



TRABALLO FIN DE GRAO
GRAO EN ENXEÑARÍA INFORMÁTICA
MENCIÓN EN COMPUTACIÓN



Estudio, análisis y desarrollo de técnicas de “interactive machine learning” y su comparación con el “machine learning” tradicional.

Estudiante: Alberto Pérez Sánchez

Dirección: Eduardo Mosqueira Rey

David Alonso Ríos

A Coruña, septiembre de 2022.

A mis padres y a Iván

Agradecimientos

Me gustaría dar las gracias a mis tutores, Eduardo Mosqueira y David Alonso, por proporcionarme el apoyo necesario y estar disponibles en todo momento para sacar el proyecto adelante.

También me gustaría dar las gracias a mi familia por apoyarme incondicionalmente y no dejar que me rinda. Especialmente a mi madre, que siempre ha querido que estudie y no deje de formarme.

Por último, gracias a mis amigos, tanto a los que he adquirido durante la carrera como a los que aún hoy me acompañan de antes. Os considero como mi familia y espero que sigamos compartiendo alegrías y penas juntos.

Resumen

El Aprendizaje Máquina es usado por gran parte de la comunidad científica como una caja negra: datos de entrada, ajuste de hiperparámetros, entrenamiento del modelo y obtención de resultados. Los modelos que se desarrollen bajo este escenario corren el riesgo de no escalar bien, volverse estáticos o ser difíciles de evaluar.

El Aprendizaje Máquina Interactivo son un conjunto de técnicas que intentan introducir activamente a los humanos en el bucle de aprendizaje de los modelos, funciona especialmente bien cuando el dominio es complejo, no hay suficientes datos o son muy costosos de anotar. El Aprendizaje Máquina Interactivo se divide dependiendo del papel que ocupa el humano dentro del ciclo de aprendizaje del modelo. En nuestro proyecto el Aprendizaje Activo es la aproximación desarrollada que consiste en que el modelo consulta al humano (oráculo) los casos que generan mayor incertidumbre en sus predicciones.

En este proyecto aplicamos las técnicas de Aprendizaje Máquina Interactivo a un problema real con datos de pacientes oncológicos con cáncer de páncreas, donde los datos y el tiempo de etiquetado de los médicos son limitados. Para aplicar la solución fue necesario utilizar técnicas de aumento de datos generando casos sintéticos de cáncer de páncreas mediante una Red Generativa Antagónica. Los resultados muestran como la incorporación del humano al bucle de entrenamiento provoca que el modelo converja de manera más rápida y eficiente con un menor coste humano que las aproximaciones clásicas.

Abstract

Machine Learning is used by most of the scientific community as a black box: input data, hyperparameters optimization, training the model and obtaining results. Models developed under this scenario run the risk of not scaling well, becoming static or difficult to evaluate.

Interactive Machine Learning are a set of techniques that attempt to actively introduce humans into the learning loop of models, works especially well when the domain is complex, not enough data or is too costly to annotate. Interactive Machine Learning is divided depending on the role of the human within the model learning loop. In our project, Active Learning is the approach developed which consists of the model consulting the human (oracle) for the cases that generate greater uncertainty in its predictions.

In this project we applied the Interactive Machine Learning techniques to a real problem with data from oncological patients with pancreatic cancer, where data and labeling time of physicians are limited. To apply the solution it was necessary to use data augmentation techniques generating synthetic cases of pancreatic cancer using an Generative Adversarial

Network. The results show how the incorporation of the human into the training loop causes the model to converge faster and more efficiently with a lower human cost than classical approaches.

Palabras clave:

- Aprendizaje Máquina
- Aprendizaje Activo
- Enseñanza Máquina
- Humano en el Bucle
- Aprendizaje Curricular
- Inteligencia Artificial Explicable
- Red Neuronal Artificial
- Red Generativa Antagónica

Keywords:

- Machine Learning
- Active Learning
- Machine Teaching
- Curriculum Learning
- Human in the Loop
- eXplainable Artificial Intelligence
- Artificial Neural Network
- Generativa Adversarial Network

Índice general

1	Introducción	1
1.1	Interactive Machine Learning	1
1.2	Tipos de HITL-ML	2
1.3	Objetivos	3
1.4	Estructura	4
2	Estado Del Arte	5
2.1	Active Learning (AL)	5
2.1.1	Definiciones de AL	5
2.1.2	Proceso, aplicaciones y herramientas AL	6
2.2	Interactive Machine Learning (IML)	6
2.2.1	Definiciones de IML	7
2.2.2	Aplicaciones IML	8
2.2.3	Herramientas IML	9
2.3	Machine Teaching (MT)	9
2.3.1	Definiciones de MT	10
2.3.2	Humanos como profesores	10
2.3.3	Máquinas como profesores	11
2.3.4	Herramientas MT	12
2.4	Curriculum Learning (CL)	12
2.4.1	Definition CL	12
2.4.2	Diseño de curriculum	13
2.5	Explainable AI (XAI)	14
2.5.1	Definiciones XAI	14
2.5.2	Técnicas	16

3	Aprendizaje Activo	18
3.1	Proceso aprendizaje activo	18
3.2	Estrategias de muestreo - Query strategies	19
3.2.1	Estrategia de anotación	21
3.2.2	Herramientas AL	22
4	Red Generativa Antagónica	23
4.1	Red Generativa Antagónica	23
4.1.1	Proceso de entrenamiento	24
4.1.2	Discriminador	24
4.1.3	Generador	25
4.1.4	Función de pérdida de las GAN	25
4.2	Conditional Tabular GAN (CTGAN)	27
5	Aplicación: Cáncer pancreático	29
5.1	Cáncer de páncreas	29
5.2	Tipos tumores pancreáticos	31
5.3	Obtención de los datos	32
6	Metodología	33
6.1	Metodología Kanban	33
6.2	Tablero Kanban	34
6.3	Planificación del proyecto	34
6.3.1	Planificación inicial	34
6.3.2	Etapa intermedia	37
6.3.3	Evaluación final	39
6.4	Estimación de costes	39
6.4.1	Costes estimados de los materiales	39
6.4.2	Costes estimados de los recursos humanos	39
7	Conjunto de datos	41
7.1	TCGA	41
7.1.1	TCGA-PAAD	41
7.1.2	Obtención de datos	42
8	Red Neuronal Artificial (RNA)	44
8.1	Red Neuronal Artificial	44
8.1.1	Neurona Artificial	44
8.1.2	Redes Neuronales Artificiales	45

8.1.3	Función de activación	46
8.1.4	Función de pérdida	46
8.1.5	Propagación del error	47
8.2	Optimización de hiperparámetros	47
8.3	Estructura	47
8.4	Validación	49
9	Aplicación Web	50
9.1	Herramientas utilizadas	50
9.1.1	Front-end	52
9.1.2	Back-end	52
9.1.3	Otras aplicaciones	52
9.2	Interacciones Humano-Máquina	53
9.3	Datos recogidos	54
10	Experimento y resultados	56
10.1	Conducción del experimento	56
10.2	Resultados	58
10.2.1	Etapas inicial	58
10.2.2	Iteración final	59
10.2.3	Comparativa	60
11	Conclusiones	61
11.1	Conclusiones	61
11.2	Futuras líneas de trabajo	62
	Bibliografía	65

Índice de figuras

3.1	Esquema AL	19
3.2	Muestreo de incertidumbre. Adaptada de [1]	20
3.3	Muestreo de diversidad. Adaptada de [1]	21
4.1	Esquema GAN	24
5.1	Gráfica estimación casos de cáncer. Fuente: Sociedad Española de Oncología Médica.	30
5.2	Fallecimientos por tumor España 2020. Fuente: Sociedad Española de Oncología Médica.	31
6.1	Depósito de planificación de tareas iniciales	35
6.2	Kanban tareas iniciales	36
6.3	Gráfica de tareas iniciales	36
6.4	Depósito de tareas planificación intermedia	37
6.5	Kanban de tareas intermedias	38
6.6	Gráfica de tareas intermedias	38
8.1	Diagrama de neurona artificial	45
8.2	Diagrama de una red neuronal artificial	45
8.3	Estructura red neuronal artificial	48
9.1	Arquitectura de las tecnologías utilizadas	51
9.2	Interfaz de usuario	54
10.1	Ciclo del experimento	57
10.2	Modelo entrenado con datos iniciales.	59
10.3	Modelo entrenado siguiendo una estrategia AL.	59
10.4	Comparación entre iteraciones en el entrenamiento de AL.	60

Introducción

CON la explosión del aprendizaje automático y el aprendizaje profundo hemos visto como el papel de los ingenieros ha pasado a un segundo plano. Recientemente ha surgido una corriente de metodologías bajo el nombre de Interactive Machine Learning que buscan resaltar el papel de los humanos y demostrar que no solamente ayudan al proceso de aprendizaje sino que lo mejoran de forma notoria.

1.1 Interactive Machine Learning

En los últimos años el aprendizaje automático se ha popularizado debido, en gran parte, a los avances hechos en el campo, la disponibilidad de grandes conjuntos de datos, la capacidad de computación cada vez más eficiente y una mejor capacidad para aprender de los datos. Cualquiera persona puede construir un modelo de aprendizaje automático desde su casa con pocos recursos, pero supone un proceso complejo que requiere un profundo conocimiento de las técnicas de aprendizaje automático.

A medida que el aprendizaje automático se ha ido imponiendo con el paso de los años, han ido surgiendo nuevas metodologías que pretendían superar las limitaciones de sus predecesoras y ampliar sus campos de aplicación a nuevos horizontes. Ahora que ya se han establecido las bases del aprendizaje automático, las aproximaciones que han sobrevivido y se han impuesto componen un grupo denominado aprendizaje máquina tradicional o aprendizaje pasivo.

Dentro del aprendizaje pasivo existen distintas variantes como el aprendizaje máquina supervisado, el aprendizaje máquina no supervisado o el aprendizaje por refuerzo. Los modelos de aprendizaje pasivo que son creados con alguna de estas aproximaciones tienden encasillar al humano en un papel secundario o menos relevante, en el que su aportación al aprendizaje del modelo se basa en alimentarlo con datos (etiquetados o sin etiquetar) y dar forma al modelo (decidir los hiper-parámetros, capas ocultas, etc.). Pero el papel del humano no es único

problema que tiene el aprendizaje máquina tradicional, otro inconveniente es que los modelos funcionan como una caja negra, y aunque se comprendan los principios matemáticos de su funcionamiento, el resultado obtenido no tiene una interpretación o explicación directa para los usuarios humanos.

Para explotar todas las posibilidades que los humanos pueden ofrecer a las máquinas los investigadores están desarrollando nuevas relaciones entre los humanos y los algoritmos de ML, centrándose principalmente en las diferentes estrategias sobre cómo incorporar a los humanos en el proceso de aprendizaje. Las distintas aproximaciones reciben el nombre de **Aprendizaje Máquina Interactivo - Interactive Machine Learning (IML)**, cada una tiene un grupo de características propias y se diferencian en quién está a cargo del proceso de aprendizaje.

Aunque podemos encontrar en la literatura al Interactive Machine Learning como el conjunto que engloba todas las categorías, con lo años el concepto ha evolucionado y ha mutado a otra terminología como **Human in the Loop - Machine Learning (HITL-ML)**; el término inicial se ha convertido en una aproximación en sí, pasando de un concepto amplio a uno más centrado en la interacción estrecha del humano con el modelo. Este cambio se debe a que algunos de los campos que veremos más adelante son extremadamente recientes y aún están buscando el consenso y la coherencia necesarios en sus nomenclaturas.

El objetivo de este trabajo es demostrar que la participación de expertos humanos en el proceso de aprendizaje mejora el aprendizaje del modelo consiguiendo que el modelo sea más preciso y que los humanos tengan una menor carga de trabajo, mejorando así la calidad de sus anotaciones y minimizando errores. También cabe resaltar la mejoras que el propio HITL-ML aporta a los humanos facilitando la interacción con el modelo, su facilidad de comprensión y la inclusión de humanos no expertos.

1.2 Tipos de HITL-ML

Los tipos de aprendizaje automático que existen dentro del HITL-ML son relativamente recientes y comparten la incorporación del humano en el proceso de aprendizaje directa o indirectamente. Esta incorporación puede hacerse a distintos niveles dependiendo de quién está al control del proceso de aprendizaje:

- **Aprendizaje Activo - Active Learning (AL)**, la máquina es quien se encarga del proceso de aprendizaje y considera al humano un “oráculo” que anote los casos sin etiquetar.
- **Aprendizaje Máquina Interactivo - Interactive Machine Learning (IML)**, en donde hay una interacción más cercana entre los humanos y los modelos de aprendizaje,

intercambiando información de forma interactiva, centrada, frecuente e incremental a diferencia del aprendizaje máquina tradicional.

- **Enseñanza Máquina - *Machine Teaching* (MT)**, los expertos humanos del dominio tienen el control del proceso de aprendizaje delimitando el conocimiento que pretenden transferir al modelo de aprendizaje automático.

La forma en que los humanos aprendemos ha inspirado diferentes diseños de algoritmos a lo largo del desarrollo del aprendizaje automático. Para enseñar tareas complejas, los profesores suelen tener que organizar los conceptos, teniendo en cuenta su complejidad. Esta organización da lugar a lo que se conoce como “*curriculum*”. El **Aprendizaje Curricular - *Curriculum Learning* (CL)**, usado por algunos algoritmos de HITL, está basado en la idea de que si introducimos al modelo los conceptos de forma estructurada conseguiremos acelerar y mejorar el aprendizaje del modelo.

Actualmente, la capacidad de explicar y transmitir conocimientos constituye un pilar en nuestra sociedad, sobre todo en ámbitos como el de la justicia, donde el razonamiento del juez supone una autoridad absoluta. Como el HITL involucra expertos para que colaboren en el aprendizaje, se puede aprovechar dicha colaboración para que los expertos den su justificación de los casos y esta justificación ser usada luego por el modelo para ofrecer explicaciones de sus propias decisiones. La **IA Explicable - *Explainable Artificial Intelligence* (XAI)** busca que los modelos de IA puedan explicar a los humanos los resultados del aprendizaje, justificando sus conclusiones y aportando su perspectiva del problema; no nos interesa solamente la precisión del modelo sino las razones detrás de la decisión que nos pueda proporcionar la máquina.

1.3 Objetivos

El principal objetivo de este proyecto es estudiar las distintas técnicas, dentro del aprendizaje máquina interactivo, que incorporan al humano al proceso de aprendizaje.

Con esto en mente, afrontamos el trabajo poniéndonos las siguientes metas:

- Estudio y análisis de las distintas aproximaciones del HITL, revisando sus ideas, técnicas, campos de aplicación y herramientas para su desarrollo.
- Aplicación de técnicas de aprendizaje activo (AL) sobre un problema del ámbito de la biomedicina.
- Creación de una página web bajo los principios del IML para facilitar y agilizar las sesiones de etiquetados de los expertos humanos.

- Revisión y aplicación de herramientas de generación de datos sintéticos que utilizaremos en nuestro proyecto.
- Análisis de los resultados obtenidos.

1.4 Estructura

El trabajo está estructurado de la siguiente manera:

- **Capítulo 1: Introducción.** Presentación del trabajo, su contexto y objetivos a conseguir.
- **Capítulo 2: Estado del arte.** Estudio profundo de las distintas aproximaciones que engloban el HITL.
- **Capítulo 3: Aprendizaje Activo.** Estudio a fondo de las técnicas de aprendizaje activo sobre la que se fundamenta el trabajo práctico.
- **Capítulo 4: Red Generativa Antagónica.** Estudio de funcionamiento de las Redes Generativas Antagónicas y su aplicación sobre datos tabulares.
- **Capítulo 5: Cáncer de páncreas.** Descripción del campo de aplicación sobre el que vamos a realizar el trabajo del proyecto.
- **Capítulo 6: Metodología.** Tipo de metodología utilizada durante el proyecto, planificación del proyecto y costes.
- **Capítulo 7: Conjunto de datos.** Descomposición de los datos procedentes del proyecto de nuestro conjunto de datos inicial (TCGA).
- **Capítulo 8: Red Neuronal Artificial.** Marco teórico de las redes neuronales artificiales, estructura de la red y método de evaluación utilizado en la obtención de los resultados .
- **Capítulo 9: Aplicación web.** Herramientas utilizadas en el desarrollo de la aplicación web y características de las mismas. Descripción de la aplicación web con los tipos de datos recolectados en ella.
- **Capítulo 10: Experimento y resultados.** Flujo de trabajo de la parte práctica del proyecto y estudio de los resultados
- **Capítulo 11: Conclusiones.** Conclusiones y análisis de posibles líneas de trabajo futuro.

Estado Del Arte

CUANDO describimos el *Human-in-the-Loop Machine Learning* (HITL-ML) en el capítulo 1 hablábamos de que la inclusión de los humanos en el proceso de aprendizaje podía hacerse en diferentes niveles dependiendo de quien estuviera en control del proceso de aprendizaje. Para entender cada aproximación se necesita, no solamente entender la ideas subyacentes, sino también saber las discrepancias en las ideas, su evolución y poder identificar ejemplos de su aplicación en la práctica. Una fuente que resume el estado del arte englobando el HITL y expandiendo a otros temáticas es [2].

2.1 Active Learning (AL)

Una de las técnicas más populares dentro del HITL es el *Active Learning* o Aprendizaje Activo. El AL es una aproximación de aprendizaje automático en la que el modelo (alumno) necesita un experto en el dominio (denominado profesor u oráculo) que etiquete los casos que generan un mayor grado de incertidumbre, por lo tanto, los casos que aportan más información al proceso de aprendizaje. Como resultado, el modelo mejora sus resultados con una menor cantidad de casos etiquetados en comparación al aprendizaje pasivo.

En esta técnica, el modelo está a cargo de los datos, y consulta una figura con un conocimiento extenso en el dominio para que anote la información sin etiquetar.

El AL puede considerarse una especie de aprendizaje semi-supervisado debido a que utiliza datos con y sin etiquetas. Los nuevos ejemplos se anotan en un proceso iterativo e incremental, en el que se utiliza una estrategia de petición para elegir un ejemplo que se va a consultar y que, una vez etiquetado por un oráculo, dará lugar a un incremento de la precisión del modelo.

2.1.1 Definiciones de AL

El aprendizaje activo, inspirado en la técnica educativa con mismo nombre escrita por Bonwell y Eison [3], está diseñado para permitir que los estudiantes se conviertan en socios

en el proceso de aprendizaje sin volverse demasiado dependientes de los maestros.

Una de las primeras aplicaciones de esta técnica al aprendizaje automático fue [4], en la que se utiliza el AL para que el alumno tome un papel activo en la adquisición de los nuevos conceptos.

El aprendizaje activo utiliza un proceso iterativo para aumentar los datos de entrenamiento, a diferencia del aprendizaje pasivo, que debe obtener datos de entrenamiento en una etapa temprana. Como mencionó Settles [5], hasta ahora se han considerado tres escenarios principales:

- *Membership Query Synthesis*: es cuando el alumno consulta al oráculo para saber si una instancia es miembro de una determinada clase de ejemplos.
- *Pool-Based Sampling*: Cuando las instancias se extraen de todo el conjunto de datos y se evalúan y se les asigna una determinada puntuación de confianza. A continuación, el modelo selecciona algunas de las instancias (basándose en una estrategia de muestreo) para ser etiquetadas por el oráculo.
- *Stream-Based Selective Sampling*: cuando cada instancia de datos sin etiquetar se examina de uno en uno, y dependiendo del algoritmo, son etiquetados por el aprendiz, o se considera que se consultan al oráculo.

