UNIVERSIDAD DE TALCA
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA CIVIL EN BIOINFORMÁTICA

Análisis de autoreactividad de anticuerpos leucémicos soportado por estrategias de Inteligencia Artificial

Por: Yasna Barrera Saavedra

Julio 2021
Talca, Chile

Profesor Guía: Álvaro Olivera Nappa
Profesor Co-Guía: David Medina Ortiz

CONSTANCIA

La Dirección del Sistema de Bibliotecas a través de su unidad de procesos técnicos certifica que el autor del siguiente trabajo de titulación ha firmado su autorización para la reproducción en forma total o parcial e ilimitada del mismo.

Talca, 2021

A mi familia. Ustedes son mi pilar de apoyo y lo más importante en mi vida.

AGRADECIMIENTOS

El primer refugio que se conoce al momento de la concepción es el vientre de la madre, y tú mamá, serás mi refugio para toda la vida. Muchas gracias por confiar en mi como en tu mano izquierda, y por estar para mi siempre que te necesito. Quisiera agradecer también a mi papá, que jamás a dudado de mis capacidades, espero no decepcionarlos jamás. Gracias a mi hermana Carol y mi hermanito Alex, su amor y preocupación me han entregado razones para seguir cuando pensé que no tenía salida. Estoy inmensamente agradecida de mi familia, por cada sonrisa, por cada abrazo y por cada palabra de aliento. Adicionalmente, quisiera agradecer a Cristofer, por apoyarme durante este proceso, mostrarme nuevas posibilidades y enseñarme nuevas cosas cada día.

Con respecto a este proceso debo agradecer a diversas instituciones y personas, que han hecho posible el desarrollo del trabajo presentado. En primer lugar, agradecer a la Universidad de Magallanes por el financiamiento otorgado para el cumplimiento de esta memoria de título, y particularmente a Marcelo Navarrete, Roberto Uribe Paredes y Jorge Torres Almonacid. Su buena disposición y las facilidades prestadas posibilitaron la finalización de este proceso. Además, agradecer a la Universidad de Chile por el apoyo otorgado, en especial, al grupo de ingeniería de proteínas dirigido por Álvaro Olivera Nappa, quien dirigió este trabajo. Dentro de este grupo, debo agradecer especialmente a David Medina co-tutor, quien siguió, corrigió y apoyó este proceso paso a paso.

Agradecer a la Universidad de Talca, que me abrió las puertas ante esta hermosa carrera. Destacar dentro de esta institución al profesor José Reyes, quien propició los primeros acercamientos al área de minería de datos, y ha guiado de igual forma este trabajo. Agradecer también a la profesora Wendy Gonzalez, cuya pasión por la investigación me enseñó que existen diversas áreas inexploradas aún en la ciencia. Muchas gracias a los profesores Jans Alzate y Julio Caballero quienes me mostraron la importancia de la disciplina y un ojo analítico.

Resumen

Los antígenos son moléculas externas reconocidas por el organismo de variada estructura y naturaleza. El sistema inmune ha desarrollado técnicas de reconocimiento para estos agentes patógenos, representando diferentes mecanismos de defensa contra una posible infección, siendo los anticuerpos los responsables de esta detección. Predecir qué anticuerpo reconocerá a un antígeno, o estimar a nivel cualitativo la intensidad de la interacción que se producirá, es una tarea ardua y compleja, representando un gran desafío en el área inmunológica. Debido a que los antígenos pueden ser distintos tipos de moléculas, y tener procedencia en diferentes patógenos, la forma en la cual un anticuerpo reconoce un conjunto de antígenos con diversas intensidades de interacción, es una pregunta que se ha abordado desde diferentes perspectivas.

Por otra parte, el organismo ha desarrollado estrategias para reconocer moléculas externas de aquellas propias. Esto evita que se genere una respuesta inmune sobre tejidos en el organismo. Las moléculas propias del organismo que desencadenan esta respuesta son denominadas auto antígenos, y al proceso de presentar defensas contra estas moléculas se le denomina auto reactividad. El análisis de auto antígenos es de gran relevancia, tanto para el estudio de enfermedades auto inmunes, como para enfermedades relacionadas a células propias del organismo. En el caso de la leucemia, un tipo de cáncer que afecta a células del tejido sanguíneo, el estudio de la auto reactividad y la interacción entre auto antígenos y anticuerpos es de gran relevancia para el diseño y propuestas que permitan diagnosticar y tratar esta enfermedad.

Gran parte de los estudios de interacción entre auto antígenos y anticuerpos se han realizado utilizando técnicas experimentales. No obstante, diversos enfoques in-silico han sido desarrollados empleando diferentes herramientas computacionales como docking o simulación molecular para cálculos de energía libre y visualización de interacciones. Pese a su gran utilidad, estas técnicas poseen un alto costo asociados a la necesidad de material experimental, necesidad de poseer estructuras definidas o modelos confiables, elevados tiempos de simulación, entre otros. De esta forma la aplicación de técnicas de machine learning y diversos métodos de codificación representan una alternativa potente al problema de reconocimiento de interacción entre proteína-proteína, particularmente, a secuencias de auto

antígenos y anticuerpos de leucemia.

A partir de la información de interacciones entre 45 secuencias de cadena pesada de anticuerpos y cerca de 8000 secuencias de auto antígenos, Se diseñó e implemento un sistema predictivo ensamblado cualitativo del nivel de intensidad de la interacción entre auto antígenos y cadenas pesadas de anticuerpos. Como estrategias de entrenamiento de modelos predictivos, se combinaron variados métodos de representación de proteínas, principalmente Natural Language Processing y propiedades fisicoquímicas, con diferentes algoritmos de aprendizaje supervisado logrando un predictor ensamblado con un rendimiento del 81 % de accuracy. Se aplicaron diferentes estrategias de validación que permiten demostrar la robustez del sistema predictivo propuesto, incluyendo sistemas de validación cruzada y métodos propios basados en estrategias Leave One Antibody Out.

Adicionalmente, se diseñó e implemento un conjunto de colecciones de moléculas inmunológicas integradas en un único sistema de base de datos, al cual acoplado a una estrategia de clasificación filogenética, se diseña e implementa una estrategia de clasificación de secuencias de auto antígenos basado en propiedades descriptivas, funcionales y componentes filogenéticos. La combinación del conjunto de colecciones con el sistema de clasificación, en conjunto con el sistema ensamblado predictivo, facilita el diseño de estrategias de identificación de secuencias auto-antígenos y su evaluación contra anticuerpos leucémicos, brindando los soportes iniciales para herramientas de diseño y descubrimiento de antígenos/anticuerpos que cumplan con características relevantes para el problema de la leucemia, denotando la usabilidad de métodos computacionales en problemas complejos de la ingeniería médica.

Keywords – Auto antígeno, Anticuerpo, Machine learning, Embedding, NLP

Abstract

Antigens are external molecules of varied structures and nature recognized by the body. The immune system has developed recognition techniques for these pathogens, representing different defense mechanisms against possible infection, the antibodies responsible for this detection. Predicting which antibody will recognize an antigen or estimating the intensity of the interaction that will occur at a qualitative level is an arduous and complex task, representing a significant challenge in the immunological area. Because antigens can be different types of molecules and have origins in various pathogens, how an antibody recognizes a set of antigens with varying intensities of interaction is a question that has been approached from different perspectives.

On the other hand, the organism has developed strategies to recognize external molecules on its own. This prevents an immune response from being generated on tissues in the body. The body's own molecules that trigger this response are called self-antigens, and the process of presenting defenses against these molecules is called self-reactivity. The analysis of self-antigens is of great relevance, both for studying autoimmune diseases and for diseases related to the body's own cells. In leukemia, a type of cancer that affects cells of the blood tissue, the study of self-reactivity and the interaction between self-antigens and antibodies is of great relevance for the design of proposals that allow the diagnosis and treatment of this disease.

Much of the interaction studies between self-antigens and antibodies have been carried out using experimental techniques. However, various in-silico approaches have been developed using different computational tools such as docking or molecular simulation for free energy calculations and interactive visualization. Despite their great utility, these techniques have a high cost associated with the need for experimental material, the need to have defined structures or reliable models, high simulation times, among others. In this way, the application of machine learning techniques and various coding methods represent a powerful alternative to the problem of protein-protein interaction recognition, particularly to leukemia self-antigen and antibody sequences.

From the information of interactions between 45 sequences of antibodies heavy chain and about 8000 sequences of self-antigens, a qualitative assembled predictive

system for the level of intensity of the interaction between self-antigens and heavy chains of antibodies was designed and implemented. As predictive model training strategies, various protein representation methods were combined, mainly Natural Language Processing and physicochemical properties, with different supervised learning algorithms, achieving an assembled predictor with a performance of 81 % accuracy. Different validation strategies were applied to demonstrate the robustness of the proposed predictive system, including cross-validation systems and proprietary methods based on Leave One Antibody Out strategies.

Additionally, a set of collections of immunological molecules integrated into a single database system was designed and implemented. Coupled with a phylogenetic classification strategy, a method for classifying self-antigen sequences based on descriptive properties was designed and implemented. This method uses different functional properties and phylogenetic components to estimate the relation of new sequences with the set of self-antigen sequences. The combination of the group of collections with the classification system, in association with the assembled predictive system, facilitates the design of strategies for the identification of self-antigen sequences and their evaluation against leukemic antibodies, providing the initial supports for tools of creation and discovery of antigens/antibodies that meet relevant characteristics for the leukemia problem, denoting the usability of computational methods in complex issues of medical engineering.

Keywords – Self-antigens, Antibody, Machine learning, Embedding, NLP

Índice general

Índice de cuadros

Índice de figuras

Capítulo 1

Introducción

El sistema inmune representa a la defensa que opone el organismo ante diversos patógenos. Dada una posible amenaza, el cuerpo puede generar distintos tipos de respuestas. En primer lugar, se despliegan las defensas del sistema inmune innato, las cuales actúan de forma generalizada. Si el patógeno no ha ingresado con anterioridad al organismo, partes de éste serán extraídas, conocidas como antígenos, y expuestas al sistema inmune adaptativo de forma que se desarrollen defensas específicas y memoria frente a esta infección (Cota and Midwinter, 2015). Las células encargadas de gestionar esta memoria son los linfocitos, divididos en linfocitos B y T. La forma de reconocimiento entre estas células de defensa y el antígeno está mediada por proteínas conocidas como anticuerpos. Por otra parte, el sistema inmune requiere de mecanismos para no reconocer moléculas propias como posibles antígenos, y evitar de esta forma respuestas auto inmunes. Aquellas que son reconocidas y desencadenan una respuesta inmune son denominadas auto antígenos. Esto es eludido mediante la generación de tolerancia. La tolerancia inmunológica corresponde a una inactivación funcional inducida por la molécula que conlleva a la incapacidad de responder ante este antígeno (Gattorno and Martini, 2011). Estrategias como la eliminación de linfocitos con alta reactividad frente a moléculas propias, inactivación de los receptores correspondientes, inducción de la muerte celular, entre otras, permiten mantener la homeostasis del organismo (Durai and Moudgil, 2007). El estudio de auto antígenos y su interacción con anticuerpos, especialmente aquellos relacionados a enfermedades como la leucemia, es de gran interés para el desarrollo de posibles nuevas terapias o vacunas que ayuden a combatir estas afecciones (Zhou et al., 2019). Enfoques tanto experimentales como

computacionales se han utilizado para analizar y comprender estas interacciones y las características de las moléculas participantes.

Los estudios experimentales se han enfocado en determinar por medio de ensayos controlados en laboratorio cuáles moléculas interaccionan, el tamaño y peso de éstas, fuerzas de unión y disociación, y caracterización estructural de estas uniones. Algunos de los ensayos mayormente utilizados se basan en la utilización de sensores o biosensores (Olkhov and Shaw, 2008; Asai et al., 2018; Kamat and Rafique, 2017; Tlili et al., 2005), diversas formas de espectroscopia (Yang et al., 2007; Sulchek et al., 2005; Morfill et al., 2007; Arkan et al., 2015; Stefanescu et al., 2007) y resonancia (Kausaite et al., 2007; Rosen and Anglister, 2009; Endo et al., 2005), tanto de forma separada como conjunta. Otra herramienta ampliamente utilizada para el análisis de estas interacciones, y que permite obtener gran cantidad de datos en un sólo análisis, corresponde a los protein microarrays o protoarrays (Robinson et al., 2002). Dada su capacidad de procesar variadas moléculas al mismo tiempo sigue siendo largamente utilizado en estudios inmunológicos (Luo et al., 2019; Butler et al., 2020; Montesinos-Rongen et al., 2020). Estos enfoques poseen diversas desventajas, como el costo económico asociado a los recursos necesarios para su producción, tiempo requerido para llevarlos a cabo, y posibles errores humanos cometidos en pasos de estos ensayos.

Debido a las desventajas en los procesos experimentales, el creciente volumen de datos y el avance de las ciencias computacionales, se implementaron acercamientos computacionales al estudio de la inmunología. Plataformas de recolección de información, como bases de datos de antígenos o anticuerpos (Ansari et al., 2010; Ferdous and Martin, 2018; Dunbar et al., 2014; Tong et al., 2008; Ansari et al., 2010), han sido implementadas para almacenar la información recopilada por medio de enfoques experimentales. Por otra parte, metodologías computacionales para el desarrollo de modelos estructurales de proteínas y docking molecular (Ambrosetti et al., 2020; Weitzner et al., 2017; Adolf-Bryfogle et al., 2018; Bazzoli et al., 2017; Kuroda and Tsumoto, 2018; Shimba et al., 2016), así como también de simulaciones (Bush and Knotts, 2017; Zhao et al., 2018; Fernandez-Quintero et al., 2020), han sido implementadas para el análisis de moléculas inmunológicas. El uso de estas técnicas ha permitido observar y predecir regiones de interacción entre epítopes y parátopes, al observar la mejor forma en la cual antígenos y anticuerpos pueden interaccionar en conformaciones energéticas teóricamente

favorables para que ésta se produzca. Además, el uso de acercamientos basados en machine learning han ganado importancia para el análisis del sistema inmune, así como para la predicción de características o interacciones de interés (Fontanella et al., 2018; Smith et al., 2019). Incluso, este enfoque ha sido implementado en el estudio de enfermedades de gran relevancia como el cáncer (Chlon, 2017).

Diversas dificultades deben ser superadas al momento de trabajar con información biológica para el desarrollo de modelos predictivos o sistemas de clasificación. Desde la recopilación de un conjunto de datos suficientemente grande como para la generalización de patrones o comportamientos, hasta la necesidad de implementar y desarrollar estrategias computacionales para la representación numérica de las secuencias (Camacho et al., 2018). Remarcablemente, pese a que el costo económico de los experimentos requeridos para desarrollar este tipo de análisis ha disminuido en los últimos años, sigue siendo una dificultad para grupos de investigación sin acceso a estos recursos. Por otra parte, a pesar de que se encuentran disponibles un conjunto de plataformas para acceso a información recopilada desde diversos estudios y laboratorios, estas poseen gran volumen de información privada o no entregan herramientas para descarga. Además, la información inmunológica se encuentra dispersa en diversas plataformas, lo cual implica alto costo en tiempo para la recopilación y procesamiento de la información.

Por su parte, diversas estrategias se han desarrollado para la representación de secuencias de proteínas como vectores numéricos. A pesar de esto, la representación de interacciones proteína-proteína sigue siendo un desafío. Dentro de los enfoques clásicos de codificación se encuentran las estrategias de One Hot Encoder. Sin embargo, su uso representa la generación de vectores altamente dimensionales. Otra estrategia se basa en el uso de propiedades fisicoquímicas. No obstante, qué propiedades y cómo utilizarlas sigue siendo un problema sin resolver. Estudios recientes se han centrado en la representación de las proteínas en el espacio de las frecuencias combinando propiedades fisicoquímicas con técnicas de Digital Signal Processing, las cuales han sido aplicadas con éxito en diferentes campos de aplicación Medina-Ortiz et al. (2020a). Pero, no existen protocolos claros para emplearlas en sistemas de interacción proteína-proteína o proteína-ligando. Por otro lado, se han diseñado e implementado métodos conocidos como Autoencoders, los cuales se enfocan en el desarrollo y entrenamiento de sistemas predictivos basados en técnicas de Natural Language Processing enfocados en aprender a

codificar y decodificar secuencias (Asgari et al., 2019). Otra estrategia de interés, se basa en la representación de proteínas como estructuras de grafos. Sin embargo se limita a ser necesaria la estructura y no existe un consenso de qué estrategias emplear para conectar los diferentes nodos (Gaudelet et al., 2020). No obstante, pese a los diferentes avances en métodos comentados, la representación numérica de interacciones proteína-proteína sigue generando un problema complejo de abordar, siendo el foco de este trabajo de memoria de título.

Los relevantes avances realizados en el desarrollo de estrategias de codificación, y en la implementación de algoritmos de machine learning en problemas biológicos manifiestan una estrategia interesante en el análisis de información inmunológica de interés. La interacción de auto antígenos y anticuerpos asociados a leucemia corresponde a uno de los tantos posibles campos en los cuales estas estrategias podrían ser utilizadas, y que aún no son exploradas. Remarcablemente, no se logró identificar estudios comparativos entre los diferentes tipos de codificación o estrategias de representación de proteínas que permiten producir modelos robustos, ni los algoritmos apropiados para llevar a cabo la tarea de predecir la clase de interacción entre auto antígenos y anticuerpos de leucemia, denotando claramente un campo poco desarrollado y que requiere un foco de desarrollo mayor. Dado esto, el diseño un sistema predictivo a partir de las secuencias de anticuerpos y auto antígenos relacionados a esta enfermedad es realmente interesante, en especial en estos días donde el desarrollo de vacunas y tratamientos inmunológicos dirigidos han denotado una demanda emergente de este tipo de metodologías y estrategias, con el fin de optimizar su producción. Por otra parte, la recopilación de información relacionada a anticuerpos, antígenos y epitopes en una sola plataforma facilitaría el desarrollo de metodologías basadas en machine learning, disponiendo la información necesaria en un solo recurso unificado. La aplicación de enfoques en machine learning permitirían observar relaciones y patrones dentro de la información proporcionadas por estas moléculas, así como la predicción de regiones o interacciones de interés, facilitando la comprensión e identificación de grupos o características relevantes a nivel biológico, o el diseño de nuevas estrategias de tratamiento para enfermedades como la leucemia.

El análisis de la interacción entre auto antígenos y anticuerpos en enfermedades como la leucemia representan un enfoque importante en la comprensión y postulación de nuevos mecanismos para el reconocimiento de esta enfermedad y,

idealmente, lograr que las defensas inmunológicas presentadas por el cuerpo detuviesen y eliminasen a las células cancerígenas. Debido a esto, tanto el reconocimiento de nuevas secuencias auto antigénicas relacionadas a esta enfermedad, como la predicción de posibles nuevas interacciones entre auto antígenos y anticuerpos de interés representan una tarea relevante en el estudio de la leucemia. La exploración de diversas metodologías de codificación de secuencias que permitan representar de manera eficiente la interacción auto antígeno- anticuerpo, y la evaluación de diversos algoritmos de machine learning orientados a la predicción de la clase de esta interacción, podrían presentar un acercamiento interesante y racional para aportar con nuevas soluciones a la problemática señalada.

1.1. Marco teórico

Todas las temáticas y propuestas formuladas a lo largo de este trabajo de memoria de título tienen como punto central el estudio de la respuesta inmune en pacientes con leucemia, en especial, de la interacción entre auto antígenos y anticuerpos. Es por esto que es importante, en primer lugar, conocer de mejor forma los términos asociados a antígenos y anticuerpos, a los mecanismos inherentes a su interacción, y las características relevantes sobre estos.

En las siguientes secciones se presentan variados términos y conceptos importantes para comprender el contexto de este proyecto, las bases biológicas detrás del trabajo realizado, y la importancia del análisis de la respuesta inmune.

1.1.1. Leucemia.

La enfermedad conocida como leucemia corresponde a un tipo de cáncer que afecta a las células sanguíneas, incluyendo la medula ósea y el sistema linfático. De forma similar a otros tipos de cáncer, mutaciones afectan el ciclo de reproducción y diferenciación, aumentando o disminuyendo el número de cierto tipo de células en el organismo, y afectando el funcionamiento de este sistema. Dependiendo del tipo de tejido afectado, la leucemia puede dividirse en dos categorías, leucemia mielogénica y leucemia linfocítica (Lewis, 1957). En el primer caso, se ven afectadas células de tipo mieloide, las cuales dan origen a células sanguíneas como glóbulos rojos o plaquetas, Así mismo, se pueden ver afectadas células de defensa no específica, como blastocitos o neutrófilos (Sabath, 2013). En el caso de la leucemia

linfocitica, se ven afectadas células del sistema linfático, específicamente, a los linfocitos como células B y T (Kelley and Patel, 2018). Además, los tipos de leucemia pueden ser clasificados dependiendo de la agresividad de la enfermedad, distinguiéndose las de tipo aguda y crónica. En el caso de las leucemias agudas, esta enfermedad evoluciona rápidamente, con una acelerada multiplicación de las células afectadas, y puede ser mortal a corto plazo en falta de un tratamiento adecuado (Sabath, 2013). Por el contrario, en el caso de leucemias crónicas, la enfermedad se desarrolla lentamente y pueden pasar años antes de ser detectada y recibir un tratamiento (Sabath, 2013).

Así, se distinguen al menos cuatro tipos de leucemias, leucemia aguda linfocitica (ALL) (Pui et al., 2004), leucemia crónica linfocítica (CLL) (Chiorazzi et al., 2021), leucemia aguda mieloide (AML)(Döhner et al., 2015) y leucemia crónica mieloide (CML) (Jabbour and Kantarjian, 2018). El diagnostico de estas enfermedades puede ser mediante inmunofenotipificación (Herold and Mitra, 2020), citometría de flujo (McKinnon, 2018), citogenética (Lawce and Brown, 2017) o secuenciación (Pellegrino et al., 2018). Mediante estas tecnologías se puede observar la forma de las células sanguíneas, mutaciones o conteo del número de estas células en una muestra sanguínea. De forma similar, es posible distinguir auto antígenos que prevengan la eliminación de estas células cancerígenas del organismo por medio del sistema inmune (Fraietta et al., 2018).

1.1.2. Antígenos y anticuerpos

El sistema inmune corresponde a un complejo conjunto de diversas células que interactúan entre ellas, ya sea de forma directa o por medio de cascadas de reacción y mediadores. Estos componentes generan una respuesta defensiva del organismo contra agentes externos, siendo ejemplos característicos patógenos, bacterias, virus, órganos, tejidos, entre otros. Otra característica del sistema inmune es poseer la capacidad de mantener un correcto control de la proliferación celular en el cuerpo(Castelo-Branco and Soveral, 2014).

El mecanismo de respuesta inmune puede ser dividido en dos categorías, componentes innatos y adaptativos. La respuesta innata incluye barreras físicas, mecánicas y bioquímicas, así como también, respuestas celulares no especificas asociadas a enzimas y proteínas que limitan el crecimiento de patógenos. Ejemplos característicos de este tipo de proteínas son lactoferrina, interferones y lisozimas.

Relacionadas al tipo de respuesta innata se encuentran células como neutrófilos, macrófagos y células asesinas (naturales y activadas por linfocinas). La respuesta especifica o adaptativa posee una gran especificidad, diversidad, memoria y tolerancia ante los elementos propios (Connect, 2020). Esta respuesta está mediada por células conocidas como linfocitos y sus productos. Los linfocitos corresponden a un tipo de célula blanca, las cuales circulan en la sangre. Su principal función es la regulación de la respuesta inmunitaria. A su vez, logran una diferenciación hacia células efectoras conocidas como células B y células T (Mastache et al., 2005).

El reconocimiento de patógenos externos es llevado a cabo por los linfocitos B, y se basa en la interacción generada por los receptores de membrana con elementos patógenos (Abbas et al., 2019). Este receptor corresponde a una inmunoglobulina M, la cual está compuesta por dos cadenas pesadas y dos cadenas ligeras idénticas (H y L), quienes se encuentran unidas por medio de puentes disulfuros. La unión de las cadenas pesadas y ligeras confieren una estructura similar a una ” Y ” al anticuerpo. Además, se distingue una región bisagra, correspondiente a una porción de la cadena pesada entre el primer y el segundo dominio. Este sector posee un alto contenido de prolinas y cisteínas, lo cual le confiere la flexibilidad suficiente para que la región variable pueda rotar e interaccionar con distintos antígenos(Adlersberg, 1976). La estructura del anticuerpo se representa en la Figura 1.1.1

En cada anticuerpo, en el extremo N terminal de las cadenas pesadas y ligeras, existe una sección de 50 a 70 residuos, que corresponden a la región de unión a elementos foráneos. Dentro de esta región, se encuentran porciones específicas de 15 a 20 residuos, en ocasiones superpuestos, que son los encargados directos de esta interacción. A esta porción se le denomina parátope. Su capacidad de interacción con patógenas está dada por la complementariedad fisicoquímica y estérica que fragmentos de este patógeno pueda tener con los aminoácidos que componen a la región parátope.

Los antígenos por su parte pueden ser definidos como cualquier sustancia o molécula patógena que desencadena una respuesta inmune. Corresponden a moléculas que pueden ser reconocidas por anticuerpos, por medio de la interacción entre los residuos del que conforman la región parátope del anticuerpo con los elementos del antígeno. Con frecuencia, las moléculas presentadas a células B y reconocidas por

anticuerpos corresponden a proteínas presentes en los agentes externos. La zona en la cual se produce la interacción con un anticuerpo se denomina región epítope. Dentro de un antígeno es posible encontrar diversos epítopes, y, por lo tanto, pueden interaccionar con distintos anticuerpos. Dependiendo de la estructura que presente la región epítope, estos pueden ser definidos como continuos o discontinuos. En el primer caso, los aminoácidos que componen la región epítope se encuentran de forma lineal en la secuencia, mientras los epítopes discontinuos estos aminoácidos no son contiguos en secuencia, pero si se encuentran cercanos al conformar la estructura (Yasser et al., 2017). Debido a que cada anticuerpo presenta diferentes parátopes capaces de unirse a epítopes pertenecientes a un antígeno u otro, un anticuerpo nunca es mono especifico, es decir, no presenta especificidad por un sólo antígeno (Van Regenmortel, 2019). Landsteiner (2013) plantea que la especificidad de un anticuerpo por un antígeno ha dejado de ser visto como un fenómeno totalitario y se ha interpretado en función de la intensidad con la cual interaccionan, dependiendo de la complementariedad química y efectos estéricos.

Todo anticuerpo posee una región hiper variable, la cual es la responsable de reconocer de diversos tipos de antígenos por medio de reconocimiento estructural de las secuencias presentadas(Wang et al., 2007). Aún sin el estímulo de antígenos, el cuerpo humano puede llegar a producir alrededor de 10^{12} anticuerpos diferentes, conocido como un pre-repertorio de anticuerpos para el sistema inmune. De esta forma, teóricamente el sistema inmune permite contar con un sitio de reconocimiento para cualquier determinante antigénico, aunque sea por medio de una interacción débil (Bruce Alberts and Walter, 2002). Este repertorio primario o pre-repertorio de anticuerpos se forma mediante diversas combinaciones de una variedad de genes que codifican para la cadena liviana y pesada. Se han identificado seis segmentos de genes en cada célula, dos para cadena pesada y cuatro para cadenas livianas. El proceso de selección y regulación de la combinación de genes a utilizar es denominado recombinación V(D)J (Bruce Alberts and Walter, 2002).

Los exones que codifican los dominios de unión a antígeno se ensamblan a partir de los denominados segmentos génicos V (variable), D (diversos) y J (unión) mediante reordenamientos de ADN de ”cortar y pegar ”. Este proceso elige un par de segmentos, introduce roturas en la cadena doble adyacentes a cada segmento, elimina (o, en casos seleccionados, invierte) el ADN que se encuentra en esta

sección y liga los segmentos juntos (Roth, 2015). Un esquema de este proceso se puede apreciar en la Figura 1.1.2.

Figura 1.1.1: Esquema representativo de un anticuerpo. En color azul se muestran las cadenas pesadas y en verde las cadenas ligeras. Los bloques en colores claros corresponden a las regiones hiper variables en ambas cadenas. Los bordes irregulares en los extremos de las cadenas ligeras y pesadas corresponden a sitios de unión de antígenos. Estas cadenas se unen por medio de puentes di sulfuros, que mantiene unidas a las cadenas pesadas con ligeras y cadenas pesadas entre sí.

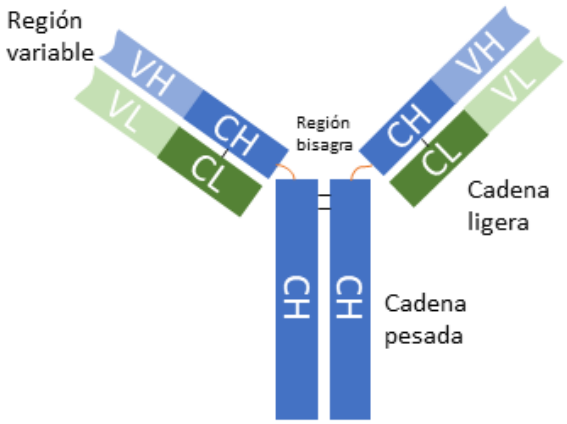
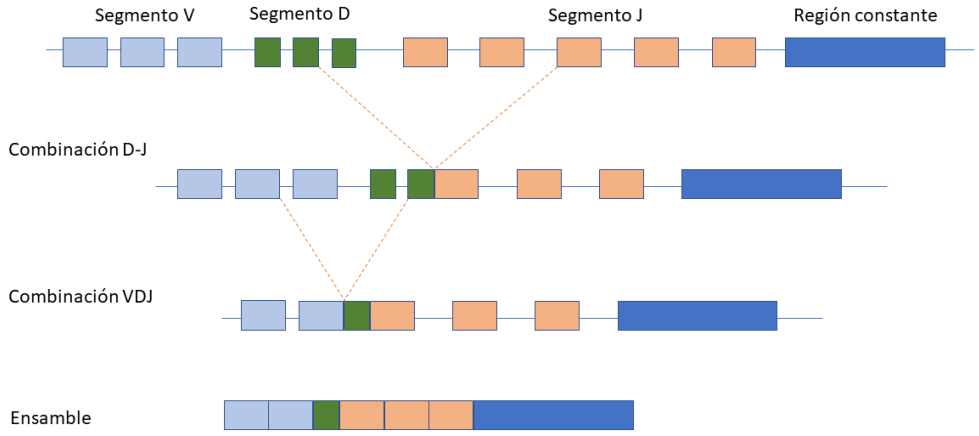


Figura 1.1.2: Esquema de la recombinación VDJ. En este proceso, los genes que codifican para cadenas pesadas o livianas pasan por un proceso de recombinación. En este, distintos segmentos de genes que componen a cada cadena son seleccionados para formar un nuevo anticuerpo. Dada las regiones de variedad (V), diversidad (D) y unión (J, en ingles join), se seleccionan puntos de corte y ligación, el cual permite realizar distintas combinaciones génicas. En primer lugar, se realiza un corte entre exones del segmento D y del segmento J. Se unen los extremos de estos exones y se realiza un corte entre exones del segmento V y D. Luego estos extremos son ligados y se produce la eliminación de intrones o regiones no codificantes, permitiendo formar una nueva versión del gen que codificara alguna cadena.



Al interaccionar varias veces con el antígeno, las células B permiten la especialización del anticuerpo. Esto se realiza modificando al anticuerpo para aumentar la afinidad en la unión al antígeno, proceso conocido como maduración de afinidad(Bruce Alberts and Walter, 2002).Esta especialización se realiza mediante mutaciones puntuales somáticas en la región hiper variable, las cuales se realizan a una tasa de aproximadamente 10^4 – 10^3 pares de bases por generación mediante la división por mitosis. Muchas de estas células B generadas por el proceso anterior, al no producir una interacción con el antígeno, son eliminadas del sistema mediante apoptosis(Van Regenmortel, 2019). Las células B con mayor afinidad por el antígeno proliferan y se diferencian en células plasmáticas productoras de anticuerpos o células B de memoria.

1.1.3. Auto antígenos y auto reactividad.

El sistema inmune requiere de mecanismos para distinguir moléculas secretadas por el mismo organismo de aquellas que ingresan mediante patógenos, o se expresan por enfermedades. La auto tolerancia corresponde a la capacidad de reconocer las moléculas del organismo como propias, y no levantar respuestas inmunes ante su reconocimiento. Por otra parte, la auto reactividad corresponde al reconocimiento de moléculas propias como foráneas, lo cual produce una respuesta inmune negativa (Sakaguchi et al., 2012). Estas moléculas propias del organismo que provocan una respuesta inmune son conocidas como auto antígenos. Existen dos principales mecanismos para la prevención de la auto reactividad, uno denominado recesivo y uno dominante (Sakaguchi et al., 2008).

Los mecanismos de control recesivo utilizan control celular para evitar auto reactividad. Una de las estrategias utilizadas es la apoptosis de células inmaduras linfocíticas, ante la presentación de auto antígenos en su etapa de desarrollo. Otra medida presentada por el sistema inmune es la desactivación de receptores que reconozcan moléculas auto inmunes en células B y T, los cuales son reemplazados, proceso llamado edición de receptores (Schwartz, 2005). Así mismo, aquellas células que hayan pasado estos mecanismos de control y madurado pueden ser posteriormente desactivadas ante la exposición a auto antígenos, o elevar sus umbrales de activación mediante la exposición de receptores inhibidores (Sakaguchi et al., 2008).

Los mecanismos de control dominante se basan en la producción de ciertas células T, denominadas células T reguladoras (Treg), las cuales poseen la tarea de vigilar y controlar a linfocitos anormales. Estas células expresan diversos factores, entre estos el factor de transcripción Foxp3, y poseen la capacidad de suprimir células T auto reactivas (Sakaguchi et al., 2008). Las células Treg suprimen la proliferación, activación y producción de citocinas de las células T CD8 +, las células B, las células presentadoras de antígenos y las células asesinas naturales (Dhar and Dyckman, 2017). Otros mecanismos de control observados corresponden a la supresión por citolisis, mediante la expresión de granzima A, lo cual conduce a la célula diana a la muerte (Vignali et al., 2008). Además, se han descrito mecanismos relacionados a la interrupción del metabolismo de las células dianas, lo cual permite controlar su proliferación, mediados por diversas enzimas y mensajeros (Vignali

et al., 2008).

Finalmente, aún cuando la labor de este mecanismo de este mecanismo es de vital importancia para el mantenimiento de la homeostasis en el organismo, también puede generar dificultades. Su mal funcionamiento se relaciona con el desarrollo de enfermedades auto inmunes, al no controlar de manera correcta estas células B y T auto reactivas. Por otra parte, su buen funcionamiento puede provocar la supervivencia de células tumorales, al evitar que el mismo sistema inmune reconozca e induzca las muerte de estas células patológicas (Mougiakakos et al., 2010). Esto se debe a que muchos de los tumores expresan antígenos similares a auto antígenos, por lo cual, el sistema inmune reconoce a estas células como propias e impide su control. Es por esto que este mecanismo de control, y los auto antígenos asociados a células cancerígenas, son blancos de estudio para el desarrollo de inmunoterapias contra el cáncer (Takeuchi and Nishikawa, 2016).

1.2. Hipótesis

La aplicación de diferentes estrategias de representación de secuencias de proteínas combinadas con el diseño e implementación de modelos ensamblados predictivos facilita la predicción cualitativa de los niveles de intensidad de interacción entre cadenas pesadas de anticuerpos y secuencias de autoantígenos, permitiendo su estudio, caracterización y reconocimiento de patrones como soporte a estrategias de diseño de anticuerpos.

1.3. Objetivo general

Diseñar, implementar y validar estrategias computacionales para el estudio, caracterización, reconocimiento y análisis de autoreactividad de anticuerpos leucémicos y su aplicación a reconocimiento de secuencias de auto antígenos.

1.3.1. Objetivos específicos.

1. Diseñar, implementar y validar diferentes estrategias de codificación y representación de la interacción entre la cadena pesada del anticuerpo y el auto antígeno .

2. Diseñar, implementar y validar métodos computacionales basados en estrategias de aprendizaje ensamblado para predecir cualitativamente el nivel de interacción entre cadena pesada de anticuerpo y auto antígeno.
3. Diseñar, implementar y consolidar base de datos de moléculas inmunológicas como soporte y validación de los métodos computacionales predictivos planteados en este trabajo de memoria de título.
4. Diseñar, implementar y validar métodos computacionales basados en estrategias filogenéticas, caracterización de propiedades y métodos estadísticos para clasificar nuevas secuencias candidatas a auto antígenos.

La hipótesis planteada y los objetivos específicos señalados serán abordados a lo largo de este documento en secciones separadas. En primer lugar, dentro del capitulo 2, se evalúan y realizan tareas asociadas al objetivo específico 1 y 2, enfocadas en la exploración de estrategias de representación de la interacción auto antígenos y anticuerpos, y indagación sobre algoritmos de machine learning para el desarrollo de modelos predictivos cualitativos de la intensidad de interacción. Además, dentro de este capitulo se presenta marco teórico asociado a la comprensión de la información abordada para el desarrollo del sistema ensamblado predictivo propuesto. Luego, dentro del capitulo 3 se presenta la información recopilada y tareas ejecutadas para el cumplimiento del objetivo específico 3. Finalmente, en el capitulo 4 se presenta la información y resultados parciales obtenidos referidos al objetivo específico 4.

Esta estructura debe ser considerada como una historia. En primera instancia se pensó en el desarrollo de un modelo predictivo entrenado sobre un conjunto de datos experimentalmente determinado, pero que es limitado. Luego, se planteó la recopilación de datos públicos que permitiesen evaluar este modelo ensamblado desarrollado, y sortear algunas de las dificultades asociadas a su recolección. Posteriormente, la falta de descripción en la información recopilada y la incapacidad de distinguir datos útiles de ruido con el cual testear al sistema predictivo desarrollado, motivó el planteamiento de un sistema de clasificación de auto antígenos, que sirviese de filtro para posibles secuencias a evaluar.

Capítulo 2

Modelamiento predictivo aplicado a sistemas de interacción antígeno anticuerpo

La aplicación de diversas áreas y herramientas computacionales en el estudio de datos biológicos es una disciplina comúnmente utilizada hoy en día. Desde el uso de tecnologías para el almacenamiento y organización de grandes volúmenes de datos generados de forma experimental, hasta el desarrollo de metodologías para el análisis y búsqueda de patrones en los datos generados. De esta forma, diversas herramientas de análisis de datos biológicos se han establecido y han permitido el avance de la ciencia en variadas áreas (Sayers et al., 2019; Thomas and Thivarkaran, 2020; Wang et al., 2021). Una de las herramientas computacionales que ha permitido un gran avance en el estudio y predicción de diversas características es el uso de minería de datos y machine learning. Haciendo uso de los datos biológicos de diverso tipo y recopilados en distintos estudios, es posible generar un modelo que permita clasificar o predecir un comportamiento en una molécula o tipo de moléculas de interés. De esta forma, ha sido posible desarrollar distintos avances en el estudio de enfermedades, como el cáncer (Amrane et al., 2018), el Alzheimer (Castellazzi et al., 2020) o diabetes (Kaur and Kumari, 2020), entre algunos ejemplos.

Uno de los campos de estudio con gran relevancia es la inmunología. La capacidad de un organismo de defenderse frente agentes extraños o patógenos permite

mantener a la especies vivas o en buen estado. Comprender los mecanismos por los cuales un organismo puede detectar y defenderse de estos elementos extraños requiere de diversos puntos de vistas y metodologías, tanto experimentales como computacionales. Grandes avances se han realizado en este ámbito, permitiendo la visualización de la interacción entre las moléculas involucradas, conocer sus secuencias y genes relacionados, e incluso medir la energía involucrada en esta interacción. Todo este conocimiento ha posibilitado el desarrollo de nuevas herramientas y metodologías, para examinar en mayor detalle las características de este mecanismo de defensa, y diseñar nuevas moléculas para el tratamiento o identificación de diversas enfermedades. Dentro de esta área se ha utilizado de igual forma la minería de datos y el machine learning, en diversas situaciones o para responder a diversas interrogantes, como la clasificación e identificación de moléculas relacionadas a enfermedades (Figgett et al., 2019; Fousteri et al., 2021), predicción de características o regiones de interés (Smith et al., 2019; Saha et al., 2005), diseño de herramientas terapéuticas (Ong et al., 2020; Wec et al., 2021), entre otros (McGowan et al., 2021; Akbar et al., 2019; Jespersen et al., 2019; Faissol, 2019). Muchos de estos algoritmos se basan en arquitecturas de neural networks, un modelo de inteligencia artificial que permite realizar inferencias, reportando casos exitosos con buenas medidas de desempeño(Nosrati et al., 2020; Collatz et al., 2020; Liberis et al., 2018; Akbar et al., 2019). Este tipo de algoritmo de machine learning permite agregar diversos niveles de abstracción al usar más capas en su arquitectura, generando un enfoque basado en Deep learning(Schmidhuber, 2015).

Es necesario aplicar distintas técnicas para la transformación de datos de forma que estos sean comprensibles para los algoritmos de machine learning. Una de estas corresponde a la utilización de encoders para la transformación de estos datos a vectores numéricos procesables por un computador. Los encoders corresponden a algoritmos o técnicas de codificación que transforman datos de un tipo de representación a otro. Diversos tipos de encoders pueden ser usados, dependiendo del objetivo, tipo de datos y resultado al cual se desee apuntar. Es posible también desarrollar autoencoders para adaptar de forma más especifica la transformación de datos.Un autoencoder corresponde a un modelo basado en neural networks que, mediante el entrenamiento no supervisado, aprende reglas de codificación de forma automática para la transformación de datos. La combinación de estas técnicas

con embedings permiten trabajar sobre un espacio más adecuado para algoritmos de machine learning. Embedding es una técnica que permite traspasar un vector numérico a un espacio de menor tamaño, mientras que en el contexto de neural networks mapea vectores de datos categóricos a vectores continuos en espacios dimensionales mas pequeños. Otro posible acercamiento puede ser la utilización de algoritmos de natural language processing, de forma que una secuencia pueda mantener una conexión estructural que sea informativa para los algoritmos de machine learning(Asgari et al., 2019).

La predicción de interacción entre un antígeno y un anticuerpo sigue siendo un área amplia y desafiante de investigación. Diversos enfoques computacionales se han adoptado para la realización de esta tarea. Sin embargo, no todos los posibles métodos han sido explorados. La utilización de embeddings y algoritmos de minería de datos, además de metodologías de encoding que permitan conservar relaciones estructurales, entregan una fuerte base para la exploración en sistemas de predicción basados en estas técnicas. Debido a la ausencia de estudios comparativos con respecto a las estrategias de codificación adecuadas para el desarrollo de modelos predictivos sobre esta interacción, o artículos referentes a los mejores algoritmos para desarrollar esta tarea de manera satisfactoria, se deduce que aun falta investigación sobre esta área.

2.1. Marco teórico.

Con el fin de comprender las diversas estrategias y metodologías que han sido desarrolladas en esta área de investigación, es necesario exponer su naturaleza y abordar diferentes conceptos y terminologías. De esta forma sustentar las metodologías, dar sentido a los resultados y permitir la justificación de las diferentes discusiones a lo largo del documento. A continuación, se hace un resumen de los principales conceptos, sus características y propiedades, las cuales serán explicadas en las siguientes secciones.

2.1.1. Data mining

Data mining es una de las técnicas y estrategias metodológicas más utilizadas en la actualidad, tiene como objetivo la aplicación de métodos matemáticos y estadísticos para la identificación de patrones en conjuntos de datos de interés. Ha

sido utilizada en diferentes campos de investigación, tales como ciencias biológicas, economía, ciencias sociales o marketing. Adicionalmente, es posible considerar al data mining como una evolución lógica del manejo, colección, almacenamiento y análisis de datos, siendo un ejemplo claro, el uso de data warehousing (Han et al., 2011).

Uno de los conceptos o metodologías estrechamente relacionadas con el data mining corresponden al proceso KDD (Knowledge discovery from data), el cual es el proceso de descubrir patrones y generar información desde grandes volúmenes de datos. No obstante, existen diversas opiniones, quienes consideran que data mining corresponde sólo a un paso dentro del proceso KDD (Han et al., 2011). La principal distinción de data mining con respecto a otras técnicas que buscan de igual forma encontrar un sentido o conocimiento dentro de los datos, está dada por el hecho de que la información obtenida no es entregada en principio por un modelo, sino que a través de los datos(Cios et al., 2007).

Para cumplir con el objetivo de la extracción de información y la identificación patrones, se han desarrollado distintos acercamientos computacionales, a través del diseño e implementación de algoritmos. Los algoritmos se definen como el conjunto de sentencias lógicas en un orden específico, enfocados en el cumplimiento de una meta en particular. Normalmente, para cumplir dichas actividades, se emplean diferentes heurísticas y cálculos que permiten construir un modelo o descubrir un patrón a partir de los datos proporcionados. De manera general, la heurística corresponde a un conjunto de técnicas, estrategias y criterios establecidos con la finalidad de lograr resolver un problema. Ejemplos de algoritmos o técnicas empleadas en el descubrimiento de patrones corresponden a análisis estadístico por medio de correlaciones para la identificación de relaciones entre variables(Fujioka and Iwai, 1997; Richard, 2007), comparación de distribuciones por medios de test de hipótesis(Banerjee et al., 2009), aplicaciones de heurísticas para la optimización de problemas por medio de técnicas como Simulated Annealing(Dowsland and Thompson, 2012), empleo de algoritmos genéticos (Kramer, 2017), búsqueda de patrones mediante algoritmos de clustering (Berkhin, 2006). entre las principales.

2.1.2. Machine learning

Machine learning corresponde a una rama de las ciencias computacionales, la cual busca por medio de diversos métodos o algoritmos que las máquinas aprendan.

Este proceso de aprendizaje se produce por medio de la disposición de conjuntos de datos con elementos etiquetados, es decir, debe existir una respuesta o variable de interés a predecir. A partir de dicha respuesta, los algoritmos en base a las características que describen los ejemplos son entrenados para " aprender ", condicionados a su vez, por la configuración del algoritmo a través de sus hiper parámetros.

Existen diversos tipos de algoritmos de aprendizajes, siendo los más reconocidos supervised learning, unsupervised learning, semi supervised learning, reinforcement learning, y self-learning (Ayodele, 2010). A continuación, se explicarán de manera simples los tipos de aprendizaje más utilizados, los cuales serán empleados en esta memoria de título, haciendo hincapié en los algoritmos de aprendizaje supervisado y no supervisado (supervised learning y unsupervised learning).

2.1.2.1. Supervised learning

En este tipo de aprendizaje, la data entregada a los algoritmos posee resultados u outputs conocidos, es decir, existe una variable de interés a aprender a predecir. De esta forma, se tiene que la meta principal de los algoritmos existentes en esta categorización consiste en comprender y generalizar los comportamientos correlacionados a la variable respuesta. De tal manera, que, dada estas generalizaciones, nuevos ejemplos puedan ser predichos con respecto a la respuesta de interés.

Es posible subdividir los algoritmos de aprendizaje supervisado con respecto al tipo de respuesta que se desea predecir. En los casos en los que la respuesta sea del tipo categórico, corresponden a métodos de clasificación. Para el caso en el que la respuesta es del tipo continuo, se asocian a métodos de regresión (Sarkar et al., 2018). Diferentes algoritmos de aprendizaje supervisado han sido propuestos, cada uno con diferentes enfoques y características. Pero, con el mismo objetivo en común. A continuación, de manera simple, se exponen algunos de los algoritmos más utilizados en diferentes campos de investigación, siendo los más icónicos hasta la fecha.

2.1.2.1.1. K-Nearest Neighbors (KNN) Algoritmo de aprendizaje supervisado, el cual tiene por objetivo asociar un elemento a una clase en particular, dada la información de ejemplos de entrada que tengan asociadas características

determinadas, que puedan declararse vecinos del nuevo ejemplo a clasificar, siendo k el número de vecinos que se está dispuesto a utilizar para aplicar la clasificación. La mejor elección de k depende fundamentalmente de los datos; generalmente, valores grandes de k reducen el efecto de ruido en la clasificación. Sin embargo, crean límites entre clases parecidas. Con el fin de evaluar la cercanía de los ejemplos existentes contra el nuevo ejemplo a clasificar, es necesario asociar ciertas medidas de distancia que permitan cuantificar esta característica, para así, poder comparar esta distancia y evaluar la cercanía para asociarle una clase a este nuevo ejemplo. La distancia que emplear para evaluar la cercanía puede ser: Euclidiana, Manhattan, coseno o Mahalanobis, entre las principales.

2.1.2.1.2. Decision Tree Se define árbol de decisión como un modelo de predicción, utilizado en el ámbito de la inteligencia artificial, en el cual, dado un conjunto de datos, se fabrican diagramas de construcciones lógicas, muy similares a los sistemas de predicción basados en reglas, que sirven para representar y categorizar una serie de condiciones que ocurren de forma sucesiva, para la resolución de un problema.

El aprendizaje basado en árboles de decisión utiliza un árbol como un modelo predictivo que mapea las observaciones de las características que presenta un elemento. En estas estructuras de árbol, las hojas representan etiquetas de conjuntos ya clasificados, los nodos, a su vez, nombres o identificadores de los atributos y las ramas representan posibles valores para dichos atributos.

Este tipo de entrenamiento es uno de los más utilizados, debido a su simplicidad a la forma en la que trabaja, ya que, permite comprender del problema, con respecto a los atributos y cómo estos van distribuyendo las respuestas, así, es posible entender las decisiones que toma el algoritmo para clasificar o predecir nuevos ejemplos, determinar comportamientos preferentes y tendencias sobre atributos y rangos de estos.

2.1.2.1.3. Naive Bayes Naive Bayes es un conjunto de algoritmos de aprendizaje supervisados basados en la aplicación del teorema de Bayes con la suposición “ ingenua ” de independencia entre cada par de características. A pesar de sus supuestos aparentemente simplificados, los clasificadores de Naive Bayes han funcionado bastante bien en muchas situaciones del mundo real, la

famosa clasificación de documentos y el filtrado de correo no deseado son ejemplos de ello. Requieren una pequeña cantidad de datos de entrenamiento para estimar los parámetros necesarios. Pueden ser extremadamente rápido en comparación con métodos más sofisticados. Existen distintos tipos de clasificadores de Naive Bayes, diferenciándose entre sí en la función de distribución de probabilidad que utilizan, dentro de los que se encuentran: Gaussian Naive Bayes, Multinomial Naive Bayes y Bernoulli Naive Bayes.

