Universidad Torcuato Di Tella
Escuela de Derecho

Revista Argentina de Teoría Jurídica (RATJ)

Volumen 23, Número 1, diciembre 2022

Análisis Computacional del Derecho Argentino

David Mielnik

Formato de cita recomendado
David Mielnik, "Análisis Computacional del Derecho Argentino", Revista Argentina de Teoría Jurídica 23 1 (2022)

ISSN edición impresa 1851-6831
ISSN edición digital 1851-684X

Análisis Computacional del Derecho Argentino[*]

David Mielnik [†]

Resumen:
El conocimiento actualizado de las normas y la jurisprudencia es una responsabilidad de primera magnitud para cualquier profesional del derecho. Ellas son la materia prima a partir de la cual elaboramos estrategias de litigación, construimos argumentos y participamos del debate académico. Pero cumplir acabadamente con esa responsabilidad nos pone frente a un desafío creciente, pues hoy en día los tribunales publican literalmente miles de decisiones por mes. En efecto, el derecho moderno se está convirtiendo cada vez más en un fenómeno difícil de abarcar para las técnicas tradicionales con las que lo abordamos.
En los últimos años, la democratización del acceso a recursos informáticos cada vez más poderosos ha impulsado a investigadoras/es a responder a este desafío desde una mirada interdisciplinaria disruptiva que podría denominarse *análisis computacional del derecho* (*ACD*): un paradigma que promueve el estudio del derecho a través de tecnologías como la inteligencia artificial para potenciar nuestras capacidades analíticas. En este trabajo ofrezco la primera introducción al ACD en castellano, realizo un estudio computacional efectuado sobre más de 50.000 sentencias judiciales, y muestro cómo de esa manera es posible echar luz sobre diversos aspectos del derecho argentino.

Abstract:
Maintaining an up-to-date knowledge of the legal rules and precedents in force is a first-order responsibility for every lawyer. They are the raw material from which we elaborate litigation strategies, construct arguments, and participate in the academic debate. But fulfilling that responsibility presents us with an surging challenge, as courts today publish thousands of decisions per month. Indeed, approaching modern law through our traditional analysis techniques is becoming increasingly difficult.
In recent years, the democratization of access to powerful computing resources has prompted researchers to respond to this challenge from a disruptive interdisciplinary perspective that may be called *computational analysis of law* (*CAL*): a paradigm that promotes the study of law through technologies such as artificial intelligence to enhance our analytical capabilities. In this paper I offer the first introduction to *CAL* in Spanish, I carry out a computational study of more than 50,000 court rulings and I show how such approach may enable us to shed light on various aspects of Argentine law.

[*] Este artículo se basa parcialmente en la tesis defendida por el autor como parte de los requisitos para obtener el título de Magíster en Derecho Penal de la UTDT, bajo la dirección de Marcelo Ferrante, sin cuya guía constante y edificantes comentarios críticos no podría haber completado. En la medida de lo posible se ha intentado también incorporar al texto las valiosas devoluciones y reflexiones del jurado evaluador, integrado por Gabriel Pérez-Barberá, Celia Lerman y Edgar Altszyler, así como las de la/del referí anónima/o, a quienes el autor agradece profundamente, junto con Agustín Gravano, Cecilia Hopp, Gerardo Mielnik y Mariela Sporn, que leyeron atentamente versiones tempranas del manuscrito. Todos los errores (y bugs) remanentes son de la entera responsabilidad del autor.
[†] Abogado y Máster en Derecho Penal por la Universidad Torcuato Di Tella. Se desempeña como Prosecretario Letrado en la Cámara Federal de Casación Penal. Es profesor titular de Derecho Penal y de Análisis Computacional del Derecho (UTDT).

Introducción: Derecho y *big data*

El conocimiento actualizado de las normas y la jurisprudencia es una responsabilidad de primera magnitud para cualquier profesional del derecho. Ellas son la materia prima a partir de la cual elaboramos nuestras estrategias de litigación, construimos nuestros argumentos, proponemos reformas y participamos del debate doctrinario y académico, entre muchas otras cosas. Esto es así incluso en tradiciones jurídicas como la argentina, en la que no rigen reglas fuertes de *stare decisis*, pues en cualquier caso se espera que las decisiones de los tribunales guarden algún grado de coherencia con las precedentes —o que al menos las reconozcan y expresen las razones por las que se apartan de ellas—.

Pero cumplir con esa responsabilidad nos enfrenta a un desafío creciente. Sólo para poner un ejemplo, si en 1994 una abogada penalista hubiera querido conocer la jurisprudencia de tan solo una Sala de la recientemente creada Cámara Nacional de Casación Penal, habría necesitado tomarse el tiempo para estudiar unas 80 sentencias. Sin embargo, algunos años después, esa tarea se habría vuelto sensiblemente más compleja: para 2006, la misma Sala publicaría alrededor de 1.200 sentencias en el año, y cerraría el 2021 habiendo protocolizado 2.205 decisiones. El fenómeno, por supuesto, atraviesa todas las áreas del derecho, y parece evidente entonces que cualquier jurista que hoy en día afirme dominar el estado general de la jurisprudencia criminal no conoce, en realidad, más que una pequeñísima fracción de la producción anual de los tribunales superiores.

Así, el derecho moderno se está convirtiendo cada vez más en un fenómeno difícil de abarcar para las técnicas tradicionales con las que lo abordamos y analizamos —un fenómeno, digamos, de *big data*[1]—. Claro, uno podría decir que todavía podemos concentrar nuestra atención en los precedentes "importantes", pero ¿cómo los elegimos? ¿Nos guiamos por intuición? ¿Por el comentario de colegas o publicaciones especializadas? No parece haber respuestas claras.[2] E incluso si pudiéramos encontrar un criterio de selección ideal, el costo que pagaríamos sería todavía muy alto: si el derecho es una construcción colectiva —una "novela en cadena", como la describió Ronald Dworkin[3]— trabajar solamente con un puñado de las miles y miles de decisiones

[1] Es el término con el que habitualmente nos referimos a una cantidad de información tan vasta o compleja que resulta difícil o imposible de analizar utilizando herramientas y técnicas tradicionales. Ver D Cielen, A Meysman y M Ali, *Introducing Data Science: Big Data, Machine Learning, and More* (Manning Publications 2016), p. 1. Así lo define también el Observatorio Nacional de Big Data, dependiente de la Subsecretaría de Tecnologías de la Información y las Comunicaciones, en su sitio web oficial (https://www.argentina.gob.ar/grupo-de-trabajo/observatorio-nacional-big-data/observatorio-big-data/que-es-big-data).

[2] Gustavo Cosacov advirtió esta dificultad de escala ya a finales de los '80: si bien reconocía que "[l]a producción jurisprudencial, su difusión y comentario es una fuente valiosa de información para explicitar los criterios con los cuales opera el sistema", alertaba que ella "no siempre será una muestra representativa", en particular, "considerando el tamaño del sistema de administración de justicia en las grandes concentraciones urbanas…" (Gustavo Cosacov, *El Mito de La No Impunidad*, DG de Publicaciones de la UNC 1988, p. 25).

[3] R Dworkin, *Law's Empire* (Harvard University Press 1986), p. 229.

que le dan forma implica invariablemente concentrarnos en una fracción cada vez más pequeña de su contenido.[4]

En los últimos cinco o diez años, la creciente democratización del acceso a recursos informáticos cada vez más poderosos y sofisticados ha impulsado a investigadoras e investigadores del derecho a responder a este desafío desde una mirada interdisciplinaria renovadora que podría denominarse *análisis computacional del derecho* (*ACD*),[5] *estudios legales computacionales* o, más sintéticamente, *legal analytics*[6]. Este paradigma, que forma parte de las llamadas *humanidades digitales,*[7] promueve el estudio de textos jurídicos a través de la lente de tecnologías emergentes —como el procesamiento del lenguaje natural y la inteligencia artificial— para potenciar y enriquecer nuestras capacidades analíticas.

Lejos de pretender reemplazar las técnicas hermenéuticas tradicionales, el ACD nos permite *amplificar* nuestra capacidad de lectura con una herramienta fundamental, a saber, el poder de procesamiento de la informática moderna. Así, podemos desarrollar una práctica de *lectura a distancia,*[8] es decir, la capacidad de abarcar no una, diez o veinte sentencias o leyes —como venimos haciendo desde siempre— sino cientos, miles y millones.

En este trabajo ofrezco —hasta donde sé— la primera introducción al análisis computacional del derecho argentino. Procederé de la siguiente manera. En primer lugar, (I.) describo con algo más de detalle cuál es la promesa del ACD y por qué es un abordaje interdisciplinario valioso. Luego, (II.) explico de qué se trata la disciplina informática que hace posible el ACD; concretamente, cuáles son algunas de las herramientas y técnicas computacionales que podemos aplicar con éxito al estudio del derecho.

Una vez precisados los contornos del área de estudio, (III.) presento, a modo de prueba de concepto, un estudio computacional efectuado sobre más de 50.000 sentencias judiciales.

Antes de continuar, dos observaciones. Primero, este trabajo está dirigido a profesionales del derecho, actuales y futuros, por lo que no presupone ningún conocimiento informático previo.[9] Intentaré así minimizar la discusión de los aspectos más técnicos del análisis computacional, procurando ganar en claridad expositiva, aun a costa de cierta imprecisión técnica. Segundo, los ejemplos con los que ilustro la mayoría de mis observaciones provienen en general del derecho penal, que ha sido mi área central de investigación. Sin embargo, los procedimientos y

---

[4] Wolfgang Alschner, «The Computational Analysis of International Law» en Rossana Deplano y Nicholas Tsagourias (eds), *Research Methods in International Law: A Handbook* (2019), pp. 1-3.

[5] Ibid.

[6] Nina Varsava, «Computational Legal Studies, Digital Humanities, and Textual Analysis», *Computational Legal Studies* (Edward Elgar Publishing 2020).

[7] Anne Burdick et al., *Digital_Humanities* (MIT Press 2012).

[8] Se atribuye el término al historiador literario Franco Moretti quien, entre otras cosas, analizó computacionalmente más de 7.000 títulos de novelas para extraer observaciones sobre el género. Ver Franco Moretti, *Lectura Distante* (FCE - Fondo de Cultura Económica).

[9] Mi impresión, no obstante, es que las técnicas y metodologías que presento aquí se convertirán, en el corto plazo, en una parte central de la formación universitaria elemental que se esperará de las y los profesionales del derecho.

metodologías que aquí discuto pueden aplicarse con pocos ajustes al estudio de cualquier rama del derecho.

1. El análisis computacional como lectura a distancia

Si nos olvidamos por un momento de los algoritmos y los *bytes*, el análisis computacional del derecho es en el fondo una forma de lectura *a distancia* (*distant reading*),[10] en contraste con la *lectura cercana* (*close reading*), propia de las metodologías clásicas de análisis jurídico, a las que más atención hemos prestado hasta ahora.

La metáfora de la *lectura a distancia* funciona en un doble sentido. Por un lado, al igual que su contraparte cercana, la lectura a distancia de leyes o sentencias exige el empleo de conocimientos técnico-jurídicos para extraer de ellas contenido sustantivo. La introducción del procesamiento informático en esos textos, sin embargo, altera la posición relativa en la que tales saberes se ponen en juego: en la lectura cercana, los aplicamos directamente sobre los discursos que decodificamos en nuestras mentes; en el análisis computacional, por otra parte, los utilizamos primero para crear los programas y algoritmos que procesarán los textos por nosotros, y, luego, para interpretar los resultados de esas operaciones informáticas.[11] En otras palabras, la analista computacional del derecho primero "enseña" a la máquina a leer como lo haría una abogada, y luego revisa críticamente el resultado de esa tarea encomendada.

Por otro lado, el término *lectura a distancia* se relaciona con la escala que utilizamos para analizar los textos jurídicos y con la perspectiva desde la cual los observamos. En cuanto a la escala, el ACD es una metodología que propone una inclusividad inusitadamente alta de textos en las colecciones a las que prestamos atención: la lectura —mediante el procesamiento computacional— de miles o millones de textos a la vez. Eso tiene un impacto directo también en la perspectiva de análisis: mientras que la lectura cercana presta atención a las oraciones, el vocabulario, el orden de las palabras, la narrativa y el estilo, la lectura a distancia está caracterizada por el descubrimiento de patrones, tendencias y otros rasgos que no surgen de cada documento tomado individualmente, sino del conjunto, y que sólo son observables cuando se los analiza a la vez.[12]

---

[10] David Carter, James Brown y Adel Rahmani, «Reading the High Court at a Distance: Topic Modelling the Legal Subject Matter and Judicial Activity the High Court of Australia, 1903–2015» (2016) Vol. 39 The University of New South Wales Law Journal; Keith Carlson, Michael A Livermore y Daniel Rockmore, «A Quantitative Analysis of Writing Style on the US Supreme Court» (2015) Vol. 93 Wash. UL Rev.; Urska Sadl y Henrik Palmer Olsen, «Can Quantitative Methods Complement Doctrinal Legal Studies: Using Citation Network and Corpus Linguistic Analysis to Understand International Courts» (2017) Vol. 30 LJIL.

[11] De nuevo adelantándose a su tiempo, Cosacov anticipaba: "[t]ampoco la informática nos puede decir nada acerca de cómo observar un sistema; sólo hace posible una clase de observación, pero no es diseñadora de esa observación". Cosacov, p. 24.