La estrategia de proporcionar al alumno nuevos ejemplos también puede relacionarse con el aprendizaje curricular (CL). Los seres humanos aprenden mejor si los ejemplos utilizados para entrenarlos están ordenados y organizados de manera que se vuelvan gradualmente más complejos. Este proceso de creación de una secuencia de ejemplos que se proporciona al alumno en diferentes etapas del proceso de aprendizaje puede ayudar a mejorar el rendimiento del alumno.

2.1.2 Proceso, aplicaciones y herramientas AL

Debido a que el trabajo práctico se centra en la aplicación de aprendizaje activo, tanto la aplicaciones que existen hasta el momento como el proceso que sigue el aprendizaje activo y sus herramientas serán tratados en mayor profundidad en la sección 3

2.2 Interactive Machine Learning (IML)

En el siguiente escalón del HITL-ML se encuentra el Aprendizaje Máquina Interactivo en el que hay una interacción más estrecha entre los usuarios y los sistemas de aprendizaje, con personas que suministran información de forma interactiva de una manera más centrada, frecuente e incremental en comparación con el aprendizaje pasivo.

Estas interacciones más cercanas pueden aparecer al comienzo, con anotaciones en los datos que tenga dificultad para entender y anotar el modelo; en el medio, proporcionando información de refuerzo y conocimientos sobre el proceso de aprendizaje; o al final, validando, interpretando y corrigiendo los resultados del modelo.

2.2.1 Definiciones de IML

La definición de *Interactive Machine Learning* surge por primera vez del artículo de Malcolm Ware et al. [6]. En este primer artículo se describe el IML como un método de construcción de árboles de decisión de manera interactiva. Aunque esta idea está encasillada dentro de la solución de un problema concreto podemos empezar observar la idea común en la mayoría de las aproximaciones de aprendizaje máquina interactivo: permitir a los usuarios construir clasificadores.

La idea de incluir a los humanos en el bucle continúa en Porter et al. [7]. Este trabajo propone que los humanos y los ordenadores trabajen juntos en la misma tarea haciendo lo que cada uno de ellos hace mejor en cada momento.

Uno de los trabajos más citados de IML es el de Fails y Olsen [8], en el que el diseñador humano entrena, corrige y enseña de forma interactiva el modelo hasta alcanzar los resultados deseados. Aquí, los autores no sólo plantean la cuestión de introducir a los humanos en el bucle del aprendizaje automático, sino que también contrastan el concepto de IML con el llamado concepto de aprendizaje pasivo.

Con el tiempo han comenzado a aparecer nuevos tópicos en los que los humanos no expertos puedan interactuar con los sistemas IML; en donde el sistema se actualice de una manera iterativa e incremental; y que las interacciones se realicen a través de interfaces diseñadas de manera que favorezcan la interactividad entre el modelo y el usuario.

De hecho Ramos et al. [9] ilustran sobre los distintos roles que los humanos pueden ocupar dentro del IML como: expertos en ML, científicos de datos o expertos de dominio. Ramos busca reflejar cómo afecta el papel del experto al sistema.

Del mismo modo, Jiang et al. [10] abarcan la forma en la que se debería actualizar el modelo. Dejan claro que para ellos el proceso debería realizarse de manera iterativa. Amershi et al. [11] se centran en la diferencia entre el ML tradicional y el IML, en cómo en el IML el modelo se actualiza de una manera más rápida, centrada e incremental.

Por último, [12] expone la diferencia entre las interfaces de aprendizaje pasivo vs IML. Expone que el sistema clásico de ML debe tener una interfaz, pero es una interfaz pasiva; en IML la interfaz es responsable de la retroalimentación bidireccional entre los usuarios y el modelo, el diseño de la interfaz es crítico para el éxito del proceso de IML.

De todas estas definiciones podemos extraer las principales características que sustentan un sistema IML:

- **Humanos en el bucle del aprendizaje máquina.** Con tareas en las que son más eficientes que las máquinas.
- **Humanos con diferentes roles.** Pueden aparecer como expertos en un dominio, usuarios no expertos, científico de datos, etc.
- **Metodología incremental.** El modelo se actualiza de forma iterativa e incremental.
- **Importancia de la interfaz de usuario.** Influye en la forma de aprender y condiciona los resultados del aprendizaje.

2.2.2 Aplicaciones IML

Los campos donde mejores resultados han obtenido al aplicar técnicas del IML son en los datos sin estructurar y los datos complejos. Esto es debido a que en el campo de los datos estructurados el aprendizaje pasivo obtiene unos resultados suficientemente buenos, provocando que los resultados y las mejoras al aspecto humano del IML no sobresalgan tanto.

Por ello, veremos algunas de estas aplicaciones en diversos ámbitos, organizando las aplicaciones según el tipo de datos que utilizan: datos sin estructurar, que englobarían los ámbitos de imagen, datos secuenciales en el tiempo y texto; y datos complejos que representan una categoría en sí.

Imagen

Dentro del aprendizaje automático uno de los campos más sobresalientes en la práctica ha sido el de las imágenes. Una aplicación que ha tenido mucho éxito en este campo, con la aplicación del IML, es la segmentación interactiva de imágenes. La idea es facilitar el proceso de obtención del conocimiento utilizando expertos como usuarios, mediante el uso de una herramienta en la que resaltan las zonas de interés en la imagen. Uno de los trabajos que destaca la importancia de este tipo de colaboración fue el de Porter et al. [7] que señala la importancia de la segmentación interactiva de imágenes especialmente en la obtención de imágenes biomédicas.

Datos secuenciales en el tiempo.

En este apartado resaltan las aplicaciones relacionadas con el ámbito del vídeo. Destaca el trabajo de Kabra et al. [13] que diseñó un marco de trabajo que permite reproducir vídeos de animales, y donde los expertos pueden anotar fotogramas concretos que contienen comportamientos específicos de los animales.

Cabe destacar en esta categoría las aplicaciones destinadas al ámbito de la música, donde sobresalen las herramientas de Wekinator [14] y BeatBox [15].

Texto.

Un ejemplo típico que se cita a menudo es el trabajo de Savelka et al. [16] que aplica el IML a la evaluación de la relevancia en el análisis legal, desarrollando un marco de trabajo en el que un experto humano coopera con el algoritmo de clasificación de texto.

Datos complejos.

Definir qué son datos complejos supone un reto y puede llevar a discrepancias entre las propias definiciones de los autores del IML. Una de las más aceptadas y recientes es la definición de Tolls [17], que describe los datos complejos como “datos cuyo tipo, estructura y heterogeneidad dificultan su análisis” y señala que este tipo de datos puede abarcar tanto conjuntos de datos grandes como pequeños.

Hay dominios, como la medicina, en los que es habitual encontrar datos complejos para alimentar nuestros sistemas de aprendizaje automático. Holzinger y Jurisica [18] afirmaron que el problema central en la investigación sanitaria y biomédica es que los modelos de datos biomédicos se caracterizan por una complejidad significativa que provoca *sobrecarga de información*, que definen como “ahogado en datos, pero hambriento de conocimiento”. Otras aplicaciones desarrolladas en el ámbito de la salud merecedoras de ser citadas son [19] y [20].

2.2.3 Herramientas IML

Algunas herramientas desarrolladas en el marco del IML son: AnchorViz [21], una herramienta de visualización interactiva que facilita el descubrimiento de errores de predicción y conceptos no vistos anteriormente; AIDE [22], un marco de anotación de imágenes para estudios ecológicos que integra a los usuarios y los modelos de aprendizaje automático; y por último JAABA [23], una herramienta que permite reproducir un vídeo y etiquetar fotogramas que contengan comportamientos característicos de animales.

2.3 Machine Teaching (MT)

La *Enseñanza Máquina - Machine Teaching* (MT) es otra aproximación para transferir el conocimiento de los humanos a las máquinas. El rol del humano en el proceso de aprendizaje supone las diferencias en las distintas aproximaciones, en este caso particular la responsabilidad recae absolutamente en el humano.

El término de enseñanza máquina se utiliza principalmente en el campo del aprendizaje automático para describir la idea de un profesor que enseña un modelo de ML. Antiguamente el papel de profesor estaba reservado exclusivamente para los humanos, pero recientemente los algoritmos que simulan a los profesores se han vuelto más populares.

2.3.1 Definiciones de MT

La idea de los algoritmos de aprendizaje que son enseñados por humanos se planteó hace ya varias décadas. Un autor importante es Angluin [24], que teoriza sobre los expertos humanos que intentan comunicar su método a un sistema experto.

Uno de los primeros autores en utilizar el término enseñanza máquina específicamente como enfoque al aprendizaje automático es Diamant [25]. Diamant tiene una visión de enseñanza máquina tradicional donde el papel de profesor es ocupado por un humano.

En otro orden de ideas, Zhu [26] se basa en el trabajo pionero de Angluin y, al igual que Diamant, utiliza explícitamente el término enseñanza máquina, que describe de forma similar pero cambiando el papel del profesor por un ente no humano. En esta versión de la enseñanza máquina, el profesor conoce de antemano el modelo objetivo, así como el algoritmo del alumno, y debe "enseñar" el primero al segundo de alguna manera.

Existen dos ramas diferenciadas en la enseñanza máquina que se diferencian en si el profesor que "educa" al modelo es humano o no.

2.3.2 Humanos como profesores

Dentro de la rama del MT que defiende el papel del humano como profesor destaca Simard et al. [27], Simard plantea un MT en el que el humano mantiene el control sobre el proceso de aprendizaje y se hace gran hincapié en la interacción del profesor con los datos.

En el mismo orden de ideas, Simard expresa que con la gran demanda actual de soluciones de aprendizaje máquina es necesario aumentar significativamente el número de personas que pueden enseñar a las máquinas. Esto significa que la construcción de sistemas de ML debe estar al alcance de todas las personas y no solamente de los expertos y científicos de datos.

Existen dos tipos de MT según como le pasamos los datos a la máquina:

- **Enseñanza por lotes**, en el que el profesor da al alumno un conjunto de entrenamiento, el orden no es importante.
- **Enseñanza secuencial**, en el que el alumno aprende en una secuencia y el orden importa y debe ser optimizado por el profesor

En los modelos por lotes, el proceso tiene una mayor eficiencia, mientras que los modelos secuenciales son menos eficientes desde el punto de vista computacional, pero intentan obtener retroalimentación inmediata sobre los resultados. Zhu et al. [26] añaden una nueva dimensión que está relacionada no solo con la cantidad de casos presentados al modelo, sino también con su orden. Entonces, en este caso, la enseñanza por lotes los casos no estarían ordenados y en la enseñanza secuencial los casos estarían ordenados en un orden determinado

y se suministrarán uno por uno. Este último concepto estaría relacionado con la técnica de Aprendizaje Curricular en la que el orden de enseñanza influye a la máquina.

Aplicaciones

En esta aproximación de los humanos enseñando a las máquinas, el ámbito de la robótica ha supuesto un campo de aplicación notorio. Un ejemplo puede ser Sena and Howard [28], en el que utilizan enseñanza máquina para enseñar tareas a robots en lugar de programarlos.

Otro ámbito cotidiano para el MT es su uso en la enseñanza, donde el modelo, una vez entrenado exhaustivamente, será el encargado de enseñar los conocimientos a otros humanos. Podemos encontrarlo en [29] una aplicación de referencia.

2.3.3 Máquinas como profesores

La literatura ofrece varias implementaciones de profesores no humanos para el paradigma de la Enseñanza Automática. Una lista de ejemplos incluiría a Liu et al. [30] que hace una clasificación según la información que el profesor tiene sobre el alumno, un trabajo posterior, también de Liu et al. [31] que está inspirado en el Aprendizaje Activo, y Chen et al. [32] que está basado en la adaptabilidad. Un resumen de estas implementaciones sería el siguiente:

- **Profesor omnisciente**, que tiene acceso total a las características del alumno, como el espacio de características, la función de pérdida y el algoritmo de optimización.
- **Profesor subrogado**, que solo tiene acceso a la función de pérdida.
- **Imitación de profesor**, donde el profesor no tiene acceso a los resultados del modelo.
- **Profesor activo**, inspirado en la educación humana, los exámenes y el Aprendizaje Activo.
- **Profesor adaptativo**, que observa las hipótesis del modelo y se adapta en cada momento.

Aplicaciones

Los principales usos del MT cuando el papel de los profesores es interpretado por una máquina suelen estar enfocados a acciones maliciosas, que funcionan como una especie de proceso de ML inverso, en donde sabemos como funciona el modelo y buscamos seleccionar los datos específicos para obtenerlo. Existen numerosos ejemplos entre los que podemos destacar [33], que consiste en crear un conjunto de atacantes alumnos, los atacantes contaminan los datos de entrenamiento para que un algoritmo de aprendizaje específico produzca un modelo sesgado que nos beneficie.

Otras aplicaciones de las máquinas que enseñan a las máquinas son de carácter más metodológico. Mosqueira-Rey et al. [34] sugieren utilizar esta técnica para obtener los *Datos Mínimos Viables - Minimum Viable Data (MVD)*” para entrenar un modelo de aprendizaje. MVD es un término acuñado por van Allen [35] que se refiere a los datos mínimos necesarios para entrenar modelos de aprendizaje automático.

2.3.4 Herramientas MT

Algunas herramientas desarrolladas en el marco del MT son: Language Understanding (LUIS) and Platform for Interactive Concept Learning (PICL) ambas desarrolladas por Microsoft. LUIS es un servicio basado en ML para integrar capacidades de lenguaje natural en aplicaciones, bots y dispositivos; PICL es una plataforma para que personas sin conocimientos de ciencia de datos puedan aprovechar el poder del aprendizaje automático a través de la enseñanza máquina.

2.4 Curriculum Learning (CL)

Las tres aproximaciones anteriores (IML, AL, MT) componen el núcleo del *Human-in-the-loop*. Pero como en alguno de los casos los conceptos abarcan una definición muy extensa, aparecen nuevas aproximaciones o técnicas utilizadas en un entorno más específico pero que tienen cierto valor y relevancia dentro del HITL. Por lo tanto, vamos a aumentar el estado del arte con dos conceptos menos conocidos como son: *Curriculum Learning* y el *eXplainable Artificial Intelligence*.

Para enseñar tareas complejas, primero ordenamos los contenidos de manera que en el curso del aprendizaje la complejidad aumente gradualmente para extraer los conceptos de los ejemplos anteriores.

La idea fundamental es empezar con algo “pequeño”, aprender los aspectos más fáciles de la tarea, y luego aumentar gradualmente el nivel de dificultad

2.4.1 Definition CL

La idea de entrenar una máquina con un plan de estudios se remonta a Elman [36], en donde desarrolla la idea de aplicar un orden a los conceptos con los que se alimentan las redes neuronales. Prosiguiendo con la idea de Elman encontramos a Bengio et al. [37], que confirman que los algoritmos de aprendizaje automático pueden beneficiarse de una estrategia curricular y que bien elegida puede ayudar a obtener mejores modelos. Bengio propone el término Aprendizaje Curricular (CL) como la estrategia de entrenamiento que entrena un modelo de aprendizaje automático con un currículo.

Dentro de los partidarios y defensores de los beneficios que aporta el CL se encuentran Wang et al. [38]. Según ellos, el potencial de la introducción del currículum en el aprendizaje máquina depende de cómo se diseñe el currículum para aplicaciones y conjuntos de datos específicos. Sus ventajas podrían resumirse en la mejora del rendimiento del modelo y la aceleración del proceso de entrenamiento de hacerse de manera correcta.

Existen dos motivos para la aplicación del CL: para guiar al modelo, regularizando el entrenamiento hacia mejores regiones en el espacio de los parámetros; y para eliminar ruido, centrándose en las áreas de mayor confianza para mitigar la intromisión de datos ruidosos. Ante todo el primer grupo busca guiar el entrenamiento en tareas difíciles donde el entrenamiento convencional no funciona, destacan aplicaciones como [39]. Por otra parte, el grupo que pretende eliminar o mitigar el ruido abarca conjuntos de datos de entrenamiento ruidosos o heterogéneos y las estrategias de CL ayudan a hacer el entrenamiento más rápido, más robusto y más generalizable, siendo una aplicación popular [40].

2.4.2 Diseño de curriculum

El diseño general del curriculum consiste en dos componentes. La naturaleza del medidor de dificultad y del calendario de entrenamiento nos lleva a dos categorías diferentes de CL:

- Cuando tanto el medidor de dificultad como el calendario de entrenamiento son diseñados por el conocimiento humano, sin que intervengan algoritmos basados en datos, el método de CL se denomina CL predefinido.
- Si alguno (o ambos) de los dos componentes se aprende mediante modelos o algoritmos basados en datos, el método de CL se conoce como CL automático.

Medición de dificultad

La medición de la dificultad está relacionada con las características de los datos, especialmente con su complejidad, diversidad. La complejidad de los datos se define como datos que tienen más dimensiones y, por lo tanto, son más difíciles de captar por los modelos. Por su parte, la diversidad representa la variedad de un grupo de ejemplos, donde un valor mayor de la diversidad significa que los datos son más diversos y es más difícil para el aprendizaje del modelo.

Planificación de entrenamiento

Las diferencias radican en el método de ajuste del subconjunto de datos de entrenamiento y se diferencian entre discretos y continuos. En la **planificación discreta**, el subconjunto de datos de entrenamiento se ajusta después de cada número fijo de ciclos (>1). Por otra parte, la

planificación continua ajusta el subconjunto de datos de entrenamiento en cada ciclo de entrenamiento mediante la asignación de una proporción de los ejemplos más fáciles disponibles en cada ciclo.

CL automático

Los métodos de CL predefinido presentan numerosas limitaciones como: necesidad de un experto del dominio para diseñar un medidor de dificultad predefinido, dificultad para obtener la mejor configuración, no tiene en cuenta la información que pueda proporcionar el modelo durante el entrenamiento, etc.

Los métodos de CL automático se introdujeron para superar estas limitaciones y reducir la necesidad de profesores humanos en el diseño del currículum. Existen tres metodologías principales: Self-Paced Learning ,Transfer Teacher and Reinforcement Learning Teacher.

- **Aprendizaje Autónomo - Self-Paced Learning (SPL)**, métodos que permiten que el modelo (alumno) actúe como maestro y mida la dificultad de los ejemplos de entrenamiento según las pérdidas que tenga en ellos.
- **Profesor Suplente - Transfer Teacher**, métodos que dejan que un modelo profesor confiable actúe como profesor y mida la dificultad de los ejemplos de entrenamiento según el rendimiento del profesor en ellos.
- **Profesor de Refuerzo - Reinforcement Learning (RL) Teacher**, métodos que utilizan modelos de aprendizaje por refuerzo como el profesor para realice una selección de datos en función de la aportación del alumno.

2.5 Explainable AI (XAI)

Los humanos tenemos la capacidad de intervenir en el proceso de aprendizaje del aprendizaje automático como distintas figuras que pueden influenciar los resultados del sistema, pero cuando obtenemos estos resultados de un sistema refinado y complejo no tenemos un razonamiento en la solución final. En ámbitos donde las vidas humanas puedan verse afectadas necesitamos entender cómo se construyen dichas decisiones mediante métodos de IA [41].

2.5.1 Definiciones XAI

Como punto de partida veamos una de las primeras referencias y definiciones del término eXplainable AI (XAI) en : “XAI creará un conjunto de técnicas de aprendizaje automático que

permita a los usuarios humanos comprender, confiar apropiadamente y gestionar de manera efectiva la emergente generación de socios artificialmente inteligentes”.

