velocidades. Por ejemplo, el primer Google, que ha ayudado a la humanidad a organizar toda la información del mundo, no era inteligente como tal, lo eran aquellos que lo construyeron. Durante años, la aparente “genialidad” de Google era simplemente el resultado de su capacidad para clasificar un número masivo de sitios web y descubrir qué páginas se posicionaban en la cima en función de cuántas otras páginas las mencionaban. Cuantas más referencias recibe una página, mayor es su importancia y relevancia para los buscadores. Este algoritmo se conoce como Page Rank (Stox, 2023). Amazon o Spotify funcionan de manera similar, observan a aquellos clientes a los cuales les ha gustado el producto que has comprado o las canciones que has escuchado y te sugieren qué más han comprado y escuchado la mayoría de esas personas (Pastukhov, 2022) (Breslin, 2023).

Ya existen programas desarrollados por empresas que se dedican a prever el desenlace de decisiones judiciales mediante el empleo de herramientas tecnológicas conocidas como justicia predictiva, entre las cuales se destacan Watson/Ross de IBM, Blue J. Legal. de Benjamin Alarie y la Universidad de Toronto. Asimismo, otros programas como Premonition Analytics o Bloomberg Litigation Analytics elaboran perfiles detallados de jueces y magistrados mediante un análisis histórico de su actividad jurisdiccional (Choo, Espejo, & Jayasinghe, 2020).

De esta manera, la IA se presenta como una herramienta muy útil en campos como el derecho, que puede facilitar la uniformidad de la jurisprudencia en casos que requieran establecer escalas o criterios, como procesos civiles relacionados con daños corporales o compensaciones por divorcio, así como en procesos laborales vinculados a despidos con indemnización. Además, se destaca su capacidad para fomentar la resolución de conflictos en línea y la prestación de servicios judiciales electrónicos, también conocidos como Métodos Alternativos de Resolución de Conflictos (ODR, por sus siglas en inglés) (Matin, 2022).

En resumen, nuestro panorama legal está evolucionando rápidamente, y uno de los disruptores más significativos de los últimos años es el uso del análisis predictivo. La analítica predictiva se refiere a la predicción de resultados a través del análisis de datos cuantitativos, lo que permite a los abogados participar en la toma de decisiones más eficientes y estratégicas (SAS Institute Inc., 2023). Esto incluye la analítica judicial predictiva, que implica el análisis del comportamiento judicial al examinar los patrones de toma de decisiones de los funcionarios judiciales y predecir el resultado probable (Media, 2019).

## 2. OBJETIVOS

La cuestión que motiva esta investigación se centra en la integración de la IA y el Big Data en el ámbito judicial, con un enfoque particular en las sentencias del TJUE. A medida que los sistemas judiciales exploran la integración de tecnologías emergentes, surge la necesidad de examinar críticamente los efectos de estas innovaciones en la administración de la justicia, especialmente en un contexto tan relevante como el marco legal de la Unión Europea.

El **Objetivo Principal** será desarrollar y evaluar un modelo predictivo capaz de determinar la probabilidad de que un caso presentado ante el TJUE resulte en una sentencia, utilizando técnicas de IA, Big Data y ML para lo que se empleará Python 3.

Los **Objetivos secundarios** serán los siguientes:

- Realizar una revisión de los antecedentes y el contexto histórico del uso de la IA y Big Data en procesos judiciales, para obtener una base sólida para la investigación.
- Un análisis del paisaje actual de la IA en el ámbito judicial.
- Un análisis de las tendencias futuras en el uso de IA y Big Data en el ámbito jurídico.
- La identificación de desafíos éticos y legales.

## 3. METODOLOGÍA

Para alcanzar el objetivo principal y secundarios de este trabajo el estudio se basará en un enfoque multidisciplinario que combina aspectos legales, éticos y tecnológicos. Primero, se trabajará en la revisión de la literatura. Segundo, se realizará un estudio de casos. Por último, se dará el proceso de la construcción del modelo predictivo.

### 3.1. Revisión de Literatura

Se llevará a cabo una revisión exhaustiva de la literatura para establecer una base teórica y contextual para la investigación. Esto va a incluir:

- Un análisis de los antecedentes y el contexto histórico del uso de IA y Big Data en procesos judiciales, lo que proporcionará una comprensión profunda de cómo estas tecnologías han evolucionado y se han aplicado hasta ahora.
- Un análisis del paisaje actual de la IA en el ámbito judicial, tanto a nivel global como específicamente en la Unión Europea, identificando las aplicaciones actuales, sus resultados y las tecnologías más utilizadas.
- Una revisión de las tendencias futuras en el uso de IA y Big Data en el ámbito jurídico, explorando las oportunidades y desafíos que se vislumbran para estas tecnologías en el futuro cercano.

- La identificación de desafíos éticos y legales asociados al uso de IA en decisiones judiciales, incluyendo cuestiones de privacidad, sesgo en los algoritmos, transparencia y el marco regulatorio actual.

### **3.2. Estudio de Casos**

Se van a estudiar casos ya existentes del uso de tecnologías en procesos judiciales tanto a nivel global como a nivel europeo. Este método es particularmente adecuado para el presente estudio, ya que permite el examen sistemático de comportamientos y prácticas judiciales del mundo real. Para ello:

- Se analizarán casos relevantes que han ilustrado el impacto de la IA en el campo de modelos predictivos de resultados judiciales, tanto en términos de riesgos como de éxitos.
- Se analizarán también nuevas tecnologías que ya se están poniendo en marcha en distintos lugares para ayudar a abogados y jueces en la toma de decisiones.

### **3.3. Elaboración de un Modelo Predictivo**

Se desarrollará un modelo predictivo utilizando técnicas de aprendizaje automático con Python 3. Este proceso incluirá:

- La recopilación y limpieza de datos públicos de decisiones del TJUE, asegurando la precisión y la eliminación de errores.
- El procesamiento de los datos para transformarlos en un formato adecuado para el análisis, incluyendo la normalización y la codificación de variables relevantes.
- El diseño del modelo predictivo, seleccionando algoritmos apropiados (como árboles de decisión) y definiendo los parámetros de entrenamiento.
- El entrenamiento y validación del modelo utilizando un conjunto de datos de entrenamiento y prueba, respectivamente, para evaluar su precisión y ajustar los parámetros según sea necesario.
- La interpretación de los resultados obtenidos, identificando las variables más relevantes y evaluando la eficacia del modelo en la predicción de sentencias judiciales.

Estas metodologías integradas permitirán desarrollar un modelo predictivo preciso y situar sus resultados dentro del contexto más amplio del uso de IA en el ámbito judicial, abordando tanto las oportunidades como los desafíos éticos y legales.

## CAPÍTULO II. LA IA Y BIG DATA EN LOS PROCESOS JUDICIALES

### 1. CONCEPTO

**Usuario:** "¿Qué es la Inteligencia Artificial?"

**ChatGPT [GPT-3]:** " La Inteligencia Artificial (IA) es un campo de la informática que busca desarrollar sistemas capaces de realizar tareas inteligentes, como aprendizaje, razonamiento y reconocimiento de patrones, imitando funciones humanas. Se clasifica en IA débil, especializada en tareas específicas, y IA fuerte, que aspira a igualar o superar la inteligencia humana."

**Usuario:** "¿Y el Big Data?"

**ChatGPT [GPT-3]:** "El Big Data se refiere al manejo y análisis de conjuntos de datos extremadamente grandes y complejos que superan la capacidad de las herramientas tradicionales de procesamiento de datos. Estos conjuntos de datos suelen caracterizarse por su volumen, velocidad y variedad, y a menudo provienen de diversas fuentes.

En conjunto, la IA y el Big Data pueden potenciarse mutuamente. La IA puede utilizar técnicas de análisis avanzado para extraer información valiosa de grandes conjuntos de datos, y el Big Data proporciona la cantidad masiva de información necesaria para entrenar y mejorar los modelos de IA. La combinación de ambas tecnologías se utiliza en diversas aplicaciones, como el análisis predictivo y toma de decisiones automatizada."

En efecto, la IA y el Big Data están causando gran impacto en la comunidad legal, representando un avance en la integración de la tecnología en el ámbito jurídico. Aunque pueda parecer un desarrollo reciente, la IA y los chatbots tienen una larga historia, remontándose a la década de 1960 con programas como ELIZA y ALICE (Weizenbaum, 1996).

El notable rendimiento de la IA en tareas analíticas y creativas, a menudo supera ya las habilidades humanas. Por ejemplo, GPT-4, una iteración reciente, hizo el LSAT y el Examen Uniforme de Abogacía, ubicándose entre el 10% superior de los examinados (Katz, 2023).

#### 1.1. Aprendizaje automático o Machine Learning

El aprendizaje de máquinas, también conocido como aprendizaje automático o Machine Learning (ML), se define como el proceso de extraer conocimiento a partir de datos, constituyendo un área de investigación que abarca las ciencias estadísticas, la inteligencia artificial y la informática. Según la definición de Tom M. Mitchell, profesor fundador del departamento de ML en la Universidad Carnegie Mellon, el aprendizaje automático comprende el estudio de algoritmos informáticos que mejoran automáticamente a medida que adquieren experiencia (Cervellini & Carbone, 2019).

Este aprendizaje automático se puede dividir en tres grupos según el objetivo y los datos disponibles (Universidad Europea, 2022) (The MathWorks, Inc., s.f.):

**a) Aprendizaje supervisado:** requiere datos etiquetados durante el entrenamiento del modelo. Los científicos de datos etiquetan estos datos para su uso, lo que puede ser un proceso intensivo en recursos. Esta técnica permite predecir resultados futuros a partir de patrones en los datos. Se usa para clasificar archivos y prever tendencias mediante la identificación de patrones en datos de entrada y salida. Ejemplos de algoritmos de aprendizaje supervisado son Regresión Lineal, Regresión Logística, Regresión de Árbol de Decisión, K-Vecinos Más Cercanos (KNN), Máquina de Vectores de Soporte (SVM) o Clasificación de Árbol de Decisión.

**b) Aprendizaje no supervisado:** en contraste, en el aprendizaje no supervisado no hay distinción entre los datos de entrenamiento y prueba. Este tipo implica entrenar modelos con datos sin etiquetar, requiriendo menos intervención humana que el aprendizaje supervisado. Aunque se establecen parámetros, como la cantidad de puntos de clúster, el modelo puede procesar grandes conjuntos de datos sin supervisión directa. Algunos ejemplos son K-Means Clustering, Agrupamiento Jerárquico, Análisis de Componentes Principales, entre otros.

**c) Aprendizaje por refuerzo (Reinforcement Learning):** en este caso, las tareas se aprenden a ejecutar a través de prueba y error en un entorno dinámico. Este enfoque permite al agente informático tomar decisiones que buscan aumentar al máximo una medida de recompensa por tarea realizada, sin intervención humana ni programación explícita. Estos programas de inteligencia artificial destacan en juegos estratégicos y videojuegos, como Go y el ajedrez, superando a los humanos. Ejemplos de (The MathWorks, Inc., s.f.) este tipo de algoritmos son Q-Learning, Proximal Policy Optimization (PPO) o Monte Carlo Methods.

## 1.2. Modelos predictivos

Las predicciones constituyen un elemento fundamental en la esfera del Big Data, ya que implica la aplicación de principios matemáticos a conjuntos masivos de información para inferir probabilidades (IFD Technologies, 2021). El concepto de anticipación del futuro se vincula con la libertad de acción o el libre albedrío, aunque el Big Data posibilita la realización de pronósticos sobre el comportamiento humano y mejora las clásicas predicciones estadísticas, aunque al igual que estas también presenta un porcentaje de error.

Dos indicadores esenciales de la actual aceleración tecnológica son: la ley de Moore (Anexo 2), que establece que la cantidad de transistores en microchips se duplica cada dos años (un patrón consistente durante las últimas cuatro décadas y que se espera mantenga su vigencia en los próximos años) (Kelleher, 2022); y la ley de Kryder (Anexo 3), que indica que el costo de almacenamiento por unidad de información se reduce a la mitad cada dos años (Walter, 2005).

En el ámbito de la probabilidad y la estadística, la correlación indica la fuerza y dirección de la relación lineal y proporcional entre dos variables estadísticas. Si esta relación es positiva implica que los valores de una variable varían sistemáticamente en relación con los valores correspondientes de la otra: si una disminuye, la otra también, y viceversa. Sin embargo, la correlación no implica automáticamente una relación causal entre estas variables (Ferrero, 2020).

### 1.3. Algoritmos

Los algoritmos, según la definición del Diccionario de la Real Academia Española, son secuencias organizadas y finitas de operaciones que permiten resolver problemas. Etimológicamente, el término se origina en el Bagdad del siglo IX, donde un matemático, Mohamed ben Musa, también conocido como Al-khwarizmi, daba clases. De su sobrenombre deriva la palabra "alguarismo", que posteriormente, influenciada por el latín medieval y la palabra "aritmio" (que significa número en griego), se transformó en "algoritmo". Los algoritmos, presentan una serie de características como su capacidad de ser escritos en un lenguaje específico, su ejecución paso a paso, su determinación estricta por la entrada de datos y los resultados previos, su finalización en un número finito de pasos independientemente de los datos de entrada o su implementación física en computadoras (Fernández, 2020).

En los primeros diseños de algoritmos, los programadores incluían los criterios para tomar decisiones. En cambio, en los algoritmos dinámicos, tienen la capacidad de aprender con el tiempo a partir de datos y experiencias, tomando decisiones de manera autónoma y generando sus propias instrucciones, distintas de las aplicadas previamente por el programador.

Un paso adicional en esta evolución es el "aprendizaje profundo" o "deep learning", que implica el funcionamiento de la IA imitando redes neuronales complejas. En estos casos, los algoritmos identifican patrones en grandes volúmenes de datos masivos, caracterizados por su cantidad, variedad y velocidad (conocidos como las tres V). Estos algoritmos producen resultados complejos que no están linealmente vinculados, lo que hace difícil establecer la causalidad entre los datos manejados y la decisión final. Esta evolución supone que la toma de decisiones ya sea acertada o no, se aleja del control humano, dado que la lógica que siguen los algoritmos se vuelve opaca, convirtiéndose en una "caja negra" para sus creadores, incapaces de seguir el proceso desde su inicio hasta su fin (Farnós, 2015).

Los algoritmos más complejos, al ser interpretaciones matemáticas de los datos, no proporcionan una explicación de la realidad subyacente que generó esos datos. Los artefactos o robots que operan con base en algoritmos presentan un cierto grado de autonomía e imprevisibilidad. Esta revolución tecnológica genera desafíos sin respuesta en la regulación existente (Mascitti, 2020).

En conclusión, las nuevas tecnologías del siglo XXI podrían alterar la perspectiva humanista, otorgando poder a algoritmos no humanos y reduciendo la autoridad humana.