[12] Carter, Brown y Rahmani, p. 1301.

Para ser claros, la lectura a distancia no reemplaza, sino que complementa, las metodologías tradicionales: el procesamiento de grandes volúmenes de información jurídica permite observar y dar cuenta del derecho desde una escala nunca antes alcanzada, pero no provee una comprensión detallada de cada texto. La lectura tradicional, por otra parte, aporta ese entendimiento y detalle, pero consume tiempo y recursos mentales finitos, y está, por lo tanto, limitada a trabajar con un número relativamente pequeño de documentos seleccionados. El truco está en la posibilidad de combinar ambos abordajes con el fin de potenciar sus respectivas fortalezas y minimizar sus debilidades.

Veámoslo con un ejemplo. Supongamos que queremos abordar la pregunta por el grado en que los tribunales penales han adoptado una mirada con perspectiva de género en el juzgamiento de los casos que llegan a sus estrados.[13] El modo más habitual de acercarse a esa interrogante desde una escala de cercanía consiste —al menos en parte— en evaluar cualitativamente algunas decisiones recientes, seleccionadas según algún criterio de relevancia razonable, intentar dilucidar la tendencia general y formular consideraciones críticas.

No hay nada incorrecto en ese enfoque; al contrario, al menos de momento, representa la única manera de alcanzar una comprensión profunda de cada decisión analizada. No obstante, es también un enfoque altamente susceptible de introducir sesgos en la selección de las decisiones estudiadas.[14] Al mismo tiempo, ofrecer un relevamiento genuinamente exhaustivo del estado de la cuestión, incluso si limitamos nuestro dictamen a un período y un distrito en particular, probablemente nos resultaría prohibitivamente costoso en términos de tiempo y energía.[15]

Por otra parte, una manera complementaria de abordar el tópico desde una posición de lectura más distante podría consistir en la construcción de un algoritmo informático capaz, al menos en principio, de identificar *todos* los fallos que de alguna manera han mirado el derecho en clave no discriminatoria, y, posteriormente, en la presentación de los resultados mediante gráficos que resuman los hallazgos. Una aproximación a este problema (por cierto, todavía muy tosca) puede verse en la Figura 1.1, que muestra cómo fue variando la proporción de fallos que reflejan, en algún sentido relevante, consideraciones con perspectiva de género, tanto en la muestra general de sentencias definitivas de los tribunales orales en lo criminal con asiento en la Ciudad de Buenos

[13] Ello, en cumplimiento de las obligaciones internacionales que conminan al Estado argentino, entre otras cosas, a "modificar prácticas jurídicas o consuetudinarias que respalden la persistencia o la tolerancia de la violencia contra la mujer" (Ver art. 7 de la Convención Interamericana para Prevenir, Sancionar y Erradicar la Violencia contra la Mujer ("Convención de Belém do Pará").

[14] Si bien ese riesgo ciertamente está presente también en los estudios computacionales, mi impresión es que éstos están mejor posicionados para mitigarlo, ya sea trabajando con muestras compuestas virtualmente por la totalidad de los fallos pronunciados, o empleando protocolos y metodologías estadísticas largamente testeados, que sencillamente no resultan aplicables en la escala en la que se manejan los estudios tradicionales del derecho.

[15] Sólo considerando el período 2014-2019, los tribunales orales en lo criminal y correccional (TOC) con asiento en la CABA publicaron nada menos que 18.170 decisiones.

Aires, como en el marco de una selección de delitos en los que la adopción de esa perspectiva resulta más urgente.[16]

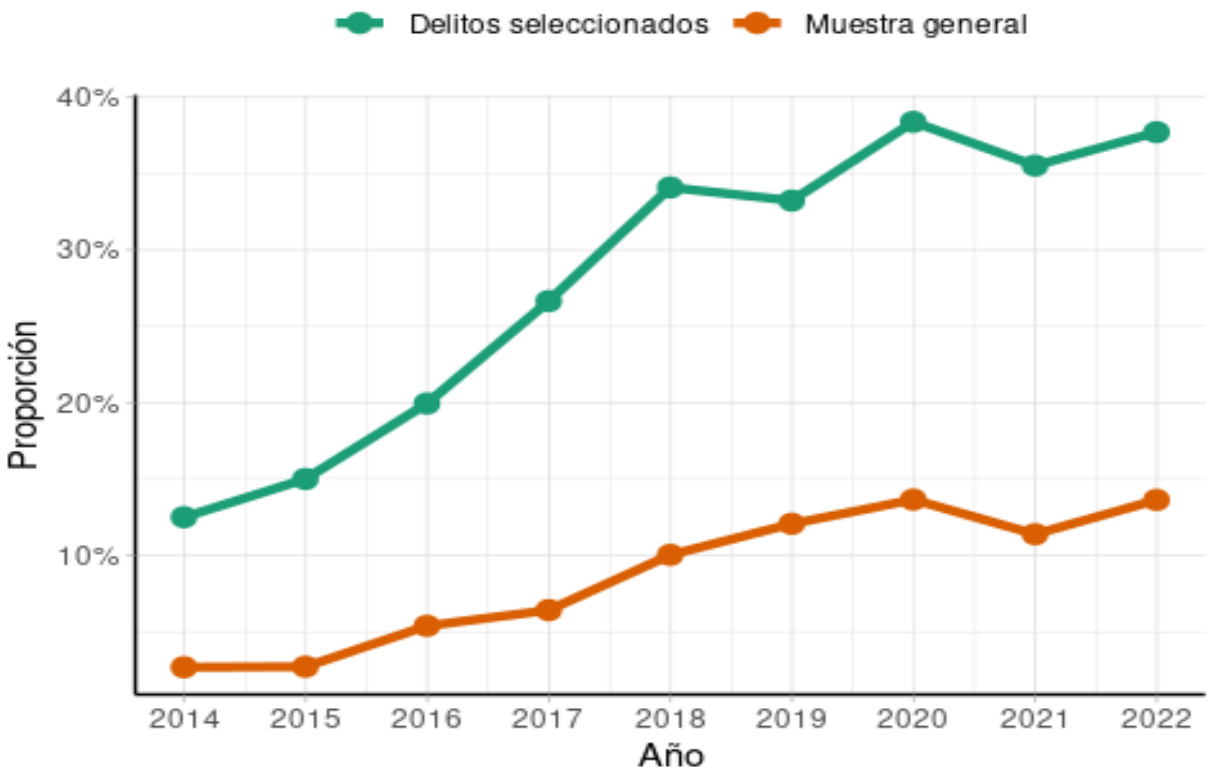

Figura 1.1: Proporción de sentencias de TOC que hacen "consideraciones con perspectiva de género" (2014-2019).

El análisis de lo que la curva sugiere y de sus implicancias (y, sobre todo, de las variables independientes que pueden explicarla) excede largamente el objeto de este trabajo. Aquí sólo la presento como una prueba de concepto. El punto que me interesa enfatizar es que ninguno de los abordajes —el tradicional y el computacional— es necesariamente superior al otro. En efecto, la invitación es a reconocer las limitaciones de cada uno y, en última instancia, a adoptarlos conjuntamente.

2. Ciencia de datos y derecho

En el corazón del análisis computacional del derecho se encuentra la observación de que el derecho es, entre otras cosas, un conjunto de datos, susceptibles de ser analizados y procesados por una máquina. Es verdad que cuando uno piensa en "datos" habitualmente se imagina un

[16] Por ejemplo, el femicidio, el abuso sexual, las lesiones y las amenazas.

conjunto de números, prolijamente organizados en filas y columnas en una tabla de Excel. Pero, esa es sólo una forma en la que los datos pueden presentarse —como datos tabulares o *estructurados*—. El video, el audio y el texto también son datos —datos *no estructurados*—, igualmente susceptibles de procesamiento y análisis computacional.

Hasta hace relativamente poco, las limitaciones en las capacidades de procesamiento de las computadoras personales hacían que el análisis de grandes volúmenes de datos (y, particularmente, de datos no estructurados) fuera sólo alcanzable para grandes corporaciones, gobiernos o centros de investigación especializados. Hoy en día, no obstante, las computadoras de escritorio modernas poseen, en general, potencia más que suficiente para llevar adelante análisis informáticos a escala, lo que ha propiciado una verdadera "democratización del análisis de datos".[17]

El conjunto de metodologías informáticas por medio de las cuales los datos son obtenidos, procesados y utilizados para obtener conocimiento que contribuya a la toma de decisiones es habitualmente denominado *ciencia de datos.*[18] El análisis computacional del derecho, así, es una forma de ciencia de datos aplicada. Dentro de la vasta caja de herramientas de la ciencia de datos hay al menos tres grandes grupos de métodos y técnicas que son especialmente interesantes: la automatización, el procesamiento del lenguaje natural (*NLP*, por sus siglas en inglés) y el aprendizaje automático (o *machine learning*) —quizás la cara más conocida y desarrollada de lo que habitualmente llamamos "inteligencia artificial"—.

2.1 Automatización

Para ser claros, todo proceso computacional supone alguna forma de automatización (por algo la traducción más habitual de *machine learning* es aprendizaje *automático*). Pero hay también un sentido más "mecánico" del término que me interesa comentar aquí, dados los propósitos de este trabajo. En esta segunda acepción, automatizar supone sencillamente realizar tareas repetitivas de manera rápida y eficiente, sin intervención humana. El objetivo no sólo es ahorrar tiempo sino, fundamentalmente, evitar introducir errores humanos.

La automatización interviene especialmente en al menos dos momentos cruciales del ciclo de trabajo del análisis computacional del derecho: en la adquisición, almacenamiento y pre-procesamiento de los datos, al comienzo, y en la presentación de los resultados para ser consumidos e interpretados, al final. Para un ejemplo del primer momento, pensemos en lo engorroso que resulta muchas veces interactuar con las bases de jurisprudencia o leyes para

[17] Laura Igual y Santi Seguí, *Introduction to Data Science* (Springer 2017), p. 2.

[18] Igual y Seguí, pp. 2-3. Quizás el término no sea el más apropiado —después de todo, toda ciencia se basa en datos— pero en cualquier caso el concepto sirve para designar un conjunto ecléctico de técnicas y habilidades informáticas que tienen como denominador común el aprovechamiento del poder de cómputo de los ordenadores, y que son empleadas en las más variadas áreas del conocimiento para potenciarlas y enriquecerlas. En el mismo sentido, Jake VanderPlas, *Python Data Science Handbook: Essential Tools for Working with Data* (O'Reilly 2017), p. 1.

obtener unas pocas sentencias relevantes (e incluso una sola). Ahora imaginemos no sólo que necesitamos obtener *miles* de esas sentencias para realizar estudios a gran escala, sino que debemos asignar a cada documento un nombre descriptivo, junto con *metadatos* como su fecha, el tribunal emisor, etc.

Por otro lado, los datos —y especialmente los datos no estructurados— muchas veces deben "limpiarse" antes de poder ser procesados computacionalmente. De las sentencias judiciales, por ejemplo, nos interesa despejar cualquier imagen (como el escudo institucional) o artefacto generado por los mismos sistemas informáticos de gestión de causas. Preparar cada archivo demandaría un enorme esfuerzo artesanal; pero, habitualmente, ese tipo de tareas pueden ser automatizadas utilizando algunas líneas de código de programación.

La automatización permite también producir reportes, gráficos y hasta visualizaciones interactivas, de forma que los resultados del procesamiento puedan ser comunicados, interpretados y aprovechados para la toma de decisiones reales.[19]

2.2 Procesamiento del lenguaje natural

El procesamiento del lenguaje natural (*Natural Language Processing*, o *NLP*) es el área de la ciencia de datos que posibilita que las computadoras dejen de ver a los archivos que contienen el texto de sentencias o leyes como meras cadenas de caracteres, y comiencen a identificar las estructuras —palabras, oraciones, párrafos— que hacen de esos documentos, un discurso humano. La Figura 2.1, por caso, muestra cómo las máquinas pueden reconocer las dependencias sintácticas de cada palabra en una oración, utilizando procesos cada vez más sofisticados y precisos.

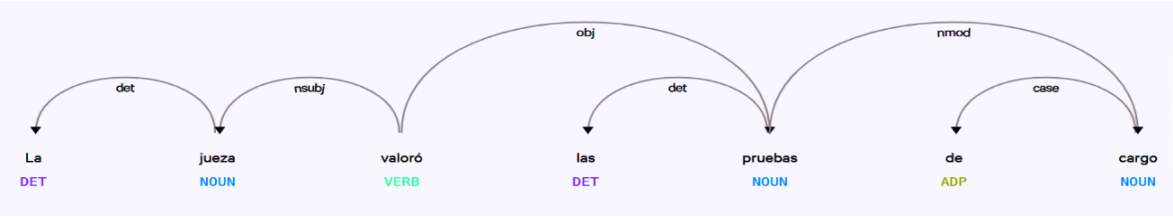

Figura 2.1: Reconocimiento automático de dependencias sintácticas

Para lograr ese propósito, casi cualquier operación de procesamiento del lenguaje exige aplicar diversas transformaciones al texto para así obtener una representación matemática de sus rasgos esenciales —un vector—, sobre las que las computadoras pueden operar de diversas maneras. Esas representaciones, o *modelos de lenguaje*, pueden obtenerse a través de procesos relativamente simples, como el recuento de la frecuencia de cada término único (los llamados modelos de "bolsa de palabras"), o tan complejos como la implementación de redes neuronales

[19] Por ejemplo: Alberto Cairo, *The Truthful Art: Data, Charts, and Maps for Communication* (New Riders 1999) y Juuso Koponen y Jonatan Hildén, *Data Visualization Handbook* (Aalto korkeakoulusäätiö 2019).

profundas, capaces de identificar el sentido de cada palabra en función de los contextos en los que aparecen (los modelos de *word embedding*).