La investigación en torno al XAI ha puesto de manifiesto diferentes objetivos para la obtención de un modelo explicable. En Barredo Arrieta et al. [42], los autores sintetizan y enumeran las definiciones de estos objetivos del XAI, para establecer un primer criterio de clasificación:

- **Accesibilidad:** Facilitar a los usuarios no expertos enfrentarse a algoritmos que parecen incomprensibles a primera vista.
- **Causalidad:** se considera la inferencia de relaciones causales a partir de datos observacionales. Un modelo explicable de ML podría validar los resultados proporcionados por las técnicas de inferencia de causalidad, o proporcionar una primera intuición de posibles relaciones causales dentro de los datos disponibles.
- **Confianza:** como generalización de un modelo robusto y que proporciona estabilidad a los usuarios.
- **Confianza:** confianza en que el modelo actuará como se esperaba ante un problema determinado.
- **Equidad:** considerado como un puente para evitar el uso injusto o poco ético de los resultados de los algoritmos.
- **Informativo:** los modelos necesitan una gran cantidad de información para poder relacionar la decisión del usuario con la solución dada por el modelo, y para evitar caer en errores de concepción.
- **Interactividad:** capacidad para ajustar e interactuar con los modelos por parte de los usuarios finales.
- **Privacidad:** la posibilidad de explicar las relaciones internas de un modelo entrenado por parte de terceros no autorizados también puede comprometer la privacidad del origen de los datos.
- **Transferibilidad:** un modelo que facilite la capacidad de un usuario para reutilizar este conocimiento en otro problema.

Sin embargo existen revisiones más recientes con distintos criterios, por ejemplo en Minh et al. [43] añade otro criterio que es la interpretabilidad, como la estimación del nivel en que los usuarios pueden comprender los resultados de los modelos de IA.

Como podemos ver, no existe una definición unificada ni objetivos específicos de la XAI, ya que suele asociarse a los esfuerzos e iniciativas para establecer una IA transparente y resolver las incógnitas de los modelos.

2.5.2 Técnicas

Las técnicas de XAI se dividen dependiendo de si son modelos interpretables por su propio diseño (modelos transparentes) o si pueden ser explicados mediante técnicas externas del XAI (post-hoc explicables)

Modelos transparentes

Los modelos transparentes pueden analizarse a nivel de todo el modelo, a nivel de los componentes individuales, y a nivel del algoritmo de entrenamiento, lo que da lugar a la siguiente categorización:

- **Simularidad** denota la capacidad de un modelo de ser simulado o pensado estrictamente por un humano.
- **Descomponibilidad** es la capacidad de explicar cada una de las partes de un modelo (entrada, parámetro y cálculo). Requiere que cada entrada sea fácilmente interpretable.
- **Transparencia algorítmica** trata de la capacidad del usuario para entender el proceso seguido por el modelo para producir cualquier resultado a partir de sus datos de entrada.

Post-hoc explicables

Los modelos de Post-hoc explicables son modelos que no son fácilmente interpretables por su diseño y tienen que mejorar su interpretabilidad recurriendo a técnicas como:

- **Explicaciones textuales:** explicaciones más simples mediante texto que también incluyen símbolos que representan el funcionamiento del modelo. Estos símbolos pueden describir la lógica del algoritmo mediante un mapeo semántico del modelo a los símbolos.
- **Técnica de explicaciones visuales:** proporcionan una visualización del comportamiento del modelo.
- **Explicaciones locales:** que segmentan el espacio de soluciones y dan explicaciones a los subespacios de soluciones menos complejas que son relevantes para todo el modelo.
- **Explicaciones con ejemplos:** ejemplos de datos representativos que se relacionan con el resultado generado por un determinado modelo.

- **Explicaciones por simplificación:** se reconstruye un nuevo sistema basado en el modelo entrenado para ser explicado.
- **Explicación de características relevantes:** que aclara el funcionamiento interno de un modelo calculando una puntuación de relevancia para sus variables.

Según el HITL, una mayor interacción del modelo con los expertos humanos del dominio puede servir para añadir capacidades de explicabilidad a los modelos.

Aprendizaje Activo

EN este capítulo enumeramos y exponemos brevemente las herramientas de desarrollo disponibles para el Aprendizaje Activo.

3.1 Proceso aprendizaje activo

El aprendizaje activo, conocido en inglés como *Active Learning* (AL), es una estrategia iterativa de aprendizaje automático que busca crear modelos que incorporen al humano en el aprendizaje de una manera eficiente. Un modelo de aprendizaje activo es un modelo entrenado bajo un conjunto de datos etiquetados que realiza predicciones sobre de datos sin etiquetar. El modelo reconoce los casos que generan un mayor grado de incertidumbre en sus predicciones y los envía al humano, al que considera un ser superior (oráculo) o experto, para que le proporcione las etiquetas correctas. Las anotaciones del experto se consideran absolutas y se añaden al conjunto de entrenamiento junto al resto de predicciones del modelo para comenzar un nuevo ciclo de aprendizaje activo sobre nuevos datos sin etiquetar.

Para completar un ciclo del aprendizaje activo y que la precisión del modelo se actualice es necesario seguir una serie de pasos:

1. Entrenar el modelo con los datos disponibles en el conjunto de entrenamiento.
2. Una vez entrenado el modelo, presentarle pequeños lotes de elementos sin etiquetar para obtener las predicciones que realiza el modelo.
3. Los elementos más dudosos son anotados por un experto u "oráculo" que decide la etiqueta.
4. Añadir los nuevos elementos al conjunto de entrenamiento y entrenar el modelo con el nuevo conjunto de datos.

5. Validar los resultados y en caso de no satisfacer los objetivos previamente marcados realizar otra iteración.

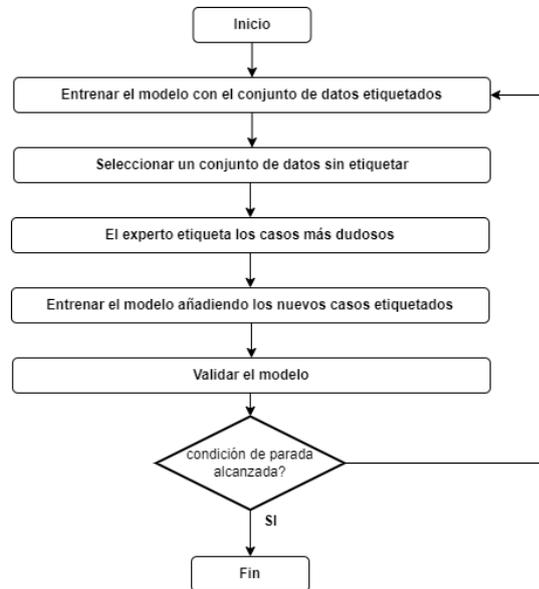


Figura 3.1: Esquema AL

3.2 Estrategias de muestreo - Query strategies

Las estrategias de muestreo son los métodos por los que el aprendizaje activo selecciona los ejemplos que serán etiquetados por el oráculo. Hay varias opciones y, dependiendo del problema, puede ser mejor utilizar una u otra.

Dentro del AL existen tres tipos de estrategias de muestreo propuestas por Munro [1]: aleatoria, de incertidumbre y de diversidad. Estas estrategias suponen las bases del HITL y las desarrollamos una a una a continuación:

Muestreo aleatorio

Es un método de muestreo que implementa la aleatoriedad de la selección de la muestra, es decir, cada muestra tiene la misma probabilidad que otras muestras de ser seleccionada para servir de representación de toda una población.

Muestreo de incertidumbre o explotación

Métricas utilizadas por la máquina para decidir las predicciones que generan una mayor incertidumbre, se dividen en cuatro:

- **Menos Confiable**, que selecciona el caso con la menor confianza en su etiqueta más probable.
- **Margen de Confianza**, que utiliza la menor diferencia entre las dos probabilidades más altas para cada etiqueta posible.
- **Ratio de Confianza**, que utiliza la proporción entre las dos predicciones más confiables.
- **Entropía**, que utiliza la diferencia entre todas las predicciones.

En la Figura 3.2 es posible observar un ciclo de muestreo de incertidumbre donde los casos más cercanos a la frontera de decisión son los que enviamos a revisar a los expertos.

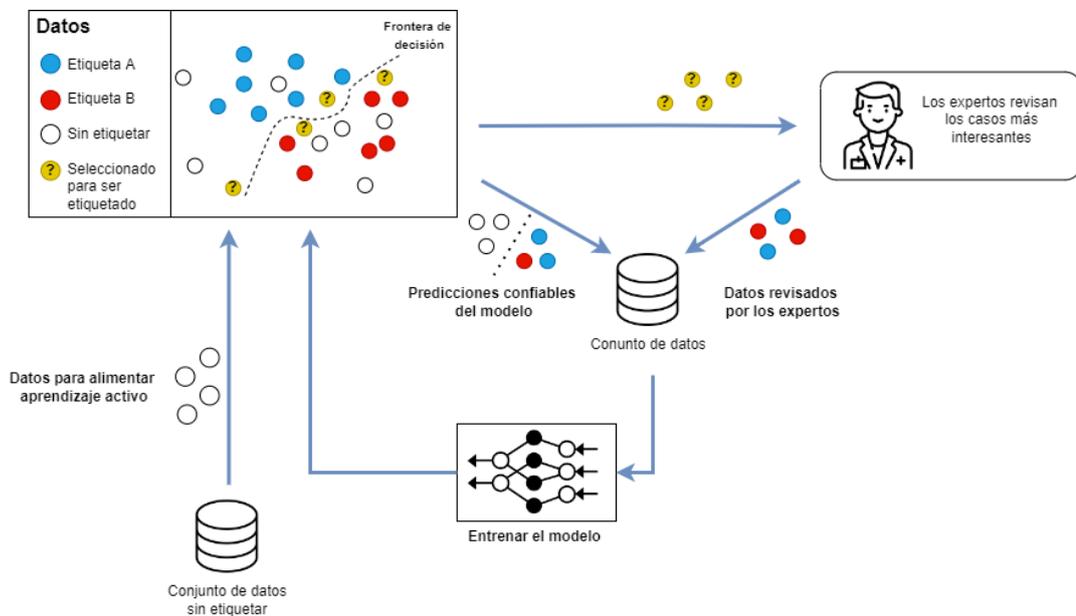


Figura 3.2: Muestreo de incertidumbre. Adaptada de [1]

Muestreo de diversidad o exploración

Selecciona elementos no etiquetados que son raros o no se han visto en los datos de entrenamiento para aumentar la imagen del espacio del problema. Aquí, encontramos:

- **Modelo basado en valores atípicos**, muestreo en busca de baja activación en las capas ocultas para encontrar elementos que confunden al modelo por falta de información.
- **Muestreo con Clustering**, que utiliza el aprendizaje no supervisado para agrupar los datos y encontrar valores atípicos que no forman parte de ninguna tendencia.

- **Muestreo Representativo**, muestreo de los elementos representativos del dominio objetivo para el modelo, en relación con los datos de entrenamiento actuales.
- **Diversidad mundo-real**, asegura que una diversidad de entidades del mundo real están presentes en nuestros datos de entrenamiento.

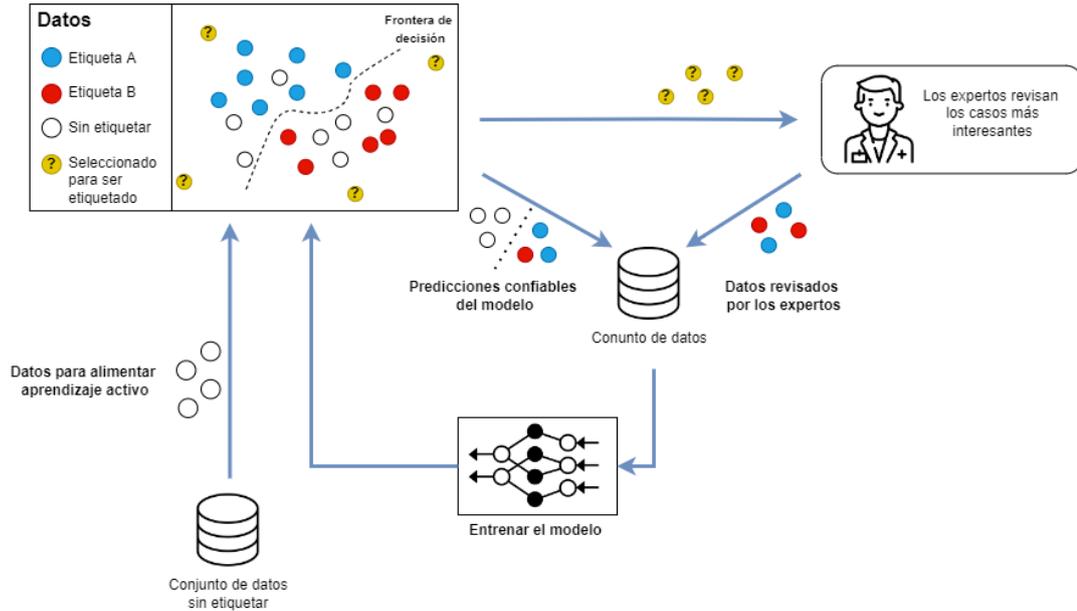


Figura 3.3: Muestreo de diversidad. Adaptada de [1]

En la Figura 3.3 es posible observar un ciclo de muestreo de diversidad donde los casos con características distintas a las del resto son enviados a los expertos para su revisión.

Pese a que las técnicas de exploración y explotación sean antagónicas y tengan finalidades distintas no implica que no se puedan utilizar de manera conjunta en un mismo proyecto. Mientras que la parte positiva de la explotación radica en encontrar zonas del mapa del modelo dónde las predicciones sean menos confiables, la exploración permite encontrar valores anómalos que confunden al modelo o nuevos grupos de clasificación que modifiquen el problema inicial. Por tanto, aplicar explotación y exploración de manera conjunta provoca, en algunos casos, que el modelo resuelva sus incertidumbres y que a mayores eliminé o cambié otros elementos que no aportan nada al entrenamiento.

3.2.1 Estrategia de anotación

Ahora que ya tenemos claro las estrategias principales del AL a la hora de seleccionar los elementos estudiaremos la forma de presentarlos. En la literatura del AL, se identifican princi-

palmente dos aproximaciones dónde la diferencia principal radica en el número de elementos que le presentamos al modelo:

- Por lotes. Donde se presenta un lote de casos que tienen que ser anotados antes de re-entrenar el modelo.
- Secuencial. El modelo se re-entrena después de cada nuevo elemento anotado.

3.2.2 Herramientas AL

La mayoría de las aplicaciones enfocadas al aprendizaje activo buscan crear entornos o marcos de trabajo que faciliten crear sistemas AL a un público menos experto. Algunos ejemplos de esto serían ALiPy y modAL.

ALiPy [44] es una herramienta de aprendizaje activo, desarrollada en Python, que permite a los usuarios evaluar, comparar y analizar cómodamente el rendimiento de los métodos de aprendizaje activo. Por su parte, modAL [45] es un marco de trabajo de aprendizaje activo que permite crear rápidamente flujos de trabajo de aprendizaje activo con casi total libertad; modAL permite sustituir fácilmente partes con soluciones construidas a medida, lo que le permite diseñar algoritmos nuevos con facilidad.

Red Generativa Antagónica

CUANDO describimos el aprendizaje activo, señalamos la necesidad de poseer un conjunto de datos sin anotar. Si no disponemos de datos sin etiquetar podemos acudir a la naturaleza del problema en su búsqueda, pero cuando esto no es posible otra opción viable es generar réplicas de manera sintética.

Desde hace mucho tiempo han ganado popularidad las técnicas de aumento de datos dentro del marco del aprendizaje automático, que aplicadas sobre un conjunto de datos generan nuevos elementos artificiales similares a los originales. De esta manera, podemos aliviar los problemas relacionados con la inconsistencia de la distribución y la insuficiente representatividad de los datos de entrenamiento. Una de las técnicas más conocidas y creciente en popularidad durante los últimos años son las Redes Generativas Antagónicas.

4.1 Red Generativa Antagónica

Las **Redes Generativas Antagónicas - *Generativa Adversarial Networks* (GANs)** son una clase de algoritmos dentro la categoría de aprendizaje profundo introducidos por primera vez por Ian Goodfellow et al. [46] que han generado grandes expectativas en la comunidad de investigación, ya que permiten la generación de nuevos datos sintéticos, indistinguibles de los reales.

Las redes generativas antagónicas son un tipo de red neural que se entrena para generar ejemplos que imiten el dominio de un conjunto de datos. Estas redes se componen de dos partes: una red generativa que crea las muestras de datos, que recibe el nombre de **Generador**, y una red discriminativa que trata de distinguir entre las muestras generadas y las muestras reales, con el nombre de **Discriminador**.

Estos modelos se denominan redes generativas antagónicas porque el generador y el discriminador se entrenan para competir entre sí: el generador intenta mejorar para engañar al discriminador con sus creaciones, mientras que el discriminador intenta mejorar para identi-

ficar las muestras generadas sintéticamente por el generador.

4.1.1 Proceso de entrenamiento

El proceso de entrenamiento de las redes GANs se realiza de manera iterativa y en cada iteración el Generador y el Discriminador compiten por superar a su rival. La arquitectura del proceso de entrenamiento de una GAN se muestra a continuación en la Figura 4.1.

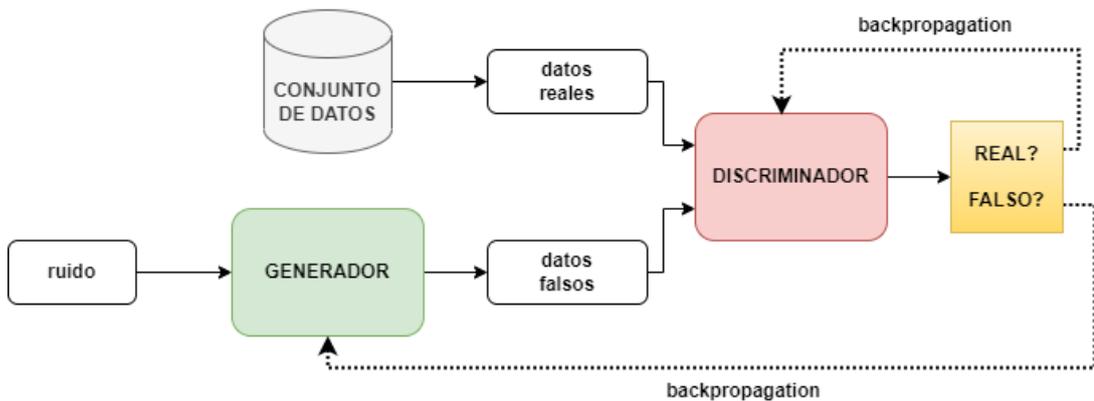


Figura 4.1: Esquema GAN

Dado que ya hemos establecido que las GANs constan principalmente de dos bloques, el generador y el discriminador, vamos a analizarlos con más detalle en las próximas secciones, explicando en profundidad cómo se entrenan y actualizan cada uno de los modelos.

4.1.2 Discriminador

El discriminador es un algoritmo de clasificación binaria que busca discernir los datos reales de nuestro conjunto de datos inicial (etiquetados como ejemplos positivos), de los datos falsos creados por el generador (etiquetados como negativos).

El discriminador consta de dos funciones de pérdida: una para el entrenamiento del generador y otra para el entrenamiento del discriminador. Durante el entrenamiento del discriminador, el generador no se entrena y sus pesos se mantienen congelados, también se ignora la función de pérdida del generador para utilizar la propia del discriminador.