2.1.2.1.4. Métodos de ensamble Los métodos de ensamble se basan en la combinación de las predicciones obtenidas por varios estimadores, construidos en base a algoritmos de aprendizaje supervisado, con el fin de mejorar la generalización del modelo y aumentar la robustez ante nuevos ejemplos. Existen dos familias de métodos de ensamble, las cuales se diferencian principalmente en la forma en que combinan los modelos para obtener la medida de desempeño final:

- **Métodos ponderados**: basados en la construcción de varios estimadores independientes y promediar sus medidas de desempeño, esto mejora el rendimiento debido a que disminuye la variabilidad de las clasificaciones. Ejemplos comunes de esto son Bagging y Random Forest.
- **Métodos boosting**: basados en la construcción secuencial de modelos, intentando disminuir el sesgo del modelo combinando diferentes estimadores débiles. Cumple con la filosofía *"la unión de varios modelos débiles, puede construir uno fuerte"*. Ejemplos comunes de esto son AdaBoost y Gradient Tree Boosting.

2.1.2.1.5. Métodos de Regresión Los métodos de regresión, se utilizan cuando los resultados esperados no corresponden a clases categóricas, sino que a valores del tipo continuo. Modelos de regresión utilizan las características de los datos entregados y los correspondientes outputs de valores numéricos continuos para obtener relaciones o asociaciones entre estos datos. Con esta información, el modelo puede predecir resultados para un nuevo input, de forma similar a la clasificación, pero estos outputs corresponden a valores continuos. En este tipo de tareas se utilizan métodos como simple linear regression, multiple regression, polynomial regression, non-linear regression entre otros. El algoritmo de simple linear regression se basa en la relación lineal entre una variable dependiente y

una variable independiente, lo cual permite extrapolar una función que describa esta relación. Por otra parte, multiple regression utiliza múltiples variables independientes para describir a la variable dependiente. En un algoritmo de polynomial regression una variable dependiente se relaciona con una variable independiente mediante una función polinomial. Finalmente, non-linear regression se basa en una relación no lineal entre la variable dependiente y una o múltiples variables independientes.

2.1.2.2. Unsupervised learning

A diferencia de los algoritmos de aprendizaje supervisados, en donde se deseaba generar modelos predictivos a partir del aprendizaje de conjuntos de datos etiquetados, los algoritmos de aprendizaje no supervisado tienen como objetivo la generalización de comportamientos o identificación de patrones. Se le llama unsupervised learning debido a que el algoritmo busca aprender desde el set de datos sin que este posea algún tipo de información con respecto al output o etiquetas a observar. Es importante mencionar, que este tipo de generalizaciones o identificación de patrones, es conocido principalmente como clustering.

Los métodos de clustering buscan encontrar patrones y relaciones entre los ejemplos que componen un set de datos, con la finalidad de formar clusters entre aquellos más similares en función de sus atributos o características. Un cluster se define como un grupo que posee ejemplos que son similares entre sí, y distintos de aquellos que conforman otro grupo. Los algoritmos de clustering pueden ser divididos en seis reconocidos grupos: Hierarchical clustering, Density based clustering, Partitioning clustering, Graph-based clustering, Grid-based y Model-based clustering (Patel and Thakral, 2016).

Los métodos basados en Hierarchical clustering puede ser divididos en dos tipos, aglomerativo y divisivo. Los algoritmos aglomerativos comienzan de la suposición de que cada ejemplo corresponde a un cluster, y en base a una función de distancia, estos ejemplos se van uniendo en grupos similares. En el caso de los algoritmos divisivos, el algoritmo comienza desde la suposición de que todo el set de datos corresponde a un cluster, y estos se van dividiendo en distintos grupos(Thilagavathi et al., 2013).

El método de density based clustering se basa en la idea de que un grupo en un

espacio de datos es una región contigua de alta densidad de puntos, separada de otros grupos por regiones contiguas de baja densidad de puntos.

En los métodos de partitioning clustering el conjunto de datos se divide en K grupos, siendo K menor que le tamaño del conjunto de datos. Cada una de estas particiones representa un cluster. En muchos de los algoritmos de partitioning clustering se busca minimizar una función de costo, normalmente, relacionada con la distancia.

Datos que puedan ser representados mediante una estructura de grafos pueden ser agrupados mediante algoritmos de graph based clustering, también conocidos como búsqueda de comunidades. En este tipo de algoritmos, cada ejemplo corresponde a un nodo y la distancia entre estos a la conexión o arista que une a cada nodo. De esta forma, un conjunto de nodos que se encuentren cercanos entre sí y separados del resto pueden ser definidos como un cluster.

Los métodos de grid based clustering se diferencian del resto de métodos en que no se preocupa directamente de los puntos de datos, sino que de los valores que rodean a estos puntos. Mediante la partición del espacio en espacios rectangulares, se calcula la densidad de cada una de las celdas resultantes y se construyen clusters(Qiu et al., 2007).

Por último, los algoritmos de clustering model-based son conocidos además como mixture model. Estos están diseñados para modelar una distribución desconocida como una mezcla de distribuciones más simples, a veces llamadas distribuciones base (Lai et al., 2018).

2.1.2.3. Problemas asociados al desarrollo de modelos

Diferentes problemáticas se encuentran a lo largo del diseño e implementación, tanto de modelos predictivos como de aplicación de algoritmos de aprendizaje supervisado. Dentro de estas problemáticas se encuentra, el pre tratamiento de conjuntos de datos, la evaluación de dimensionalidad, problemas de sobreajuste, entre otros. A continuación, se explican de manera resumida cada uno de estos, así como también las estrategias que son utilizadas para poder resolverlos.

2.1.2.3.1. Evaluación de conjuntos de datos En muchos casos los datos deben ser procesados antes de ser sometidos a un algoritmo de aprendizaje. En

muchos casos, el conjunto de datos se encuentra afectado por ruido, missing values y gran dimensionalidad tanto en ejemplos como en sus atributos. Un set de datos se considera ruidoso, si éste se encuentra distorsionado. Datos producidos por errores en la medición, en el procesamiento o en la recolección se reconocen como random noise. Por otra parte, es posible encontrar ejemplos que parecen no pertenecer al conjunto seleccionado. A este tipo de datos se les conocen como outliers. Pueden ser producidos por errores en el etiquetado o transcripción o errores de digitación. Si estos no son eliminados del set pueden conducir a errores en el modelo o predicciones generadas a partir de éste, ya que muchas veces pueden otorgar preferencias hacia las clases o respuestas en los que han sido etiquetados. No obstante, dicho tratamiento debe ser con precaución y considerando la posibilidad de que el valor detectado como outlier realmente exista dentro de la distribución. Por lo tanto, normalmente se deja a criterio y conocimiento experto. Missing values es un fenómeno recurrente en los conjuntos de datos. Es posible que no se haya identificado un valor adecuado para el ejemplo, no fue posible registrarlo en la toma de datos, ocurrió un error en la anotación de los ejemplos, entre otros. En general, corresponde a una o más características en un ejemplo que no poseen un valor. Existen diversas formas de solucionar este problema, ya sea mediante la eliminación de ejemplos con missing values, asignación de variables globales a estos atributos, reemplazar estos missing values por los valores más frecuentes (si son categóricos) o el promedio de estos (si son continuos), o utilizando técnicas de modelamiento como nearest neighbors(Bhatia et al., 2010), Bayes' rule(Fukumizu et al., 2011), decision tree(Niuniu and Yuxun, 2010) u otros. De forma racional, si los datos entregados al algoritmo son de mala calidad, el modelo entrenado a partir de estos datos tendrá un bajo performance.

2.1.2.3.2. Dimensionalidad de conjuntos de datos Uno de los mayores problemas corresponde a la maldición de la dimensionalidad. Este problema, definido por primera vez por (Bellman, 2015), hace referencia a la cantidad de ejemplos necesarios para estimar una función arbitraria con un nivel de precisión esperado, en función a la cantidad de variables de entrada, es decir, la dimensionalidad del set de datos. En otras palabras, si se cuenta con una gran cantidad de features o dimensiones, se requiere mayor cantidad de datos para lograr un modelo exitoso. En un espacio de gran dimensionalidad, los datos se encuentran más dispersos. De forma práctica, este incremento en la dispersión o

escasez hace más difícil recolectar datos representativos de la población (Altman and Krzywinski, 2018). A medida que aumenta el número de variables, el número de sujetos en cada conjunto de categorías disminuye y la correlación entre dos sujetos cualquiera de las variables también disminuye.

En el caso de que el número de ejemplos sea mucho mayor a la cantidad de features representadas, es posible que la data sea redundante y favorezca a alguna clase, o realice una mala generalización de los datos. En general, en preguntas científicas es mejor tener mayor cantidad de datos. Sin embargo, la proliferación de datos que pueden no estar relacionados con la pregunta de interés conducen a esta maldición de dimensionalidad, lo que dificulta la capacidad de detectar relaciones y patrones reales.

Existen diversas maneras de abordar el problema de la dimensionalidad. Uno de los métodos más reconocidos para la reducción de dimensionalidad corresponde a principal component analysis (PCA). PCA construye una transformación dimensional en el cual el nuevo sistema de coordenadas se basa en aquellos features que posean mayor varianza en sus ejemplos(Abdi and Williams, 2010). Para construir esta transformación lineal debe construirse primero la matriz de covarianza o matriz de coeficientes de correlación. De esta forma, solo aquellos features que posean mayor varianza serán consideradas dentro del nuevo set de datos. Es posible utilizar una aproximación kernel PCA (Schreurs and Suykens, 2018), de forma que la función mediante la cual se realizará la reducción de componentes será no lineal. Joint mutual information (Yang and Moody, 1999) por otra parte observa la dependencia entre los features, es decir, que tanto se puede observar de un feature B a partir de uno A observado. Minimum-redundancy-maximum-relevance (mRMR) feature selection (Radovic et al., 2017) puede ser utilizado con mutual information, correlation o puntajes de distancia o similitud para la selección de features. En general, se basa en seleccionar features en base a la redundancia de estas, si es encontrada en otras características. Es posible utilizar otros algoritmos o aproximaciones como, t- Distributed Stochastic Neighbor Embedding (t-SNE) (Belkina et al., 2018), Uniform Manifold Approximation and Projection (UMAP)(McInnes et al., 2018) entre otras.

2.1.2.3.3. Sobreajuste y validacion de modelos. Otro de los problemas recurrentes observados al momento de generar un modelo corresponde a overfitting

y underfitting. Ambos problemas se relacionan con la capacidad de generalizar que posea el modelo. Overfitting se refiere a cuando el algoritmo aprende los detalles y el ruido en el set de datos, lo cual afecta negativamente el rendimiento del modelo. En otras palabras, ocurre cuando el modelo se adapta demasiado a los datos. En este caso el ruido y las fluctuaciones aleatorias son tomadas como conceptos para el modelo. (Altman and Krzywinski, 2018). Si el modelo se ajusta extremadamente bien a los datos de entrenamiento no será posible clasificar o predecir correctamente el valor de un nuevo ejemplo que no se encuentra en el set de entrenamiento.

El underfitting se da cuando el modelo no es capaz de capturar el patrón que se encuentra en los datos. En este caso, el algoritmo pierde la capacidad de generalizar debido a que no cuenta con suficiente información para observar el patrón presente en los datos. Usualmente puede ocurrir cuando la cantidad de ejemplos es insuficiente para observar esta relación, o cuando se busca implementar un algoritmo lineal en un set de datos no lineal.

La validación de modelos busca observar y resolver estos problemas mencionados de overfitting y underfitting. Se busca prevenir que el modelo obtenga buenos resultados para los datos de entrenamiento. Esto significa, que se quiere implementar un proceso que permita al modelo ajustarse de forma adecuada al patrón observado en la totalidad a esperar. De esta forma, si se presenta un nuevo ejemplo al modelo, se esperaría que este obtenga buenas medidas de desempeño. Puesto que se trabaja con un conjunto finito de datos, que en algunos casos son costosos de recolectar o medir, es necesario poder aplicar operaciones sobre un conjunto de datos especifico que permitan generalizar un patrón a todos los posibles valores observados.

Se espera que a través de este proceso sea posible evaluar y seleccionar los mejores parámetros para obtener un buen rendimiento en ejemplos fuera de los presentados en el entrenamiento. Pero, ¿Como obtenemos datos para validar el modelo generado si contamos con un conjunto limitado de datos y recursos? En este caso, es posible implementar métodos que dividan el set de datos original para generar nuevos conjuntos. En general, la división del conjunto produce tres sets de datos, para entrenamiento, validación y testeo del modelo. Esto con la finalidad de utilizar una porción de los datos para entrenar al modelo, otra determinada como validación para evaluar que tan bien está actuando el modelo frente a nuevos datos, mediante

medidas de desempeño, y finalmente, un tercer conjunto de datos de testeo para obtener resultados finales y evaluarlos frente a los outputs que se conocen o esperan del modelo. Algunos de los métodos más reconocidos para realizar esta operación de división sobre el set de datos de entrenamiento corresponden a k fold cross-validation, leave-one-out cross-validation, bootstrap y bagging (Kiranmai and Damodaram, 2014).

K fold cross-validation corresponde a un tipo de validación cruzada, en el cual, dado un parámetro K, el set de datos es dividido en k grupos. De estos K grupos, K-1 son utilizados para entrenar el modelo, y el restante es utilizado para validar el modelo construido a través del aprendizaje. Este proceso se repite hasta que todos los distintos grupos hayan sido utilizados para validar el modelo. Para cada set de datos de entrenamiento, se genera un nuevo modelo, y se obtiene un valor de desempeño a partir del set de datos de validación. Se retiene este valor y se descarta el modelo para entrenar otro desde cero con un nuevo set de datos de entrenamiento. Finalmente, se extrae un valor de desempeño a partir de aquellos obtenidos para cada modelo generado en el proceso de validación.

Cuando el valor de K es igual al tamaño de ejemplos N en el set de datos, el algoritmo de Cross validation pasa a ser una operación de leave one out. En este caso, el set de datos se divide en N grupos, esto significa, que cada ejemplo corresponde a un grupo. De esta forma, se entrena el modelo con N-1 datos, y uno de estos ejemplos es dejado para la validación del modelo. Al igual que en k-fold cross-validation, el proceso se repite hasta que cada grupo (o ejemplo en este caso) haya sido utilizado para validar el modelo. Este tipo de validación es más exhaustivo, sin embargo, se considera que genera menos errores dado que cada observación es considerada de forma individual y no depende de sets de datos con divisiones aleatorias.

El método bootstrap muestrea los datos de entrenamiento proporcionadas de manera uniforme con reemplazo. Es decir, cada vez que se selecciona un ejemplo, es igualmente probable que se vuelva a seleccionar y se vuelva a agregar al conjunto de entrenamiento. Si se cuenta con un set de datos con N ejemplos, este será muestreado N veces, en el cual un dato se extraerá hacia el set de entrenamiento y será de vuelto al conjunto de datos completos. De esta forma, es posible que el mismo ejemplo sea seleccionado más de una vez. Por otra parte, aquellos datos que no se encuentran dentro del set de datos de entrenamiento son utilizados

posteriormente para la evaluación del modelo(Han et al., 2011). Finalmente, bagging corresponde a un método también conocido como bootstrap aggregation, el cual, al igual que en bootstrap, el set de datos es dividido utilizando reemplazo entre los ejemplos. Para cada uno de estos sets se genera una serie de modelos, y los valores de performance son obtenidos como un ensamble de los resultados obtenidos por cada modelo (Han et al., 2011).

2.1.2.4. Medidas de desempeño.

El desempeño corresponde a una evaluación con respecto a la eficiencia del modelo. Esta eficiencia se basa en la comparación entre los resultados obtenidos con respecto a aquellos reales. Estas medidas de desempeño pueden basarse en tasas de error o aciertos presentados por un clasificador, o en valores matemáticos que indiquen precisión en un modelo predictivo. De forma general, un clasificador utiliza medidas extrapolables desde una matriz de confusión, las cuales corresponden al accuracy, precision, recall, False positive rate (FTP), True positive rate (TPR) o F score (Sokolova and Lapalme, 2009). Por otra parte, modelos predictivos utilizan medidas como como el Coeficiente de Pearson (Hauke and Kossowski, 2011), Coeficiente de Spearman (Hauke and Kossowski, 2011), Kendall τ rank (Johnson et al., 2014), Coeficiente de determinación R^2 score (Heagerty and Zheng, 2005) y Error cuadrático medio (Brassington, 2017).

Una matriz de confusión es una representación de las predicciones hechas por un modelo de clasificación. Se presentan los valores obtenidos para clases reales vs las clases predichas por el modelo, indicando en cada celda los ejemplos clasificados dentro de cada clase. De esta forma, es posible visualizar de forma más clara la cantidad de ejemplos clasificados en cada clase. Un ejemplo de matriz se presenta en la Figura 2.1.1. La diagonal en esta matriz representa aquellos ejemplos clasificados correctamente, llamados verdaderos positivos. Se denominan falsos positivos a los ejemplos clasificados dentro de una clase, que no pertenecen realmente a ésta (los valores en la columna sin considerar la diagonal). Por otra parte, se denominan falsos negativos a aquellos ejemplos pertenecientes a una clase especifica que fueron clasificados en otra (los valores presentes en una fila que no corresponden a la diagonal). Finalmente, se definen como verdaderos negativos a aquellos valores que pertenecen a la columna o la fila correspondiente a esta clase. Distintas medidas de error pueden ser obtenidas a partir de esta matriz, como las mencionadas

anteriormente.

- La accuracy es una medida que refleja el radio de predicciones correctas con respecto al total de predicciones realizadas por el modelo.
- Precision busca reflejar la precisión del modelo con respecto a una clase considerada como positiva. De esta forma, refleja la cantidad de ejemplos dentro de esta clase realmente correspondientes con respecto a los verdaderos positivos y falsos positivos en esta clase.
- El recall muestra el radio de ejemplos correctamente clasificados en una clase con respecto a los que pertenecen a esta (verdaderos positivos y falsos negativos). Suele utilizarse la métrica de precision cuando existe un mayor interés en los falsos positivos con respecto a los falsos negativos. Por otra parte, suele utilizarse cuando la cantidad de falsos negativos supera a los falsos positivos.
- El F score corresponde al promedio armónico entre precision y recall, dando una idea generalizada con respecto a ambos valores. Este valor alcanza su máximo cuando precision y recall son iguales, sin embargo, es posible demostrar la importancia de recall mediante un escalamiento de esta variable.

Otra medida importante extraída desde esta matriz de confusión corresponde a la sensibilidad y especificidad. Estas medidas permiten construir una Receiver operating characteristic curve o curva ROC mediante la cual es posible obtener un valor de desempeño denominada Area under the Curve o AUC. La sensibilidad corresponde al radio de verdaderos positivos con respecto al total de casos correspondientes a esta clase, es decir, a los verdaderos positivos y los falsos negativos. La especificidad hace referencia a la probabilidad de clasificar correctamente a un individuo en una clase denominada negativa. Este valor se obtiene como 1- FPR. En general, una curva ROC se construye tomando la sensibilidad en el eje Y, y 1- especificidad en el eje X. De esta forma es posible compara la tasa de verdaderos positivos con respecto a los falsos positivos de forma visual. Mientras más cerca se encuentre la curva del punto (0,1) mejor será la performance de este. Por el contrario, si la curva se encuentra cerca o por debajo del punto (x,y), donde x=y, entonces podría interpretarse que este modelo entrega resultados aleatorios o que comete más errores que aciertos. Cada punto en la curva ROC representa un par de sensibilidad / especificidad correspondiente

a un umbral de decisión particular. Otra medida que es posible extraer a partir de este gráfico corresponde a el área bajo la curva o AUC. El AUC proporciona una medición agregada del rendimiento en todos los umbrales de clasificación posibles. Una forma de interpretar el AUC es como la probabilidad de que el modelo clasifique un ejemplo positivo aleatorio más alto que un ejemplo negativo aleatorio.

En el caso de métricas utilizadas para modelos de predicción, el coeficiente de Pearson es uno de los más conocidos. Este coeficiente muestra una correlación lineal con respecto a dos variables. Al tomar valores entre -1 y 1, permite observar el nivel de correlación entre dos variables, donde un valor de -1 indica una relación indirecta (al aumentar una la otra disminuye) y un valor de 1 indica una relación directa (al aumentar una la otra también aumenta)(Sedgwick, 2012). El coeficiente de Spearman, a diferencia del coeficiente de Pearson, puede ser utilizado tanto para relaciones lineales como no-lineales. Además, posee la capacidad de medir la correlación entre dos variables continuas o discretas. Al igual que en el coeficiente de Pearson, este coeficiente oscila entre -1 y 1 (Sedgwick, 2014). Por otra parte, el coeficiente Kendall τ rank se utiliza para medir una relación ordinal entre dos variables. Se considera un test de hipótesis no paramétrico, debido a que se basa en el coeficiente τ. Este coeficiente entrega valores entre 0 y 1, donde 0 indica que no existe una relación entre ambas variables, mientras un valor de 1 indica una relación perfecta entre estas variables (Puka, 2011). El Coeficiente de determinación R^2 score indica la cantidad proporcional de variación en la variable de respuesta explicada por las variables independientes en el modelo de regresión lineal. Cuanto mayor sea el R cuadrado, más variabilidad se explica por el modelo de regresión lineal(Nagelkerke et al., 1991).Finalmente, el Error cuadrático medio mide el promedio de errores al cuadrado, es decir, compara un valor predicho con respecto al valor esperado o conocido. La diferencia se produce debido a la aleatoriedad o porque el estimador no tiene en cuenta la información que podría producir una estimación más precisa(Neill and Hashemi, 2018).

2.1.3. Neural networks o redes neuronales

Una red neuronal artificial es un modelo computacional o matemático inspirado en el funcionamiento del cerebro (Sinkov et al., 2016). Múltiples unidades conocidas como neuronas procesan la información y la traspasan entre ellas

		Clase Real		
		Clase1	Clase2	Clase3
Clase predicha	Clase1	P11	P12	P13
	Clase2	P21	P22	P23
	Clase3	P31	P32	P33

Figura 2.1.1: Ejemplo matriz de confusión. En este se ejemplifican 3 clases, denominadas clase 1, clase 2 y clase 3. La diagonal representa los ejemplos predichos correctamente por el modelo. Los valores en las filas correspondientes a las clases predichas, que no se encuentran en la diagonal corresponden a falsos positivos. Los valores presentes en la columna correspondiente a cada clase, que no se presentan en la diagonal corresponden a falsos negativos. Cualquier valor fuera de la fila correspondiente a la clase predicha y la columna de la clase real pertenecen a verdaderos negativos.

de forma homóloga a las neuronas biológicas. A nivel biológico, estas células conocidas como neuronas reciben un input de información por medio de puertos de entrada conocidos como dendritas. Dentro de esta célula, la información es procesada y transmitida hacia neuronas cercanas por medio del axón hacia los extremos terminales y expulsada. Computacionalmente, se busca expresar este comportamiento por medio de ecuaciones matemáticas y nodos interconectados. Si se considera que se entrega un input X al nodo, éste agrega un peso W a este input. Dado que puede recibir múltiples inputs, cada uno de estos es multiplicado por el peso entregado por la neurona, y posteriormente sumado entre sí. Si el valor obtenido supera un valor de activación, entonces el estado de esta neurona cambia a activado y se transmite un output a las neuronas conectadas o cercanas (Rebala et al., 2019). En resumidas cuentas, los elementos principales en este algoritmo corresponden a : Neuronas artificiales, peso asociado y función de transferencia.

Un input es entregado a una neurona artificial. Observando este input, se genera un valor de peso sobre la conexión de la cual proviene el input. Dentro del nodo, todos estos inputs son multiplicados por los correspondientes pesos de origen, y posteriormente sumados. En algunos casos, se agregan aportes desde el entorno. Si ha sido establecida una función de activación, entonces se compara el valor obtenido en la sumatoria con respecto al esperado en esta función. Si el valor supera un límite, entonces el resultado pasa hacia la función de transferencia, en la cual se genera un output de este nodo. Este output puede ser utilizado como input

para neuronas contiguas o conectadas en la red neuronal (Pagel and Kirshtein, 2017).

Una red neuronal está compuesta de forma básica en tres partes: una arquitectura, un algoritmo de aprendizaje y una función de activación(Ripundeep Singh Gill, 2014).

Las redes neuronales artificiales son sistemas adaptativos que cambian su arquitectura basados en la información entregada en el proceso de aprendizaje (Singh and Chauhan, 2009). La primera red neuronal construida por Frank Rosenblatt (Rosenblatt, 1960) corresponde a la arquitectura más simple en una red neuronal. Ésta consiste, básicamente en una sola neurona, a la cual ingresan inputs con pesos sinápticos, y que posee un umbral de activación. A partir de esta neurona se produce un output de acuerdo con la función umbral de activación. Las arquitecturas corresponden a la disposición topológica en la red neuronal, es decir, la disposición de los nodos, la conexión entre estos y la dirección del flujo de información. De forma general estas arquitecturas pueden agruparse en tres grupos: Feed-Forward Neural Networks, Self-organization networks (Gaur, 2012). y Recurrent Networks.

Algunas de sus ventajas, además de su plasticidad, corresponden a su naturaleza no paramétrica (no requiere de suposiciones previas) y la capacidad de generalizar (Zhang, 2009).

- Feed-forward neural networks(FFNN) : Uno de los modelos de redes neuronales más simples, que consta de tres capas (Singh and Chauhan, 2009). Una capa de entrada, una capa escondida y una capa de salida. Utiliza normalmente algoritmos de back propagation (Hecht-Nielsen, 1992) para ajustar los pesos en la red y minimizar el error del modelo, mediante una retroalimentación al algoritmo. Back propagation corresponde a una técnica ampliamente utilizada para la corrección de pesos en una red neuronal. Mediante el cálculo de un gradiente de error o perdida, si la diferencia es mayor a aquella aceptada, el algoritmo retorna hasta la capa escondida para ajustar los pesos y generar mejores resultados. Este modelo se utiliza generalmente para áreas de predicción y reconocimiento de patrones. Se denominan de esta forma dado que la información fluye entre las capas en una sola dirección. Un esquema de este tipo de arquitecturas se presenta en

la Figura 2.1.2.

- Recurrent Neural networks (RNN): También conocida como Auto Associative o Feedback Network, pertenece a una clase de redes neuronales artificiales donde las conexiones entre unidades forman un ciclo dirigido (Poznyak et al., 2019). Este tipo de redes neuronales puede trabajar con secuencias temporales. La NN se encuentra conectada desde outputs a inputs, esto quiere decir, que outputs producidos por un nodo pueden ser utilizados como input para otro. Esta característica le permite tener un concepto de memoria temporal. Utiliza los mismos parámetros para un input en cada entrada ya que realiza la misma tarea en todas las entradas o capas ocultas para producir la salida. Esto reduce la complejidad de los parámetros, a diferencia de otras redes neuronales. Un ejemplo de esta arquitectura se observa en la Figura 2.1.3.

- Self-organization networks: Este algoritmo realiza una búsqueda en el espacio del modelo, mediante la construcción de hiper-superficies. En una red de nodos, cada nodo representa una hiper superficie, la cual es organizada para ser una aproximación del modelo real (Tenorio and Lee, 1989). En este proceso de aprendizaje o self-organization, las neuronas activas seleccionan los inputs recibidos dado un criterio especificado, y determina los pesos de confianza para estas conexiones. De esta forma, las mismas neuronas estructuran la red neuronal (Ivakhnenko et al., 1994).

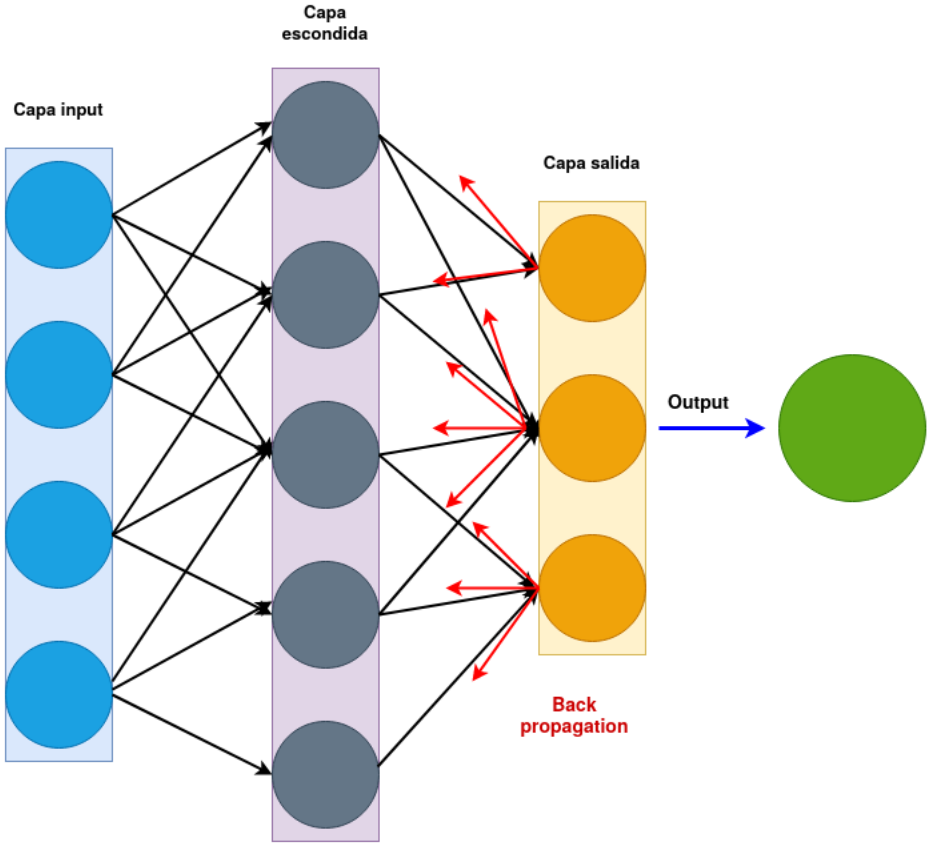


Figura 2.1.2: Esquema de una red neuronal Feed forward. Esta arquitectura está compuesta por una capa de entradas o inputs, las cuales traspasan la información a la siguiente capa con cierto valor o peso asociado, el cual puede ser modificado por medio de la experiencia adquirida por el modelo utilizando el algoritmo de back propagation. Si el valor entregado a la capa de salida no coincide con aquel observado o esperado, entonces el algoritmo de back propagation permite que la red neuronal ajuste los pesos utilizados para los inputs. Finalmente, la información se traspasa hacia la capa de salida, la cual genera un valor de salida asociado al input ingresado.

Con la finalidad de capturar la capacidad de abstracción del cerebro humano se ha desarrollado un nuevo modelo de aprendizaje artificial conocido como deep learning, el cual utiliza un gran número de capas de procesamiento (Schmidhuber, 2015). Diferentes arquitecturas pueden ser utilizados en este tipo de machine learning, como deep learning network, matrices de convoluciones o redes neuronales recurrentes. Esta tecnología cuenta con un costo computacional generalmente más alto en comparación con otro tipo de inteligencias artificiales, sin embargo, posee una capacidad destacable en precisión en tareas de clasificación (Tavanaei et al.,

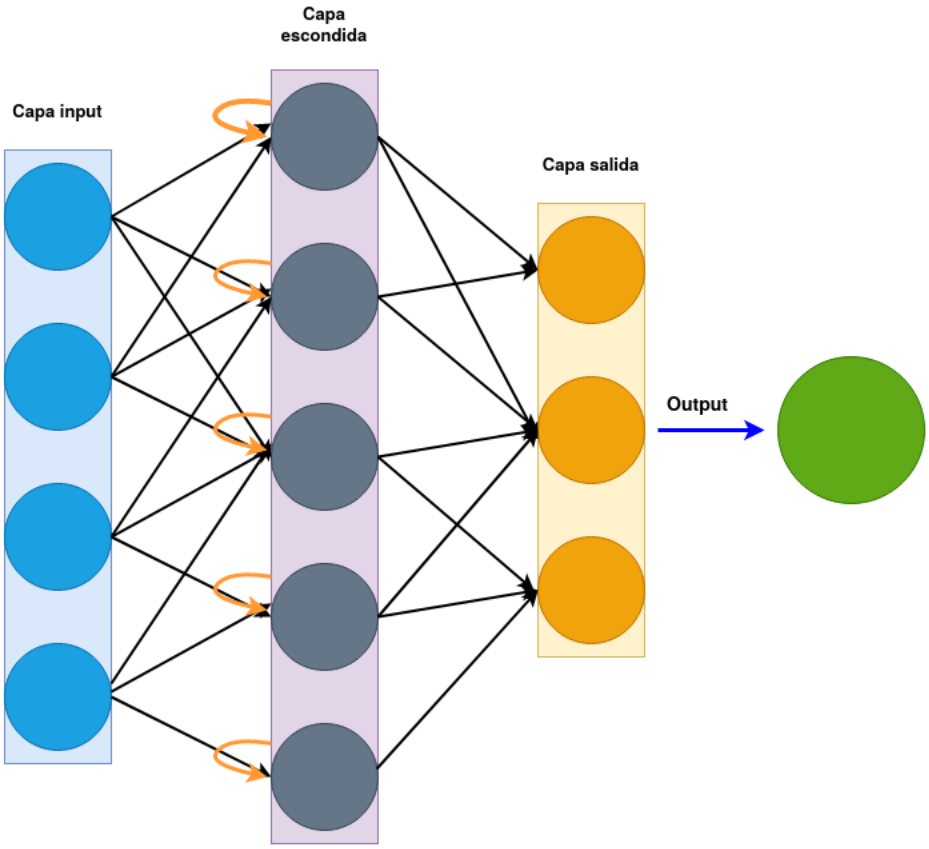


Figura 2.1.3: Esquema de una red neuronal Recurrent. Esta arquitectura está compuesta por las tres capas características en una red neuronal. A diferencia de las arquitecturas FFNN, en este tipo de modelos la información obtenida desde un nodo en la capa escondida puede ser utilizados como inputs. Esto permite modelar secuencias temporales, y mantener cierta memoria con respecto a los valores encontrados en el proceso de entrenamiento.

2019). Se han desarrollado diversas técnicas en busca de mejorar el desempeño de estas redes neuronales, como un acercamiento al funcionamiento de las neuronas biológicas utilizando los altos de intensidad de activación variables (Tavanaei et al., 2019), o utilizando la entropía de transferencia en las rutas de las redes profundas para identificar candidatos de retroalimentación entre las capas convolucionales y determinar sus pesos sinápticos finales utilizando la programación genética(Herzog et al., 2020).

2.1.4. Codificación de secuencias

Si se cuenta con set de datos categórico, al utilizar métodos de minería de datos, es necesario codificar estas secuencias a un lenguaje comprensible e informativo para el algoritmo seleccionado. Esta codificación puede ser definida como cifrar la información que, en palabras simples, es la transformación de los datos a un lenguaje entendible por computadores. Algunos de los métodos utilizados para resolver esta problemática son One-hot encoder (Brownlee, 2017) , ordinal encoder (Potdar et al., 2017) , frequency (García, 2018) y propiedades fisicoquímicas (Gök et al., 2016).

One hot encoder corresponde a un método de codificación en el cual datos categóricos son transformados a un vector numérico. En primer lugar, esto requiere de la asignación de valores enteros a los valores categóricos entregados. Luego, cada uno de estos valores se representa como un vector binario, en el cual, cada valor es cero a excepción de aquel correspondiente índice del valor entregado, cuyo valor es 1. Este tipo de codificación es especialmente útil cuando no existe una relación ordinal entre los valores que representan los datos (como podría ser si se hablase de temperatura).

En el caso de ordinal encoder los elementos entregados poseen una relación ordinal o de secuencia, es decir, el orden que siguen los elementos también entrega información, por ejemplo, alto, medio o bajo. En este caso, se asigna un valor a cada etiqueta o elemento tratando de mantener la relación de orden entre estos.

Otros acercamientos como frequency encoding o codificación de frecuencia se basan en la cantidad de veces que se observa una categoría con respecto al total de ejemplos, es decir, la frecuencia de cada categoría en el set de datos. En la utilización de secuencias de aminoácidos, la consideración de las propiedades fisicoquímicas al momento de entregar información al modelo de minería de datos puede producir resultados muy provechosos. En esto se basa la codificación de propiedades fisicoquímicas. Cada uno de los elementos en la secuencia es transformado a un valor que refleja alguna propiedad fisicoquímica de este, como puede ser el volumen, hidrofobicidad, pKa entre otros. El uso de alguno de estos métodos de codificación debe ser basado en el problema a abordar, considerando la información que puede entregar cada uno y la capacidad de relación con el set de datos a utilizar.

2.1.5. Embeddings y auto encoders.

Embeddings corresponde a esta técnica mediante la cual es posible transcribir vectores de un espacio dimensional grande a otro espacio dimensional de menor tamaño. Idealmente, un embeddings captura parte de la semántica de la entrada al colocar entradas semánticamente similares juntos en el espacio reducido generado mediante este proceso. Generalmente, esta técnica suele ser utilizada con dos estructuras de datos, correspondientes a grafos y palabras Camacho-Collados and Pilehvar (2018).

Embeddings de grafos realiza una transformación de esta estructura de nodos interconectados a vectores. Cada vector debe reflejar las características del nodo y sus conexiones con el resto de los nodos. Por lo tanto, un embeddings en grafos asigna cada nodo a un vector de características de baja dimensión e intenta preservar las fuerzas de conexión entre los vértices (Goyal and Ferrara, 2018). Algunos algoritmos de embedding utilizan la matriz de adyacencia extraída desde un grafo para conservar estas relaciones de conectividad entre nodos Graph Factorization (Ahmed et al., 2013) y LINE (Tang et al., 2015). Otras se basan en la matriz de similitud para considerar relaciones de orden mayor, como es el caso de HOPE (Ou et al., 2016).

Embeddings en palabras busca transformar estas palabras en vectores numéricos, manteniendo el contexto de estas palabras(Camacho-Collados and Pilehvar, 2018). Esto implica que, palabras con significado contextual similar, se encontraran cercanas en el espacio vectorial. De forma general, el planteamiento de relación de una palabra con el contexto puede ser realizado desde dos puntos de vista distintos. Aquellos en los cuales las palabras se expresan como vectores de palabras concurrentes, y los que el vector es expresado en función de contextos lingüísticos en los que las palabras aparecen (Lavelli et al., 2004).

Actualmente, gran parte de los algoritmos de embeddings utilizados se encuentran basados en arquitecturas de Neural networks. La herramienta de embeddings en palabras mayormente conocida corresponde a word2vec(Goldberg and Levy, 2014), la cual fue desarrollada por un equipo en Google liderado por Tomas Mikolov. La base de este algoritmo es el skip-algorithm. En este, si se posee un cuerpo de palabras w, dada por un contexto c, se calcula la probabilidad de que estas palabras hayan surgido de este contexto y, por lo tanto, conformen el texto deseado

(Goldberg and Levy, 2014). El objetivo es encontrar los parámetros que maximicen la probabilidad de este texto. La parametrización y optimización de esta se realiza por medio de un modelo de Neural network. Por otra parte, la definición del contexto de una palabra está definida por un conjunto de parámetros extraídos de las palabras vecinas(Mikolov et al., 2013) . La ventana k desde la cual se extraen los parámetros de las palabras vecinas es variable, es decir, k solo denota el máximo de valores a tomar desde la vecindad. Por otra parte, palabras que aparezcan un menor número de veces que el mínimo establecido Lmin no serán contadas como vectores o como aportes en contexto. Mientras que, palabras que aparezcan un mayor número de veces que el límite establecido Lmax, serán muestreadas hacia abajo. Las palabras extremadamente redundantes serán removidas del texto antes de que estas sean convertidas a vectores (Goldberg and Levy, 2014).

Es posible desarrollar algoritmos que construyan reglas o métodos de codificación. Un auto encoder corresponde a un tipo de NN utilizado para desarrollar un codificador eficiente a partir de un set de datos, y de forma no supervisada (Walter Hugo and Mechelli, 2020) . El objetivo de un auto encoder es aprender una representación (codificación) para un conjunto de datos, típicamente para la reducción de dimensionalidad, entrenando a la red para ignorar el ruido"de los datos (Wang et al., 2014). Al mismo tiempo que el algoritmo aprende la forma de codificación, desarrolla el método de reconstrucción, donde el auto encoder intenta generar a partir de la codificación reducida una representación lo más cercana posible a su entrada original. En la Figura 2.1.4 se presenta un esquema del funcionamiento de un auto encoder.

La estructura de un auto encoder por lo general corresponde a un feed-forward NN, las cuales tienen una estructura simétrica y está diseñado para aprender una aproximación a la función de identidad, para que la salida sea lo más similar posible a la entrada (Tan and Eswaran, 2008). El auto encoder está compuesto por dos redes, una utilizada para pasar desde el espacio de alta dimensión a features en un espacio de menor dimensión, mientras la otra es utilizada para reconstruir los datos luego de la codificación. Ambas redes se entrenan conjuntamente , ajustando los pesos de la red del decodificador primero y la red del codificador después. El objetivo es minimizar la deferencia entre el output obtenido en la reconstrucción y el input entregado al codificador (Wang et al., 2012). Los autoencoders pueden ser dividirse en dos grupos: Regularized y Variational.

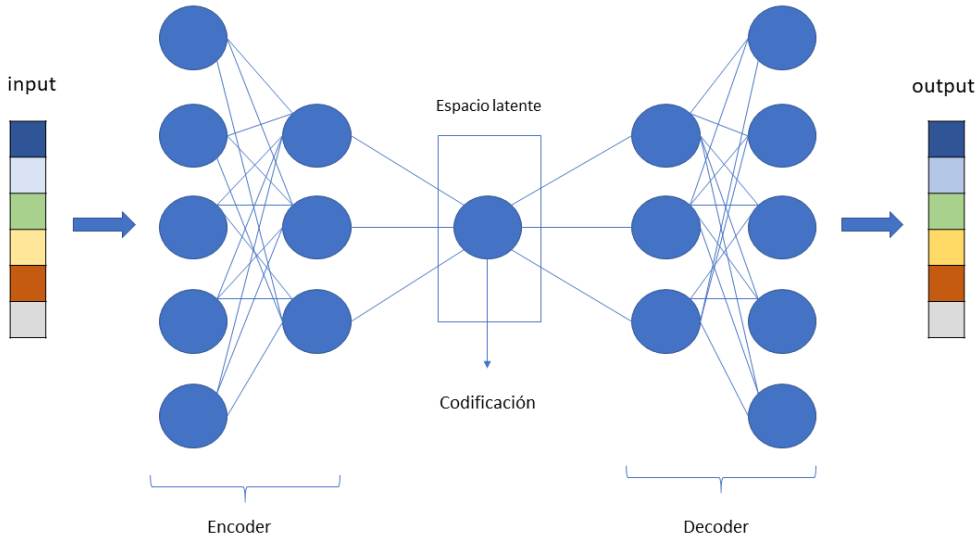


Figura 2.1.4: Esquema de autoencoder.Un autoencoder corresponde a una red neuronal, la cual recibe un input, el cual por medio de capas en esta red es transformado en un vector en el espacio latente, es decir, la representación codificada del input. Esta representación puede ser decodificada utilizando las capas de decodificación del autoencoder, para reconstruir, idealmente, el input. Es posible que la reconstrucción genere una aproximación muy cercana del input.

Regularized autoencoders capturan la estructura de la distribución del entrenamiento gracias a la oposición productiva entre el error de reconstrucción y un regularizador (Alain and Bengio, 2014). La función de error en la reconstrucción penaliza el error cometido, considerando la función de reconstrucción de un ejemplo respecto al valor codificado o entregado. Un regularized auto encoder utiliza una función reguladora, la cual busca hacer que la función de decodificación (o codificación) sea tan simple como es posible, es decir, tan constante como sea posible y tan insensible al valor de ejemplo utilizado como sea posible (Alain and Bengio, 2014).

Por otra parte, Variational Autoencoders (VAEs) son dirigidos por modelos gráficos probabilísticos (DPGM). Estos son denominados auto encoders porque el objetivo final de entrenamiento que se deriva de esta configuración tiene un codificador y un decodificador, y se asemeja a un auto encoder tradicional (Doersch, 2016). Estos modelos, correspondientes a modelos generativos, intentan simular cómo se generan los datos para comprender las relaciones causales subyacentes. El marco de Variational Autoencoders proporciona un método basado en principios para el aprendizaje conjunto de modelos variables latentes profundos y modelos de

inferencia correspondientes utilizando stochastic gradient descent (Kingma and Welling, 2019). Un modelo de inferencia , también llamado encoder o recognition model, aproxima la distribución de las variables en el modelo generativo.

2.1.6. Herramientas de auto encoding

Diversas librerías han sido implementadas para el diseño e implementación de auto encoders. Entre estas se encuentran Keras (Chollet et al., 2015), pytorch (Paszke et al., 2019) y TensorFlow (Abadi et al., 2016). Keras corresponde a una biblioteca de redes neuronales de acceso libre, implementada en lenguaje de programación Python y ejecutable sobre diversas herramientas. La librería cuenta con diversos módulos integrados, que permiten trabajar sobre las capas, funciones de activación, optimizadores, entre otros. De igual forma, esta herramienta permite la implementación de modelos basados en deep learning, aumentando de esta forma la capacidad de abstracción de la red neuronal implementada. Pytorch corresponde a una librería de minería de datos orientada a deep learning, planteada por el laboratorio de inteligencia artificial de Facebook, la cual posee diversos módulos implementados orientados a facilitar el manejo e aplicación de redes neuronales y sus arquitecturas. La librería busca disminuir el tiempo de cálculo requerido por este tipo de algoritmos, utilizando técnicas de aceleración de hardware como el uso de la GPU. De igual manera, TensorFlow permite la disminución del tiempo de cálculo, distribuyendo el proceso tanto a CPU como a GPU. Esta biblioteca, desarrollada por Google, permite la implementación de herramientas de machine learning, y el trabajo con redes neuronales para descifrar patrones y correlaciones.

Las principales desventajas de la utilización de estas herramientas para el entrenamiento de un auto encoder son, en primer lugar, el largo periodo de tiempo y costo computacional requerido para su entrenamiento. Dado que posee diversos parámetros modificables, como el número de capas, número de nodos, funciones de activación, entre otros, la complejidad y tiempo de calculo requerido aumenta en función de estos. Por otra parte, debido a la implementación aplicada a este tipo de algoritmos, y a los múltiples cálculos ejecutados, el entrenamiento de auto enconders requiere de una capacidad considerable de memoria en un computador. Por otra parte, el entrenamiento de un auto encoder que posea una capacidad de abstracción general con buenas medidas de desempeño requiere de un conjunto de datos adecuado para esta tarea, ya sea en tamaño como en contenido,

y puede llegar a requerir múltiples ciclos de perfeccionamiento. En otras palabras, la construcción de un autoencoder requiere que se produzca una arquitectura adecuada, seleccionando una cantidad específica de neuronas y capas escondidas, el peso de éstas y la forma en la cual se conectan. Además, es necesario entregar un conjunto de secuencias adecuado para que la red neuronal pueda capturar las características apropiadas de éstas, y evaluar el modelo generado hasta recibir outputs similares a las entradas entregadas.

Por otro lado, se han diseñado herramientas orientadas a la aplicación de codificación por auto encoder en datos de origen biológico. Tasks Assessing Protein Embeddings (TAPE) (Rao et al., 2019) corresponde a una herramienta que cuenta con conjunto de cinco tareas de aprendizaje semi supervisado que son relevantes en la ingeniería de proteína. De esta forma, el algoritmo puede hacer uso de unos pocos ejemplos etiquetados y muchos no etiquetados, aumentando la exactitud del aprendizaje (Zhu and Goldberg, 2009). La herramienta cuenta con tres modelos pre entrenados disponibles, Transformer model (Vaswani et al., 2017), UniRep model (Alley et al., 2019) y TrRosetta model (Yang et al., 2020), orientados a diversos tipos de datos y problemáticas en ingeniería de proteínas. En primer lugar, Transformer model posee una arquitectura de auto atención, y nodos de capa de encoder y decoder completamente conectados. Una función de atención le permite mapear un conjunto de vectores a un output obtenido. Esto permite al modelo trabajar de forma mas rápida en comparación a modelos RNN. Este modelo pre entrenado es de gran utilidad en tareas de traducción entre idiomas. UniRep model, en cambio, se basa en la codificación de secuencias biológicas. Su arquitectura fue entrenada en aproximadamente 24 millones de secuencias extraídas desde UniRef50, mediante un algoritmo de RNN. Debido al tipo de información con el cual fue entrenado UniRep model, este modelo es de gran utilidad para representar secuencias con relación evolutiva. Finalmente, el modelo pre entrenado TrRosseta model, corresponde a una arquitectura basada en Convolutional Neural Networks, entrenada sobre un conjunto de datos con información de distancia evolutiva, así como de orientación, distancias y ángulos de residuos en su estructura tridimensional. Este modelo es de gran utilidad en la predicción de estructuras tridimensionales, y el diseño de novo de proteínas.

2.1.7. Natural Language Process (NLP)

Text mining es una nueva y novedosa arista en el análisis predictivo y data mining. Se basa en el uso de técnicas derivadas desde estadística, machine learning y lingüística (Talabis et al., 2015). En gran parte, los datos entregados a un algoritmo de machine learning se encuentran estructurados en filas y columnas, correspondiendo las columnas a los features y las filas a ejemplos. ¿Sin embargo, como es posible entregar a un algoritmo datos no estructurados en este formato como es el texto? Text mining ha desarrollado herramientas para trabajar con estos datos no estructurados (Allahyari et al., 2017).Este enfoque surge de la necesidad de procesar el lenguaje natural humano. Al hablar de esta organización, se hace referencia al tipo de estructura con la cual trabajan los computadores, debido a que, para un ser humano, un texto en un libro o documento posee un sentido, coherencia y semántica, lo cual debe ser reflejado en la estructura del texto, la cual es comprensible para el lector. En text mining se busca encontrar palabras claves entre el conjunto de palabras que conforman el texto, con la finalidad de encontrar patrones o realizar predicciones (Kotu and Deshpande, 2015).

Natural language processing es el uso de computadoras para procesar un texto o discurso. Corresponde a un área de investigación que busca la forma en la cual los computadores puedan comprender y utilizar el lenguaje natural. Entiéndase por lenguaje natural aquel desarrollado por la sociedad, el cual ha evolucionado a lo largo del tiempo y son difíciles de describir con reglas específicas (Bird et al., 2009). Los investigadores en esta área buscan comprender como el ser humano entiende y utiliza el lenguaje de forma que sea posible desarrollar herramientas y técnicas apropiadas para que los sistemas informáticos comprendan y manipulen idiomas, y de esta forma, realizar las tareas deseadas. Esta disciplina incluye desde inteligencia artificial y ciencias de la computación hasta psicología y lingüística .Mediante la integración de minería de datos, se puede utilizar este método para extraer relaciones en una cadena de texto no visualizadas de forma común, o que se encuentran entre un gran volumen de datos. Por otra parte, la lingüística se utiliza para comprender la estructura y el significado de un texto mediante el análisis de diferentes aspectos como la sintaxis(relación y orden), la semántica (significado de la expresión), la pragmática (contexto como situación en la cual se produce la comunicación) y la morfología (estructura de las palabras).