## 2. HITOS IMPORTANTES

La profesora Katie Atkinson de la Universidad de Liverpool fue una de las pioneras en el desarrollo de algoritmos predictivos de resultados. Un ejemplo icónico que exploró fue el incidente en el año 2001 en San Francisco, donde una bola de jonrón de Barry Bonds se convirtió en un objeto de gran valor. En medio de la disputa sobre quién era el legítimo propietario (Mr. Hayashi o Mr. Popov) la Profesora Atkinson, experta en inteligencia artificial, utilizó modelos computacionales de argumentos para resolver la cuestión (BBC, 2016).

Su programa de IA examinó los argumentos involucrados, mostrando que podía generar razonamientos similares a los que hubieran tenido los jueces. En este contexto, el modelo desarrollado por Atkinson presentó razonamientos tanto a favor de Mr. Hayashi como de Mr. Popov, concluyendo que ambos eran igualmente aceptables.

La Corte Superior de California, siguiendo un enfoque similar, decidió dividir los ingresos obtenidos por la venta de la bola entre ambas partes, reconociendo la validez de los argumentos de cada una. Este caso, uno de los primeros casos analizados por el modelo de Atkinson, demostró la capacidad de la IA para analizar y resolver cuestiones legales de manera imparcial y eficaz (Popov v. Hayashi , 2002).

En casos posteriores analizados por el programa de Atkinson, la IA coincidió en un 96% de las ocasiones con las decisiones tomadas en situaciones reales. Esta alta precisión nos hace plantearnos sobre la probabilidad de error humano en el 4% restante, sugiriendo la posibilidad de que la máquina no haya fallado, sino que el error podría haber sido atribuible al juez en esos casos específicos.

Otro ensayo innovador en este ámbito surgió a principios del siglo con la creación de un algoritmo diseñado para anticipar las decisiones del Tribunal Supremo de los Estados Unidos (Ruger, Kim, Martin, & Quinn, 2004). Este algoritmo buscaba predecir tanto los votos individuales de los nueve jueces como la orientación final de las decisiones en todos los casos atendidos por el tribunal durante el año judicial 2002-2003. Para entrenar el algoritmo, se utilizaron los resultados de 628 casos resueltos por el "tribunal Rehnquist", cuya composición se mantuvo inusualmente constante desde 1994. El algoritmo se basaba en un árbol de clasificación<sup>2</sup> que consideraba seis variables: el circuito federal de origen del caso, el tema en cuestión, el tipo de demandante y demandado, la tendencia ideológica (liberal o conservadora) del precedente fijado por el tribunal inferior y si cuestionaba o no la constitucionalidad de una norma o práctica.

Un aspecto destacado de este experimento fue la comparación entre las predicciones del algoritmo y las realizadas por un grupo de académicos y abogados expertos en la doctrina del tribunal (un panel de tres expertos predecía cada decisión). Los resultados fueron sorprendentes: de los 78 asuntos revisados por el Tribunal Supremo durante ese año judicial, el algoritmo predijo correctamente el 75% de sus decisiones y el 66,7% de los votos individuales de sus jueces, mientras que los expertos acertaron en un 59% y 67,9%

---

<sup>2</sup> Un árbol de clasificación es una estructura de decisiones en forma de árbol que separa datos en diferentes categorías basándose en preguntas sobre sus características. Es usado en aprendizaje automático para clasificar ejemplos en diferentes grupos o clases.

respectivamente. Nuevamente nos podemos plantear si la IA supera ya la inteligencia humana, si comete menos errores y lo que esto puede suponer para nuestro futuro.

### 3. PAISAJE ACTUAL

En 2016 un grupo de académicos británicos y estadounidenses crearon un algoritmo, diseñado para anticipar las decisiones del Tribunal Europeo de Derechos Humanos (TEDH) respecto a la aplicación de los artículos 3<sup>3</sup>, 6<sup>4</sup> y 8<sup>5</sup> del Convenio Europeo de Derechos Humanos (CEDH) (Aletras, Tsarapatsanis, Preoțiuc-Pietro, & Lampos, 2016). La elección de este tribunal no fue al azar, ya que sus sentencias presentan una estructura sistemática distintiva con secciones y subsecciones estandarizadas, lo que facilita un tratamiento y análisis automatizado de los textos. Para desarrollar este modelo predictivo, se emplearon 584 decisiones, extrayendo información específica de secciones clave como: (a) "procedimiento"; (b) "los hechos", subdivididos en "circunstancias del caso" y "Derecho relevante" (leyes nacionales, principalmente); y (c) "el Derecho", con subsecciones "argumentos de las partes" y "méritos" (justificación de la decisión del tribunal). Se tomaron ciertos elementos textuales como atributos (n-gramas<sup>6</sup> y clustering<sup>7</sup>) para entrenar un algoritmo de clasificación basado en "máquinas de vectores soporte"<sup>8</sup>.

Este algoritmo logró alcanzar una precisión del 79% en la predicción de los resultados. Además, un hallazgo significativo del estudio fue que la descripción de los hechos y el contexto casuístico contenidos en la subsección "circunstancias del caso" resultó ser la información más relevante para predecir adecuadamente las sentencias, muestra que los detalles específicos del caso son cruciales en la decisión final del tribunal. Esto evidencia que las decisiones no se basan únicamente en la aplicación objetiva de la ley, sino en cómo esos principios se aplican a situaciones concretas, respaldando la teoría del realismo jurídico americano<sup>9</sup> sobre la importancia fundamental de los hechos y el contexto de un caso en la toma de decisiones judiciales (Cayón, 2020).

---

<sup>3</sup> Art 3 CEDH sobre la prohibición de la tortura

<sup>4</sup> Art 6 CEDH sobre el derecho a un proceso equitativo

<sup>5</sup> Art 8 CEDH sobre el derecho al respeto a la vida privada y familiar

<sup>6</sup> Un n-grama es una subsecuencia de n elementos (como palabras o caracteres) consecutivos en un texto, usado para analizar patrones y frecuencias en el lenguaje.

<sup>7</sup> El clustering es una técnica que agrupa datos similares en conjuntos o clusters para identificar patrones o estructuras en la información.

<sup>8</sup> El algoritmo de clasificación basado en "máquinas de vectores de soporte" (SVM, por sus siglas en inglés) es una técnica de aprendizaje automático utilizada para clasificar o predecir datos. Funciona encontrando un límite o hiperplano óptimo que separa los datos en clases diferentes en un espacio multidimensional.

<sup>9</sup> La teoría del realismo jurídico americano cuestiona la idea de que los jueces son siempre objetivos, argumentando que las decisiones judiciales están influenciadas por factores subjetivos y contextuales, como las experiencias personales y las circunstancias del caso.

### 3.1. En el mundo

Los algoritmos que pueden manejar grandes volúmenes de datos cada vez se valoran mejor, especialmente en el ámbito legal. En situaciones legales complejas con extensos documentos, estos sistemas superan a abogados menos experimentados al identificar con precisión documentos relevantes (Gobierno de Nueva Gales del Sur, 2022). Este avance es relevante hoy en día, ya que los litigios involucran frecuentemente largos hilos de correos electrónicos entre empresas y partes. En el pasado, los abogados invertían horas revisando documentos físicos en salas supervisadas, pero ahora, los sistemas automatizados pueden realizar estas revisiones de forma más rentable y con un estándar más elevado.

Por otro lado, en la región de Hebei, al norte de China, 178 tribunales locales han adoptado una aplicación llamada Intelligent Trial 1.0 con inteligencia artificial desde julio de 2016. Esta herramienta tecnológica ha reducido considerablemente la carga laboral de los jueces, permitiendo a casi 3.000 jueces manejar más de 150.000 casos y reducir su carga laboral en un tercio. Estos avances forman parte de la ambiciosa iniciativa de China de liderar el campo de la inteligencia artificial a nivel mundial para el año 2030. Además, China ha puesto en marcha el portal web "China Judgments Online", que contiene una vasta base de datos con más de 120 millones de documentos legales. Recientemente, los tribunales chinos han estado experimentando con la incorporación de la IA en sus procesos judiciales, utilizando software que examina pruebas, sugiere resultados y asegura la coherencia en las sentencias (E. Stern, L. Liebman, Z. Wang, & Roberts, 2021).

Finalmente, en Latinoamérica destaca el Laboratorio de Innovación e Inteligencia Artificial de la Facultad de Derecho de Buenos Aires (IALAB), que ha logrado avances significativos en Buenos Aires y Colombia con el sistema *Prometea* (Mutualidad, 2022). Con una precisión del 96%, *Prometea* puede predecir la resolución de un caso judicial en tan solo 20 segundos. En el contexto administrativo de Buenos Aires, esto se traduce en la elaboración de aproximadamente mil dictámenes legales en el campo de vivienda en solo 45 días, en contraste con los 174 días que requeriría un procedimiento estándar.

Además, en Buenos Aires también nace Pretoria, una plataforma que toma su nombre del auxiliar de la justicia romana "Pretor". Pretoria constituye un sistema que se incorpora al conjunto de soluciones digitales. Su propósito es respaldar y mejorar el procedimiento de selección, análisis y organización de las sentencias de tutela destinadas a ser revisadas por la Corte Constitucional. Esta plataforma ha reducido el tiempo para la selección de casos urgentes de 96 días a tan solo dos minutos. Bajo supervisión humana, el sistema *Prometea* (programa en el que se basa Pretoria) determina la prioridad de cada expediente, generando hasta 14 documentos necesarios en 16 minutos, una tarea que tomaría aproximadamente 2 horas y 40 minutos para un funcionario experimentado. Es así reconocido como el primer sistema de este tipo utilizado en un tribunal judicial de máximo nivel a escala mundial.

### 3.2. En la UE

En 2019, la revista empresarial Wired informó que el Ministerio de Justicia de Estonia estaba diseñando un “juez robot” que adjudicaría disputas de reclamaciones pequeñas. Las partes cargarían documentos y otra información relevante, y la IA emitiría una decisión. El título del artículo planteaba la pregunta '¿Puede la IA ser un juez imparcial en un tribunal?'. Más tarde, el Ministerio de Justicia de Estonia aclaró que de hecho no estaban desarrollando un juez robot de IA. Sin embargo, están investigando formas de utilizar la IA para automatizar ciertos procedimientos (Tuulik, 2022).

Aunque el juez robot aún no existe, ni en Estonia ni en ningún otro país, el uso de la IA en el campo legal está creciendo rápidamente. La tecnología ya está encontrando su camino en los tribunales de diversas maneras, desde un mediador que afirma que una aplicación de negociación basada en la IA se utilizó para resolver un caso judicial hasta una empresa que afirma que su aplicación basada en la IA anuló 160,000 multas de estacionamiento en Londres y Nueva York en sus primeros 21 meses en el mercado.

Y actualmente se está considerando un sistema judicial en línea en Inglaterra y Gales, que podría aprovechar la tecnología informática para ampliar el acceso a la justicia (Gibbs, 2016).

## 4. PERSPECTIVAS

A medida que los modelos de predicción de resultados judiciales y sistemas de ayuda para los jueces son cada vez más adoptados por jurisdicciones que buscan modernizar y agilizar sus procesos, se anticipa que estas herramientas comenzarán a incorporar más tipos de datos y enfoques de modelado más complicados. Podemos esperar ver más herramientas desarrolladas a través de un proceso de entrenamiento completamente algorítmico, sin depender del tipo de procesamiento manual que es común en la mayoría de las herramientas actuales (Chouldechova & Lum, 2020).

### 4.1. Nuevas fuentes de datos

“La única manera de predecir el futuro es inventarlo”, y aunque no podemos estar seguros de lo que está por venir, hay algunas pistas. La primera se encuentra en la doctrina académica, donde expertos proponen nuevas técnicas para la predicción de riesgos que aún no se han implementado ampliamente. El segundo lugar al que recurrimos son otros modelos predictivos relacionados con la justicia, como la predicción policial (análisis de redes sociales, predicción de reincidentes, etc).

Las nuevas y diversas fuentes de datos no resuelven problemas fundamentales relacionados con la representatividad, o sesgo. Hasta ahora, las herramientas de evaluación de riesgos previos al juicio han dependido de factores calculables a partir de datos de los tribunales y la administración. Sin embargo, a medida que se recopila,

almacena y procesa más información sobre individuos, pueden añadirse diversas y nuevas fuentes de información a las ya utilizadas en la evaluación de un juicio. Entre estas nuevas fuentes se pueden encontrar datos administrativos gubernamentales que se recopilan cada vez más, se sistematizan y se agrupan en conjuntos de datos unificados. Por ejemplo, se pueden fusionar datos recopilados por diversas entidades gubernamentales, como los cuerpos de seguridad, servicios sociales, salud mental y servicios para niños y familias, para crear registros unificados que abarcan información sobre un mismo individuo en estas diferentes fuentes gubernamentales. También se está considerando utilizar datos comerciales, como los provenientes de redes sociales, ejecuciones hipotecarias, estacionamientos de pago o datos telefónicos para algunos sistemas de predicción policial. En particular, en Estados Unidos, se ha sugerido la intención del gobierno de utilizar las redes sociales como parte del perfil algorítmico en aras de la seguridad pública, según anuncios del Departamento de Seguridad Nacional americano (Chouldechova & Lum, 2020).

#### **4.2. Nuevos horizontes**

¿Estamos cerca de un sistema judicial que tome decisiones por IA? Se ha debatido sobre este tema, numerosos expertos en tecnología, jueces y personas vinculadas a la judicatura han expresado sus opiniones, unas favorables otras menos, sobre la viabilidad de que un robot con IA tome decisiones en el ámbito judicial.

Desde un punto de vista más personal, abundan juicios de todo tipo, pero actualmente no tenemos la capacidad de responder a la pregunta a ciencia cierta. Pero sí podemos imaginarnos una sociedad en la que este cambio fuese posible, y si es así, como sería esta sociedad ¿estaríamos ante un cambio positivo? Como ya se ha mencionado en el punto 3 de este Capítulo, esta tecnología ya se está implementando en algunas partes del mundo, por lo que tecnológicamente, sí es posible. En cuanto a la transformación de nuestra sociedad, habría que debatir los problemas que esta tecnología conllevaría: sesgos, falta de transparencia y problemas con la privacidad de datos. Todo esto se analizará con más detenimiento en el Capítulo IV de este trabajo.

Finalmente, hay opiniones más valiosas que otras, por ejemplo, la opinión del juez Alfonso Peralta Gutiérrez, juez titular del juzgado de primera instancia nº2 de Guadix y experto en Derecho Digital, responde justamente a esta pregunta sobre el futuro de los modelos predictivos judiciales, sugiere que la presencia de un robot con IA tomando decisiones en los tribunales de justicia no parece inminente. Sin embargo, se especula sobre la posibilidad de que una IA pueda contribuir como auxiliar en la toma de decisiones, sin sustituir al juez y sin tomar decisiones propias. En palabras de Ariza, “se intenta incorporar una herramienta en forma de algoritmos, que coadyuva en la decisión judicial” y dando a entender que este sistema “puede contribuir a establecer líneas jurisprudenciales, sus manifestaciones, existencia de jurisprudencia contradictoria, mayoritaria, etc” (Ariza, M., op. cit., p. 53).