El entrenamiento del discriminador sigue el proceso normal de una red neuronal artificial que se desglosa en:

1. El discriminador clasifica datos que recibe de entrada como reales o falsos.
2. La función de pérdida del discriminador penaliza al discriminador por clasificar erróneamente una instancia (falsa como verdadera o viceversa).

3. El discriminador actualiza sus pesos mediante *backpropagation* a través de la red discriminatoria.

4.1.3 Generador

El entrenamiento del generador depende de manera directa del discriminador, sin la clasificación del discriminador no podemos conocer cómo de buenas son las réplicas generadas. Como el generador no está directamente conectado con la salida y la función de pérdida necesitamos que el discriminador nos produzca y transmita los cambios a través de su estructura. El generador solamente es penalizado y sus pesos son actualizados cuando el discriminador identifica como falso uno de los datos sintéticos generados.

Durante el entrenamiento del generador el discriminador se mantiene congelado, de manera que solo se actualizan los valores del generador. Con esto conseguimos que el generador produzca datos sintéticos buscando confundir al mismo sistema. Si el discriminador se actualiza provocaría que el generador tendría que ajustarse a un modelo cambiante lo que conlleva un proceso casi imposible.

El proceso de entrenamiento del generador, aunque más complejo que el del discriminador, no supone un proceso particularmente complicado:

1. Presentamos al generador como valor de entrada un vector compuesto por valores aleatorios o ruido.
2. El generador produce una salida de tamaño idéntico a los datos que buscamos replicar.
3. El discriminador clasifica como las réplicas del generador como reales o falsas.
4. Calculamos la pérdida del generador a partir de la clasificación del discriminador
5. Propagación hacia atrás a través del discriminador y el generador para obtener los gradientes.
6. Empleamos los gradientes para cambiar sólo los pesos del generador.

4.1.4 Función de pérdida de las GAN

Como mencionamos anteriormente las GANs constan de dos funciones de pérdida, una para entrenar cada red neuronal. Estas dos funciones se pueden resumir en una única función, conocida en inglés como *min-max loss*, donde el generador y el discriminador tienen una competencia o juego *min-max*. Para entender mejor la función de pérdida min-max primero es necesario repasar algunos conceptos y desglosar la función entre las dos partes de la GAN.

Función de pérdida

La función de pérdida es un parámetro importante que determina el rendimiento de un modelo de aprendizaje automático para un conjunto de datos determinado. Calcula la diferencia entre el valor esperado y el valor predicho y lo representa como un único número real.

Pérdida entropía cruzada binaria - binary cross entropy loss

Tanto el generador como el discriminador utilizan la pérdida de entropía cruzada binaria para entrenar los modelos. La pérdida de entropía cruzada binaria puede describirse como:

$$P(y, \hat{y}) = -(y \cdot \log \hat{y} + (1 - y) \cdot \log (1 - \hat{y}))$$

Discriminador

El discriminador puede tener dos salidas distintas, verdadero o falso, que generan distintas pérdidas. Combinando las dos posibles pérdidas para el discriminador, una para una entrada real y la otra para una entrada falsa, se obtiene la función final que el discriminador debe minimizar.

En primer lugar, en los datos reales tendremos $y = 1$ e $\hat{y} = D(x)$. Si realizamos las sustituciones obtendremos:

$$P(1, D(x)) = -(1 \cdot \log(D(x)) + (1 - 1) \cdot \log(1 - D(x))) = -\log(D(x))$$

De la misma manera para los datos falsos tendremos $y = 0$ e $\hat{y} = D(G(z))$. Si realizamos las sustituciones obtendremos:

$$\begin{aligned} P(0, D(G(z))) &= -(0 \cdot \log(D(G(z))) + (1 - 0) \cdot \log(1 - D(G(z)))) = \\ &= -\log(1 - D(G(z))) \end{aligned}$$

Combinando ambas funciones de pérdida obtenemos:

$$P(\text{Discriminador}) = -[\log(D(x)) + \log(1 - D(G(z)))]$$

El objetivo del discriminador es minimizar la función de pérdida para disminuir el error de las predicciones; al ser la función negativa podemos eliminar el signo negativo y en vez de minimizar deberemos maximizar la ecuación.

$$P(\text{Discriminador}) = \max [\log(D(x)) + \log(1 - D(G(z)))]$$

Generador

La función de pérdida del generador proviene del discriminador. Por lo tanto, sólo es posible una entrada para el discriminador, que es la entrada negativa con datos sintéticos. Esto anula el primer término de la ecuación de pérdida del discriminador a 0, nos quedaría la siguiente ecuación:

$$P(\text{Generador}) = \min [\log(1 - D(G(z)))]$$

Min-max loss

La función más común en las GAN es la de *min-max loss* descrita por Ian Goodfellow durante la creación del propio concepto de las GANs. Esta función representa el enfrentamiento de las funciones de error de cada red neuronal peleando por conseguir engañar a su rival. La definición de la función es:

$$\min_G \max_D V(D, G) = E_x [\log(D(x))] + E_z [\log(1 - D(G(z)))]$$

- E_x es el valor esperado sobre todas las instancias de datos reales.
- $D(x)$ probabilidad de que el discriminador clasifique como verdadera una instancia de los datos reales.
- E_z es el valor esperado sobre todas las entradas aleatorias al generador
- $G(z)$ réplica generada a partir de ruido z .
- $D(G(z))$ probabilidad de que el discriminador clasifique como verdadera una instancia de los datos sintéticos.

El generador intenta minimizar esta función, mientras que el discriminador intenta maximizarla, convirtiendo en un juego min-max para el modelo.

4.2 Conditional Tabular GAN (CTGAN)

La aplicación de las GANs ha sobresalido en el campo de la generación de imágenes utilizando redes neuronales convolucionales. Uno de los problemas en el que se ha aplicado es para super-resolución de imágenes [47], es decir, generar una imagen de mayor resolución a

partir de una imagen de baja resolución. Esto se hace mediante la creación de una GAN que tome como entrada la imagen de baja resolución y genere como salida la imagen de mayor resolución.

Las aplicaciones sobre imágenes cargan la mayor parte de los esfuerzos de la comunidad científica debido a la facilidad de evaluación subjetiva de los resultados y mayor vistosidad, sin embargo, la aplicación sobre datos tabulares también supone una importante y apasionante rama de las GANs.

Uno de los ejemplos de aplicación de GANs sobre datos tabulares, y que utilizamos durante el transcurso del proyecto, son las *Conditional Tabular Generative Adversarial Networks* (CTGANs) [48]. CTGAN es una colección de distintos generadores de datos sintéticos basados en aprendizaje profundo para generar tablas de datos, que son capaces de aprender de datos reales y generar datos sintéticos muy fieles a los originales. CTGAN es una parte del proyecto *Synthetic Data Vault* (SDV) [49], el objetivo de SDV es desarrollar librerías software para la creación de datos sintéticos para la anonimización de datos o aumentar conjuntos de datos.

El CTGAN permite asignar condiciones o restricciones sobre la tabla generada sintéticamente, pudiendo así asignar valores de forma fija o calcular valores respecto a otras columnas. Estas reglas mejoran la precisión de los datos, prohibiendo posibles combinaciones de columnas que no puedan existir en nuestro conjunto de datos. Este es un escenario muy común al que nos enfrentamos cuando trabajamos con datos tabulares, encontrando columnas que tienen relaciones muy particulares entre ellas que son muy difíciles de modelar y que pueden confundir fácilmente a nuestro generador de datos.

Por ejemplo, una tabla que tenga las columnas de países y ciudades. En este caso, podría ser muy difícil saber a qué país pertenece cada ciudad, y al realizar un muestreo probabilístico, es probable que el modelo acabe generando combinaciones de país/ciudad no válidas.

Este tipo de relaciones especiales entre columnas se denominan restricciones, y SDV proporciona un mecanismo muy potente y flexible para tenerlas en cuenta y garantizar que los datos muestreados siempre las respeten.

Aplicación: Cáncer pancreático

EL ámbito de la biomedicina constituye unos de los campos más prósperos para la aplicación de técnicas de HITL. Siguiendo con esta línea de trabajo, nuestro esfuerzo se centrará en el cáncer de páncreas que supone uno de los mayores retos dentro del campo de la salud.

5.1 Cáncer de páncreas

En los seres humanos cuando una célula sufre alguna anomalía o envejece, se produce un proceso llamado apoptosis. La apoptosis es un proceso biológico natural en el cual las células del cuerpo se eliminan de manera controlada. El cáncer surge cuando esta función se produce de manera anómala, ocasionando que las células anormales se reproduzcan y las células viejas no se eliminen como sería natural. Cuando estas células se acumulan en los tejidos del páncreas se produce lo que se denomina como cáncer pancreático o cáncer de páncreas.

La función principal del páncreas consiste en producir insulina, que ayuda a que la glucosa de la sangre entre en las células y proporciona energía. El páncreas también produce otros jugos digestivos que ayudan a descomponer los alimentos en el estómago. Sin embargo, cuando surge el cáncer de páncreas estas funciones no se ven alteradas en lo absoluto, lo que provoca la proliferación del cáncer de manera silenciosa, y conduce en la mayoría de los casos a un resultado fatal.

Para observar las consecuencias del cáncer de páncreas solo es necesario echar un vistazo a la situación actual del cáncer en España, donde el cáncer pancreático ocupa el octavo lugar en diagnóstico y no obstante, debido a su alta letalidad, supone la tercera causa de muerte por esta enfermedad. La Sociedad Española de Oncología Médica (SEOM) es el organismo que proporciona los datos estimados sobre los distintos tipos de cáncer, haciendo a mayores una diferenciación por el género, en la gráfica de la Figura 5.1 aparece una visión general de los

casos de cáncer detectados en España durante el 2020.

La propia SEOM aclara que sus datos son estimaciones realizadas previamente a la pandemia de la COVID-19, por lo que es probable que estas estimaciones no reflejan exactamente la realidad, aunque sí se acercan de manera bastante precisa.

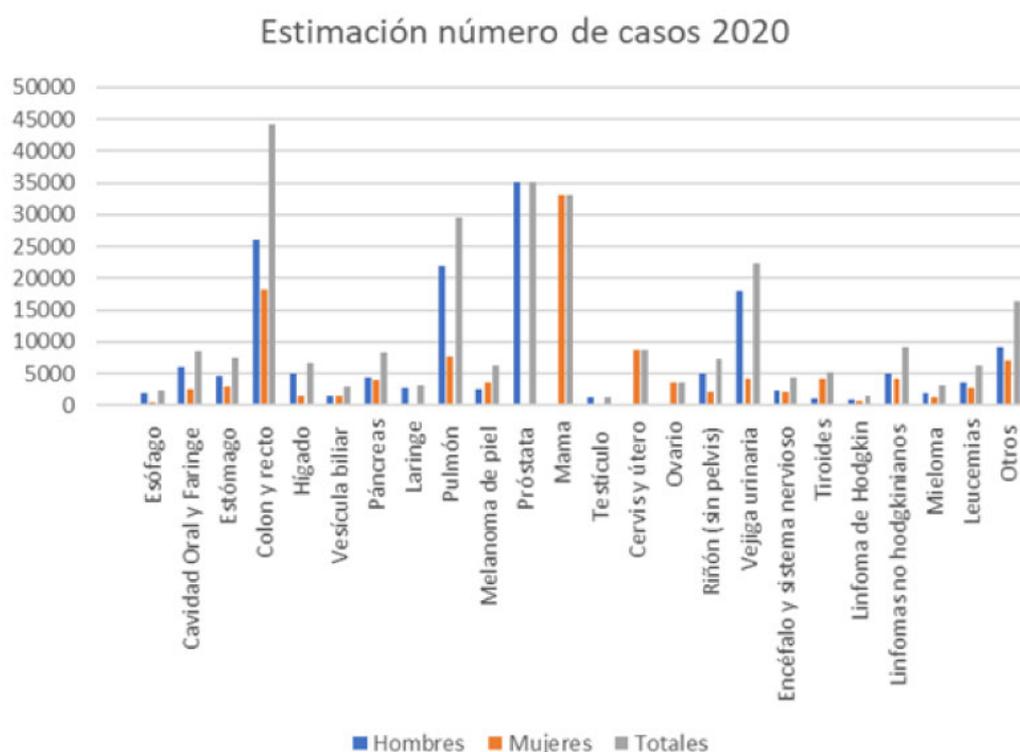


Figura 5.1: Gráfica estimación casos de cáncer. Fuente: Sociedad Española de Oncología Médica.

Aunque la situación del cáncer pancreático no resalta especialmente por su incidencia en el número de casos diagnosticados, siendo del orden de 5 veces inferior a los tipos de cáncer más comunes, en el apartado de la mortalidad se encuentra entre los tres primeros, solamente superado por el de pulmón y el de colon. Para que tomemos conciencia podemos comparar los datos entre el cáncer de colón, con 44.231 casos, y el cáncer de páncreas, con 8.338 casos. Con la diferencia tan drástica en casos, podríamos esperar que el número de fallecimientos fuera proporcional, pero la realidad es que el cáncer de colon solamente registra 3704 defunciones (Figura 5.2) más que el de páncreas.

	Total
Tumores	112.741
Tumor maligno de la tráquea, de los bronquios y del pulmón	21.918
Tumor maligno del colon	11.131
Tumor maligno del páncreas	7.427
Tumor maligno de la mama	6.651

Figura 5.2: Fallecimientos por tumor España 2020. Fuente: Sociedad Española de Oncología Médica.

El cáncer de páncreas es un caso aislado de cáncer con unas características propias que lo convierten en riesgo para cualquiera persona que lo padezca. La condición de los pacientes que sufren el cáncer de páncreas viene dada por distintas razones:

- **Dificultad de detección.** Los síntomas del cáncer de páncreas pueden ser muy sutiles y no aparecen, en la mayoría de los casos, hasta que la enfermedad está en una etapa avanzada.
- **Dificultad de prevención.** La posición biológica del páncreas imposibilita poder realizar una exploración física, lo que provoca no poder identificar bultos o masas. Las pruebas que realmente son capaces de identificar el cáncer de páncreas (ecografía abdominal, tomografía computarizada, resonancia magnética,...) son costosas y debido a que el porcentaje de afectados total no es elevado no es posible su realización al público general.
- **Extensión.** Debido a que su detección es muy tardía, en muchos casos los pacientes diagnosticados la enfermedad se ha expandido a los órganos vecinos, lo que se denomina como metástasis.

A causa de que generalmente el cáncer de páncreas se detecta en un estadio avanzado de la enfermedad el tiempo de reacción es considerado un factor clave. Los tratamientos del cáncer de páncreas deben indicarse y proporcionarse lo antes posible; debe considerarse que el tiempo de respuesta es una prioridad absoluta. Bajo esta premisa estará cementado el estudio y desarrollo de nuestro proyecto.

5.2 Tipos tumores pancreáticos

Analizar y conocer los tipos de cáncer de páncreas permite comprender y actuar en consecuencia dependiendo del caso. Existen distintos tipos de cáncer con distintos diagnósticos dentro del conjunto del cáncer de páncreas que pueden ser clasificados según J. M. Moreno Planas et al.[50] como:

- Neoplasias del páncreas exocrino
- Neoplasias de diferenciación combinada o incierta
- Tumores del páncreas endocrino
- Tumores del mesénquima pancreático
- Tumores secundarios del páncreas

Dentro de estos tipos, los tumores más frecuentes son los adenocarcinomas pancreáticos, que residen en la categoría de neoplasias del páncreas exocrino. El adenocarcinoma pancreático ductal supone más del 80% de los casos de casos diagnosticados y se desarrolla a partir de las células que recubren los pequeños tubos, denominados conductos, en el páncreas. Estos tubos transportan los jugos digestivos, que contienen enzimas, hacia el conducto pancreático principal. Este tipo de adenocarcinoma también es el responsable de la mayoría de las muertes teniendo una peor tasa de supervivencia que el resto en comparación.

Para la realización del proyecto utilizamos como base para el diagnóstico el libro de *Pancreatic Adenocarcinoma - NCCN Clinical Practice Guidelines in Oncology*, documento aceptado por toda la comunidad oncológica que proporciona las guías para que todos los pacientes reciban los servicios de prevención, diagnóstico, tratamiento y apoyo que más probablemente conduzcan a resultados óptimos.

5.3 Obtención de los datos

Debido a las leyes que garantizan la protección de datos, conseguir información sobre la que trabajar en el ámbito de la medicina puede ser considerado una auténtica odisea. El caso del cáncer de páncreas no es una excepción, ya que la petición de información se vuelve incluso más complicada a razón de que la persona diagnosticada con la enfermedad gran parte de las veces ya ha fallecido.

Los datos que se obtienen a partir de pacientes con cáncer de páncreas son de gran valor para los científicos, pues permiten el desarrollo de nuevas aplicaciones y soluciones médicas. Por lo tanto, si se quiere obtener información sobre cáncer de páncreas, la forma más fácil es acudir a fuentes de información de confianza, como los centros de investigación especializados en esta enfermedad. En este caso, para la aplicación hemos recurrido a uno de los proyectos más grandes y ambiciosos como es el *The Cancer Genome Atlas (TCGA)* [51]. Tanto los datos como las características del proyecto TCGA serán estudiadas en profundidad en el Capítulo 7.

Metodología

Los proyectos de investigación necesitan de metodologías flexibles que permitan hacer frente a las dificultades que surgen a medida que avanza el proyecto. La metodología Kanban se basa en la flexibilidad, simplicidad y la visualización de las tareas y su proceso; una serie de principios idóneos para los proyectos de investigación.

6.1 Metodología Kanban

Kanban es una metodología originada en Japón en las plantas de producción de Toyota como una forma de optimización de la productividad. La palabra Kanban se traduce como “tarjeta de señal”, ya que esta metodología se basa en el uso de tarjetas visuales para llevar a cabo la producción.

Cada una de las tareas de un proyecto Kanban se representa como una tarjeta o nota en un tablero. Esto permite a cada miembro del equipo elegir en qué tarea trabajar, y actualizar el progreso de esa tarea de forma totalmente transparente para el resto del equipo de desarrollo.

La metodología de Kanban otorga un gran valor a la mejora continua y a la optimización del trabajo y se caracteriza fundamentalmente por:

- **Visualizar el flujo de trabajo.** Visualizar las tareas en una tabla ayuda a todos los miembros del proyecto a estar atentos al trabajo que se está realizando y el avance del mismo.
- **Limitar el trabajo en curso.** Balancear el trabajo, limitando las tareas en paralelo para evitar posibles desajustes.
- **Gestionar el flujo de trabajo.** Supervisión, medición y análisis del flujo de trabajo en todas las fases para gestionar eficazmente los cambios y desviaciones del plan.
- **Políticas explícitas.** Es necesario definir reglas y directrices que gestionen el trabajo y que determinen, por ejemplo, cuándo hay que actualizar una tarea y cómo.

Dentro del proyecto de investigación hay diferentes tipos de tareas, bibliográficas, de investigación, análisis, diseño, programación, etc. El carácter flexible y dinámico de la metodología Kanban permite integrar estos aspectos sin tener impacto en el proyecto de investigación.

Para la realización del proyecto optamos por utilizar la herramienta de Planner desarrollada por Microsoft. Planner es una herramienta en línea, basada en Kanban, que proporciona una forma fácil de organizar proyectos mediante distintas vistas optimizadas donde podemos observar las tareas del proyecto, fechas, tareas en curso o gráficas del proyecto.

6.2 Tablero Kanban

El tablero Kanban sirve para que el equipo de trabajo visualice el estado actual del proyecto y lleve una organización en la realización de tareas. En Planner las tareas se dividen en las siguientes categorías:

- **No iniciado:** sección que contiene tareas que aún no han sido asignadas/aceptadas por los miembros del equipo o que aún no han sido iniciadas.
- **En curso:** sección que contiene las tareas en las que, uno o más miembros del proyecto, están trabajando.
- **Completada:** sección que contiene las tareas ya completadas por miembros del proyecto.