La naturaleza usa ciertos lenguajes para describir secuencias biológicas como el ADN, el ARN y las proteínas. Al igual que los humanos adoptan idiomas para comunicarse, los organismos biológicos usan lenguajes sofisticados para transmitir información dentro y entre las células. En el campo de la bioinformática las técnicas de NLP ha permitido extraer relaciones proteína-proteína(Yu et al., 2018), gen-enfermedad (Guan et al., 2019) o descifrar la estructura secundaria de una proteína a partir de su secuencia(Zeng et al., 2015). La capacidad de comprender el lenguaje le permitiría a la máquina, dado una secuencia de aminoácidos, comprender la relación evolutiva subyacente a estos.

Los enfoques de machine learning hacia la NLP requieren que las palabras se expresen en forma vectorial. Para lograr esta tarea, se utilizan técnicas de embeddings como las revisadas anteriormente, las cuales permiten traspasar estas palabras a vectores en un espacio de menor dimensión (de forma general, desde vectores dispersos a vectores densos)(Song and Roth, 2015).Son utilizadas además técnicas de codificación para la transformación de secuencia a vector numérico, ya sea mediante One-hot encoding, frecuencia o tomando sus propiedades fisicoquímicas. NLP permite a las computadoras realizar una amplia gama de tareas relacionadas con el lenguaje natural en todos los niveles, que van desde el análisis y etiquetado de part-of-speech (POS), hasta machine translation y dialogue systems. Las dos principales clasificaciones para enfoques de NLP son Statistical learning approach y rule based(Sharma and Kaushik, 2017).

Statistic NLP comprende todos los enfoques cuantitativos para el procesamiento automatizado del lenguaje, incluidos modelado probabilístico, teoría de la información y algebra lineal(Kocaleva et al., 2016). El uso de modelos estadísticos ofrece una buena solución al problema de la ambigüedad: los modelos estadísticos son robustos, se generalizan bien y se comportan con gracia ante la presencia de errores y nuevos datos. Dado que los modelos estadísticos poseen la capacidad de generalizar bien los sets de datos y comportarse de forma robusta, se han utilizado en conjunto con NLP para evitar tratar con la ambigüedad. Por otra parte, poseen la ventaja de que los parámetros para este modelo estadístico con frecuencia pueden ser extraídos desde el cuerpo del texto (Hristea, 2011).

Deep learning y neural networks han ganado importancia en el área de NLP, debido a sus capas ocultas entre entradas y salidas, y una amplia red para proporcionar los mejores resultados (Sharma and Kaushik, 2017). En recursive

neural networks la semántica es extraída utilizando estructuras de árbol. Sin embargo, en frases largas el tiempo de construcción de este árbol puede ser excesivo y, por lo tanto, el algoritmo se vuelve ineficiente. Por otra parte, en recurrent neural networks información previa puede ser almacenada en capaz escondidas y utilizada posteriormente para obtener información con respecto al contexto. El problema con este enfoque es que este tipo de red neuronal parece presentar un sesgo hacia el final del documento. De esta forma, es posible que palabras claves en el texto sean ignoradas (Sharma and Kaushik, 2017). La arquitectura de NN que ha generado mejores resultados y es ampliamente utilizada corresponde a Convolutional Neural Network. Esta arquitectura presenta un modelo sin sesgos, que utiliza kernels convolucionales como parte de su arquitectura deep learning. Uno de los softwares más conocidos para Natural language processing corresponde a Tensor Flow (Abadi et al., 2016). Esta herramienta desarrollada por Google, y utilizada en muchas de sus aplicaciones como speech recognition o Google Photos. Este software esta desarrollado sobre un algoritmo de deep learning en neural networks, el cual proporciona una entrada como tensor (una entidad algebraica que generaliza los conceptos de escalar, vector y matriz independientemente del sistema de coordenadas utilizado) el cual es fluye a través de la red agregando cierto peso y finalmente es procesado utilizando una función softmax(función que permite comprimir un vector de valores reales en un vector de rango 0 a 1).

2.2. Metodología y estrategias de desarrollo.

Análisis de ProtoArrays (Robinson et al., 2002) se emplearon con el fin de obtener un conjunto de 45 anticuerpos y sus interacciones contra 8000 antígenos (Frick, 2009). Los resultados obtenidos de estas técnicas fueron codificadas en imágenes que permiten evaluar el nivel de interacción entre dos moléculas en base a técnicas de tratamiento imágenes con métodos de convolución, logrando cuantificar la interacción (Torres Almonacid, 2020).

A partir de los conjuntos de datos asociados a los niveles de intensidad de interacción, métodos de análisis y limpieza de datos en base a propiedades estadísticas y diseño de filtros fueron aplicados con el fin de detectar interacciones anómalas según los controles positivos y negativos empleados. Así como también, la detección y eliminación de ruido, reduciendo satisfactoriamente el tamaño del

conjunto de datos a 262.412, los cuales pueden considerarse confiables de acuerdo a los test estadísticos aplicados (Torres-Almonacid et al., 2019).

Finalmente, los niveles de intensidad asociados a la interacción antígeno-anticuerpo, fueron categorizados en clases **Alta**, **Media**, y **Baja**, en base a la aplicación de diferentes test estadísticos[1]. De esta forma, archivos multifasta con las secuencias aminoacídicas de los antígenos y secuencias nucleotídicas de los anticuerpos, en adición a conjuntos de datos que representan el nivel de interacción entre un antígeno y un anticuerpo, serán los conjuntos de datos a manipular durante el desarrollo de este proyecto de memoria de título.

A continuación se presentan los materiales y métodos a utilizados para el cumplimiento y evaluación de la hipótesis planteada, desarrollados por objetivo especifico planteado.

2.2.1. Limpieza y preparación del set de datos.

Dado que la información entregada con respecto a los anticuerpos corresponde a secuencias de nucleótidos, fue necesario realizar la traducción de esta secuencia. Se utilizó la herramienta Translate Tool de ExPASy (Artimo et al., 2012), seleccionando el marco de lectura cuyo resultado presentara el mayor largo, y menor número de aminoácidos faltantes. Luego, se revisó si existían entradas repetidas dentro del set de datos categorizado, secuencias de aminoácidos vacías, o pares de antígeno anticuerpo sin clase otorgada. Dado que el número de datos es suficientemente grande, los ejemplos que presentaran estos errores fueron eliminados del set de datos.

2.2.2. Codificación de secuencias

Se utilizó la herramienta TAPE, mediante un script implementado en Python v3.6. Se utilizaron los modelos pre entrenados Babbler1900 y Bert-Base. Para el modelo Babbler se utilizó el tokenizer Unirep, mientras que el tokenizer IUPAC se utilizó en el caso del modelo Bert-base. Las secuencias de anticuerpos y antígenos fueron sometidas a esta metodología, generando dos set de datos distintos, ambos basados en codificación con embedding. El output obtenido de esta codificación

[1]El desarrollo de los test, así como también los análisis de datos relacionados a la interacción son parte de trabajo en desarrollo por el grupo de investigación CeBiB-UMAG.

fue posteriormente trabajado utilizando la librería NumPy disponible en Python, para obtener los vectores numéricos ponderados correspondientes a cada secuencia. Estos vectores fueron posteriormente relacionados de acuerdo a los identificadores de cada secuencia, y a los ejemplos presentados por el set de datos categorizados. De esta forma, se obtuvo un set de datos que contiene el vector codificado de anticuerpo, concatenado con el vector codificado de antígeno, y la clase respuesta correspondiente a esta interacción. En la Figura 2.2.1 se presenta un esquema de la metodología utilizada para la realización de esta tarea.

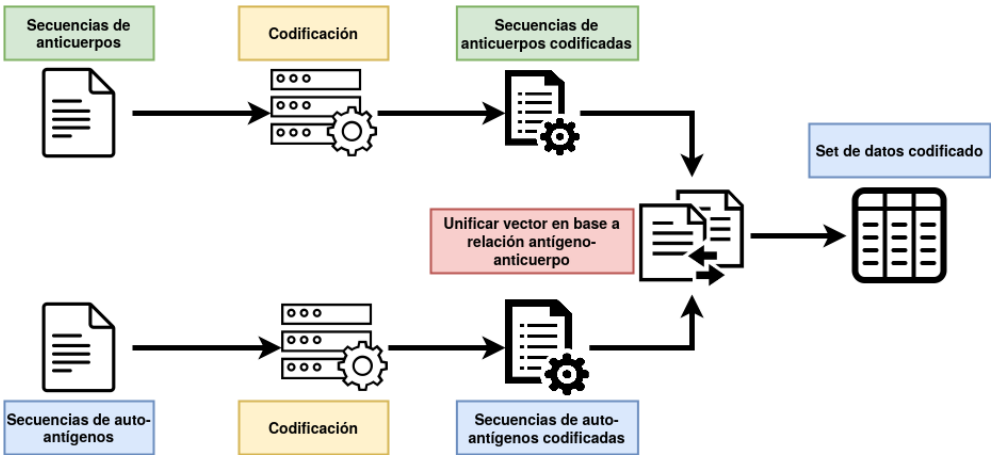


Figura 2.2.1: Esquema codificación de secuencias. En primer lugar, las secuencias de anticuerpos y antígenos son codificadas utilizando embedding. Luego, los vectores obtenidos para cada secuencia son relacionados de acuerdo a las interacciones observadas en los datos entregados. Los vectores relacionados, a los cuales se les ha asignado una clase de acuerdo a la interacción observada conformarán el set de datos.

2.2.3. Estrategias de entrenamiento y selección de modelos predictivos.

Se evaluó el desarrollo de un modelo ensamblado para llevar a cabo la predicción de posibles interacciones entre antígenos y anticuerpos. Para esto, se utilizó DMAkit-lib (Medina-Ortiz et al., 2020b). Esta librería posibilitó la exploración en los distintos algoritmos de aprendizaje supervisados y los hiper parámetros aplicables a estos, para generar el conjunto de modelos a evaluar. Para cada combinación de algoritmo e hiper parámetros se utilizó el 70 % de los datos, y se aplicó cross validation, con un parámetro K igual a 5, durante el entrenamiento de el modelo. Luego, el 30 % restante de los datos fue utilizado para la evaluación de los modelos.

Un esquema de este proceso se muestra en la Figura 2.2.2 Los algoritmos, tipo de algoritmo, parámetros requeridos, uso e iteraciones realizadas se muestran en la siguiente tabla.

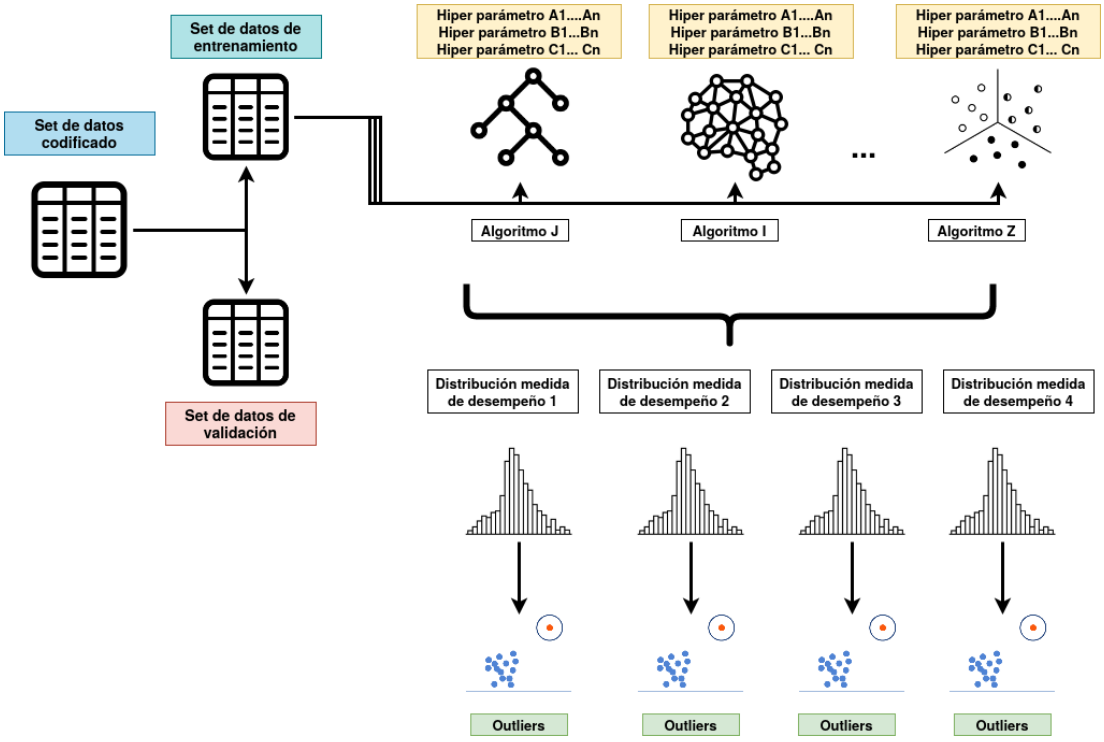


Figura 2.2.2: Esquema de entrenamiento de modelos. Utilizando el set de datos de entrenamiento, se generarán distintos modelos, los cuales corresponden a los algoritmos a analizar y distintos hiper parámetros a variar. Una vez se han realizado todas las combinaciones esperadas entre algoritmos e hiper parámetros, la distribución de las medidas de desempeño a considerar obtenidas por los modelos generados serán analizadas utilizando métodos estadísticos para la detección de outliers. Estos outliers detectados para cada distribución de las medidas de desempeño serán evaluados para la conformación del meta modelo.

Cuadro 2.2.1: Tabla de algoritmos e iteraciones. Las iteraciones se basan en el número de parámetros a variar.

Algoritmos y parámetros empleados en la etapa de exploración					
#	**Algoritmo**	**Tipo**	**Parámetros**	**Uso**	**Iteraciones**
1	AdaBoost	Ensamble.	Algoritmo. Número de estimadores.	Clasificación y regresión.	5
2	Bernoulli Naive Bayes	Probabilístico.	Default.	Clasificación.	1

Cuadro 2.2.1: Tabla de algoritmos e iteraciones. Las iteraciones se basan en el número de parámetros a variar.

Algoritmos y parámetros empleados en la etapa de exploración					
3	Decision Tree	Características.	Criterio y división.	Clasificación y regresión.	4
4	Gaussian Naive Bayes	Ensamble.	Default	Clasificación y regresión.	1
5	K-Nearest Neighbors	Distancias.	Número vecinos. Algoritmo. Métrica distancias. Peso.	Clasificación y regresión.	5
6	TensorFlow Keras	Redes neuronales.	Número de neuronas. Número de capas. Optimizador. Función de pérdida.	Clasificación y regresión.	12
7	Random forest	Ensamble.	Número de estimadores. Criterio. Bootstrap.	Clasificación y regresión.	10
				Total iteraciones	38

Para cada modelo, en la fase de entrenamiento, validación cruzada y testeo, se obtuvieron las medidas de desempeño accuracy, precisión, recall y f-score, las cuales fueron utilizadas para seleccionar los modelos generados. En primer lugar se filtraron los modelos generados de acuerdo con la performance observada en el proceso de entrenamiento, utilizando técnicas estadísticas de detección de outliers. De este universo de modelos seleccionados, se aplicó un segundo filtro por la performance obtenida en el proceso de testeo, utilizando de igual forma un método

estadístico de identificación de outliers. Luego, se revisaron las tasas de sobreajuste como forma de comprobación de que los modelos seleccionados no presentaran una diferencia importante entre la performance observada en el proceso de testeo, con respecto a la validación cruzada. Finalmente, se filtró el conjunto de modelos resultante, obteniendo solo aquellos únicos. Se consideró un modelo repetido aquel que fue seleccionado para dos distintas medidas de desempeño, por medio de los filtros aplicados.

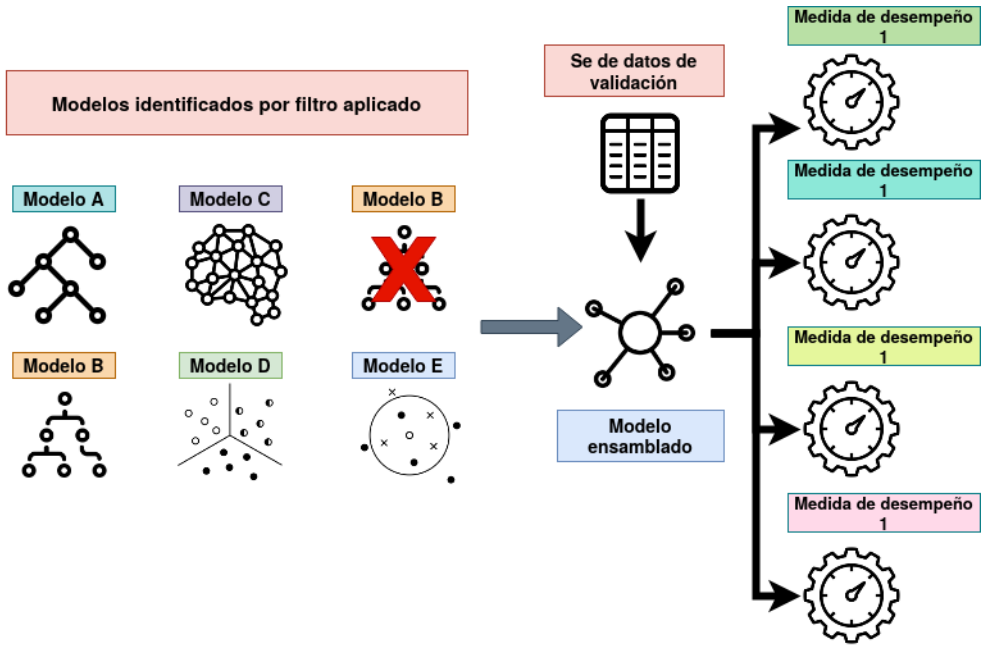


Figura 2.2.3: Esquema construcción del modelo ensamblado. Una vez han sido identificados aquellos modelos outliers por el test estadístico aplicado, estos fueron filtrados para obtener solo los modelos únicos. Luego, los modelos seleccionados fueron re entrenados, y se utilizó el set de datos de validación para obtener las medidas de desempeño correspondientes a los modelos ensamblados.

Finalmente, aquellos modelos obtenidos luego de esta evaluación fueron utilizados para la construcción del modelo ensamblado, el cual realiza predicciones mediante votación. En la Figura 2.2.3 se representa la forma en la cual fue construido el modelo ensamblado, utilizando los modelos únicos seleccionados en base a su desempeño. Por otra parte, en la Figura 2.2.4 se muestra un esquema de la forma en la cual se realizan las predicciones dentro del modelo ensamblado. Mediante votos entre los modelos que componen al modelo ensamblado, se seleccionan aquellas predicciones que posean mayor número de votos. Finalmente, se realizó una comparación entre las medidas de performance obtenidas para cada modelo individualmente, y para el modelo ensamblado.

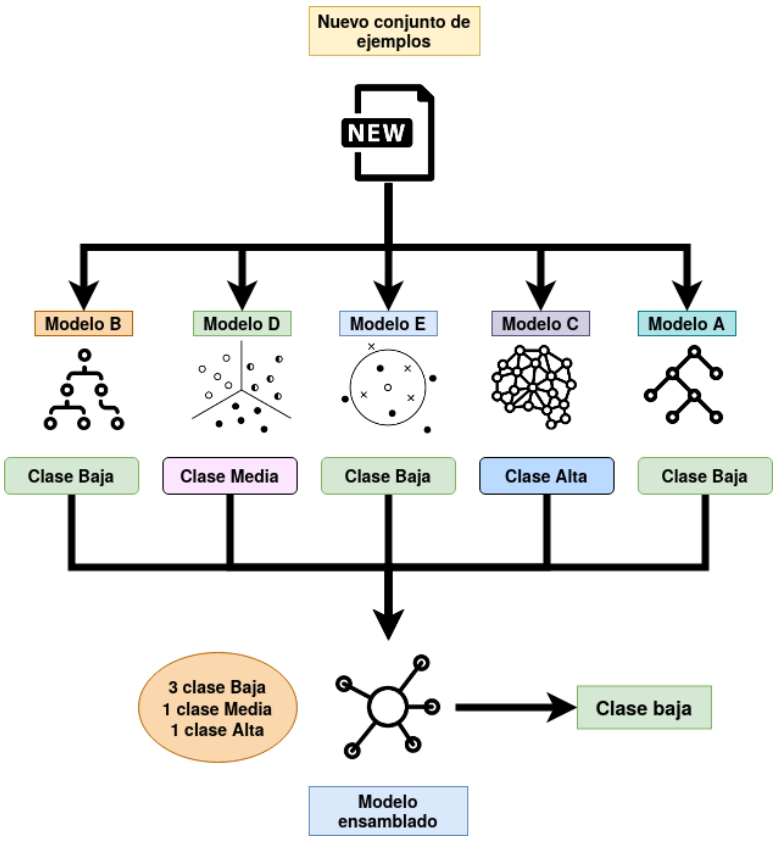


Figura 2.2.4: Esquema predicción en modelo ensamblado. Al ingresar un nuevo ejemplo, cada modelo que compone el modelo ensamblado realiza una predicción con respecto a la clase de interacción. Cada una de las predicciones realizada por los modelos se considera como un voto, y aquellas predicciones que sean mayoritarias, o en otras palabras, la predicción que cuente con mayor número de votos, corresponde a la predicción derivada del modelo ensamblado.

2.2.4. Comparar y evaluar estrategias de codificación.

Con el propósito de comparar los resultados obtenidos al utilizar codificación con embedding generada con TAPE (Rao et al., 2019) con respecto a otro tipo de codificaciones, se aplicaron distintas técnicas de codificación de secuencias. Se realizó la codificación de las secuencias utilizando one hot encoder, ordinal encoder, frecuencia de residuos y propiedades fisicoquímicas. La codificación de One Hot fue implementada utilizando módulos de Pandas (McKinney et al., 2011). En el caso de Ordinal encoder, el orden utilizado en la codificación fue otorgado en orden alfabético de los aminoácidos. Por otra parte, la codificación en base de frecuencias se basó en la cantidad de veces que se observó cada aminoácido a lo largo de la secuencia en cuestión. En cuanto a las propiedades

fisicoquímicas a considerar, estas han sido obtenidas desde la base de datos AAIndex (Kawashima and Kanehisa, 2000), en la cual se han reportado variadas propiedades fisicoquímicas y bioquímicas de aminoácidos o pares de aminoácidos. La selección de las propiedades fisicoquímicas a consideradas ha sido realizada por el co-tutor, en base a su trabajo realizado en Medina-Ortiz et al. (2020a). Ejemplos de estas codificaciones se muestran en la Figura 2.2.6

Utilizando cada una de las metodologías mencionadas anteriormente se codificaron las secuencias de anticuerpos y antígenos, y se relacionaron los vectores resultantes con respecto a las interacciones presentadas en el set de datos. Posteriormente, estos datos fueron divididos en conjuntos de entrenamiento y validación, siguiendo el protocolo señalado en el punto anterior. 2.2.3

Para cada set de datos generado mediante las codificaciones planteadas, se realizó nuevamente el proceso exploratorio de algoritmos e hiper parámetros, planteado en 2.2.3. Todos las codificaciones fueron sometidas a las iteraciones presentadas en la Tabla 2.2.1, registrando las performance observadas en el proceso de entrenamiento, validación cruzada y testeo. Luego, de acuerdo a la metodología señalada anteriormente, se aplicó un filtro de selección en base a técnicas de identificación de outliers a todo el conjunto de modelos generados, para obtener los posibles modelos participantes en el modelo ensamblado.

2.2.5. Corroborar y estimar escalabilidad del modelo predictivo.

Una vez fue seleccionada la combinación de codificación, algoritmos e hiper parámetros que generan la mejor clasificación de interacciones se evaluó la escalabilidad del modelo ensamblado desarrollado. Para esto, el set de datos codificado bajo la metodología seleccionada como más conveniente fue utilizado casi completamente para el entrenamiento de cada modelo, eligiendo sistemáticamente un anticuerpo, del cual se eliminaron del set de entrenamiento todas sus interacciones registradas con antígenos. A este método se le llamó Leave One Antibody Out, un esquema de este proceso se observa en la Figura 2.2.7. Estos datos fueron utilizados posteriormente para la evaluación del modelo ensamblado. Los valores obtenidos de medida de desempeño fueron registrados, y el proceso se repitió para cada anticuerpo en el set de datos. Mediante esta metodología, fue

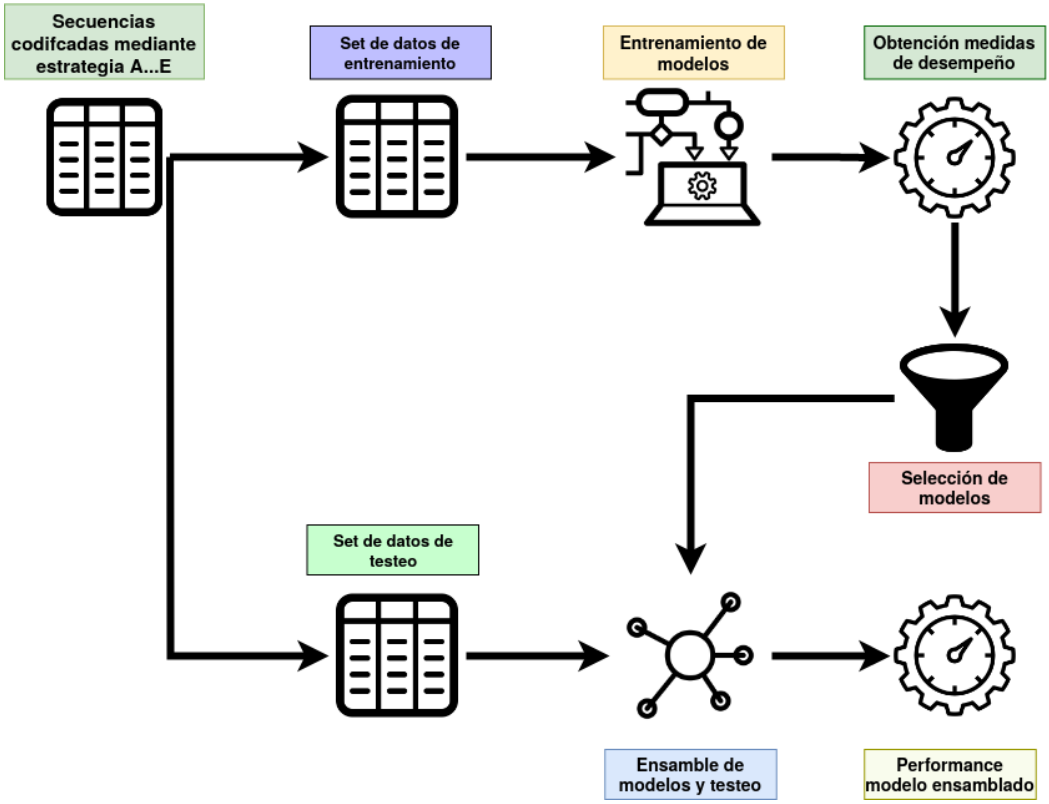


Figura 2.2.5: Esquema de evaluación de estrategias de codificación. El conjunto de secuencias codificado bajo las diversas estrategias planteadas será dividido en set de datos de entrenamiento y testeo. El conjunto de entrenamiento fue utilizado para la exploración de algoritmos e hiper parámetros, de cada cual se obtuvieron medidas de desempeño. Una vez terminada la etapa exploratoria, se seleccionaron los mejores modelos utilizando test estadísticos de identificación de outliers. Con los modelos seleccionados se construyo el modelo ensamblado, el cual fue validado con el segundo conjunto de datos y a partir del cual se obtuvieron las medidas de desempeño para el meta modelo generado.

posible observar la influencia de ciertos anticuerpos en el entrenamiento de cada modelo, y la performance observada en dependencia de estos. Esta metodología fue aplicada en todos los modelos integrantes del modelo ensamblado, utilizando el mismo set de datos de entrenamiento y evaluación.

2.2.6. Herramientas.

En la Tabla 2.2.2 se presentan las herramientas y librerías a utilizadas para el cumplimiento de los objetivos planteados. Las tareas y actividades relacionadas con programación fueron realizadas utilizando el lenguaje Python v3.6, de forma que el pipeline diseñado sea coherente y fácil de conectar. Todas las herramientas

Secuencia: MKQYLELMQKVLDEGTQKNDRTGTGTLSIF...

		D (1)	E (2)	F (3)	G (4)	I (5)	K (6)	L (7)	M (8)	Q (9)	S (10)
One hot encoder:		0	0	0	0	0	0	0	1	0	0
		0	0	0	0	0	1	0	0	0	0
		0	0	0	0	0	0	0	0	1	0
Ordinal encoder:		8	6	9	...	..	...	7	8	9	6
Frecuencia :		2	2	1	3	1	3	4	2	3	1
Propiedades fisicoquímicas	A	169.2	168.6	143.8	193.6	166.7	138.4	166.7	169.2	143.8	168.6
A) Volumen aa. B) Masa molecular aa. C) Índice hidropatía aa.	B	149	146	146	181	131	147	131	149	146	146
	C	1.9	-3.9	-3.5	-1.3	3.8	-3.5	3.8	1.9	-3.5	-3.9

Figura 2.2.6: Ejemplos de codificaciones utilizadas. Se implementaron las estrategias de One Hot encoder, Ordinal encoder, codificador de frecuencia y propiedades fisicoquímicas. Estas estrategias utilizan diversas metodologías para transformar la misma secuencia, generando distintos vectores.

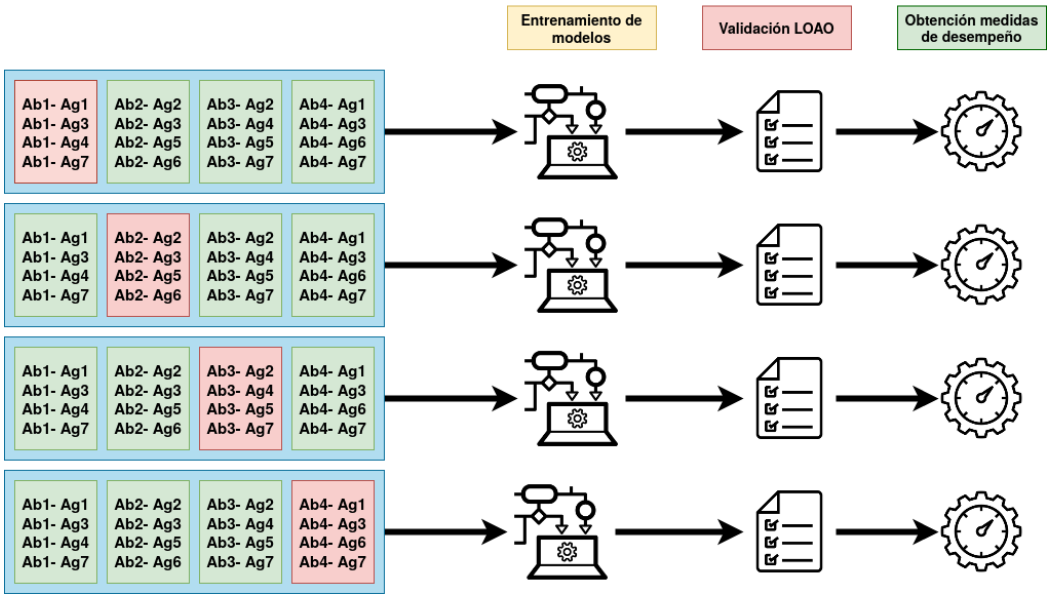


Figura 2.2.7: Esquema proceso Leave One Antibody Out.En primer lugar, se selecciona un anticuerpo y sus interacciones. Todos los anticuerpos, a excepción del seleccionado, son utilizados para entrenar. Luego, el anticuerpo no utilizado en el entrenamiento es empleado como set de datos de testeo. A partir de esto se obtienen las medidas de desempeño.

reportadas en esta tabla son gratuitas.

Cuadro 2.2.2: Tabla de herramientas utilizadas durante el desarrollo de la metodología planteada

Herramienta	**Uso**
Trello (Atlassian, 2011)	Organización de tareas a realizar.
Python 3 (Foundation, 2008)	Lenguaje de programación base.
TAPE (Rao et al., 2019)	Codificación de secuencias basada en embedding.
DMAKit-lib (Medina-Ortiz et al., 2020b)	Exploración de algoritmos de aprendizaje supervisado e hiper parámetros.
Pandas (McKinney et al., 2011)	Análisis set de datos. Limpieza set de datos.
NumPy (Walt et al., 2011)	Manejo de vectores. Limpieza de datos. Análisis estadístico.
Plotly (Inc., 2015)	Generación de gráficos.
SciPy (Virtanen et al., 2020)	Análisis estadístico

2.3. Resultados y discusión.

El conjunto de datos utilizado durante este proceso presenta 262,412 ejemplos, correspondiente a las interacciones entre 45 secuencias de cadenas pesadas de anticuerpos y 8,186 secuencias de auto-antígenos de leucemia. De las tres clases establecidas anteriormente, 65,603 ejemplos corresponden a interacciones de nivel Alto, 131,205 a interacciones de nivel Medio, y 65,604 a ejemplos de interacciones de nivel Bajo. La distribución de estas interacciones es similar entre los anticuerpos analizados, como se muestra en la Figura 2.3.1. Sin embargo, algunos anticuerpos presentan una distribución anormal del número de datos por clase, como es el caso de A003, el cual presenta un número de interacciones de clase Alta similar a la cantidad de ejemplos con interacción de clase media. Por otra parte, los anticuerpo A097, C003 y A021, presentan un mayor número de ejemplos con interacción Alta que el resto de los anticuerpos entregados. Incluso, en el caso de A097 y A021, el número de ejemplos con esta clase de interacción supera a los correspondientes a la clase media, que es la más común dentro del universo de anticuerpos estudiados. Es posible que esta diferencia se deba a la intensidad de la interacción que generan

estos anticuerpos, los cuales al parecer forman interacciones más fuertes que el resto de los anticuerpos.

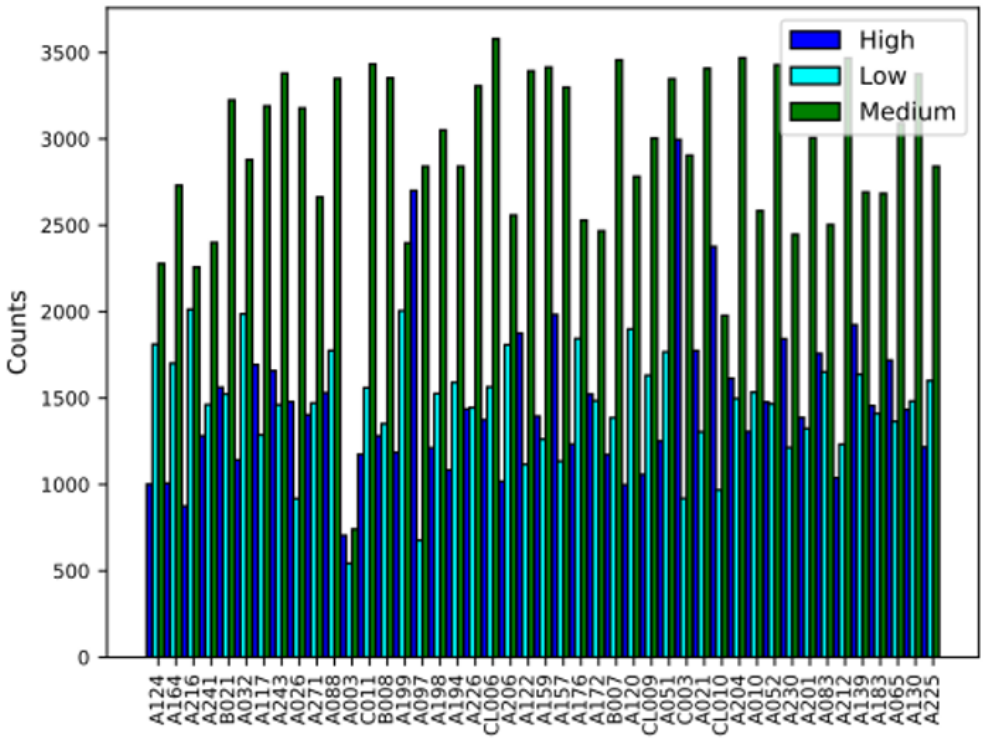


Figura 2.3.1: Número de ejemplos de cada clase por anticuerpo. La categorización realizada sobre la intensidad de la interacción generó una distribución similar de ejemplos de clase baja, media y alta para cada anticuerpo. Sin embargo, en algunos casos como A003, A097, C003 Y A021 la cantidad de interacciones observadas es menor, o poseen clases des balanceadas con respecto a lo observado en el resto de anticuerpos.

Debido a estas diferencias significativas con respecto a la distribución de clases en los anticuerpos mencionados anteriormente,se evaluó la similitud de las secuencias mediante un alineamiento multiple realizado con Clustal Omega (Sievers and Higgins, 2014). Esta herramienta permitió observar los porcentajes de similitud entre los anticuerpos analizados, siendo C003 el anticuerpo con menor identidad entre los seleccionados, con porcentajes de identidad entre 45 % y 49 %, siendo este máximo de porcentaje de identidad el obtenido contra A021. Por otro lado, el anticuerpo A003 y A021 son los más similares dentro de este grupo, alcanzando un 86 % de identidad, mientras que A021 y A097 poseen un 81 % de identidad. A partir de los resultados de este alineamiento múltiple se realizó un logos (Schneider and Stephens, 1990), para observar posibles aminoácidos conservados que explicaran la diferencia de distribución de clases, el cual se muestra en la Figura 2.3.2. No se

identificaron grandes diferencias con respecto a los aminoácidos conservados en el universo de anticuerpos analizados. No obstante, se distinguió la preferencia de leucina sobre valina en 5 posiciones dentro de la posible región conservada en estos anticuerpos. Es posible que, el aumento del largo de la cadena lateral posibilite un mayor número de interacciones electrostáticas débiles, o estabilice de mejor forma la estructura de estos anticuerpos, orientando los residuos que interaccionan dentro de la región hipervariable de mejor forma, posibilitando así mayor número de interacciones con los auto antígenos. En la Figura 2.3.3 se observa el logos generado a partir del alineamiento múltiple de todos los anticuerpos proporcionados. Se aprecia un conjunto de aminoácidos conservados en estas cadenas pesadas, entre los cuales destacan aminoácidos con anillos o cadenas laterales cargadas. Entre estos destacan residuos arginina, triptófano, prolina, glutamina, ácido aspártico y ácido glutámico. Además, entre las posiciones 54 y 64, y 105 a 125, se observa una alta variación a nivel de secuencia, lo cual puede indicar que estas secciones corresponden a las regiones hiper variables de los anticuerpos. Por otra parte, en anticuerpos con mayor número de interacciones de clase Baja, se observó una preferencia por aminoácidos no cargados en la región hipervariable, con respectos al resto de anticuerpos. Esto puede influir en el número de interacciones posibles entre estos anticuerpos y los auto antígenos con los cuales interaccionan.

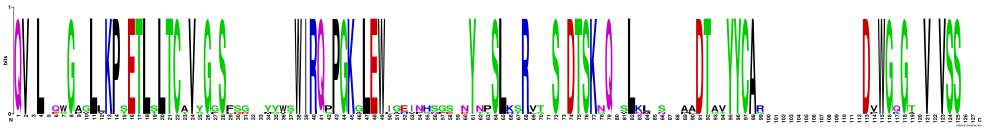


Figura 2.3.2: Logos conjunto de anticuerpos analizado. Se presenta la conservación de aminoácidos por posición en el conjunto de anticuerpos analizados. De forma general, se observan los mismos aminoácidos conservados con respecto al universo de anticuerpos proporcionados. No obstante, existe una preferencia por el aminoácido leucina, en posiciones que dentro del universo corresponden de igual forma a valinas.

Desde otro enfoque, se analizaron estos anticuerpos desde un punto de vista estructural. Entre los anticuerpos con una distribución singular, con mayor número de interacciones con clase Alta, se seleccionó al anticuerpo A021 para examinar las características de su estructura e interacción como anticuerpo. Como punto comparativo, se seleccionó al anticuerpo A120, el cual posee una distribución de clases similar a la observada de forma general. Como antígeno comparativo, se seleccionó el antígeno uORF:IOH38079, el cual posee interacción con ambos

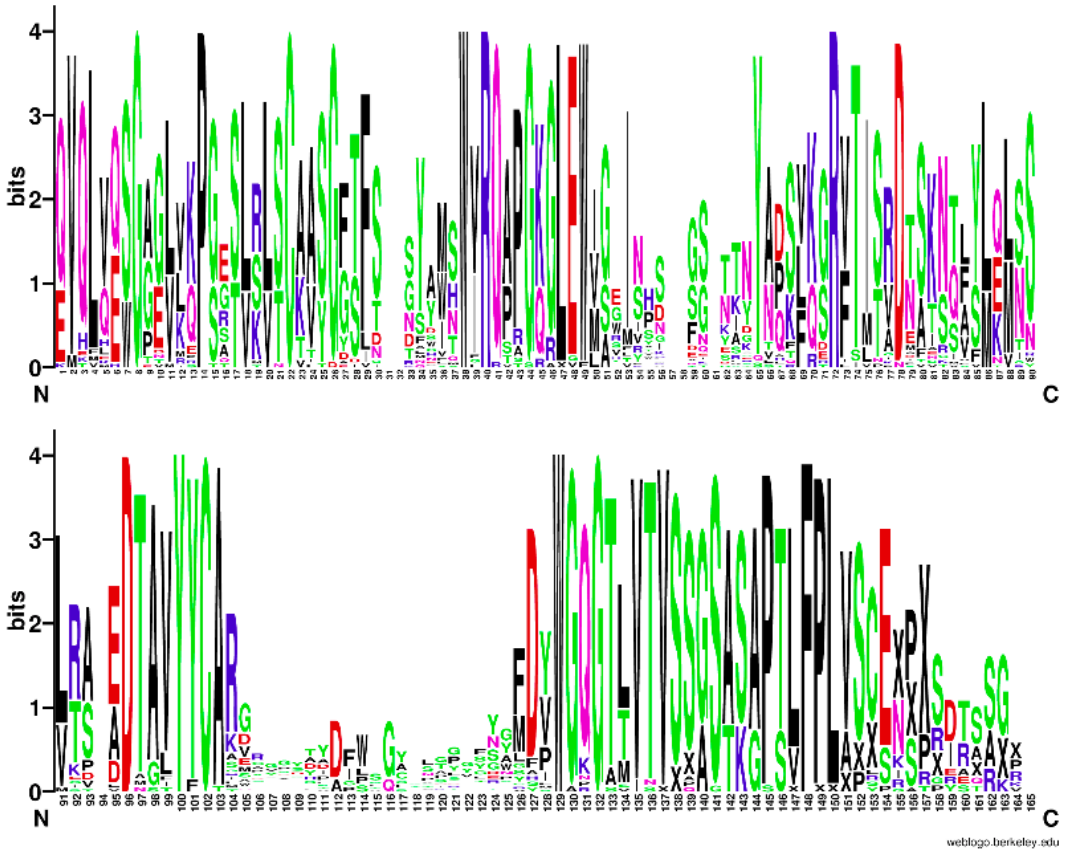


Figura 2.3.3: Logos conjunto completo de anticuerpos. Se presenta la conservación de aminoácidos por posición en la totalidad de los anticuerpos facilitados. Se visualizan aminoácidos muy conservados dentro de estas secuencias, correspondiendo a glicinas, triptófanos, arginina, ácido aspártico y glutámico, fenilalaninas y serinas. Además, se aprecia una región poco conservada entre los aminoácidos 104 y 126, lugar en el que podría encontrar la región hiper variante.

anticuerpos, ambas catalogadas como fuertes. Sin embargo, con distintos niveles de intensidad. Dado que ninguno de estos anticuerpos presenta una estructura reportada en PDB, se utilizó el servidor de Swiss Model (Schwede et al., 2003), el cual permitió obtener los modelos para estas moléculas. Se evaluó la calidad de los modelos generados mediante gráficos de Ramachandran (presentados en material suplementario), seleccionado aquellos donde todos o casi todos los aminoácidos se encuentren dentro de las regiones de ángulo permitidas. Luego, mediante el servidor CPORT (de Vries and Bonvin, 2011) se determinaron los aminoácidos dentro de la interfaz de contacto, los cuales fueron utilizados como parámetros para el acoplamiento molecular. Mediante el software Haddock2.2 (Van Zundert et al., 2016) se realizó el docking de cada anticuerpo con el antígeno seleccionado,

estableciendo como residuos pasivos y activos las predicciones entregadas por CPORT.

El complejo obtenido por la proteína A021 y uORF:IOH38079 se presenta en la Figura 2.3.4. En color gris se presenta a la estructura obtenida para el antígeno, mientras que el anticuerpo se presenta en color azul. En la figura se representa la estructura secundaria y la superficie de ambas proteínas. Dentro de las regiones presentadas en un recuadro se observan las posibles interacciones entre el antígeno y el anticuerpo, las cuales fueron divididas en dos secciones. Dentro del recuadro rojo se presentan dos posibles interacciones, entre los residuos ASN105 y TRP110 de A021, y ARG398 Y ARG399 de uORF:IOH38079. La interacción entre ASN105 y ARG398 posee una distancia de 3,76 Å, entre el hidrógeno de ARG398 y el oxígeno de ASN105. Esta distancia podría indicar un enlace puente hidrógeno entre ambas proteínas. Por otra parte, el residuo TRP110 y ARG399 poseen una distancia de 3,86 Å. La distancia fue medida entre el carbono mas cercano en el anillo del triptófano y uno de los nitrógenos de la cadena de arginina. Es posible que, dada la distancia y la naturaleza del anillo, esta corresponda a una interacción π-catión. En cuanto a la sección analizada dentro del recuadro negro, se observan tres posibles interacciones, correspondientes a los residuos THR117, SER119 Y SER120 del anticuerpo A021, y LYS385, ASP374 y ARG377 del antígeno. Los residuos SER120 y THR117 presentan enlaces mediados por oxigeno, entre los residuos ARG377 y LYS385 respectivamente. Estos enlaces podrían corresponder a puentes hidrógenos, a pesar de la distancias medidas en el modelo de docking generado, correspondientes a 2,12 Å y 1,81 Å respectivamente. Por otra parte, el residuo SER119 interacciona por medio de su hidrógeno con el oxigeno del residuo ARG374, con una distancia de 2,73 Å. Todas estas interacciones débiles parecen contribuir en la intensidad de la interacción presentada por A021 frente al antígeno uORF:IOH38079.

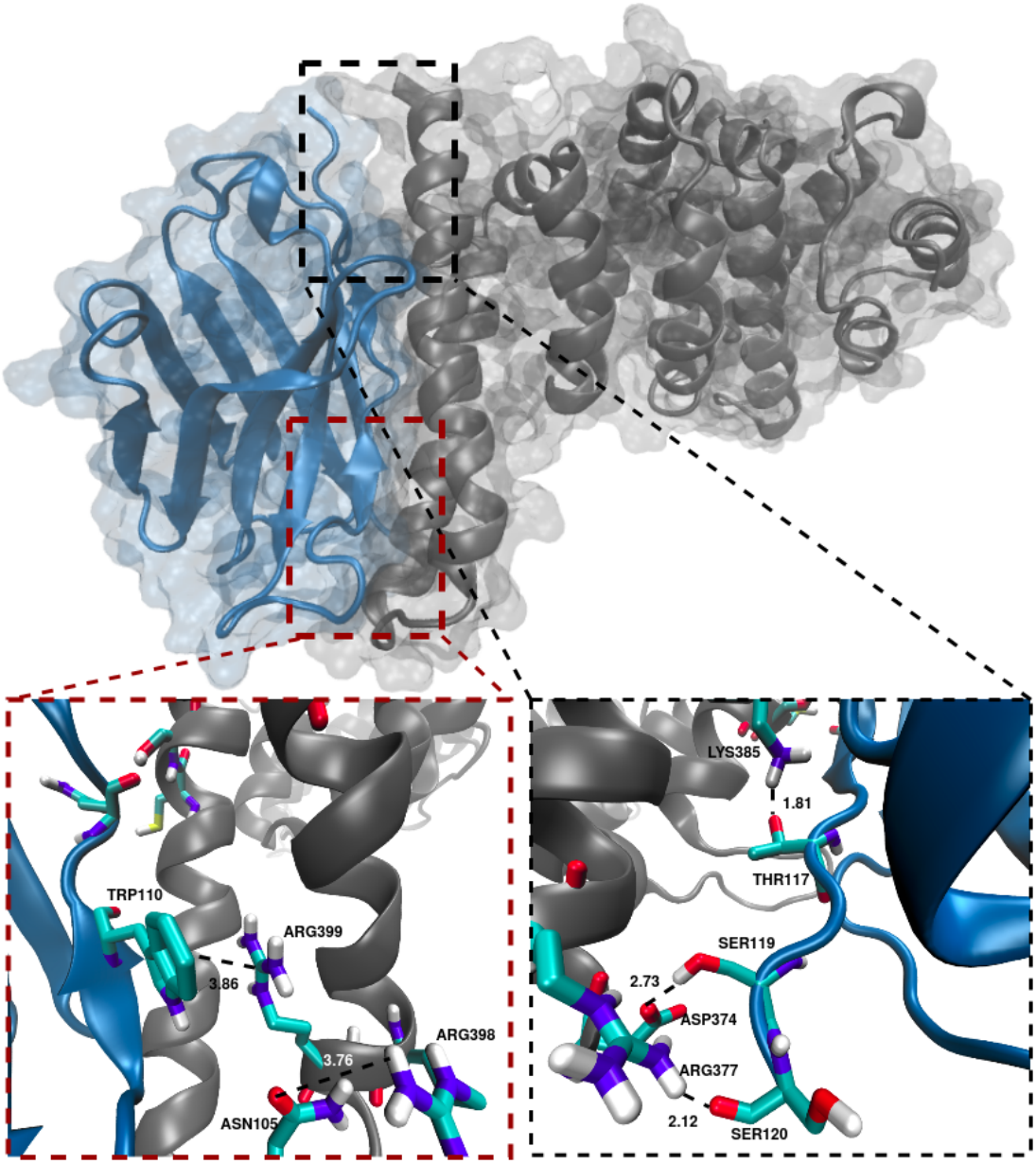


Figura 2.3.4: Docking anticuerpo A021 y antígeno uORF:IOH38079. En la parte superior se observa el docking generado por Haddock 2.2 entre el anticuerpo A021, representado en color azul, y el antígeno uORF:IOH38079, representado en color gris. En los recuadros inferiores se detallan las posibles interacciones observadas en el complejo, donde los residuos participantes se encuentran representados utilizando el metodo de dibujado licorice, y con los átomos coloreados de acuerdo a su tipo. Un total de 6 posibles interacciones fueron observadas, correspondientes a 5 enlaces mediante puente hidrógeno, y una probable interacción pi-catión, entre el residuo triptófano 110 del anticuerpo y el residuo arginina 399 del antígeno.