## CAPÍTULO III. MODELO PREDICTIVO DE SENTENCIAS

### 1. METODOLOGÍA

#### 1.1. Fuente de datos

La IUROPA CJEU Database Platform va a constituir nuestra base datos, desarrollada por el Proyecto IUROPA, esta base de datos es la más extensa y completa orientada a la investigación sobre el Tribunal de Justicia de la Unión Europea (TJUE). La base de datos incluye información detallada sobre la totalidad de los casos, procedimientos, decisiones y jueces del TJUE, recopilada de fuentes oficiales como el Registro del Tribunal, InfoCuria (la base de datos oficial del Tribunal) y EUR-Lex (la base de datos oficial de documentos legales de la UE).

La integración de datos de estas fuentes garantiza una cobertura completa y precisa de la actividad del TJUE, proporcionando una base sólida para el análisis y la generación de un modelo desplegable.

#### 1.2. Recopilación y limpieza de datos

La recopilación y limpieza de datos constituyen etapas esenciales en cualquier proyecto de análisis de Big Data. Antes de que los datos puedan ser utilizados para construir modelos predictivos precisos y fiables, es fundamental asegurarse de que estos datos sean recopilados de manera sistemática y limpiados adecuadamente para eliminar errores, duplicados y datos irrelevantes. Este proceso garantiza que las predicciones y análisis posteriores se basen en información de alta calidad, proporcionando resultados más precisos y útiles.

Para este trabajo, los datos han sido filtrados específicamente para incluir únicamente los casos que involucran a instituciones de la UE, agencias ejecutivas de la UE y, de manera más significativa, a personas jurídicas, en el período comprendido entre los años 1953-2022.

Una vez filtrados, se han transformado los datos a un formato csv que es más fácil de trabajar en Python 3.

Se ha realizado también una revisión de los valores nulos en el conjunto de datos para garantizar su integridad. El método `isnull().sum()` aplicado al DataFrame proporciona el conteo de valores nulos por cada columna, la información resultante se encuentra en la figura 1.

Figura 1. Visualización de valores nulos

```
# Revisar si hay valores nulos
df.isnull().sum()

key_id          0
court_id        0
court           0
iuropa_case_id  0
cjeu_case_id    0
case_year       0
case_number     0
case_name       1
is_pending      0
is_removed      0
is_transferred  0
transferred_to  0
is_joined       0
joined_to       0
count_on_appeal 0
list_on_appeal  0
count_appeal_of 0
list_appeal_of  0
count_proceedings 0
list_proceedings 0
count_decisions 0
list_decisions  0
has_judgment    0
has_opinion     0
party_role      0
iuropa_party_id 0
party_type      0
dtype: int64
```

Fuente: *Tabla Python 3.*  
Elaboración propia

Para asegurar la consistencia en el manejo de las columnas, se realizó una limpieza de los nombres de estas. Se eliminaron los espacios en blanco y los guiones, y se mantuvo la consistencia en el estilo de escritura (por ejemplo, todas las letras en minúsculas).

### 1.3. Descripción de datos del modelo

#### 1.3.1. Variables a estudiar

Para este estudio, se han seleccionado las siguientes variables del conjunto de datos proporcionado por la plataforma IUROPA CJEU Database:

- *is\_pending*: Variable dummy<sup>10</sup> que indica si el procedimiento principal del caso está pendiente al 31 de diciembre de 2022. Es relevante porque los casos pendientes tienen una dinámica diferente en comparación con los casos ya resueltos, lo que podría afectar la probabilidad de que se emita una sentencia.
- *is\_removed*: Variable dummy que indica si el procedimiento principal del caso ha sido eliminado del Registro. Es importante porque los casos eliminados no avanzan a sentencia, por lo tanto, esta variable ayuda a identificar y excluir estos casos del análisis predictivo.
- *is\_joined*: Variable dummy que indica si el procedimiento principal del caso se ha unido a otro caso. Este dato es crucial ya que los casos unidos pueden tener un tratamiento especial en el sistema judicial, afectando su probabilidad de llegar a sentencia.
- *count\_on\_appeal*: Variable entera<sup>11</sup> que representa el número de apelaciones relacionadas con cada caso. Es importante porque los casos con múltiples apelaciones tienen una mayor probabilidad de resultar en sentencias debido a su complejidad y al proceso de revisión continuo y que, en general, si hay un valor mayor que cero en esta variable, implica que hubo al menos una sentencia o decisión previa que fue apelada.
- *count\_appeal\_of*: Variable entera que indica el número de casos que están siendo apelados por el procedimiento principal. Normalmente, esto también implica la existencia de decisiones o sentencias en esos casos.
- *count\_proceedings*: Variable entera que representa el número de procedimientos asociados a este caso. Esta variable es relevante ya que un mayor número de procedimientos puede indicar una mayor actividad judicial y, por ende, una mayor probabilidad de sentencia.
- *count\_decisions*: Variable entera que indica el número de decisiones que la Corte ha emitido respecto al caso. Es crucial porque un mayor número de decisiones puede ser un indicativo de un caso activo y con alta probabilidad de llegar a una sentencia.
- *has\_opinion*: Variable dummy que indica si hubo una opinión del Abogado General relacionada con el caso. Las opiniones del Abogado General son

---

<sup>10</sup> Toma únicamente 2 valores, típicamente 0 y 1.

<sup>11</sup> Toma valores numéricos enteros, sin decimales.

importantes y frecuentemente influyen en las sentencias de la Corte, por lo que esta variable puede ser un fuerte predictor de la existencia de una sentencia.

Las variables seleccionadas para el análisis se centran en aspectos del proceso judicial y características del caso que pueden influir directamente en la probabilidad de que un caso tenga una sentencia que es lo que buscamos predecir.

### 1.3.2. Variable objetivo

- *has\_judgment*: Variable dummy que indica si hubo un juicio relacionado con el caso. Mas concretamente, se refiere a si hay al menos una sentencia del TJUE relacionada con un caso. Si es así, la variable "has\_judgment" se establece como verdadera (1); de lo contrario, se establece como falsa (0).

Esta es la variable objetivo o “target” de nuestro modelo, es decir, la variable dependiente que queremos predecir basándonos en las otras variables del conjunto de datos. Si la variable target indica un (0) significa que no hay sentencia y esto se puede deber a múltiples razones como el desistimiento de las partes, allanamiento, la existencia de un acuerdo extrajudicial o incluso la inadmisión final por el propio Tribunal, por ejemplo, por considerarse incompetente.

### 1.3.3. Descripción de las demás variables del dataset

A continuación, se presenta una descripción detallada de las demás variables incluidas en el dataset que no se han incluido en el modelo predictivo:

- *key\_id* (entera): Un número de identificación que identifica de manera única cada observación. No es relevante para el modelo ya que es puramente un identificador y no aporta valor predictivo.
- *court\_id* (entera): Un número de identificación que identifica de manera única cada corte. Al igual que *key\_id*, es un identificador y no tiene valor predictivo directo.
- *court* (string<sup>12</sup>): La corte en la que se presentó el caso. Aunque puede ofrecer alguna información contextual, no se considera directamente relevante para predecir una sentencia.
- *iuropa\_case\_id* (string): El número de identificación del caso en la base de datos IUROPA, que identifica de manera única cada caso.
- *cjeu\_case\_id* (string): El número de identificación del caso en la CJEU, que identifica de manera única cada caso. Esta y la anterior son identificadores sin valor predictivo.
- *case\_year* (entera): El año en que se presentó el caso, codificado en base a *cjeu\_case\_id*. Si bien puede proporcionar alguna información temporal, no se selecciona para este análisis específico.

---

<sup>12</sup> Almacena una secuencia de caracteres, como palabras, frases o cualquier texto.

- *case\_number* (entera): El número del caso. Los números de caso se reinician a 1 al comienzo de cada año. Similar a los identificadores, no tiene valor predictivo.
- *case\_name* (string): El nombre del caso según el Registro. Es una etiqueta descriptiva sin valor predictivo.
- *transferred\_to* (string): Si el procedimiento principal del caso ha sido transferido a otra corte, esta variable contiene el *cjeu\_case\_id* del caso después de la transferencia. Aunque puede ser interesante, no se considera esencial para este análisis.
- *joined\_to* (string): Si el procedimiento principal del caso se ha unido a otro caso, esta variable contiene el *cjeu\_case\_id* del caso de referencia. Esta información ya se captura en *is\_joined*, por lo que es redundante.
- *list\_on\_appeal* (string): Una lista de las apelaciones relacionadas con cada caso, separadas por comas.
- *list\_appeal\_of* (string): Si el caso es una apelación, una lista de los casos que están siendo apelados, separada por comas.
- *list\_proceedings* (string): Una lista de los procedimientos asociados con este caso, separados por comas.
- *list\_decisions* (string): Una lista de las decisiones que la Corte ha emitido respecto al caso, separadas por comas. Estas últimas 4 variables contienen listas de datos relacionadas con apelaciones, procedimientos y decisiones, respectivamente. Aunque pueden ofrecer información detallada, para este análisis se consideran más prácticas las variables numéricas derivadas de estas listas (como “*count\_on\_appeal*” y “*count\_decisions*”) para simplificar el análisis y evitar la complejidad de manejar listas de texto.

Estas variables no se incluyen en el análisis porque no aportan información directamente relevante para predecir la existencia de una sentencia o son identificadores que no influyen en el resultado del caso.

#### 1.3.4. Dimensión y descripción estadística de los datos

El conjunto de datos cuenta con un total de 30,639 observaciones y 27 variables. Del total de variables se van a estudiar las 8 variables mencionadas en el apartado 1.3.1. más la variable target “*has\_judgement*” y se va a analizar la relación entre 3 de ellas (“*count\_decisions*”, “*count\_on\_appeal*” y la target).

La dimensión de los datos se obtiene en Python mediante el método *shape* aplicado al DataFrame de pandas que contiene los datos de la siguiente manera:

```
# Obetener la estructura de los datos
# Describir los datos
df.shape
```

(30639, 27)

Además, en las figuras 2 y 3. se presenta una descripción estadística de las variables utilizadas en el modelo. La función *describe()* proporciona estadísticas resumidas para las

variables numéricas, incluyendo el conteo de observaciones, la media, la desviación estándar, los valores mínimos, los percentiles 25, 50 y 75, y los valores máximos.

**Figura 2. Descripción estadística de las variables**

```
# DESCRIBIR LOS DATOS
descr=df.describe()
descr
```

|              | key_id       | court_id     | case_year    | case_number  | is_pending   | is_removed   | is_transferred |
|--------------|--------------|--------------|--------------|--------------|--------------|--------------|----------------|
| <b>count</b> | 30639.000000 | 30639.000000 | 30639.000000 | 30639.000000 | 30639.000000 | 30639.000000 | 30639.000000   |
| <b>mean</b>  | 15320.000000 | 1.155912     | 2000.365123  | 278.033324   | 0.036457     | 0.194556     | 0.020954       |
| <b>std</b>   | 8844.861785  | 0.362778     | 14.252507    | 220.090714   | 0.187427     | 0.395865     | 0.143232       |
| <b>min</b>   | 1.000000     | 1.000000     | 1953.000000  | 1.000000     | 0.000000     | 0.000000     | 0.000000       |
| <b>25%</b>   | 7660.500000  | 1.000000     | 1991.000000  | 104.000000   | 0.000000     | 0.000000     | 0.000000       |
| <b>50%</b>   | 15320.000000 | 1.000000     | 2001.000000  | 236.000000   | 0.000000     | 0.000000     | 0.000000       |
| <b>75%</b>   | 22979.500000 | 1.000000     | 2013.000000  | 397.000000   | 0.000000     | 0.000000     | 0.000000       |
| <b>max</b>   | 30639.000000 | 2.000000     | 2022.000000  | 1322.000000  | 1.000000     | 1.000000     | 1.000000       |

Fuente: Tabla Python 3. Elaboración propia

**Figura 3. Descripción estadística de las variables**

|              | is_joined    | count_on_appeal | count_appeal_of | count_proceedings | count_decisions | has_judgment | has_         |
|--------------|--------------|-----------------|-----------------|-------------------|-----------------|--------------|--------------|
| <b>count</b> | 30639.000000 | 30639.000000    | 30639.000000    | 30639.000000      | 30639.000000    | 30639.000000 | 30639.000000 |
| <b>mean</b>  | 0.157087     | 0.028167        | 0.154574        | 1.036228          | 1.417703        | 0.603675     | 0.399034     |
| <b>std</b>   | 0.363889     | 0.228000        | 0.927525        | 0.201161          | 0.747218        | 0.489141     | 0.489708     |
| <b>min</b>   | 0.000000     | 0.000000        | 0.000000        | 1.000000          | 0.000000        | 0.000000     | 0.000000     |
| <b>25%</b>   | 0.000000     | 0.000000        | 0.000000        | 1.000000          | 1.000000        | 0.000000     | 0.000000     |
| <b>50%</b>   | 0.000000     | 0.000000        | 0.000000        | 1.000000          | 1.000000        | 1.000000     | 0.000000     |
| <b>75%</b>   | 0.000000     | 0.000000        | 0.000000        | 1.000000          | 2.000000        | 1.000000     | 1.000000     |
| <b>max</b>   | 1.000000     | 14.000000       | 49.000000       | 6.000000          | 9.000000        | 1.000000     | 1.000000     |

Fuente: Tabla Python 3. Elaboración propia

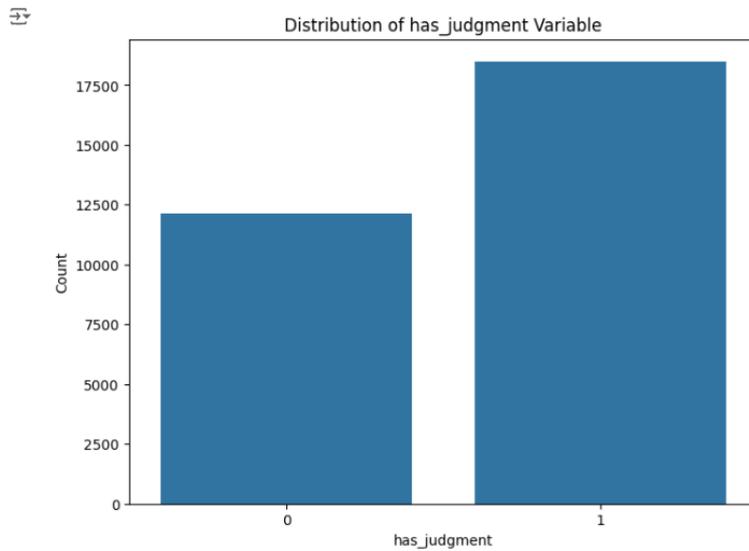
Estas estadísticas proporcionan una visión general de la distribución y la variabilidad de los datos en cada variable. Por ejemplo, se puede observar que la mayoría de las variables binarias tienen una baja proporción de observaciones positivas (1).

### 1.3.5. Estructura de los datos

En esta sección, se analizará la estructura de los datos utilizados en el presente análisis. Más concretamente, se analizará la distribución de la variable “has\_judgment” y sus relaciones con las variables “count\_decisions” y “count\_on appeal”.