En cada categoría se irán depositando las tareas a medida que los recursos del proyecto actualicen el desarrollo de la propia tarea.

6.3 Planificación del proyecto

En un proyecto normal organizado con la metodología de Kanban dispondremos de distintos recursos que serían los encargados de escoger y completar las tarjetas del tablero, pero como el proyecto de Trabajo Fin de Grado (TFG) es un trabajo de desarrollo individual en este caso solamente dispondremos de un recurso (el alumno).

Pese a que en Kanban no se contemplan las etapas sino el desarrollo de tareas mediante el tablero, para poder mostrar la evolución del proyecto mostraremos tres fases del proyecto.

6.3.1 Planificación inicial

En un primer momento, definimos las tareas que componen nuestro proyecto. En la aplicación de Planner creamos un tablero donde definimos las tareas organizándolas en columnas,

en cada columna desplegamos las subtareas en forma de tarjetas que componen la tarea principal (Figura 6.1). Cada tarjeta se compone de las fechas de comienzo y fin estimadas, asignación de recursos, nivel de prioridad o una etiqueta que define el tipo de la tarea.

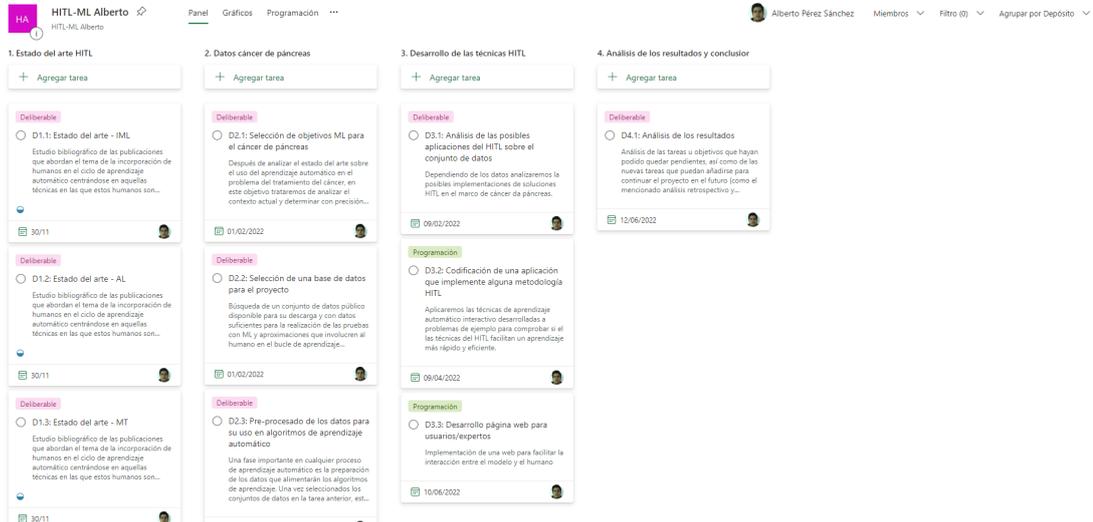


Figura 6.1: Depósito de planificación de tareas iniciales

En esta primera etapa, como consecuencia de la falta de conocimiento en el área de estudio y la necesidad de avanzar para encauzar el marco teórico y práctico del proyecto, definimos un conjunto de tareas iniciales de manera provisional y genéricas que sirvan para comenzar a trabajar en el desarrollo del proyecto. Las tareas se dividen en:

1. **Estudio bibliográfico de HITL y sus aplicaciones.** División del estudio del estado del arte entre las distintas ramas del HITL y las aproximaciones cercanas al mismo. A mayores estudiaremos los campos de aplicación donde resaltan cada uno.
2. **Obtención de un conjunto de datos.** Búsqueda de conjuntos de datos en los que las técnicas de HITL tengan relevancia al ser aplicadas. Generalmente son conjuntos pequeños donde los datos son difíciles de adquirir y tener una persona etiquetando supone un coste elevado.
3. **Desarrollo de las técnicas HITL.** Una vez estudiadas las aproximaciones y con los datos a nuestra disposición debemos aplicar alguna (o varias) de las aproximaciones que hagan partícipes a los humanos en el aprendizaje del modelo. También realizar una manera sencilla en la que los humanos interactúen con el modelo.
4. **Análisis de los resultados.** Dependiendo de los objetivos impuestos en desarrollo de las técnicas, al finalizar el proyecto será necesario revisar los objetivos y si fuese necesario tomar otras medidas.

Como ya comentamos, en las características de Kanban está la organización del trabajo de una manera controlada. En la Figura 6.2, se aprecia un pequeño grupo de tareas en curso, un grupo de tareas sin iniciar y un grupo de tareas finalizadas, que debido a que no encontramos en el comienzo del proyecto todavía no existe ninguna.

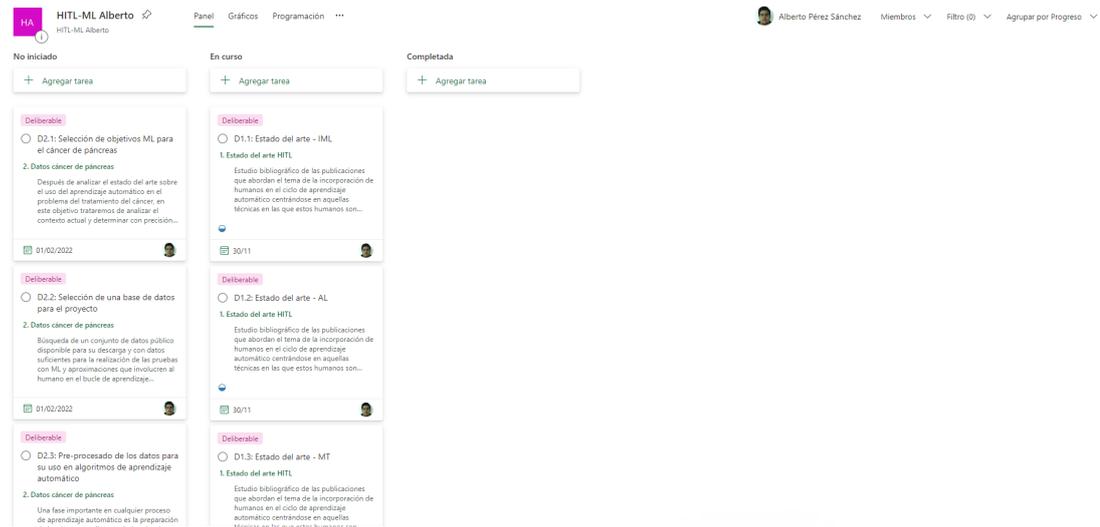


Figura 6.2: Kanban tareas iniciales

Por último, podemos tener una perspectiva más clara de la evolución del proyecto en la Figura 6.3, donde podemos apreciar como en este momento, únicamente están siendo desarrolladas las distintas tareas del estado del arte y su estudio de las aproximaciones HITL.

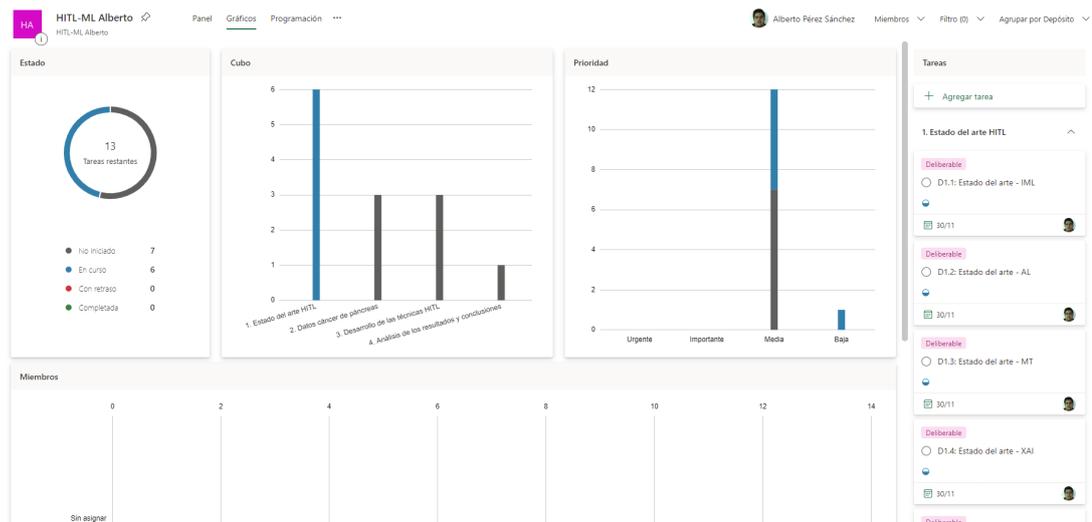


Figura 6.3: Gráfica de tareas iniciales

6.3.2 Etapa intermedia

Cuando un proyecto de investigación evoluciona en el tiempo es muy posible que los objetivos y tareas se modifiquen en función de los avances del proyecto. En nuestro caso particular las tareas principales en su conjunto se mantienen, aunque se despeja ya algunas de las incógnitas principales como: el papel del humano, representado por un experto en el ámbito de la salud; los datos, utilizando datos procedentes del TCGA de cáncer de páncreas; y el uso de aprendizaje activo, que aplicaremos sobre los datos del cáncer observando de cerca el comportamiento de esta técnica con la participación de los expertos. El conjunto de todas las tareas podemos en este punto del proyecto podemos encontrarlo en la Figura 6.4.

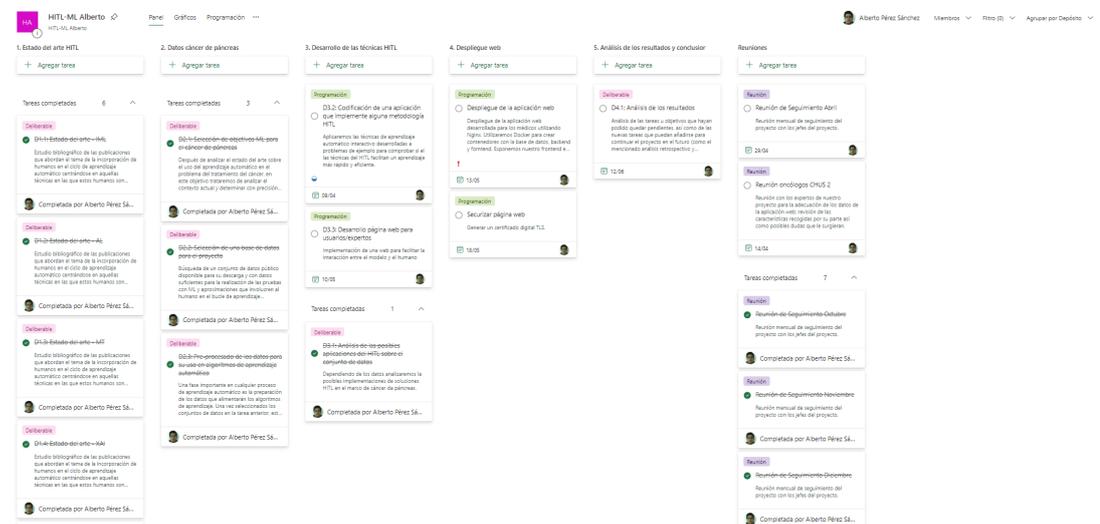


Figura 6.4: Depósito de tareas planificación intermedia

Algunas tareas que fueron apareciendo durante el transcurso del proyecto fueron:

- **Desarrollo de elementos necesarios para aplicar el aprendizaje activo.** Una vez decidimos aplicar el aprendizaje activo sobre los datos de cáncer de páncreas, necesitamos codificar las herramientas necesarias para entrenar al sistema con esta técnica.
- **Desarrollo de aplicación web, despliegue y securización.** Desarrollo y despliegue de una aplicación web para que los médicos puedan realizar las anotaciones de una manera más cómoda y efectiva.
- **Reuniones con oncólogos del CHUS.** Desde el momento de la introducción de los oncólogos del CHUS en el proyecto necesitamos realizar reuniones para contrastar datos, relevancia de los datos en el diagnóstico y su organización en la interfaz de la web.

- **Reuniones de seguimiento con los jefes de proyecto.** Reuniones simples para observar el progreso del proyecto y posibles problemas que surgen durante el mismo. Realizándose de manera sistemática una vez al mes.

A este punto del proyecto, en nuestro tablero Kanban muestra que en este punto nos estamos centrando en el desarrollo de las técnicas del HITL, en concreto de aprendizaje activo como se puede apreciar en la Figura 6.5.

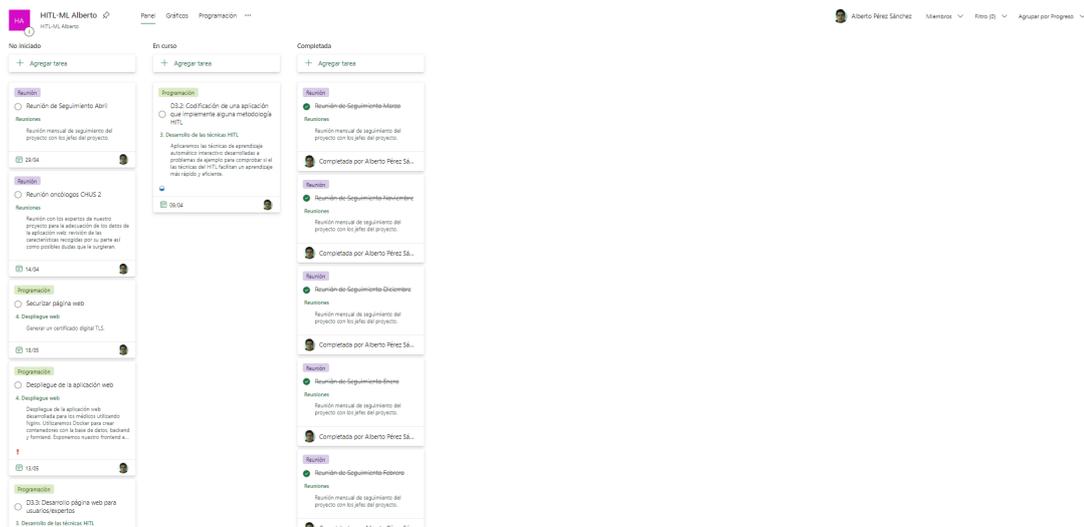


Figura 6.5: Kanban de tareas intermedias

El transcurso del proyecto, aún con los nuevas tareas y necesidades del proyecto, avanza de la manera planificada inicialmente como se puede observar en la Figura 6.6.

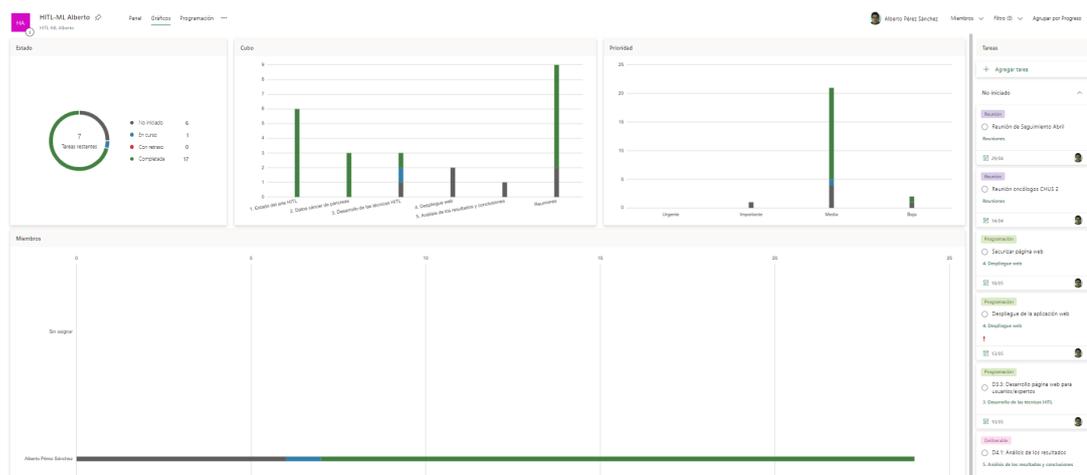


Figura 6.6: Gráfica de tareas intermedias

6.3.3 Evaluación final

Con la conclusión del proyecto, la duración total suma un total de 32 semanas, incluyendo el retraso causado por la anotación de los médicos y en el análisis de los datos por nuestra parte.

6.4 Estimación de costes

Cuando realizamos una estimación de costes tenemos que tener claro dos aspectos: el coste del material utilizado y el coste de nuestros recursos humanos. Para desglosar de donde proviene el coste total del proyecto entraremos en detalle en cada una de estas categorías.

6.4.1 Costes estimados de los materiales

En cuanto al hardware, el único gasto que debemos tener en cuenta es el de la máquina proporcionada por el CITIC para el proyecto, que nosotros utilizamos para desplegar la aplicación web para los médicos, y que estimamos un coste de 200€ desde el despliegue de la aplicación. En el aspecto del software todas las herramientas utilizadas son de código abierto, y por lo tanto, no suponen ningún coste para el proyecto.

6.4.2 Costes estimados de los recursos humanos

Para poder estimar los costes de los recursos humanos primero tenemos que definir todos los recursos que poseemos durante la realización del proyecto. Los roles de los que disponemos para el proyecto son: jefe de proyecto, analista y programador.

El rol de jefe de proyecto será ocupado por los dos directores de proyecto que estarán encargados de revisar el trabajo realizado, mientras que el alumno ocupará el resto de los roles (analista y programador) por ser el Trabajo Fin de Grado (TFG) un proyecto de carácter individual.

	Nº	Salario	Horas	Coste
Jefe de Proyecto	2	15	48	1440€
Analista-Programador	1	30	480	7200€
Experto (oncólogo)	4	30	40	1200€

Total: 9840€

En términos de jornada laboral realizamos media jornada o lo que es lo mismo 4 horas/día. Como el proyecto ha tenido una duración de 24 semanas efectivas (descontando tiempo de espera en las anotaciones de los médicos). La estimación total para el alumno sería 480 horas como analista y programador; los 2 directores del TFG con el papel de jefe de proyecto les asignamos 1 hora a la semana, resultando en un total de 48 horas totales en conjunto. El papel del experto es ocupado por los oncólogos encargados de asignar tratamientos a los pacientes mediante su historiales clínico, para tomar la decisión realizan un comité de expertos en el que participan entre dos y cuatro oncólogos, dependiendo de la carga de trabajo, por lo que supondremos el máximo, el número total de horas sería 10 horas/persona por cada experto lo que resultaría en 40 horas de desempeño en el proyecto.

Para el cálculo de los costes podemos referirnos a la tabla, dónde exponemos el número de recursos, el salario de cada recurso, las horas por recurso y el coste total que se obtiene de multiplicar los tres valores anteriores. Finalmente, el coste de todos los recursos humanos supone 9840€, que añadiendo el coste del hardware resultaría en un total de 10040€ para la realización del proyecto en su totalidad.

Conjunto de datos

PARA que el aprendizaje sea posible necesitamos un conjunto de datos que proporcione el historial clínico de pacientes afectados con cáncer de páncreas. Como conjunto de datos utilizaremos el proyecto del TCGA debido a su rigor y completitud en la información que proporcionan.