Si bien hoy en día gran parte del procesamiento del lenguaje depende de alguna forma de aprendizaje automático, lo cierto es que, para los propósitos de la investigación jurídica, el valor del NLP no puede reducirse a un subconjunto de procesos de inteligencia artificial,[20] y de hecho puede ofrecer herramientas que por sí mismas son capaces de satisfacer una enorme cantidad de demandas del análisis computacional del derecho, muchas veces de manera más simple, accesible y eficiente que las soluciones más sofisticadas y técnicamente demandantes a las que con frecuencia (innecesariamente) se recurre. Los *lexicones*[21] y las *expresiones regulares,*[22] por caso, son herramientas comparativamente sencillas, pero extraordinariamente poderosas. Estas últimas, en particular, permiten identificar y extraer patrones relativamente complejos en el texto para así, por ejemplo -y entre muchas otras cosas- identificar a los integrantes del tribunal, la fecha y el objeto de la sentencia, o las referencias normativas y doctrinarias; segmentar las decisiones en sus partes relevantes (p. ej., antecedentes, votos de los jueces, fallo, etc.); o, incluso, clasificar los documentos de acuerdo a la rama del derecho a la que pertenecen.

Representar los textos como vectores también abre las puertas a la aplicación de técnicas y métodos tradicionalmente reservados para aplicaciones matemáticas. Por ejemplo, podemos estimar la similitud entre sentencias o leyes como una función de la cantidad de fragmentos de texto que comparten utilizando herramientas analíticas de la teoría de conjuntos, o calculando la distancia angular entre las coordenadas de los vectores que representan los textos en el espacio. De esa manera, por ejemplo, es posible identificar la emergencia de doctrinas recurrentes o el grado en el que dos códigos procesales comparten una matriz común.

2.3 Aprendizaje automático

El aprendizaje automático (o *machine learning*) es la tecnología detrás del mayor proceso de maduración que ha atravesado la inteligencia artificial en los últimos años; algunos, incluso, lo identifican como motor principal de una nueva revolución industrial. La ciencia de datos es en gran medida posible gracias a los avances logrados en esta área y, en particular, es en la intersección entre NLP y *machine learning* de donde seguramente surjan —aunque ciertamente no de manera excluyente— algunas de las principales herramientas del análisis computacional del derecho, como la que describo en las secciones siguientes.

A modo de brevísima (y probablemente inexacta) introducción al aprendizaje automático para abogados, hay que decir que el término refiere a sistemas inductivos que tienen la capacidad de

[20] Ver Sección 2.3.

[21] Por ejemplo: Venkateswarlu Bonta, Nandhini Kumaresh y N Janardhan, «A Comprehensive Study on Lexicon Based Approaches for Sentiment Analysis» (2019) 8 Asian Journal of Computer Science and Technology, p. 1.

[22] H Lane, H Hapke y C Howard, *Natural Language Processing in Action: Understanding, Analyzing, and Generating Text with Python* (Manning Publications 2019), pp. 343-346.

ajustar automáticamente sus parámetros internos de funcionamiento y comportamiento, en respuesta a la exposición a información.[23] Este paradigma contrasta con la programación tradicional, en la que el o la programadora debe anticiparse y prever cada una de las respuestas posibles a los *inputs* que recibe el *software*.[24]

Veámoslo con un ejemplo simple. Supongamos que queremos programar un sistema que nos permita traducir temperaturas expresadas en Celsius (°C) a Farenheit (°F). La forma tradicional de hacer eso es introduciendo explícitamente la regla de conversión de una unidad a otra ($F = \frac{9}{5}C + 32$) dentro del programa, de modo que si ingreso el número 100, el sistema reemplazará *C* por ese valor y calculará fácilmente el valor de *F*. Bajo el paradigma del aprendizaje automático, por otro lado, lo que haríamos sería exponer al programa a un conjunto de mediciones de temperaturas expresadas tanto en °C como en °F, y dejaríamos que el sistema *aprendiera* (metafóricamente) la regla de conversión implícita.

A primera vista esto luce como un desperdicio de recursos, y para ciertos problemas en efecto lo es (otro recordatorio de que la inteligencia artificial no siempre es la solución correcta). Sin embargo, en muchos casos simplemente *no* conocemos las reglas de conversión relevantes; es más: a medida que la cantidad de variables involucradas crece, la inferencia de esas reglas se vuelve cada vez más impracticable con métodos tradicionales. Pensemos por ejemplo en la estimación del valor de una casa —para lo cual tendríamos en cuenta su ubicación, superficie, cantidad de habitaciones, servicios, etc.— o, para ir acercándonos al objeto de este trabajo, la identificación de las cientos o miles de propiedades de un caso que potencialmente pueden explicar la decisión adoptada por el tribunal que lo adjudica.

Kevin Ashley explica este proceso inductivo en los siguientes términos:

> "Los algoritmos de *machine learning* identifican patrones en los datos, resumen esos patrones en un *modelo*, y utilizan ese modelo para identificar los mismos patrones en nuevos datos".[25]

Así, el resultado de un proceso de aprendizaje automático es la construcción de un *modelo*, esto es, "la representación matemática de las relaciones entre las variables que surgen de los datos a los que el algoritmo es expuesto".[26] En pocas palabras, es la definición de lo que algo —un perro, una naranja, un recurso de casación exitoso, etc.— es, en función de las propiedades con las que describimos ese algo. Una vez construida esa definición, podemos luego generalizarla a datos a los que todavía el sistema *no* ha sido aplicado.

Para ponerlo con un ejemplo: si exponemos un algoritmo de *machine learning* a una cantidad suficiente de observaciones de perros, gaviotas y elefantes —o mejor dicho, a las

[23] Pedro Domingos, «A Few Useful Things to Know about Machine Learning» (2012) Vol. 55 Communications of the ACM, p. 81.

[24] Igual y Seguí, p. 67.

[25] Kevin D Ashley, *Artificial Intelligence and Legal Analytics: New Tools for Law Practice in the Digital Age* (Cambridge University Press 2017), p. 234.

[26] Joel Grus, *Data Science from Scratch: First Principles with Python* (O'Reilly Media 2015), pp. 141-142.

propiedades con las que podríamos describirlos: su peso, altura, tipo de dieta, si tiene trompa o no, si vuela, etc.— el sistema será capaz de construir un modelo, una definición aproximada, de lo que un perro, una gaviota y un elefante son, respectivamente. Así, será luego capaz de reconocer un animal no identificado —con una probabilidad habitualmente muy elevada— en función de esa definición aprendida (siempre y cuando, claro, el animal pertenezca a alguna de las clases aprendidas).

La estimación de una cantidad continua (como la predicción del valor de mercado de un inmueble o un automóvil, o el tiempo que insumirá un proceso judicial) es formalmente una tarea de *regresión*; mientras que la identificación de un valor discreto, como la determinación de la rama del derecho a la que pertenece predominantemente una ley, es una de *clasificación.*[27] Las dos son a su vez tareas propias de algoritmos de aprendizaje *supervisado* —la variedad más ampliamente utilizada y madura[28]—.

Hay una gran cantidad de algoritmos que permiten realizar tareas de clasificación pero,[29] más allá de diferencias de funcionamiento interno, todos los algoritmos de *machine learning* supervisado siguen aproximadamente el mismo proceso de aprendizaje, que a grandes rasgos consta de tres etapas. En primer lugar, el sistema inicializa aleatoriamente los coeficientes que expresan los "pesos" $(\theta_1, \theta_2 \dots, \theta_n)$ de cada parámetro $(x_1, x_2 \dots, x_n)$ para la inferencia final. Luego, el algoritmo calcula la "función de error" o "pérdida" para cada predicción; esto es, una medida del desvío entre la respuesta estimada $(\hat{y})$ y la respuesta correcta (y). La función de error "penaliza" al sistema con menor o mayor intensidad, dependiendo de qué tan acertadas hayan sido las estimaciones. Por supuesto, al comienzo del proceso el error suele ser muy grande; después de todo, la predicción fue hecha sobre la base del puro azar.

La tercera etapa es donde ocurre el metafórico "aprendizaje" que le da el nombre a todo el proceso: en ella, el algoritmo estima la pendiente de la función de error obtenida en la fase anterior y, con esa información, identifica la dirección y el sentido en el que deben variar los coeficientes para acercarse al punto en el que la pendiente sea mínima. Formalmente, se trata de un problema de optimización matemática, en el que el cálculo infinitesimal es el gran protagonista. Las iteraciones se suceden, habitualmente en la forma de *descenso de gradiente* (o algún *spin-off*),[30] hasta que el error de la función objetivo es mínimo —aunque nunca nulo—.

Por otro lado, los algoritmos de aprendizaje *no supervisado* son aquellos en los cuales no existe una variable dependiente a predecir o estimar —de modo que no hay ninguna señal o etiqueta conocida que "supervise" u oriente al algoritmo para que pueda ir mejorando el modelo generado—. En su lugar, el aprendizaje no supervisado busca desentrañar relaciones ocultas entre

[27] GM James et al., *An Introduction to Statistical Learning* (Springer 2021), pp. 28-29.
[28] Domingos, p. 78.
[29] James et al., pp. 129-170.
[30] Sebastian Ruder, «An Overview of Gradient Descent Optimization Algorithms».

las variables u observaciones, normalmente con el propósito de agruparlas de algún modo (*clustering*) o inferir la naturaleza de patrones subyacentes a la muestra.[31]

3. Análisis Computacional de la Jurisprudencia Penal Argentina

Ahora que ya he presentado el campo de estudio del ACD, así como algunas de las técnicas y metodologías que lo hacen posible, en lo que queda de este trabajo ofrezco un primer aporte al análisis computacional del derecho argentino. Como adelanté, si bien voy a concentrarme en el procesamiento de la jurisprudencia de la Cámara Federal de Casación Penal, los métodos que presento aquí pueden ser fácilmente aplicados a los precedentes de cualquier tribunal argentino, independientemente de su especialidad o jerarquía.

3.1 Una nueva mirada sobre el derecho argentino

Un tema recurrente en la investigación jurídica —en Argentina y el mundo— consiste en la elucidación de los criterios que subyacen a los pronunciamientos de un tribunal dado. Tales "comentarios doctrinarios" suelen adoptar la forma de un análisis inferencial, en el que unas pocas sentencias son contrastadas críticamente y presentadas como ejemplo de un patrón de razonamiento judicial reconocible. La metodología, muy fructífera, es regularmente empleada en la reconstrucción de doctrinas sobre tópicos específicos,[32] pero también en la búsqueda de tendencias generales en el proceso decisional de los tribunales,[33] o incluso de juezas/ces en particular.[34] En el caso específico de la Cámara Federal de Casación Penal (CFCP), la pregunta que guía muchos de estos estudios de doctrina es acerca de las condiciones en virtud de las cuales podemos esperar que un cierto recurso sea admitido o rechazado por el tribunal.

Como he venido señalando, empero, este enfoque no está exento de limitaciones, propias de la escala analítica *cercana* de la que depende. El inescapable sesgo en la selección de los fallos comentados y la minúscula proporción que ellos representan en la producción agregada de los tribunales son sólo algunas de las más obvias.[35]

Con esas limitaciones en mente, ofrezco aquí una mirada diferente de la jurisprudencia penal argentina, formada desde una perspectiva *distante*, mediante el procesamiento de miles de sentencias publicadas por la CFCP en los últimos años. En particular, pretendo mostrar que el

---

[31] James et al., pp. 26-28.

[32] Sólo por mencionar algunos trabajos representativos, particularmente sobre doctrinas del derecho penal, ver: Sanford H Kadish, «Complicity, Cause and Blame: A Study in the Interpretation of Doctrine» (1985) Vol. 73 California Law Review; Luis E Chiesa, «Derecho Penal Sustantivo» (2019) Vol. 88 Rev. Jur. UPR; y HV Gullco, *Principios de La Parte General Del Derecho Penal: Jurisprudencia Comentada* (Editores del Puerto 2006).

[33] Pablo Larsen, «¿Cómo Razonan —o Podrían Razonar— Los Jueces Penales?» (2017) Vol. 2 Jurisprudencia de Casación Penal.

[34] Ileana Arduino, «La Nueva Conformación de La Sala III de La Cámara Nacional de Casación Penal y Su Incidencia En Las Resoluciones.» [2005] Revista Pensamiento Penal.