El gráfico de barras de la figura 4 muestra la distribución de la variable "has\_judgment". El gráfico tiene un eje de categorías en la parte inferior que muestra los valores posibles de la variable, que son 0 y 1. El eje de valores en el lado izquierdo muestra el número de observaciones para cada valor de la variable, como se observa hay más observaciones (más de 17500) que tienen juicio, lo que sugiere que los juicios son relativamente comunes en el conjunto de datos.

Figura 4. Distribución de la variable “has\_judgement”

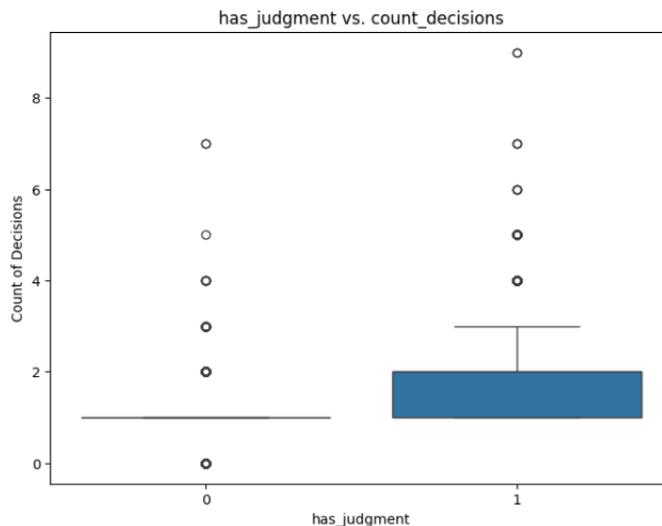


Fuente: Gráfico Python 3. Elaboración propia

A continuación, el diagrama de caja de la figura 5 muestra la distribución de la variable "count\_decisions" para cada valor de la variable "has\_judgment". El diagrama tiene un eje de categorías en la parte inferior que muestra los valores posibles de la variable "has\_judgment", que son 0 y 1. El eje de valores en el lado izquierdo muestra el número de decisiones ("count\_decisions").

Para las observaciones con un valor de 0 para la variable "has\_judgment", la mediana de la variable "count\_decisions" es menor que para las observaciones con un valor de 1 para la misma variable. Esto sugiere que existe una relación positiva entre las variables. Es decir, las observaciones que tienen un juicio tienden a tener un mayor número de decisiones.

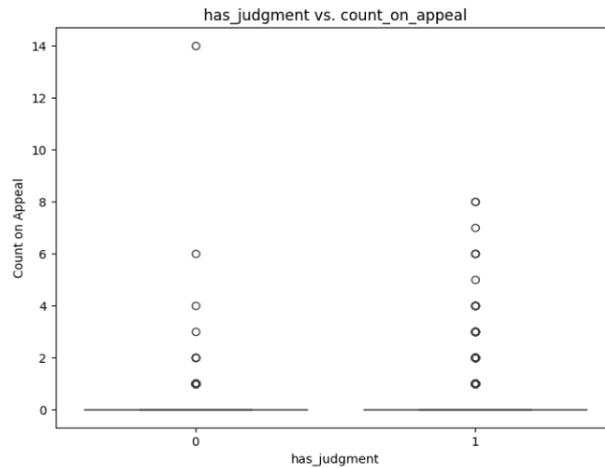
Figura 5. Relación entre las variables has\_judgment y count\_decisions



Fuente: Gráfico Python 3. Elaboración propia

En la figura 6 se analiza la relación entre la variable target y “count\_on\_appeal”. El diagrama de caja muestra que existe una relación positiva entre las variables. Lógicamente, las observaciones que tienen un juicio tienden a tener un mayor número de apelaciones.

Figura 6. Relación entre las variables has\_judgment y count\_on\_appeal



Fuente: Gráfico Python 3. Elaboración propia

## 1.4. Procesamiento de datos

En esta sección se describe el proceso de procesamiento de los datos antes de su utilización en el análisis y modelado. El procesamiento de datos en Python es fundamental para preparar los datos antes de construir cualquier modelo para conseguir resultados más precisos y confiables.

### 1.4.1. Eliminación de Columnas Irrelevantes y Valores Faltantes

Se han eliminado las columnas que se consideraron irrelevantes para el análisis o el modelado. Estas columnas son las variables del apartado 1.3.3. que incluyen identificadores únicos, nombres de casos, información de transferencia y unión, listas de apelaciones y procedimientos, así como variables relacionadas con opiniones y roles de las partes involucradas.

En caso de existir valores faltantes en el conjunto de datos, se han asignado valores predeterminados para garantizar la integridad del análisis. En este caso, se asignó el valor cero (0) a los valores faltantes en el conjunto de datos.

```
[ ] # Preprocesamiento
# Eliminar columnas irrelevantes
df.drop(['key_id', 'court_id', 'court', 'iuropa_case_id', 'cjeu_case_id', 'case_year', 'case_number',
        'case_name', 'transferred_to', 'joined_to', 'list_on_appeal', 'list_appeal_of', 'list_proceedings',
        'list_decisions', 'has_opinion', 'party_role', 'iuropa_party_id', 'party_type'], axis=1, inplace=True)
```

## 1.4.2. División de Datos

Los datos se dividieron en dos conjuntos principales: características (X) que incluirán las variables a estudiar del apartado 1.3.1. y la variable objetivo “has\_judgement” (y). Además, estos conjuntos se dividieron luego en conjuntos de entrenamiento y prueba para su uso en el modelado. Se ha utilizado la función `train_test_split` para realizar esta división, con una proporción de datos de prueba del 20% y una semilla aleatoria para garantizar la reproducibilidad de los resultados.

```
# Dividir los datos en conjuntos de entrenamiento y prueba
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
print(len(X_train), len(X_test), len(y_train), len(y_test))
```

24511 6128 24511 6128

Los resultados indican que la longitud del conjunto de entrenamiento para características (X\_train) es de 24511, la longitud del conjunto de prueba para características (X\_test) es de 6128, la longitud del conjunto de entrenamiento para la variable objetivo (y\_train) es de 24511 y la longitud del conjunto de prueba para la variable objetivo (y\_test): 6128

Esta división asegura que el modelo se entrene en un conjunto de datos separado del que se utiliza para evaluar su rendimiento, esto se hace para garantizar la generalización del modelo a nuevos datos.

## 2. DESARROLLO DEL MODELO PREDICTIVO

### 2.1. Diseño del modelo

En esta etapa de diseño, se construye el primer modelo utilizando regresión logística.

```
[14] # Utilizar regresión logística para construir el primer modelo
lr = LogisticRegression()
lr.fit(X_train, y_train)
lr_predict = lr.predict(X_test)
lr_predict_prob = lr.predict_proba(X_test)
print(lr_predict)
print(lr_predict_prob[:,1])
```

[1 1 1 ... 1 0 0]  
[9.79512534e-01 9.91123270e-01 5.35671601e-01 ... 9.91123270e-01  
4.46210880e-04 2.01271575e-03]

Los resultados del modelo de regresión logística indican las predicciones de las etiquetas predichas por el modelo (1 para presencia de juicio y 0 para ausencia de juicio), así como las probabilidades asociadas con cada predicción. La probabilidad indica la confianza del modelo en su predicción, donde valores más cercanos a 1 indican una mayor certeza de la predicción positiva.

### 2.1.1. Métricas de evaluación

Para evaluar el rendimiento de cada modelo, se han utilizado las siguientes métricas:

- Matriz de Confusión: Esta matriz permite visualizar el desempeño del algoritmo, mostrando los verdaderos positivos, falsos positivos, verdaderos negativos y falsos negativos.
- Precisión (Accuracy): Es la proporción de predicciones correctas entre el total de casos evaluados.

```
# Métricas
lr_conf_matrix = confusion_matrix(y_test, lr_predict)
lr_accuracy = accuracy_score(y_test, lr_predict)
print(lr_conf_matrix)
print(lr_accuracy)
```

```
[[1714  702]
 [   10 3702]]
0.8838120104438643
```

La matriz de confusión y la precisión obtenidas indican que el modelo de regresión logística tiene una precisión del 88.38%, con 1714 verdaderos negativos, 702 falsos positivos, 10 falsos negativos y 3702 verdaderos positivos.

### 2.1.2. Clasificadores

A continuación, se han evaluado diversos clasificadores: Naive Bayes, Árboles de Decisión y Bosques Aleatorios.

En el primer modelo de **Naive Bayes** se observa una precisión del 85.39%.

La matriz de confusión muestra 1527 verdaderos negativos, 889 falsos positivos, 6 falsos negativos y 3706 verdaderos positivos. Aunque es un modelo sencillo, presenta una precisión menor en comparación con la regresión logística.

```
# Con Naive Bayes
gnb = GaussianNB()
gnb.fit(X_train, y_train)
gnb_predict = gnb.predict(X_test)
gnb_predict_prob = gnb.predict_proba(X_test)
print(gnb_predict)
print(gnb_predict_prob)
gnb_conf_matrix = confusion_matrix(y_test, gnb_predict)
gnb_accuracy_score = accuracy_score(y_test, gnb_predict)
print(gnb_conf_matrix)
print(gnb_accuracy_score)
```

```
[1 1 1 ... 1 0 0]
[[1.23302412e-005 9.99987670e-001]
 [1.18933278e-005 9.99988107e-001]
 [3.24250941e-004 9.99675749e-001]
 ...
 [1.18933278e-005 9.99988107e-001]
 [1.00000000e+000 0.00000000e+000]
 [1.00000000e+000 4.60811099e-104]]
[[1527  889]
 [    6 3706]]
0.8539490861618799
```

El modelo de **Árboles de Decisión** muestra una precisión del 91.02%.

La matriz de confusión indica 1969 verdaderos negativos, 447 falsos positivos, 103 falsos negativos y 3609 verdaderos positivos. Este modelo mejora significativamente la precisión en comparación con Naive Bayes y la regresión logística.

```
# Con Árboles de decisión para clasificar
# Establecemos el número de árboles en 10
dt = DecisionTreeClassifier(max_depth=10)
dt.fit(X_train,y_train)
dt_predict = dt.predict(X_test)
dt_predict_proba = dt.predict_proba(X_test)
dt_conf_matrix = confusion_matrix(y_test, dt_predict)
dt_accuracy_score = accuracy_score(y_test, dt_predict)
print(dt_conf_matrix)
print(dt_accuracy_score)
```

```
[[1969  447]
 [ 103 3609]]
0.9102480417754569
```

El clasificador de **Bosques Aleatorios o “Random Forest”** ha conseguido la mejor precisión, con un 91.14%.

La matriz de confusión muestra 1969 verdaderos negativos, 447 falsos positivos, 96 falsos negativos y 3616 verdaderos positivos. Este modelo es el más efectivo entre los evaluados, con una precisión ligeramente superior a la del Árbol de Decisión.

```
# Con Random Forest
rf = RandomForestClassifier(max_depth=10) #10 trees
rf.fit(X_train, y_train)
rf_predict = rf.predict(X_test)
rf_predict_proba = rf.predict_proba(X_test)
rf_conf_matrix = confusion_matrix(y_test,rf_predict)
rf_accuracy_score = accuracy_score(y_test, rf_predict)
print(rf_conf_matrix)
print(rf_accuracy_score)
```

```
[[1969  447]
 [   96 3616]]
0.9113903394255874
```

## 2.2. Entrenamiento del modelo

Para el siguiente paso en la creación del modelo se va a entrenar y ajustar dos pipelines de aprendizaje automático, uno utilizando “Random Forest” y otro utilizando “Gradient Boosting”. Se han optimizado los hiperparámetros de cada modelo utilizando validación cruzada para seleccionar el mejor conjunto de parámetros y maximizar la precisión.

Primero se definen dos pipelines de aprendizaje automático: Random Forest (rf) y Gradient Boosting (gb). Para cada pipeline, se utilizaron diferentes hiperparámetros para optimizar el rendimiento como se muestra en la figura 6.

En Random Forest los hiperparámetros son:

- `n_estimators`: Número de árboles en el bosque (100, 200).
- `max_features`: Número máximo de características consideradas para dividir un nodo ("sqrt", "log2", 0.33).

En Gradient Boosting son:

- `n_estimators`: Número de árboles en el conjunto (100, 200).
- `learning_rate`: Tasa de aprendizaje (0.05, 0.1, 0.2).
- `max_depth`: Profundidad máxima de los árboles (1, 3, 5).

Figura 7. Entrenamiento y ajuste de los pipelines rf y gb

```
# Entrenar y ajustar dos pipelines de aprendizaje automático, uno para Random Forest (rf) y otro para Gradient Boosting (gb)
pipelines = {
    "rf": make_pipeline(StandardScaler(), RandomForestClassifier(random_state=123)),
    "gb": make_pipeline(StandardScaler(), GradientBoostingClassifier(random_state=123))
}

# Para Random Forest (rf), los hiperparámetros incluyen el número de estimadores (n_estimators)
# y el número máximo de características a considerar para el particionado (max_features)

rf_hyperparameters = {
    "randomforestclassifier__n_estimators": [100, 200],
    "randomforestclassifier__max_features": ["sqrt", "log2", 0.33] # Valid options for max_features
}

# Para Gradient Boosting (gb), los hiperparámetros incluyen el número de estimadores (n_estimators),
# la tasa de aprendizaje (learning_rate) y la profundidad máxima de los estimadores de regresión individuales (max_depth)

gb_hyperparameters = {
    "gradientboostingclassifier__n_estimators": [100, 200],
    'gradientboostingclassifier__learning_rate': [0.05, 0.1, 0.2],
    'gradientboostingclassifier__max_depth': [1, 3, 5]
}

hyperparameters = {"rf": rf_hyperparameters, "gb": gb_hyperparameters}
```

Fuente: Código Python 3. Elaboración propia

### 2.3. Validación del modelo

El código anterior define los pipelines y los hiperparámetros. Ahora, hay que crear objetos de validación cruzada (GridSearchCV) para cada pipeline y ajustar los modelos a los datos de entrenamiento (X\_train, y\_train). Después del ajuste, los modelos se van a guardar en el diccionario fitted\_models como muestra la figura 8. Después, se van a imprimir las mejores puntuaciones obtenidas durante la validación cruzada.

Figura 8. Validación cruzada y ajuste

```
# Iterar a través de los pipelines de modelos, ajustando cada uno y guardándolo en fitted_models
for name, pipeline in pipelines.items():

    # Crear un objeto de validación cruzada a partir del pipeline y los hiperparámetros
    model = GridSearchCV(pipeline, hyperparameters[name], cv=10, n_jobs=-1)

    # Ajustar el modelo en X_train, y_train
    model.fit(X_train, y_train)

    # Guardar el modelo en fitted_models[name]
    fitted_models[name] = model

    # Imprimir '{name} ha sido ajustado'
    print(name, 'ha sido ajustado.')
```

rf ha sido ajustado.  
gb ha sido ajustado.