Por otra parte, el complejo resultante entre el anticuerpo A120 y el antígeno uORF:IOH38079 se presenta en la Figura 2.3.5. De manera similar, el anticuerpo

se presenta en color azul, mientras que el antígeno es representado por el color gris. Se presenta la estructura secundaria, y la superficie de cada proteína. En este complejo se observaron un conjunto de posibles interacciones, presentadas en los recuadros negro y rojo. En el recuadro negro se observa la potencial interacción entre el residuo LYS43 del anticuerpo, y el residuo ASP374 del antígeno, la cual sería mediada por hidrógeno y oxigeno de cada residuo, respectivamente, con una distancia de 1.61 Å. En cuanto al recuadro rojo, se presenta un total de 4 posibles interacciones, mediadas por los residuos PHE105, THR115, SER117 Y SER118 correspondientes al anticuerpo A21, y los residuos ARG399, ALA410, ASP413 Y ASP414 del antígeno uORF:IOH38079. La interacción PHE105 - ARG399 posee una distancia de 4.04 Å. No obstante, la orientación entre el nitrógeno del residuo de arginina y el anillo de fenilalanina no evidencia una interacción π-catión. Las probables interacciones SER117 - ASP413 y SER118 - ASP414 se encuentran mediadas entre los hidrógenos de las serinas y los oxígenos de los residuos ácido aspártico, con una distancia de 1.69 Å y 1.65 Å respectivamente. Finalmente, otra posible interacción se observa entre el residuo THR115 y ALA410, con una distancia de 3.02 Å.

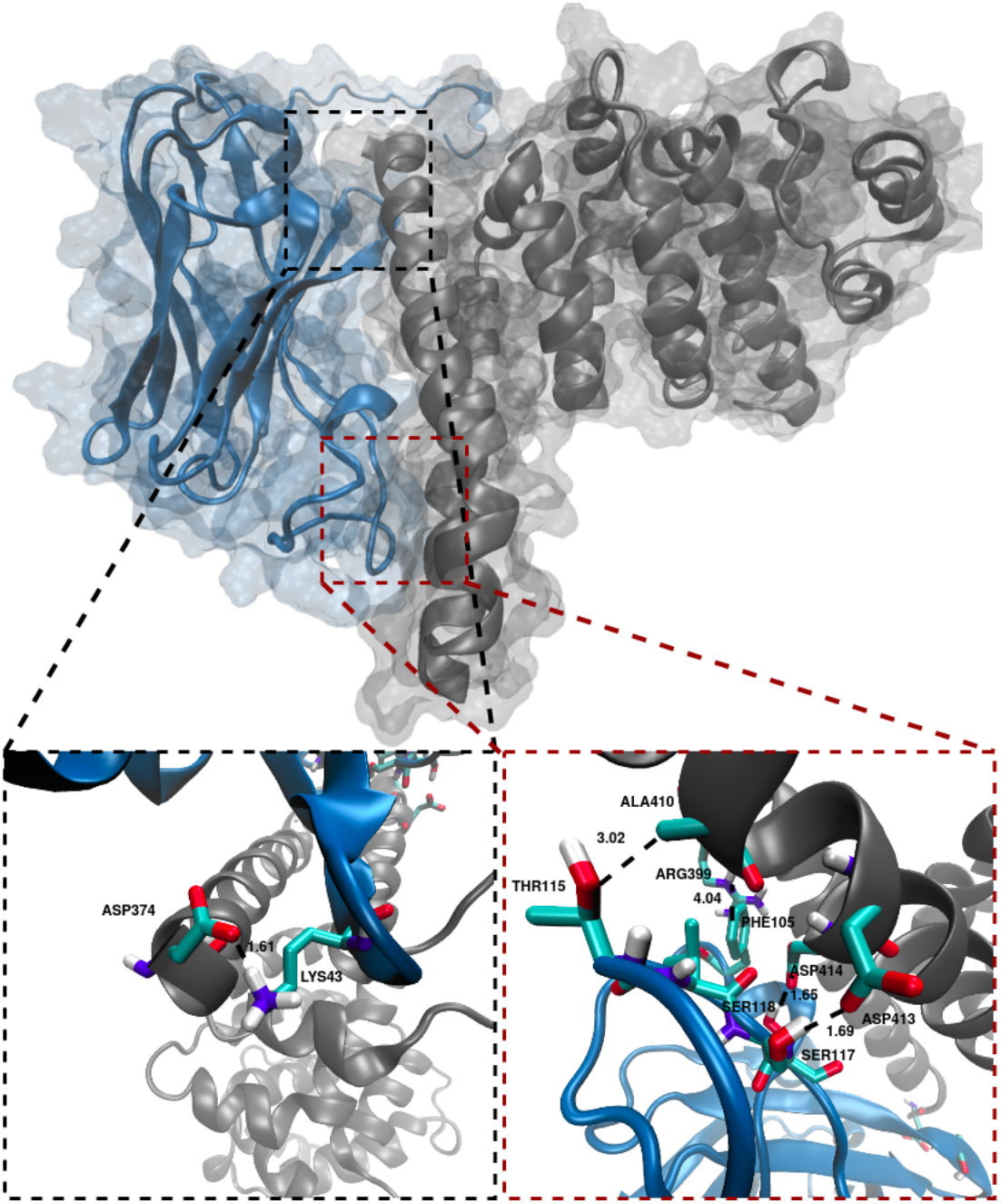


Figura 2.3.5: Docking anticuerpo A120 y antígeno uORF:IOH38079. En la parte superior se observa el docking generado por Haddock 2.2 entre el anticuerpo A120, representado en color azul, y el antígeno uORF:IOH38079, representado en color gris. En los recuadros inferiores se detallan las posibles interacciones observadas en el complejo, donde los residuos participantes se encuentran representados utilizando el metodo de dibujado licorice, y con los átomos coloreados de acuerdo a su tipo. Un total de 5 posibles interacciones fueron observadas, correspondientes a 4 enlaces mediante puente hidrógeno, y una posible interacción pi-catión, entre el residuo fenilalanina 105 del anticuerpo y el residuo arginina 399 del antígeno. Sin embargo, la orientación del anillo con respecto al nitrógeno de arginina no permite formar esta interacción.

A nivel de interacciones, se observaron al menos 2 menos en el anticuerpo A120 en comparación contra el anticuerpo A021. Si bien no se observó una relación directa entre la preferencia por leucina en lugar de valina y las interacciones formadas, si es posible que tenga relación respecto a la orientación en los aminoácidos. De forma general, se percibió que el modelo obtenido para el anticuerpo A021 posee estructuras secundarias más definidas, en comparación al anticuerpo A120. Además, en el caso de este último anticuerpo, el modelo seleccionado presenta un mayor porcentaje de coils y estructuras no definidas. Finalmente, a nivel de secuencia se evidenció una deleción de un conjunto de 4 aminoácidos, entre los cuales se encuentra el triptófano que posibilita la interacción con el antígeno. Mediante un alineamiento estructural, se determinó un RMSD de 1.53 Å y un porcentaje de identidad de 32 %. Estas diferencias estructurales y de secuencia podrían explicar las diferencias respecto al nivel de interacción, y al número de interacciones pertenecientes a la clase alta para los anticuerpos similares a A021.

De acuerdo con lo planteado en la metodología todos los ejemplos presentes en el set de datos fueron codificados bajo las diversas estrategias mencionadas anteriormente, generando vectores de diverso largo. Estos vectores incluyen el vector codificado del anticuerpo, el vector codificado del antígeno, y la clase correspondiente a la interacción entre estos. En los casos necesarios, de acuerdo con la codificación, los vectores de anticuerpos y antígenos generados fueron estandarizados, de forma que el set formado contara con un número constante de columnas o features. La excepción corresponde a los vectores codificados utilizando Embedding. Esto se debe a que esta técnica de codificación, al utilizar la red neuronal entrenada en secuencias, entrega un vector de largo constante que contiene la información referente a la secuencia, independiente del largo de aminoácidos de ésta. El largo de los vectores generados por cada estrategia de codificación se muestran en la Figura 2.3.6. En total, se obtuvieron un total de 11 set de datos. De estos, 8 corresponden a datasets obtenidos mediante la codificación por propiedades fisicoquímicas señaladas propuestas por Medina-Ortiz et al. (2020a), 2 a las codificaciones basadas en embedding, y el último set de datos corresponde al obtenido mediante One Hot encoder.

La estrategia de codificación por Ordinal enconder fue descartada, debido a que los vectores generados no eran informativos. Esto se basa en que, al otorgar una relación de orden arbitraria sobre los aminoácidos que componen una secuencia,

Codificación	Largo vector anticuerpo	Largo vector antígeno	Largo vector ejemplo
Embedding Babbler	1900	1900	3801
Embedding Bert-base	768	768	1537
Frecuencia de aminoácidos	156	1786	1943
One hot	3120	35720	38841
Ordinal	156	1786	1943
Propiedades fisicoquímicas	156	1786	1943

Figura 2.3.6: Largo de vector por codificación. En la figura se presenta el largo de los vectores codificados por cada una de las estrategias seleccionadas. El largo del vector ejemplo corresponde a la suma del largo del vector de anticuerpo, el vector de antígeno y la clase respuesta

no se entrega información real con respecto a las propiedades de éste al modelo a entrenar. De forma similar, la codificación en base a frecuencia también fue descartada de los set de datos utilizados para el entrenamiento. EL motivo por el cual no se utilizó esta codificación es que la frecuencia, al igual que Ordinal encoder, es poco informativa. Además, de acuerdo literatura, hace al menos 8 años que esta codificación no se utiliza por si sóla. Se observó una diferencia respecto al largo de vector obtenido mediante los modelos seleccionados para embedding. Mientras el modelo pre entrenado Babbler1900 entregó vectores de largo 1,900, el modelo bert-base proporcionó vectores de largo 768. Esta diferencia se debe a la arquitectura implementada por los modelos seleccionados. Por otro lado, la codificación por One Hot es la que presenta el vector resultante de mayor largo. Esto se debe a la estrategia utilizada por este codificador, en el cual, cada posición dentro de la secuencia es representada por un conjunto de 20 columnas. Finalmente, es necesario señalar que a partir de la codificación por propiedades fisicoquímicas se obtuvieron un total de N + M donde N corresponde al mayor largo de la secuencia de antígeno y M corresponde al mayor largo de la secuencia de la cadena pesada de anticuerpo.

A partir de estas codificaciones se entrenaron un total de 413 modelos, los cuales fueron desarrollados empleando los algoritmos mencionados en la Tabla 2.2.1. Para todos los modelos se realizó una validación cruzada de $k = 5$, con el fin de evaluar la persistencia de la performance, prevenir el sobreajuste y determinar las características principales de la generalidad de los modelos. En la Tabla 2.3.1 se muestra la totalidad de modelos generados por algoritmo utilizado, independientemente del dataset empleado para su entrenamiento y validación. Dado al número de hiper-parámetros modificables, los algoritmos con mayor

número de modelos generados corresponden a Neural networks y Random forest. De la misma forma, el menor número de modelos generados corresponde a Gaussian Naive Bayes y Benoulli Naive Bayes.

Cuadro 2.3.1: Tabla de modelos por algoritmo. Se presenta el número total de modelos generados por algoritmo utilizado.

Algoritmo	N° modelos generados
Neural Networks	132
Random Forest	110
AdaBoost	55
K Nearest Neighbors	50
Decision Tree	44
Gaussian Naive Bayes	11
Bernoulli Naive Bayes	11
Total	413

En el caso de la codificación por OneHot, no se realizó el entrenamiento de modelos utilizando el algoritmo de K-Nearest Neighbors. En primer lugar, esta decisión se basa en el tamaño de los vectores obtenidos por esta metodología. Además de esto, el cálculo de distancias entre los puntos que componen a estos vectores se transforma en un problema debido a la forma en la cual se representan los datos. De esta forma, ciertas combinaciones de vectores no relacionados pueden dar la misma distancia y afectar en el desempeño del modelo incrementando las tasas de error.

El desempeño de los modelos observados es bastante variado. siendo los algoritmos Bernoulli Naive Byaes, Gaussian Naive bayes y AdaBoost los que presentan las performance promedio más bajas, sin importar la codificación utilizada. Por el contrario, los algoritmos con el mejor desempeño corresponden a Neural networks y Random forest. En la Figura 2.3.7 se presentan los valores promedios obtenidos para estas métricas en el proceso de validación cruzada. La performance más alta observada en este proceso corresponde a 60 %, obtenida mediante algoritmos de redes neuronales. Por otro lado, random forest presenta una performance cercana a 58 %. Los modelos obtenidos con Gaussian Naive Bayes presentan una performance cercana a 40 %, valores similares a los observados en modelos entrenados utilizando

Bernoulli Naive Bayes, que alcanzan una performance cercana a 42 %. Finalmente, los modelos entrenados bajo los algoritmos K-Nearest Neighbors y Decision Tree poseen una performance cercana a 50 %. Los valores de performance observados en el proceso de entrenamiento y testeo se presentan en el material suplementario.

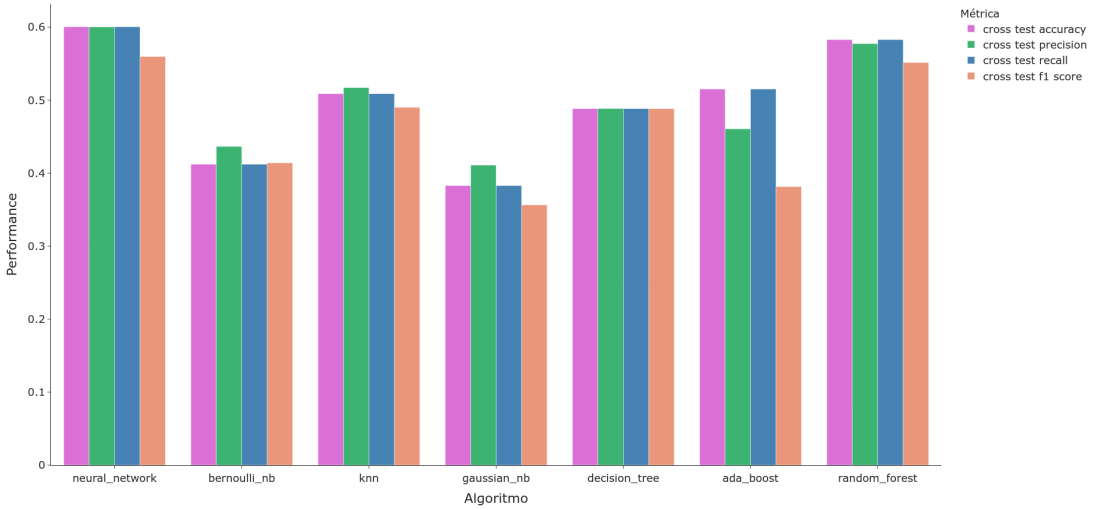


Figura 2.3.7: Performance promedio algoritmos. Se presenta el valor promedio obtenido de performance para cada métrica seleccionada, con respecto a los modelos generados por cada algoritmo, en la etapa de validación cruzada. Se presenta en color rosa el accuracy, en verde precision, en azul el recall y en naranjo el f-score.

De manera general, la exploración de muestra que los modelos predictivos generados presentan un desempeño deficiente. Esto puede deberse a que existe una redundancia de información entre las secuencias de antícuerpos, ya que presentan una similitud en términos generales, existiendo una mayor diferencia en términos de su región hipervariante. Para evaluar esto, diversas técnicas y metodologías pueden ser planteadas, tales como.

- Evaluar distancias numéricas entre los vectores de las secuencias de anticuerpos.
- Análisis de varianza de las características y matrices de correlación o mutual information para determinar las propiedades independientes.
- Maximizar la varianza del espacio muestral mediante una representación por técnicas de Principal Component Analysis.

Sin embargo, el mejoramiento del rendimiento, no sólo depende de la representación de las secuencias de anticuerpos. Si no también de cómo se represente la interacción en sí. Esto representa un problema complejo si sólo se contempla la existencia de secuencias lineales. Uno de los métodos propuestos para esto es generar las representaciones de la misma forma desarrollada y generar estrategias de convolución sobre los datos. Otro método implicaría el uso de representaciones de proteínas en espacios de frecuencia empleando técnicas de digital signal processing Medina-Ortiz et al. (2020a) y multiplicar las frecuencias de antígeno y anticuerpo para obtener un espectro de la relación de ambas señales, con el fin de identificar los efectos físicos que estas convoluciones generan sobre la representación. De manera más simple, se podrían aplicar estrategias de reducción de dimensionalidad sobre los conjuntos de datos, de manera independiente (antígneo y anticuerpo) o considerando ambos en conjunto. Finalmente, una estrategia factible pero con una gran demanda computacional implicaría el hecho de generar los modelos estructurales para cada secuencia de antígeno y de anticuerpo y los docking moleculares para obtener los complejos, y empleando estrategias de estructuras de grafos, representar las interacciones. No obstante, todas las estrategias comentadas representan un trabajo a futuro considerando como base lo planteado en este trabajo de memoria de título.

Mediante el proceso de selección de modelos, el cual implicó la identificación de modelos reconocidos como outliers dentro de la distribución de desempeño en etapa de entrenamiento, testeo, evaluación de sobreajuste y posterior filtrado por modelos únicos, se obtuvieron 14 modelos. De estos, 13 corresponden a modelos entrenados utilizando random forest, empleando los set de datos codificados mediante embedding. Además, de la selección surgió un modelo basado en neural networks, el cual fue entrenado considerando como input el dataset codificado mediante One Hot encoder. Este modelo destaca en cuanto a la performance observada en validación cruzada, alcanzando una performance cercana al 80 %. Por otra parte, los modelos seleccionados correspondientes a random forest presentan una performance cercana al 63 %.

En cuanto al análisis realizado a la tasa de sobreajuste. De forma general esta medida osciló entre 1 y 2.2. Para el modelo seleccionado generado mediante la codificación One hot, se obtuvo una tasa de sobreajuste de 1.16. Por otra parte, en el caso de los modelos entrenados utilizando embedding, la tasa de sobreajuste

se encuentra alrededor de los 1.6. Los modelos que presentaron una tasa de ajuste de 1 o cercana a 1, son aquellos que obtuvieron valores de performance bajos, tanto en etapa de entrenamiento como de testeo y validación. Es por esto que, a pesar de presentar una tasa de sobreajuste nula, estos modelos no pasaron el filtro aplicado de outliers con respecto a la distribución de valores en proceso de entrenamiento. En la Figura 2.3.8 se muestra la performance obtenida por cada uno de los modelos seleccionados en validación cruzada.

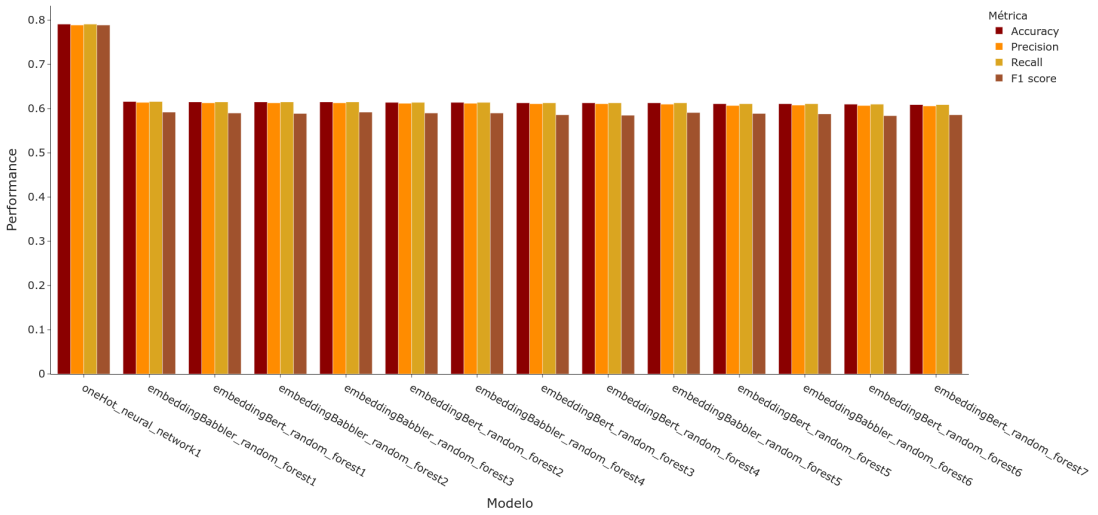


Figura 2.3.8: Performance validación cruzada.Para cada modelo seleccionado, se presentan los valores de performance obtenidos para el accuracy (rojo), precision (naranja), recall (amarillo) y f1 score (café), en el proceso de validación cruzada. A los modelos con igual algoritmo y codificación base se les otorgó un número distintivo.

Para evaluar la robustez de los modelos seleccionados, y su posible dependencia con respecto a la información relacionada a un anticuerpo en específico, se realizaron pruebas en los modelos seleccionados utilizando la metodología de Leave One Antibody Out. De esta forma, cada conjunto de modelo seleccionado fue re-entrenado, utilizando todo el set de datos codificado con la estrategia seleccionada, excepto los ejemplos correspondientes a un anticuerpo en específico. Siguiendo la metodología de entrenamiento aplicada anteriormente, se utilizó el 70 % para entrenamiento y el 30 % para el testeo, y se realizó una validación cruzada de k igual 5. Luego, los modelos generados fueron utilizados para predecir la clase de interacción de los ejemplos apartados correspondientes al anticuerpo en cuestión.

De acuerdo con los resultados observados en este proceso, los modelos no reflejan

una dependencia con respecto a las interacciones observadas entre un anticuerpo específico, y sus antígenos. Sin embargo, se contemplaron algunos anticuerpos que, al ser utilizados como set de datos de testeo, presentaron una menor performance. Uno de estos anticuerpos corresponde a A003. Como fue analizado anteriormente, este anticuerpo presenta una distribución diferente a la observada dentro del universo de interacciones estudiadas, con respecto a la cantidad de ejemplos con interacciones de clase Alta, Media y Baja. De forma similar, el anticuerpo A097 también presenta una disminución en la performance con respecto a lo evidenciado en el resto de los modelos. En los anticuerpos A124, A199, A206 y A216, la cantidad de ejemplos con interacción de clase Baja es mayor a la observada en la distribución general. Esto puede generar la diferencia con respecto a la performance observada en el resto de los modelos. A pesar de estas observaciones, las performance generadas al utilizar modelos mediante Leave One Antibody Out son frecuentemente similares a las observadas en los modelos entrenados utilizando el 70 % de los datos en entrenamiento. Esto indica que los modelos generados por las codificaciones e hiper parámetros seleccionados generan modelos robustos. La performance observada en uno de estos modelos entrenados, el cual corresponde a un modelo en base a random forest y codificación embedding-Babbler, es presentada en la Figura 2.3.9, como ejemplo del comportamiento general de los modelos entrenados bajo esta metodología.

Una vez fue corroborada la robustez de los modelos seleccionados, se realizó el ensamble de estos modelos. El ensamble utiliza una sistema de votación para generar la respuesta. De esta forma, al entregar un nuevo ejemplo al modelo ensamblado, cada uno de los modelos que lo componen realiza una predicción, y el resultado o predicción final corresponde a la clase con mayor número de votos. En este caso, el voto de cada uno de los modelos tiene el mismo peso. Se utilizaron todos los modelos seleccionados anteriormente para realizar nuevas predicciones sobre un conjunto aleatorio de ejemplos. Se evaluó el total de predicciones realizadas y se obtuvo de esta forma una predicción final. Así, se calcularon las performance en cada métrica propuesta para el modelo ensamblado, las cuales se observan en la Figura 2.3.10. Análisis de sensibilidad y especificidad fueron desarrollados para cada clase identificándose que la clase de interacción Baja es la que presenta menores valores de rendimiento en el modelo ensamblado, lo cual puede deberse a un desbalance de clases, o mal caracterización de la información, o a su vez,

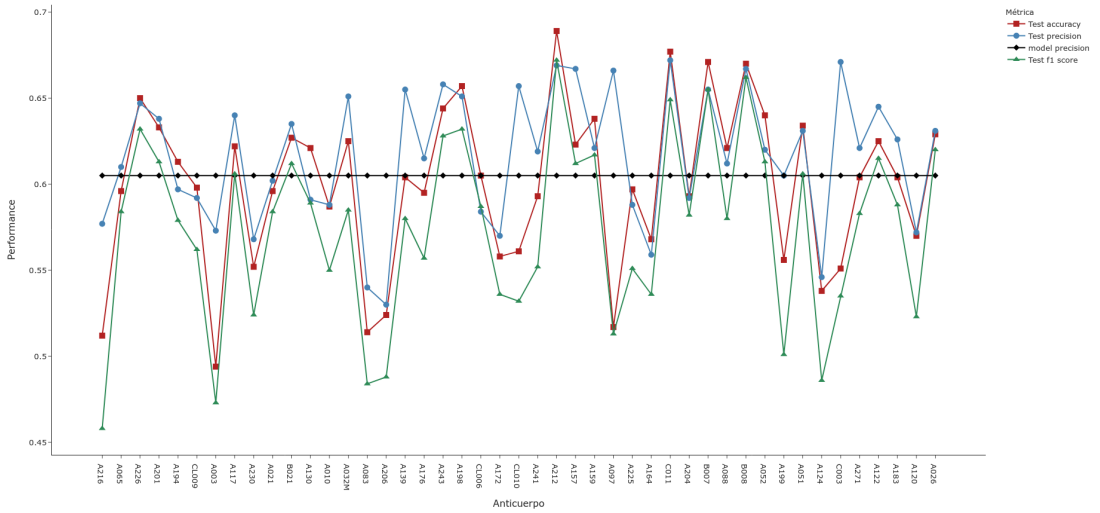


Figura 2.3.9: Performance Leave One Antibody Out. Se presenta la performance por anticuerpo obtenida mediante la metodología LOAO, en un modelo de random forest utilizando el dataset de codificación embeddingBabbler. En color rojo se presenta el accuracy en etapa de testeo por anticuerpo, en azul se presenta el valor de precision para la misma etapa, y en verde los valores observados para F1 score. En color negro se presenta el precision del modelo entrenado con todos los anticuerpos. De forma general, la performance se mantiene estable al eliminar algunos anticuerpos del set de datos de entrenamiento. Las excepciones presentes en esto pueden ser atribuidas a la diferencia en las distribuciones de clases presentes en estos anticuerpos.

que los ejemplos etiquetados como clase Baja tengan similitudes con otros para el conjunto de datos tratado. Aun así, el accuracy y el precision se ven incrementados notoriamente en comparación a los modelos predictivos individuales. En cuanto a las clases Media y Alta, todas las medidas de performance fueron aumentadas, alcanzando medidas de accuracy y precision sobre 80 %. Finalmente, la utilización del modelo ensamblado aumentó el accuracy promedio desde un 62 % a un 84.3 %, lo cual corresponde a una mejora significativa en la capacidad de predicción de las clases de interacciones entre estas moléculas.

El desbalance de clases en el set de datos entregado para evaluar el modelo ensamblado es visible en la Figura 2.3.11. Las clases Media y baja son las que presentan el menor número de ejemplos, correspondiendo a 41 y 150 respectivamente. En cambio, de la clase Baja se entregaron 309 ejemplos. En el conjunto de interacciones entregadas, la clase Baja y Alta cuentan con una cantidad muy similar de ejemplos, mientras que la clase Media posee más de el

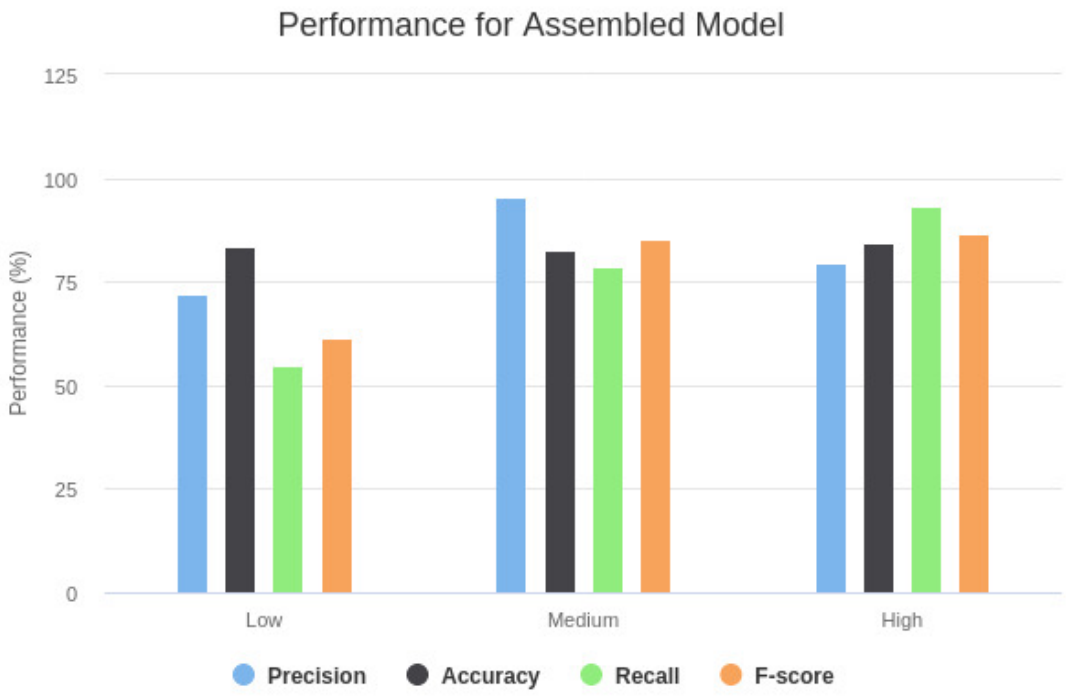


Figura 2.3.10: Performance modelo ensamblado. Se observa la performance obtenida por el modelo ensamblado para cada una de las clases presentadas. Las medidas de performance obtenidas para cada una de las clases aumentó con respecto a aquellas observadas en los modelos individualmente.

doble de ejemplos que cualquiera de las dos clases. Al realizarse una extracción aleatoria de ejemplos para evaluar el modelo ensamblado, se seleccionaron más ejemplos de la clase baja, por lo cual, el modelo ensamblado erró algunas de las predicciones realizadas sobre esta clase en el conjunto de datos utilizado para el testeo. Aún así, las predicciones realizadas por el modelo ensamblado en su mayoría corresponden a las clases correspondientes de los ejemplos, lo cual demuestra la efectividad del uso de esta metodología.

Un punto importante a destacar es que es posible evaluar este efecto desde un punto de vista estadístico, es decir, generar el proceso N veces con el fin de generar una distribución de las medidas de desempeño. No obstante, se dejará como trabajo a futuro o proyecciones debido al costo computacional que implicaría desarrollar este tipo de proceso y que presente una validez estadística.

Finalmente, a pesar de estas observaciones realizadas sobre ciertos anticuerpos que presentan una performance menor en validación LOAO, y el desbalance de clases observado en el set de datos utilizado para la evaluación del modelo ensamblado,

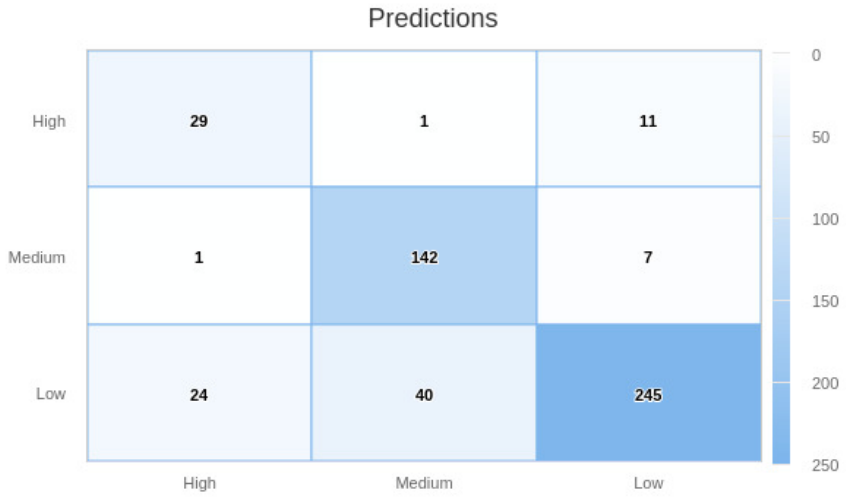


Figura 2.3.11: Predicción de clases modelo ensamblado. La matriz de confusión muestra el número de predicciones realizadas sobre una clase con respecto a la clase real de los ejemplos. La clase baja cuenta con el mayor número de ejemplos, lo cual no corresponde a la realidad del set de datos empleado para entrenamiento y validación. A pesar de esto, el número de predicciones realizadas sobre las clases reales es notable, lo cual se ve reflejado en el aumento de la performance en las métricas del modelo ensamblado en comparación a los modelos individuales

en los modelos generados mediante esta metodología se evidenció la capacidad de transferencia de aprendizaje. La metodología permite, de cierta forma, presentar nuevos ejemplos a los modelos seleccionados, los cuales no son entregados en el proceso de entrenamiento de los mismos. Es posible que un grupo interacciones, entre los 45 anticuerpos seleccionados, represente a la totalidad del universo de interacciones presentadas en el set de datos entregado. Sin embargo, sería necesario evaluar los limites de este universo, y la capacidad del modelo ensamblado diseñado para predecir nuevos ejemplos fuera de el conjunto proporcionado

Una de las características más importantes a la de desarrollar modelos predictivos es la usabilidad de estos en torno a nuevos ejemplos o nuevas tareas. Es decir, al entrenar un modelo para resolver una tarea específica, puede ese mismo modelo ser empleado para trabajar en una tarea similar. De manera análoga, es posible visualizar esto de manera tal de cómo se comportaría modificando o alterando los ejemplos de entrada. En este caso, el transfer learning se evaluó considerando la variación de performance entre el modelo genérico y los modelos individuales. No obstante, esto sólo es asociado a un componente del espacio latente que se

está trabajando, ya que, considerar el total de elementos posibles es imposible. No obstante, se puede proponer un método que permita estimar la variabilidad factible del espacio latente incluido dentro del conjunto de entrenamiento. Para ello, es necesario estimar métricas de comparación entre las secuencias a nivel distancia numérica, filogenética o frecuencia, dependiendo del espacio donde se encuentre trabajando y a partir de ello, generar distribuciones estadísticas punto a punto para definir zonas de confianza dentro de cada feature. Siendo un punto importante a analizar como propuesta a futuro.

2.4. Conclusiones

El conjunto de secuencias de anticuerpos empleado para el entrenamiento de modelos predictivos, correspondientes a cadenas pesadas de anticuerpos humanos, posee un variado porcentaje de identidad, desde el 38 % a incluso un 94 % de identidad. Se observó un conjunto de aminoácidos muy conservados, así como las posibles regiones hiper variantes en estos, que participan en las interacciones con los diversos antígenos. La clasificación entregada de las interacciones cataloga al 50 % de las interacciones observadas en el mundo de intensidades observadas como clase Media. El otro 50 % de las interacciones entregadas se dividen en igual proporción, en clase Baja y Alta. De forma general, todos los anticuerpos presentan una distribución de clases de interacciones similar, a excepción de un conjunto pequeño, que poseen un menor número de interacciones de forma general, o disponen de un conjunto mayor de interacciones clasificadas como Altas.

En cuanto a las diversas codificaciones aplicadas, se observaron diferencias tanto en el largo de estos vectores, como su performance sobre los algoritmos aplicados. En primer lugar se descartó la codificación de Ordinal encoder, debido a la falta de un contexto o información biológica en la codificación que fuese informativo para los algoritmos de machine learning. Luego, dada una revisión bibliográfica y la observación de las performance obtenidas en primera etapa por los modelos generados mediante la codificación de frequency, esta fue descartada. La codificación con respecto a frecuencia de un aminoácido, dentro de la misma secuencia y no como conjunto, puede ser información arbitraria para un modelo, y no presentar suficiente contexto o dimensiones claras para la separación de las clases correspondientes. Con respecto a los largos de vector, estos variaron entre

1,537 y 38,841, siendo este último el obtenido aplicando One Hot encoder.

Utilizando los set de datos obtenidos por cada estrategia de codificación implementada se generaron un conjunto de 413 modelos. Todas las codificaciones generaron un total de 38 modelos, a excepción de One Hot, del cual solo se generaron 33. Por otra parte, las performance obtenidas a partir de estas codificaciones y algoritmos implementados, en etapa de validación cruzada, varían desde 30 % a 80 %. Los algoritmos utilizados por los modelos con mejor performance corresponden a neural networks y random forest, mientras que los modelos con peor performance pertenecen a algoritmos Naive Bayes, tanto con distribución gaussiana como bernoulli.Mediante la aplicación de técnicas estadísticas de identificación de outliers se identificaron 14 modelos. De estos, 6 corresponden a modelos entrenados utilizados embedding Babbler, en un algoritmo de random forest, 7 corresponden a modelos entrenados con el set de datos embedding Bert-base, empleando el algoritmo de random forest. Además, se seleccionó un modelo entrenado usando ejemplos codificados mediante One hot encoder, en una estructura de redes neuronales. Estos modelos fueron utilizados para generar el modelo ensamblado por votación. De esta forma, se pudo aumentar la performance general desde 61 % a cerca de 81 %. Se evaluó la performance del modelo ensamblado frente a las 3 clases de interacciones presentadas, siendo la clase de interacción baja la con menor performance. No obstante la performance obtenida para esta clase también mejoró con respecto a los modelos individuales, alcanzando un performance promedio cercano a 73 %.

Los modelos utilizados para el desarrollo del modelo ensamblado fueron sometidos a la metodología de Leave One Antibody Out, con el fin de evaluar la robustez de los modelos generados, y el cómo escala la performance en estos al presentar nuevos ejemplos. Los resultados obtenidos mediante este procedimiento mostraron que, aun cuando la performance de los modelos no es destacable, alcanzando un 61 % aproximadamente, los modelos generados son robustos en su mayoría, no disminuyendo en gran medida la performance obtenida en ausencia de un anticuerpo y sus interacciones. Las excepciones observadas corresponden a anticuerpos que, usualmente, presentan distribuciones de clases diferentes a la observadas en el universo de interacciones analizadas. En mucho de estos casos, los anticuerpos poseen mayor número de ejemplos con interacciones de clase Baja, mientras que en algunos sucede la situación contraria, presentando más ejemplos

con interacciones de clase Alta. A pesar de esto, los modelos generados por esta metodología poseen una performance similar al modelo entrenado con ejemplos de todos los anticuerpos, por lo cual, estos modelos pueden ser considerados robustos, y se puede esperar un comportamiento similar ante nuevos ejemplos.

Dado que gran parte de los modelos seleccionados corresponden a modelos entrenados utilizando codificaciones embedding, y que la evaluación realizada sobre los modelos seleccionados probó que estos son robustos frente a nuevos ejemplos, es posible decir que la hipótesis planteada en este capitulo se cumple. La codificación mediante embedding permite una representación adecuada de las secuencias de anticuerpo y auto antígenos derivados de pacientes de leucemia. Esto posibilitó el entrenamiento de un conjunto de modelos que, a pesar de que individualmente no poseen un performance destacable, realizan buenas predicciones en conjunto. Aun mas, los modelos generados mediante esta codificación presentaron un comportamiento generalmente estable frente a nuevos ejemplos.

A pesar de las diversas pruebas realizadas sobre el modelo ensamblado propuesto, aún queda un conjunto de interrogantes respecto a su capacidad frente a nuevos ejemplos de anticuerpos y auto antígenos, y a los limites establecidos por las secuencias e interacciones utilizadas en el entrenamiento de estos modelos. Es por esto que, seria posible tomar datos almacenados en diversas bases de datos, y utilizar esta información para evaluar el modelo ensamblado. Sin embargo, esto requiere la identificación de secuencias auto antígenas dentro del conjunto de antígenos disponibles de dominio público.

Capítulo 3

Bases de datos

Diferentes estudios han sido realizados para comprender y analizar las características de la respuesta inmune, los agentes que participan en ella, y las cualidades principales que definen estas interacciones, tales como: variación de especies, propiedades de las secuencias de proteínas (Harris et al., 2018), información de las regiones hiper variantes, parátopes y epítopes (Haque et al., 2018), nivel de intensidad de esta interacción, la exposición de péptidos antígenos por el complejo mayor de histocompatibilidad (Petersdorf, 2017), entre otras, generando un enorme volumen de datos con un potencial valioso para identificar patrones y generalizar comportamientos aplicando estrategias de Data Mining, lo cual, pese a su enorme relevancia para estudios de drug discovery, aún presenta amplias aristas de estudio.

Por otra parte, diversos estudios se han desarrollado para analizar cambios genéticos, genómicos o meta-genómicos involucrados en el desarrollo de enfermedades autoinmunes, como es el estudio realizado sobre la artritis reumatoíde (Firestein, 2018), o en la mediación del reconocimiento de patógenos por un organismos (Yao et al., 2018; Dhiman et al., 2008; Stepanov and Trifonova, 2013). Todos estos datos son relevantes para comprender de mejor forma los mecanismos moleculares que median la selección o interacción de un anticuerpo y un antígeno, permitiendo el desarrollo de terapias inmunológicas, diseño de nuevas vacunas, creación e implementación de ensayos de diagnostico clínico, entre otros (Rappuoli, 2001; Yong et al., 2019; Niuniu and Yuxun, 2010; Buder-Bakhaya and Hassel, 2018; Feng et al., 2018; Faraji et al., 2018).

Previamente, se han desarrollado distintas bases de datos con la finalidad de agrupar y manipular los datos referentes a las moléculas relacionadas al sistema inmune. Principalmente, las bases de datos encontradas se centran en registrar información referente a anticuerpos, antígenos, la interacción entre estas moléculas o las regiones epítopes presentes en antígenos. Los datos contenidos pueden provenir de diversas fuentes, y corresponden a resultados experimentales observados, secuencias de nucleótidos o proteínas, o estructuras reportadas de estas moléculas. Por otra parte, se han generado bases de datos enfocadas en moléculas específicas de interés, como pueden ser datos relacionados a una enfermedad o la clasificación dada a estas moléculas.

Pese al enorme volumen de datos generado a la fecha, y las diversas plataformas diseñadas con la finalidad de facilitar el acceso a estos , múltiples dificultades surgen al requerir recopilar o analizar esta información. En algunos casos, actualmente los sistemas mencionados o encontrados en la literatura no se encuentran disponibles. Por otra parte, los mismos sistemas pueden presentar información de acceso restringido, o no facilitar herramientas para su descarga. Por otra parte, en el conjunto de bases de datos que permiten el acceso y descarga a las secuencias e información relacionada a estas, no existe un consenso con respecto al formato adoptado para entregar la información al usuario. Este problema se ve acrecentado debido a que la información de interés sobre una molécula en especifico, como anticuerpos por ejemplo, se encuentra dispersa en distintas bases de datos, lo cual implica que diversos formatos deben ser revisados para agrupar y analizar la información requerida . Todas estas problemáticas dificultan, limitan y retrasan el acceso y estudio de la información generada a la fecha, haciendo en muchos casos necesario contar con conocimientos en el manejo de datos mediante lenguajes de programación u otras herramientas como OpenRefine (Verborgh and De Wilde, 2013).

Las principales problemáticas relacionadas a los sistemas diseñados para el almacenamiento y disposición de información, y algunas de las bases de datos que presentan estos problemas, corresponden a los siguientes:

1. Falta de organización o formato en los datos entregados por estas bases de datos.

2. Sistemas no disponibles o fuera de servicio, lo cual imposibilita el acceso a

los datos almacenados (Eroshkin et al., 2014; Wang et al., 2006).

3. Información des actualizada o con errores, en algunas bases de datos con diferencia de más de 10 años (Kabat et al., 1992).

4. Datos no accesibles, ya sea por pertenecer a laboratorios o tener carácter privado, o debido a que las herramientas de descarga señaladas en los artículos no están presentes en los sistemas asociados (Björling and Uhlén, 2008; Xie, 2017).

5. Información básica o en bruto, lo cual obliga a los interesados en estos datos a utilizar otras herramientas para la obtención de características fisicoquímicas o estructurales de interés.

Como consecuencia a los problemas mencionados anteriormente, y con la finalidad de utilizar los datos recopilados para testear el modelo ensamblado implementado en este trabajo de memoria de título, asociado a la clasificación del nivel de interacción entre antígeno-anticuerpo (Ag-Ab), se planteó el diseño, implementación y validación de una base de datos integrada, que agrupe información de secuencia de aminoácidos de antígenos, anticuerpos, epítopes e interacciones antígeno-anticuerpo (David Medina-Ortiz, 2021). Además, se incluyó para las entradas de antígeno y anticuerpo información predicha empleando diversas herramientas bioinformáticas que facilitan la estimación de propiedades fisicoquímicas, estructurales, frecuencia de aminoácidos y grupos de aminoácidos, dominios resultantes de la búsqueda en servicios como Pfam, y predicción de términos Gene Ontology (GO).Adicionalmente, el sistema implementado en conjunto con el grupo de investigación de CeBiB, incluye un conjunto de herramientas que permiten analizar y predecir información de interés sobre estas moléculas, por lo tanto, entregando una plataforma integral de recopilación y análisis de anticuerpos, antígenos, interacción antígeno- anticuerpo y epítopes.

3.1. Bases de datos previamente reportadas

Las bases de datos, dependiendo del enfoque o fuente de datos considerada, pueden presentar información de diversos tipos, desde secuencias de nucleótidos o aminoácidos, a cristales dilucidados de estas moléculas, o la interacción entre estos. Además,en algunos casos se posee validación experimental que presenta el valor

de la intensidad de interacción, información con respecto a los aminoácidos que componen la región epítope, entre otros. Por otra parte, algunas bases de datos evalúan manualmente sus entradas, para verificar que estos datos sean correctos o agregar información de interés presente en los artículos asociados o bases de datos anexas. También, en algunos casos, se hace uso de herramientas para predecir elementos como epítopes en un antígeno o enlaces en una estructura.

Se recopiló información correspondiente a diversas bases de datos relacionadas a antígenos, anticuerpos, epítopes e interacciones antígeno - anticuerpo. Se consideraron bases de datos comúnmente usadas, como aquellas publicadas en revistas oficiales, tratando de recopilar el mayor número de información posible de las moléculas mencionadas anteriormente. Las bases de datos fueron divididas en 4 grupos, correspondientes a las tres moléculas de interés y la interacción antígeno - anticuerpo (Ag-Ab). Además, se realizaron sub categorizaciones para evaluar bases de datos de anticuerpos y epítopes, dependiendo del tipo de información contenida. Estas divisiones se presentan en el esquema representado en la imagen 3.1.1, en las cuales se nombran las bases de datos clasificadas dentro de cada grupo.

En las siguientes secciones se presenta un resumen de las bases de datos revisadas, los datos contenidos en estas y las fuentes desde las cuales se recopilaron estos datos. Igualmente, se presenta un resumen de las herramientas observadas en cada base de datos. Finalmente,en cada sección se presenta una tabla resumen con estos datos.

3.1.1. Bases de datos de interacción Ab-Ag.

La interacción antígeno - anticuerpo (Ag-Ab) es uno de los principales ámbitos de estudio al momento de diseñar ensayos clínicos, terapias inmunogénicas, nuevas vacunas, entre otras aplicaciones (Spiess et al., 2015; Jain et al., 2007; Tuttle iV et al., 2006; Crowe Jr, 2017; Xu et al., 2018). De esta interacción puede extraerse información como las características fisicoquímicas de las moléculas participantes, la intensidad de la interacción, regiones epítopes y parátopes que interaccionan, ubicación celular que explica su reconocimiento, entre otros. De igual forma, se generan y recopilan cristales de estos complejos Ag-Ab, con el fin de observar de forma tridimensional los aminoácidos que participan, así como el ángulo y la distancia de las interacciones generadas. Por otra parte, diversas metodologías

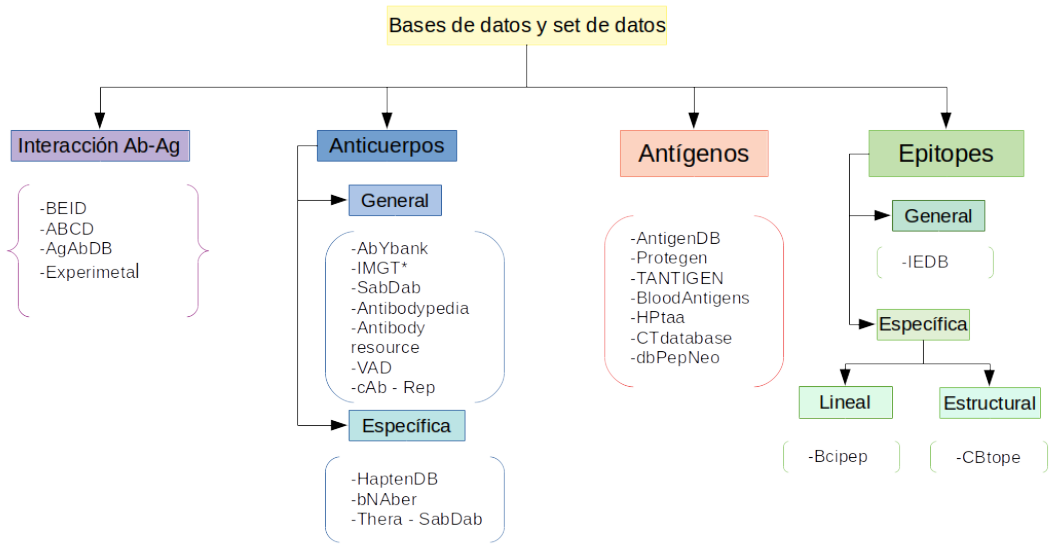


Figura 3.1.1: Esquema de bases de datos. En el esquema se presenta la división realizada para el análisis de la información, en bases de datos de interacción antígeno - anticuerpo, generales y específicas de anticuerpos, bases de datos de antígenos, y bases de datos generales y específicas de epítopes. En el caso de estas últimas, se distinguen aquellas bases de datos enfocadas en epítopes de tipo lineal y aquellos de tipo estructural.

y herramientas han sido desarrolladas para la simulación de la interacción entre estructuras de anticuerpos y antígenos de interés (Yamashita, 2018; Cloutier et al., 2019; Jabbar et al., 2018). De esta forma, y dada la relevancia que poseen este tipo de datos, diversas bases de datos han surgido para el almacenamiento de esta información.

Las bases de datos que contiene este tipo de información se enfrentan con frecuencia al mismo problema. En gran parte de los casos, los estudios realizados hacen uso de anticuerpos comerciales, por lo cual, la información disponible sobre estos es limitada. Esto impide que investigadores puedan trabajar sobre la secuencia de estos anticuerpos. Por otra parte, en aquellas bases de datos que presentan un valor de interacción, no se presenta una escala por la cual guiarse, por lo cual, no es posible generalizar la información extraída desde estos sistemas. Si bien se expone un valor obtenido experimentalmente, es necesario conocer la unidad de medida en la cual se presenta el valor de intensidad, dado que la unidad de medida es la que entrega la referencia para el establecimiento de una escala en la cual evaluar si este valor corresponde a una intensidad alta o baja.