7.1 TCGA

El *The Cancer Genome Atlas (TCGA)* [51] es un proyecto del **National Cancer Institute (NCI)** en conjunto con el **National Human Genome Research Institute**, que tiene como objetivo crear una gran base de datos de información genética y clínica de cánceres. Los datos del TCGA están disponibles para el público y se pueden descargar de su portal web.

Aunque la mayor parte de los proyectos TCGA sobresalen en el aspecto genético, dónde centran sus esfuerzos en encontrar alguna novedad científica que pueda ayudar a encontrar causas del tumor, tratamiento o cura del propio cáncer, nosotros nos centraremos en los datos clínicos. La razón es debido a que no poseemos los conocimientos necesarios en el ámbito genético y que no son datos realistas con los que trabajar debido a que los propios médicos oncólogos no los utilizan.

Dentro del TCGA los proyectos se dividen en 33 categorías distintas de tumores. Como ya hemos comentado antes en este capítulo nosotros nos centraremos en el adenocarcinoma de páncreas que se conoce como el proyecto TCGA-PAAD.

7.1.1 TCGA-PAAD

Como ya comentamos cuando hablábamos del cáncer de páncreas en la Sección 5 el adenocarcinoma de páncreas es el tumor más común y no tendría sentido centrarse en otro tipo mientras que cerca del 90% de casos pertenecen a esta afección.

El *The Cancer Genome Atlas-Pancreatic Adenocarcinoma (TCGA-PAAD)* es el proyecto que contiene los casos de adenocarcinoma de páncreas. Este proyecto será utilizado para la realización de pruebas de aprendizaje activo con expertos oncólogos.

7.1.2 Obtención de datos

Para permitir el acceso a los datos el *National Cancer Institute (NCI)* proporciona una plataforma de divulgación de datos oncológicos: el *Genomic Data Commons (GDC)*. El GDC no es solo una base de datos o una herramienta, es una red de conocimiento ampliable que apoya la importación y estandarización de datos genómicos y clínicos de los programas de investigación del cáncer.

Los datos del TCGA-PAAD se componen de dos partes: datos genéticos y datos clínicos. Los datos genéticos se centran en el estudio exhaustivo de muestras causadas por tumores del tipo adenocarcinoma de páncreas que procuran mutaciones o alteraciones en el ADN. Los datos clínicos están compuestos por 185 historias clínicas reales de pacientes procedentes de distintos hospitales. Los datos se dividen en distintas categorías que desglosamos a continuación:

- **Datos de diagnóstico.** Datos procedentes de la investigación, el análisis y el reconocimiento de la presencia y la naturaleza de una enfermedad, afección o lesión a partir de los signos y síntomas expresados; también, la determinación científica de cualquier tipo; los resultados concisos de dicha investigación.
- **Datos de seguimiento.** Una visita de un paciente o participante del estudio a un profesional médico. Un encuentro clínico que abarca las intervenciones, procedimientos y evaluaciones planificadas y no planificadas del ensayo que pueden realizarse a un sujeto. Una visita tiene un comienzo y un final, cada uno descrito con una regla. Proceso mediante el cual se obtiene información sobre el estado de salud de un individuo antes y después de que se haya cerrado oficialmente un estudio; una actividad que continúa algo que ya ha comenzado o que repite algo que ya se ha hecho.
- **Factores de riesgo.** Información clínicamente relevante sobre el paciente que no es resultado inmediato de las predisposiciones genéticas.
- **Datos demográficos.** Datos para la caracterización del paciente mediante la segregación de la población (por ejemplo, caracterización por edad, sexo o raza).
- **Datos de tratamiento.** Registro de la administración e intención de los agentes terapéuticos suministrados a un paciente para alterar el curso de un proceso patológico.

- **Datos administrativos.** Datos que recogen las fechas de recolección de los datos y su procedencia.

En total el conjunto de datos contiene 158 atributos divididos entre las categorías anteriores, aunque dentro de los atributos existen datos que proporcionan gran cantidad de información en algunos casos no son relevantes o incluso están totalmente vacíos con lo que no aportan información. Por esto es necesario antes realizar un preprocesado eliminando los datos que no aportan información interesante para el proyecto.

Red Neuronal Artificial (RNA)

Las redes neuronales artificiales han supuesto un gran avance en el campo de la inteligencia artificial, teniendo un gran impacto como sistema de clasificación. Nuestro proyecto aborda el problema de asignación de tratamiento de quimioterapia a pacientes de cáncer de páncreas utilizando una red neuronal artificial para su resolución, configurándola de manera óptima para los datos que disponemos.

8.1 Red Neuronal Artificial

Las redes neuronales artificiales (RNAs) son modelos computacionales que procesan información imitando el funcionamiento del cerebro humano y de cómo interactúan las neuronas. El propósito de una RNA es imitar el funcionamiento del cerebro humano con la capacidad de "aprender" y "memorizar". La arquitectura de una red neuronal artificial consiste en un número de elementos simples interconectados conocidos como **neuronas artificiales**, que se usan como réplica a la forma de pensamiento humana.

8.1.1 Neurona Artificial

Las **neuronas artificiales** o **neuronas de McCulloch-Pitts** son modelos matemáticos de procesamiento de la información que emulan el funcionamiento biológico de las neuronas. Cada neurona artificial recibe un vector de características de entrada (x) con un peso asociado (w), se calcula el sumatorio de todas las entradas multiplicadas por sus pesos al que se le añade un factor de sesgo o *bias* (b). El resultado del sumatorio se le aplica una función de activación $f(x)$ que es la responsable de decidir si la neurona se activa o no. En la Figura 8.1 se aprecia los componentes y cómo se realizan los cálculos.

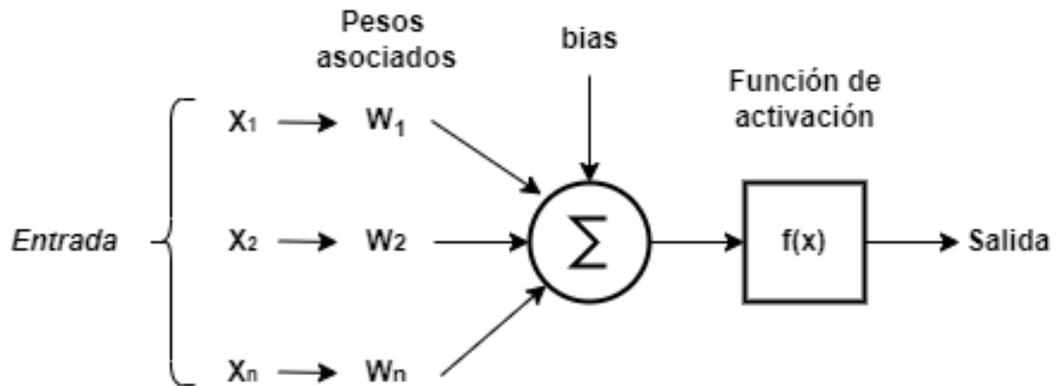


Figura 8.1: Diagrama de neurona artificial

El mayor inconveniente que presenta una neurona es su limitación a la hora de resolver problemas no lineales, debido a solamente pueden realizar separaciones lineales sobre los datos de entrada. Para solventar problemas no lineales se realizan agrupaciones de neuronas artificiales de manera secuencial que reciben el nombre de redes neuronales artificiales.

8.1.2 Redes Neuronales Artificiales

Las redes neuronales artificiales son agrupaciones de neuronas artificiales organizadas en capas conectadas entre sí, donde cada capa es una colección de neuronas que operan de manera conjunta.

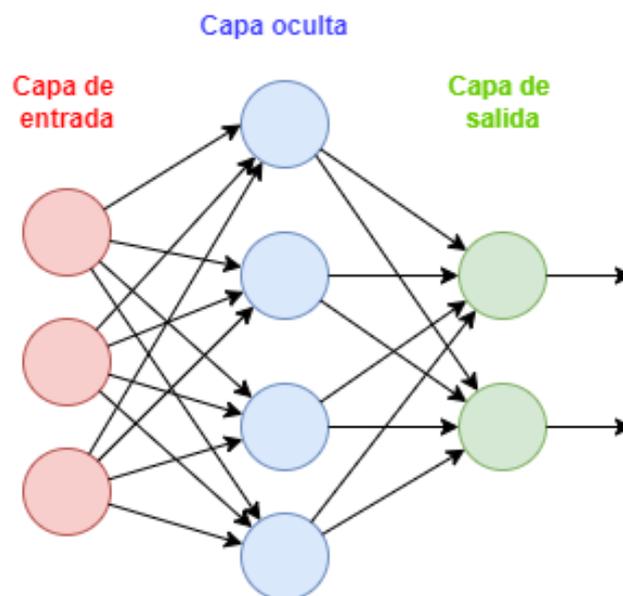


Figura 8.2: Diagrama de una red neuronal artificial

La Figura 8.2 muestra la estructura habitual de un red de neuronas artificiales dividida en tres capas: la **capa de entrada**, que representa el vector de características de entrada; las **capas ocultas**, dónde se produce el aprendizaje; la **capa de salida**, que es la encargada de proporcionar el resultado. Cabe señalar que el número de capas ocultas puede variar dependiendo del problema de aprendizaje.

La forma en que se conectan las neuronas da lugar a distintas estructuras dependiendo de las direcciones que toman las salidas de las neuronas. La más común son las **redes unidireccionales o feedforward**, un ejemplo de este tipo de red sería la Figura 8.2. Estas redes como su propio nombre indica son redes que solo contemplan la conexión de salida con la siguiente capa. Cuando las conexiones entre neuronas tienen uniones con distintas capas, con su misma capa o con la propia neurona se denominan **redes recurrentes o feedback**.

8.1.3 Función de activación

La función de activación es una parte fundamental del diseño de una red neuronal. Cuando nos encontramos con problemas no lineales la suma ponderada de las neuronas no es capaz de solucionar el problema, para ello utilizamos la función de activación que distorsiona el valor de la salida de la neurona añadiendo deformaciones no lineales para así con la encadenación de distintas neuronas resolver problemas imposibles de otra forma.

Técnicamente en cada neurona utiliza una función de activación independiente, aunque las redes neuronales están diseñadas para utilizar la misma función de activación para cada capa independientemente.

8.1.4 Función de pérdida

La función de pérdida sirve para medir la efectividad de nuestro modelo, a través de ella podemos minimizar y mejorar la pérdida de nuestro modelo. Esta función se utiliza en los algoritmos de aprendizaje supervisado y sirve para estudiar la desviación entre las predicciones de la red neuronal artificial y el valor real. El objetivo de cualquier red neuronal es minimizar el resultado de la función de pérdida para mejorar la precisión de nuestro modelo.

Dependiendo del tipo de problema existen distintos tipos de funciones de pérdida. Una de las más comunes es la pérdida de entropía cruzada o *cross-entropy loss* empleada en problemas de clasificación. En la red neuronal artificial desarrollada se ha aplicado esta función de pérdida durante el proceso de entrenamiento.

Aunque la función de pérdida nos indique la precisión de nuestro modelo es necesario actualizar el modelo dependiendo de su valor. Para ello utilizamos un algoritmo llamado *back-propagation* que expande el error por las neuronas de nuestro modelo modificando los pesos y repartiendo responsabilidades del error entre las neuronas.

8.1.5 Propagación del error

La **propagación hacia atrás** o *backpropagation* es una técnica de entrenamiento que permite que la red neuronal aprenda y corrija sus errores. En los modelos de redes neuronales el error se propaga hacia atrás desde la señal de error a las otras capas, se realiza de esta manera debido a que el error de las capas anteriores tiene influencia directa en el error de las capas posteriores. A mayor sea la profundidad de la propagación del error más se diluye el error, haciendo que en las capas más alejadas de la salida la variación de los pesos sea menor.

8.2 Optimización de hiperparámetros

Durante el proceso de aprendizaje el sistema es capaz de modificar sus pesos para mejorar el aprendizaje y minimizar el error en la predicción, pero existen parámetros que el propio sistema no puede ajustar, conocidos como **hiperparámetros**, y que tienen que ser ajustados de manera manual por el humano.

El ajuste de hiperparámetros, también conocido como optimización de hiperparámetros es el proceso encontrar la configuración de hiperparámetros que produzca el mejor rendimiento.

En los comienzos del aprendizaje máquina los hiperparámetros comenzaron ajustándose de manera manual, lo que suponía una aproximación muy costosa en el tiempo. Con el aumento de la capacidad computacional de los ordenadores se comenzaron a utilizar técnicas automáticas más eficientes. En nuestro modelo se ha aplicado la técnica de *grid search* en la optimización de los hiperparámetros, esta técnica consiste en una búsqueda exhaustiva de los hiperparámetros en un rango de valores proporcionado por el humano. El *grid search* es una técnica muy costosa computacionalmente debido a que realiza pruebas con todas las configuraciones posibles de hiperparámetros. En redes con pocos parámetros a configurar y sobretodo pocas capas ocultas (nuestro caso particular) es una de las mejores opciones debido a que siempre se encontrarán los mejores hiperparámetros, en contra si tenemos una red con una gran cantidad de capas ocultas quizás no sea posible su uso debido a que tiene una complejidad exponencial.

8.3 Estructura

La estructura de la red neuronal de nuestro proyecto puede observarse en la Figura 8.3 y está relacionada de una manera directa con los datos y el problema a resolver. La entrada del modelo tiene una dimensión de 56 debido a que representa la cantidad de datos preprocesados del nuestro conjunto de datos inicial con los que alimentamos el modelo. La salida de la última capa está asociada con la solución del problema de predicción entre las opciones de "Quimioterapia" o "Sin tratamiento".

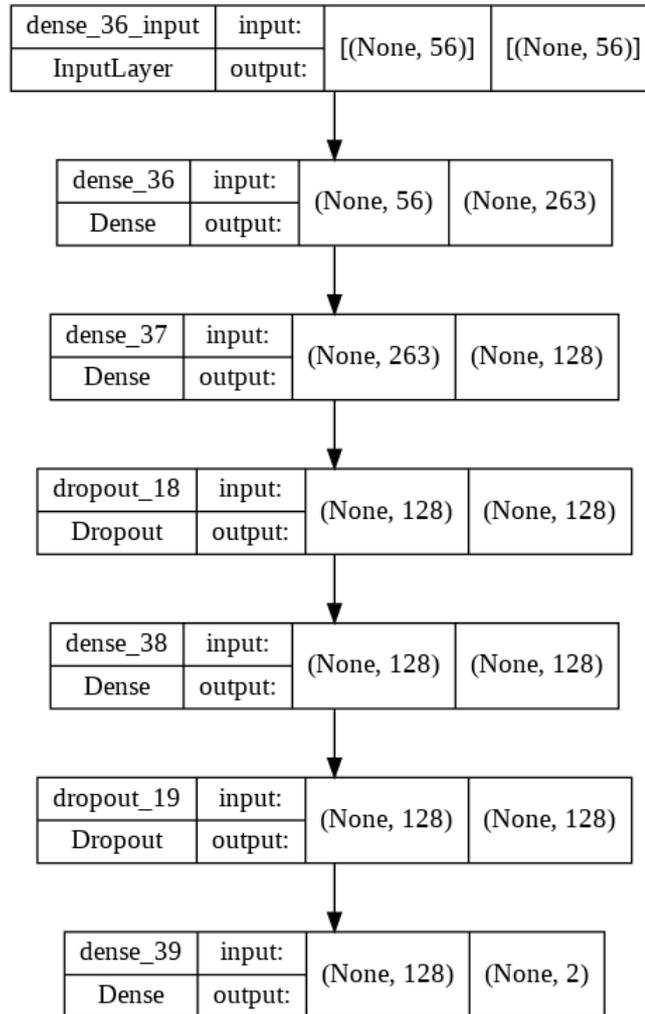


Figura 8.3: Estructura red neuronal artificial

Nuestro sistema final está compuesto por dos capas ocultas, lo que lo convierte en una red de aprendizaje profundo simple, de tamaño 128 cada una. Definir el número de capas ocultas y su tamaño puede ser un proceso complejo, que de ser realizado manualmente puede suponer un esfuerzo que no asegura una buena configuración y estructura en la red neuronal. En la optimización de los hiperparámetros se han automatizado también distintas estructuras con distintos número de capas ocultas y la que obtuvo los resultados óptimos es la que hemos implementado. Añadimos en cada una de las capas ocultas una capa de abandono, conocida en inglés como *dropout*, para mejorar los resultados y evitar el sobreajuste.

El sobreajuste de un modelo puede definirse como el fenómeno en el cual un modelo de aprendizaje automático se adapta de forma excesiva al conjunto de entrenamiento y, por lo tanto, pierde capacidad de generalización para el conjunto de pruebas. El *dropout* en las redes

neuronales artificiales consiste en desactivar de manera aleatoria un porcentaje de las neuronas de la capa sobre la que se aplica durante la fase entrenamiento. De este modo forzamos a que el modelo aprenda los patrones de entrada de forma más robusta, al tener que aprender los mismos patrones de entrada con neuronas diferentes.

8.4 Validación

La validación es un paso fundamental del modelo que permite conocer la capacidad del modelo de generalizar las características de los datos y no simplemente memorizar los datos de entrenamiento. Se han analizado las dos aproximaciones más comunes:

- **Hold-out:** divide el conjunto de datos en dos: entrenamiento y test. El conjunto de entrenamiento es sobre el que se entrena el modelo, mientras que el conjunto de test sirve para observar cómo de bien funciona el modelo sobre casos que nunca ha visto antes. La aproximación de *hold-out* depende de la separación a la hora de escoger los conjuntos, en conjuntos pequeños pequeños puede afectar de manera notable a la hora de la construcción y resultado final del modelo.
- **Validación cruzada k-iteraciones:** el conjunto de datos es dividido en k grupos. Uno de los grupos es usado como conjunto de test y el resto son usados como conjunto de entrenamiento. Este proceso se repite hasta que todos los grupos hayan sido utilizados como conjunto de test. Entre sus desventajas se sobresale el aspecto computacional debido a que hay que entrenar y validar el modelo k veces.

Nosotros utilizaremos la técnica de validación cruzada, en gran medida debido al número reducido de datos que comprende nuestro conjunto de datos. Si utilizamos una estrategia de *hold-out* con tan pocos ejemplos podemos provocar sesgos en nuestro conjunto de datos de entrenamiento que no permitirían evaluar correctamente como de precisas son las predicciones del modelo.

Aplicación Web

EN el aprendizaje activo es necesario proporcionar una aplicación para que la persona que tiene el conocimiento pueda observar los casos que le proporciona la máquina. Puede ser incluso de la manera más simple, como en un terminal en el que se muestran los datos por pantalla, pero la verdadera necesidad es realizar entornos de trabajos optimizados y agradables a los usuarios, tanto expertos como no expertos, para así conseguir mejores anotaciones y optimizar el tiempo de los humanos.

9.1 Herramientas utilizadas

Para hablar de la estructura de la aplicación web llevaremos a cabo una descomposición de los componentes que la constituyen. En la Figura 9.1 observamos que el sistema se compone principalmente de dos elementos, el servidor, también conocido como *back-end*, y el cliente web o *front-end*.