Fuente: Código Python 3. Elaboración propia

Para determinar los valores específicos de los hiperparámetros que produjeron las mejores puntuaciones durante la validación cruzada para ambos modelos (Random Forest y Gradient Boosting), debemos inspeccionar los resultados del GridSearchCV para cada

modelo. Esto incluye revisar los atributos `best_params_` y `best_score_` del objeto `GridSearchCV` después de haber ajustado el modelo.

En Random Forest (rf):

- Mejor puntuación: 0.9079
- Mejores hiperparámetros:
  - `randomforestclassifier__n_estimators`: 200
  - `randomforestclassifier__max_features`: 'sqrt'

En Gradient Boosting (gb):

- Mejor puntuación: 0.9080
- Mejores hiperparámetros:
  - `gradientboostingclassifier__n_estimators`: 200
  - `gradientboostingclassifier__learning_rate`: 0.1
  - `gradientboostingclassifier__max_depth`: 3

Las mejores puntuaciones muestran que ambos modelos, Random Forest y Gradient Boosting, obtuvieron una precisión muy similar durante la validación cruzada, con una ligera ventaja para Gradient Boosting (0.9080 frente a 0.9079).

```
# Imprimir los mejores resultados
for name, model in fitted_models.items():
    print(name, model.best_score_)

rf 0.9079185477446033
gb 0.908040963404756
```

### 3. ANÁLISIS DE RESULTADOS

#### 3.1. Interpretación de resultados

En esta sección se presenta la interpretación de los resultados obtenidos de los modelos Random Forest y Gradient Boosting que se han creado en la sección anterior. Se van a evaluar ambos modelos en términos de precisión y se va a analizar la matriz de confusión.

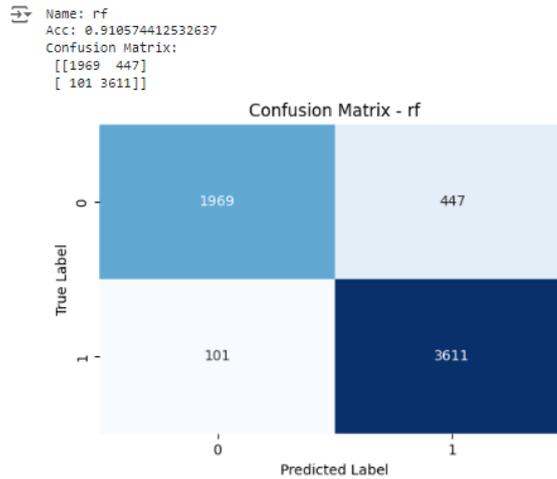
##### 3.1.1. Resultados del Modelo Random Forest

El modelo Random Forest obtuvo una precisión de 0.9106 en el conjunto de prueba. Esto indica que aproximadamente el 91.06% de las predicciones realizadas por el modelo fueron correctas.

La matriz de confusión del modelo Random Forest (figura 9) muestra 3611 verdaderos positivos, 1969 verdaderos negativos, 447 falsos positivos y 101 falsos negativos. En conclusión, el modelo Random Forest tiene un alto número de verdaderos positivos y

negativos, lo que contribuye a su alta precisión. Sin embargo, también presenta una cantidad notable de falsos positivos y bastantes falsos negativos.

Figura 9. Matriz de confusión de rf

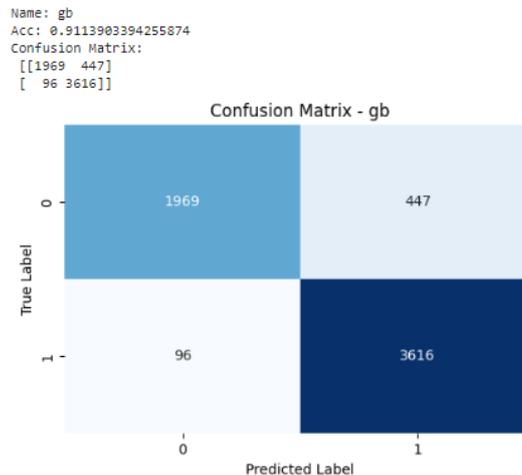


Fuente: *Gráfico Python 3. Elaboración propia*

### 3.1.2. Resultados del Modelo Gradient Boosting

El modelo Gradient Boosting alcanzó una precisión ligeramente superior de 0.9114 en el conjunto de prueba, un 91.14% de las predicciones fueron correctas. La matriz de confusión del modelo Gradient Boosting (figura 10) en este caso muestra 3616 verdaderos positivos, 1969 verdaderos negativos, 447 falsos positivos y 96 falsos negativos. Estos resultados indican que el modelo Gradient Boosting también tiene un alto número de verdaderos positivos y negativos, con una ligera mejora en la reducción de falsos negativos en comparación con el modelo Random Forest.

Figura 10. Matriz de confusión de gb



Fuente: *Gráfico Python 3. Elaboración propia*

### 3.1.3. Comparación de Modelos

Ambos modelos, Random Forest y Gradient Boosting, muestran un rendimiento similar en términos de precisión y estructura de la matriz de confusión. Sin embargo, el modelo Gradient Boosting tiene una ligera ventaja en precisión y en la reducción de falsos negativos. Esta diferencia puede ser crítica en aplicaciones donde los falsos negativos tienen una alta importancia elevada, por ejemplo, en el caso de que estemos intentando predecir si una sentencia tiene ya un juicio específico y obtenemos una respuesta negativa cuando en verdad sí lo tiene, lo que llevaría a un posible abogado a concluir que el caso no tiene precedentes cuando en realidad sí.

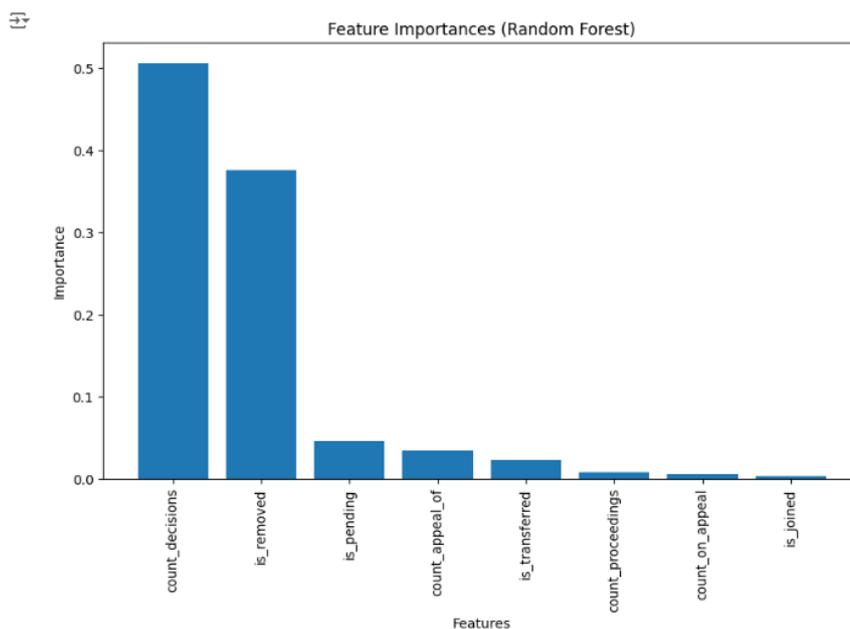
En resumen, aunque ambos modelos son altamente precisos, la elección más lógica sería optar por Gradient Boosting al tener una ligera ventaja en precisión y manejo de falsos negativos.

### 3.2. Relevancia de las variables

En esta sección, se presenta el análisis de la relevancia de las variables utilizando el modelo Random Forest y Gradient Boosting. El objetivo es identificar qué variables tienen mayor impacto en las predicciones de los modelos.

En Random Forest el gráfico resultante (figura 11) muestra las variables ordenadas según su relevancia para el modelo. Las variables con mayor importancia son “count\_decisions” y “is\_removed” y, por ende, tienen mayor impacto en las predicciones del modelo, siendo las que menos relevancia tienen “count\_proceedings”, “count\_on\_appeal” y “is\_joined”.

Figura 11. Relevancia de las variables en rf

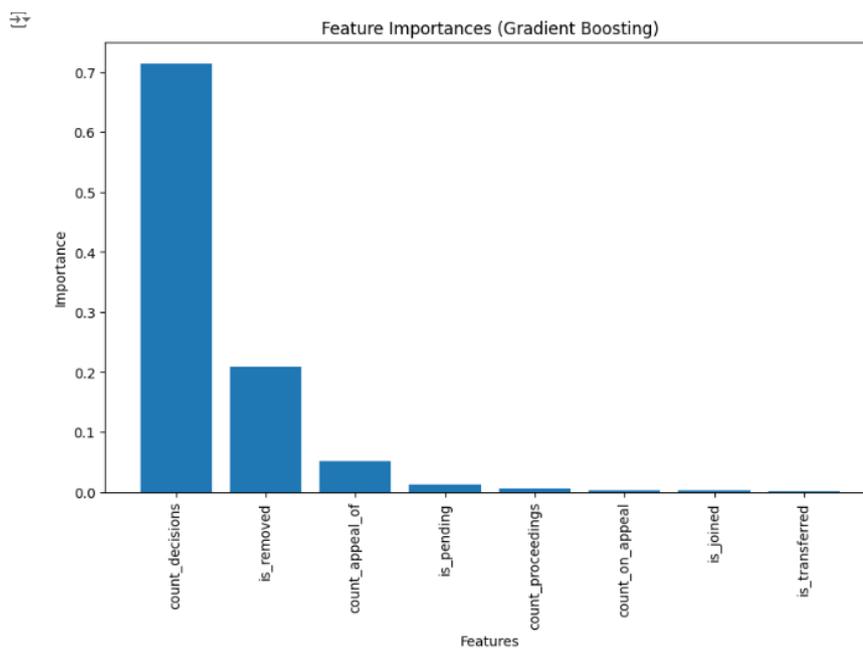


Fuente: *Gráfico Python 3. Elaboración propia*

En Gradient Boosting el gráfico (figura 12) las variables con mayor importancia son las mismas que en Random Forest: “count\_decisions” y “is\_removed”. Sin embargo, en este modelo la variable “count\_decisions” tiene más importancia (0.71) y el resto de las variables se reparten el 30% de relevancia restante del modelo.

Se observa que en ambos modelos la variable “count\_decisions” es la más relevante incluso del 70% en el modelo de gb lo que significa que esta característica tiene una fuerte influencia en la predicción de si un caso tendrá una sentencia o no.

Figura 12. Relevancia de las variables en gb



Fuente: Gráfico Python 3. Elaboración propia

Esta variable entera representa el número total de resoluciones emitidas sobre un caso, mide la complejidad y la frecuencia de intervenciones judiciales, lo que es crucial para los modelos predictivos al identificar patrones y prever la emisión de una sentencia. Además, Los casos con un mayor número de decisiones tienden a tener una sentencia final debido al extenso tratamiento judicial que han recibido. Un mayor número de decisiones refleja la complejidad y el nivel de atención judicial de un caso, proporcionando una rica fuente de información que reduce la incertidumbre y mejora la precisión de las predicciones.

Por otro lado, la variable “is\_removed” es una variable dummy que toma el valor de 1 si el procedimiento principal del caso ha sido eliminado del Registro y 0 en caso contrario. La eliminación del Registro puede ocurrir por varias razones, como que las partes involucradas decidan retirar el caso antes de que se llegue a una sentencia, la resolución extrajudicial, lo que haría innecesaria una sentencia judicial e incluso por errores administrativos o duplicaciones en los registros.

La alta relevancia de “is\_removed” en el modelo puede explicarse por varias razones:

- **Indicador directo de no sentencia:** Si un caso ha sido eliminado del Registro, es altamente probable que no haya llegado a una sentencia, ya que la eliminación sugiere que el caso no ha seguido el curso judicial habitual.
- **Simplificación del proceso:** La eliminación de un caso del Registro puede indicar que el proceso judicial se ha detenido, lo que reduce significativamente la probabilidad de que se emita una sentencia.
- **Filtración de casos irrelevantes:** Casos eliminados pueden considerarse menos complejos o no suficientemente relevantes para ser tratados por el Tribunal, lo que influye en su exclusión de las estadísticas de sentencias.

En resumen, la variable “is\_removed” es crucial porque actúa como un fuerte predictor negativo de la emisión de sentencias. Su alta relevancia sugiere que cualquier caso que haya sido eliminado del Registro tiene una probabilidad muy baja de haber llegado a una sentencia, lo cual es información valiosa para el modelo predictivo.

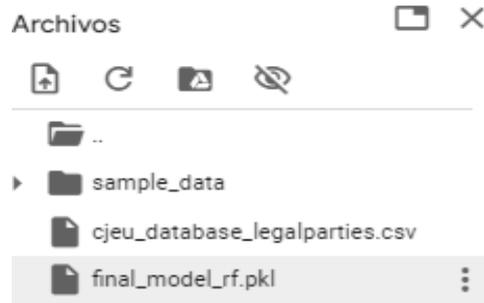
Finalmente, cabe destacar que, aunque el uso de técnicas de ensamblado de modelos como Random Forest o Gradient Boosting distribuyen la importancia entre varias variables, en este caso el modelo de Gradiente Boosting teniendo una precisión ligeramente mayor puede no ser el mejor, ya que no es ideal que un modelo tenga una variable que acapare la mayoría de la relevancia (un 70%) por varias razones. En primer lugar, el modelo puede ajustarse demasiado a los datos de entrenamiento y no generalizar bien a nuevos datos, lo que se conoce como sobreajuste. Además, un modelo con una variable dominante puede ser frágil; si esta variable cambia o tiene errores, el rendimiento del modelo puede degradarse significativamente. También, la dominancia de una sola variable puede ocultar la importancia de otras variables valiosas, resultando en una pérdida de información crucial para el modelo.

#### 4. GENERACIÓN DEL MODELO

Para concluir la generación del modelo en este trabajo, se ha procedido a exportar el modelo entrenado utilizando la biblioteca “pickle”. Pickle es un módulo en Python diseñado para serializar y deserializar objetos de Python, la serialización implica convertir un objeto en memoria en un formato que puede almacenarse o transmitirse, mientras que la deserialización es el proceso inverso de reconstruir el objeto a partir de su formato serializado. Es útil para guardar estructuras de datos complejas de Python, como modelos entrenados (como es este caso) o diccionarios, en archivos binarios para su almacenamiento y posterior uso. ( Python Software Foundation, 2004)

Este paso asegura que el modelo Random Forest optimizado esté disponible en archivos (figura 13) para su uso futuro en aplicaciones prácticas. El modelo exportado encapsula el conocimiento adquirido durante el proceso de entrenamiento, incluyendo la importancia de las variables y los parámetros ajustados. Finalmente, este modelo final se podría desplegar en un servidor.

Figura 13. Disponibilidad del modelo en archivos



Fuente: *Python 3. Elaboración propia*

## CAPÍTULO IV. DESAFÍOS ÉTICOS Y LEGALES

Los modelos predictivos evocan una amplia variedad de preocupaciones éticas, especialmente en temas sensibles como lo sería la justicia. Esta sección aborda varias de ellas, como preguntas sobre la privacidad y la protección de datos, la selección de datos y el sesgo de la máquina, la transparencia y la falta de regulación dentro del marco legal europeo. Esta lista no es completa, pero proporciona una visión general de algunos aspectos destacados que deben abordarse al diseñar herramientas judiciales predictivas.