En la Tabla 3.1.1 se presenta un resumen con información general respecto a las bases de datos revisadas en esta sección, las cuales corresponden a BEID (Tong et al., 2008), ABCD (Lima et al., 2020), AgAbDb (Kulkarni-Kale et al., 2014) y los datos experimentales utilizados como input en el entrenamiento de modelos (Frick, 2009). En el caso de BEID, el sistema web no se encuentra accesible, por lo cual, la información y herramientas presentadas no pudieron ser corroboradas. Por otra parte, los datos experimentales utilizados fueron procesados de acuerdo a la metodología planteada por Torres Almonacid (2020), y se categorizaron las interacciones de acuerdo a intensidad baja, media o alta. Dado que estos datos fueron proporcionados por el co-tutor de este proyecto, no existen problemas de autoría ante su utilización.

Cuadro 3.1.1: Tabla resumen bases de datos de interacción antígeno - anticuerpo.

Base de datos	**Información contenida**	**Origen de la información**	**Herramientas**
BEID (Tong et al., 2008)	Secuencias y estructuras de interacción Ig- Ag.	Cristales desde PDB revisados manualmente.	Búsqueda bajo diversos criterios (nombre Ig, nombre antígeno, cristal asociado, entre otros).
ABCD (Lima et al., 2020)	Información general, secuencias de Ig y Ag químicamente caracterizados.	Diversas bases de datos (Pubmed, IMGT, RAN, entre otras).	Buscador simple mediante palabras claves.
AgAbDb (Kulkarni-Kale et al., 2014)	Estructuras de anticuerpos en interacción con antígenos.	Cristales recopilados desde PDB.	Búsqueda simple y avanzada por palabras claves o regiones de interés.

Cuadro 3.1.1: Tabla resumen bases de datos de interacción antígeno - anticuerpo.

Base de datos	Información contenida	Origen de la información	Herramientas
Experimental (Frick, 2009)	Secuencias de anticuerpos y antígenos.	Experimento realizado utilizando ProtoArray.	No posee portal de acceso a datos.

3.1.2. Bases de datos de anticuerpos.

Estas bases de datos se enfocan en recopilar y presentar información respecto a anticuerpos.Se realizó una clasificación de estos sistemas, donde se dividieron los repositorios analizados en bases de datos generales y bases de datos específicas. Las primeras corresponden a bases de datos en las cuales la recopilación de datos se hace indistintamente de la especie de origen, funciones observadas o antígenos reconocidos. En la segunda clase se encuentran las bases de datos que aplican filtros en la recopilación de datos, como es la unión a un antígeno específico, relación con cierta enfermedad, o actividad de interés asociada. En adición a las problemáticas mencionadas anteriormente, con respecto a dispersión de los datos y falta de formato en su recopilación desde diversas fuentes, en algunos sistemas no se especifica la especie a la cual corresponde el anticuerpo, si es un fragmento o la molécula completa, o si la secuencia corresponde a la cadena pesada o ligera.

Se presenta una Tabla resumen 3.1.2 con respecto a la información contenida en las bases de datos comprendidas en esta sección, así como las herramientas que poseen y la clase a la cual pertenecen de acuerdo a la clasificación mencionada anteriormente. En ciertos sistemas, se presentan mas de una base de datos, las cuales concentrar información desde diversas fuentes, o se enfocan en información de diversos tipos. Un ejemplo de esto es abYbank, un sistema que congrega las bases de datos AbPDBseq, Kabat(Kabat et al., 1992) , EMB-ENA (Leinonen et al., 2010), AbDB(Ferdous and Martin, 2018) y SACS (Allcorn and Martin, 2002). De igual forma, el sistema IMGT incluye una serie de bases de datos enfocadas en moléculas específicas del sistema inmune, las cuales incluyen inmunoglobulinas, anticuerpos monoclonales, receptores de células T (TcR) y moléculas del complejo mayor de histocompatibilidad (MHC). Las bases de datos incluidas dentro de

IMGT corresponden a IMGT/LIGM-DB (Giudicelli et al., 2006), IPD-IMGT/HLA (Robinson et al., 2020), IMGT/PRIMER-DB(Giudicelli et al., 2005), IMGT/CLL-DB, IMGT/GENE-DB (Giudicelli et al., 2005), IMGT/mAb-DB (Poiron et al., 2010) y IMGT/3Dstructure-DB (Kaas et al., 2004).

Cuadro 3.1.2: Tabla resumen bases de datos de anticuerpos.

Base de datos	**Información contenida**	**Origen de la información**	**Herramientas**	**Tipo**
SACS(Allcorn and Martin, 2002)	Estructuras	Entradas recopiladas desde PDB mediante palabras claves.	Herramientas presentes en abYbank. Modelamiento, BLAST, anotación de secuencias y alineamiento	General
Kabat(Johnson and Wu, 2001)	Secuencias de aminoácidos.	Secuencias publicadas en el libro "Sequences of protein of immunological interest."	Herramientas presentes en abYbank. Modelamiento, BLAST, anotación de secuencias y alineamiento.	General

Cuadro 3.1.2: Tabla resumen bases de datos de anticuerpos.

Base de datos	Información contenida	Origen de la información	Herramientas	Tipo
EMBL-ENA(Leinonen et al., 2010)	Secuencias de nucleótidos.	Aportes de usuarios, colaboraciones y diversas bases de datos (Uniprot, RNAcentral, Ensembl, entre otros).	Herramientas presentes en abYbank. Modelamiento, BLAST, anotación de secuencias y alineamiento.	General
AbDb(Ferdous and Martin, 2018)	Estructuras.	Información extraída desde SACS	Herramientas presentes en abYbank. Modelamiento, BLAST, anotación de secuencias y alineamiento.	General

Cuadro 3.1.2: Tabla resumen bases de datos de anticuerpos.

Base de datos	Información contenida	Origen de la información	Herramientas	Tipo
AbPDBSeq (Johnson and Wu, 2001)	Secuencias de aminoácidos.	Secuencias presentadas por los cristales recopilados desde AbDb.	Herramientas presentes en abYbank. Modelamiento, BLAST, anotación de secuencias y alineamiento.	General
IMGT LIGM-DB (Giudicelli et al., 2006)	Secuencias de nucleótidos de TcR y Ig.	Información obtenida desde diversas bases de datos generales.	Búsqueda simple y avanzada bajo diversos criterios.	General
IPD-IMGt /HLA (Robinson et al., 2020)	Secuencias de nucleótidos de MHC.	Información entregada por laboratorios asociados desde 46 países y usuarios del sistema.	Búsqueda bajo distintos términos, BLAST y alineamiento de secuencias.	General

Cuadro 3.1.2: Tabla resumen bases de datos de anticuerpos.

Base de datos	Información contenida	Origen de la información	Herramientas	Tipo
IMGT PRIMER-DB(Giudicelli et al., 2005)	Primers de secuencias de Ig y TcR.	Artículos científicos revisados.	Búsqueda usando palabras claves o por similitud de secuencias.	General
IMGT CLL-DB(Giudicelli et al., 2005)	Información de pacientes con leucemia linfocítica crónica.	Instituciones asociadas en convenio.	Sistema no accesible. No hay mención de herramientas en artículos asociados.	General
IMGT GENE-DB(Giudicelli et al., 2005)	Secuencias génicas de Ig y TcR.	Bases de datos generales.	Búsqueda por diversos criterios.	General
IMGT mAb-DB(Poiron et al., 2010)	Secuencias de anticuerpos monoclonales con uso clínico.	Bases de datos generales.	Búsqueda por diversos criterios.	General

Cuadro 3.1.2: Tabla resumen bases de datos de anticuerpos.

Base de datos	Información contenida	Origen de la información	Herramientas	Tipo
IMGT 3D/2D-DB(Kaas et al., 2004)	Estructuras y secuencias de Ig, TcR y MHC.	Información desde PDB, INN y Kabat.	Búsqueda usando diversos criterios o parámetros.	General
SabDab (Dunbar et al., 2014)	Estructuras.	Información obtenida desde PDB.	Búsqueda y filtrado de resultados usando diversos criterios.	General
Antibodypedia (Björling and Uhlén, 2008)	Información de anticuerpos validados.	Laboratorios y aportes de usuarios con evidencia experimental.	Búsqueda simple utilizando palabras claves.	General
Antibody Resource	Información de anticuerpos diseñados por laboratorio.	No se menciona	Búsqueda simple mediante palabras claves.	General

Cuadro 3.1.2: Tabla resumen bases de datos de anticuerpos.

Base de datos	Información contenida	Origen de la información	Herramientas	Tipo
VAD (Xie, 2017)	Información de anticuerpos comerciales y públicos.	Información publicada en artículos.	Búsqueda simple usando palabras.	General
cAb-Rep (Sheng et al., 2019)	Secuencias de Ig desde células B.	Información proveniente de 121 pacientes voluntarios.	Búsqueda de motivos, BLAST, estimación de frecuencia de mutación y evaluación de frecuencia de glicosilación.	General
HaptenDB (Singh et al., 2006)	Secuencias de anticuerpos anti-haptenos.	Bases de datos generales y artículos publicados.	Búsqueda usando palabras claves o por similitud de estructura.	Específica

Cuadro 3.1.2: Tabla resumen bases de datos de anticuerpos.

Base de datos	Información contenida	Origen de la información	Herramientas	Tipo
bNAber (Eroshkin et al., 2014)	Secuencias de anticuerpos ampliamente neutralizantes de HIV	Información publicada en artículos.	Sistema no accesible. No hay mención de herramientas en artículos asociados.	Específica
Thera-SabDab (Raybould et al., 2020)	Estructuras de anticuerpos con usos terapéuticos.	Información presentada y reconocida por OMS y presente en SabDab.	Búsqueda usando palabras claves o similitud de secuencia.	Específica

3.1.3. Bases de datos de antígenos.

Así como se realizó una búsqueda de secuencias de anticuerpos, se hizo una revisión de bases de datos de antígenos disponibles. La búsqueda se vio limitada a bases de datos y sistemas que especificaran la recolección de datos correspondientes a antígenos. La información de antígenos es de interés al momento del estudio de la interacción antígeno- anticuerpo, o la identificación de las regiones que participan en esta interacción (llamados epitopes). De igual forma, las características fisicoquímicas de estas moléculas, su ubicación o funcion asociada, pueden dar información con respecto a patrones de interacción con anticuerpos. De esta forma,

el conocimiento del antígeno que presenta un cierto patógeno, permite el diseño de anticuerpos con una actividad específica, los cuales podrían tener uso terapéutico o ser utilizados en ensayos de laboratorio para la detección de enfermedades (Sifniotis et al., 2019; Balmaseda et al., 2017; Cao et al., 2019).

Las bases de datos recopiladas presentan información asociada tanto a aminoácidos y proteínas, como a genes o secuencias de nucleótidos. Por otra parte,a diferencia de los repositorios de anticuerpos, en este caso no se realizo una clasificación de antígenos en base a su especie de origen o enfermedades asociadas. En algunos casos, las bases de datos presentan secuencias o regiones epítopes asociadas un antígeno en específico. De igual forma, en parte de estos repertorios se añade información con respecto a los genes asociados a estos antígenos, modificaciones como glicosidaciones, entre otros. Las bases de datos incluidas en este análisis corresponden a AntigenDB (Ansari et al., 2010), Protegen (Yang et al., 2011), TANTIGEN (Olsen et al., 2017), BloodAntigens(Lane et al., 2018), HPtaa (Wang et al., 2006), CTdatabase (Almeida et al., 2009) y dbPepNeo (Tan et al., 2020).

Cuadro 3.1.3: Tabla resumen bases de datos de antígenos.

Base de datos	**Información contenida**	**Origen de la información**	**Herramientas**
AntigenDB (Ansari et al., 2010)	Información asociada tanto a nucleótidos como aminoácidos y estructuras cristalizadas.	Bases de datos como BciPep, IEDB, GenBank y PDB. Tambien integra información de artículos revisados.	Búsqueda mediante palabras claves, búsqueda de epítopes, mapeo de péptidos y BLAST.
Protegen (Yang et al., 2011)	Información asociada tanto a nucleótidos como aminoácidos.	Información extraída desde artículos científicos. Se incluye información desde NCBI.	Búsqueda simple o mediante diversos criterios y BLAST.

Cuadro 3.1.3: Tabla resumen bases de datos de antígenos.

Base de datos	Información contenida	Origen de la información	Herramientas
TANTIGEN (Olsen et al., 2017)	Secuencias de aminoácidos de antígenos e información asociada de genes.	Información extraída desde bases de datos generales y artículos publicados.	Búsqueda mediante palabras claves, BLAST y visualización de epítopes.
BloodAntigens (Lane et al., 2018)	Información de antígenos presentes en células sanguíneas rojas y plaquetas.	Datos de secuenciación completa de genoma proporcionados por integrantes del proyecto MedSeq.	El sistema no presenta herramientas integradas.
HPtaa (Wang et al., 2006)	Información de antígenos asociados a tumores.	Set de datos obtenidos desde análisis de microarray en pacientes con diversos tipos de cáncer.	Búsqueda rápida o utilizando distintos criterios.
CTdatabase (Almeida et al., 2009)	Información de antígenos asociados a cáncer de testículos.	Información obtenida desde artículos publicados, relacionados con datos entregados por bases de datos generales.	Búsqueda simple por palabras claves.

Cuadro 3.1.3: Tabla resumen bases de datos de antígenos.

Base de datos	Información contenida	Origen de la información	Herramientas
dbPepNeo (Tan et al., 2020)	Información de péptidos neo antígenos asociados a unión con a HLA-I.	Artículos publicados en Pubmed y bases de datos IEDB y Cancer Immunity Peptide database.	Búsqueda por palabras claves o secuencias. ProGeo-neo e INeo-Epp para la predicción de neo antígenos.

3.1.4. Bases de datos de epítopes.

La región o porción de aminoácidos que son reconocidos por anticuerpos y producen la interacción se conoce como región epítope, y es de gran interés debido a que corresponde a la región mas relevante en la interacción, cuyo reconocimiento puede guiar el diseño de vacunas (Patronov and Doytchinova, 2013; Parvizpour et al., 2020), empleo en ensayos y diagnostico (Sadam et al., 2021; Heiss et al., 2020), o explicar la diferencia en la intensidad de la interacción entre diversos antígenos (Qi et al., 2021). Los epítopes se dividen en dos tipos: epítopes lineales y epitopes estructurales o discontinuos. Los epitopes lineales corresponden a fragmentos continuos de aminoácidos en la secuencia primaria, mientras que los epítopes estructurales corresponden a un conjunto de aminoácidos que se encuentran en diversas posiciones en la secuencia lineal, pero que son cercanos en su forma tridimensional. De esta forma, el reconocimiento de epítopes estructurales requiere de conocimiento con respecto a la estructura tridimensional del antígeno.

Debido a esto diversos repositorios de epítopes han surgido durante los años. Estos pueden ser generales o enfocarse en un tipo de epítope, y poseer distintos objetivos o fuentes de información. Es por esto que se hizo una clasificación de las bases de datos, declarando una clase general y otra específica. Dentro de la clase especifica se declaran dos categorías, correspondientes a específicas de epítopes lineales y específica de epítopes estructurales. Las bases de datos presentes en la clase general corresponden a aquellas que almacenan y entregan ambos tipos de epítopes. Por

el contrario, aquellas contenidas en clases específicas es porque se enfocan en el tipo de epítope respectivo a su subclase. Las bases de datos de epitopes revisadas, presentadas en la tabla, corresponden a IEDB(Vita et al., 2015), HLA epitope register (Duquesnoy et al., 2019), BciPep (Saha et al., 2005), CBtope (Ansari and Raghava, 2010), CED (Huang and Honda, 2006), SEDB (Sharma et al., 2012) y Epitome (Schlessinger et al., 2006).

Cuadro 3.1.4: Tabla resumen bases de datos de epítopes.

Base de datos	**Información contenida**	**Origen de la información**	**Herramientas**	**Tipo**
IEDB (Vita et al., 2015)	Información general y secuencias de epítopes lineales y estructurales.	Artículos científicos revisados y aportes de usuarios.	Diversas herramientas de predicción y análisis de epítopes (TepiTool, BepiPred2.0, DiscoTope2.0, entre otros).	General
HLA epitope register (Duquesnoy et al., 2019)	Información general de genes y estructuras asociadas a antígenos HLA.	Diversas bases de datos.	No posee herramientas integradas al sistema.	General.
BciPep (Saha et al., 2005)	Secuencias de epítopes lineales provenientes de antígenos con interacción con células B.	Información recopilada desde artículos científicos.	Búsqueda simple y avanzada, mapeo de epítopes y BLAST.	Específica lineal.

Cuadro 3.1.4: Tabla resumen bases de datos de epítopes.

Base de datos	Información contenida	Origen de la información	Herramientas	Tipo
CBtope (Ansari and Raghava, 2010)	Información de epítopes estructurales utilizandos en el entrenamiento de la herramienta.	Base de datos IEDB y set de datos.	Predicción de epítopes estructurales.	Específica estructural.
CED (Huang and Honda, 2006)	Información relacionada a epítopes estructurales.	Artículos científicos publicados en revistas revisadas por pares.	Búsqueda mediante diversos filtros y visualización de epítopes.	Específica estructural.
SEDB (Sharma et al., 2012)	Información relacionada a epítopes estructurales.	Artículos científicos y bases de datos PDB, IMGT-3D, BciPep y IEDB.	Visualización de las estructuras y la calidad de estas mediante gráficos de Ramachandran.	Específica estructural.
Epitome (Schlessinger et al., 2006)	Información relacionada a epítopes estructurales.	Estructuras PDB recopiladas en BLAST utilizando una secuencia anticuerpo consenso.	Búsqueda mediante diversos criterios.	Específica estructural.

3.2. Metodología.

A partir de las bases de datos mencionadas anteriormente se aplicó el proceso ETL (Extract, transform and load), el cual corresponde a la descarga o extracción, transformación y cambio de formato, y carga de las entradas recopiladas correspondientes a secuencias de anticuerpo, antígenos y epitopes. Para cada una de las bases de datos se desarrolló una metodología de procesamiento, dependiendo del tipo de . Todos los procedimientos se realizaron utilizando el lenguaje de programación Python v3.6 (Foundation, 2008), de forma que el flujo de trabajo fuese coherente y fácil de seguir.

3.2.1. Descarga de set de datos.

En las bases de datos mencionadas anteriormente se buscó una pestaña u opción directa para la descarga de los datos contenidos. De no presentarse esta opción, si los sistemas poseen herramientas de búsqueda, se utilizó este medio para la obtención de los datos de interés. En el caso de anticuerpos, de presentarse el filtro de especie en la búsqueda de entradas, se aplicó filtro por especie *Homo Sapiens*. Para secuencias de antígeno o epítopes no se aplicaron filtros, debido a que se desea no sólo recopilar información, sino que también probar estos datos contra el modelo ensamblado desarrollado. Finalmente, si las bases de datos no presentaron la información directamente en sus sitios web, pero poseían acceso a sitios FTP de almacenamiento, se utilizó este recurso para la recopilación de los datos.

3.2.2. Procesamiento de set de datos.

Para cada set de datos descargado se implementó un script en Python v3.6 que permita procesar la información, de forma que se obtengan las secuencias de interés. En los casos en que la información se presentó dentro de un archivo tabular, estos fueron procesados utilizando la librería Pandas (McKinney et al., 2011), para extraer las secuencias y los identificadores proporcionados por estas bases de datos. Por otra parte, las bases de datos que presentaron la información en referencia a otras bases de datos (como Uniprot, NCBI o PDB) se hizo uso de wget, Python Entrez y BioPython para descargar cada secuencia individualmente y construir un multifasta de las secuencias presentadas por el repositorio analizado. En las situaciones en las cuales la información entregada por las bases de datos

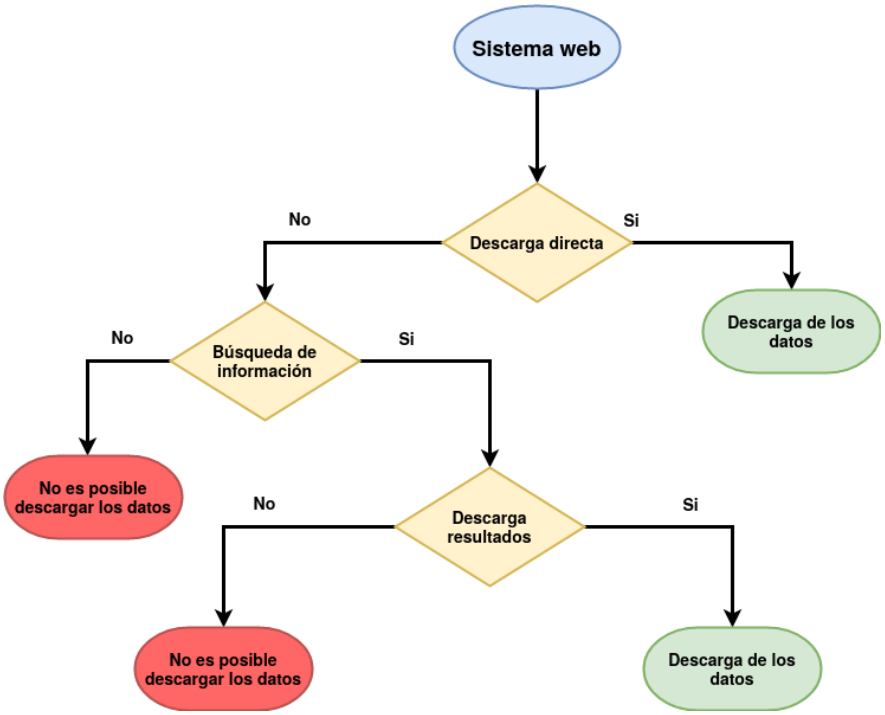


Figura 3.2.1: Esquema descarga de datos. En la figura se presenta un esquema respecto a los pasos seguidos en la recopilación de datos. Dependiendo de la accesibilidad a los datos presentada por cada sistema, se utilizaron diversas estrategias de descarga.

correspondían a archivos PDB, se uso de la librería PDB de BioPython para el procesamiento del cristal y obtención de las secuencias de interés. En aquellas bases de datos que proporcionaron un archivo especificando las cadenas presentes en cada cristal, se utilizó este archivo como referencia para filtrar las cadenas de interés. Por otra parte, si no se proporcionó tal archivo, se observó el detalle del archivo PDB para buscar palabras claves en las cadenas como "anticuerpo" o "inmunoglobulina". De igual forma, se evaluaron las descripciones de las cadenas para clasificarlas como pesada o ligera. Las palabras claves encontradas fueron incluidas en el identificador de la secuencia correspondiente. De esta forma, de cada base de datos se generó un archivo multifasta utilizado en los siguientes pasos.

3.2.3. Unión y depuración set de datos.

A partir de los archivos multifasta generados para cada bases de datos fue generado un archivo csv, el cual contiene el identificador dado por la base de datos, la secuencia de aminoácidos, el largo, y columnas correspondientes a las bases de

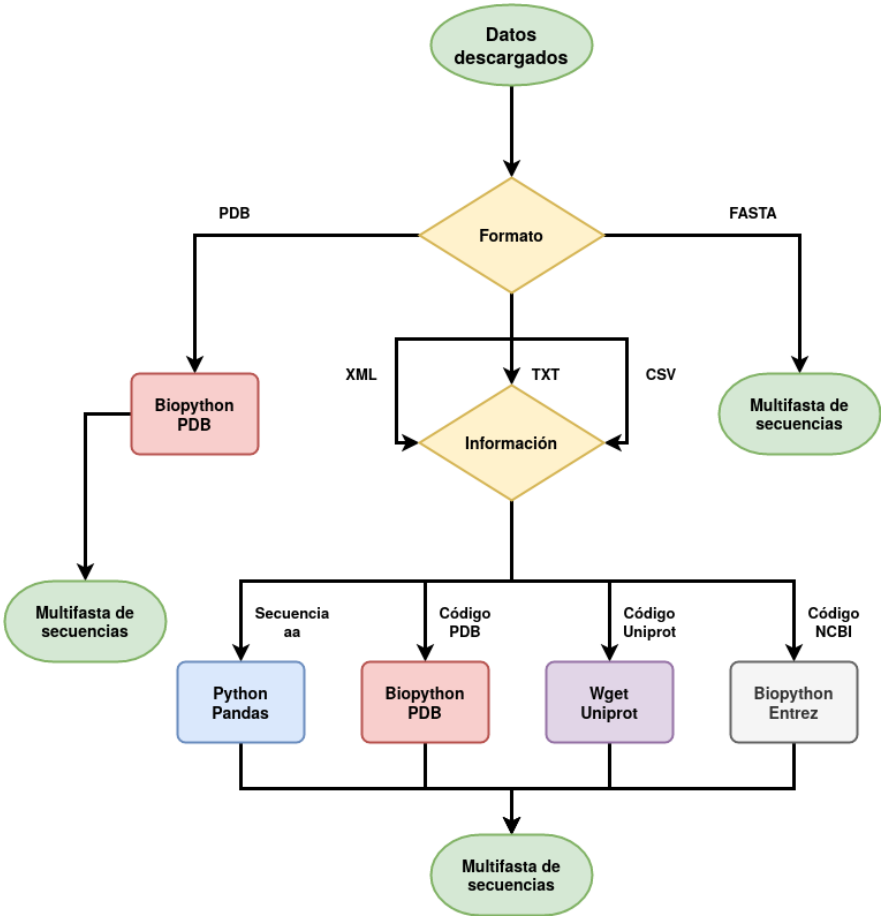


Figura 3.2.2: Esquema procesamiento de datos. Para cada una de las bases de datos descargadas y analizadas se observó el formato en el cual fueron proporcionados los datos. Basado en este formato se utilizaron diversas herramientas para procesar la información y obtener un archivo multifasta por cada base de datos.

datos de las cuales se obtenga la información para anticuerpos, antígenos o epítopes. En aquella base de datos en la cual se observó la secuencia se otorgo un valor 1, mientras que en las que no se registro un 0. En el caso de anticuerpos, si el identificador indica si la secuencia corresponde a una cadena pesada, se marcó con valor 1 esta columna. Al contrario, si es cadena liviana, se señaló un 1 en la columna de liviana. Por otra parte, si el identificador no indica si la secuencia corresponde a cadena pesada o liviana, ambas cadenas se establecieron en 0. Como filtro adicional, en el caso de los anticuerpos, se consultó de igual forma, en todos los set de datos recopilados desde las bases de datos,si el identificador poseía palabras claves de *Homo Sapiens* o *Human*, para filtrar aquellas bases de datos que no fueron filtradas al momento de la descarga o desarrollo del multifasta. De

esta forma, se generó por cada base de datos dos conjuntos de datos, uno que posee las secuencias cuyos identificadores poseen una palabra clave relacionada a humanos, y otro para todas las secuencias descargadas. Dado que se conocen las bases de datos fueron filtradas al momento de descarga o procesamiento de los datos, se conoce el set de datos que posee las secuencias a utilizar para el caso de anticuerpos.

Una vez se seleccionaron los set de datos a utilizar, estos archivos fueron concatenados en un solo archivo, el cual corresponde al conjunto general de secuencias. Por otra parte, se implementó un script el cual lee las secuencias de los archivos individuales seleccionados para cada base de datos, y se determinaron las secuencias únicas presentes en cada una de estas bases de datos, y del conjunto general. El grupo de secuencias únicas presentes en la totalidad de los datos procesados fue utilizado para filtrar el archivo general. Durante este proceso se evaluaron las secuencias presentadas por mas de una base de datos. De igual forma, se realizó el conteo de secuencias descargadas y secuencias únicas por base de datos.

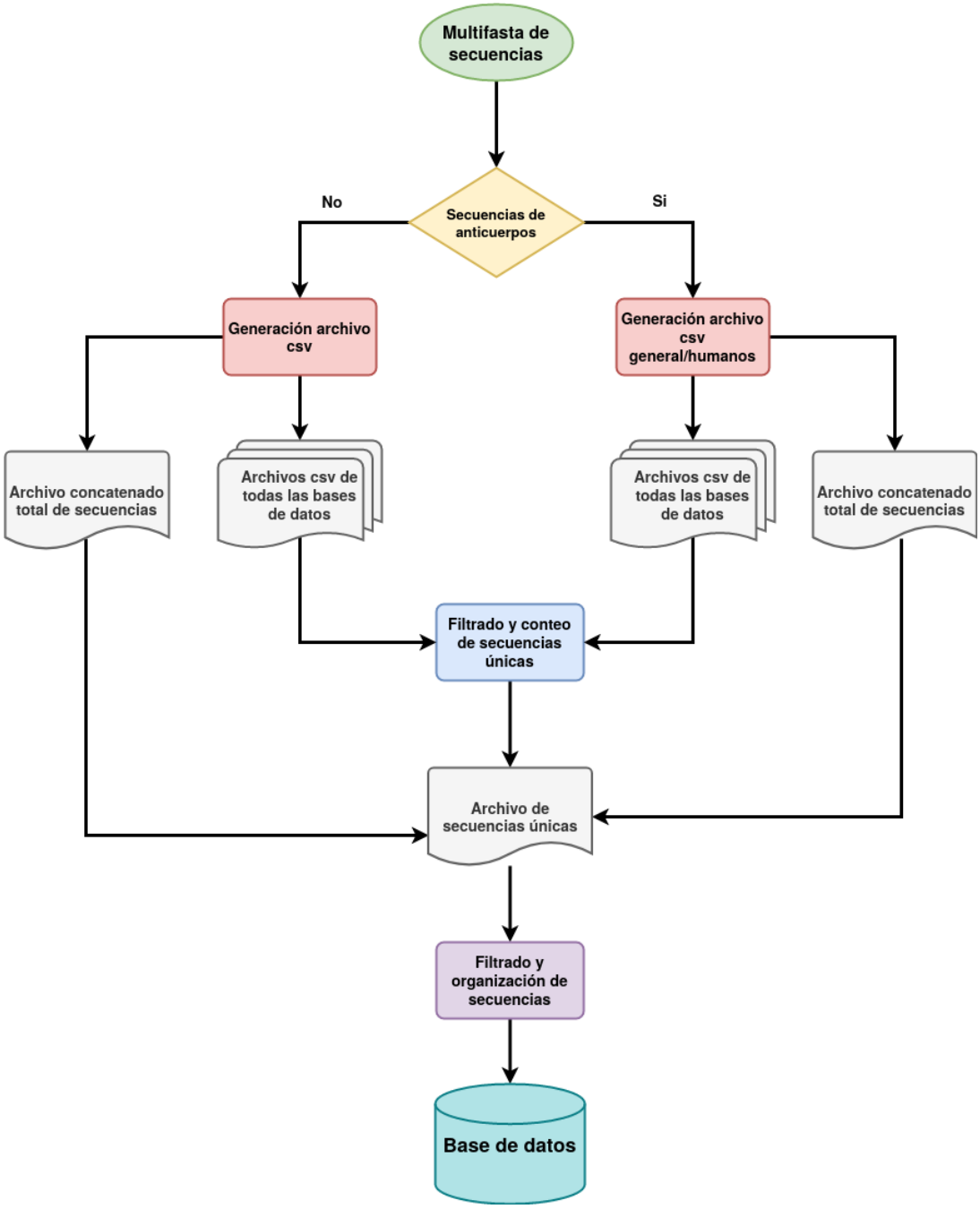
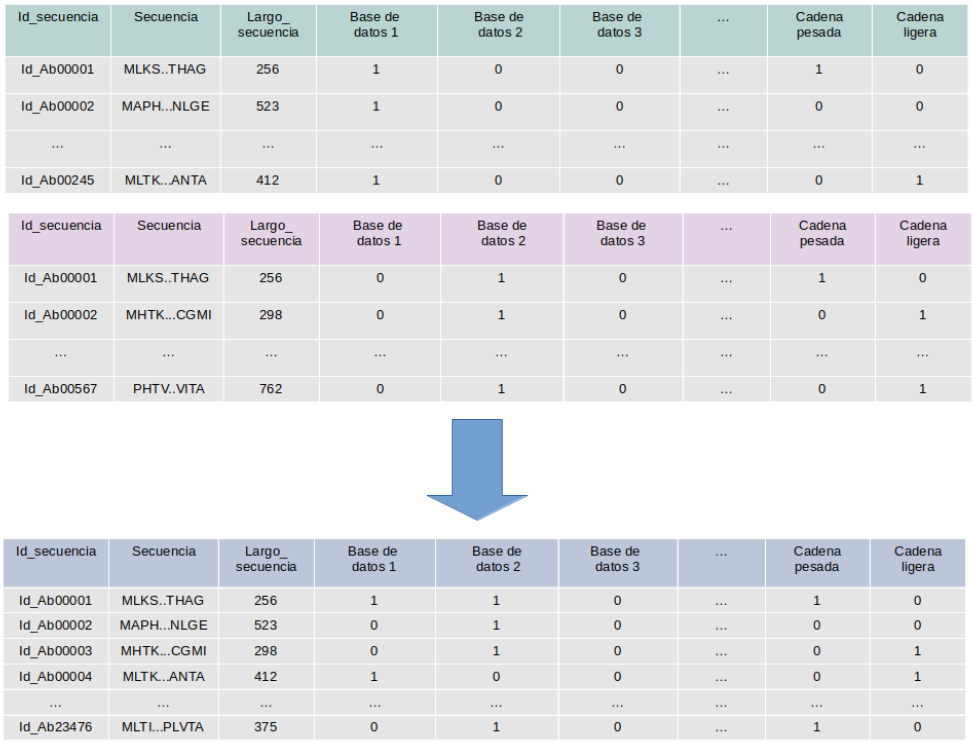


Figura 3.2.3: Esquema unión y depuración de datos. En la figura se presenta en flujo desarrollado para la unión y depuración de los datos provenientes desde los sistemas considerados. Si los datos correspondían a anticuerpos, se realizó un filtro por palabras claves referentes a humano. Una vez unificado el set, se realizo un filtrado de entradas redundantes y se proceso nuevamente el archivo general de secuencias, de modo de revisar la presencia de cada secuencia en los registros de cada base de datos y eliminar la redundancia de los set de datos finales.



Id_secuencia	Secuencia	Largo_ secuencia	Base de datos 1	Base de datos 2	Base de datos 3	...	Cadena pesada	Cadena ligera
Id_Ab00001	MLKS..THAG	256	1	0	0	...	1	0
Id_Ab00002	MAPH...NLGE	523	1	0	0	...	0	0
...	...	...	...	...	...	...	...	...
Id_Ab00245	MLTK...ANTA	412	1	0	0	...	0	1

Id_secuencia	Secuencia	Largo_ secuencia	Base de datos 1	Base de datos 2	Base de datos 3	...	Cadena pesada	Cadena ligera
Id_Ab00001	MLKS..THAG	256	0	1	0	...	1	0
Id_Ab00002	MHTK...CGMI	298	0	1	0	...	0	1
...	...	...	...	...	...	...	...	...
Id_Ab00567	PHTV..VITA	762	0	1	0	...	0	1

Id_secuencia	Secuencia	Largo_ secuencia	Base de datos 1	Base de datos 2	Base de datos 3	...	Cadena pesada	Cadena ligera
Id_Ab00001	MLKS..THAG	256	1	1	0	...	1	0
Id_Ab00002	MAPH...NLGE	523	0	1	0	...	0	0
Id_Ab00003	MHTK...CGMI	298	0	1	0	...	0	1
Id_Ab00004	MLTK...ANTA	412	1	0	0	...	0	1
...	...	...	...	...	...	...	...	...
Id_Ab23476	MLTI...PLVTA	375	0	1	0	...	1	0

Figura 3.2.4: Ejemplo set de datos. La figura corresponde a un ejemplo de archivo tabular generado para 2 bases de datos (separadas por colores), en las cuales se presentan entradas provenientes desde estas bases de datos. Al momento de unificar estas bases de datos y eliminar la redundancia se actualizaron los índices correspondientes a las bases de datos en las cuales se observo cada secuencia.

3.2.4. Diseño e implementación base de datos.

Se diseñó un modelo de datos basado en estructuras no relacionales, conocidos como NoSQL, con el fin de optimizar el costo computacional a la hora de generar las consultas y facilitar la administración de información al considerar estructuras de datos en forma JSON como flujo de información para un posterior desarrollo de sistema computacional basado en NodeJS.

Como gestor de base de datos se seleccionó MongoDB, mientras que se desarrollaron las colecciones con respecto a cada tipo de molécula inmunológica, contemplando para esto, las secuencias de antígenos, anticuerpo y las de epítopes. Además, se implementaron colecciones para adicionar la información de las interacciones entre antígeno y anticuerpo y colecciones para registrar la presencia de epítopes dentro de una secuencia de antígenos.

Finalmente, pese a que no fue desarrollado como objetivo central de este proyecto de memoria de título, se participó en un proyecto de integración de la base de datos generada con diversas herramientas computacionales para el estudio y análisis de este tipo de moléculas, liderado por el co-tutor de este trabajo de título en colaboración con los centros de investigación CeBiB y Universidad de Magallanes. Para ello, se diseñó un sistema web amigable con el usuario, en el cual se habilitaron diversas herramientas para la búsqueda y análisis de los datos presentes, así como de nuevas secuencias. El sistema web fue diseñado con un modelo vista controlador (MVC) El componente de vista y los controladores se implementaron utilizando el lenguaje de programación JavaScript a través del framework Express. Los componentes de la visualización se optimizaron empleando el framework Bootstrap 4, mientras que todas las herramientas implementadas fueron desarrolladas utilizando el lenguaje de programación Python v3.6.

3.2.5. Caracterización de secuencias.

Los set de datos generados en el paso anterior fueron caracterizados de acuerdo a las siguientes propiedades y criterios:

Cuadro 3.2.1: Tabla resumen propiedades caracterizadas y predichas.

Característica	Descripción	Herramienta
Frecuencia de aminoácidos	Conteo de cada aminoácido en la secuencia	Pandas (McKinney et al., 2011)
Frecuencia de grupos de aminoácidos	Conteo de residuos pertenecientes a los grupos no polares alifaticos, polares sin carga, aromáticos, aminoácidos con carga positiva y aminoácidos con carga negativa.	Pandas (McKinney et al., 2011)
Peso molecular.	Masa molecular de la proteína, calculada a partir de la suma de los pesos moleculares de los aminoácidos en la secuencia (g/mol)	Peptipedia (Quiroz et al., 2021)

Cuadro 3.2.1: Tabla resumen propiedades caracterizadas y predichas.

Característica	Descripción	Herramienta
Carga eléctrica.	Carga neta de la proteína, calculada a partir de los valores de los aminoácidos a un pH 7.0 (C)	Peptipedia (Quiroz et al., 2021)
Densidad de carga eléctrica.	Cantidad de carga electrica en el volumen de la proteína (C*g/mol)	Peptipedia (Quiroz et al., 2021)
Punto isoeléctrico	Cálculo de pH al cual la proteína posee una carga neta 0	Peptipedia (Quiroz et al., 2021)
Predicción estructura secundaria	Predicción de estructura secundaria de 3 y 8 estados a partir de la secuencia de aminoácidos. Cálculo de porcentaje de cada estructura y estado de cada posición en la secuencia.	Predict Property (Wang et al., 2016)
Predicción accesibilidad a solvente	Predicción de topología de membrana de la proteina a partir de la secuencia de aminoácidos. Cálculo de porcentaje de cada topología y correspondencia con la secuencia de amminoácidos.	Predict Property (Wang et al., 2016)
Predicción desorden de secuencia.	Predicción del orden o desorden de los aminoácidos en la secuencia. la predicción evalua la falta de átomos en los aminoácidos de acuerdo al modelo generado por el programa	Predict Property (Wang et al., 2016)

Cuadro 3.2.1: Tabla resumen propiedades caracterizadas y predichas.

Característica	Descripción	Herramienta
GO función molecular.	Predicción de término Gene Ontology asociado a funcion molecular de acuerdo a lo predicho por el programa	metastudent (Yachdav et al., 2014)
GO proceso biológico.	Predicción de término Gene Ontology asociado a proceso biológico de acuerdo a lo predicho por el programa	metastudent (Yachdav et al., 2014)
GO componente celular.	Predicción de término Gene Ontology asociado a componente celular de acuerdo a lo predicho por el programa	metastudent (Yachdav et al., 2014)
Predicción dominio Pfam.	Predicción de dominios Pfam presentes en la secuencia de aminoácidos.	Pfam (El-Gebali et al., 2019)

Las secuencias de antígeno y anticuerpo se sometieron a scripts que permiten calcular su largo, peso molecular, punto isoeléctrico, densidad de carga y carga eléctrica, mediante módulos implementados en Peptipedia (Quiroz et al., 2021). Por otra parte, haciendo uso de códigos entregados por el co-tutor se calculó la frecuencia de residuos y grupo de aminoácidos para cada secuencia. Además, utilizando la herramienta Predict Property de Raptor X (Wang et al., 2016) se realizaron predicciones de propiedades estructurales, como la estructura secundaria, la accesibilidad a solvente o el desorden de la secuencia. Así mismo, se hizo uso del módulo metastudent de Predictprotein (Yachdav et al., 2014) para la predicción de los términos GO correspondientes a función, ubicación y componentes celulares relacionados. Finalmente, se realizó una predicción de dominios para las secuencias mediante consultas web en la base de datos Pfam.

3.3. Resultados y discusión.

A partir de la extracción, transformación y carga realizada a los datos fue posible obtener el tipo y número real de datos obtenidos desde cada base de datos. Para todos los sistemas integrados se encontraron diferencias con respecto al número de secuencias reportadas por éstas, y el número de secuencias únicas obtenidas desde estas fuentes de información. Se observó tanto el número de secuencias mencionadas dentro de los artículos de referencia o sistemas de alojamiento de las bases de datos, como la cantidad de secuencias descargadas desde estos. En el caso del set de datos de secuencias de anticuerpos recopiladas, las bases de datos desde las cuales se obtuvo información corresponden a ABCD (Lima et al., 2020), Antibodypedia (Björling and Uhlén, 2008), SACS (Allcorn and Martin, 2002), AbDb (Ferdous and Martin, 2018), SabDab (Dunbar et al., 2014), IMGT- 3D (Kaas et al., 2004), Bcipep Saha et al. (2005), abYsis (Swindells et al., 2017) y abYbank Kabat (Martin, 1996). Además, se incluye el set de datos proporcionado por Frick (2009), el cual es referido como Experimental.

De todas las fuentes de información ofrecidas por IMGT, solo se pudo hacer uso de IMGT-3D, debido a que el resto de las plataformas contenían información referente a secuencias de nucleótidos, no presentaban información accesible para su descarga o la página no arroja resultados al realizar la consulta correspondiente (mAb-DB). Por otra parte, desde los set de datos disponibles en abYbank no se utilizó AbPDBSeq, debido a que las secuencias presentadas en este set no poseen descripción con respecto a si corresponden a secuencias obtenidas desde cristales con anticuerpos humanos, y presentan datos recopilados en AbDb, los cuales si fueron analizados de la manera deseada. Similarmente, tampoco se utilizó EMBL-ENA (Leinonen et al., 2010) o cAb-Rep (Sheng et al., 2019), debido a que se presentan datos de secuencias de nucleótidos correspondientes a anticuerpos las cuales no se encuentran dentro del grupo de interés de los objetivos planteados. Las bases de datos Antibody Resource, VAD (Xie, 2017) y HaptenDB (Singh et al., 2006) tampoco fueron utilizadas debido a que no se encontraron herramientas para la descarga de los datos. Por otra parte, no se hizo uso de bNAber Eroshkin et al. (2014) debido a que al dirigirse al sistema proporcionado por el artículo la pagina web retornó un error. Finalmente, no se hizo uso de Thera-SabDab (Raybould et al., 2020) debido a que la fuente utilizada por esta base de datos es

SabDab, la cual si fue incluida.

En la figura 3.3.1 se presenta un gráfico de barras que muestra la cantidad de secuencias descargadas y secuencias únicas por base de datos utilizada. En el caso de ABCD y Antibodypedia, estas mencionan en sus artículos o sitios asociados un total de 21,542 y 4,365,328 entradas pertenecientes a anticuerpos respectivamente. Sin embargo, estas plataformas presentan muchos datos pertenecientes a laboratorios o empresas privadas, por lo cual, sus secuencias no son accesibles. De forma similar, el sistema abYsis posee, de acuerdo a su sitio web, posee un total de 100357 entradas correspondientes a secuencias de anticuerpos de humanos. No obstante, esta herramienta permite la descarga de sólo 2000 secuencias. En algunos casos, como SACS y SabDab, el número de secuencias descargadas desde las plataformas entregadas por estas bases de datos es mucho mayor a la cantidad de secuencias mencionadas en sus artículos, lo cual puede deberse al tiempo transcurrido desde su publicación, y a la constante adquisición de nuevas entradas. Por otra parte, la base de datos SabDab es la que presenta la mayor redundancia entre los datos descargados y los datos únicos observados, alcanzando un 45 % de redundancia.

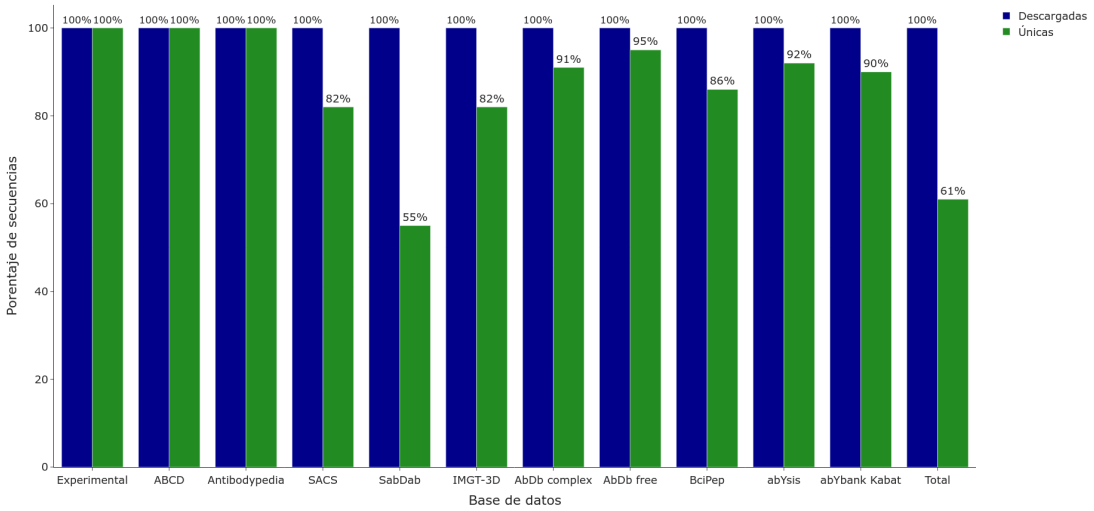


Figura 3.3.1: Resumen base de datos de anticuerpos. Se presenta el porcentaje de secuencias descargadas de cada base de datos utilizada, y el porcentaje de secuencias únicas dentro de cada una de estas bases de datos.

En cuanto a las bases de datos de antígenos, se utilizó IEDB (Vita et al., 2015), Protegen (Yang et al., 2011), SACS (Allcorn and Martin, 2002), ABCD (Lima

et al., 2020), AntigenDB (Ansari et al., 2010), Antibodypedia (Björling and Uhlén, 2008), BciPep (Saha et al., 2005), AbDb (Ferdous and Martin, 2018), SabDab (Dunbar et al., 2014) y CBtope (Ansari and Raghava, 2010). Se incluyen bases de datos de interacción antígeno - anticuerpo como ABCD y Antibodypedia, las cuales fueron accedidas utilizando Uniprot, y proporcionaron un número mayor de datos de antígenos en comparación a aquellos recopilados de anticuerpos en estas bases de datos. Por otra parte, bases de datos de anticuerpos con información estructural fueron utilizadas, en aquellos casos en los cuales los cristales reportaran un antígeno y éste fuese de tipo proteico. Además,se usaron bases de datos de epítopes que reportaran las fuentes desde las cuales se extrajeron estas secuencias, es decir, CBtope (Ansari and Raghava, 2010) y BciPep (Saha et al., 2005). Finalmente, se incluyen las secuencias proporcionadas por Frick (2009) las cuales son referidas como Experimental.

La base de datos TANTIGEN no pudo ser utilizada, debido a que si bien posee un buscador que permite recopilar las entradas entregadas por este sistema, no posee un método de descarga de todas estas entradas recopiladas, por lo cual no pudo ser utilizada. De igual forma, el sistema BloodAntigens proporciona información con respecto a los genes respectivos de cada antígeno, por lo cual la información proporcionada no es de utilidad para los objetivos planteados. Adicionalmente, BloodAntigens no proporciona herramientas para la descarga del conjunto de datos, por lo cual tampoco podrían ser accedidos en caso de poseer secuencias de aminoácidos correspondientes a antígenos. Además, la base de datos de antígenos HPtaa no pudo ser considerada debido a que el sitio web referenciado en el artículo asociado no se encuentra en ingles o español, y no presenta herramientas de traducción. CTdatabse, por otro lado, si bien entrega información correspondiente tanto a secuencias génicas como a secuencias de aminoácidos relacionadas a antígenos, estos datos no pueden ser descargados como conjunto, por lo cual no fueron utilizados. Finalmente, la base de dbPepNeo si bien corresponde a un sistema de neo antígenos, al ser péptidos de tamaño corto fueron considerados como epítopes, sin especificar si estos corresponden a epítopes lineales o estructurales. Además, las referencias proporcionadas por la base de datos a posibles secuencias de antígenos corresponden a artículos, por lo cual no puede ser utilizada para la obtención de estas secuencias.

Al igual que con las bases de datos de anticuerpos, a partir del procesamiento

realizado se pudo comparar el número de secuencias descargadas contra el número de secuencias reportadas en los artículos o sistemas correspondientes a repertorios de antígenos, y la cantidad de antígenos únicos. Una imagen comparativa del número de secuencias descargadas y el número de secuencias únicas por base de dato se presenta en 3.3.2. Las bases de datos que entregaron el mayor número de secuencias de antígenos corresponden a Antibodypedia y IEDB, a pesar de que no son sistemas dedicados a este tipo de moléculas. Por otra parte, algunas bases de datos como ABCD, SabDab y AbDb complex, mencionan en sus artículos o plataformas un mayor número de secuencias asociadas, en comparación a los datos realmente descargados. Por el contrario, otras bases de datos como AntigenDB y BciPep aumentaron el número de entradas disponibles en comparación a lo mencionado en sus artículos.