[35] Cosacov, p. 25.

análisis algorítmico de los precedentes del máximo tribunal penal del ordenamiento federal argentino, y la comprensión de los procesos de inteligencia artificial que intervienen en él, puede ayudarnos a identificar fiablemente la clase de propiedades de un caso —es decir, sus antecedentes procesales, los argumentos de las partes, los temas debatidos, etc.— que más peso tienen para inclinar la decisión del tribunal de casación en un sentido u otro. De esa manera, al igual que los comentarios doctrinales tradicionales, el análisis computacional permite comprender un poco mejor cómo es el proceso decisional de la CFCP.[36]

3.2 Datos

3.2.1 La Cámara Federal de Casación Penal y sus sentencias

Hay varias razones que aconsejan centrar la atención en la CFCP para realizar este primer estudio computacional de la jurisprudencia argentina. Primero, la CFCP es, con excepción de la Corte Suprema de Justicia de la Nación, el único tribunal del país que resuelve casos penales cuya competencia se extiende a todo el territorio argentino.

Segundo, en los recursos de su especialidad, la CFCP ejerce la función de revisión de las sentencias definitivas, y de resoluciones equiparables a ellas, dictadas por los tribunales orales y las cámaras de apelación de los diferentes circuitos federales.[37] Así, el análisis de la jurisprudencia de casación puede potencialmente decir algo también acerca de los tribunales cuyas decisiones revisa.

Tercero, si bien fue originalmente concebida como un tribunal nomofiláctico de intervención más bien extraordinaria, actualmente, de acuerdo con la interpretación que se ha dado al derecho a recurrir el fallo condenatorio (art. 8.2 "h" de la C.A.D.H.) en nuestro país —tomando las ideas de Marcelo Ferrante,[38] luego desarrolladas por Julio Maier[39] y, finalmente, proyectadas a la jurisprudencia de la Corte Interamericana de Derechos Humanos[40] y la Corte Suprema de la

[36] Con la finalidad de facilitar la reproducibilidad del análisis, es importante destacar que todos los fallos analizados fueron obtenidos de colecciones oficiales gratuitas y abiertas, como las que dependen del Centro de Información Judicial (CIJ), y que la programación de los diferentes algoritmos y funciones utilizadas fue realizada íntegramente con herramientas de código abierto, libremente disponibles.

[37] El art. 457 CPPN establece que el recurso de casación procede contra "sentencias definitivas y los autos que pongan fin a la acción o a la pena, o hagan imposible que continúen las actuaciones o denieguen la extinción, conmutación o suspensión de la pena". Asimismo, tal y como lo había resuelto, en relación con los tribunales superiores provinciales, en los precedentes "Strada" (Fallos: 308:490) y "Di Mascio" (Fallos: 311:2478), la Corte federal estableció en "Di Nunzio" (Fallos: 328:1108) que la Cámara Federal de Casación Penal es el "tribunal superior de la causa" a los efectos de la interposición de todo recurso extraordinario en el orden penal federal".

[38] Marcelo Ferrante, «La Garantía de Impugnabilidad de La Sentencia Penal Condenatoria Sobre La Base Del Caso "Giroldi"», *Seminario de Derecho Penal y Procesal Penal de La Facultad de Derecho y Ciencias Sociales de La UBA.* (1994).

[39] Julio BJ Maier, *Derecho Procesal Penal: Fundamentos* (2da edn, Del Puerto 2004), pp. 705-717.

[40] Caso "Herrera Ulloa v. Costa Rica". Sentencia del 2 de julio de 2004 (Excepciones Preliminares, Fondo, Reparaciones y Costas).

Nación— se exige que la CFCP efectúe una revisión integral de los casos que llegan a su conocimiento, incluyendo de las denominadas "cuestiones de hecho y prueba", hasta agotar sus posibilidades de rendimiento revisorio. Eso significa que sus decisiones constituyen una síntesis de todo el proceso y contienen, en general, una reseña más o menos detallada del procedimiento que condujo a su dictado, lo que también enriquece potencialmente el análisis y sus conclusiones.

Todas las sentencias utilizadas en esta investigación fueron obtenidas del repositorio público y de libre acceso que mantiene el Centro de Información Judicial (CIJ). En efecto, desde el 21/8/2013, la normativa vigente dispone que todas las decisiones de la Corte Suprema, las cámaras federales y nacionales de apelación y casación, así como las sentencias definitivas de los tribunales orales, se publiquen centralizadamente en ese portal de datos abiertos.[41]

La consolidación de la base de datos utilizada para análisis computacional fue automatizada mediante el desarrollo de un programa simple (*script*), que almacena cada fallo con un código identificatorio y extrae del texto —y de los metadatos del archivo— información básica de cada decisión (fecha, tribunal emisor, etc.). De esa manera, no solamente se logra realizar eficientemente la tarea, sino, sobre todo, minimizar las chances de error humano, dada su naturaleza repetitiva y mecánica.[42]

Mediante esta técnica, en definitiva, produje una base de datos conformada por 56.507 decisiones jurisdiccionales, dictadas por las cuatro salas que integran la Cámara Federal de Casación Penal, más la sala de feria.

3.2.2 Análisis exploratorio preliminar

Podemos complementar las observaciones cualitativas precedentes sobre las sentencias de la CFCP con una breve exploración cuantitativa de la muestra de fallos, con el fin de obtener algunas impresiones preliminares que permitan tomar decisiones metodológicas sobre ella. El análisis

[41] La ley 26.856, sancionada el 8 de mayo de 2013, establece que "…la Corte Suprema de Justicia de la Nación y los tribunales de segunda instancia que integran el Poder Judicial de la Nación deberán publicar íntegramente todas las acordadas y resoluciones que dicten, el mismo día de su dictado". Asimismo, dispone que "Las publicaciones precedentemente dispuestas se realizarán a través de un diario judicial en formato digital que será accesible al público, en forma gratuita, por medio de la página de internet de la Corte Suprema de Justicia de la Nación, resguardando el derecho a la intimidad, a la dignidad y al honor de las personas, y en especial los derechos de los trabajadores y los derechos de los niños, niñas y adolescentes". En la misma dirección, la ley 27.275 ratifica expresamente que las sentencias judiciales constituyen "información pública" (arts. 1 y 7 "g") y establece el principio de "transparencia activa", en virtud del cual el Poder Judicial de la Nación, como sujeto obligado, "…[deberá] facilitar la búsqueda y el acceso a la información pública a través de su página oficial de la red informática, de una manera clara, estructurada y entendible para los interesados y procurando remover toda barrera que obstaculice o dificulte su reutilización por parte de terceros" (art. 32 "q").
A su turno, La CSJN reglamentó la publicidad de las decisiones judiciales en sus acordadas nº 15/13 y 24/13, en las que también obligó a los tribunales orales, y vinculó la publicidad de las sentencias con su carga en el sistema informático que utilizan los tribunales federales y nacionales para gestionar el trámite de los expedientes. Eso asegura, al menos como regla, que la totalidad de las sentencias estén disponibles públicamente de forma automática apenas son cargadas en el sistema.

[42] Ver Sección 2.1.

exploratorio de la jurisprudencia permite además responder algunas interrogantes generales acerca de las instituciones que administran el servicio de justicia, con el propósito de comprender mejor su funcionamiento, evaluarlo y, eventualmente, proponer reformas. Entre muchos otros datos judiciales relevantes -que no podré incluir aquí por razones de espacio-, algunos de los que es posible extraer de la muestra de fallos de la CFCP son los siguientes.

3.2.2.1 El sentido de las decisiones

Un primer dato relevante es la proporción de recursos admitidos y rechazados por el tribunal de casación. La CFCP utiliza distintos giros lingüísticos para adoptar una u otra decisión (y cada uno de ellos, habitualmente, entraña diferencias procesales más o menos relevantes). Así, la CFCP decide "hacer lugar al recurso" (típicamente de casación o queja), pero también resuelve en muchos casos "conceder el recurso" (cuando se refiere a la admisibilidad del recurso extraordinario federal, o de un recurso de casación denegado en la instancia anterior). Inversamente, la CFCP puede "rechazar" o "no hacer lugar" al recurso, pero también lo puede declarar "inadmisible", "desierto" (cuando la parte recurrente desiste de su presentación), "abstracto" (cuando la pretensión carece de interés al momento de la resolución) o "mal concedido" (cuando se declara que el recurso no reúne las exigencias de admisibilidad, no obstante haber superado ese filtro en la instancia anterior), etc.

Dado que lo que queremos es construir un sistema inteligente que nos permita identificar las condiciones que mejor pueden explicar una decisión jurisdiccional, las distintas variantes fueron simplificadas en categorías más generales de, digamos, resoluciones "favorables" y "no favorables" a la parte recurrente, que en el contexto de la CFCP se corresponden con la aceptación o el rechazo del correspondiente recurso.

La Figura 3.1 muestra la distribución de resoluciones favorables al recurrente (4.774), y decisiones que, por el motivo que fuere, resultan contrarias a su pretensión (33.120), sobre el total de sentencias. También se muestran "otras" decisiones: homologación o no de una prórroga de prisión preventiva,[43] aceptación de una excusación o recusación, etc. (18.146), que no serán tenidas en cuenta para el entrenamiento del algoritmo ya que no constituyen la adjudicación de un recurso. Tampoco se incluirán aquellas que resuelven más de una cuestión en litigio (decisiones "mixtas").

[43] Ejerciendo el control de oficio previsto en el artículo 1 de la ley 24.390.

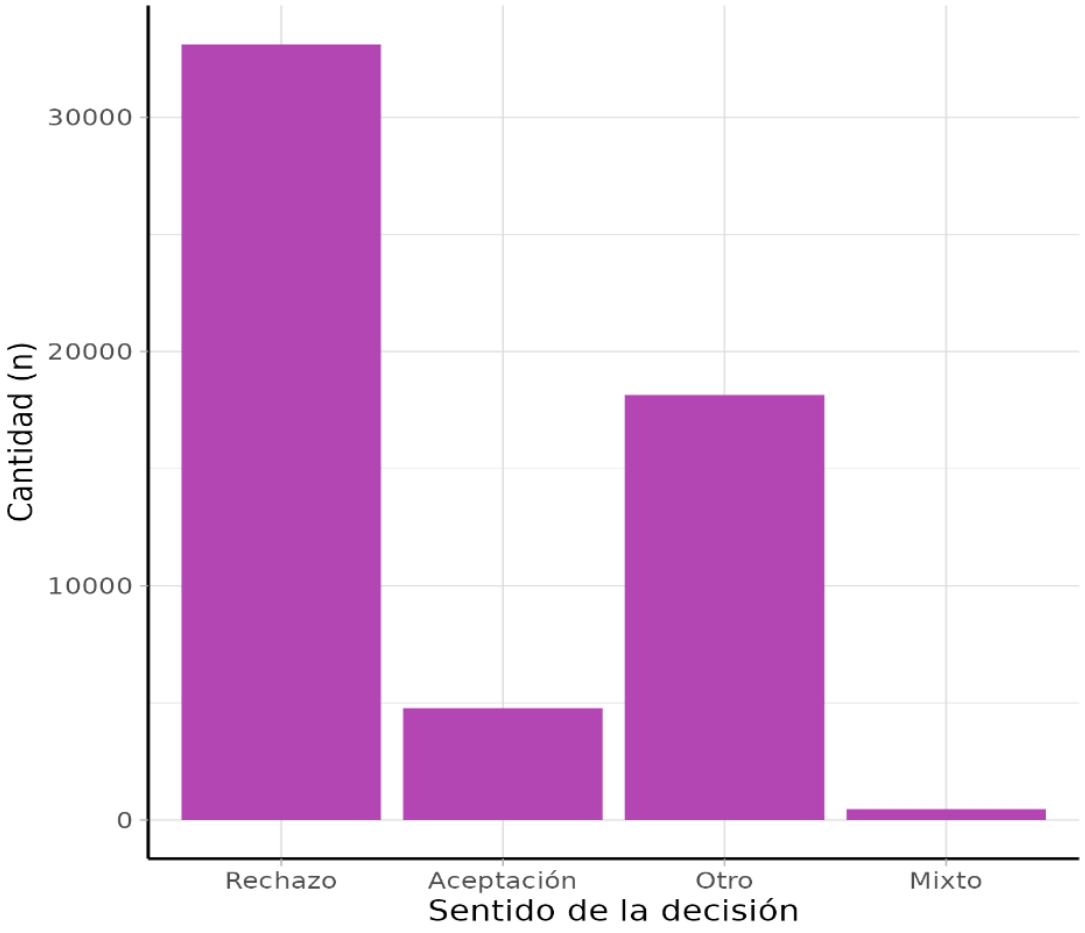

Figura 3.1: Distribución de sentencias según su sentido

3.2.2.2 La parte recurrente

De la misma manera, es posible conocer la distribución de decisiones de acuerdo con la parte recurrente (típicamente, defensa, fiscal o querellante), lo cual permite refinar la construcción del modelo, en tanto la intuición sugiere que las condiciones asociadas a la aceptación o al rechazo de un recurso probablemente varíe dependiendo de si se trata de uno presentado por la defensa o por la acusación. La Figura 3.2 muestra esa distribución, y permite ver también que una parte importante de los fallos de la muestra (n = 12.054) no identifican al recurrente, por lo cual también deben excluirse del procesamiento.