### 1. PRIVACIDAD Y PROTECCIÓN DE DATOS

Los sistemas de inteligencia artificial a menudo dependen de conjuntos de datos extensos, que pueden incluir información altamente confidencial y sensible, como datos personales y conversacionales. En casos aislados, esto puede no representar una gran amenaza, pero los incidentes de vigilancia masiva y recopilación de datos personales sin consentimiento son problemas que no tardan en convertirse en algo más grande. El ejemplo más destacado es la constante vigilancia de los ciudadanos en China (Qian, Xiao, Mozur y Cardia, 2022).

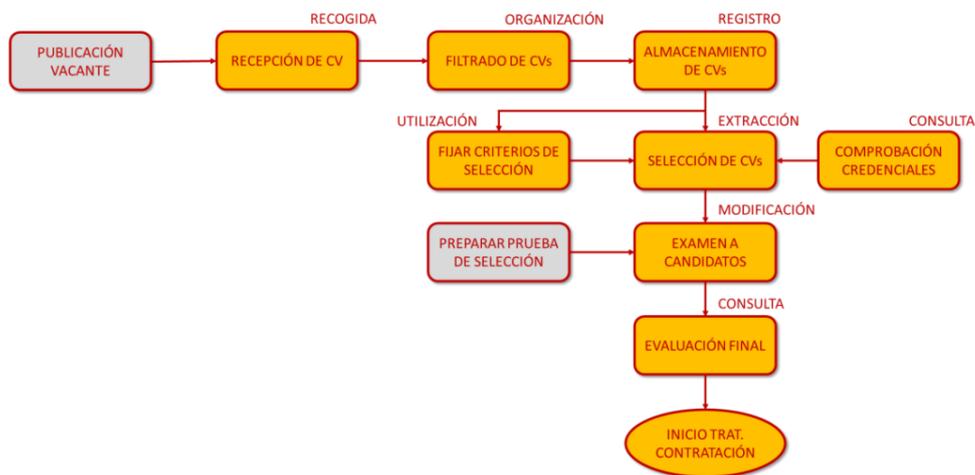
La privacidad de los datos no debe considerarse como un fin en sí mismo, sino que es un derecho habilitador estrechamente vinculado a otros derechos, como la libertad de expresión o el derecho a la información. La voluntad y el consentimiento se vuelven entonces un factor fundamental a la hora de dar los datos. La definición de consentimiento del interesado la encontramos en el art. 4 del Reglamento de Protección de Datos (RGPD), donde se indica que es “toda manifestación de voluntad libre, específica, informada e inequívoca por la que el interesado acepta, ya sea mediante una declaración o una clara acción afirmativa, el tratamiento de datos personales que le conciernen”. De acuerdo con el artículo 7 del mismo Reglamento, se establece que el consentimiento debe ser expresado de manera clara. Obtener el consentimiento del individuo afectado será

necesario para incluir sus datos personales en un archivo. La responsabilidad de determinar qué datos pueden ser registrados y procesados recae en el titular de los datos, quien debe recibir información de “modo expreso, preciso e inequívoco” antes de consentir su tratamiento (Sanmateo, 2017/2018).

Para implementar adecuadamente las medidas de Protección de Datos Personales, el encargado debe examinar detalladamente el procesamiento que tiene la intención de realizar. Este análisis debe superar la simple consideración del tratamiento como una caja negra, siendo necesario identificar las operaciones específicas que se llevan a cabo y la relación entre ellas dentro del proceso. Se deben identificar las operaciones relevantes que pueden formar parte del tratamiento y que son de interés para la protección de datos, están definidas, de forma no exhaustiva, en el artículo 4.2 del RGPD como: “recogida, registro, organización, estructuración, conservación, adaptación o modificación, extracción, consulta, utilización, comunicación por transmisión, difusión o cualquier otra forma de habilitación de acceso, cotejo o interconexión, limitación, supresión o destrucción”, un ejemplo de este proceso podría ser el tratamiento de datos en relación con los procesos de selección de personal, ilustrado en la figura 14 (AEPD, 2023).

Como conclusión, la privacidad y la protección de datos en los modelos predictivos de sentencias judiciales son un desafío. La recopilación y el uso de grandes conjuntos de datos que incluyen información personal sensible, como historiales judiciales, información financiera y detalles de casos específicos, generan serios riesgos para la privacidad de las personas involucradas.

Figura 14. Ejemplo de tratamiento de datos relativo a la selección de personal



Fuente: AEPD, 2023

## 2. EL SESGO EN LOS ALGORITMOS

Los modelos predictivos judiciales son una herramienta para prever el resultado de una sentencia basada en métodos estadísticos y mecanismos de interpretación que evolucionan gradualmente. Por lo tanto, la selección de datos, que es la base de la predicción, es esencial. Si no hay suficientes datos, datos irrelevantes, datos inexactos, datos desactualizados o datos de baja calidad, es probable que las predicciones finales reflejen esto. Además, los datos históricos de delitos desempeñan un papel importante en el "entrenamiento" de muchas herramientas de predicción judicial, lo que plantea la pregunta de si las decisiones automatizadas resultantes solo refuerzan posibles sesgos y discriminaciones inherentes.

Aproximadamente el 64% de los desarrolladores de software son hombres y el 62% son blancos y proceden de Estados Unidos (Yilmaz H. , 2022), por lo que los datos que recopilan serán principalmente de hombres blancos estadounidenses, lo que puede reflejar sus creencias e ideas. Podemos pensar que esta es la razón principal de los sesgos en la IA. De la misma manera, estos sistemas pueden producir resultados sesgados de manera inadvertida cuando los datos históricos en los que se basan están sesgados en su origen, especialmente en los datos utilizados para entrenar el algoritmo.

En el ámbito legal, este sesgo puede conducir a resultados injustos y perpetuar la discriminación, especialmente en aplicaciones como la predicción de reincidencia. Por ejemplo, un algoritmo puede proporcionar un puntaje de riesgo para la probabilidad de reincidencia de un criminal, pero si se basa en datos de una región con discriminación racial, puede reforzar sesgos sistémicos. Por ejemplo, el programa COMPAS (Correctional Offender Management Profiling for Alternative Sanctions), desarrollado por Northpointe, es una herramienta de evaluación de riesgos utilizada en los Estados Unidos para predecir la probabilidad de reincidencia de los delincuentes. Se ha utilizado en casos como Estado de Wisconsin vs. Loomis (2016) o Boudin vs. Cuomo (2019). Este programa tiene una mayor eficiencia en el sistema de justicia penal norteamericano y se puede considerar una herramienta objetiva y neutral. Sin embargo, COMPAS tiene un mayor sesgo en contra de los acusados negros y latinos, prediciendo una mayor probabilidad de reincidencia para estos grupos, esto ha generado acusaciones de discriminación racial y violaciones de la igualdad ante la ley. Además, los algoritmos de COMPAS son complejos y opacos, lo que dificulta comprender cómo se generan las predicciones de riesgo. Esta falta de transparencia dificulta la evaluación de la precisión y confiabilidad del programa, y limita la capacidad de los acusados para impugnar las decisiones basadas en sus resultados (Blánquez, 2020).

Por lo tanto, una cultura sólida de gestión de datos es esencial y está ganando cada vez más atención (Puente, 2019). Mientras marcos legislativos detallados pueden ayudar a establecerlo, como ampliaremos a continuación, la selección de datos adecuados es vital para evitar un modelo sesgado.

### 3. TRANSPARENCIA

Una de las cuestiones clave se refiere a la transparencia, existe el peligro de que las predicciones se conviertan en resultados de un proceso oculto en una caja negra. Como resultado, son difíciles de entender tanto para los ciudadanos como, quizás, para los propios jueces. Y para que las sentencias se ejecuten se necesita que los propios ciudadanos que están siendo sentenciados entiendan, en cierta medida, cómo se ha llegado a esa decisión, es decir, se busca mantener la seguridad jurídica. De hecho, algunas entidades reguladoras en Europa, tanto en los ámbitos de competencia y protección al consumidor como en supervisión financiera, han estado utilizando auditorías de algoritmos durante varios años como parte de sus mecanismos de monitoreo y verificación. La auditoría de algoritmos implica la revisión de documentación relacionada con los resultados de las pruebas realizadas a los algoritmos o la inspección de su funcionamiento interno. De esta manera, se busca evitar que los algoritmos se conviertan en estas cajas negras sin transparencia. Además, en Europa, la recién promulgada Ley de Servicios de los Mercados Digitales (DSA, por sus siglas en inglés) representa un esfuerzo significativo por parte de la Comisión Europea para salvaguardar los derechos de los consumidores en este tipo de mercados. La DSA prohíbe diversas prácticas que infringen los derechos a la privacidad y a la información de los usuarios, como el intercambio de datos personales entre diferentes servicios de la plataforma sin la autorización del usuario. Además, impone obligaciones de rendición de cuentas y transparencia a las plataformas digitales con el objetivo de prevenir comportamientos abusivos o engañosos (Méndez & Sánchez, 2023).

Los algoritmos de redes neuronales, base de muchos modelos predictivos, son intrínsecamente complejos y opacos. Su funcionamiento se basa en una intrincada red de conexiones entre neuronas artificiales que aprenden a partir de grandes cantidades de datos. Esta complejidad dificulta, si no imposibilita, la comprensión humana de los procesos internos que llevan a la generación de predicciones. La falta de transparencia en los modelos predictivos de sentencias judiciales genera diversas preocupaciones. Primero, la dificultad para evaluar la precisión y confiabilidad, al no comprender cómo se generan las predicciones, es difícil evaluar si estas son precisas y confiables. Esto podría conducir a errores judiciales y a la toma de decisiones injustas. Segundo, la limitación del derecho a un juicio justo ya que los acusados y las partes involucradas en un proceso judicial tienen derecho a comprender los fundamentos de las decisiones que los afectan. La falta de transparencia en los modelos predictivos limita este derecho y podría afectar la confianza en el sistema judicial. Tercero, crea un riesgo para la discriminación y el sesgo, dado que la falta de transparencia dificulta la detección y la corrección de posibles sesgos introducidos en la programación (Méndez & Sánchez, 2023, págs. 1-38). Como ya mencionábamos en el apartado anterior modelos opacos como redes neuronales ya se están utilizando en Estado Unidos, como por ejemplo con el programa COMPAS. Mientras que en Europa tendrán que ser compatibles con los principios de transparencia y rendición de cuentas establecidos en el nuevo Reglamento de la Unión Europea sobre Inteligencia Artificial (Reglamento IA).

#### 4. MARCO LEGAL Y REGULACIONES ACTUALES

Los problemas éticos mencionados anteriormente se pueden mapear de manera paralela en el ámbito legal. Estas preocupaciones sobre el aumento de la supervisión y vigilancia dirigidas por el gobierno pueden vincularse al debate sobre la privacidad en la era digital. Sin embargo, al igual que la privacidad de los datos no debe considerarse como un fin en sí mismo. Más bien, como ya se indicó anteriormente, es un derecho habilitador estrechamente vinculado a otros derechos, como la libertad de expresión o el derecho a la información (Cannataci, 2017).

En el marco de las Naciones Unidas solo se refiere a la privacidad en el Artículo 12. de la Declaración Universal de Derechos Humanos de 1948 “Nadie será objeto de injerencias arbitrarias en su vida privada, su familia, su domicilio o su correspondencia, ni de ataques a su honra o a su reputación. Toda persona tiene derecho a la protección de la ley contra tales injerencias o ataques”. Por otro lado, la Carta de Derechos Fundamentales de la Unión Europea (CDFUE) ofrece una forma única de proteger este derecho. No solo está cubierto en el Artículo 7 “Toda persona tiene derecho al respeto de su vida privada y familiar, de su domicilio y de sus comunicaciones.” (Respeto a la vida privada y familiar), sino también con una segunda disposición que aborda 'la protección de datos personales' en el Artículo 8.1. “Toda persona tiene derecho a la protección de los datos de carácter personal que le conciernan.”

Otros instrumentos legales internacionales, como el Convenio Europeo de Derechos Humanos de 1950 (CEDH) con su jurisprudencia rica en desarrollo sobre la privacidad, no contienen una disposición específica relacionada con datos personales digitales.

Sin embargo, la norma más relevante en la actualidad se trata del Reglamento (UE) 2021/2100 del Parlamento Europeo y del Consejo de 19 de octubre de 2021 sobre la inteligencia artificial y por la que se modifican determinados actos legislativos de la Unión y se derogan el Reglamento (UE) 2016/679 y la Directiva (UE) 2019/2161, conocido como Reglamento IA.

Este nuevo Reglamento tiene varios objetivos entre los que destacan (Parlamento Europeo, 2021):

- **Promover la transparencia y la explicabilidad de los sistemas de IA:** El Reglamento exige a las empresas y organizaciones que proporcionen información clara sobre cómo funcionan sus sistemas de IA, incluyendo los datos utilizados, los algoritmos subyacentes y los procesos de toma de decisiones.
- **Establecer evaluaciones de riesgos y requisitos de registro:** Ciertos sistemas de IA considerados de alto riesgo, como aquellos utilizados en el ámbito judicial, deben someterse a evaluaciones de conformidad y registro.
- **Proteger los derechos fundamentales:** El Reglamento IA incorpora principios éticos y legales para garantizar que los sistemas de IA respeten los derechos fundamentales, como la no discriminación, la privacidad y la protección de datos.

Entonces, según lo analizado en el apartado anterior ¿Pueden utilizarse modelos predictivos opacos en el ámbito judicial bajo el Reglamento IA? La respuesta no es sencilla. Si bien el Reglamento IA no prohíbe explícitamente el uso de modelos predictivos opacos en el ámbito judicial, sí establece requisitos estrictos de transparencia y explicabilidad para este tipo de sistemas. En particular, el Artículo 13 del Reglamento exige que los sistemas de IA de alto riesgo sean “comprensibles” y que las personas puedan “comprender los motivos de las decisiones tomadas por esos sistemas”.

En este contexto, la opacidad inherente a algunos modelos predictivos, especialmente aquellos basados en redes neuronales complejas, podría dificultar el cumplimiento de estos requisitos. Sin embargo, es importante destacar que el Reglamento IA no exige una transparencia absoluta, sino que busca un equilibrio entre la complejidad de los algoritmos y la necesidad de que las personas puedan comprender los fundamentos de las decisiones tomadas por estos sistemas.

## **CAPÍTULO V. CONCLUSIONES**

La investigación desarrollada en este estudio tenía como objetivo principal desarrollar y evaluar un modelo predictivo capaz de determinar la probabilidad de que un caso presentado ante el Tribunal de Justicia de la Unión Europea resulte en una sentencia, utilizando técnicas de Inteligencia Artificial, Big Data y ML. El análisis del paisaje actual de la IA en el ámbito judicial, a nivel global y más concretamente en la Unión Europea, reveló que estas tecnologías están siendo cada vez más adoptadas y muestran resultados prometedores en diversas aplicaciones legales.