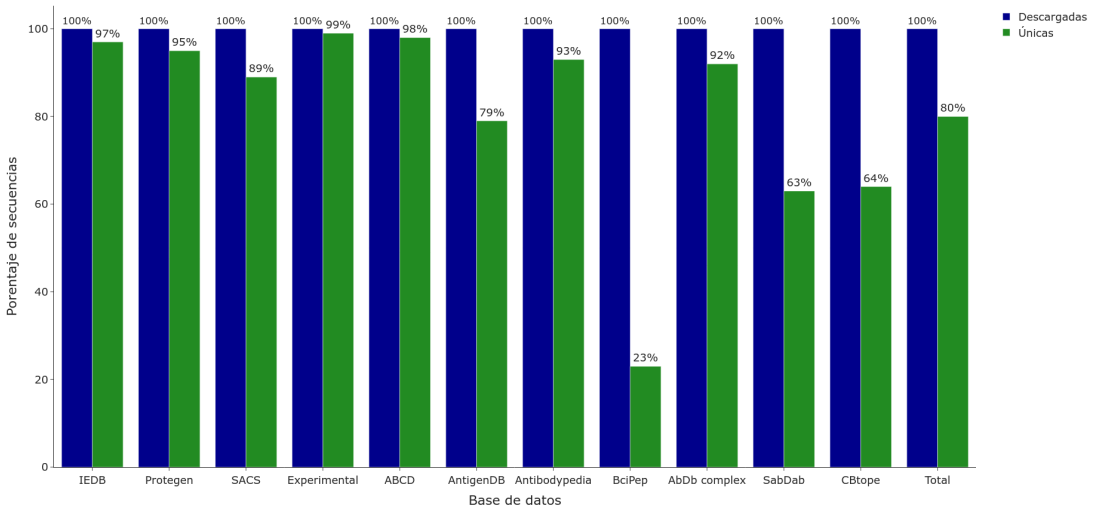


Figura 3.3.2: Resumen base de datos de antígenos. Se presenta el porcentaje de secuencias descargadas de cada base de datos utilizada, y el porcentaje de secuencias únicas correspondientes a cada una de éstas.

De forma similar, las bases de datos de epítopes utilizadas fueron IEDB (Vita et al., 2015), dbPepNeo (Tan et al., 2020), BciPep (Saha et al., 2005) y CBtope (Ansari and Raghava, 2010). Las secuencias de cada base de datos presentan una etiqueta que indica si la secuencia corresponde a un epítope lineal o estructural, a excepción de las secuencias presentadas por dbPepNeo, debido a que no se sabe con certeza si todo el péptido corresponde a un epítope de interacción lineal o se encuentran sólo unos pocos residuos interaccionando de forma estructural. Por otra parte, desde

los datos entregados por dbPepNeo, solo se tomaron las secuencias de péptidos que fueran señaladas con un nivel de confianza alto y medio (Tan et al., 2020). En el caso de epítopes estructurales, estos son representados anteponiendo el número correspondiente a la ubicación del residuo en la secuencia y luego la letra que representa al aminoácido en esta posición. No todos los ejemplos extraídos desde IEDB presentan un antígeno de origen cuya secuencia se encuentre disponible, por lo cual, existen secuencias analizadas que no poseen un contexto con respecto al lugar de la secuencia en la cual se encuentran insertos. Por otra parte, al igual que con las bases de datos de antígenos, se realizó un análisis delos datos descargados desde cada base de datos, comparando el número de entradas reportadas por los sitios web o los artículos asociados con respecto al número de secuencias únicas observadas en el análisis.

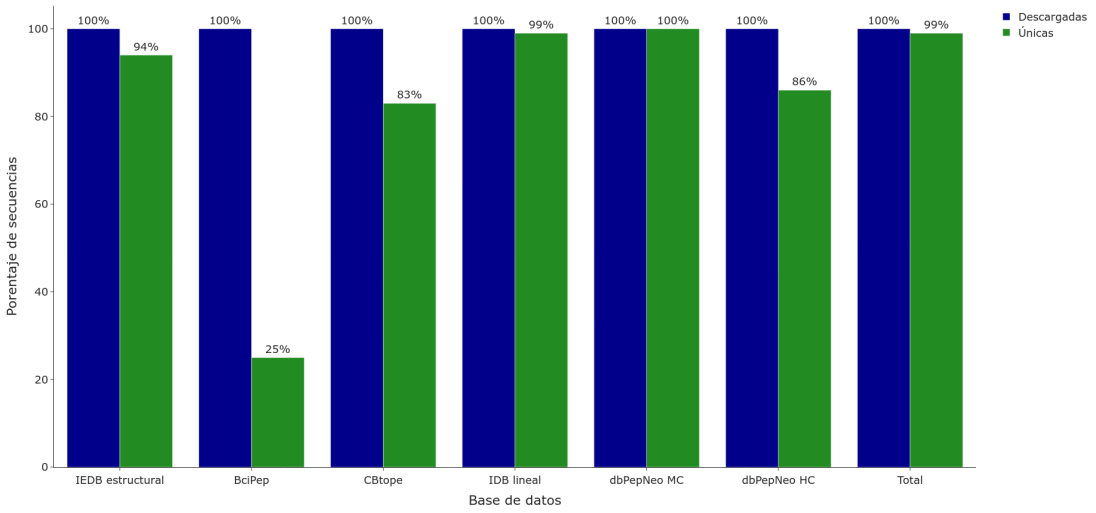


Figura 3.3.3: Resumen base de datos de epítopes. Se presenta el porcentaje de secuencias descargadas de cada base de datos utilizada, y el porcentaje de secuencias únicas observadas dentro de estos recursos.

A partir de las bases de datos analizadas se recopilaron 9,596 anticuerpos, 145,795 antígenos y 1,299,153 epítopes. El aporte realizado por cada uno de estos sistemas a las colecciones correspondientes se representa en las Figuras 3.3.4, 3.3.5 y 3.3.6. En el caso de la colección de anticuerpos, la base de datos SabDab aportó el mayor número de entradas,aun cuando el porcentaje de redundancia de esta base de datos en específico es de alrededor del 50 %. Para la colección de antígenos, AntibodyPedia entrego el mayor número de secuencias. Si bien esta base de datos

corresponde a anticuerpos, por medio de Uniprot se obtuvo un mayor número de secuencias asociadas a antígenos. Finalmente en el caso de epitopes, mas del 99 % de las entradas fueron proporcionadas por la base de datos de IEDB. Por otra parte, se analizó la redundacia observada en los set de datos aportados para cada una de las moléculas de interés, por las bases de datos incluidas. En la figura 3.3.7 se presenta el porcentaje promedio de redundancia para estas colecciones. Este análisis exhibe un porcentaje de redundancia similar entre epitopes y antígenos, mientras que las bases de datos de anticuerpos, de forma general, presentan menor redundancia, a pesar del gran número de entradas repetidas en SabDab.

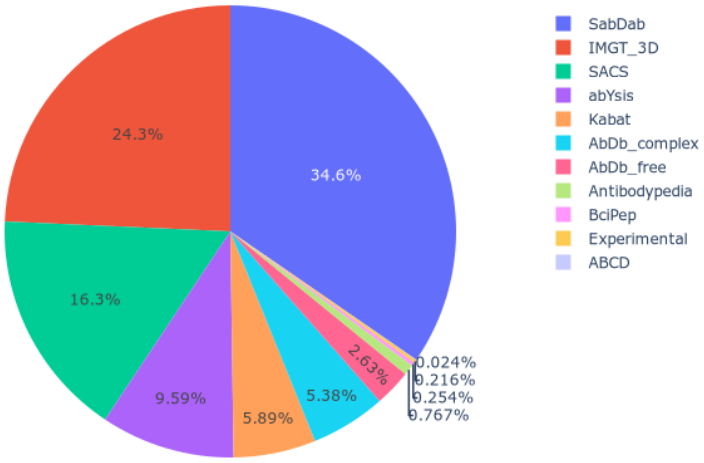


Figura 3.3.4: Aporte realizado a colección de anticuerpos. Se muestra el porcentaje de datos proporcionados por cada base de datos utilizada en el conjunto de entradas de anticuerpos.

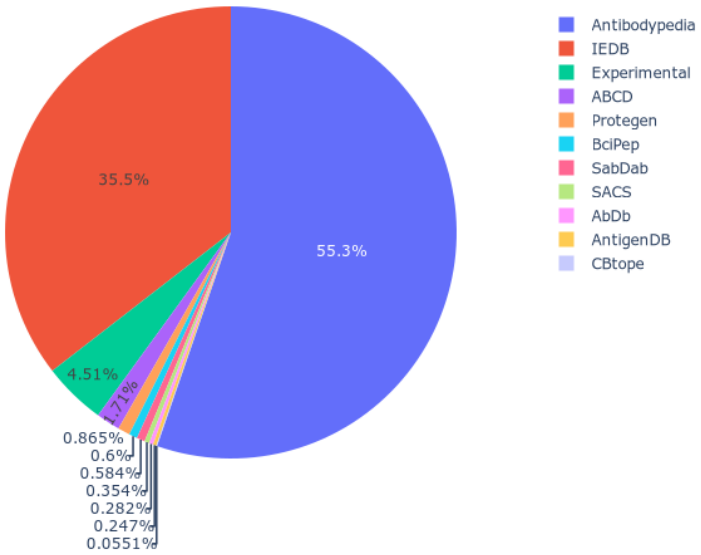


Figura 3.3.5: Aporte realizado a colección de antígenos. Se señala el porcentaje de datos proporcionados por cada base de datos utilizada en el conjunto de entradas de antígenos.

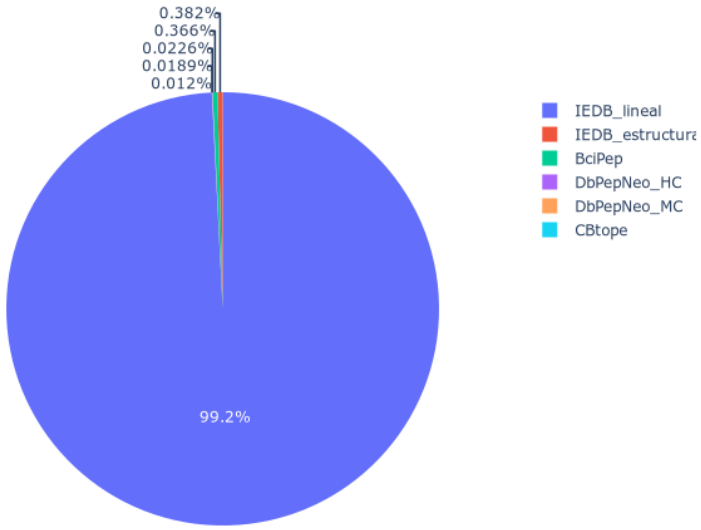


Figura 3.3.6: Aporte realizado a colección de epítopes. Se presenta el porcentaje de datos proporcionados por cada base de datos utilizada en el conjunto de entradas de epítopes.

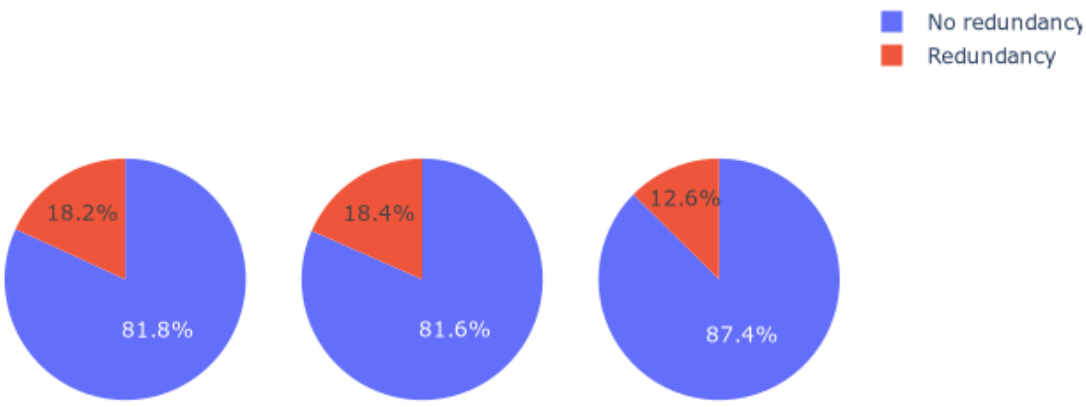


Figura 3.3.7: Redundancia en los set de datos. Se calculó el promedio de redundancia en los set de datos procesados de epítopes, antígenos y anticuerpos respectivamente. La redundancia en los dos primeros grupos es similar, mientras que para los anticuerpos es menor.

A partir de todos los registros recopilados, y las observaciones realizadas con respecto a las diferencias en formato, dispersión de la información accesible y problemas en la accesibilidad a esta, es que se desarrolló una base de datos integral alrededor de estas moléculas.De igual forma, se planteó el diseño e implementación de un sistema que respondiese a las necesidades observadas. De esta forma, el sistema propuesto, actualmente en desarrollo por miembros del grupo de investigación de CeBiB, corresponde a un sistema web que agrupa toda la información recopilada desde estas distintas fuentes, e integra diversas herramientas para el análisis de información referente a las moléculas de interés. Además, presenta la caracterización realizada a las secuencias recopiladas. En el caso de secuencias de antígenos y anticuerpos, cada secuencia posee valores respectivos a las bases de datos en las cuales fue observada, el largo de esta secuencia, el identificador otorgado por la base de datos desde donde se extrajo, el cálculo realizado de la frecuencia de aminoácidos y grupos de aminoácidos, la predicción de carga, densidad de carga, punto isoeléctrico y peso molecular. Además, para estas secuencias se expone la frecuencia de grupos de estructura secundaria predichos, y la secuencia de caracteres que representan a estas estructuras secundarias por cada aminoácido. En adición a esto, se presentan las predicciones de accesibilidad a solvente y desorden de cada secuencia. De forma similar, se muestran las predicciones de términos GO presentados por PredictProtein para las secuencias de antígeno y anticuerpo, las cuales incluyen un valor de predicción. De igual manera, se exhiben las predicciones realizadas por Pfam con respecto a los dominios

encontrados en cada secuencia, el identificador, tipo, evalue y bitscore presentado por estos resultados.

Cuando corresponden a interacciones antígeno-anticuerpo, esta interacción se define por compartir parte del identificador entre ambas secuencias. En el caso de interacciones extraídas desde cristales, el cual representa todas las interacciones obtenidas desde bases de datos exteriores, al no poder establecer de manera sencilla cual es la cadena con la cual interacciona el antígeno cristalizado, la interacción es representada por todas las secuencias obtenidas que posean un código PDB común. Las secuencias de anticuerpo presentan el campo adicional que indica si la secuencia corresponde a una cadena pesada, ligera, o esta información no se específica. Por otra parte, en el caso de entradas correspondiente a antígenos pueden poseer asociadas entradas correspondientes a epítopes dentro de la base de datos generada. Los epitopes contienen información con respecto al identificador presentado por la base de datos de origen, bases de datos que presentan este epítope, el largo de la secuencia y si corresponde a epítope lineal o estructural.

Diversas herramientas han sido integradas al sistema en desarrollo, con la finalidad de facilitar el acceso a datos de interés con ciertas características, y permitir la caracterización y analisis de estas moléculas bajo diversos puntos de vista de interés biológico. En primer lugar se presenta una herramienta de búsqueda avanzada, el cual permite al usuario seleccionar el tipo de molécula de interés, y diversos filtros se despliegan a partir de esta selección. En el caso de anticuerpos y antígenos, el usuario puede usar características como un largo específico, palabras claves referentes a términos GO, estructuras relacionadas, entre otros. Es posible en el caso de anticuerpos, buscar entradas que posean reportada interacción con antígenos, y en el caso de antígenos limitar la búsqueda a entradas que posean epítopes asociados. Además, el sistema integra una herramienta de alineamiento de secuencias, el cual permite buscar dentro de las entradas correspondientes al tipo de molécula de interés, en base a un parámetro de similitud. De forma similar, es posible realizar el mapeo de epítopes lineales en una secuencia de interés. Como resultado se presentan las secuencias de epitopes que se encuentren exactamente en la secuencia proporcionada, y las posiciones en las cuales se presentan.

En la figura 3.3.8 se presenta un resumen con respecto al sistema diseñado en conjunto con el grupo de investigación, y que actualmente se encuentran en etapa de implementación, las fuentes de datos utilizadas y el numero de entradas

recopiladas por cada una de las moléculas comprendidas en la base de datos diseñada. De igual forma, se señala la información correspondiente a cada entrada, y las herramientas integradas al sistema con una breve descripción con respecto a sus funciones.

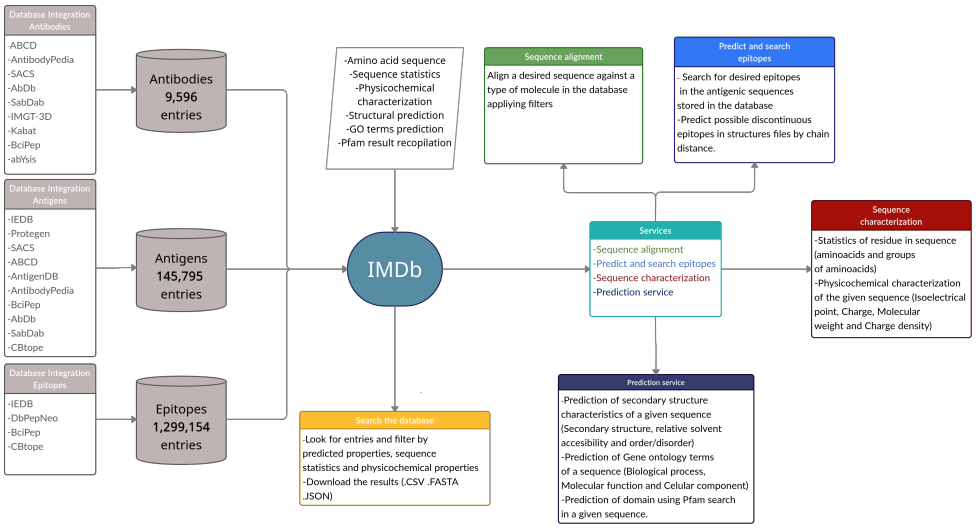


Figura 3.3.8: Esquema representativo del sistema IMDb. El sistema integra tres colecciones de moléculas inmunológicas, cuyos datos fueron obtenidos desde diversas fuentes.Todas las secuencias de anticuerpos y antígenos recopiladas fueron caracterizadas mediante diversos puntos de vista de interés biológico. Adicionalmente, el sistema proporciona una plataforma de búsqueda que permite acceder de forma sencilla a los datos recopilados. Finalmente, se integraron diversas herramientas que permiten estudiar a estas moléculas inmunológicas, como es el mapeo de epítopes, el alineamiento de secuencias y la predicción de interacción antígeno anticuerpo mediante las cadenas reportadas en una estructura tridimensional.

Por otra parte, para propiciar la caracterización de secuencias en base a diversos puntos de vista implementados en la data recopilada, se agregaron distintas herramientas al sistema, que permiten realizar esta caracterización sobre secuencias de interés para el usuario. De esta forma, entregando un archivo fasta a estas herramientas, el usuario podrá obtener de forma sencilla las propiedades de interés sobre sus datos, como pueden ser la información estadística de residuos en sus secuencias, propiedades fisicoquímicas y termodinámicas, predicciones respecto a su estructura secundaria, predicción de términos de Gene Ontology en referencia a funciones, procesos o ubicaciones. Finalmente, el usuario podrá utilizar esta plataforma para realiza una predicción de dominios en cada secuencia entregada de acuerdo a los resultados presentados por Pfam.

Una herramienta de gran interés que no se encuentra a la fecha en ningún sistema analizado corresponde a la predicción de interacción antígeno - anticuerpo mediante uniones electrostáticas débiles. Utilizando un archivo o código pdb la herramienta WhatIf procesa la estructura obteniendo la red optimizada de puentes de hidrógeno, a partir del cual se analizan las distancias entre residuos de distintas cadenas. De completarse con éxito este proceso, el usuario podrá descargar los resultados obtenidos, los cuales serán presentados en una tabla resumen. De igual forma, es posible interactuar con la estructura mediante un visualizador integrado en el sistema, para obtener las vistas de las interacciones identificadas que considere oportunas. De esta forma, el usuario conseguirá identificar de manera tentativa residuos que compongan un epítope estructural en el caso de antígenos, o posibles residuos pertenecientes a la región hipervariable de un anticuerpo.

Un ejemplo del uso que puede darse a esta herramienta se realizó utilizando un cristal con código 1A2Y, el cual corresponde a un complejo anticuerpo - lisozima (Dall'Acqua et al., 1998). Esta molécula corresponde a un análisis mutagenético respecto a la interacción y reconocimiento de un anticuerpo monoclonal de ratón sobre una lisozima de clara de huevo de gallina. La estructura posee 3 cadenas, donde A y B corresponden a la cadena ligera y pesada del anticuerpo monoclonal, y la cadena C pertenece a la lisozima de clara de huevo. Mediante la utilización de la herramienta de predicción de interacción antígeno - anticuerpo se generó una red optimizada de interacciones por puente de hidrógeno, la cual permite identificar los residuos de distintas cadenas que podrían participar en el reconocimiento e interacción entre este anticuerpo monoclonal y la lisozima analizada. Entre las posibles interacciones señaladas por la herramienta, se visualizan en la figura 3.3.9 2 en específico. En B se observa la posible interacción entre el residuo Arg96 de la cadena A y Glu98 de la cadena B. Estos residuos podrían participar en la conexión entre cadena pesada y ligera del anticuerpo, la cual le permite mantener su estructura de Y. Por otra parte, en la sección C de la figura se observa la posible interacción entre la Ser24 de la cadena C y Asp100 de la cadena B. De forma similar, se presenta una posible interacción mediada por una molécula de agua, entre los residuos Gly22 de la cadena C y Arg102 perteneciente a la cadena B. En el artículo asociado a esta estructura se analizan los efectos mutacionales sobre el residuo 100 de la cadena pesada, y el residuo 24 de la lisozima, los cuales fueron identificados como algunos de los cuales participan en el contacto entre

las cadenas de anticuerpo y antígeno. Si bien, de acuerdo a lo mencionado en el artículo, estos residuos no poseen un efecto con respecto al reconocimiento del anticuerpo sobre este antígeno, en el caso de no poseer mayor información un acercamiento mediante la visualización y cálculo de distancias entre cadenas puede ser un buen punto de comienzo para un análisis mas formal.

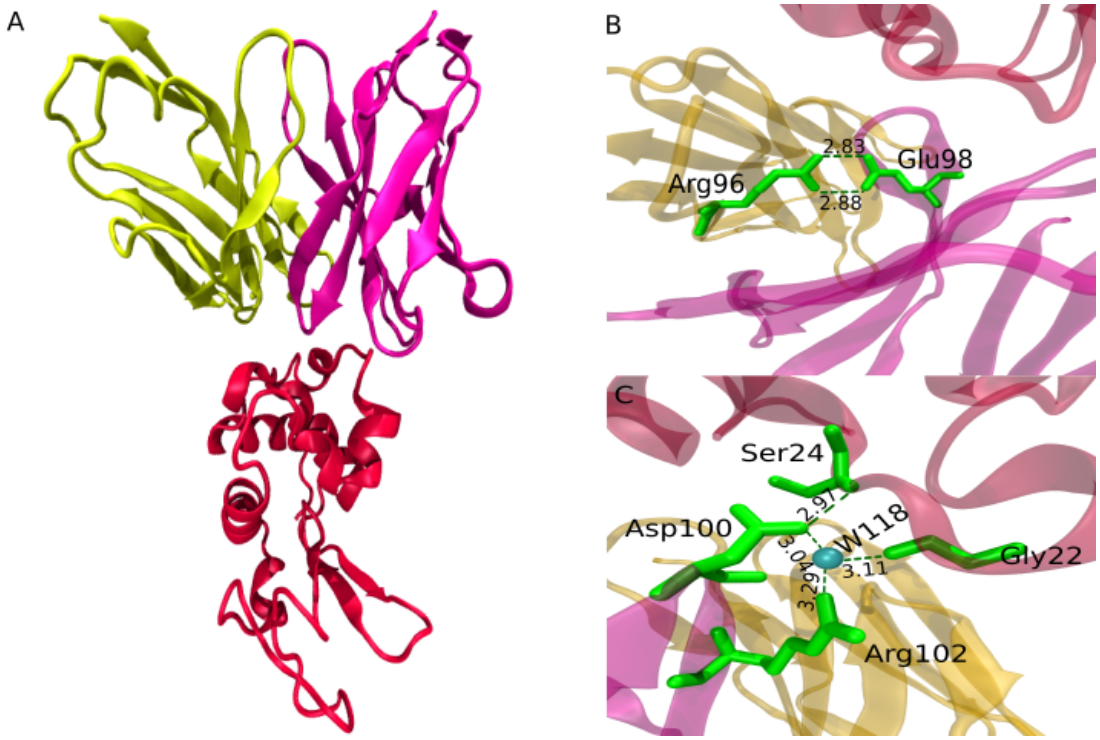


Figura 3.3.9: Predicción de interacciones electrostáticas débiles utilizando el servicio de IMDb sobre la estructura 1A2Y. **A.** Visualización de la estructura general del complejo anticuerpo/lisozima. La cadena A se presenta en color amarillo, la cadena B en color rosado y a cadena C se representa en color rojo. **B.** Se muestran las interacciones entre el residuo Arg96 de la cadena A y el residuo Glu98 de la cadena B. **C.** Se presentan las interacciones entre el residuo Ser24 de la cadena C y el residuo Asp100 de la cadena B. Se observan posibles interacciones mediadas por una molécula de agua W118, la cual parece coordinar los residuos Arg102 de la cadena B y Gly22 de la cadena C.

Debido a los problemas señalados anteriormente, como es la dispersión de los datos en diversos sistemas, la diferencias en los formatos en los cuales se entrega la información, sistemas no disponibles o con secuencias no accesibles para su uso, entre otros, surge el diseño y desarrollo del sistema IMDb. IMDb presenta la ventaja de agrupar el mayor número de secuencias de antígenos, anticuerpos y epitopes disponibles de dominio público, en un formato ordenado y accesible, y con información adicional de interés biológico, como la caracterización fisicoquímica,

la predicción de estructura secundaria, predicción de dominios Pfam, entre otros. Además, el sitio web planteado fue diseñado de forma amigable para el usuario, con una interfaz intuitiva que facilita el acceso a los datos almacenados, mediante búsquedas focalizadas y extracción de la información en distintos formatos. De igual forma, se realizó una comparación con respecto a las herramientas presentadas por cada base de datos analizada e integrada, con el fin de presentar como base aquellas herramientas que generalmente son requeridas por los usuarios de estos sistemas. Finalmente, se integraron herramientas, no disponibles en otras plataformas, al sistema diseñado, con el fin de entregar al usuario mayor facilidad para el análisis de secuencias de interés.

Toda la data recopilada por medio de este proceso será utilizada para evaluar la metodología diseñada para la predicción auto antígenos, los cuales podrían contribuir en el testeo del modelo ensamblado diseñado en el capitulo anterior. Así, sería posible presentar nuevos ejemplos de interacción entre los anticuerpos entregados por (Frick, 2009) y las posibles secuencias de auto antígenos incluidas dentro de la base de datos implementada.

3.4. Conclusiones

Dado el aumento en el número de datos generados bajo diversas metodologías experimentales, ha sido necesario diseñar e implementar distintos sistemas de almacenamiento. Grandes plataformas web han sido levantadas en torno a esta información, entregando no solo la información recopilada, sino que herramientas que permitan analizar o transformar esta información. No obstante, diversas dificultades surgen en el aumento de la información disponible. Las posibles problemáticas van desde la necesidad de computadores con gran capacidad de memoria para el almacenaje de estos datos, a la imposibilidad de revisarlos exhaustivamente. Dado este último punto, muchos sistemas recurren a entradas entregadas por usuarios, o a la recopilación automática. Estas metodologías de recopilación, sin la debida validación, pueden generar entradas con información contradictoria, o entradas repetidas.

Las información redundante representa una dificultad importante al momento de trabajar con uno o varios de estos sistemas. La sobre estimación de la información disponible puede llevar a errores relevantes, en especial, al ser utilizados en

metodologías de entrenamiento de modelos. Un conjunto de datos redundante puede producir confusión sobre un algoritmo, sobre ajustando el modelo generado a un patrón o característica menos común en el universo general de datos. Debido a esto, si se desea trabajar con datos disponibles en bases de datos y plataformas web, es necesario aplicar técnicas computacionales para filtrar esta información y eliminar la redundancia disponible. Para ejecutar este procedimiento, es requerido conocimiento con respecto a herramientas computacionales o lenguajes de programación, que automaticen la tarea de revisión sobre un gran volumen de datos. Además de estas dificultades, el aumento en el conjunto de sistemas que almacenan el mismo tipo de información exige una revisión exhaustiva de distintas plataformas, lo cual es costoso en tiempo. Adicionalmente, estas plataformas utilizan diversos formatos en la entrega de la información, lo cual dificulta su procesamiento, y acentúa aun más el tiempo requerido para la recopilación de los datos de interés.

Dentro de las bases de datos revisadas fue común observar redundancia en los datos, lo cual evidencia una falta de control o revisión sobre la información recopilada y almacenada en estas plataformas. Por otra parte, muchos de los datos presentes en las bases de datos con mayor número de registros, de acuerdo a sus sitios web o artículos, corresponden a entradas proporcionadas por laboratorios o empresas privadas, por lo cual, la información no es accesible para su estudio o análisis.Por otra parte, en cuanto a bases de datos dedicadas a la recopilación de información de las moléculas inmunológicas estudiadas, no se encontraron sistemas que incluyeran el mayor número de secuencias de aminoácidos de anticuerpos humanos de dominio público, antígenos de cualquier tipo, o epítopes lineales como estructurales. Sistemas como IMGT/mAb-DB, Antibodypedia, HaptenDB y TANTIGEN no permiten la descarga de los datos almacenados. Mientras que otros sistemas como no se encontraban disponibles, como es el caso de bNAber, o los datos descargados presentaron errores de formato, situación observada en SEDB.

La recopilación de datos empleada en este trabajo, el diseño e implementación de una base de datos y su disposición en un sistema web esperan solucionar algunas de las problemáticas observadas. El sistema web, a la fecha aún en implementación por el grupo de investigación de CeBiB, presenta un conjunto de ventajas sobre los sistemas analizados. En primer lugar, incluyendo el mayor

número de secuencias de dominio público para cada una de las moléculas a la fecha, se espera que los usuarios de IMDb aumenten su productividad, disminuyendo el tiempo de búsqueda en diversas plataformas. Por otro lado, la disposición de un conjunto de herramientas de análisis de anticuerpos, antígenos, epítopes y su interacción, podrían complementar la investigación sobre estos datos. Por sobre esto, la caracterización realizada a las secuencias de antígenos y anticuerpos recopiladas posibilita el análisis bajo puntos de vista biológicos, ahorrando de esta manera el tiempo y las dificultades de predecir y obtener estas características de manera autónoma.

Finalmente, el sistema web IMDb fue diseñado con una interfaz amigable e intuitiva con el usuario, de forma que el acceso a la información de interés y la utilización de las herramientas sea simple y cómodo. Un conjunto de filtros permite acotar la información extraída desde la base de datos implementada, entregando de esta forma sólo las entradas correspondientes a la consulta realizada. La agrupación tanto de información de secuencia, como estructuras relacionadas, regiones epítopes, características estructurales y propiedades fisicoquímicas, convierten a la plataforma IMDb en una importante fuente de información. Este sistema no solo permite encontrar el mayor número de datos públicos sobre anticuerpos humanos, antígenos y epitopes, sino que también, entrega variadas herramientas de análisis, que van desde l caracterización de secuencias propias, a mapeo de epítopes lineales y visualización de complejos. Este ultimo corresponde a una potencial herramienta de gran utilidad para la predicción de posibles epítopes estructurales, así como paratopes dentro del anticuerpo. Sin embargo, este recurso debe ser utilizado con cautela, considerando que las predicciones realizadas por WhatIf no corresponden a una prueba real con respecto a los sitios de interacción, por el contrario, representan un punto de partida para posibles nuevos análisis en complejos poco analizados.

Capítulo 4

Proyecciones y trabajo a futuro

El sistema inmune ha desarrollado diversas técnicas y metodologías para distinguir entre las moléculas propias del organismo y aquellas correspondientes a agentes foráneos. Esto permite que no se produzcan reacciones en contra del mismo organismo. El mal funcionamiento de este sistema de control puede provocar un conjunto de enfermedades auto inmunes, tales como diversos tipos de diabetes (Buzzetti et al., 2017), artritis reumatoide (Alam et al., 2017), celiaquía (Christophersen et al., 2019), entre otras afecciones. Por otra parte, en algunas ocasiones este mecanismo de defensa puede corresponder a una desventaja frente a enfermedades que involucran a las mismas células del cuerpo. Un ejemplo de esto, que ha sido estudiado ampliamente, es el cáncer (Banchereau and Palucka, 2018; Fritz and Lenardo, 2019).

Debido a la naturaleza de esta enfermedad, en el cual las células del cuerpo presentan mutaciones aberrantes, el organismo posee dificultades para su identificación. Estas células presentan auto antígenos muy similares a los auto antígenos de células sanas, por lo cual, evitan los mecanismos de defensa del organismo. Igualmente, estas mutaciones pueden conducir a aumentar el número de copias expresadas de un auto antígeno (Zhang et al., 2018). Además, dadas las mutaciones presentadas, algunas células presentan nuevas secuencias proteínicas conocidas como neo antígenos. Es por esto que los auto antígenos presentan una oportunidad importante en el estudio de su reconocimiento, y en el desarrollo de estrategias terapéuticas para diversos tipos de cáncer (Fritz and Lenardo, 2019).

Previos estudios han demostrado que, en individuos sanos, se han observado células

T con capacidad de reconocer antígenos tumorales, como también auto antígenos genéticamente anormales, todo esto sin afectar la capacidad de producción de células reguladoras T por auto antígenos (Tanaka and Sakaguchi, 2019). Esto ilumina el camino hacia diversas estrategias para el tratamiento inmunológico del cáncer. Uno de los acercamientos con gran atractivo corresponde al desarrollo de vacunas que promuevan la activación del sistema inmunológico para la eliminación de células tumorales (Banchereau and Palucka, 2018). Para el desarrollo de estas vacunas es necesario reconocer a los auto antígenos expuestos exclusivamente por estas células cancerígenas, evitando así dañar a células sanas pertenecientes al mismo tejido. El reconocimiento e identificación de estos auto antígenos puede ser altamente costoso, tanto en recursos económicos como en tiempo (Chesson and Zloza, 2017). Es por esto que, estrategias de reconocimiento costo-eficientes son inmensamente necesarias.

Diversas estrategias podrían ser utilizadas para el reconocimiento de estas moléculas mediante metodologías in silico. Sin embargo, en primer lugar es necesario identificar patrones o grupos dentro de estos antígenos, que sirvan de guía para el reconocimiento de posibles nuevos antígenos dentro de un grupo de antígenos no categorizado. A partir de la secuencia de aminoácidos es posible calcular o predecir diversas propiedades. Algunas de estas, podrían entregar la información necesaria para identificar la función, la ubicación y la participación a nivel biológico de la secuencia presentada. Además de esto, conocer posibles dominios conservados puede ayudar a discernir las secuencias de interés en base a regiones conservadas a nivel de secuencia.

En base a la idea anteriormente mencionada se utilizó el programa metastudent (Hamp et al., 2013) perteneciente a la suite de Predict Protein (Yachdav et al., 2014), el cual permite predecir a partir de una secuencia de aminoácidos, términos Gene Ontology (Consortium, 2021) relacionados a función molecular, proceso biológico y componente celular. Esta predicción fue realizada para las secuencias de auto antígenos proporcionadas por (Frick, 2009), las cuales se encuentran comprobadas experimentalmente. Adicionalmente, se realizó una predicción de dominios utilizando la base de datos Pfam (El-Gebali et al., 2019). Estas predicciones sobre cada secuencia permitieron observar grupos, o conjuntos de secuencias que compartían términos y dominios predichos por las herramientas mencionadas anteriormente.

Estos grupos identificados a partir de las secuencias auto antigénicas fueron utilizados posteriormente para buscar dentro de la base de datos implementada en IMDb. Para esto, todas las secuencias de antígenos recopiladas dentro de la base de datos fueron caracterizadas mediante predicciones de términos GO y Pfam realizadas por metastudent y Pfam respectivamente. De esta forma, secuencias dentro de la base de datos que compartieran predicciones con alguno de los grupos identificados en las secuencias experimentales fueron almacenadas en un archivo con posibles secuencias auto antígenas. Este filtro realizado permite trabajar con secuencias que se encuentran dentro de un mismo espacio de propiedades y dominios Pfam.

Mediante la utilización de información filogenética extraíble desde la comparación de secuencias, así como la aplicación de métodos estadísticos, sería posible el desarrollo de un modelo de clasificación de secuencias auto antígenas. Esto en consideración, en primer lugar, del trabajo con secuencias dentro de un mismo espacio de características, correspondientes a las observadas en las predicciones realizadas sobre las secuencias auto antígenas experimentales.

4.1. Metodología

4.1.1. Alineamiento de secuencias auto antígenas

En primer lugar, las secuencias de auto antígenos relacionadas a células de leucemia proporcionadas por Frick (2009) fueron filtradas con respecto a largo, en base a distribución de largos de secuencias observadas. De esta forma, se redujo el número de secuencias de alrededor de 8000 a aproximadamente 6000. Estas secuencias filtradas fueron alineadas utilizando el programa Clustal Omega (Sievers and Higgins, 2014), utilizando los parámetros por defecto. Cada secuencia de antígeno fue alineada con respecto a las otras N-1. De este alineamiento múltiple se obtuvo la matriz de distancia entre secuencias, el cual posee valores que representan una distancia evolutiva entre secuencias.

4.1.2. Distribución de distancias

El archivo obtenido en el paso anterior fue procesado, para obtener el triangulo superior de los valores de distancia entre las secuencias. Luego, se obtuvo la

distribución de estas distancias, mediante un script implementado en Python v3.6. Para cada secuencia, se obtuvo el promedio de distancia con respecto a las secuencias alineadas, y a partir de esto se obtuvo un histograma de distribución de distancias.

4.1.3. Categorización con respecto a distribución

En base a la distribución de distancias obtenida anteriormente, se realizo una categorización en base a cuartiles. A partir de esta categorización se establecieron límites con respecto a valores de distancia para determinar putativamente a una secuencia como auto antígena.

4.1.4. Alineamiento contra secuencias experimentales

Las secuencias identificadas en la base de datos IMDb, que compartiesen predicciones con respecto a términos GO y dominio Pfam con alguna de las secuencias experimentales, fueron posteriormente filtradas de acuerdo a los límites de largo establecidos en el paso 4.1.1 de la metodología. Las secuencias filtradas fueron posteriormente alineadas contra las secuencias auto antígenas identificadas experimentalmente. Cada una de las secuencias de la base de datos fueron alineadas en contra de todas las secuencias de auto antígenos, y se obtuvo el vector de distancias calculado por Clustal Omega.

4.1.5. Comparación y categorización

Finalmente, una vez obtenido este vector de distancias para cada secuencia, cada una de estas será comparada con respecto a la distribución de distancias generada anteriormente. De esta forma, cada distancia presente en el vector es traducida en un voto, con respecto a si la secuencia analizada corresponde o no a una probable secuencia de auto antígeno. Así, se utilizará la estrategia de votación para determinar la clase de la secuencia a predecir. En esta estrategia, la mayoría de votos sobre una categoría corresponderá a la predicción final.

4.2. Resultados parciales

De la caracterización realizada a las secuencias experimentales, se observó un conjunto de predicciones de términos GO para función molecular, proceso biológico y componente celular que presentaban mayor frecuencia. En cuanto a función molecular, el término con mayor frecuencia observado corresponde a "binding", que de acuerdo con lo revisado en QuickGO, pertenecen a moléculas con capacidad de enlace selectivo, no covalente, también definidas como ligandos. El segundo término observado con mayor frecuencia corresponde a la unión de tri fosfatos ribonucleósidos de purinas. Finalmente, para función molecular se observo también frecuénteme que metastudent no produjo predicciones, lo que se traduce en "Sin Resultados". En la Figura 4.2.1 se presentan todos los términos de funcion molecular predichos por metastudent para el conjunto de secuencias auto antígenas.

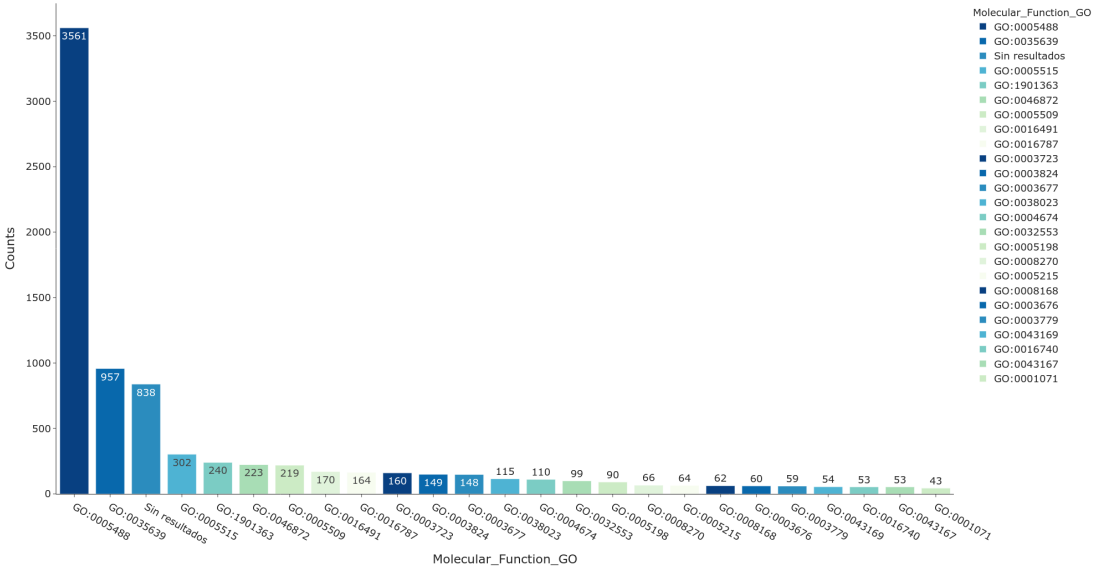


Figura 4.2.1: Histograma términos GO función molecular auto antígenos. En la figura se presenta la frecuencia con la cual se observó cada término predicho en las secuencias auto antígenas determinadas experimentalmente.

Por otra parte, para proceso biológico el término observado con mayor frecuencia corresponde a proceso metabólico de proteínas. Este término hace referencia a cualquier tipo de proceso, anabólico o catabólico, que permite transformar un sustrato. El segundo resultado mas frecuente es "Sin resultados". En cuanto al tercer término mayormente observado en las secuencias auto antígenas, corresponde directamente a "proceso biológico.[el] cual, de acuerdo a QuickGO, se utiliza

para proteínas con asociación a un proceso biológico desconocido. La Figura 4.2.2 muestra el número de veces que se observó cada predicción realizada por metastudent en el set de datos experimental.

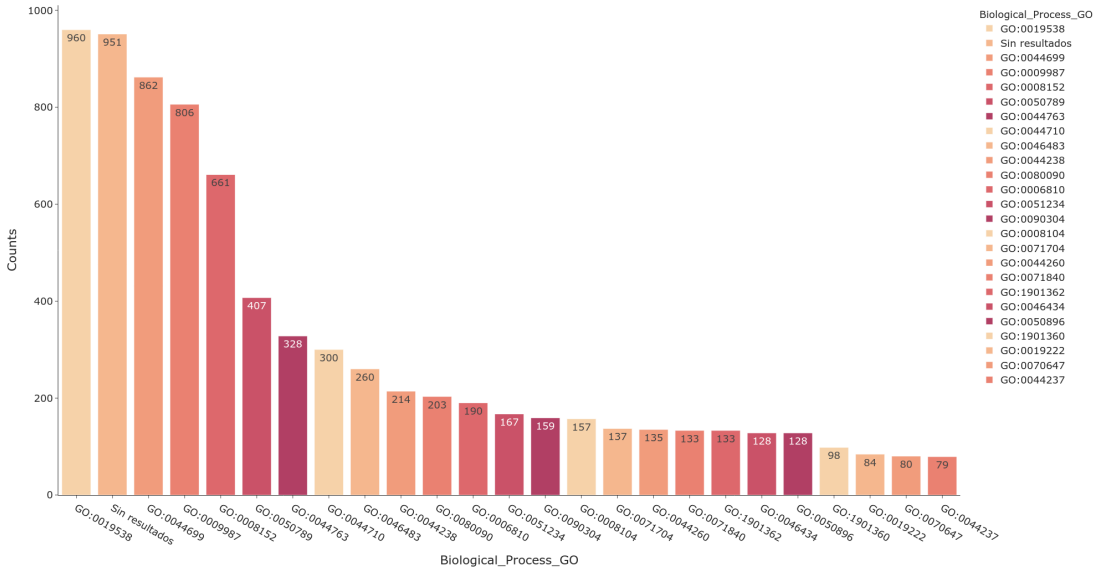


Figura 4.2.2: Histograma términos GO proceso biológico auto antígenos. En la figura se presenta la frecuencia con la cual se observó cada término predicho en las secuencias auto antígenas determinadas experimentalmente.

Mientras tanto, para términos asociados a componente celular, se observó con mayor frecuencia un término ya obsoleto, denominado parte celular. Este término hace referencia a cualquier parte de la célula. El segundo término, también actualmente obsoleto, corresponde a componente intracelular, es decir, contenido entre las membranas plasmáticas. En cuanto al tercer término observado con mayor frecuencia, este corresponde a componentes de organelos intracelulares. Este se utiliza para moléculas ubicadas en el núcleo, mitocondrias, plástidos, vacuolas, vesículas, ribosomas y el cito-esqueleto. En la Figura 4.2.3 se presenta el conjunto de términos GO predichos para componente celular, y la cantidad de veces que fueron observados en el set de datos de auto antígenos.

Adicionalmente, las predicciones realizadas para dominios Pfam, en su mayoría, fueron infructuosas. Las búsquedas de dominios realizadas en Pfam, para muchas de las secuencias, no arrojaron resultados. En cambio, la segunda predicción de dominio con mayor frecuencia corresponde a un dominio de quinasa. Similarmente,el tercer dominio mas frecuente es un dominio de tirosina quinasa. Extrañamente, el cuarto dominio más frecuente corresponde al dominio V-set

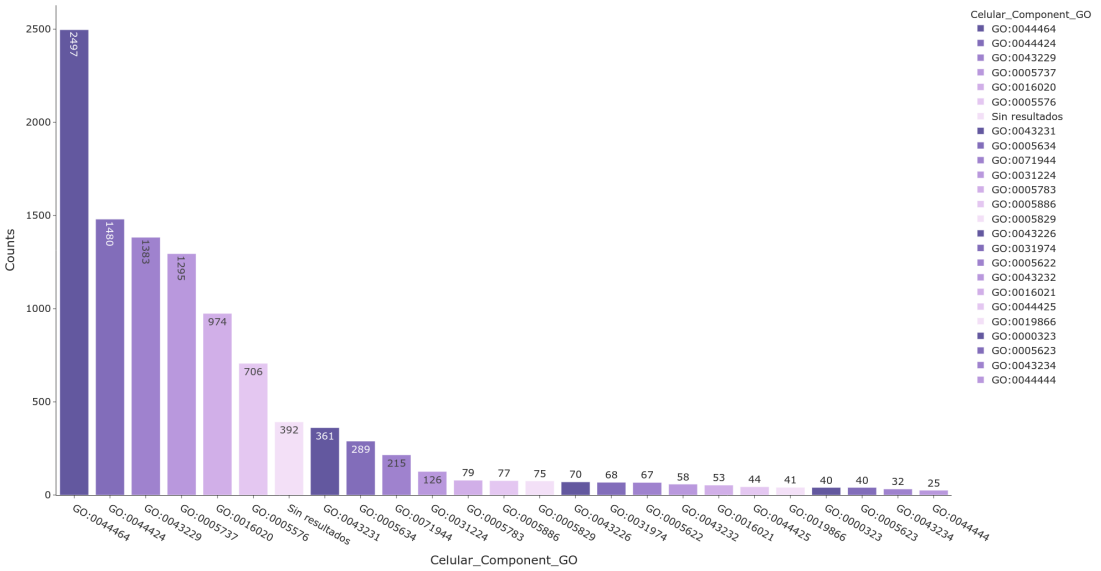


Figura 4.2.3: Histograma términos GO componente celular auto antígenos. En la figura se presenta a frecuencia con la cual se observó cada término predicho en las secuencias auto antígenas determinadas experimentalmente.

de inmunoglobulina. Sin embargo, de acuerdo a lo señalado por la descripción de InterPro, este dominio también es visualizado en proteínas de adhesión a membranas de mielinas, en receptores de proteínas tirosina quinasas, y en proteínas de muerte celular programada. EL conjunto de dominios predicho para estas secuencias se presenta en la Figura 4.2.4, así como el número de veces que se observó cada dominio.

En cuanto a las secuencias de antígenos almacenadas en IMDb, se realizó de igual forma esta caracterización de función molecular, proceso biológico, componente celular y dominios Pfam, utilizando las herramientas mencionadas anteriormente.Con respecto a funciones moleculares predichas, dentro de las secuencias de la base de datos la mas común coincide con la más predicha dentro de las secuencias de auto antígenos, correspondiendo a unión o binding. En segundo lugar, se observa que no se obtuvieron resultados para las predicciones de función molecular. El tercer resultado mas frecuente para este término corresponde a la unión de tri fosfatos ribonucleósidos de purinas, el cual es el segundo término mas frecuente en las secuencias experimentales. Esto se puede apreciar en la Figura 4.2.5, donde se muestran las predicciones de función molecular para las secuencias de antígenos en la base de datos IMDb.

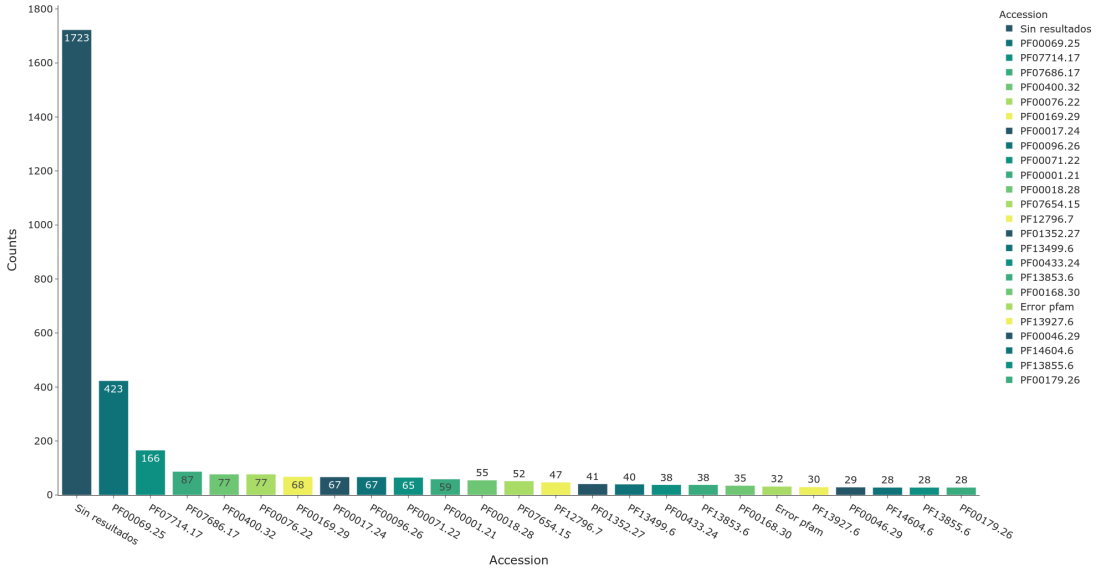


Figura 4.2.4: Histograma dominios Pfam auto antígenos. En la figura se presenta a frecuencia con la cual se observó cada dominio predicho en las secuencias auto antígenas determinadas experimentalmente.