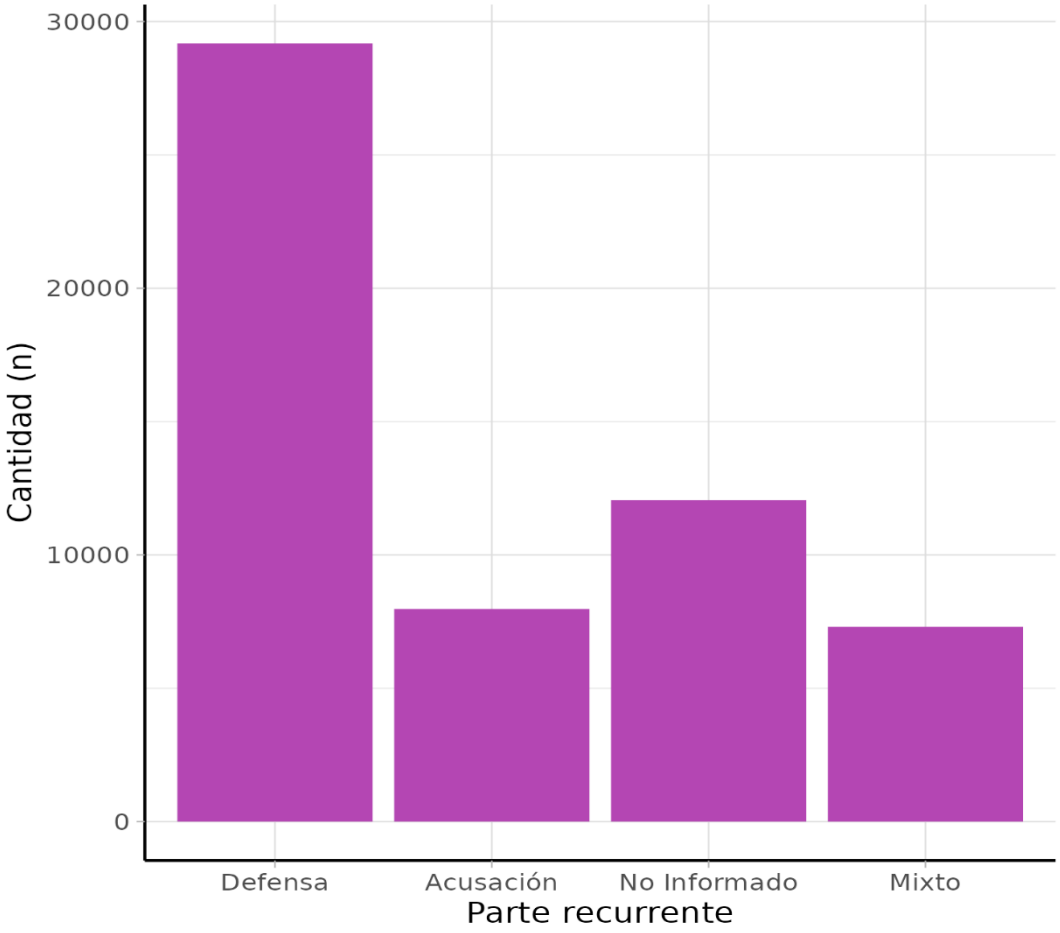

Figura 3.2: Distribución de resoluciones según la parte recurrente

Cruzar ambas distribuciones (Figura 3.3) permite ver, por lo demás, que los recursos de defensores y acusadores siguen patrones diferentes. Como explicaré en breve, el desbalance entre sentencias favorables y desfavorables al recurrente, si bien natural y esperable, tiene consecuencias sobre el proceso de aprendizaje automático —que deben ser tomadas en consideración apropiadamente—.[44]

[44] Ver Sección 3.4.2.

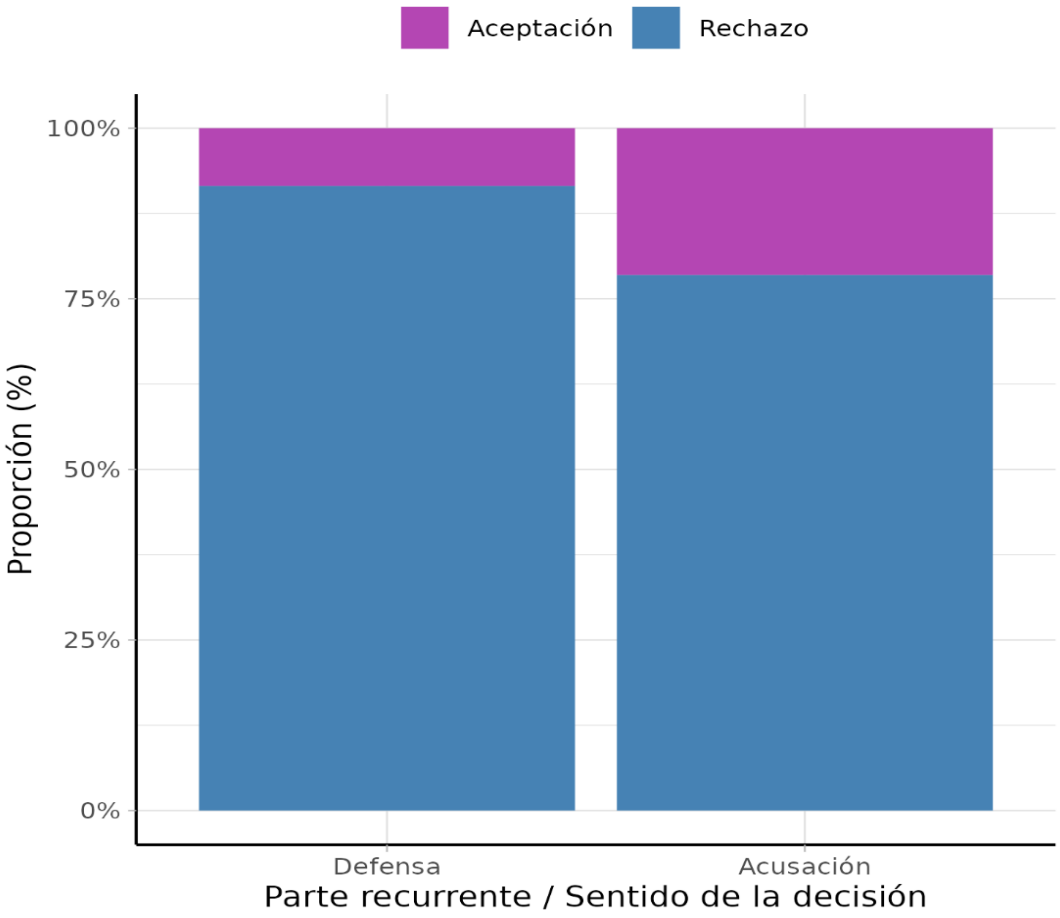

Figura 3.3: Patrones de decisión según la parte recurrente

En función de observaciones como éstas, ya es posible advertir que no todas las sentencias publicadas por la CFCP en el período abarcado por esta investigación (2014-2021) pueden ser tenidas en cuenta para entrenar y probar el algoritmo. Además de excluir las decisiones que no constituyen la resolución de un recurso, y aquellas que no identifican a la parte recurrente, es imprescindible evitar que las variables dependiente e independientes se mezclen (es decir, que el sentido del fallo aparezca entre las propiedades del caso). Afortunadamente, la estructura habitual de las sentencias de la CFCP contiene expresiones regulares,[45] como muestra la Figura 3.4, que permiten segmentar automáticamente el texto del fallo, de modo que las propiedades sólo sean extraídas de los fundamentos de la decisión, y el resultado del juicio, de la parte dispositiva. Técnicas similares permiten enmascarar cierto vocabulario de los fundamentos de los fallos que también podría adelantar el sentido de la decisión, sesgando los resultados.

[45] Ver Sección 2.2

> En mérito al resultado habido en la votación que antecede, el Tribunal, por mayoría, **RESUELVE: DECLARAR INADMISIBLE** el recurso de casación interpuesto por la defensa, **CON COSTAS** (Arts. 444, 465 *bis*, 530 y concordantes del C.P.P.N).

Figura 3.4: Parte dispositiva habitual en las sentencias de casación. Expresiones como "el Tribunal RESUELVE:" pueden utilizarse para delimitar los fundamentos y el resultado.

Por otra parte, para evitar que propiedades asociadas a decisiones de un signo contaminen las correlacionadas con la polaridad opuesta, se preservaron solamente los casos resueltos por unanimidad. También, se eliminaron fallos con menos de 1000 caracteres (aproximadamente 30 palabras), ya que una proporción importante son en realidad errores de carga en el sistema de gestión.

Finalmente, dado que las características de un caso que pueden inclinar el sentido de la decisión probablemente varíen dependiendo de cuál sea la parte recurrente, tiene sentido construir en realidad dos modelos: uno entrenado para identificar las características asociadas a que la CFCP acepte o no un recurso presentado por una parte acusadora (*i.e.*, el Ministerio Público Fiscal o la parte querellante); y un segundo modelo, ajustado a los casos en los que recurre la defensa.

La Tabla 3.1 resume las submuestras resultantes de aplicar estos criterios.

Tabla 3.1: Submuestras finales

Submuestra	*n*
Recurrente: Defensa	11.791
Recurrente: Acusación	3.806

3.3 Inferencia algorítmica

3.3.1 Predicción y retrodicción

Como adelanté, de manera similar a los comentarios doctrinarios tradicionales —pero a gran escala— el objetivo de esta investigación es identificar computacionalmente las propiedades de un caso que más peso tienen, en promedio, para inclinar una decisión de la CFCP en un sentido u otro (es decir, favorable o desfavorable al recurrente). En esta subsección, explico la metodología

que permite inferir esas propiedades relevantes a partir del procesamiento algorítmico de la muestra obtenida de los repositorios públicos de jurisprudencia nacional.

Una de las aplicaciones de los algoritmos clasificatorios de aprendizaje automático que más atención ha recibido es la construcción de modelos predictivos, capaces de anticipar eventos futuros a partir del reconocimiento de patrones inferidos de un conjunto de observaciones etiquetadas —denominado "conjunto de entrenamiento"—. Una sub-área de la literatura sobre análisis computacional del derecho —llamada a veces *Legal Judgment Prediction*[46] (LJP) o *Quantitative Legal Prediction*[47] (QLP)— también se ha ocupado de la cuestión, demostrando que es posible anticipar, por ejemplo, el sentido de las decisiones de la Corte Suprema de Estados Unidos o del Tribunal Europeo de Derecho Humanos.[48]

El pronóstico de decisiones futuras sobre la base de datos conocidos antes del dictado de las sentencias es, empero, sólo una de las aplicaciones posibles de los algoritmos predictivos.[49] Al igual que en la estadística clásica, el modelado implícito en la clasificación algorítmica permite también formular hipótesis sobre el *proceso* por el cual se llega a un determinado resultado. En otras palabras, inferir los aspectos o propiedades de un caso que están más asociados a una cierta decisión *pasada.*[50]

Esa es la capacidad de los algoritmos que me interesa explotar en este trabajo para estudiar el derecho argentino. En efecto, para entender mejor la manera en que los tribunales toman las decisiones que toman, debemos recorrer el camino inverso al de la predicción de resoluciones futuras: en lugar de interesarnos directamente por el rendimiento predictivo, posamos nuestra atención en los patrones que los algoritmos de *machine learning* identifican como informativos de la decisión judicial y los analizamos bajo el prisma de nuestros saberes sobre el derecho y su adjudicación. En esta reinterpretación del procesamiento algorítmico, la efectividad del algoritmo no es tanto una medida de su poder de pronóstico de decisiones futuras, sino de su capacidad para reconocer fiablemente los patrones que permiten modelar las relaciones entre las propiedades de un caso (variables independientes) y el sentido de la decisión adoptada (variable dependiente, o *etiqueta*).

Por esta razón, aunque los métodos que describiré se basan en los denominados algoritmos "predictivos", aquí utilizaré la expresión *retrodicción* (o *postdicción*)[51] para describir la

[46] Haoxi Zhong et al., «How Does NLP Benefit Legal System: A Summary of Legal Artificial Intelligence» [2020] arXiv preprint.

[47] Daniel Martin Katz, «Quantitative Legal Prediction — or How I Learned to Stop Worrying and Start Preparing for the Data-Driven Future of the Legal Services Industry» (2012) Vol. 62 Emory LJ.

[48] Relevé algunos de estos trabajos en la Sección 2.4.

[49] Masha Medvedeva, Martijn Wieling y Michel Vols, «Rethinking the Field of Automatic Prediction of Court Decisions» [2022] Artificial Intelligence and Law, pp. 2-11.

[50] Ibid, pp. 6-9.

[51] Agradezco a Marcelo Ferrante por esta sugerencia. Importo el neologismo de la literatura sobre la psicología del sesgo retrospectivo, en la que es utilizado para describir el mecanismo inferencial con el que los seres humanos estimamos retrospectivamente la probabilidad y previsibilidad de un evento cuyo resultado ya conocemos,

metodología computacional empleada. Retrodecir es contrastar el resultado de un evento pasado conocido (*ie*., el resultado de una sentencia ya publicada), con el resultado que un cierto mecanismo habría anticipado. La retrodicción es, así, una herramienta de validación: cuánto mejores sean las retrodicciones algorítmicas de las decisiones pasadas, más confianza podemos tener en que los rasgos de un caso, identificados por el modelo de inteligencia artificial, están efectivamente asociados al resultado correspondiente.