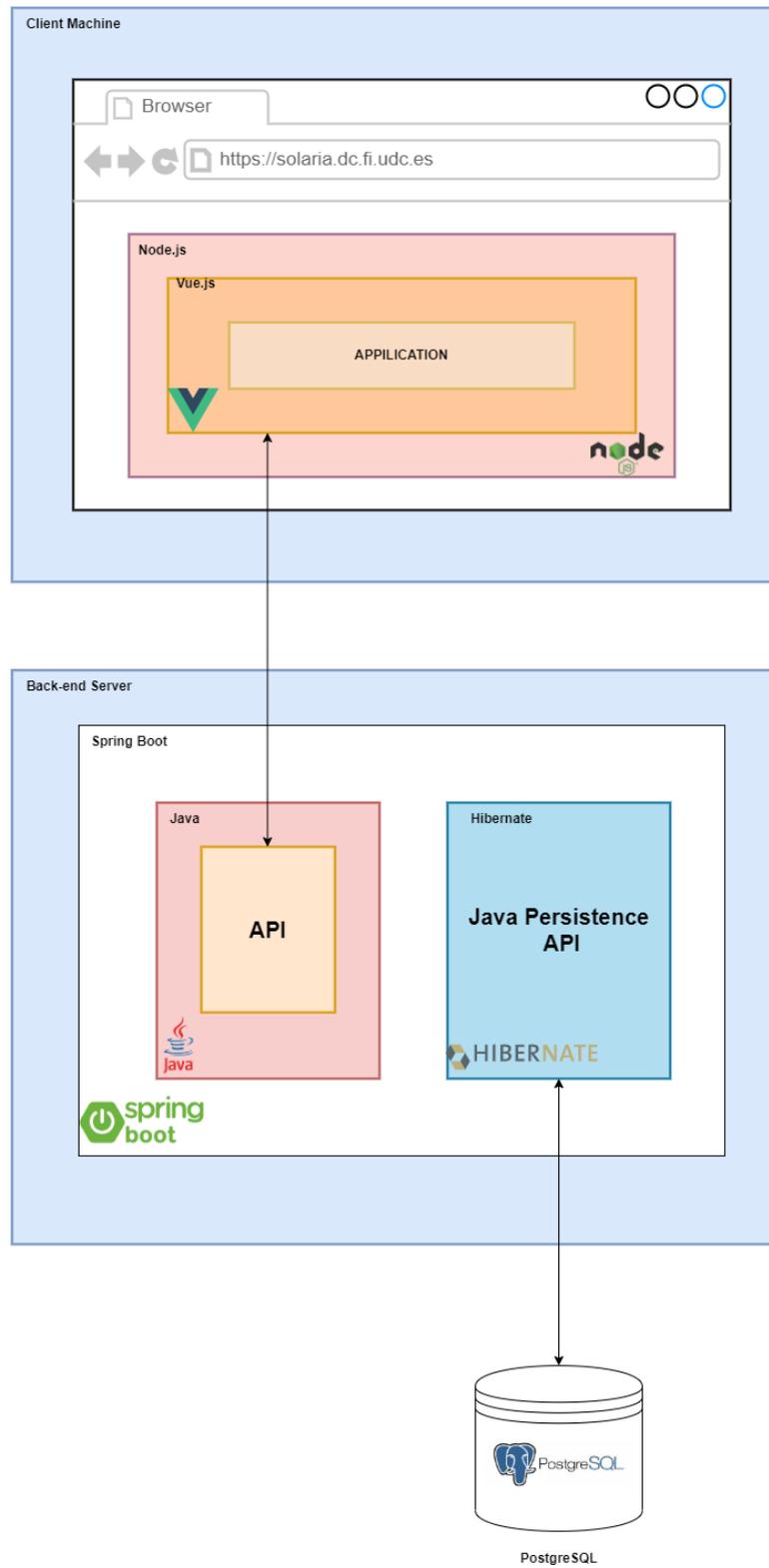


Figura 9.1: Arquitectura de las tecnologías utilizadas

9.1.1 Front-end

El *front-end* o cliente es la parte encargada de proporcionar una interfaz gráfica y de gestionar la interacción con el usuario, mediante peticiones al servidor. En este apartado destacan el uso de las siguientes tecnologías:

- **Vue.js:** Entorno de trabajo de código abierto de JavaScript enfocado en construir interfaces de usuario de una manera sencilla y escalable. Utilizado en el desarrollo del apartado del cliente de la aplicación.
- **Node.js:** Entorno en tiempo ejecución basado en Javascript. Es la plataforma que se ha utilizado para el desarrollo del cliente de la aplicación.
- **Vuetify.js:** Librería de Vue.js destinada a la creación de interfaces de usuario que implementa componentes de usabilidad listos para usar. Utilizamos esta librería para el diseño del cliente web.
- **Axios:** Cliente HTTP basado en promesas que permite gestionar las llamadas AJAX procedentes del cliente web desarrollado.
- **Npm:** Sistema de gestión de paquetes por defecto de Node.js.

9.1.2 Back-end

El *back-end* o servidor es el encargado de proporcionar las funcionalidades de la aplicación mediante un API REST, que ofrece al cliente los servicios necesarios para su correcto funcionamiento. En el desarrollo del servidor se utilizaron tecnologías como:

- **SpringBoot:** Entorno de trabajo de código abierto para creación de aplicaciones en el lenguaje de programación Java. Comúnmente utilizado en aplicaciones basadas en Spring y escogida para el desarrollo del servidor.
- **PostgreSQL:** Sistema de gestión de bases de datos relacional orientado a objetos que utiliza el estándar SQL.
- **Hibernate:** Framework que agiliza la relación entre la aplicación y la base de datos.
- **Maven:** Herramienta software para la gestión y construcción de proyectos Java.

9.1.3 Otras aplicaciones

Debido a que el CITIC nos proporcionó una máquina para el desarrollo de nuestro proyecto decidimos desplegar la aplicación web en ella, para ello utilizamos:

- **Docker:** Plataforma creada con el fin de desarrollar, implementar y ejecutar aplicaciones dentro de contenedores.
- **Nginx:** Servidor web de código abierto que ofrece contenido estático de un sitio web con un bajo uso de memoria y alta concurrencia. También ofrece funcionalidades de proxy inverso y equilibrio de carga.

9.2 Interacciones Humano-Máquina

En la metodología Human-in-the-loop un aspecto relevante y a tener en cuenta es la interactividad. El aprendizaje máquina interactivo (IML) y la enseñanza máquina (MT) son aproximaciones con una interacción más compleja y cercana entre humanos y máquinas que requiere de una interacción mayor que en el caso del aprendizaje activo (AL). Esto no significa que en el aprendizaje activo no existan interacciones que nos puedan afectar de manera directa y de las que debemos cuidar los detalles.

En términos generales, el AL busca minimizar el tiempo que emplean los humanos revisando datos. Con esto buscamos: evitar la fatiga o aburrimiento, para garantizar que las anotaciones son de alta calidad; evitar desperdiciar el tiempo de los expertos que son personas ocupadas, en nuestro caso particular del dominio de la salud pública dónde los médicos tienen labores vitales para otras vidas.

Para minimizar el tiempo de anotación en nuestro proyecto seguimos dos estrategias:

1. Minimizar el número de casos por lotes que presentamos a los expertos; prefiriendo hacer mayor número de sesiones de anotaciones.
2. Minimizar el tiempo que los expertos gastan anotando cada caso individualmente.

Estas estrategias se consiguen proporcionando a los expertos un entorno de anotación cómodo y enfocado al problema particular; un firme partidario de la aplicación diseñada específicamente para la anotación de expertos es Munro en [1]. En su libro Munro dedica especial importancia al diseño de páginas de anotación orientadas a mejorar la eficacia de las anotaciones.

En nuestro proyecto hemos desarrollado una aplicación web para que los expertos puedan anotar los casos con facilidad y disponibilidad (Figura 9.2). También hemos tenido en cuenta la bibliografía sobre usabilidad, en especial la heurística de Nielsen, como el “diseño estético y minimalista” y la “flexibilidad y eficiencia de uso” [52]. Un ejemplo de la primera sería generar el menor número posible de estímulos visuales para que los expertos sólo se centren en proporcionar la información, por lo que el aspecto visual de nuestra página es sencillo y sobrio, sin distracciones adicionales. Un ejemplo de lo segundo es organizar los datos en categorías y

hacer de cada categoría un elemento ampliable dentro de la página web, lo que permite a los expertos centrarse en determinadas partes de los datos sin exponer todo el conjunto de datos de una sola vez.

The screenshot displays a web form with several sections:

- 'Patient'**: A dropdown menu with a downward arrow.
- 'StageEvent'**: A section with an upward arrow, containing the following text:
 - System Version : "7th"
 - Pathologic Stage : "Stage IIB"
 - Pathologic_T : "T3"
 - Pathologic_N : "N1"
 - Pathologic_M : "MX"
- 'ClinicalData'**: A dropdown menu with a downward arrow.
- 'NewTumorEvent'**: A dropdown menu with a downward arrow.
- 'ExternalFactor'**: A dropdown menu with a downward arrow.
- 'Radiation'**: A dropdown menu with a downward arrow.
- Treatment to patient**: A section with three radio buttons: Chemotherapy, No treatment, and Undecided. Below this is a text input field labeled "reason for chemotherapy assignment".
- Synthetic?**: A section with two radio buttons: Yes and No.
- Comment on the case**: A section with a text input field containing the placeholder text: "Anything related to the case that deserves to be mentioned. (optional)".

Figura 9.2: Interfaz de usuario

9.3 Datos recogidos

Una vez que tenemos a los médicos trabajando es nuestro deber obtener la máxima utilidad posible de cada lote de anotaciones. Cada lote consta de 10 casos clínicos de pacientes con afecciones de cáncer de páncreas, entre los cuales hay dos casos del conjunto de datos iniciales, y por cada caso se recogen los siguientes:

- **Tratamiento.** Selección del tratamiento del paciente entre aplicar quimioterapia o no. A mayores si existen incoherencias que no permitan un diagnóstico claro existe la op-

ción de “indeciso”.

- **Razonamiento del tratamiento.** Razonamiento escrito del porqué del tratamiento seleccionado. Esta categoría se emplea para en un futuro poder crear un sistema XAI explicado en la Sección 2.5.
- **Veracidad del caso.** En cada lote de casos existen dos casos reales que provienen de los datos del TCGA. El experto selecciona si cree que los datos son reales o generados sintéticamente. Este apartado sirve para comprender cómo de bien funciona el generador de datos sintéticos.
- **Sección de comentarios.** Apartado reservado a la aportación que el experto desee proporcionar. En la gran mayoría de casos expone las incoherencias o peculiaridades de los casos. A partir de los comentarios generamos las reglas que aplicamos en la generación de los datos en la siguiente iteración.

Experimento y resultados

EN este capítulo analizaremos los resultados obtenidos durante la aplicación de las técnicas estudiadas en el proyecto. Desde una primera fase con los datos del conjunto de datos inicial hasta el último ciclo de aprendizaje activo realizado con los datos recolectados a los expertos.

10.1 Conducción del experimento

En un experimento de aprendizaje máquina clásico los resultados suponen un punto clave, observando hasta qué punto nuestro sistema de clasificación es capaz de perfeccionar la tarea para la que está diseñado. En nuestro caso particular, pese a que los resultados son importantes, también tenemos en cuenta otros factores como la evolución del modelo en cada etapa de aprendizaje activo, minimizando el esfuerzo de nuestros expertos e intentando mejorar los resultados.

Una vez ya hemos construido todas las herramientas necesarias para la aplicación del aprendizaje activo solamente nos queda comenzar con el flujo de trabajo. En la Figura 10.1 mostramos un panorama general de cómo se realiza el proceso de entrenamiento, sin entrar en cuestiones técnicas.

En el sistema consta de cuatro partes:

- **La red neuronal artificial.** Modelo ajustado a las características del problema del cáncer de páncreas. Utilizado para la asignación de tratamiento de quimioterapia a los pacientes.
- **El conjunto de datos.** Fuente de conocimiento para el modelo que alberga los casos anotados iniciales y los casos añadidos durante las iteraciones de aprendizaje activo.
- **La CTGAN.** Parte encargada de la generación de historiales de cáncer pancreático sin etiquetar (sin tratamiento asignado).

- **Los expertos.** Ente de conocimiento extenso en el campo de estudio encargado de etiquetar los casos que generan mayor incertidumbre al modelo.

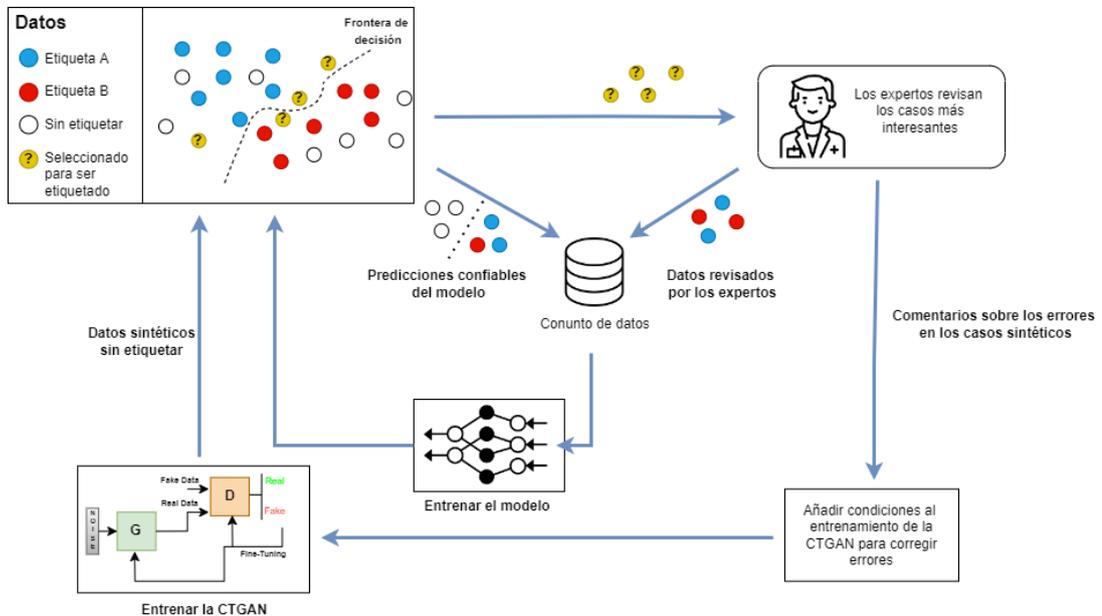


Figura 10.1: Ciclo del experimento

Todos los componentes utilizados se comunican entre ellos y siguen un proceso ordenado de trabajo que ilustramos a continuación:

1. Entrenar a la CTGAN para generar los datos sin etiquetar.
2. Entrenar el modelo con los datos disponibles en el conjunto de entrenamiento.
3. Una vez entrenado el modelo, presentarle pequeños lotes de elementos sin etiquetar para obtener las predicciones que realiza el modelo.
4. Los elementos más dudosos son anotados por un experto u "oráculo" que decide la etiqueta.
5. El experto proporciona información sobre la percepción del caso presentado, indicando si cree que el caso es generado sintéticamente, razón del tratamiento o comentando si existe algún tipo de error o incongruencia en los datos. Con la información recolectada ajustamos las reglas nuestro generador de datos sintéticos para mejorar las réplicas generadas en las siguientes iteraciones.
6. Añadir los nuevos elementos al conjunto de entrenamiento y entrenar el modelo con el nuevo conjunto de datos.

7. Validar los resultados y en caso de no satisfacer los objetivos previamente marcados comenzar de nuevo.

Nuestro proceso de entrenamiento de aprendizaje activo difiere del enfoque clásico en dos puntos. Primero, en el aprendizaje activo clásico necesitamos de un conjunto de datos sin etiquetar. Segundo, en el aprendizaje activo solamente se tiene en cuenta las anotaciones de los expertos y no se recoge información auxiliar.

Debido a que la decisión de asignar quimioterapia no es trivial y, por lo tanto, no es una decisión fácil ni rápida debemos seleccionar cuidadosamente dónde centrar nuestros esfuerzos. Los oncólogos que realizan el papel del experto en nuestro proyecto son personas ocupadas y no disponemos de su tiempo como nos gustaría. La falta de expertos que etiqueten los casos es otra de las razones por las que aplicar aprendizaje activo permite etiquetar solamente los casos más críticos.

10.2 Resultados

Como consecuencia de los esfuerzos puestos en el proyecto es necesario analizar los resultados del proyecto, fijándonos en la precisión de clasificación del modelo, y su evolución desde el comienzo del experimento hasta la última iteración.

Los resultados son relativos, de manera análoga al movimiento donde un objeto siempre se mueve respecto a otro, la mejora que el modelo consigue con las iteraciones del aprendizaje activo solo es apreciable respecto a una iteración anterior. Esto no significa que los resultados no sean importantes para nosotros, sino que tenemos que tener claro nuestro punto de partida para observar cómo evolucionan los resultados y su impacto desde el punto inicial.

Exactitud (*accuracy*)

En el análisis de los resultados utilizaremos la métrica de exactitud o *accuracy*. Esta métrica se define como la proporción de casos clasificados correctamente entre los casos totales.

$$accuracy = \frac{VP + VN}{VP + FP + VN + FN}$$

Siendo VP la cantidad de casos positivos etiquetados correctamente, VN la cantidad de casos negativos etiquetados correctamente, FP la cantidad de casos negativos etiquetados como positivos y FN la cantidad de casos positivos etiquetados como negativos.

10.2.1 Etapa inicial

Nuestro conjunto de datos inicial contiene solo 181 casos de cáncer de páncreas (algunos casos fueron descartados debido a estar incompletos). Con tan pocos datos vamos utilizar la

técnica de validación cruzada como ya explicamos en la sección 8.4 para así minimizar la aleatoriedad de los conjuntos de test y entrenamiento.

Nuestro punto de partida se encuentra representado en la Figura 10.2 donde se puede observar que la precisión de nuestro clasificador ronda el 60% de acierto.

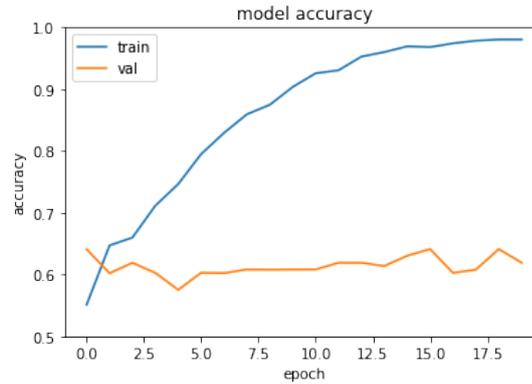


Figura 10.2: Modelo entrenado con datos iniciales.

10.2.2 Iteración final

El proyecto actualmente consta de tres iteraciones del bucle AL. Tras la finalización de la tercera iteración obtuvimos un conjunto de datos con 292 casos, de los cuales solamente 30 casos fueron revisados y etiquetados por los médicos.

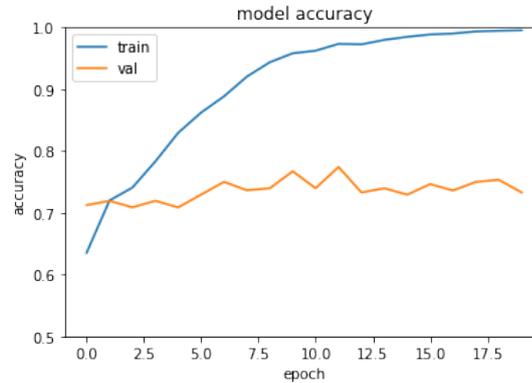


Figura 10.3: Modelo entrenado siguiendo una estrategia AL.

El resultado final obtenido tras la tercera iteración ronda el 73% de precisión, lo que supone una mejoría total de un 13% entre los datos anotados inicialmente y los datos aplicando técnicas de aprendizaje activo.

10.2.3 Comparativa

La mejor forma de observar el avance iteración a iteración es en la Figura 10.4 en la que podemos observar como, en cada iteración, la clasificación del modelo aumenta razonablemente con el pequeño conjunto de datos que se añaden en cada iteración.