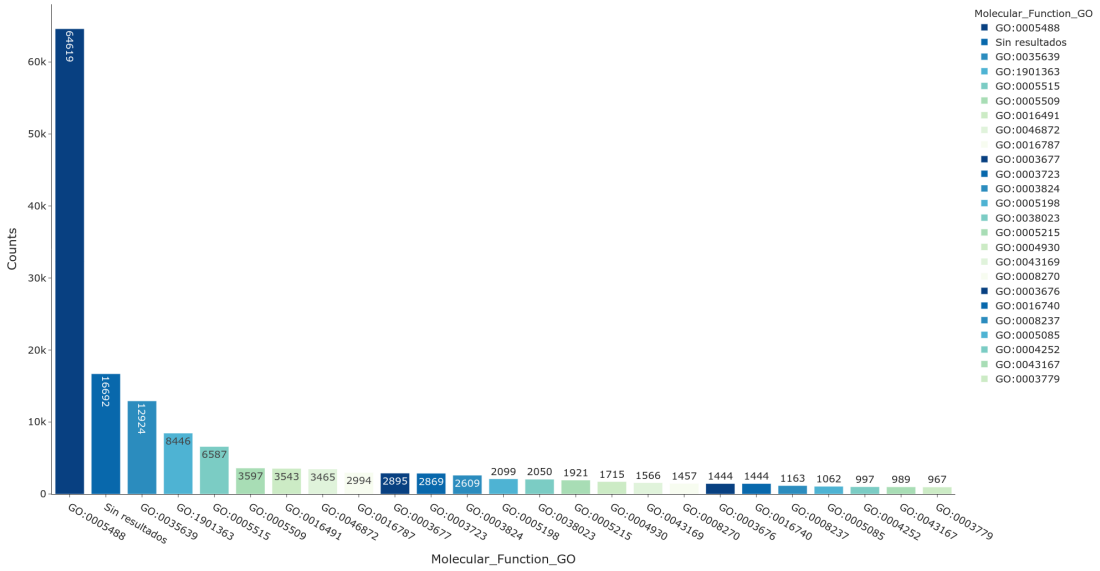


Figura 4.2.5: Histograma términos GO función molecular antígenos IMDb. En la figura se presenta la frecuencia con la cual se observó cada término predicho en las secuencias de antígenos almacenadas en la base de datos.

En cuanto a las predicciones realizadas para proceso biológico, se observó que en su mayoría no se obtuvieron resultados para las secuencias almacenadas en IMDb. Por otra parte, la segunda predicción con mayor número de secuencias registradas corresponde a proceso biológico, la cual es la tercera mas frecuente

dentro de las secuencias de auto antígenos experimentales. El tercer término de proceso biológico mas frecuente dentro de la base de datos, para las secuencias de antígenos, corresponde a proceso biológico, el cual posee la misma descripción que el tercer término mas predicho para las secuencias auto antígenas. Esto se debe a que el código GO:0044699 es un identificador secundario para este término de proceso biológico. En la Figura 4.2.6 se observan los términos predichos para las secuencias de antígenos almacenadas en la base de datos.

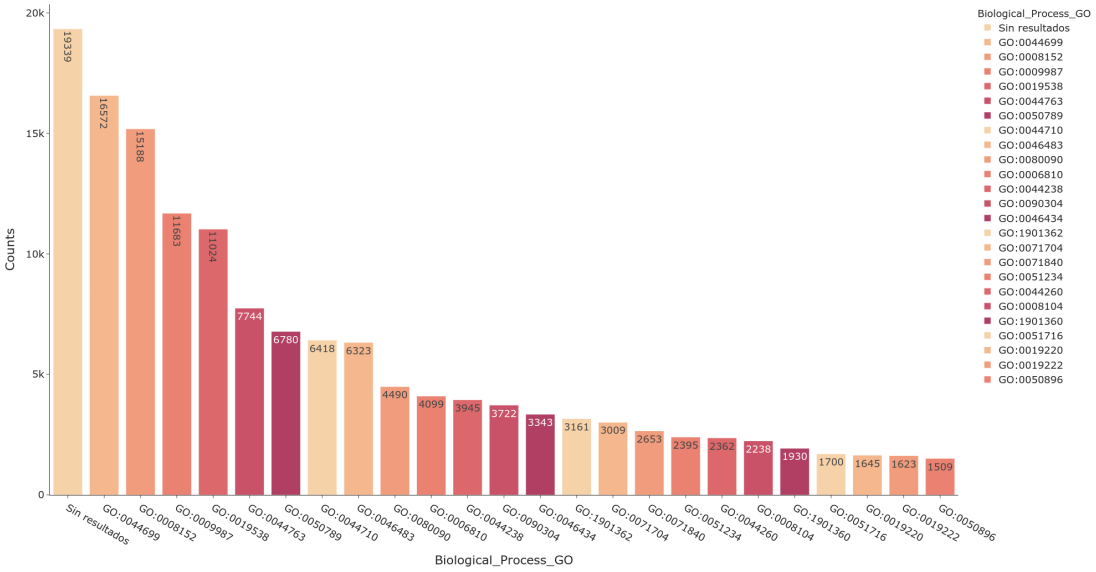


Figura 4.2.6: Histograma términos GO proceso biológico antígenos IMDb. En la figura se presenta la frecuencia con la cual se observó cada término predicho en las secuencias de antígenos almacenadas en la base de datos.

Por otra parte, las predicciones realizadas por metastudent con respecto a los componentes celulares para las secuencias de la base de datos, presentan en primer lugar al mismo término observado en el set de datos de secuencias auto antígenas. Esta predicción esta asociada a un término obsoleto de parte celular. El segundo término de componente celular mas frecuente corresponde a componente intracelular, el cual fue la tercera predicción mas frecuente entre las secuencias de auto antígenos. El tercer componente celular observado con mayor frecuencia dentro de la base de datos corresponde al término relacionado a citoplasma. Este término se utiliza para moléculas ubicadas dentro de la célula, sin incluir ningún organelo o las membranas celulares. La Figura 4.2.7

Por otra parte, las predicciones realizadas de dominios Pfam sobre las secuencias de antígenos en la base de datos IMDb fueron en su mayoría, infructuosas, no

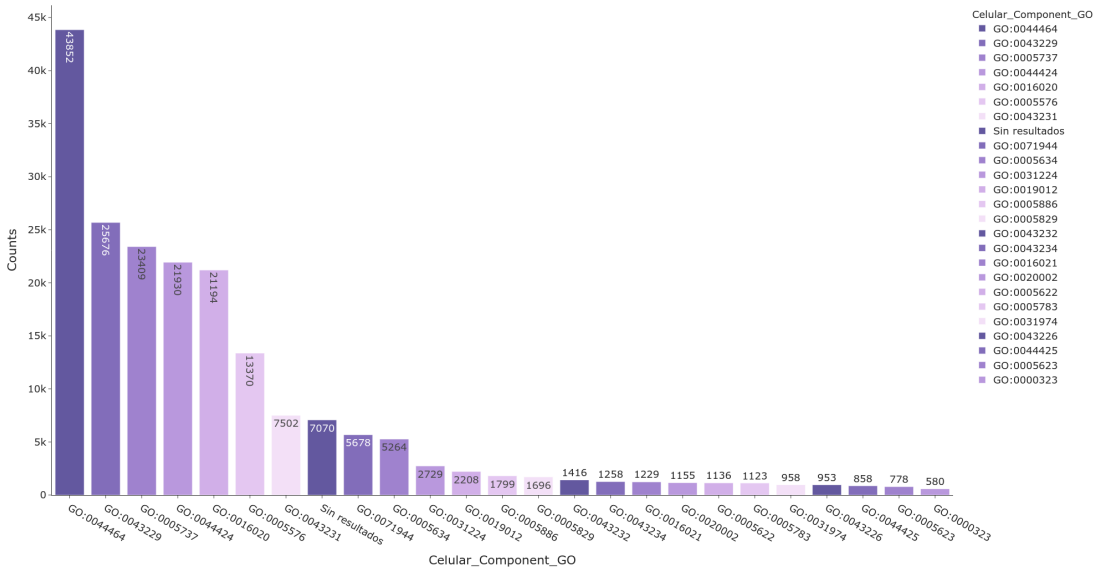


Figura 4.2.7: Histograma términos GO componente celular antígenos IMDb. En la figura se presenta la frecuencia con la cual se observó cada término predicho en las secuencias de antígenos almacenadas en la base de datos.

arrojando resultados con respecto a posibles dominios. En segundo lugar con respecto a frecuencia de predicciones, se observa el dominio de proteína quinasa, igual a como se observó en las secuencias de auto antígenos. El dominio de zinc finger, o dedos de zinc, se observó como la tercera predicción mas frecuente en la base de datos. Proteínas con este dominio poseen diversas funciones, desde reconocimiento de regiones de ADN y activación de la transcripción, a plegamiento de proteínas y control de la apoptosis.

De acuerdo a los términos observados, muchas de las proteínas de auto antígenos poseen funciones quinasas, las cuales modifican sustratos mediante fosforilación. Esto se relaciona tanto con los dominios mayormente observados como con las funciones moleculares mayormente predichas, y la predicción mas frecuente de proceso biológico. Por otra parte, el componente celular al cual puedan pertenecer estos antígenos en gran número de secuencias no queda definido, debido a términos obsoletos y descripciones poco específicas.

Para las secuencias de antígenos extraídas desde IMDb se observaron resultados similares con respecto a las secuencias de auto antígenos experimentales. Los términos GO predichos con mayor frecuencia para función molecular son los mismos observados para los auto antígenos. Situación similar se observó respecto

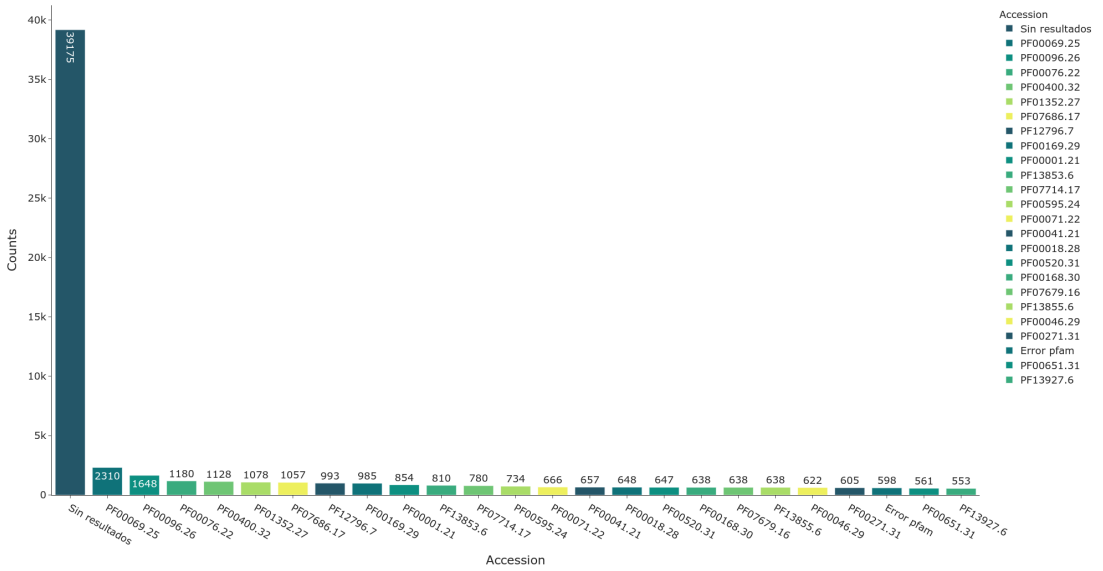


Figura 4.2.8: Histograma dominios Pfam antígenos IMDb. En la figura se presenta la frecuencia con la cual se observó cada término predicho en las secuencias de antígenos almacenadas en la base de datos.

a proceso biológico y componentes celulares. El mayor número de diferencias observadas se encuentran en los dominios Pfam, debido a que en la base de datos el tercer dominio predicho con mayor frecuencia corresponde a dedos de zinc. No obstante, sigue siendo dominio de proteína quinasa el segundo mas frecuente.

Estos resultados obtenidos pueden señalar que, en el universo de auto antígenos general, este tipo de características son observadas frecuentemente. Puede deberse a su naturaleza, que moléculas con funciones o procesos asociados definidos sean reconocidos con mayor frecuencia por los organismos como moléculas antígenas. Es posible que los factores definitivos para la clasificación de auto antígenos se encuentren a nivel de secuencia lineal, o de características estructurales definidas.

4.2.1. Métodos de clasificación binaria basados en propiedades filogenéticas

Dado lo expuesto anteriormente, el problema de identificación de secuencias de auto antígenos puede ser enfocado como un sistema de clasificación. De manera inmediata surge la idea de diseñar e implementar modelos predictivos basados en estrategias de algoritmos de aprendizaje supervisado. Sin embargo, para armar el conjunto de datos inicial de entrenamiento, se requiere de un número igual o similar

de secuencias que represente la clase negativa y a partir de ello, diseñar técnicas de codificación y entrenar modelos empleando alguna estrategia de Machine Learning. No obstante, nada asegura el correcto funcionamiento del predictor ni tampoco aseguramos fehacientemente que las secuencias consideradas como negativas no presenta la condición de auto antígenos.

Como alternativa a dicho planteamiento y con el fin no sólo de trabajar en este proyecto si no que de plantear un método alternativo, se diseñó un sistema de clasificación basado en las propiedades filogenéticas de las secuencias de una familia o que presentan alguna característica en común.

La Figura 4.2.9 resume las etapas principales de esta metodología. Primero, se alinean de a pares todas las secuencias de un conjunto de datos empleando alguna estrategia de alineamiento de preferencia. En una segunda etapa, cada resultado de los alineamientos se emplean para formar la distribución de conservación del conjunto de datos. Luego por medio de técnicas de distribución cuantil se categoriza la distribución y se obtienen los rangos de conservación. Estos rangos pueden variar dependiendo del número de cuantiles que se tome. Además, su interpretación está condicionada al tipo de score que genere la estrategia de alineamiento. Finalmente, un nuevo conjunto de secuencias que se desea trabajar, se someten al proceso de clasificación, donde cada secuencia se alínea contra todas las secuencias del conjunto de datos target y se obtienen la totalidad de los scores, luego a partir de ellos y empleando los rangos obtenidos en la etapa de categorización, se puede clasificar a qué cuantil pertenece la secuencia evaluada contemplando un sistema de votación. Finalmente, las secuencias que demuestren una alta homología con respecto al total de secuencias en el conjunto de datos target representarán candidatos a presentar dicha actividad/función/propiedad.

Remarcablemente, esta metodología ha sido previamente implementada y validada por el co-tutor de este trabajo de título, obteniendo resultados satisfactorios en diferentes casos de estudio. No obstante, la aplicación de esta metodología al conjunto de datos de secuencias de auto antígeno a evaluar en este proyecto se ha trabajado aún debido a temas de costo computacional requeridos para alinear y evaluar la totalidad de secuencias existentes en este conjunto de datos. Solamente se ha obtenido la matriz de distribución y generado los sistemas de categorización, empleando como estrategia de alineamiento clustalO.

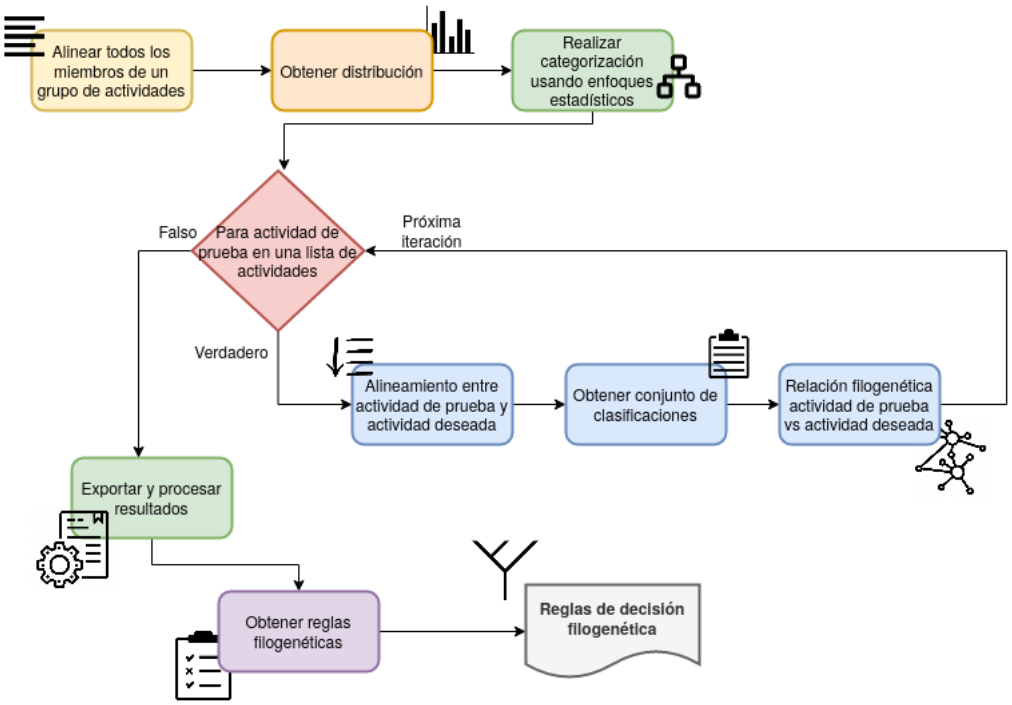


Figura 4.2.9: Esquema para sistema de clasificación propuesto. Para las secuencias correspondientes a una actividad de interés, en primer lugar, se realiza un alineamiento y se obtiene su distribución de distancia, o conservación. Esta distribución es utilizada para realizar una categorización, empleando estrategias estadísticas. Posteriormente, cada secuencia dentro de un conjunto de secuencias de putativas para esta actividad es alineada contra el conjunto de secuencias a partir del cual se realizó la categorización. Los valores obtenidos a partir de cada alineamiento son utilizados para obtener las correspondientes categorías y relaciones filogenéticas. Una vez finalizado este ciclo, los resultados obtenidos son exportados y procesados, para obtener las reglas de decisión filogenéticas para determinar las posibles secuencias con actividad deseada.

Finalmente el uso de esta metodología, en combinación con las propiedades obtenidas por MetaStudent y los dominios identificados por Pfam, facilitarán una propuesta de secuencias candidatos de auto antígenos más precisa, permitiendo usar estas secuencias para un análisis de auto reactividad empleando el sistema de clasificación construido por estrategias de ensamble, analizando cómo se comportan estas secuencias con el conjunto de anticuerpos disponibles, demostrando la usabilidad de las estrategias propuestas y cómo se crea un efecto sinérgico entre ellas para facilitar diferentes tipos de estudios en el área inmunológica.

4.3. Discusión

El estudio y predicción de proteínas con una actividad deseada en un ámbito con amplio interés, tanto a nivel de investigación como a nivel comercial. EL reconocimiento de estas moléculas posee diversas aplicaciones. En el caso de enfermedades que involucran células propias del organismo, como la leucemia, la identificación de posibles auto antígenos con una interacción deseada con anticuerpos, podría acelerar el proceso de reconocimiento de nuevos antígenos guía en el desarrollo de tratamientos inmunológicos. Información entregada por la función, participación en procesos, ubicación dentro de la célula, y dominios conservados, podrían entregar información para el reconocimiento temprano de posibles nuevas proteínas con una propiedad deseada, dentro de un conjunto general de proteínas. También, información proporcionada a nivel filogenético podría ayudar a reconocer estas moléculas de manera mas acertada.

Entre las secuencias de auto antígenos de leucemia analizadas y las secuencias de antígenos generales recopiladas desde IMDb no se encontraron grandes diferencias en la caracterización realizada. Los términos Gene Ontology observados para función molecular, tanto en secuencias auto antígenas de leucemia y secuencias antigénicas generales, presentan un orden similar en frecuencia. Solo 5 términos de los expuestos en las secuencias auto antígenas no son observados dentro de la base de datos. Una situación similar se observa con respecto a los términos asociados a proceso biológico, en el cual sólo un término no esta presente en los resultados de antígenos de IMDb, y componente celular, donde dos términos se presentan exclusivamente en secuencias de auto antígenos. E el caso de dominios Pfam predichos, 6 de estos dominios son observados solo en secuencias de auto antígenos. No obstante, todas las observaciones exclusivas sobre estas predicciones en el caso de secuencia de auto antígenos de leucemia no son las mas frecuentes entre las realizadas.

La metodología propuesta en este capitulo ha sido implementada anteriormente por el co-tutor, en trabajos relacionados a péptidos con actividades deseadas, obteniendo resultados favorables. No obstante, aún falta evaluar los resultados obtenidos sobre secuencias de auto antígenos. No es posible hablar hasta la fecha con respecto a los posibles resultados esperados a partir de esta metodología, debido a que los alineamientos requeridos para la obtención de distribución de

distancias, y posterior categorización, aún se encuentran en ejecución. Sin embargo, sería destacable que los resultados obtenidos de la aplicación de esta metodología a secuencias auto antigénicas sean igualmente favorables. Mediante esto, sería posible evaluar un conjunto de secuencias putativas, y categorizarlas como posibles auto antígenos relacionados a leucemias. Estas proteínas identificadas como posibles auto antígenos corresponden a un buen punto de inicio en el desarrollo de investigaciones tanto experimentales como in silico. Esto, indudablemente, podría disminuir los costos asociados a la validación experimental de un conjunto extenso de secuencias, y acelerar el proceso de análisis de secuencias auto antígenas frente a anticuerpos de interés.

Capítulo 5

Discusión y conclusiones generales.

El análisis de la interacción auto antígeno - anticuerpo es de gran relevancia, debido a las diversas aplicaciones que esta puede tener. Diversos enfoques pueden ser utilizados en el análisis de la información correspondiente a esta interacción. Acercamientos experimentales y computacionales posibilitan la recopilación de información respecto a estructuras y secuencias, y el análisis de diversas características propias tanto de estas moléculas como de su interacción. Ambos enfoques poseen diversas ventajas, asociadas a su precisión, capacidad de procesamiento, claridad de resultados obtenidos, entre otros. En el caso de las estrategias computacionales, una de sus grandes ventajas es el bajo costo económico asociado a su desarrollo y obtención de resultados. Además, metodologías computacionales permiten procesar y analizar grandes volúmenes de datos simultaneamente. Sin embargo, estas técnicas y herramientas son dependientes de los datos obtenidos y validados mediante procedimientos experimentales. De esta forma, ambas aristas se complementan y hacen posible grandes avances en investigación.

En enfermedades como el cáncer, el análisis de auto antígenos, anticuerpos y la auto reactividad generada mediante su interacción es crucial para comprender los mecanismos de este padecimiento, y nuevas estrategias a abordad para su diagnostico y tratamiento. La similitud entre los auto antígenos presentados por células sanas y aquellos característicos de células cancerígenas impiden que el sistema inmune pueda diferenciarlos, y por lo tanto, evita la presentación de auto reactividad frente a estas células tumorales. No obstante, el análisis de propiedades

únicas en su secuencia, y la interacción entre estos auto antígenos con anticuerpos, puede contribuir en la identificación de características que faculten el desarrollo de moléculas o tratamientos que permitan al sistema inmune superar esta dificultad.

De esta forma, utilizando solamente la información correspondiente a la secuencia de aminoácidos, fue posible diseñar e implementar un sistema de predicción cualitativo del nivel de interacción entre auto antígenos de pacientes de leucemia y cadenas pesadas de anticuerpos. Empleando datos experimentales obtenidos por (Frick, 2009) correspondientes a 45 anticuerpos y 8221 antígenos, anteriormente procesados por (Torres Almonacid, 2020) para la determinación de intensidades de interacción y eliminación de ruido en placas de ProtoArray, fue factible evaluar diversas estrategias de codificación y algoritmos de machine learning para el entrenamiento de sistemas predictivos en base a los conjuntos de datos correspondientes a secuencias de antígenos y anticuerpos proporcionadas.

Las estrategias de codificación de secuencias utilizadas en el entrenamiento de los modelos no presentaron grandes diferencias en cuanto a su desempeño durante la etapa de entrenamiento. No obstante, las codificaciones basadas en estrategias de embedding fueron mayormente seleccionadas de acuerdo a los enfoques estadísticos de identificación de outliers utilizados. Remarcablemente, este tipo de codificación destacó en cuanto a testeo y validación cruzada, manteniendo una baja tasa de sobreajuste en los algoritmos con mejor desempeño. Algo similar se observa con la codificación por One hot, la cual presenta un desempeño destacado si se combina con algoritmos de aprendizaje supervisado basado en técnicas de neural network. Por otra parte, se eliminaron las codificaciones por Ordinal encoder y por frecuencia, debido a que estas estrategias no proporcionan información útil a los algoritmos implementados. Por lo demás, sería posible aplicar técnicas de disminución de dimensionalidad, para trabajar en un espacio mas acotado, pero con varianza maximizada, con los algoritmos e hiper parámetros seleccionados.

Respecto a los diversos algoritmos de aprendizaje supervisado implementados, los algoritmos basados en una distribución estadística, como Bernoulli Naive Bayes y Gaussian Naive Bayes, presentaron la performance mas baja observada. Esto puede atribuirse a la distribución de los datos presentados por los vectores generados en cada codificación, y al número de datos entregados. De manera adicional, estos algoritmos requieren de un elevado costo computacional para calcular los estimadores correctos, y debido a esto, generalmente eran finalizados

antes de terminar satisfactoriamente el proceso debido a las configuraciones de uso y políticas de seguridad del Centro de Computo donde fueron lanzados estos trabajos. Por el contrario, los algoritmos de Random Forest y Neural Networks suelen destacarse por su desempeño. Esto podría indicar que existe un conjunto de features o columnas que presentan valores dentro de rangos establecidos para cada clase, y que podrían facilitar la separación de los ejemplos presentados para cada una de las clases de interacción definidas.

De un total de 413 modelos generados mediante el proceso exploratorio implementado, fueron seleccionados 14 de acuerdo a las estrategias y criterios definidos para ello. 13 de los modelos seleccionados corresponden a modelos basados en algoritmos de random forest, con una codificación basada en embedding.Además, se seleccionó un modelo de neural network, basado en codificación One hot. La tasa de sobreajuste calculada para los modelos seleccionados se encuentra alrededor de 1.16 y 1.6, lo cual es aceptable con respecto a las medidas de desempeño obtenidas por estos. Todos los modelos seleccionados presentan una performance similar, la cual supera el 60 %.

Posteriormente, todos los modelos seleccionados fueron sometidos a la metodología de Leave One Antibody Out. Esta metodología permitió evaluar la robustez de los modelos seleccionados ante la ausencia de las interacciones correspondientes a la cadena pesada de un anticuerpo especifico. Cada modelo fue entrenado con 44 de los anticuerpos, y testeado con el no fue utilizado para el entrenamiento. De esta forma, fue posible observar la dependencia de los modelos a alguno de los anticuerpos y las interacciones con auto antígenos presentadas por éste. De forma general, los modelos seleccionados presentan una performance similar, o mejor, ante la ausencia de las interacciones de un anticuerpo en el set de datos de entrenamiento. No obstante, para algunos anticuerpos la performance obtenida disminuyó. Esto puede deberse a las diferencias en la cantidad de ejemplos categorizados dentro de cada clase observados para el anticuerpo de testeo en cuestión. De esta forma, puede ser que debido a que el modelo es entrenado observando cierta distribución de clases, cometa equivocaciones en los ejemplos con clases des balanceadas en el set de testeo.

A pesar de que la performance observada de forma individual para estos modelos no es destacable, estos pueden ser considerados como robustos. Además, la utilización de codificación por embedding, y algoritmos de random forest, presentó modelos

con la capacidad de realizar predicciones sobre interacciones auto antígeno- cadenas pesadas de anticuerpos leucémicos no observadas en el proceso de entrenamiento, lo cual fue corroborado mediante la implementación de la estrategia Leave One Antibody Out. Igualmente, esta observación se realizó sobre el modelo generado mediante la codificación One hot, y el algoritmo neural network.

Los 14 modelos seleccionados y evaluados fueron utilizados para el desarrollo del modelo ensamblado. Los modelos re-entrenados fueron evaluados con un conjunto de datos aleatorio. Esto llevó a generar un conjunto de testeo con desbalance de clases, en donde el número de ejemplos con clase de interacción Baja supera a lo observado normalmente en la distribución total de interacciones. Fuera de esto, la performance observada para el modelo ensamblado, para cualquiera de las clases, supera a los valores obtenidos para los modelos de manera individual. La implementación de un modelo ensamblado por votación, utilizando los modelos individuales seleccionados, permitió el aumento del performance promedio desde 62 % a 84.3 %.

Los resultados obtenidos empelando la metodología para desarrollo de modelos utilizada permitieron probar que la utilización de secuencias codificadas utilizando una estrategia de natural language processing, especialmente embedding, permiten desarrollar modelos robustos para la predicción de la clase de interacción de auto antígenos y anticuerpos en pacientes con leucemia. Además, se pudo comprobar que un enfoque de modelo ensamblado aumenta la performance de la metodología propuesta. Por otra parte, la selección de modelos entrenados sobre diversas codificaciones posibilitó la integración de diversas metodologías de representación de datos, lo cual puede reflejar el aumento en la performance observada en el modelo ensamblado, con respecto a lo obtenido a partir de modelos únicos.

Aun quedan diversas aristas en explorar con respecto a esta metodología, y que podrían aumentar la performance de los modelos, o disminuir el tiempo y recursos computacionales requeridos para su entrenamiento y predicción. Integrar mayor número de puntos de vista al set de datos podría entregar información mas detallada a los modelos para separar claramente las clases de interacción definidas. Por otra parte, la disminución de la dimensionalidad podría ser una estrategia interesante sobre los vectores generados. Por sobre esto, la evaluación de los límites de similitud de secuencia aceptados para los modelos es aún un ámbito a determinar. Saber hasta que punto del espacio de secuencias utilizado puede

soportar el modelo ensamblado, y los modelos individuales, es un análisis necesario para evaluar su efectividad real ante nuevos ejemplos de secuencias auto antígenas y cadenas pesadas de anticuerpos.

La recopilación de información para la prueba tanto del modelo ensamblado como de nuevas herramientas o metodologías, dependiendo del enfoque adoptado, puede presentar diversas dificultades. En el caso de estrategias experimentales, es necesario recopilar muestras y procesarlas en base a los datos de interés requeridos. Esto puede ser costoso en recursos económicos, así como en tiempo. Aun cuando se cuenta con herramientas que pueden producir grandes volúmenes de datos simultáneamente, como pueden ser las técnicas de ProtoArray, estos pueden requerir procesamiento externo mediante otras herramientas, incrementando de esta forma el tiempo y los recursos necesarios para su filtrado y disposición. Por otra parte, estrategias computacionales poseen dificultades asociadas al volumen de datos real disponible, la dispersión de estos en diversas plataformas, errores en la descripción de diversas entradas,entre otros. De forma similar a lo observado experimentalmente,debido a la dispersión y diferencias en los datos presentados por cada fuente de información disponible, es necesario utilizar diversas herramientas para procesar la información y obtener los datos de interés en el formato requerido.

El aumento en la conectividad mediada por internet, y el desarrollo de infraestructura computacional adecuada, ha respaldado la implementación de diversas plataformas para el almacenamiento de información, y la disposición de herramientas de análisis. Es por esto que, en la última década, el número de bases de datos para diversos tipos de moléculas ha aumentado considerablemente. Esta gran ventaja que dispone la información cerca de cualquier potencial usuario con acceso a internet, trae además diversas dificultades.

La dispersión de la data disponible en diversas plataformas, algunas no actualizadas en años, dificulta la recopilación de toda la información disponible a la fecha. Es necesario realizar una revisión exhaustiva de sitios web y artículos relacionados para agrupar el mayor número de datos de interés. En algunos casos, bases de datos mencionadas en diversos artículos no presentaron paginas web accesibles. Por otra parte, la mayoría de las fuentes de información revisadas contaban con errores en la cantidad de información contenida realmente. La redundancia en las secuencias entregadas por las bases de datos es algo común, lo cual puede indicar una falta de control o filtros adecuados en la información proporcionada.

Similarmente, en bases de datos ampliamente conocidas como AntibodyPedia o Antibody Register, la cantidad de datos señalados por sus sistemas corresponde en su mayoría a datos privados, por lo cual sus secuencias no son accesibles.

La falta de herramientas presentadas por las páginas web para la descarga de sus datos, o para su descarga como conjunto, es una situación observada en sistemas publicados recientemente, como es el caso de ABCD. De forma similar, en bases de datos como SACS, SabDab, dbPepNeo y IEDB, la información entregada hace relación a códigos pertenecientes a otros recursos, como NCBI o PDB resource, por lo cual, es necesario acceder a estas plataformas para obtener las secuencias deseadas. Este problema puede ser sorteado mediante la búsqueda de estas secuencias de forma manual, o utilizando herramientas computacionales que faciliten la realización de consultas de forma automática. Así, la recopilación de esta información obliga al usuario a poseer cierto manejo en herramientas computacionales para el procesamiento automático de un gran volumen de datos.

De acuerdo con lo visualizado en el trabajo realizado, además de todas las problemáticas mencionadas anteriormente, una variada gama de formatos fue presentada por los sistemas analizados, forzando a la utilización de scripts de procesamiento específicos. Dependiendo de la información entregada por cada archivo, su disposición y las palabras claves que permitieren acceder a la información de interés, fue necesario implementar diversas estrategias, e integrar distintas herramientas. Esto significa una cantidad considerable de tiempo en solo reunir los datos requeridos para un propósito deseado, sin considerar el tiempo requerido en el desarrollo de estrategias de procesamiento y transformación a formatos deseados.

Todas estas observaciones llevaron al desarrollo de la base de datos Immune Molecules Database (IMDb). Esta cuenta con el mayor número de registros públicos para secuencias de anticuerpos humanos, antígenos, epítopes lineales y estructurales, y complejos de interacción antígeno-anticuerpo. Mediante la extracción y transformación de la información proporcionada por cada base de datos analizada se consiguió reunir 9,596 secuencias de anticuerpos, 145,795 secuencias de antígenos, y 1,299,153 epítopes lineales y estructurales. IMDb permitiría a los usuarios acceder al mayor número de datos disponibles a la fecha en un solo sitio, evitando la laboriosa tarea de recopilación y procesamiento descrita anteriormente.

Además, con apoyo del equipo de investigación de CeBiB, actualmente se encuentra en desarrollo un sistema web que no solo permita el acceso a la base de datos creada, sino que también, entregue un conjunto de herramientas de interés para el análisis de moléculas inmunológicas. Los servicios propuestos fueron seleccionados en base a la frecuencia con la cual fueron observados en las bases de datos analizadas, y a posibles nuevos análisis de interés biológico. De esta forma, sus usuarios podrían encontrar en IMDb un gran recurso tanto en información como en usabilidad. Herramientas de gran interés como WhatIf se encuentran integrados al sistema propuesto, permitiendo realizar estudios estructurales en base a cristales o modelos en formato PDB. La evaluación de distancias inter moleculares entre cadenas puede cumplir como punto inicial para proponer epitopes estructurales, así como regiones parátopes en un anticuerpo. Esta herramienta no se observó en ningún otro sistema, entregando una ventaja más a IMDb por sobre otros sistemas disponibles.

Dentro del trabajo postulado, se plantea el reconocimiento de posibles secuencias auto antígenas relacionadas a leucemia entre la información recopilada en la base de datos desarrollada en este proyecto de memoria. La utilización de esta metodología planteada podría permitir probar los límites de los modelos generados en el capitulo 1 con respecto a la variabilidad de las secuencias de auto antígenos. Además, no se encontraron metodologías para predecir posibles auto antígenos relacionados a leucemia, por lo cual, corresponde a un área aun sin explorar que podría aportar mayor número de información al trabajo ya realizado.

Es posible que un conjunto de características, así como relaciones filogenéticas, permitan describir propiedades inherentes a estas secuencias, y que las distingan de otros grupos de antígenos. La información con respecto a dominios y regiones conservadas que contribuyan en su reconocimiento, así como funciones moleculares relacionadas o ubicación en una célula, podrían aportar la información necesaria para filtrar secuencias con una función deseada desde un grupo general de moléculas.

Es por esto que se utilizaron las herramientas metastudent y Pfam para predecir, a partir de la secuencia de aminoácidos, términos Gene Ontology y dominios Pfam respectivamente. Esto permitió formar grupos de características observadas dentro del set de datos experimental de secuencias auto antígenas, con mayor o menor frecuencia. Estos grupos fueron utilizados para filtrar las secuencias almacenadas

en IMDb, las cuales fueron caracterizadas bajo diversos puntos de vista, incluyendo la predicción de términos GO y dominios Pfam.

El alineamiento de las secuencias experimentales permitiría observar la distribución de distancias entre estas secuencias, y establecer límites a partir de los cuales una secuencia puede o no puede ser considerada como auto antígeno. Nuevas secuencias que pasen el filtro con respecto a los grupos observados en datos experimentales pueden ser posteriormente alineadas contra las secuencias de auto antígenos, a partir del cual se obtendrá un vector de distancias. Luego, cada distancia observada podría corresponder a una predicción con respecto a si corresponde o no corresponde a una secuencia auto antígena, respecto a los límites establecidos anteriormente. Siguiendo una metodología de votación, la predicción final se basará en la clase con mayor frecuencia.

Hasta el momento se han observado grupos variados dentro de los datos experimentales, con diversos dominios predichos con mayor frecuencia, como los observados relacionados a quinasas. Por otra parte, las funciones moleculares predichas parecen relacionarse con unión a ligandos o sustratos, y con menor frecuencia, a la unión a tri fosfatos de ribonucleósidos de purinas. Esto hace sentido considerando que las quinasas modifican sustratos mediante la fosforilación. En cuanto a proceso biológico, el observado con mayor frecuencia es de proceso metabólico, el cual, nuevamente, se relaciona con lo señalado anteriormente. Sin embargo, en cuanto a ubicación celular, esta no es posible de precisar con certeza. Esto se debe, en parte, a la presencia de términos obsoletos dentro de las predicciones realizadas por metastudent.

Para las secuencias de la base de datos se observaron resultados similares, lo cual puede indicar que los organismos poseen preferencia por moléculas con las características descritas anteriormente para ser reconocidas como antígenos. Por lo demás, a pesar de que la frecuencia de los términos en la base de datos es similar a lo observado en las secuencias auto antígenas, el número de secuencias recopiladas por el filtro aplicado fue de alrededor de 6,000. Es posible que las diferencias y propiedades que definen su caracterización se encuentren a nivel de secuencia, la cual se puede definir por residuos conservados o distancias evolutivas. Se espera observar esto en resultados posteriores.

Finalmente, las metodologías planteadas y herramientas desarrolladas a lo largo de

este trabajo presentan un avance considerable en la exploración de metodologías para el estudio de secuencias del sistema inmune. Hasta el momento, el trabajo realizado con la exploración de modelos y proposición de un modelo ensamblado es el único realizado para secuencias de antígeno y anticuerpos leucémicos, que busca predecir la clase de interacción en base a la información de sus secuencias, y empleando diversas aristas de codificación. Además, el sistema web IMDb ofrece no solo acceso al mayor número de registros públicos con respecto a antígenos, anticuerpos, epítopes e interacción, sino que también, herramientas de análisis que pueden apoyar investigaciones inmunológicas futuras. Además, la implementación de la metodología de predicción de auto antígenos planteada, de presentar resultados favorables como aquellos observados por el co-tutor dentro de la postulación de péptidos con actividad deseada, podría contribuir con la identificación de nuevas secuencias de auto antígenos leucémicos no reconocidas anteriormente. Estudios de auto reactividad no llevados a cabo previamente pueden ser postulados empleando estas posibles secuencias en combinación con el modelo ensamblado diseñado, prediciendo clases de interacción para estas putativas secuencias de auto antígenos con cadenas pesadas de anticuerpos. Este planteamiento refleja una correlación entre ambas estrategias, y que posibilitan el desarrollo de diversos análisis con respecto al sistema inmune, específicamente, estudios referentes a auto reactividad en pacientes con leucemia, y las secuencias de auto antígenos y anticuerpos involucradas. Sin duda, estos avances podría colaborar en el planteamiento de nuevas aristas sobre esta enfermedad, y la postulación de nuevos tratamientos inmunológicos.

Bibliografía

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. (2016). Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pages 265–283.

Abbas, A. K., Lichtman, A. H., and Pillai, S. (2019). *Basic Immunology E-Book: Functions and Disorders of the Immune System*. Elsevier Health Sciences.

Abdi, H. and Williams, L. J. (2010). Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459.

Adlersberg, J. B. (1976). The immunoglobulin hinge (interdomain) region. *Ricerca in clinica e in laboratorio*, 6(3):191.

Adolf-Bryfogle, J., Kalyuzhniy, O., Kubitz, M., Weitzner, B. D., Hu, X., Adachi, Y., Schief, W. R., and Dunbrack Jr, R. L. (2018). Rosettaantibodydesign (rabd): A general framework for computational antibody design. *PLoS computational biology*, 14(4):e1006112.

Ahmed, A., Shervashidze, N., Narayanamurthy, S., Josifovski, V., and Smola, A. J. (2013). Distributed large-scale natural graph factorization. In *Proceedings of the 22nd international conference on World Wide Web*, pages 37–48.

Akbar, R., Jeliazkov, J. R., Robert, P. A., Snapkov, I., Pavlović, M., Slabodkin, A., Weber, C. R., Safonova, Y., Sandve, G. K., and Greiff, V. (2019). A finite vocabulary of antibody-antigen interaction enables predictability of paratope-epitope binding. *bioRxiv*, page 759498.

Alain, G. and Bengio, Y. (2014). What regularized auto-encoders learn from t he data-generating distribution. *The Journal of Machine Learning Research*, 15(1):3563–3593.

Alam, J., Jantan, I., and Bukhari, S. N. A. (2017). Rheumatoid arthritis: recent advances on its etiology, role of cytokines and pharmacotherapy. *Biomedicine & Pharmacotherapy*, 92:615–633.

Allahyari, M., Pouriyeh, S., Assefi, M., Safaei, S., Trippe, E. D., Gutierrez, J. B., and Kochut, K. (2017). A brief survey of text mining: Classification, clustering and extraction techniques. *arXiv preprint arXiv:1707.02919*.

Allcorn, L. C. and Martin, A. C. (2002). Sacs—self-maintaining database of antibody crystal structure information. *Bioinformatics*, 18(1):175–181.

Alley, E. C., Khimulya, G., Biswas, S., AlQuraishi, M., and Church, G. M. (2019). Unified rational protein engineering with sequence-only deep representation learning. *bioRxiv*.

Almeida, L. G., Sakabe, N. J., Deoliveira, A. R., Silva, M. C. C., Mundstein, A. S., Cohen, T., Chen, Y.-T., Chua, R., Gurung, S., Gnjatic, S., et al. (2009). Ctdatabase: a knowledge-base of high-throughput and curated data on cancer-testis antigens. *Nucleic acids research*, 37(suppl_1):D816–D819.

Altman, N. and Krzywinski, M. (2018). The curse (s) of dimensionality. *Nat Methods*, 15:399–400.

Ambrosetti, F., Jiménez-García, B., Roel-Touris, J., and Bonvin, A. M. (2020). Modeling antibody-antigen complexes by information-driven docking. *Structure*, 28(1):119–129.

Amrane, M., Oukid, S., Gagaoua, I., and Ensari, T. (2018). Breast cancer classification using machine learning. In *2018 Electric Electronics, Computer Science, Biomedical Engineerings' Meeting (EBBT)*, pages 1–4. IEEE.

Ansari, H. R., Flower, D. R., and Raghava, G. (2010). Antigendb: an immunoinformatics database of pathogen antigens. *Nucleic acids research*, 38(suppl_1):D847–D853.

Ansari, H. R. and Raghava, G. P. (2010). Identification of conformational b-cell epitopes in an antigen from its primary sequence. *Immunome research*, 6(1):1–9.

Arkan, E., Saber, R., Karimi, Z., and Shamsipur, M. (2015). A novel antibody–antigen based impedimetric immunosensor for low level detection of her2 in serum samples of breast cancer patients via modification of a gold nanoparticles decorated multiwall carbon nanotube-ionic liquid electrode. *Analytica chimica acta*, 874:66–74.

Artimo, P., Jonnalagedda, M., Arnold, K., Baratin, D., Csardi, G., De Castro, E., Duvaud, S., Flegel, V., Fortier, A., Gasteiger, E., et al. (2012). Expasy: Sib bioinformatics resource portal. *Nucleic acids research*, 40(W1):W597–W603.

Asai, N., Shimizu, T., Shingubara, S., and Ito, T. (2018). Fabrication of highly sensitive qcm sensor using aao nanoholes and its application in biosensing. *Sensors and Actuators B: Chemical*, 276:534–539.

Asgari, E., McHardy, A. C., and Mofrad, M. R. (2019). Probabilistic variable-length segmentation of protein sequences for discriminative motif discovery (dimotif) and sequence embedding (protvecx). *Scientific reports*, 9(1):1–16.

Atlassian (2011). Trello. https://trello.com/es.

Ayodele, T. O. (2010). Types of machine learning algorithms. *New advances in machine learning*, pages 19–48.

Balmaseda, A., Stettler, K., Medialdea-Carrera, R., Collado, D., Jin, X., Zambrana, J. V., Jaconi, S., Cameroni, E., Saborio, S., Rovida, F., et al. (2017). Antibody-based assay discriminates zika virus infection from other flaviviruses. *Proceedings of the National Academy of Sciences*, 114(31):8384–8389.

Banchereau, J. and Palucka, K. (2018). Cancer vaccines on the move. *Nature reviews Clinical oncology*, 15(1):9–10.

Banerjee, A., Chitnis, U., Jadhav, S., Bhawalkar, J., and Chaudhury, S. (2009). Hypothesis testing, type i and type ii errors. *Industrial psychiatry journal*, 18(2):127.

Bazzoli, A., Vance, D. J., Rudolph, M. J., Rong, Y., Angalakurthi, S. K., Toth IV, R. T., Middaugh, C. R., Volkin, D. B., Weis, D. D., Karanicolas, J., et al. (2017). Using homology modeling to interrogate binding affinity in neutralization of ricin toxin by a family of single domain antibodies. *Proteins: Structure, Function, and Bioinformatics*, 85(11):1994–2008.

Belkina, A. C., Ciccolella, C. O., Anno, R., Spidlen, J., Halpert, R., and Snyder-Cappione, J. (2018). Automated optimal parameters for t-distributed stochastic neighbor embedding improve visualization and allow analysis of large datasets. *bioRxiv*, page 451690.

Bellman, R. E. (2015). *Adaptive control processes: a guided tour*. Princeton university press.

Berkhin, P. (2006). A survey of clustering data mining techniques. In *Grouping multidimensional data*, pages 25–71. Springer.

Bhatia, N. et al. (2010). Survey of nearest neighbor techniques. *arXiv preprint arXiv:1007.0085*.

Bird, S., Klein, E., and Loper, E. (2009). *Natural language processing with Python: analyzing text with the natural language toolkit*. .ºReilly Media, Inc.".

Björling, E. and Uhlén, M. (2008). Antibodypedia, a portal for sharing antibody and antigen validation data. *Molecular & Cellular Proteomics*, 7(10):2028–2037.

Brassington, G. (2017). Mean absolute error and root mean square error: which is the better metric for assessing model performance? In *EGU General Assembly Conference Abstracts*, volume 19, page 3574.

Brownlee, J. (2017). Why one-hot encode data in machine learning. *Machine Learning Mastery*.

Bruce Alberts, Alexander Johnson, J. L. M. R.-K. R. and Walter, P. (2002). *Molecular Biology of the Cell*. Garland Science, fourth edition edition.

Buder-Bakhaya, K. and Hassel, J. C. (2018). Biomarkers for clinical benefit of immune checkpoint inhibitor treatment—a review from the melanoma perspective and beyond. *Frontiers in immunology*, 9:1474.

Bush, D. B. and Knotts, T. A. (2017). Probing the effects of surface hydrophobicity and tether orientation on antibody-antigen binding. *The Journal of Chemical Physics*, 146(15):155103.

Butler, C. L., Hickey, M. J., Jiang, N., Zheng, Y., Gjertson, D., Zhang, Q., Rao, P., Fishbein, G. A., Cadeiras, M., Deng, M. C., et al. (2020). Discovery of non-hla antibodies associated with cardiac allograft rejection and development and validation of a non-hla antigen multiplex panel: From bench to bedside. *American Journal of Transplantation*.

Buzzetti, R., Zampetti, S., and Maddaloni, E. (2017). Adult-onset autoimmune diabetes: current knowledge and implications for management. *Nature Reviews Endocrinology*, 13(11):674.

Camacho, D. M., Collins, K. M., Powers, R. K., Costello, J. C., and Collins, J. J. (2018). Next-generation machine learning for biological networks. *Cell*, 173(7):1581–1592.

Camacho-Collados, J. and Pilehvar, M. T. (2018). From word to sense embeddings: A survey on vector representations of meaning. *Journal of Artificial Intelligence Research*, 63:743–788.

Cao, Y., Shi, H., Le, T., Tang, R., and Xie, Y. (2019). Development a monoclonal antibody-based enzyme-linked immunosorbent assay for screening pyrimethanil in fruits and vegetables. *Food and Agricultural Immunology*, 30(1):548–563.

Castellazzi, G., Cuzzoni, M. G., Cotta Ramusino, M., Martinelli, D., Denaro, F., Ricciardi, A., Vitali, P., Anzalone, N., Bernini, S., Palesi, F., et al. (2020). A machine learning approach for the differential diagnosis of alzheimer and vascular dementia fed by mri selected features. *Frontiers in neuroinformatics*, 14:25.

Castelo-Branco, C. and Soveral, I. (2014). The immune system and aging: a review. *Gynecological Endocrinology*, 30(1):16–22.

Chesson, C. B. and Zloza, A. (2017). Nanoparticles: augmenting tumor antigen presentation for vaccine and immunotherapy treatments of cancer. *Nanomedicine*, 12(23):2693–2706.

Chiorazzi, N., Chen, S.-S., and Rai, K. R. (2021). Chronic lymphocytic leukemia. *Cold Spring Harbor perspectives in medicine*, 11(2):a035220.

Chlon, L. (2017). *Machine Learning Methods for Cancer Immunology*. PhD thesis, University of Cambridge.

Chollet, F. et al. (2015). Keras. https://github.com/fchollet/keras.

Christophersen, A., Lund, E. G., Snir, O., Solà, E., Kanduri, C., Dahal-Koirala, S., Zühlke, S., Molberg, Ø., Utz, P. J., Rohani-Pichavant, M., et al. (2019). Distinct phenotype of cd4+ t cells driving celiac disease identified in multiple autoimmune conditions. *Nature medicine*, 25(5):734–737.