Desde el punto de vista de los algoritmos de *machine learning*, retrodecir el sentido de una decisión judicial a partir de ciertos rasgos del caso es una tarea de clasificación inductiva de textos; esto es, una tarea cuyo objetivo es "inferir una regla de clasificación a partir de una muestra de documentos etiquetados [como los fallos], de modo que sea posible clasificar nuevas observaciones con alta exactitud".[52] Como ya expliqué, la clasificación forma parte de las tareas de aprendizaje *supervisado*[53] y, en el caso particular de la retrodicción de resultados judiciales, es una tarea de clasificación *binaria*, puesto que el problema a resolver consiste en determinar, con una cierta probabilidad, a cuál de dos valores discretos, únicos y mutuamente excluyentes (clases) pertenece un fallo, de acuerdo con sus rasgos identificados: por ejemplo, "absolución" o "condena" del acusado; "aceptación" o "rechazo" del recurso, etc.[54]

3.3.2 De sentencias a datos informáticos

Los algoritmos de *machine learning* son algoritmos matemáticos y, como tales, no pueden procesar directamente el *texto* de los fallos. La aplicación de un algoritmo de aprendizaje supervisado a fallos judiciales, entonces, requiere, en primer lugar, que los representemos a través de propiedades o rasgos característicos cuantificables.[55] Un modo habitual y simple para hacerlo consiste en asignar un puntaje a cada palabra o construcción que compone el texto del fallo, dependiendo de la frecuencia con la que se presente en cada documento. En algún sentido, este proceso se parece bastante a la manera en que los seres humanos extraemos información al leer. Consideremos por ejemplo el fragmento de la Figura 3.5, tomado de uno de los fallos procesados.

---

habitualmente exagerando ambas medidas (Por ejemplo: (Baruch Fischhoff y Ruth Beyth, «I Knew It Would Happen: Remembered Probabilities of Once Future Things» (1975) Vol. 13 Organizational Behavior and Human Performance). También es utilizada en metaciencia para comparar el desempeño relativo de teorías en la explicación de un fenómeno, como la precesión del perihelio de la órbita de Mercurio, cuya retrodicción correcta por parte de la teoría de la relatividad funciona como prueba de su superioridad sobre el modelo newtoniano clásico. Para una discusión sobre el lugar del sesgo retrospectivo —entre otros— en la justificación de algunas de las instituciones centrales del derecho penal moderno, ver (Marcelo Ferrante, «Deterrence and Crime Results» (2007) Vol. 10 New Criminal Law Review).

[52] Thorsten Joachims, *Learning to Classify Text Using Support Vector Machines* (Springer US 2002) 24.

[53] James et al., pp. 129-170.

[54] Ashley, pp. 237.

[55] Lane, Hapke y Howard, pp. 79-83; Ashley, p. 256.

Así, luego de exponer sobre la procedencia del recurso y las cuestiones de admisibilidad, sostuvo que en la sentencia en crisis se han inobservado normas que el Código Procesal establece bajo pena de nulidad y se ha realizado una errónea aplicación de la ley sustantiva.

En esa dirección, planteó la nulidad de la sentencia por violación al principio de congruencia y afectación a la garantía de defensa en juicio, puesto que se le aplicó a su defendido agravantes por los cuales no había sido indagado ni procesado, en referencia a los incisos 1, 4 y penúltimo párrafo del art. 145 ter del Código Penal.

Figura 3.5: Fragmento de uno de los fallos procesados

A una jurista entrenada probablemente no le llevaría más que unos segundos advertir que, al menos a grandes rasgos, los párrafos se refieren a una nulidad procesal. En efecto, sus saberes sobre el derecho le permitirían "enmascarar" mentalmente la mayor parte del texto e identificar rápidamente que *nulidad* es el término relevante más frecuente en este pequeño fragmento. Algunas técnicas de NLP permiten explotar ese rasgo de la comunicación humana —esto es, que los principales temas sobre los que hablamos o escribimos son los que más frecuentemente aparecen en nuestro discurso— para inducir a una computadora a realizar observaciones similares.

Nótese que, para eso, no es suficiente calcular la frecuencia absoluta con la que los términos aparecen en el texto para identificar su importancia: si esa fuera la medida, los términos más sobresalientes serían "la", "en", "del", "que"; conectores ubicuos en nuestro lenguaje (a veces llamados *stopwords*) pero carentes de información útil para nuestros propósitos. Sin embargo, un ajuste sobre esta medición permite aproximarnos mucho más al objetivo: lo que deberíamos hacer para representar un fallo es quedarnos solamente con los términos más frecuentes en una decisión dada, pero que, al mismo tiempo, sean relativamente *infrecuentes* en la colección general. La intuición aquí es que palabras como "el", "y" o el tan judicial "máxime" seguramente figurarán en casi cualquier fallo; pero, términos como "nulidad", "violencia de género" o "crímenes contra la humanidad" sólo aparecerán en un puñado de ellos; y en particular, aparecerán con más frecuencia en los fallos en los que esos conceptos son más importantes. Esos términos son, entonces, los que mejor permiten representar a los textos que los contienen.

La técnica que modela los fallos como una función de la frecuencia de los términos y el inverso de su frecuencia en los documentos se conoce en inglés como *term frequency - inverse document frequency,* o *tf-idf*.[56] Para ser más precisos, en lugar de "términos" deberíamos hablar de *tokens*, una expresión que designa todo "bloque de información susceptible de ser contabilizado como elemento discreto"[57]: no sólo palabras individuales, sino también signos de puntuación y (más importante) construcciones sintácticas de mayor extensión, con sentido propio. Así, el bloque de información contenido en una *token* puede estar conformado por términos individuales como "homicidio" o "nulidad" (llamados *uni*gramas), pero también por secuencias de palabras que contienen sentidos diferentes a la suma de sus partes, como los *bi*gramas "falsedad ideológica" y "cosa juzgada" (que difieren en su sentido de "falsedad" e "ideológica", o "cosa" y "juzgada" por separado); o los *tri*gramas "violencia de género" y "riesgo no permitido", etc.[58]

En el modelo *tf-idf*, entonces, cada *token* es puntuada como el producto entre la frecuencia de un término *t* en un documento *d* (*term frequency*) y el logaritmo del inverso de la proporción de documentos *d* que contengan *t* a lo largo del corpus *D* (*document frequency*). Formalmente:

$$tfidf_{t,d,D} = tf_{t,d}.log\ log\ \left(\frac{|D|}{1 + |\{d \in D: t \in d\}|}\right)$$

Aplicando esta fórmula a cada *token*, el modelo tiende a excluir los términos genéricos e inespecíficos, y asigna los puntajes más altos a los *tokens* más característicos de un fallo. Por ejemplo, si tuviéramos un fallo de 1.000 *tokens* en el que la expresión "centro clandestino de detención" aparece 3 veces, la frecuencia del término sería tf=0,003. A su vez, si la expresión apareciera en 10 de los 1.000 fallos de la colección, el logaritmo del inverso de su frecuencia de documento sería idf=4, de modo que el puntaje del *token* en el documento sería tfidf=0,003*4=0,0012. Por otro lado, el *token* "de" puede aparecer 100 veces en un fallo de 1.000 tokens (tf=0,1) pero probablemente aparecerá en todos los fallos de la colección, de modo que su puntaje *tfidf* será 0.

Al aplicar este modelo a la colección de fallos analizados para esta investigación, el total de *tokens* (incluyendo bigramas y trigramas) asciende a 62.777.

3.3.3 Selección del algoritmo

[56] Lane, Hapke y Howard, pp. 70-71.

[57] Ibid, p. 33.

[58] Técnicamente, los *n*-gramas son conjuntos de *tokens* contiguos, que no necesariamente poseen un significado concreto (por ejemplo, "violencia de" o "interpuesto por" son bigramas, a pesar de que están incompletos semánticamente). Es posible no obstante "filtrar" los *n*-gramas con sentido propio —a veces llamados *collocations*— con técnicas estadísticas que, por ejemplo, miden la co-ocurrencia de los términos; esto es, la probabilidad de su coincidencia, dada su distribución conjunta y sus distribuciones individuales. Empleé esta técnica para la presente investigación.

Como señalé en la Sección 2.3, existen diversos algoritmos de aprendizaje automático capaces de realizar tareas de clasificación. La selección del algoritmo es una decisión no trivial, asociada a la clase de restricciones a las que se enfrente la analista computacional. Los algoritmos de *aprendizaje profundo*, por ejemplo, suelen ser los más poderosos en términos de precisión y sensibilidad; pero, también son computacionalmente los más costosos (en tiempo de procesamiento y exigencias de *hardware*) y, en general, requieren mayores volúmenes de datos para funcionar correctamente. Asimismo, su funcionamiento interno tiende a ser especialmente opaco —"funcionan esencialmente como cajas negras"[59]— y es muy difícil explicar cómo llegan a sus retrodicciones. Éste último rasgo es el que, al menos en principio, descalifica a este tipo de algoritmos para una investigación como ésta.

Con esta clase de restricciones en mente, para esta investigación utilicé una implementación del algoritmo denominado *Máquinas de Vector-Soporte* (*Support-Vector Machines*).[60] No es el algoritmo más preciso, en promedio, para tareas predictivas en general, pero diversos trabajos han demostrado su efectividad en tareas de clasificación de textos, particularmente en el ámbito legal,[61] al menos en idioma inglés.

Más importante aún, su implementación lineal para tareas de clasificación (*Linear Support-Vector Classifier*, o *LSVC*) entraña esencialmente el cómputo de una función lineal ordinaria, en la que cada propiedad del fallo se asume independiente a las demás y cada peso o coeficiente aprendido representa la contribución de esa propiedad a la conclusión.[62] Ese rasgo permite "mirar adentro" del algoritmo y entender *cómo* éste llega a los resultados que arroja.[63]

El funcionamiento de un clasificador lineal de vector-soporte es conceptualmente sencillo. Se trata de un algoritmo que aborda la clasificación como un problema geométrico, en el que cada una de las observaciones del conjunto de entrenamiento (i.e., cada fallo) son ubicadas en un espacio vectorial, que tiene tantas dimensiones como propiedades se hayan utilizado para describir cada observación. Luego, el algoritmo encuentra la ecuación más simple que pueda describir un *hiperplano* (básicamente, una recta en múltiples dimensiones) capaz de partir la

[59] Zhong et al. 6; y Brian M Barry, *How Judges Judge: Empirical Insights into Judicial Decision-Making* (Taylor & Francis 2020), p. 277.

[60] Corinna Cortes y Vladimir Vapnik, «Support-Vector Networks» (1995) Vol. 20 Machine Learning.

[61] Por ejemplo, Joachims, ; Clavance Lim, «An Evaluation of Machine Learning Approaches to Natural Language Processing for Legal Text Classification» (Tesis doctoral, Imperial College London 2019); Octavia-Maria Sulea et al., «Exploring the Use of Text Classification in the Legal Domain» [2017] Proceedings of 2nd Workshop on Automated Semantic Analysis of Information in Legal Texts; y Zhong et al., . Al menos en parte, esto se debe a que, a diferencia de otros algoritmos lineales, SVM es capaz de construir modelos generalizables aun en contextos de alta dimensionalidad (es decir, cuando el número de variables independientes supera al número de observaciones de la muestra), como lo exigen habitualmente las tareas de clasificación de textos. Ver James et al., pp. 371-372 y 377-378.

[62] En contrapartida, el costo de esta simplificación es que el modelo resultante no puede capturar relaciones no lineales entre las propiedades de un fallo (por ejemplo, combinaciones no aditivas entre ellas) y la decisión alcanzada por el tribunal.

[63] Esto no implica, sin embargo, que esa propiedad sea una característica exclusiva de los LSVC.

muestra, de forma tal que, por ejemplo, la mayor cantidad posible de fallos favorables al recurrente queden de un lado, y los desfavorables, del otro.

La dificultad aquí es que existen infinitos hiperplanos capaces de separar correctamente la muestra. El algoritmo da entonces un paso adicional: en su fase de optimización, el LSVC "aprende" cuál es el hiperplano que, además de clasificar correctamente los datos a los que fue expuesto inicialmente para su entrenamiento, es suficientemente robusto para poder generalizar el modelo a una muestra a la que no ha tenido acceso, minimizando las chances de clasificación errónea, como muestra la Figura 3.6. Para encontrar ese hiperplano óptimo, el algoritmo calcula la distancia perpendicular (margen) entre cada hiperplano posible y los vectores que más cerca están de ese hiperplano: son ellos los "vectores de soporte" que le dan nombre al algoritmo y de los que, en definitiva, depende la ubicación del hiperplano óptimo.[64] El modelo producido es aquel en el que se maximiza el margen entre el hiperplano y los vectores de soporte, pues es éste en el que las nuevas observaciones tienen menos posibilidades de quedar del lado incorrecto de la separación.