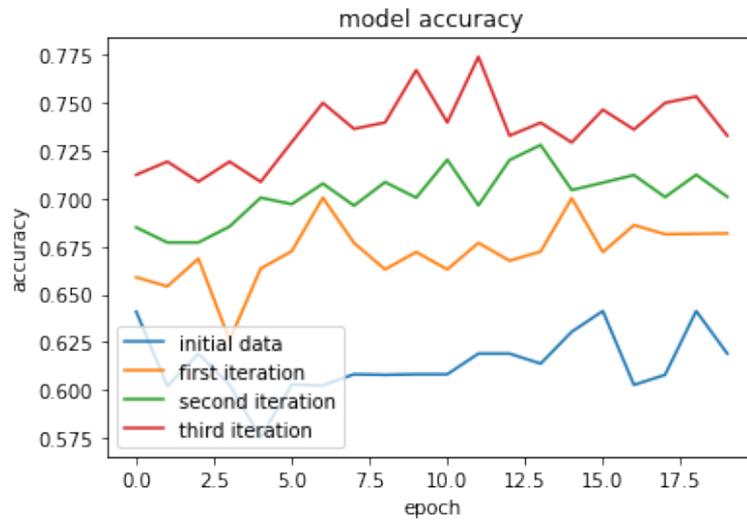


Figura 10.4: Comparación entre iteraciones en el entrenamiento de AL.

Con el paso de las iteraciones el aumento en la precisión disminuye de manera comprensible debido al aumento del número de casos totales. Aún así con sesiones de 10 casos anotados por los oncólogos la mejoría es notable y con más tiempo sería interesante ver que punto de refinamiento es capaz de alcanzar el modelo.

Conclusiones

EL papel de los humanos en el aprendizaje máquina tradicional está actualmente estancado sin aprovechar los puntos fuertes donde pueden mejorar las capacidades del modelo. Para desmontar esta situación y aportar una visión renovada del aprendizaje máquina creamos un aplicación basada en aprendizaje activo.

11.1 Conclusiones

Las técnicas del *Human in the loop* están en plena expansión principalmente debido el estancamiento en algunos apartados de los sistemas de aprendizaje pasivo y a las capacidades argumentativas que los humanos pueden aportar a los modelos de aprendizaje automático.

Podemos destacar, a razón de los objetivos marcados al comienzo del proyecto, que se pueden observar en la Sección 1.3, que todos los objetivos han sido completados de manera satisfactoria. Cabe decir que aunque los resultados no son brillantes en precisión y tienen margen de mejora, no era la prioridad principal obtener unos resultados excepcionales en la predicción sino ilustrar el papel del humano como factor clave en el bucle de entrenamiento máquina.

Para la realización del proyecto, debido a la falta de datos del propio dominio de aplicación, aplicamos técnicas de aumento de datos, escogiendo entrenar un tipo específico de GAN para esta tarea (CTGAN). Hemos refinado los resultados de los propios datos sintéticos gracias a los comentarios de los expertos, que hemos convertido en reglas para la GAN. Quizás este apartado suponga la zona donde el papel del humano tuviera más impacto aunque no sea fácil de observar su incidencia.

Durante el desarrollo de técnicas de muestreo para el aprendizaje activo, hemos aplicado los conocimientos de las ramas de computación en la creación de sistemas de aprendizaje automático, puliendo los conceptos y aumentando las bases expuestas en las asignaturas del grado. Otros aspecto relevante de nuestra aplicación, utilizando los conocimientos de asig-

naturas como Internet e Sistemas Distribuidos (ISD), es el desarrollo de una página diseñada específicamente para la anotaciones de los expertos. No simplemente por el hecho de la realización de una aplicación web, sino por las técnicas aplicadas en su desarrollo .

Se ha realizado un análisis de resultados por cada iteración, configurando el sistema y generando nuevos casos en un proceso iterativo. Los resultados se han comportado de la manera esperada desde el primer momento mejorando de manera considerable y continuada mediante el uso responsable de nuestro recursos humanos.

En conclusión, se ha probado que el desarrollo de las técnicas que incorporan al humano en el bucle de aprendizaje pueden contribuir de manera positiva, en casos en los que no disponemos de suficientes datos anotados, provocando un menor gasto de nuestros recursos de anotación y maximizando la calidad de las anotaciones.

11.2 Futuras líneas de trabajo

Nuestro proyecto abre las puertas a la colaboración de humanos en el proceso de aprendizaje de las máquinas, demostrando que su incorporación provoca mejoras sustanciales de manera directa o indirectamente. Sin embargo, quedan muchas alternativas por explorar que servirían para ilustrar el impacto humano en distintas situaciones y aproximaciones:

- El conocimiento de los expertos (oncólogos) parte de distintas fuentes de conocimiento (guías, libros, etc) que siguen una serie de reglas. Crear un sistema de reglas que realice el mismo diagnóstico sobre los mismos casos que los expertos y observaremos si existen diferencias y la procedencia de las discrepancias.
- Como las sesiones de anotación son escasas, debido principalmente a la apretada agenda de los oncólogos, si disponemos de más tiempo para realizar pruebas sería interesante comprobar la eficacia del *uncertainty sampling* comparándolo con un muestreo aleatorio.
- Durante la propia evolución del proyecto se han ido encontrando errores en los propios casos iniciales del TCGA, encontrar mediante *diversity sampling* estos casos extraños e intentar mitigar su impacto.
- Crear un sistema explicable que a mayores del propio diagnóstico proporcione un razonamiento que justifique su elección.
- Un sistema de aprendizaje mediante currículum, con la ayuda de los expertos, donde los casos se enseñan de manera gradual ordenados por su complejidad.

- Los datos genéticos del TCGA esconden una complejidad difícil de trabajar. Con la ayuda de un biólogo con extensos conocimientos en el campo podríamos estudiar los datos genéticos que esconden la vanguardia del cáncer de páncreas.
- Utilizando los datos del CHUS (Complejo Hospitalario Universitario de Santiago) adaptar el modelo creado a sus datos. Los datos que disponemos en este proyecto proceden de Estados Unidos y no son exactamente iguales a los que se utilizan en nuestros sistemas de salud.

Bibliografía

- [1] R. Munro, *Human-in-the-Loop Machine Learning*. Manning Publications, 2020.
- [2] E. Mosqueira-Rey, E. Hernández-Pereira, D. Alonso-Ríos, J. Bobes-Bascarán, and A. Fernández-Leal, “Human-in-the-loop machine learning: A state of the art,” *Artificial Intelligence Review*, vol. in press, 2022.
- [3] C. C. Bonwell and J. A. Eison, “Active learning: Creating excitement in the classroom. 1991 ashe-eric higher education reports.” in *Active Learning: Creating Excitement in the Classroom. 1991 ASHE-ERIC Higher Education Reports.*, 1991.
- [4] C. Sammut and R. Banerji, *Learning Concepts by Asking Questions*, 01 1986, vol. 2, pp. 167–192.
- [5] B. Settles, “Active learning literature survey,” University of Wisconsin–Madison, Computer Sciences Technical Report 1648, 2009. [En línea]. Disponible en: <http://axon.cs.byu.edu/~martinez/classes/778/Papers/settles.activelearning.pdf>
- [6] M. WARE, E. FRANK, G. HOLMES, M. HALL, and I. H. WITTEN, “Interactive machine learning: letting users build classifiers,” *International Journal of Human-Computer Studies*, vol. 55, no. 3, pp. 281–292, 2001. [En línea]. Disponible en: <https://www.sciencedirect.com/science/article/pii/S1071581901904999>
- [7] R. Porter, J. Theiler, and D. Hush, “Interactive machine learning in data exploitation,” *Computing in Science & Engineering*, vol. 15, no. 5, pp. 12–20, 2013.
- [8] J. A. Fails and D. R. Olsen, “Interactive machine learning,” in *Proceedings of the 8th International Conference on Intelligent User Interfaces*, ser. IUI '03. New York, NY, USA: Association for Computing Machinery, 2003, p. 39–45. [En línea]. Disponible en: <https://doi.org/10.1145/604045.604056>
- [9] G. Ramos, C. Meek, P. Simard, J. Suh, and S. Ghorashi, “Interactive machine teaching: a human-centered approach to building machine-learned models,” *Human-*

- Computer Interaction*, vol. 35, no. 5-6, pp. 413–451, 2020. [En línea]. Disponible en: <https://doi.org/10.1080/07370024.2020.1734931>
- [10] L. Jiang, S. Liu, and C. Chen, “Recent research advances on interactive machine learning,” 2018. [En línea]. Disponible en: <https://arxiv.org/abs/1811.04548>
- [11] S. Amershi, M. Cakmak, W. B. Knox, and T. Kulesza, “Power to the people: The role of humans in interactive machine learning,” *AI Magazine*, vol. 35, no. 4, pp. 105–120, Dec. 2014. [En línea]. Disponible en: <https://ojs.aaai.org/index.php/aimagazine/article/view/2513>
- [12] J. J. Dudley and P. O. Kristensson, “A review of user interface design for interactive machine learning,” *ACM Trans. Interact. Intell. Syst.*, vol. 8, no. 2, jun 2018. [En línea]. Disponible en: <https://doi.org/10.1145/3185517>
- [13] M. Kabra, A. Robie, M. Rivera-Alba, S. Branson, and K. Branson, “Jaaba: Interactive machine learning for automatic annotation of animal behavior,” *Nature methods*, vol. 10, 12 2012.
- [14] R. Fiebrink and P. Cook, “The wekinator: A system for real-time, interactive machine learning in music,” *Proceedings of The Eleventh International Society for Music Information Retrieval Conference (ISMIR 2010)*, 01 2010.
- [15] K. Hipke, M. Toomim, R. Fiebrink, and J. Fogarty, “Beatbox: End-user interactive definition and training of recognizers for percussive vocalizations,” in *Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces*, ser. AVI ’14. New York, NY, USA: Association for Computing Machinery, 2014, p. 121–124. [En línea]. Disponible en: <https://doi.org/10.1145/2598153.2598189>
- [16] J. Šavelka, G. Trivedi, and K. D. Ashley, “Applying an interactive machine learning approach to statutory analysis,” in *Legal Knowledge and Information Systems*. IOS Press, 2015, pp. 101–110.
- [17] V. Tolls, “An event-based approach to modeling complex data in critical care,” Ph.D. dissertation, 2018, copyright - Database copyright ProQuest LLC; ProQuest does not claim copyright in the individual underlying works; Descriptor de analito - N.A; Última actualización - 2022-02-28. [En línea]. Disponible en: <https://www.proquest.com/dissertations-theses/event-based-approach-modeling-complex-data/docview/2125429900/se-2>
- [18] A. Holzinger and I. Jurisica, *Knowledge Discovery and Data Mining in Biomedical Informatics: The Future Is in Integrative, Interactive Machine Learning Solutions*. Berlin,

- Heidelberg: Springer Berlin Heidelberg, 2014, pp. 1–18. [En línea]. Disponible en: https://doi.org/10.1007/978-3-662-43968-5_1
- [19] N. Kosmyna, F. Tarpin-Bernard, and B. Rivet, “Adding human learning in brain–computer interfaces (bcis): Towards a practical control modality,” *ACM Trans. Comput.-Hum. Interact.*, vol. 22, no. 3, may 2015. [En línea]. Disponible en: <https://doi.org/10.1145/2723162>
- [20] A. Fadhil, “Towards automatic & personalised mobile health interventions: An interactive machine learning perspective,” 2018. [En línea]. Disponible en: <https://arxiv.org/abs/1803.01842>
- [21] J. Suh, S. Ghorashi, G. Ramos, N.-C. Chen, S. Drucker, J. Verwey, and P. Simard, “Anchorviz: Facilitating semantic data exploration and concept discovery for interactive machine learning,” *ACM Trans. Interact. Intell. Syst.*, vol. 10, no. 1, aug 2019. [En línea]. Disponible en: <https://doi.org/10.1145/3241379>
- [22] B. Kellenberger, D. Tuia, and D. Morris, “Aide: Accelerating image-based ecological surveys with interactive machine learning,” *Methods in Ecology and Evolution*, vol. 11, 11 2020.
- [23] M. Kabra, A. Robie, M. Rivera-Alba, S. Branson, and K. Branson, “Jaaba: Interactive machine learning for automatic annotation of animal behavior,” *Nature methods*, vol. 10, 12 2012.
- [24] D. Angluin, “Learning regular sets from queries and counterexamples,” *Information and Computation*, vol. 75, no. 2, pp. 87–106, 1987. [En línea]. Disponible en: <https://www.sciencedirect.com/science/article/pii/0890540187900526>
- [25] E. Diamant, “Learning to understand image content: Machine learning versus machine teaching alternative,” in *2006 International Conference on Information Technology: Research and Education*, 2006, pp. 26–29.
- [26] X. Zhu, A. Singla, S. Zilles, and A. N. Rafferty, “An overview of machine teaching,” 2018. [En línea]. Disponible en: <https://arxiv.org/abs/1801.05927>
- [27] P. Y. Simard, S. Amershi, D. M. Chickering, A. E. Pelton, S. Ghorashi, C. Meek, G. Ramos, J. Suh, J. Verwey, M. Wang, and J. Wernsing, “Machine teaching: A new paradigm for building machine learning systems,” 2017. [En línea]. Disponible en: <https://arxiv.org/abs/1707.06742>

- [28] A. Sena and M. Howard, “Quantifying teaching behavior in robot learning from demonstration,” *The International Journal of Robotics Research*, vol. 39, no. 1, pp. 54–72, 2020. [En línea]. Disponible en: <https://doi.org/10.1177/0278364919884623>
- [29] D. Weitekamp, E. Harpstead, and K. R. Koedinger, *An Interaction Design for Machine Teaching to Develop AI Tutors*. New York, NY, USA: Association for Computing Machinery, 2020, p. 1–11. [En línea]. Disponible en: <https://doi.org/10.1145/3313831.3376226>
- [30] W. Liu, B. Dai, A. Humayun, C. Tay, C. Yu, L. B. Smith, J. M. Rehg, and L. Song, “Iterative machine teaching,” in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, D. Precup and Y. W. Teh, Eds., vol. 70. PMLR, 06–11 Aug 2017, pp. 2149–2158. [En línea]. Disponible en: <https://proceedings.mlr.press/v70/liu17b.html>
- [31] W. Liu, B. Dai, X. Li, Z. Liu, J. Rehg, and L. Song, “Towards black-box iterative machine teaching,” in *Proceedings of the 35th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, J. Dy and A. Krause, Eds., vol. 80. PMLR, 10–15 Jul 2018, pp. 3141–3149. [En línea]. Disponible en: <https://proceedings.mlr.press/v80/liu18b.html>
- [32] Y. Chen, A. Singla, O. M. Aodha, P. Perona, and Y. Yue, “Understanding the role of adaptivity in machine teaching: The case of version space learners,” in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, ser. NIPS’18. Red Hook, NY, USA: Curran Associates Inc., 2018, p. 1483–1493.
- [33] S. Mei and X. Zhu, “Using machine teaching to identify optimal training-set attacks on machine learners,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 29, no. 1, Feb. 2015. [En línea]. Disponible en: <https://ojs.aaai.org/index.php/AAAI/article/view/9569>
- [34] E. Mosqueira-Rey, D. Alonso-Ríos, and A. Baamonde-Lozano, “Integrating iterative machine teaching and active learning into the machine learning loop,” *Procedia Computer Science*, vol. 192, pp. 553–562, 2021, knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 25th International Conference KES2021. [En línea]. Disponible en: <https://www.sciencedirect.com/science/article/pii/S1877050921015441>
- [35] P. van Allen, “Prototyping ways of prototyping ai,” *Interactions*, vol. 25, no. 6, p. 46–51, oct 2018. [En línea]. Disponible en: <https://doi.org/10.1145/3274566>

- [36] J. L. Elman, "Learning and development in neural networks: the importance of starting small," *Cognition*, vol. 48, no. 1, pp. 71–99, 1993. [En línea]. Disponible en: <https://www.sciencedirect.com/science/article/pii/0010027793900584>
- [37] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *Proceedings of the 26th Annual International Conference on Machine Learning*, ser. ICML '09. New York, NY, USA: Association for Computing Machinery, 2009, p. 41–48. [En línea]. Disponible en: <https://doi.org/10.1145/1553374.1553380>
- [38] X. Wang, Y. Chen, and W. Zhu, "A survey on curriculum learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2021.
- [39] C. Florensa, D. Held, M. Wulfmeier, M. Zhang, and P. Abbeel, "Reverse curriculum generation for reinforcement learning," in *Proceedings of the 1st Annual Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, S. Levine, V. Vanhoucke, and K. Goldberg, Eds., vol. 78. PMLR, 13–15 Nov 2017, pp. 482–495. [En línea]. Disponible en: <https://proceedings.mlr.press/v78/florensa17a.html>
- [40] G. Kumar, G. Foster, C. Cherry, and M. Krikun, "Reinforcement learning based curriculum optimization for neural machine translation," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 2054–2061. [En línea]. Disponible en: <https://aclanthology.org/N19-1208>
- [41] B. Goodman and S. Flaxman, "European union regulations on algorithmic decision-making and a "right to explanation"," *AI Magazine*, vol. 38, no. 3, pp. 50–57, Oct. 2017. [En línea]. Disponible en: <https://ojs.aaai.org/index.php/aimagazine/article/view/2741>
- [42] A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, and F. Herrera, "Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai," *Information Fusion*, vol. 58, pp. 82–115, 2020. [En línea]. Disponible en: <https://www.sciencedirect.com/science/article/pii/S1566253519308103>
- [43] D. Minh, H. Wang, Y. Li, and T. Nguyen, "Explainable artificial intelligence: a comprehensive review," *Artificial Intelligence Review*, vol. 55, 06 2022.
- [44] Y.-P. Tang, G.-X. Li, and S.-J. Huang, "ALiPy: Active learning in python," Nanjing University of Aeronautics and Astronautics, Tech. Rep., Jan. 2019, available as arXiv preprint <https://arxiv.org/abs/1901.03802>. [En línea]. Disponible en: <https://github.com/NUAA-AL/ALiPy>

- [45] T. Danka and P. Horvath, “modAL: A modular active learning framework for Python,” available on arXiv at <https://arxiv.org/abs/1805.00979>. [En línea]. Disponible en: <https://github.com/modAL-python/modAL>
- [46] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, Eds., vol. 27. Curran Associates, Inc., 2014. [En línea]. Disponible en: <https://proceedings.neurips.cc/paper/2014>
- [47] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, “Photo-realistic single image super-resolution using a generative adversarial network,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 105–114.
- [48] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni, “Modeling tabular data using conditional gan,” in *Advances in Neural Information Processing Systems*, 2019.
- [49] N. Patki, R. Wedge, and K. Veeramachaneni, “The synthetic data vault,” in *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, Oct 2016, pp. 399–410.
- [50] J. M. Moreno Planas, A. Sánchez Ortega, J. M. García Bueno, and C. Ramírez Castillejo, *Avances en Cáncer de Páncreas, del laboratorio a la clínica*. UCLM, AECC, 2010.
- [51] K. Tomczak, P. Czerwińska, and M. Wiznerowicz, “The cancer genome atlas (TCGA): an immeasurable source of knowledge,” *Contemporary Oncology*, vol. 19, no. 1A, pp. 68–77, 2015. [En línea]. Disponible en: <http://dx.doi.org/10.5114/wo.2014.47136>
- [52] J. Nielsen, “10 usability heuristics for user interface design,” <https://www.nngroup.com/articles/ten-usability-heuristics/>, 2020, accessed: 2022-07-06.