A pesar de los desafíos éticos y legales que surgen de la aplicación de estas nuevas tecnologías, las tendencias futuras apuntan a un aumento de la integración de IA y Big Data y a una mejora de las tecnologías ya existentes en el ámbito jurídico.

El desarrollo del modelo predictivo se ha centrado en la creación de pipelines de aprendizaje automático para dos algoritmos: Random Forest y Gradient Boosting. Mediante el uso de técnicas validación cruzada y ajustes, se lograron modelos con altas precisiones, destacando el modelo de Gradient Boosting con una ligera ventaja en rendimiento (precisión de 0.9114). Este resultado respalda la hipótesis de que es posible anticipar con alta precisión la probabilidad de que un caso resulte en una sentencia, utilizando datos públicos del TJUE y técnicas de ML.

Sin embargo, la investigación presenta ciertos límites. La principal limitación radica en la naturaleza y calidad de los datos disponibles. Si bien los datos del TJUE son exhaustivos, la falta de acceso a datos privados y más detallados podría haber limitado la precisión y generalización del modelo. Además, la complejidad y diversidad de los casos judiciales ha tenido que ser muy reducida para poder estudiarlos sin necesidad de análisis altamente complejos que no se han estudiado en la carrera. Asimismo, el enfoque en solo dos algoritmos de ML significa que otras técnicas potencialmente útiles no han sido exploradas. Por otro lado, ambos modelos muestran una relevancia muy alta de una única

variable lo que puede no ser ideal. Finalmente, el conocimiento que se ha empleado en este trabajo dista de ser un conocimiento experto en la creación de modelos predictivos de cualquier tipo y cuenta con recursos muy limitados tanto en procesamiento de datos (límites RAM del propio programa empleado) como en acceso a recursos.

En cuanto a futuras líneas de investigación, sería beneficioso ampliar el conjunto de datos incluyendo fuentes de datos privados y más variados para mejorar la robustez del modelo, como datos de otros tribunales o incluso experiencias privadas de despachos de abogados. Además, explorar otros algoritmos y técnicas de ML, como redes neuronales profundas y modelos de aprendizaje por refuerzo, podría ofrecer mejoras adicionales en la precisión y utilidad del modelo predictivo.

Dicho esto, este estudio ha demostrado el potencial significativo de la IA y el Big Data en la predicción de resultados judiciales. Si en vez de una estudiante hablamos de equipos profesionales de programadores y abogados expertos con recursos sustancialmente más amplios de los que se han tenido para la realización de este trabajo, se podrían crear modelos mucho más sofisticados, con mayor precisión, basándose en datos complejos públicos y privados.

De esta manera, la implementación de modelos predictivos en sentencias judiciales y en el ámbito jurídico ya es una realidad y nos enfrenta a un debate crucial: ¿Estamos dispuestos a aceptar estas innovaciones tecnológicas y los cambios profundos que implican en la esfera legal? Este es el momento de decidir si abrazamos esta nueva realidad o si nos resistimos a su integración en nuestro sistema de justicia.

## 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, Silvia Lázaro Trasobares, estudiante de E3-Analytics de la Universidad Pontificia Comillas al presentar mi Trabajo Fin de Grado titulado " Modelo predictivo de resultados judiciales. El caso de la IA y Big Data en sentencias del Tribunal de Justicia Europeo.", 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:

1. **Corrector de estilo literario y de lenguaje:** Para mejorar la calidad lingüística y estilística del texto.
2. **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: 19/06/2024

Firma:

A handwritten signature in black ink, appearing to read 'Silvia Lázaro', written over a horizontal line.

## BIBLIOGRAFÍA

### Libros y Revistas:

- Aletras, N., Tsarapatsanis, D., Preoțiuc-Pietro, D., & Lampos, V. (2016). *Predicting judicial decisions of the European Court of Human Rights: a Natural Language Processing perspective*. PeerJ Computer Science.
- Blánquez, R. B. (2020). *El sesgo de la máquina en la toma de decisiones en el proceso penal*. Sevilla, 23-35. Editorial Universidad de Sevilla.
- Cannataci, J. (2017). *Games people play unvarnished insights about privacy at the global level*. Data Protection and Privacy under Pressure – Transatlantic tensions, EU.
- Cervellini, M., & Carbone, M. (2019). *Utilización de un modelo predictivo como soporte para la revisión de los recursos ordinarios de apelación*. Fiscalía de Estado de la Provincia de Buenos Aires, 33-34. La plata. <https://49jaiio.sadio.org.ar/pdfs/sid/SID-03.pdf>
- Chaisse, J., & Kirkwood, J. (2022). *Smart courts, smart contracts, and the future of online dispute resolution*. [https://papers.ssrn.com/sol3/papers.cfm?abstract\\_id=4344466](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4344466)
- Choo, K., Espejo, D., & Jayasinghe, D. (2020). *SULS Publications*. Sydney University Law Society: <https://www.suls.org.au/citations-blog/2020/9/25/predictive-judicial-analytics-implications-for-rule-of-law-and-the-legal-profession>
- Chouldechova, A., & Lum, K. (2020). *The present and future of ai in pre-trial risk assessment instruments*. Safety+Justice. [https://www.ncsc.org/\\_\\_data/assets/pdf\\_file/0019/52516/AI-in-Pre-Trial-Risk-Assessment-Brief-June-2020-R2.pdf](https://www.ncsc.org/__data/assets/pdf_file/0019/52516/AI-in-Pre-Trial-Risk-Assessment-Brief-June-2020-R2.pdf)
- E. Stern, R., L. Liebman, B., Z.Wang, A., & Roberts, M. (2021). *Automating Fairness? Artificial Intelligence in the Chinese Court*. Columbia Law School. [https://scholarship.law.columbia.edu/faculty\\_scholarship/2940/](https://scholarship.law.columbia.edu/faculty_scholarship/2940/)
- Fernández-Vicente, A. (2020). *Hacia una teoría crítica de la razón algorítmica*. Palabara Clave, 23 (2). <https://palabraclave.unisabana.edu.co/index.php/palabraclave/article/view/10839>
- Gawdat, M. (2021). *Scary Smart*. Dublin, 10-45. Bluebird
- Katz, D. M. (2023). *GPT-4 Passes the Bar Exam*. [https://papers.ssrn.com/sol3/papers.cfm?abstract\\_id=4389233](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4389233)
- Lawlor, R. C. (1963). *What Computers Can Do: Analysis and Prediction of Judicial Decisions*. American Bar Association Journal, 49, 337 - 344. <https://www.jstor.org/stable/25722338>
- Mascitti, M. (2020). *La función conjetural del Derecho reforzada por los*. Buenos Aires: Universidad de Buenos Aires. <https://revistascientificas.us.es/index.php/ies/article/view/13358>

- Polson, N., & Scott, J. (2018). *How Artificial Intelligence works and how we can harness its power for a better world*. London, 25-39. Penguin Random House.
- Richardson, R., Schultz, J., & Crawford, K. (2019). *Dirty Data, Bad Predictions: How Civil Rights Violations Impact Police Data, PP Systems, and Justice*. New York University Law Review Online. <https://www.nyulawreview.org/online-features/dirty-data-bad-predictions-how-civil-rights-violations-impact-police-data-predictive-policing-systems-and-justice/>
- Ruger, T. W., Kim, P. T., Martin, A. D., & Quinn, K. M. (2004). *The Supreme Court Forecasting Project: Legal and Political Science Approaches to Predicting Supreme Court Decision-Making*. Columbia Law Review. <https://www.law.berkeley.edu/files/columbia04.pdf>
- Sáez, A. J. (2023). *El posible uso de la inteligencia artificial en el ámbito judicial: contexto jurídico español y europeo. Especial referencia al contencioso-administrativo*. Rivista Italiana di Informatica e Diritto, 34, 35. <https://idus.us.es/handle/11441/151723>
- Sanmateo, M. G. (2017/2018). *El impacto del big data en la protección de datos personales*. Universitat Jaume. <https://repositori.uji.es/xmlui/handle/10234/175806>
- Serna, P. E. (2019). *Desarrollo e innovación en ingeniería*. ResearchGate. [https://www.researchgate.net/profile/Edgar-Serna-M/publication/339177129\\_Desarrollo\\_e\\_innovacion\\_en\\_ingenieria\\_4\\_ed/links/5e42a2f4458515072d91c468/Desarrollo-e-innovacion-en-ingenieria-4-ed.pdf](https://www.researchgate.net/profile/Edgar-Serna-M/publication/339177129_Desarrollo_e_innovacion_en_ingenieria_4_ed/links/5e42a2f4458515072d91c468/Desarrollo-e-innovacion-en-ingenieria-4-ed.pdf)
- Weizenbaum. (1996). *ELIZA—A Computer Program for the Study of Natural Language Communication Between Man and Machine*. <https://web.stanford.edu/class/cs124/p36-weizenbaum.pdf>
- Wolstenholme, K., Sutton, A., & Hutchinson, L. (2021). *Facing the future with confidence*. PwC Law Firms. <https://www.pwc.co.uk/industries/law-firms/law-firm-survey-report-2021.pdf>

#### **Sitios web:**

- AEPD. (2023). *Guía de Protección de Datos por Defecto*. <https://www.aepd.es/guias/guia-proteccion-datos-por-defecto.pdf>
- Breslin, S. (2023). *Everything You Need To Know About Amazon's A9 Algorithm*. Rrepricerexpress. <https://www.repricerexpress.com/amazons-algorithm-a9/>
- Cayón, J. I. (2020). *La inteligencia artificial jurídica: nuevas herramientas y perspectivas metodológicas para el jurista*. Open Edition, 41. <https://journals.openedition.org/revus/6547>
- Farnós, J. D. (2015). *Proyección algorítmica y aprendices: El rol de la Inteligencia artificial*. Educación Disruptiva. <https://juandomingofarnos.wordpress.com/2015/08/20/proyeccion-algoritmica-y-aprendices-ed-disruptiva/>
- Ferrero, R. (2020). *¿Qué es la correlación estadística y cómo interpretarla?* Máxima Formación. <https://www.maximaformacion.es/blog-dat/que-es-la-correlacion-estadistica-y-como-interpretarla/>

- Gibbs, S. (2016). *Chatbot Lawyer Overturns 160,000 Parking Tickets in London and New York*. The Guardian. <https://www.theguardian.com/technology/2016/jun/28/chatbot-ai-lawyer-donotpay-parking-tickets-london-new-york>
- Gobierno de Nueva Gales del Sur. (2022). *Artificial intelligence and litigation-future possibilities*. [https://www.judcom.nsw.gov.au/publications/benchbks/judicial\\_officers/artificial\\_intelligence\\_and\\_litigation.html](https://www.judcom.nsw.gov.au/publications/benchbks/judicial_officers/artificial_intelligence_and_litigation.html)
- IBM. (2023). *What is artificial intelligence (AI)?*. ibm.com. <https://www.ibm.com/topics/artificial-intelligence>
- Innovación digital 360. (2023). *Test de Turing: qué es, cómo funciona y qué robots lo superan*. Innovación digital 360. <https://www.innovaciondigital360.com/blockchain/prueba-de-concepto-que-es-para-que-sirve-y-por-que-es-importante/>
- IFD Technologies. (2021). *Big data y análisis predictivo, un trabajo en conjunto*. <https://ifd.com.mx/es/analisis-predictivo/>
- Kelleher, D. A. (2022). *Ley de Moore – Ahora y en el Futuro*. Intel. <https://www.intel.la/content/www/xl/es/newsroom/opinion/moore-law-now-and-in-the-future.html>
- Media, A. (2019). *Thomson Reuters to Offer Judicial Analytics with Westlaw Edge Update*. Law.com. [https://finance.yahoo.com/news/thomson-reuters-offer-judicial-analytics-080103346.html?guccounter=1&guce\\_referrer=aHR0cHM6Ly93d3cuZ29vZ2xlLmNvbS8&guce\\_referrer\\_sig=AQAAAAnhxkGZnLHIVJhRPDzBqF3XDsQI-V7U4DK-TaMIJBd3sFPWOpP7Y4-gT1h-ecf8iEu7bFAspgioMDCUNxkBqjB](https://finance.yahoo.com/news/thomson-reuters-offer-judicial-analytics-080103346.html?guccounter=1&guce_referrer=aHR0cHM6Ly93d3cuZ29vZ2xlLmNvbS8&guce_referrer_sig=AQAAAAnhxkGZnLHIVJhRPDzBqF3XDsQI-V7U4DK-TaMIJBd3sFPWOpP7Y4-gT1h-ecf8iEu7bFAspgioMDCUNxkBqjB)
- Méndez, D., & Sánchez, A. (2023). *La inteligencia artificial y los algoritmos en el ámbito judicial: ¿una oportunidad o una amenaza para la seguridad jurídica?* Revista de Derecho Privado y Empresarial. <https://dialnet.unirioja.es/servlet/r>
- Mutualidad. (2022). *Del sistema Prometea al juez-robot: la inteligencia artificial aplicada en la resolución judicial*. Mutualidad. <https://tudefinestufuturo.mutualidad.com/innovacion/del-sistema-prometea-al-juez-robot-la-inteligencia-artificial-aplicada-en-la-resolucion-judicial/2022/>
- Pastukhov, D. (2022). *Inside Spotify's Recommender System: A Complete Guide to Spotify Recommendation Algorithms*. Music Tomorrow. <https://www.music-tomorrow.com/blog/how-spotify-recommendation-system-works-a-complete-guide-2022>
- Puente, M. (2019). *LAPD pioneered predicting crime with data. Many police don't think it works*. Los Angeles Times. <https://www.latimes.com/local/lanow/la-melapd-precision-policing-data-20190703-story.html>
- Python Software Foundation. (2004). Python.org. <https://docs.python.org/es/3/library/pickle.html>
- Qian, I., Xiao, M., Mozur, P., & Cardia, A. (2022). *Four Takeaways From a Times Investigation Into China's Expanding Surveillance State*. The New York Times. <https://www.nytimes.com/2022/06/21/world/asia/china-surveillance-investigation.html>