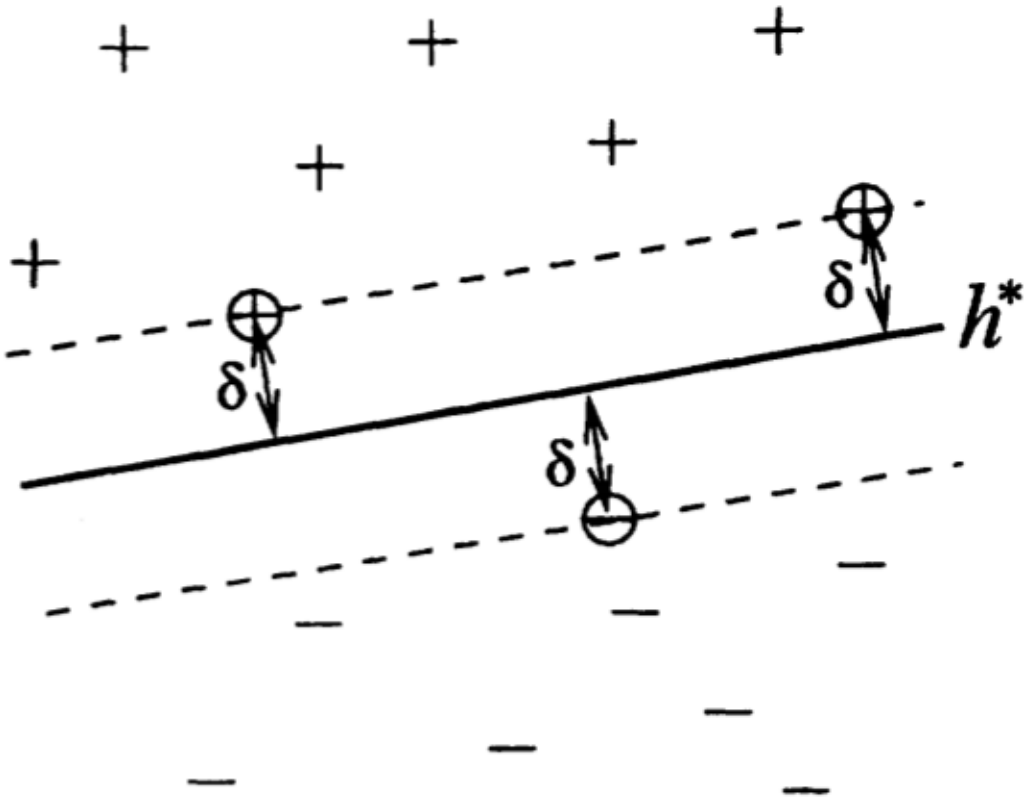

Figura 3.6: Hiperplano óptimo h^*, con margen máximo δ.

[64] James et al., pp. 371-372.

3.3.3 Selección del algoritmo

Con los textos representados vectorialmente y el *corpus* construido, el entrenamiento del clasificador de vector-soporte se realiza con unas pocas líneas de código. Sin embargo, hay que hacer una observación adicional: si bien la característica distintiva de los algoritmos de *machine learning* es su capacidad para regular sus parámetros de funcionamiento interno y aprender de los datos con los que son entrenados, ellos poseen también *hiperparámetros*, esto es, ajustes externos a la función de aprendizaje que inciden directamente en la manera en la que ella se manifiesta. Cada algoritmo posee sus propios hiperparámetros, que dependen de los principios matemáticos en los que se basan. En el caso de los clasificadores lineales mediante máquinas de vector-soporte, el hiperparámetro más importante es el término de regularización *C*, que determina qué tan tolerante es el algoritmo a observaciones que quedan del lado incorrecto del hiperplano de separación: si la tolerancia es baja, el algoritmo tenderá a hacer más esfuerzo para adaptarse a los datos de entrenamiento, con el propósito de minimizar la cantidad de errores. Inversamente, si la tolerancia al error es más alta, el algoritmo será más "rígido", y sólo reajustará su aprendizaje si hay un mayor número de errores de clasificación.

Uno podría pensar que no hay nada mejor que un algoritmo absolutamente intolerante al error y perfectamente ajustado a los datos de entrenamiento. Pero, lamentablemente, las cosas no son tan simples. En efecto, no tiene sentido evaluar el rendimiento de un algoritmo por su aptitud para ajustarse a los datos de entrenamiento; por el contrario, deberíamos juzgarlo por su capacidad para generalizar el modelo generado y retrodecir casos *desconocidos*, tomados de un conjunto de prueba con los que no ha sido entrenado –que representan los casos reales a los que el algoritmo debería enfrentarse–. Y el caso es que un modelo demasiado flexible y apegado a los datos de entrenamiento es habitualmente incapaz de generalizar bien: se dice de él que está "sobreadaptado" (*overfitted*)[65] a aspectos de la muestra de entrenamiento que pueden no estar presentes en la población general.

Esta tensión entre flexibilidad y capacidad de generalización es una derivación del *compromiso sesgo-varianza* (*bias-variance trade-off*),[66] que ha sido ampliamente estudiado por la estadística teórica. En general, se asume que el compromiso es inescapable —un dilema— y la clave está en encontrar el punto óptimo, por encima del cual aumentar la varianza disminuiría ineficientemente el sesgo estadístico, y viceversa.

Afortunadamente, existen mecanismos efectivos para encontrar automáticamente ese óptimo. En particular, se suele aprovechar el poder de cómputo de las máquinas modernas para establecer algo así como una "grilla de búsqueda" (*grid search* o incluso *random search*) en combinación con la técnica de validación cruzada,[67] que recorre diferentes combinaciones sensatas de valores

[65] James et al., p. 22.
[66] Ibid, pp. 33-37.
[67] Ibid, pp. 197-205.

para los hiperparámetros del algoritmo y devuelve aquél que haya tenido el mejor desempeño, de acuerdo con alguna medida como las que comento en la Sección 3.4.2.

3.4 Resultados y discusión

3.4.1 Los rasgos de un caso que más inclinan la decisión en un sentido u otro

Ahora que ya tenemos una intuición más formada acerca de cómo funcionan la representación vectorial de textos y los algoritmos de *machine learning*, veamos en definitiva cuál es el resultado del procesamiento.

Recordemos que el modelo *tf-idf* transforma cada fallo en un vector, cuyos elementos representan la preponderancia relativa de las unidades de información —*tokens*— que los componen y, de alguna manera, les dan a los textos su identidad. Como vimos en la Sección 3.3.2, el supuesto sobre el que descansa la adopción de este modelo es que —por ejemplo— cuando los argumentos contra la decisión recurrida refieren al modo en que las instancias inferiores resolvieron los planteos de nulidad absoluta, entre los *tokens* que representarán al texto con puntajes comparativamente superiores seguramente encontraríamos los *n*-gramas "nulidad", "nulidad absoluta", "art. 167 CPPN", "declaración de oficio", etc. Por otro lado, si los cuestionamientos se centraran en el juicio de imputación objetiva, los *tokens* con puntajes *tf-idf* más altos serían "riesgo no permitido", "imputación objetiva", "realización del riesgo", etc.[68]

En segundo lugar, sabemos que adoptar una implementación lineal del clasificador SVM implica que el modelo construido consiste en una ecuación lineal vulgar, cada uno de cuyos términos está compuesto por la variable que identifica un *token* y un coeficiente que expresa el peso que ese *token* tiene para describir el hiperplano de separación. En un sentido importante, entonces, analizar la estructura de los modelos lineales generados e identificar los *tokens* con coeficientes más altos es una manera de ver qué clase de consideraciones son las que tienden a inclinar la balanza de la decisión en un sentido u otro.

En el caso particular de los modelos que desarrollé para esta investigación, eso quiere decir que, cuanta mayor centralidad tengan esos *tokens* en el cuerpo de un fallo, mayor será la probabilidad de que el caso en cuestión sea resuelto en una cierta dirección.

[68] Hay que aclarar, empero, que si bien muchos *tokens* remiten a ideas o conceptos más o menos claros, algunas de ellas pueden no tener el sentido unívoco que podríamos estar tentados de adscribirles. Por ejemplo, uno podría pensar que el término "condena" tiene mayor presencia en casos en los que se está discutiendo la condena de una persona; pero lo cierto es que ella puede aparecer en una pluralidad de escenarios distintos dentro de un proceso penal. Afortunadamente, no tenemos por qué estar adivinando: podemos validar el sentido de un *token* identificando los casos en los que él aparece con mayor puntaje *tf-idf*, para determinar, por el contexto en el que aparece, si tiene el sentido que sugiere, o no.

Las Tablas 3.2 y 3.3 muestran algunas de los *tokens* que más fuertemente se asocian a una decisión u otra,[69] y que, puestas bajo el prisma de nuestros conocimientos sobre su significado técnico y sus implicancias para el derecho y el proceso penal, permiten inferir qué clase de aspectos de un caso están —en promedio— especialmente asociados a las decisiones del tribunal estudiado.

Tabla 3.2: Principales tokens asociadas a la aceptación/rechazo de un recurso interpuesto por la defensa

Aceptación	Rechazo
Prisión preventiva	Prisión domiciliaria
Estupefacientes	Evasión tributaria
Inconstitucionalidad	Secuestro extorsivo
Pandemia	Contrabando de importación
Extradición	Mercaderías
Nulidad / Nulidades	Medida disciplinaria
Violencia de género	Auto de procesamiento
Hábeas corpus	Estímulo educativo
Unificación de penas	Requisito etario (para la prisión domiciliaria)
Allanamiento	Habilitación de feria

Tabla 3.3: Principales tokens asociadas a la aceptación/rechazo de un recurso interpuesto por la acusación

Aceptación	Rechazo
Fuero Penal Económico	Plazo razonable

[69] Se trata por cierto de una selección, efectuada dentro del ~2% de las *tokens* más importantes para la clasificación. Junto con ellas, el algoritmo identifica también patrones carentes de relevancia jurídica, que es responsabilidad del analista filtrar. Por ejemplo, si el 2017 fue un año en el que los recursos de las defensas tendieron a ser aceptados, es probable que el algoritmo aprenda que el *token* "2017" tiene alguna importancia para la clasificación, aunque es obviamente irrelevante.

Tributario / Tributos / Ganancias	Prisión preventiva / Excarcelación
Querellante	Recurso de queja
Funcionarios públicos	Testigos / Testimonio
Pretenso querellante	Aduana / Aduanero
Cuestión federal	Prisión domiciliaria
Recusación	Libertad condicional
Ley penal más benigna	Inconstitucionalidad
Juez de ejecución	Delito de trata (de personas)
Niños	Caso de gravedad institucional

No puedo detenerme demasiado en el análisis detenido de estos hallazgos, y tampoco es el propósito de este primer trabajo exploratorio, en el que los expongo simplemente como prueba de concepto y metodología de investigación. Basta decir que, al menos en principio, los tokens identificados algorítmicamente como informativas de cada tipo de decisión parecen en general sensatas a los ojos de cualquier profesional con experiencia en litigio ante la CFCP. No sorprende, por ejemplo, que los recursos de la defensa tiendan a ser más efectivos (en promedio) cuando discuten el rechazo a la pretensión de que el acusado transite el proceso penal en libertad que, por caso, cuando objetan sanciones o rechazos a la incorporación del condenado a regímenes más bien excepcionales de cumplimiento de las penas. Correlativamente, el rechazo relativamente más frecuente de recursos de la acusación contra las excarcelaciones de los imputados, y una mayor tasa de éxito cuando se recurren decisiones de un juez de ejecución, respaldan la misma clase de observaciones.

En todo caso, lo que me interesa mostrar aquí es que, de manera similar a lo que hacemos cuando leemos fallos, mediante el procesamiento computacional de miles de sentencias podemos obtener una suerte de mapa de la topografía jurisprudencial de la CFCP, que nos permite ver el contorno de su proceso decisional a una escala antes inaccesible.

3.4.2 Fiabilidad de las inferencias

Ahora bien, hemos identificado algorítmicamente una serie de *tokens* y, a través de las ideas y los conceptos que ellas representan, inferimos algunas condiciones cuya presencia está especialmente asociada a una cierta clase de decisión. Sin embargo, ¿qué tan confiable es esta

lectura a distancia de la jurisprudencia de la CFCP? En otras palabras, ¿hasta qué punto los *tokens* identificados están en efecto correlacionadas con las decisiones que presuntamente anticipan?

Una manera de saberlo es, como anticipé, mediante el mecanismo de la retrodicción: básicamente, exponemos al modelo a un conjunto diferente de fallos que utilizamos durante la fase de entrenamiento y, en particular, un conjunto de fallos *sin etiquetar* (es decir, sin que el algoritmo sepa si se trata de casos de aceptación o rechazo del recurso. Lo llamamos *conjunto de validación*). De esa forma, podemos comparar las retrodicciones que el algoritmo hace a partir de los *tokens* identificados como relevantes, con el resultado verdadero del fallo, que de hecho conocemos (pero insisto, el algoritmo no). En un sentido intuitivo, entonces, la efectividad retrodictiva del modelo es una medida del grado en el que esos *tokens* permiten explicar el resultado de un fallo y, de esa manera, de cuánto influyen directa o indirectamente en el proceso decisional de la CFCP. En efecto, si las retrodicciones fracasaran, sabríamos que los *tokens* encontrados son más bien aleatorias y no están realmente asociadas a ningún tipo de decisión en particular.

Bien, el desempeño de un modelo de *machine learning* se evalúa habitualmente a través de la denominada *matriz de confusión*, que resume el sentido verdadero de los fallos del *conjunto de validación* (es decir, si se trata de fallos que efectivamente aceptaron o rechazaron un recurso), y el sentido pronosticado por el sistema.[70]

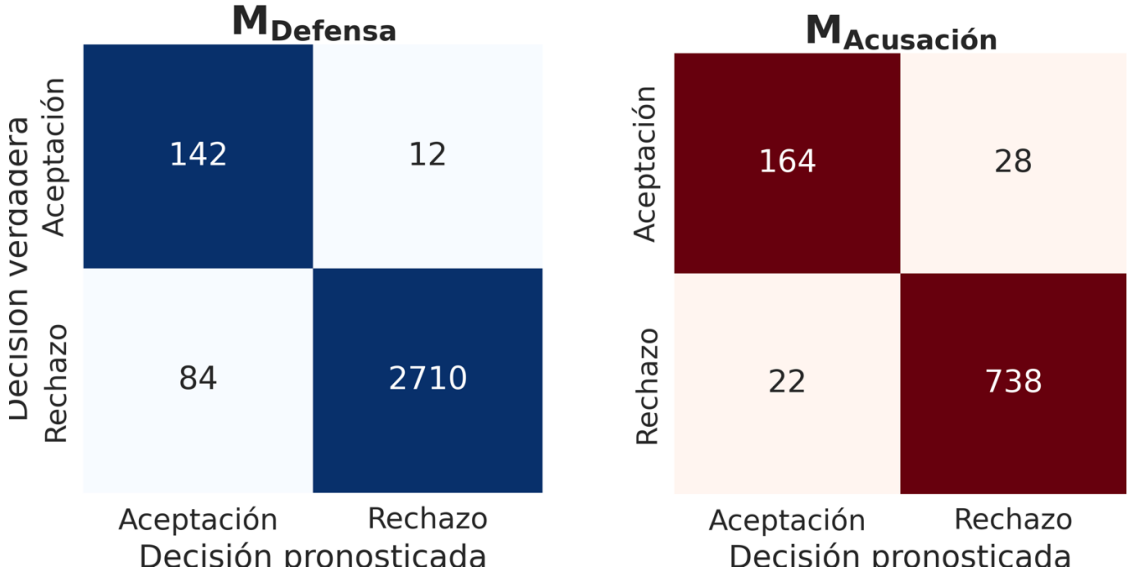

Figura 3.7: Matrices de confusión de los modelos generados

[70] Ver Lane, Hapke y Howard, pp. 455-456; Ashley, pp. 238-239.

De esta manera, se pueden ver los aciertos en la retrodicción, *i.e.*, cuando el sentido real y el pronosticado están alineados, y los errores, cuando ellos no coinciden.[71] En el primer cuadrante (superior-izquierdo), la matriz reporta la cantidad $V_{aceptación}$ de casos que el modelo identificó correctamente como aquellos en los que la CFCP falló a favor del recurrente (*i.e.*, admitiendo el recurso). En el segundo, registra el número $F_{rechazo}$ de fallos que el modelo retrodijo erróneamente como de rechazo a la pretensión (es decir, en los que la CFCP en realidad falló a favor del recurrente). En el tercer cuadrante se muestra la cantidad $F_{aceptación}$ de fallos erróneamente retrodichos como aceptando el recurso (y que en realidad fueron rechazados); y en el cuarto, el número $V_{rechazo}$ de fallos que el modelo clasificó correctamente como de rechazo. La Figura 3.7 muestra las matrices de confusión de $M_{Defensa}$ y $M_{Acusación}$.

Las relaciones entre aciertos y errores permiten calcular, a su turno, las principales métricas de evaluación de los modelos de predicción algorítmica: *Exactitud* (*Accuracy*), *Precisión* (*Precision*), *Sensibilidad* (*Recall*), y el promedio armónico de Precisión y Sensibilidad, llamado *Puntaje F1*.[72]

La primera y más simple medida de evaluación es la Exactitud, definida como la proporción de aciertos sobre el total de predicciones efectuadas. Formalmente:

$$E = \frac{V_{aceptación} + V_{rechazo}}{V_{aceptación} + V_{rechazo} + F_{aceptación} + F_{rechazo}}$$

La exactitud alcanzada por $M_{Defensa}$ y $M_{Acusación}$ fue de 0.97 y 0.95 , respectivamente, lo cual puede interpretarse como que, en promedio, las condiciones identificadas son directa o indirectamente relevantes para una decisión de la CFCP en un 0.96 de los casos. Como explico en la siguiente subsección, en el contexto de la literatura sobre algoritmos predictivos aplicados a decisiones judiciales, el rendimiento alcanzado es extraordinariamente elevado.

Sin embargo, si bien la exactitud es probablemente la medida de desempeño más popular, puede ser potencialmente engañosa. Imaginemos, por ejemplo, que queremos evaluar el desempeño de un modelo que permita pronosticar operaciones bancarias fraudulentas. Si 2 de cada 100 operaciones bancarias fueran fraudulentas (el número real es todavía mucho más bajo), un modelo que siempre clasificara las operaciones como legítimas acertaría en 98 de cada 100 casos; es decir, tendría una exactitud del 98%. El problema, claro, es que sería un modelo completamente *insensible* a las operaciones irregulares, que son precisamente aquellas que queremos identificar. Este matiz está presente en este estudio ya que, si bien la muestra de casos en los que la acusación recurre ante la CFCP está relativamente balanceada, la muestra compuesta por casos donde la defensa es recurrente está más bien desbalanceada (aunque todavía lejos de los casos extremos).

[71] Ver Ashley, pp. 113-114.
[72]

Para lidiar con esta dificultad, junto con el reporte de exactitud, suelen adoptarse también medidas de desempeño algo más sofisticadas, como los denominados puntajes de Precisión y Sensibilidad, que se aplican directamente sobre cada una de las clases que el algoritmo debe ser capaz de identificar correctamente.

La Precisión se calcula como la proporción de aciertos en cada clase, sobre el total de retrodicciones hechas en cada una, de modo que:

$$P_{aceptación} = \frac{V_{aceptación}}{V_{aceptación} + F_{aceptación}}$$

$$P_{rechazo} = \frac{V_{rechazo}}{V_{rechazo} + F_{rechazo}}$$

En efecto, la Precisión disminuye cuando aumentan los errores de retrodicción en la clase correspondiente, de modo que ella puede interpretarse como una medida de la calidad o fiabilidad del modelo. En otras palabras, un bajo puntaje de Precisión sugeriría que no se puede confiar en las predicciones del modelo porque una proporción importante de ellas serán, en realidad, "falsos positivos".

Por otro lado, la Sensibilidad se define como la cantidad de aciertos en cada clase, sobre el total de observaciones de esa clase en la muestra. Así:

$$S_{aceptación} = \frac{V_{aceptación}}{V_{aceptación} + F_{rechazo}}$$

$$S_{rechazo} = \frac{V_{rechazo}}{V_{rechazo} + F_{aceptación}}$$

De esa manera, la Sensibilidad disminuye cuando crecen los casos de observaciones erróneamente identificados como de la clase opuesta —esto es, observaciones de la clase relevante que el modelo debió identificar como tales, pero fueron pasadas por alto—. De allí que la Sensibilidad funcione como medida de la exhaustividad del modelo.

En general, se asume que existe una tensión inescapable (*trade-off*) entre la precisión y la sensibilidad de un modelo, en el sentido de que el aumento en un indicador suele ocurrir a expensas del otro, a menos que se produzca una mejora cualitativa en la configuración del modelo en sí (*i.e.*, consiguiendo más o mejores datos de entrenamiento, o utilizando un algoritmo de aprendizaje diferente, etc.). Por ejemplo, si forzamos un modelo a ser más exigente antes de emitir una retrodicción —digamos, aumentando el umbral de probabilidad por encima del cual asigna las etiquetas de cada clase—, la precisión subiría (habría menos falsos positivos), pero un mayor número de casos de la clase correspondiente serían identificados con la etiqueta errónea (más falsos negativos).

Eso no necesariamente es un problema: la preferencia por mayor precisión o sensibilidad es en gran medida situacional y depende de cuál de los tipos de error (falso positivo o falso negativo) consideremos más costoso en el caso concreto. Las propias reglas de evidencia del proceso penal ilustran esta observación: al imponer condiciones más estrictas para condenar que para absolver, adoptamos un mecanismo que tiende a ser más preciso, pero menos sensible que alternativas como, por ejemplo, la regla de preponderancia de la evidencia adoptada habitualmente en procesos no-penales. Ello así, ya que privilegiamos minimizar el número de condenados inocentes (falsos positivos) aun a costa de una mayor cantidad de absueltos culpables (falsos negativos). Inversamente, probablemente elegiríamos una prueba de COVID-19 que privilegie la sensibilidad por sobre la precisión, ya que el costo de un falso positivo implicaría que una persona sana deba hacer una cuarentena innecesaria, pero un falso negativo puede llevar a que una persona contagiada continúe con su vida normal como si no lo estuviera, incrementando el riesgo de transmisión.

En cualquier caso, la robustez general del modelo se suele expresar por medio del Puntaje F1: la media *armónica*[73] entre la precisión y la sensibilidad, que pretende expresar una preferencia neutral hacia ambas métricas. Formalmente:

$$F_1 = 2.\ \frac{Precisión\ .\ Sensibilidad}{Precisión + Sensibilidad}$$

La Tabla 3.4 muestra los puntajes alcanzados por cada modelo.

	Exactitud	Precisión	Sensibilidad	F1
$\boldsymbol{M_{Defensa}}$	0.97			
Aceptación		0.92	0.63	0.75
Rechazo		0.97	1.00	0.98
$\boldsymbol{M_{Acusación}}$	0.95			
Aceptación		0.85	0.88	0.87
Rechazo		0.97	0.96	0.97
Media	0.96	0.93	0.87	0.89

[73] A diferencia de la media aritmética, la media armónica es más sensible a los valores relativamente más pequeños, de modo que, por ejemplo, si Precisio´n=1 pero Sensibilidad=0, la media aritmética sería 0,5, pero la media armónica es 0, lo cual expresa mejor la (in)utilidad de un modelo predictivo completamente insensible.

3.4.3 Desempeño alcanzado y próximos pasos

Desde el punto de vista de lo que puede llegar a ofrecer un modelo de *machine learning*, el desempeño de $M_{Defensa}$ y $M_{Acusación}$ puede ser considerado muy elevado. En efecto, como muestra la Tabla 3.4, ellos no sólo pueden anticipar el sentido de las decisiones de la CFCP con muy alta exactitud (0.96), sino que sus puntajes de sensibilidad y precisión revelan también que se trata de modelos robustos, en el sentido de que sus predicciones son confiables (precisas) y relativamente exhaustivas. Es cierto que la sensibilidad de $M_{Defensa}$ podría ser más alta pero, como indiqué más arriba, la alta precisión del modelo asegura que los *tokens* identificados como relevantes sean confiables de modo que, en todo caso, lo que no podemos asegurar es que haya otras ideas también relevantes que permitan explicar el éxito de algunos recursos. A mi modo de ver, es un compromiso aceptable entre precisión y sensibilidad para la clase de tarea emprendida y, a su vez, funciona como línea de base sobre la cual ir perfeccionando la técnica.

De todos modos, si bien los resultados son auspiciosos y pueden ser entendidos como muestra de la estabilidad de las decisiones de la CFCP, hay que considerar también la posibilidad de que ellos estén sugiriendo que las resoluciones del tribunal tienden a parecerse en exceso y que eso, a su vez, puede ser un síntoma de que el tribunal hace colapsar en categorías demasiado simplistas casos que tienen matices relevantes. Esto puede ser objeto de una futura investigación.

Por lo demás, debo subrayar que en este primer trabajo mi interés ha estado puesto en presentar una metodología de análisis jurídico virtualmente inexplorada en nuestro país. Por razones de espacio, entonces, los hallazgos sustantivos del procesamiento computacional realizado no han sido abordados en profundidad. Sin embargo, quiero llamar la atención acerca de que el método empleado, más allá de posibilitar el descubrimiento de los rasgos de un caso que típicamente tienen más peso para inclinar la decisión de la CFCP en uno u otro sentido, puede servir específicamente para convertir preguntas jurídicas concretas en hipótesis objetivamente contrastables —por ejemplo, si una regla jurídica en particular se verifica empíricamente como una regularidad en el proceso decisional de un tribunal dado—abriendo así un amplio abanico de posibilidades para la investigación legal.

3.5 Conclusión

A lo largo de esta investigación he ofrecido una primera introducción al área de estudios legales que podemos denominar Análisis Computacional del Derecho y he argumentado que ella está llamada a ocupar un lugar relevante entre los múltiples abordajes interdisciplinarios del derecho que lo iluminan y enriquecen. A su vez, he efectuado un primer aporte exploratorio a esa literatura en idioma castellano, en el que he mostrado que es posible construir modelos predictivos de inteligencia artificial capaces de anticipar el sentido de las decisiones de un tribunal argentino, como la Cámara Federal de Casación Penal.

De esa manera, creo haber aportado evidencia en favor de la observación, más general, de que los resultados de los modelos pueden ayudar a comprender, evaluar y mejorar el proceso de deliberación judicial. Si eso es así, hay razones para creer que la adopción y proliferación de algoritmos como los que he descripto alterarán radicalmente la manera en la que estudiamos y ejercemos el derecho —si es que eso no ha ocurrido aún—.

Es mi deseo entonces que esta investigación funcione como una invitación a profundizar en el análisis computacional del derecho argentino —penal y no penal— y a repensar nuestra relación, como profesionales del derecho, con las tecnologías disruptivas que están cambiando casi todos los aspectos de la vida en sociedad.