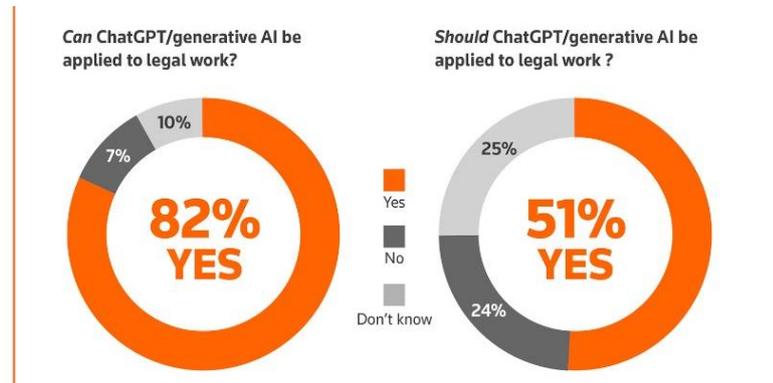
- Rosen, J. (2022). *Flawed AI Makes Robots Racist, Sexist*. Georgia Tech. College of Computing. <https://www.cc.gatech.edu/news/flawed-ai-makes-robots-racist-sexist>
- SAS Institute Inc. (2023). *Analítica predictiva. Qué es y por qué es importante*. Méjico. [https://www.sas.com/es\\_es/insights/analytics/what-is-analytics.html](https://www.sas.com/es_es/insights/analytics/what-is-analytics.html)
- Smith, N. (2021, April 22). *Engineering & Technology*. IET Sites. <https://eandt.theiet.org/content/articles/2021/04/you-need-to-identify-the-task-or-range-of-tasks-that-ai-can-help-with-professor-katie-atkinson/>
- Stox, P. (2023). *Google Still Uses PageRank. Here's What It Is & How They Use It*. Ahrefs blog. <https://ahrefs.com/blog/google-pagerank/>
- The MathWorks, Inc. (s.f.). *¿Qué es Reinforcement Learning?* MathWorks. <https://es.mathworks.com/discovery/reinforcement-learning.html>
- Thompson Reuters. (2023). *New report on ChatGPT & generative AI in law firms shows opportunities abound, even as concerns persist*. Thomsonreuters.com. <https://www.thomsonreuters.com/en-us/posts/technology/chatgpt-generative-ai-law-firms-2023/>
- Tuulik, M.-E. (2022). *Estonia does not develop AI Judge*. Republic of Estonia Ministry of Justice. <https://www.just.ee/en/news/estonia-does-not-develop-ai-judge> accesse
- UNESCO. (2023). *La IA y el Estado de derecho. Fortalecimiento de capacidades para los sistemas judiciales*. UNESCO. <https://www.unesco.org/es/artificial-intelligence/rule-law/mooc-judges>
- Universidad Europea. (2022). *Aprendizaje supervisado y no supervisado*. Empresa y Tecnología. <https://universidadeuropea.com/blog/aprendizaje-supervisado-no-supervisado/>
- Walter, C. (2005). *Kryder's Law*. Scientific America. <https://www.scientificamerican.com/article/kryders-law/>
- Yilmaz, H. (2022). *Is AI Ethic?: Ethical Concerns About AI Development*. Plugger.ai. <https://www.plugger.ai/blog/is-ai-ethic-ethical-concerns-about-ai-development>

#### **Otras fuentes:**

- BBC (2016). *Artificial Intelligence and the Law* [Grabado por R. Fenton-Smith, & B. Crighton]. Londres. <https://www.bbc.co.uk/programmes/b07dlxmj>
- Parlamento Europeo . (2021). *Reglamento (UE) 2021/2100 del Parlamento Europeo y del Consejo de 19 de octubre de 2021 sobre la inteligencia artificial*. EUR-Lex. <https://eur-lex.europa.eu/oj/direct-access.html>
- Popov v. Hayashi , WL 31833731 ( Ca. Sup. Ct. 2002). <https://www.quimbee.com/cases/popov-v-hayashi>

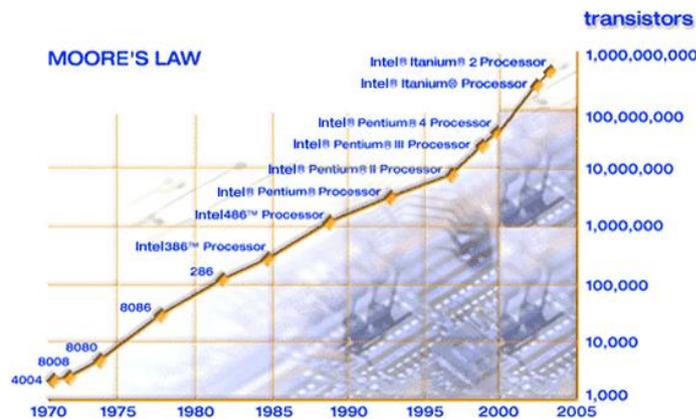
## ANEXOS

### Anexo 1. Actitudes hacia ChatGPT y la IA generativa para el trabajo legal



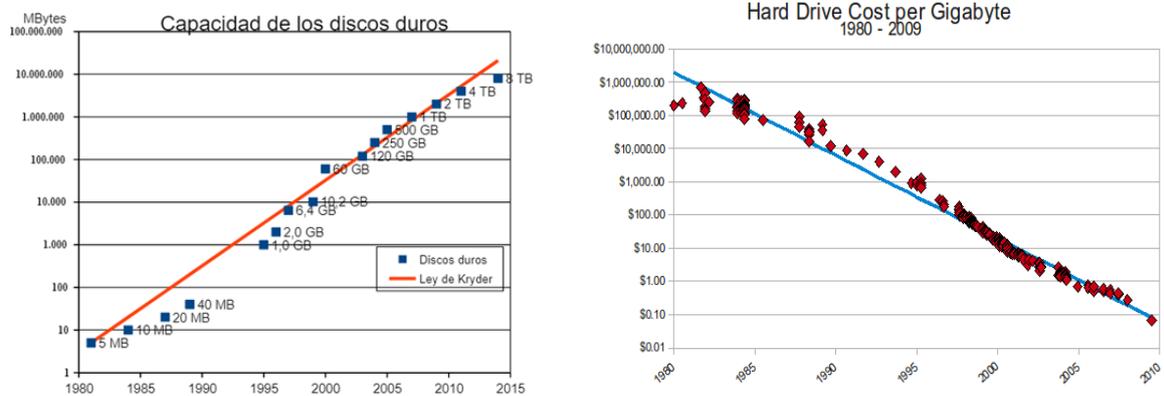
Fuente: *Thompson Reuters (2023)*

### Anexo 2. La Ley de Moore



Fuente: *Tecnología e Investigación en Intel. la Ley de Moore, el futuro (2022)*

### Anexo 3. La Ley de Kryder



Fuente: *Data Science. ¿Qué es Big Data? (2023)*

## Código de Python 3

Link: <https://colab.research.google.com/drive/1R6eJMIQs5mCHBsNtRt8b-XMsudTd2KWd?usp=sharing>

```
# Importar las librerías
import pandas as pd
import numpy as np
import pickle
import seaborn as sns
import matplotlib.pyplot as plt

# Modelos
from sklearn.ensemble import RandomForestClassifier
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.model_selection import GridSearchCV
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import GaussianNB
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier

# Evaluación
from sklearn.preprocessing import StandardScaler
from sklearn.pipeline import make_pipeline
from sklearn.metrics import accuracy_score, confusion_matrix
from sklearn.metrics import confusion_matrix, accuracy_score

# Cargar el dataset
df = pd.read_csv("cjeu_database_legalparties.csv", sep=";")
# Mostrar las primeras filas del dataset
1
df.head()

# Obetener la dimensión de los datos
# Describir los datos
df.shape
# obtener alguna información del dataset
df.info()

# Revisar si hay valores nulos
df.isnull().sum()

# Limpiar los datos
df.columns = [c.replace(' ', '') for c in df.columns]
df.columns = [c.replace('-', '') for c in df.columns]
df.columns

# Describir los datos
descr=df.describe()
descr

# Obtener todos
abt = pd.get_dummies(df)
abt.head()

# Obtener el gráfico de la distribución de la variable has_judgment
plt.figure(figsize=(8, 6))
sns.countplot(x='has_judgment', data=df)
plt.title('Distribution of has_judgment Variable')
```

```

plt.xlabel('has_judgment')
plt.ylabel('Count')
plt.show()
# Obtener el gráfico de has_judgment vs. count_decisions
plt.figure(figsize=(8, 6))
sns.boxplot(x='has_judgment', y='count_decisions', data=df)
plt.title('has_judgment vs. count_decisions')
plt.xlabel('has_judgment')
plt.ylabel('Count of Decisions')
plt.show()
# Obtener el gráfico de has_judgment vs. count_on_appeal
plt.figure(figsize=(8, 6))
sns.boxplot(x='has_judgment', y='count_on_appeal', data=df)
plt.title('has_judgment vs. count_on_appeal')
plt.xlabel('has_judgment')
plt.ylabel('Count on Appeal')
plt.show()
# Preprocesamiento
# Eliminar columnas irrelevantes
df.drop(['key_id', 'court_id', 'court', 'iuropa_case_id',
        'cjeu_case_id', 'case_year', 'case_number', 'case_name',
        'transferred_to', 'joined_to', 'list_on_appeal', 'list_appeal_of',
        'list_proceedings', 'list_decisions', 'has_opinion', 'party_role',
        'iuropa_party_id', 'party_type'], axis=1, inplace=True)
# Arreglar valores que faltan (si los hubiese)
df.fillna(0, inplace=True)
# Dividir los datos en características y target variable
X = df.drop('has_judgment', axis=1)
y = df['has_judgment']
# Dividir los datos en conjuntos de entrenamiento y prueba
X_train, X_test, y_train, y_test = train_test_split(X, y,
        test_size=0.2, random_state=42)
print(len(X_train), len(X_test), len(y_train), len(y_test))
# Utilizar regresión logística para construir el primer modelo
lr = LogisticRegression()
lr.fit(X_train, y_train)
lr_predict = lr.predict(X_test)
lr_predict_prob = lr.predict_proba(X_test)
print(lr_predict)
print(lr_predict_prob[:,1])
# Métricas
lr_conf_matrix = confusion_matrix(y_test, lr_predict)
lr_accuracy = accuracy_score(y_test, lr_predict)
print(lr_conf_matrix)
print(lr_accuracy)
# Con Naive Bayes
gnb = GaussianNB()
gnb.fit(X_train, y_train)
gnb_predict = gnb.predict(X_test)

```

```

gnb_predict_prob = gnb.predict_proba(X_test)
print(gnb_predict)
print(gnb_predict_prob)
gnb_conf_matrix = confusion_matrix(y_test, gnb_predict)
gnb_accuracy_score = accuracy_score(y_test, gnb_predict)
print(gnb_conf_matrix)
print(gnb_accuracy_score)
# Con Árboles de decisión para clasificar
# Establecemos el número de árboles en 10
dt = DecisionTreeClassifier(max_depth=10)
dt.fit(X_train,y_train)
dt_predict = dt.predict(X_test)
dt_predict_prob = dt.predict_proba(X_test)
dt_conf_matrix = confusion_matrix(y_test, dt_predict)
dt_accuracy_score = accuracy_score(y_test, dt_predict)
print(dt_conf_matrix)
print(dt_accuracy_score)
# Con Random Forest
rf = RandomForestClassifier(max_depth=10) #10 trees
rf.fit(X_train, y_train)
rf_predict = rf.predict(X_test)
rf_predict_prob = rf.predict_proba(X_test)
rf_conf_matrix = confusion_matrix(y_test,rf_predict)
rf_accuracy_score = accuracy_score(y_test, rf_predict)
print(rf_conf_matrix)
print(rf_accuracy_score)
# Entrenar y ajustar dos pipelines de aprendizaje automático, uno
para Random Forest (rf) y otro para Gradient Boosting (gb)
pipelines = {
    "rf": make_pipeline(StandardScaler(),
RandomForestClassifier(random_state=123)),
    "gb": make_pipeline(StandardScaler(),
GradientBoostingClassifier(random_state=123))
}
# Para Random Forest (rf), los hiperparámetros incluyen el número
de estimadores (n_estimators)
# y el número máximo de características a considerar para el
particionado (max_features)

rf_hyperparameters = {
    "randomforestclassifier__n_estimators": [100, 200],
    "randomforestclassifier__max_features": ["sqrt", "log2",
0.33] # Valid options for max_features
}

# Para Gradient Boosting (gb), los hiperparámetros incluyen el
número de estimadores (n_estimators),

```

```

# la tasa de aprendizaje (learning_rate) y la profundidad máxima de
los estimadores de regresión individuales (max_depth)

gb_hyperparameters = {
    "gradientboostingclassifier__n_estimators": [100, 200],
    'gradientboostingclassifier__learning_rate': [0.05, 0.1, 0.2],
    'gradientboostingclassifier__max_depth': [1, 3, 5]
}

hyperparameters = {"rf": rf_hyperparameters, "gb":
gb_hyperparameters}

# Crear un diccionario vacío llamado fitted_models
fitted_models = {}

# Iterar a través de los pipelines de modelos, ajustando cada uno y
guardándolo en fitted_models
for name, pipeline in pipelines.items():

    # Crear un objeto de validación cruzada a partir del pipeline y
los hiperparámetros
    model = GridSearchCV(pipeline, hyperparameters[name], cv= 10,
n_jobs= -1)

    # Ajustar el modelo en X_train, y_train
    model.fit(X_train, y_train)

    # Guardar el modelo en fitted_models[name]
    fitted_models[name] = model

    # Imprimir '{name} ha sido ajustado'
    print(name, 'ha sido ajustado.')

# Imprimir los mejores resultados
for name, model in fitted_models.items():
    print(name, model.best_score_)
# Imprimir la precisión estimada y la matriz de confusión
for name, model in fitted_models.items():
    print('Name:', name)
    pred = model.predict(X_test)

    # Calcular la precisión
    accuracy = accuracy_score(y_test, pred)
    print('Acc:', accuracy)

    # Calcular la matriz de confusión
    cm = confusion_matrix(y_test, pred)
    print("Confusion Matrix:\n", cm)

```

```

# Mostrar la gráfica de la matriz de confusión
plt.figure(figsize=(6, 4))
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', cbar=False)
plt.title(f'Confusion Matrix - {name}')
plt.xlabel('Predicted Label')
plt.ylabel('True Label')
plt.show()

# Relevancia de las variables del Random Forest
importances_rf =
fitted_models['rf'].best_estimator_.named_steps['randomforestclassi
fier'].feature_importances_
features_rf = X_train.columns
sort_indices_rf = np.argsort(importances_rf)[::-1]
sorted_features_rf = [features_rf[idx] for idx in sort_indices_rf]

plt.figure(figsize=(10, 6))
plt.bar(range(len(importances_rf)),
importances_rf[sort_indices_rf], align='center')
plt.xticks(range(len(importances_rf)), sorted_features_rf,
rotation='vertical')
plt.xlabel('Features')
plt.ylabel('Importance')
plt.title('Feature Importances (Random Forest)')
plt.show()

# Importancia de las características del Gradient Boosting
importances_gb =
fitted_models['gb'].best_estimator_.named_steps['gradientboostingcl
assifier'].feature_importances_
features_gb = X_train.columns
sort_indices_gb = np.argsort(importances_gb)[::-1]
sorted_features_gb = [features_gb[idx] for idx in sort_indices_gb]

plt.figure(figsize=(10, 6))
plt.bar(range(len(importances_gb)),
importances_gb[sort_indices_gb], align='center')
plt.xticks(range(len(importances_gb)), sorted_features_gb,
rotation='vertical')
plt.xlabel('Features')
plt.ylabel('Importance')
plt.title('Feature Importances (Gradient Boosting)')
plt.show()

# Usando pickle para exportar el modelo entrenado
with open('final_model_rf.pkl', 'wb') as filePickle:
    pickle.dump(fitted_models['rf'].best_estimator_, filePickle)

```