Cios, K. J., Pedrycz, W., Swiniarski, R. W., and Kurgan, L. A. (2007). *Data mining: a knowledge discovery approach*. Springer Science & Business Media.

Cloutier, T., Sudrik, C., Mody, N., Sathish, H. A., and Trout, B. L. (2019). Molecular computations of preferential interaction coefficients of igg1 monoclonal antibodies with sorbitol, sucrose, and trehalose and the impact of these excipients on aggregation and viscosity. *Molecular pharmaceutics*, 16(8):3657–3664.

Collatz, M., Mock, F., Hölzer, M., Barth, E., Sachse, K., and Marz, M. (2020). Epidope: A deep neural network for linear b-cell epitope prediction. *bioRxiv*.

Connect, E. (2020). Tipos de inmunidad adaptativa, la respuesta 'mutante' contra la infección. urlhttps://www.elsevier.com/es-es/connect/medicina/edu-tipos-de-inmunidad-adaptativa. Accedido 07-05-2020.

Consortium, G. O. (2021). The gene ontology resource: enriching a gold mine. *Nucleic Acids Research*, 49(D1):D325–D334.

Cota, A. M. and Midwinter, M. J. (2015). The immune system. *Anaesthesia Intensive Care Medicine*, 16(7):353–355. Intensive Care / Transplantation.

Crowe Jr, J. E. (2017). Principles of broad and potent antiviral human antibodies: insights for vaccine design. *Cell host & microbe*, 22(2):193–206.

Dall'Acqua, W., Goldman, E. R., Lin, W., Teng, C., Tsuchiya, D., Li, H., Ysern, X., Braden, B. C., Li, Y., Smith-Gill, S. J., et al. (1998). A mutational analysis of binding interactions in an antigen- antibody protein- protein complex. *Biochemistry*, 37(22):7981–7991.

David Medina-Ortiz, Yasna Barrera Saavedra, G. C.-M. F. R. C. Q. J. T.-A. R. U.-P. A. O.-N. M. N. (2021). Immune molecules databases, imdb, a user-friendly web application tool and a comprehensive database to support immunological research. *Work in Progress*.

de Vries, S. J. and Bonvin, A. M. (2011). Cport: a consensus interface predictor and its performance in prediction-driven docking with haddock. *PloS one*, 6(3):e17695.

Dhar, T. M. and Dyckman, A. (2017). 5.12 - evolution of small-molecule immunology research—changes since cmc ii. In Chackalamannil, S., Rotella, D., and Ward, S. E., editors, *Comprehensive Medicinal Chemistry III*, pages 395–419. Elsevier, Oxford.

Dhiman, N., Ovsyannikova, I. G., Vierkant, R. A., Ryan, J. E., Pankratz, V. S., Jacobson, R. M., and Poland, G. A. (2008). Associations between snps in toll-like receptors and related intracellular signaling molecules and immune responses to measles vaccine: preliminary results. *Vaccine*, 26(14):1731–1736.

Doersch, C. (2016). Tutorial on variational autoencoders. *arXiv preprint arXiv:1606.05908*.

Döhner, H., Weisdorf, D. J., and Bloomfield, C. D. (2015). Acute myeloid leukemia. *New England Journal of Medicine*, 373(12):1136–1152.

Dowsland, K. A. and Thompson, J. (2012). Simulated annealing. *Handbook of natural computing*, pages 1623–1655.

Dunbar, J., Krawczyk, K., Leem, J., Baker, T., Fuchs, A., Georges, G., Shi, J., and Deane, C. M. (2014). Sabdab: the structural antibody database. *Nucleic acids research*, 42(D1):D1140–D1146.

Duquesnoy, R., Marrari, M., Marroquim, M., Borges, A., da Mata Sousa, L., Socorro, A., and do Monte, S. (2019). Second update of the international registry of hla epitopes. i. the hla-abc epitope database. *Human immunology*, 80(2):103–106.

Durai, M. and Moudgil, K. (2007). Autotolerance*. In Fink, G., editor, *Encyclopedia of Stress (Second Edition)*, pages 290–297. Academic Press, New York, second edition edition.

El-Gebali, S., Mistry, J., Bateman, A., Eddy, S. R., Luciani, A., Potter, S. C., Qureshi, M., Richardson, L. J., Salazar, G. A., Smart, A., et al. (2019). The pfam protein families database in 2019. *Nucleic acids research*, 47(D1):D427–D432.

Endo, T., Yamamura, S., Nagatani, N., Morita, Y., Takamura, Y., and Tamiya, E. (2005). Localized surface plasmon resonance based optical biosensor using surface modified nanoparticle layer for label-free monitoring of antigen–antibody reaction. *Science and Technology of Advanced Materials*, 6(5):491.

Eroshkin, A. M., LeBlanc, A., Weekes, D., Post, K., Li, Z., Rajput, A., Butera, S. T., Burton, D. R., and Godzik, A. (2014). bnaber: database of broadly neutralizing hiv antibodies. *Nucleic acids research*, 42(D1):D1133–D1139.

Faissol, D. (2019). Creating feature representations of antibody-antigen complexes for fast binding prediction with machine learning. Technical report, Lawrence Livermore National Lab.(LLNL), Livermore, CA (United States).

Faraji, F., Tajik, N., Behdani, M., Shokrgozar, M. A., Zarnani, A. H., Shahhosseini, F., and Habibi-Anbouhi, M. (2018). Development and characterization of a camelid single-domain antibody directed to human cd22 biomarker. *Biotechnology and applied biochemistry*, 65(5):718–725.

Feng, N., Simanski, S., Islam, K., Hynan, L. S., Kodadek, T., and German, D. C. (2018). Antibody biomarker for de novo parkinson disease: attempted validation. *npj Parkinson's Disease*, 4(1):1–5.

Ferdous, S. and Martin, A. C. (2018). Abdb: antibody structure database—a database of pdb-derived antibody structures. *Database*, 2018.

Fernandez-Quintero, M. L., Hoerschinger, V. J., Lamp, L. M., Bujotzek, A., Georges, G., and Liedl, K. R. (2020). V h -v l interdomain dynamics

observed by computer simulations and nmr. *Proteins: Structure, Function, and Bioinformatics*.

Figgett, W. A., Monaghan, K., Ng, M., Alhamdoosh, M., Maraskovsky, E., Wilson, N. J., Hoi, A. Y., Morand, E. F., and Mackay, F. (2019). Machine learning applied to whole-blood rna-sequencing data uncovers distinct subsets of patients with systemic lupus erythematosus. *Clinical & translational immunology*, 8(12):e01093.

Firestein, G. S. (2018). Pathogenesis of rheumatoid arthritis: the intersection of genetics and epigenetics. *Transactions of the American Clinical and Climatological Association*, 129:171.

Fontanella, S., Frainay, C., Murray, C. S., Simpson, A., and Custovic, A. (2018). Machine learning to identify pairwise interactions between specific ige antibodies and their association with asthma: A cross-sectional analysis within a population-based birth cohort. *PLoS medicine*, 15(11):e1002691.

Foundation, P. S. (2008). Python 3. https://www.python.org/download/releases/3.0/.

Fousteri, G., Rodrigues, E. M., Giamporcaro, G. M., and Falcone, M. (2021). A machine learning approach to predict response to immunotherapy in type 1 diabetes. *Cellular & Molecular Immunology*, 18(3):515–517.

Fraietta, J. A., Lacey, S. F., Orlando, E. J., Pruteanu-Malinici, I., Gohil, M., Lundh, S., Boesteanu, A. C., Wang, Y., O'Connor, R. S., Hwang, W.-T., et al. (2018). Determinants of response and resistance to cd19 chimeric antigen receptor (car) t cell therapy of chronic lymphocytic leukemia. *Nature medicine*, 24(5):563–571.

Frick, M. (2009). *Untersuchung des Antigenbindungsverhaltens der Antigen-Rezeptoren von B-Zell-Lymphomen mittels Phage Display und Proteinmicroarrays*. PhD thesis, Albert-Ludwigs-University Freiburg i. Br. Abteilung Innere Medizin I (Hämatologie und Onkologie).

Fritz, J. M. and Lenardo, M. J. (2019). Development of immune checkpoint therapy for cancer. *Journal of Experimental Medicine*, 216(6):1244–1254.

Fujioka, M. and Iwai, H. (1997). Statistical pattern analysis and its procedure. *Bulletin of Labour Statistics*, pages 3–4.

Fukumizu, K., Song, L., and Gretton, A. (2011). Kernel bayes' rule. In *Advances in neural information processing systems*, pages 1737–1745.

García, F. C. (2018). *Data acquisition techniques based on frequency-encoding applied to capacitive mems microphones*. PhD thesis, Universidad Carlos III de Madrid.

Gattorno, M. and Martini, A. (2011). Chapter 3 - immunology and rheumatic diseases. In Cassidy, J. T., Laxer, R. M., Petty, R. E., and Lindsley, C. B.,

editors, *Textbook of Pediatric Rheumatology (Sixth Edition)*, pages 16–52. W.B. Saunders, Philadelphia, sixth edition edition.

Gaudelet, T., Day, B., Jamasb, A. R., Soman, J., Regep, C., Liu, G., Hayter, J. B., Vickers, R., Roberts, C., Tang, J., et al. (2020). Utilising graph machine learning within drug discovery and development. *arXiv preprint arXiv:2012.05716*.

Gaur, P. (2012). Neural networks in data mining. *International Journal of Electronics and Computer Science Engineering*.

Giudicelli, V., Chaume, D., and Lefranc, M.-P. (2005). Imgt/gene-db: a comprehensive database for human and mouse immunoglobulin and t cell receptor genes. *Nucleic acids research*, 33(suppl_1):D256–D261.

Giudicelli, V., Duroux, P., Ginestoux, C., Folch, G., Jabado-Michaloud, J., Chaume, D., and Lefranc, M.-P. (2006). Imgt/ligm-db, the imgt® comprehensive database of immunoglobulin and t cell receptor nucleotide sequences. *Nucleic acids research*, 34(suppl_1):D781–D784.

Gök, M., Koçal, O. H., and Genç, S. (2016). Prediction of disordered regions in proteins using physicochemical properties of amino acids. *International Journal of Peptide Research and Therapeutics*, 22(1):31–36.

Goldberg, Y. and Levy, O. (2014). word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *arXiv preprint arXiv:1402.3722*.

Goyal, P. and Ferrara, E. (2018). Graph embedding techniques, applications, and performance: A survey. *Knowledge-Based Systems*, 151:78–94.

Guan, M., Cho, S., Petro, R., Zhang, W., Pasche, B., and Topaloglu, U. (2019). Natural language processing and recurrent network models for identifying genomic mutation-associated cancer treatment change from patient progress notes. *JAMIA Open*, 2(1):139–149.

Hamp, T., Kassner, R., Seemayer, S., Vicedo, E., Schaefer, C., Achten, D., Auer, F., Boehm, A., Braun, T., Hecht, M., et al. (2013). Homology-based inference sets the bar high for protein function prediction. In *BMC bioinformatics*, volume 14, page S7. Springer.

Han, J., Pei, J., and Kamber, M. (2011). *Data mining: concepts and techniques*. Elsevier.

Haque, E., Banik, U., Monwar, T., Anthony, L., and Adhikary, A. K. (2018). Worldwide increased prevalence of human adenovirus type 3 (hadv-3) respiratory infections is well correlated with heterogeneous hypervariable regions (hvrs) of hexon. *PloS one*, 13(3):e0194516.

Harris, K. E., Aldred, S. F., Davison, L. M., Ogana, H. A. N., Boudreau, A., Brüggemann, M., Osborn, M., Ma, B., Buelow, B., Clarke, S. C., et al. (2018). Sequence-based discovery demonstrates that fixed light chain human transgenic

rats produce a diverse repertoire of antigen-specific antibodies. *Frontiers in immunology*, 9:889.

Hauke, J. and Kossowski, T. (2011). Comparison of values of pearson's and spearman's correlation coefficients on the same sets of data. *Quaestiones geographicae*, 30(2):87–93.

Heagerty, P. J. and Zheng, Y. (2005). Survival model predictive accuracy and roc curves. *Biometrics*, 61(1):92–105.

Hecht-Nielsen, R. (1992). Theory of the backpropagation neural network. In *Neural networks for perception*, pages 65–93. Elsevier.

Heiss, K., Heidepriem, J., Fischer, N., Weber, L. K., Dahlke, C., Jaenisch, T., and Loeffler, F. F. (2020). Rapid response to pandemic threats: immunogenic epitope detection of pandemic pathogens for diagnostics and vaccine development using peptide microarrays. *Journal of proteome research*, 19(11):4339–4354.

Herold, N. C. and Mitra, P. (2020). Immunophenotyping. *StatPearls [Internet]*.

Herzog, S., Tetzlaff, C., and Wörgötter, F. (2020). Evolving artificial neural networks with feedback. *Neural Networks*, 123:153–162.

Hristea, F. T. (2011). *Statistical Natural Language Processing*, pages 1452–1453. Springer Berlin Heidelberg, Berlin, Heidelberg.

Huang, J. and Honda, W. (2006). Ced: a conformational epitope database. *BMC immunology*, 7(1):7.

Inc., P. T. (2015). Collaborative data science.

Ivakhnenko, A., Ivakhnenko, G., and Muller, J. (1994). Self-organization of neural networks with active neurons. *Pattern Recognition and Image Analysis*, 4(2):185–196.

Jabbar, B., Rafique, S., Salo-Ahen, O. M., Ali, A., Munir, M., Idrees, M., Mirza, M. U., Vanmeert, M., Shah, S. Z., Jabbar, I., et al. (2018). Antigenic peptide prediction from e6 and e7 oncoproteins of hpv types 16 and 18 for therapeutic vaccine design using immunoinformatics and md simulation analysis. *Frontiers in immunology*, 9:3000.

Jabbour, E. and Kantarjian, H. (2018). Chronic myeloid leukemia: 2018 update on diagnosis, therapy and monitoring. *American journal of hematology*, 93(3):442–459.

Jain, M., Kamal, N., and Batra, S. K. (2007). Engineering antibodies for clinical applications. *Trends in biotechnology*, 25(7):307–316.

Jespersen, M. C., Mahajan, S., Peters, B., Nielsen, M., and Marcatili, P. (2019). Antibody specific b-cell epitope predictions: Leveraging information from antibody-antigen protein complexes. *Frontiers in Immunology*, 10:298.

Johnson, G. and Wu, T. T. (2001). Kabat database and its applications: future directions. *Nucleic Acids Research*, 29(1):205–206.

Johnson, M. J., Bland, J. M., Davidson, P. M., Newton, P. J., Oxberry, S. G., Abernethy, A. P., and Currow, D. C. (2014). The relationship between two performance scales: New york heart association classification and karnofsky performance status scale. *Journal of pain and symptom management*, 47(3):652–658.

Kaas, Q., Ruiz, M., and Lefranc, M.-P. (2004). Imgt/3dstructure-db and imgt/structuralquery, a database and a tool for immunoglobulin, t cell receptor and mhc structural data. *Nucleic acids research*, 32(suppl_1):D208–D210.

Kabat, E. A., Te Wu, T., Perry, H. M., Foeller, C., and Gottesman, K. S. (1992). *Sequences of proteins of immunological interest*. DIANE publishing.

Kamat, V. and Rafique, A. (2017). Designing binding kinetic assay on the bio-layer interferometry (bli) biosensor to characterize antibody-antigen interactions. *Analytical biochemistry*, 536:16–31.

Kaur, H. and Kumari, V. (2020). Predictive modelling and analytics for diabetes using a machine learning approach. *Applied computing and informatics*.

Kausaite, A., van Dijk, M., Castrop, J., Ramanaviciene, A., Baltrus, J. P., Acaite, J., and Ramanavicius, A. (2007). Surface plasmon resonance label-free monitoring of antibody antigen interactions in real time. *Biochemistry and Molecular Biology Education*, 35(1):57–63.

Kawashima, S. and Kanehisa, M. (2000). Aaindex: amino acid index database. *Nucleic acids research*, 28(1):374–374.

Kelley, T. W. and Patel, J. L. (2018). 8 - genetic aspects of hematopoietic malignancies. In Rifai, N., Horvath, A. R., and Wittwer, C. T., editors, *Principles and Applications of Molecular Diagnostics*, pages 201–234. Elsevier.

Kingma, D. P. and Welling, M. (2019). An introduction to variational autoencoders. *arXiv preprint arXiv:1906.02691*.

Kiranmai, B. and Damodaram, A. (2014). A review on evaluation measures for data mining tasks. *International Journal Of Engineering And Computer Science*, 3(07).

Kocaleva, M., Stojanov, D., Stojanovic, I., and Zdravev, Z. (2016). Pattern recognition and natural language processing: State of the art. *TEM Journal*, 5(2):236–240.

Kotu, V. and Deshpande, B. (2015). Chapter 9 - text mining. In Kotu, V. and Deshpande, B., editors, *Predictive Analytics and Data Mining*, pages 275 – 303. Morgan Kaufmann, Boston.

Kramer, O. (2017). *Genetic algorithm essentials*, volume 679. Springer.

Kulkarni-Kale, U., Raskar-Renuse, S., Natekar-Kalantre, G., and Saxena, S. A. (2014). Antigen–antibody interaction database (agabdb): a compendium of antigen–antibody interactions. In *Immunoinformatics*, pages 149–164. Springer.

Kuroda, D. and Tsumoto, K. (2018). Antibody affinity maturation by computational design. In *Antibody Engineering*, pages 15–34. Springer.

Lai, K., Twine, N., O'brien, A., Guo, Y., and Bauer, D. (2018). Artificial intelligence and machine learning in bioinformatics. *Encyclopedia of Bioinformatics and Computational Biology: ABC of Bioinformatics*, 55:272.

Landsteiner, K. (2013). *The specificity of serological reactions*. Courier Corporation.

Lane, W. J., Westhoff, C. M., Gleadall, N. S., Aguad, M., Smeland-Wagman, R., Vege, S., Simmons, D. P., Mah, H. H., Lebo, M. S., Walter, K., et al. (2018). Automated typing of red blood cell and platelet antigens: a whole-genome sequencing study. *The Lancet Haematology*, 5(6):e241–e251.

Lavelli, A., Sebastiani, F., and Zanoli, R. (2004). Distributional term representations: an experimental comparison. In *Proceedings of the thirteenth ACM international conference on Information and knowledge management*, pages 615–624.

Lawce, H. J. and Brown, M. G. (2017). Cytogenetics: an overview. *The AGT Cytogenetics Laboratory Manual*, pages 25–85.

Leinonen, R., Akhtar, R., Birney, E., Bower, L., Cerdeno-Tárraga, A., Cheng, Y., Cleland, I., Faruque, N., Goodgame, N., Gibson, R., et al. (2010). The european nucleotide archive. *Nucleic acids research*, 39(suppl_1):D28–D31.

Lewis, E. B. (1957). Leukemia and ionizing radiation. *Science*, 125(3255):965–972.

Liberis, E., Veličković, P., Sormanni, P., Vendruscolo, M., and Liò, P. (2018). Parapred: antibody paratope prediction using convolutional and recurrent neural networks. *Bioinformatics*, 34(17):2944–2950.

Lima, W. C., Gasteiger, E., Marcatili, P., Duek, P., Bairoch, A., and Cosson, P. (2020). The abcd database: a repository for chemically defined antibodies. *Nucleic acids research*, 48(D1):D261–D264.

Luo, H., Wang, L., Bao, D., Wang, L., Zhao, H., Lian, Y., Yan, M., Mohan, C., and Li, Q.-Z. (2019). Novel autoantibodies related to cell death and dna repair pathways in systemic lupus erythematosus. *Genomics, proteomics & bioinformatics*, 17(3):248–259.

Martin, A. C. (1996). Accessing the kabat antibody sequence database by computer. *Proteins: Structure, Function, and Bioinformatics*, 25(1):130–133.

Mastache, E. F., Fernández, A. G., and Abalde, S. L. (2005). Linfocitos t y b. clasificación. receptores. generación de diversidad: mecanismos moleculares.

capacidades funcionales. *Medicine: Programa de Formación Médica Continuada Acreditado*, 9(33):2162–2173.

McGowan, E., Rosenthal, R., Fiore-Gartland, A., Macharia, G., Balinda, S., Kapaata, A., Umviligihozo, G., Muok, E., Dalel, J., Streatfield, C. L., et al. (2021). Utilizing computational machine learning tools to understand immunogenic breadth in the context of a cd8 t-cell mediated hiv response. *Frontiers in immunology*, 12:367.

McInnes, L., Healy, J., and Melville, J. (2018). Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

McKinney, W. et al. (2011). pandas: a foundational python library for data analysis and statistics. *Python for High Performance and Scientific Computing*, 14(9).

McKinnon, K. M. (2018). Flow cytometry: an overview. *Current protocols in immunology*, 120(1):5–1.

Medina-Ortiz, D., Contreras, S., Amado-Hinojosa, J., Torres-Almonacid, J., Asenjo, J. A., Navarrete, M., and Olivera-Nappa, Á. (2020a). Combination of digital signal processing and assembled predictive models facilitates the rational design of proteins. *arXiv preprint arXiv:2010.03516*.

Medina-Ortiz, D., Contreras, S., Quiroz, C., Asenjo, J. A., and Olivera-Nappa, Á. (2020b). Dmakit: A user-friendly web platform for bringing state-of-the-art data analysis techniques to non-specific users. *Information Systems*, page 101557.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Montesinos-Rongen, M., Terrao, M., May, C., Marcus, K., Blümcke, I., Hellmich, M., Küppers, R., Brunn, A., and Deckert, M. (2020). The process of somatic hypermutation increases polyreactivity for central nervous system antigens in primary central nervous system lymphoma. *Haematologica*.

Morfill, J., Blank, K., Zahnd, C., Luginbühl, B., Kühner, F., Gottschalk, K.-E., Plückthun, A., and Gaub, H. E. (2007). Affinity-matured recombinant antibody fragments analyzed by single-molecule force spectroscopy. *Biophysical journal*, 93(10):3583–3590.

Mougiakakos, D., Choudhury, A., Lladser, A., Kiessling, R., and Johansson, C. C. (2010). Chapter 3 - regulatory t cells in cancer. In Vande Woude, G. F. and Klein, G., editors, *Advances in Cancer Research*, volume 107 of *Advances in Cancer Research*, pages 57–117. Academic Press.

Nagelkerke, N. J. et al. (1991). A note on a general definition of the coefficient of determination. *Biometrika*, 78(3):691–692.

Neill, S. P. and Hashemi, M. R. (2018). Chapter 8 - ocean modelling for resource

characterization. In Neill, S. P. and Hashemi, M. R., editors, *Fundamentals of Ocean Renewable Energy*, E-Business Solutions, pages 193 – 235. Academic Press.

Niuniu, X. and Yuxun, L. (2010). Review of decision trees. In *2010 3rd International Conference on Computer Science and Information Technology*.

Nosrati, M., Mohabatkar, H., and Behbahani, M. (2020). Introducing of an integrated artificial neural network and chou's pseudo amino acid composition approach for computational epitope-mapping of crimean-congo haemorrhagic fever virus antigens. *International Immunopharmacology*, 78:106020.

Olkhov, R. V. and Shaw, A. M. (2008). Label-free antibody–antigen binding detection by optical sensor array based on surface-synthesized gold nanoparticles. *Biosensors and Bioelectronics*, 23(8):1298–1302.

Olsen, L. R., Tongchusak, S., Lin, H., Reinherz, E. L., Brusic, V., and Zhang, G. L. (2017). Tantigen: a comprehensive database of tumor t cell antigens. *Cancer Immunology, Immunotherapy*, 66(6):731–735.

Ong, E., Wong, M. U., Huffman, A., and He, Y. (2020). Covid-19 coronavirus vaccine design using reverse vaccinology and machine learning. *Frontiers in immunology*, 11:1581.

Ou, M., Cui, P., Pei, J., Zhang, Z., and Zhu, W. (2016). Asymmetric transitivity preserving graph embedding. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1105–1114.

Pagel, J. and Kirshtein, P. (2017). Chapter six - neural networks: The hard and software logic. In Pagel, J. and Kirshtein, P., editors, *Machine Dreaming and Consciousness*, pages 83 – 92. Academic Press, San Diego.

Parvizpour, S., Pourseif, M. M., Razmara, J., Rafi, M. A., and Omidi, Y. (2020). Epitope-based vaccine design: a comprehensive overview of bioinformatics approaches. *Drug discovery today*, 25(6):1034–1042.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. (2019). Pytorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems*, pages 8026–8037.

Patel, K. A. and Thakral, P. (2016). The best clustering algorithms in data mining. In *2016 International Conference on Communication and Signal Processing (ICCSP)*, pages 2042–2046. IEEE.

Patronov, A. and Doytchinova, I. (2013). T-cell epitope vaccine design by immunoinformatics. *Open biology*, 3(1):120139.

Pellegrino, M., Sciambi, A., Treusch, S., Durruthy-Durruthy, R., Gokhale, K., Jacob, J., Chen, T. X., Geis, J. A., Oldham, W., Matthews, J., et al. (2018).

High-throughput single-cell dna sequencing of acute myeloid leukemia tumors with droplet microfluidics. *Genome research*, 28(9):1345–1352.

Petersdorf, E. W. (2017). Role of major histocompatibility complex variation in graft-versus-host disease after hematopoietic cell transplantation. *F1000Research*, 6.

Poiron, C., Wu, Y., Ginestoux, C., Ehrenmann, F., Duroux, P., and Lefranc, M. (2010). Imgt/mab-db: the imgt® database for therapeutic monoclonal antibodies. *Poster no101*, 11.

Potdar, K., Pardawala, T. S., and Pai, C. D. (2017). A comparative study of categorical variable encoding techniques for neural network classifiers. *International journal of computer applications*, 175(4):7–9.

Poznyak, T. I., Oria], I. C., and Poznyak, A. S. (2019). Chapter3 - background on dynamic neural networks. In Poznyak, T. I., Oria], I. C., and Poznyak, A. S., editors, *Ozonation and Biodegradation in Environmental Engineering*, pages 57 – 74. Elsevier.

Pui, C.-H., Relling, M. V., and Downing, J. R. (2004). Acute lymphoblastic leukemia. *New England Journal of Medicine*, 350(15):1535–1548.

Puka, L. (2011). *Kendall's Tau*, pages 713–715. Springer Berlin Heidelberg", address="Berlin, Heidelberg.

Qi, H., Ma, M., Hu, C., Xu, Z.-w., Wu, F.-l., Wang, N., Lai, D.-y., Li, Y., Zhang, H., Jiang, H.-w., et al. (2021). Antibody binding epitope mapping (abmap) of hundred antibodies in a single run. *Molecular & Cellular Proteomics*, page 100059.

Qiu, B.-Z., Li, X.-L., and Shen, J.-Y. (2007). Grid-based clustering algorithm based on intersecting partition and density estimation. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 368–377. Springer.

Quiroz, C., Saavedra, Y. B., Armijo-Galdames, B., Amado-Hinojosa, J., Olivera-Nappa, Á., Sanchez-Daza, A., and Medina-Ortiz, D. (2021). Peptipedia: a comprehensive database for peptide research supported by assembled predictive models and data mining approaches. *arXiv preprint arXiv:2101.12210*.

Radovic, M., Ghalwash, M., Filipovic, N., and Obradovic, Z. (2017). Minimum redundancy maximum relevance feature selection approach for temporal gene expression data. *BMC bioinformatics*, 18(1):9.

Rao, R., Bhattacharya, N., Thomas, N., Duan, Y., Chen, P., Canny, J., Abbeel, P., and Song, Y. (2019). Evaluating protein transfer learning with tape. In *Advances in Neural Information Processing Systems*, pages 9686–9698.

Rappuoli, R. (2001). Reverse vaccinology, a genome-based approach to vaccine development. *Vaccine*, 19(17-19):2688–2691.

Raybould, M. I., Marks, C., Lewis, A. P., Shi, J., Bujotzek, A., Taddese, B., and Deane, C. M. (2020). Thera-sabdab: the therapeutic structural antibody database. *Nucleic acids research*, 48(D1):D383–D388.

Rebala, G., Ravi, A., and Churiwala, S. (2019). *(Artificial) Neural Networks*, pages 103–116. Springer International Publishing, Cham.

Richard, A. J. (2007). *Applied multivariate statistical analysis*. Pearson Educational Inc.

Ripundeep Singh Gill, A. (2014). Neural networks in data mining. *IOSR Journal of Engineering*, 4.

Robinson, J., Barker, D. J., Georgiou, X., Cooper, M. A., Flicek, P., and Marsh, S. G. (2020). Ipd-imgt/hla database. *Nucleic acids research*, 48(D1):D948–D955.

Robinson, W. H., DiGennaro, C., Hueber, W., Haab, B. B., Kamachi, M., Dean, E. J., Fournel, S., Fong, D., Genovese, M. C., De Vegvar, H. E. N., et al. (2002). Autoantigen microarrays for multiplex characterization of autoantibody responses. *Nature medicine*, 8(3):295–301.

Rosen, O. and Anglister, J. (2009). Epitope mapping of antibody–antigen complexes by nuclear magnetic resonance spectroscopy. In *Epitope Mapping Protocols*, pages 37–57. Springer.

Rosenblatt, F. (1960). Perceptron simulation experiments. *Proceedings of the IRE*, 48(3):301–309.

Roth, D. B. (2015). V (d) j recombination: mechanism, errors, and fidelity. *Mobile DNA III*, pages 311–324.

Sabath, D. (2013). Leukemia. In Maloy, S. and Hughes, K., editors, *Brenner's Encyclopedia of Genetics (Second Edition)*, pages 226–227. Academic Press, San Diego, second edition edition.

Sadam, H., Pihlak, A., Jaago, M., Pupina, N., Rähni, A., Toots, M., Vaheri, A., Nieminen, J. K., Siuko, M., Tienari, P. J., et al. (2021). Identification of two highly antigenic epitope markers predicting multiple sclerosis in optic neuritis patients. *EBioMedicine*, 64:103211.

Saha, S., Bhasin, M., and Raghava, G. P. (2005). Bcipep: a database of b-cell epitopes. *BMC genomics*, 6(1):1–7.

Sakaguchi, S., Powrie, F., and Ransohoff, R. M. (2012). Re-establishing immunological self-tolerance in autoimmune disease. *Nature medicine*, 18(1):54–58.

Sakaguchi, S., Yamaguchi, T., Nomura, T., and Ono, M. (2008). Regulatory t cells and immune tolerance. *cell*, 133(5):775–787.

Sarkar, D., Bali, R., and Sharma, T. (2018). Practical machine learning with

python. *A Problem-Solvers Guide To Building Real-World Intelligent Systems. Berkely: Apress.*

Sayers, E. W., Cavanaugh, M., Clark, K., Ostell, J., Pruitt, K. D., and Karsch-Mizrachi, I. (2019). Genbank. *Nucleic acids research*, 47(D1):D94–D99.

Schlessinger, A., Ofran, Y., Yachdav, G., and Rost, B. (2006). Epitome: database of structure-inferred antigenic epitopes. *Nucleic acids research*, 34(suppl_1):D777–D780.

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, 61:85–117.

Schneider, T. D. and Stephens, R. M. (1990). Sequence logos: a new way to display consensus sequences. *Nucleic acids research*, 18(20):6097–6100.

Schreurs, J. and Suykens, J. (2018). Generative kernel pca. *ESANN 2018*, pages 129–134.

Schwartz, R. H. (2005). Natural regulatory t cells and self-tolerance. *Nature immunology*, 6(4):327–330.

Schwede, T., Kopp, J., Guex, N., and Peitsch, M. C. (2003). Swiss-model: an automated protein homology-modeling server. *Nucleic acids research*, 31(13):3381–3385.

Sedgwick, P. (2012). Pearson's correlation coefficient. *Bmj*, 345:e4483.

Sedgwick, P. (2014). Spearman's rank correlation coefficient. *Bmj*, 349:g7327.

Sharma, A. R. and Kaushik, P. (2017). Literature survey of statistical, deep and reinforcement learning in natural language processing. In *2017 International Conference on Computing, Communication and Automation (ICCCA)*, pages 350–354. IEEE.

Sharma, O., Das, A., Krishna, R., Suresh Kumar, M., and Mathur, P. (2012). Structural epitope database (sedb): a web-based database for the epitope, and its intermolecular interaction along with the tertiary structure information. *J Proteomics Bioinform*, 5:084–089.

Sheng, Z., Guo, Y., Chen, K., Kwong, P. D., and Shapiro, L. (2019). cab-rep: A database of curated antibody repertoires for exploring antibody diversity and predicting antibody prevalence. *Frontiers in immunology*, 10:2365.

Shimba, N., Kamiya, N., and Nakamura, H. (2016). Model building of antibody–antigen complex structures using gbsa scores. *Journal of chemical information and modeling*, 56(10):2005–2012.

Sievers, F. and Higgins, D. G. (2014). Clustal omega. *Current protocols in bioinformatics*, 48(1):3–13.

Sifniotis, V., Cruz, E., Eroglu, B., and Kayser, V. (2019). Current advancements

in addressing key challenges of therapeutic antibody design, manufacture, and formulation. *Antibodies*, 8(2):36.

Singh, M. K., Srivastava, S., Raghava, G., and Varshney, G. C. (2006). Haptendb: a comprehensive database of haptens, carrier proteins and anti-hapten antibodies. *Bioinformatics*, 22(2):253–255.

Singh, Y. and Chauhan, A. S. (2009). Neural networks in data mining. *Journal of Theoretical & Applied Information Technology*, 5(1).

Sinkov, A., Asyaev, G., Mursalimov, A., and Nikolskaya, K. (2016). Neural networks in data mining. In *2016 2nd International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM)*, pages 1–5. IEEE.

Smith, C. C., Chai, S., Washington, A. R., Lee, S. J., Landoni, E., Field, K., Garness, J., Bixby, L. M., Selitsky, S. R., Parker, J. S., et al. (2019). Machine-learning prediction of tumor antigen immunogenicity in the selection of therapeutic epitopes. *Cancer immunology research*, 7(10):1591–1604.

Sokolova, M. and Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information processing & management*, 45(4):427–437.

Song, Y. and Roth, D. (2015). Unsupervised sparse vector densification for short text similarity. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1275–1280.

Spiess, C., Zhai, Q., and Carter, P. J. (2015). Alternative molecular formats and therapeutic applications for bispecific antibodies. *Molecular immunology*, 67(2):95–106.

Stefanescu, R., Iacob, R. E., Damoc, E. N., Marquardt, A., Amstalden, E., Manea, M., Perdivara, I., Maftei, M., Paraschiv, G., and Przybylski, M. (2007). Mass spectrometric approaches for elucidation of antigen—antibody recognition structures in molecular immunology. *European Journal of Mass Spectrometry*, 13(1):69–75.

Stepanov, V. and Trifonova, E. (2013). Multiplex snp genotyping by maldi-tof mass spectrometry: Frequencies of 56 immune response gene snps in human populations. *Molecular Biology*, 47(6):852–862.

Sulchek, T. A., Friddle, R. W., Langry, K., Lau, E. Y., Albrecht, H., Ratto, T. V., DeNardo, S. J., Colvin, M. E., and Noy, A. (2005). Dynamic force spectroscopy of parallel individual mucin1–antibody bonds. *Proceedings of the National Academy of Sciences*, 102(46):16638–16643.

Swindells, M. B., Porter, C. T., Couch, M., Hurst, J., Abhinandan, K., Nielsen, J. H., Macindoe, G., Hetherington, J., and Martin, A. C. (2017). abysis: integrated antibody sequence and structure—management, analysis, and prediction. *Journal of molecular biology*, 429(3):356–364.

Takeuchi, Y. and Nishikawa, H. (2016). Roles of regulatory t cells in cancer immunity. *International immunology*, 28(8):401–409.

Talabis, M. R. M., McPherson, R., Miyamoto, I., Martin, J. L., and Kaye, D. (2015). Chapter 1 - analytics defined. In Talabis, M. R. M., McPherson, R., Miyamoto, I., Martin, J. L., and Kaye, D., editors, *Information Security Analytics*, pages 1 – 12. Syngress, Boston.

Tan, C. C. and Eswaran, C. (2008). Performance comparison of three types of autoencoder neural networks. In *2008 Second Asia International Conference on Modelling & Simulation (AMS)*, pages 213–218. IEEE.

Tan, X., Li, D., Huang, P., Jian, X., Wan, H., Wang, G., Li, Y., Ouyang, J., Lin, Y., and Xie, L. (2020). dbpepneo: a manually curated database for human tumor neoantigen peptides. *Database*, 2020.

Tanaka, A. and Sakaguchi, S. (2019). Targeting treg cells in cancer immunotherapy. *European journal of immunology*, 49(8):1140–1146.

Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., and Mei, Q. (2015). Line: Large-scale information network embedding. In *Proceedings of the 24th international conference on world wide web*, pages 1067–1077.

Tavanaei, A., Ghodrati, M., Kheradpisheh, S. R., Masquelier, T., and Maida, A. (2019). Deep learning in spiking neural networks. *Neural Networks*, 111:47–63.

Tenorio, M. F. and Lee, W.-T. (1989). Self organizing neural networks for the identification problem. In *Advances in neural information processing systems*, pages 57–64.

Thilagavathi, G., Srivaishnavi, D., Aparna, N., et al. (2013). A survey on efficient hierarchical algorithm used in clustering. *International Journal of Engineering*, 2(9):165–176.

Thomas, J. S. and Thivarkaran, T. (2020). Data mining algorithms and statistical techniques for identification of schizophrenia: A survey. In *2020 International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE)*, pages 246–251. IEEE.

Tlili, A., Jarboui, M. A., Abdelghani, A., Fathallah, D., and Maaref, M. (2005). A novel silicon nitride biosensor for specific antibody–antigen interaction. *Materials Science and Engineering: C*, 25(4):490–495.

Tong, J. C., Song, C. M., Tan, P. T. J., Ren, E. C., and Sinha, A. A. (2008). Beid: Database for sequence-structure-function information on antigen-antibody interactions. *Bioinformation*, 3(2):58.

Torres Almonacid, J. (2020). *Aplicación de minería de datos para la búsqueda de patrones relacionados al sistema de interacción antígeno-anticuerpo*. Bachelor's thesis, Universidad De Magallanes FACULTAD DE INGENIERÍA, DEPARTAMENTO DE INGENIERÍA EN COMPUTACIÓN.

Torres-Almonacid, J., Medina-Ortiz, D., Alvarez-Saravia, D., Águila-Guerrero, J., Olivera-Nappa, Á., and Navarrete, M. (2019). Pattern recognition on antigen-antibody interactions from protein microarrays based on data mining and bioinformatics analysis. In *2019 38th International Conference of the Chilean Computer Science Society (SCCC)*, pages 1–8. IEEE.

Tuttle iV, P. V., Rundell, A. E., and Webster, T. J. (2006). Influence of biologically inspired nanometer surface roughness on antigen–antibody interactions for immunoassay–biosensor applications. *international Journal of nanomedicine*, 1(4):497.

Van Regenmortel, M. H. (2019). *HIV/AIDS: Immunochemistry, Reductionism and Vaccine Design: A Review of 20 Years of Research*. Springer Nature.

Van Zundert, G., Rodrigues, J., Trellet, M., Schmitz, C., Kastritis, P., Karaca, E., Melquiond, A., van Dijk, M., De Vries, S., and Bonvin, A. (2016). The haddock2. 2 web server: user-friendly integrative modeling of biomolecular complexes. *Journal of molecular biology*, 428(4):720–725.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Verborgh, R. and De Wilde, M. (2013). *Using OpenRefine*. Packt Publishing Ltd.

Vignali, D. A., Collison, L. W., and Workman, C. J. (2008). How regulatory t cells work. *Nature reviews immunology*, 8(7):523–532.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., et al. (2020). Scipy 1.0: fundamental algorithms for scientific computing in python. *Nature methods*, 17(3):261–272.

Vita, R., Overton, J. A., Greenbaum, J. A., Ponomarenko, J., Clark, J. D., Cantrell, J. R., Wheeler, D. K., Gabbard, J. L., Hix, D., Sette, A., et al. (2015). The immune epitope database (iedb) 3.0. *Nucleic acids research*, 43(D1):D405–D412.

Walt, S. v. d., Colbert, S. C., and Varoquaux, G. (2011). The numpy array: a structure for efficient numerical computation. *Computing in science & engineering*, 13(2):22–30.

Walter Hugo, Sandra Vieira, R. G.-D. and Mechelli, A. (2020). Chapter 11 - autoencoders. In Mechelli, A. and Vieira, S., editors, *Machine Learning*, pages 193 – 208. Academic Press.

Wang, J., He, H., and Prokhorov, D. V. (2012). A folded neural network autoencoder for dimensionality reduction. *Procedia Computer Science*, 13:120–127.

Wang, S., Li, W., Liu, S., and Xu, J. (2016). Raptorx-property: a web server for

protein structure property prediction. *Nucleic acids research*, 44(W1):W430–W435.

Wang, W., Huang, Y., Wang, Y., and Wang, L. (2014). Generalized autoencoder: A neural network framework for dimensionality reduction. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 490–497.

Wang, W., Singh, S., Zeng, D. L., King, K., and Nema, S. (2007). Antibody structure, instability, and formulation. *Journal of pharmaceutical sciences*, 96(1):1–26.

Wang, X., Zhao, H., Xu, Q., Jin, W., Liu, C., Zhang, H., Huang, Z., Zhang, X., Zhang, Y., Xin, D., et al. (2006). Hptaa database-potential target genes for clinical diagnosis and immunotherapy of human carcinoma. *Nucleic acids research*, 34(suppl_1):D607–D612.

Wang, Y., Zhang, H., Zhong, H., and Xue, Z. (2021). Protein domain identification methods and online resources. *Computational and Structural Biotechnology Journal*, 19:1145.

Wec, A. Z., Lin, K. S., Kwasnieski, J. C., Sinai, S., Gerold, J., and Kelsic, E. D. (2021). Overcoming immunological challenges limiting capsid-mediated gene therapy with machine learning. *Frontiers in Immunology*, 12:1443.

Weitzner, B. D., Jeliazkov, J. R., Lyskov, S., Marze, N., Kuroda, D., Frick, R., Adolf-Bryfogle, J., Biswas, N., Dunbrack Jr, R. L., and Gray, J. J. (2017). Modeling and docking of antibody structures with rosetta. *Nature protocols*, 12(2):401.

Xie, H. (2017). Validated antibody database: A curated database of antibodies cited in formal publications.

Xu, K., Acharya, P., Kong, R., Cheng, C., Chuang, G.-Y., Liu, K., Louder, M. K., O'Dell, S., Rawi, R., Sastry, M., et al. (2018). Epitope-based vaccine design yields fusion peptide-directed antibodies that neutralize diverse strains of hiv-1. *Nature medicine*, 24(6):857–867.

Yachdav, G., Kloppmann, E., Kajan, L., Hecht, M., Goldberg, T., Hamp, T., Hönigschmid, P., Schafferhans, A., Roos, M., Bernhofer, M., et al. (2014). Predictprotein—an open resource for online prediction of protein structural and functional features. *Nucleic acids research*, 42(W1):W337–W343.

Yamashita, T. (2018). Toward rational antibody design: Recent advancements in molecular dynamics simulations. *International immunology*, 30(4):133–140.

Yang, B., Sayers, S., Xiang, Z., and He, Y. (2011). Protegen: a web-based protective antigen database and analysis system. *Nucleic acids research*, 39(suppl_1):D1073–D1078.

Yang, H. and Moody, J. (1999). Feature selection based on joint mutual information.

In *Proceedings of international ICSC symposium on advances in intelligent data analysis*, pages 22–25. Citeseer.

Yang, J., Anishchenko, I., Park, H., Peng, Z., Ovchinnikov, S., and Baker, D. (2020). Improved protein structure prediction using predicted interresidue orientations. *Proceedings of the National Academy of Sciences*, 117(3):1496–1503.

Yang, W., Butler, J. E., Russell Jr, J. N., and Hamers, R. J. (2007). Direct electrical detection of antigen–antibody binding on diamond and silicon substrates using electrical impedance spectroscopy. *Analyst*, 132(4):296–306.

Yao, D., Su, H., Zhu, J., Zhao, X., Aweya, J. J., Wang, F., Zhong, M., and Zhang, Y. (2018). Snps in the toll1 receptor of litopenaeus vannamei are associated with immune response. *Fish & shellfish immunology*, 72:410–417.

Yasser, E.-M., Dobbs, D., and Honavar, V. G. (2017). In silico prediction of linear b-cell epitopes on proteins. In *Prediction of Protein Secondary Structure*, pages 255–264. Springer.

Yong, C. Y., Ong, H. K., Yeap, S. K., Ho, K. L., and Tan, W. S. (2019). Recent advances in the vaccine development against middle east respiratory syndrome-coronavirus. *Frontiers in microbiology*, 10:1781.

Yu, K., Lung, P.-Y., Zhao, T., Zhao, P., Tseng, Y.-Y., and Zhang, J. (2018). Automatic extraction of protein-protein interactions using grammatical relationship graph. *BMC medical informatics and decision making*, 18(2):42.

Zeng, Z., Shi, H., Wu, Y., and Hong, Z. (2015). Survey of natural language processing techniques in bioinformatics. *Computational and mathematical methods in medicine*, 2015.

Zhang, G. P. (2009). Neural networks for data mining. In *Data mining and knowledge discovery handbook*, pages 419–444. Springer.

Zhang, M., Fritsche, J., Roszik, J., Williams, L. J., Peng, X., Chiu, Y., Tsou, C.-C., Hoffgaard, F., Goldfinger, V., Schoor, O., et al. (2018). Rna editing derived epitopes function as cancer antigens to elicit immune responses. *Nature communications*, 9(1):1–10.

Zhao, J., Nussinov, R., and Ma, B. (2018). Antigen induced dynamic conformation changes of antibody to facilitate recognition of fc receptors. *Biophysical Journal*, 114(3).

Zhou, S., Zhu, X., Shen, N., Li, Q., Wang, N., You, Y., Zhong, Z., Cheng, F., Zou, P., and Zhu, X. (2019). T cells expressing cd26-specific chimeric antigen receptors exhibit extensive self-antigen-driven fratricide. *Immunopharmacology and immunotoxicology*, 41(4):490–496.

Zhu, X. and Goldberg, A. B. (2009). Introduction to semi-supervised learning. *Synthesis lectures on artificial intelligence and machine learning*, 3(1):1–130.

Apéndice A

Material suplementario

Cuadro A.0.1: Tabla modelos generados por set de datos utilizado en proceso exploratorio

Algoritmo	**Dataset**	**N° modelos**
Neural Network	Embedding Babbler	12
Neural Network	Embedding Bert-base	12
Neural Network	One hot	12
Neural Network	Physicochemical- Betha structure	12
Neural Network	Physicochemical- Alpha structure	12
Neural Network	Physicochemical- Hydropathy	12
Neural Network	Physicochemical- Hydrophobicity	12
Neural Network	Physicochemical- Index	12
Neural Network	Physicochemical- Volume	12
Neural Network	Physicochemical- Secondary structure properties	12
Neural Network	Physicochemical- Energetic	12

Cuadro A.0.1: Tabla modelos generados por set de datos utilizado en proceso exploratorio

Algoritmo	**Dataset**	**N° modelos**
Random Forest	Embedding Babbler	10
Random Forest	Embedding Bert-base	10
Random Forest	One hot	10
Random Forest	Physicochemical- Betha structure	10
Random Forest	Physicochemical- Alpha structure	10
Random Forest	Physicochemical- Hydropathy	10
Random Forest	Physicochemical- Hydrophobicity	10
Random Forest	Physicochemical- Index	10
Random Forest	Physicochemical- Volume	10
Random Forest	Physicochemical- Secondary structure properties	10
Random Forest	Physicochemical- Energetic	10
K-Nearest Neighbors	Embedding Babbler	5
K-Nearest Neighbors	Embedding Bert-base	5
K-Nearest Neighbors	Physicochemical- Betha structure	5
K-Nearest Neighbors	Physicochemical- Alpha structure	5
K-Nearest Neighbors	Physicochemical- Hydropathy	5
K-Nearest Neighbors	Physicochemical- Hydrophobicity	5
K-Nearest Neighbors	Physicochemical- Index	5

Cuadro A.0.1: Tabla modelos generados por set de datos utilizado en proceso exploratorio

Algoritmo	**Dataset**	**N° modelos**
K-Nearest Neighbors	Physicochemical-Volume	5
K-Nearest Neighbors	Physicochemical-Secondary structure properties	5
K-Nearest Neighbors	Physicochemical-Energetic	5
AdaBoost	Embedding Babbler	5
AdaBoost	Embedding Bert-base	5
AdaBoost	One Hot	5
AdaBoost	Physicochemical- Betha structure	5
AdaBoost	Physicochemical- Alpha structure	5
AdaBoost	Physicochemical-Hydropathy	5
AdaBoost	Physicochemical-Hydrophobicity	5
AdaBoost	Physicochemical- Index	5
AdaBoost	Physicochemical-Volume	5
AdaBoost	Physicochemical-Secondary structure properties	5
AdaBoost	Physicochemical-Energetic	5
Decision Tree	Embedding Babbler	4
Decision Tree	Embedding Bert-base	4
Decision Tree	One Hot	4
Decision Tree	Physicochemical- Betha structure	4

Cuadro A.0.1: Tabla modelos generados por set de datos utilizado en proceso exploratorio

Algoritmo	**Dataset**	**N° modelos**
Decision Tree	Physicochemical- Alpha structure	4
Decision Tree	Physicochemical- Hydropathy	4
Decision Tree	Physicochemical- Hydrophobicity	4
Decision Tree	Physicochemical- Index	4
Decision Tree	Physicochemical- Volume	4
Decision Tree	Physicochemical- Secondary structure properties	4
Decision Tree	Physicochemical- Energetic	4
Gaussian Naive Bayes	Embedding Babbler	1
Gaussian Naive Bayes	Embedding Bert-base	1
Gaussian Naive Bayes	One Hot	1
Gaussian Naive Bayes	Physicochemical- Betha structure	1
Gaussian Naive Bayes	Physicochemical- Alpha structure	1
Gaussian Naive Bayes	Physicochemical- Hydropathy	1
Gaussian Naive Bayes	Physicochemical- Hydrophobicity	1
Gaussian Naive Bayes	Physicochemical- Index	1
Gaussian Naive Bayes	Physicochemical- Volume	1
Gaussian Naive Bayes	Physicochemical- Secondary structure properties	1

Cuadro A.0.1: Tabla modelos generados por set de datos utilizado en proceso exploratorio

Algoritmo	**Dataset**	**N° modelos**
Gaussian Naive Bayes	Physicochemical-Energetic	1
Bernoulli Naive Bayes	Embedding Babbler	1
Bernoulli Naive Bayes	Embedding Bert-base	1
Bernoulli Naive Bayes	One Hot	1
Bernoulli Naive Bayes	Physicochemical- Betha structure	1
Bernoulli Naive Bayes	Physicochemical- Alpha structure	1
Bernoulli Naive Bayes	Physicochemical-Hydropathy	1
Bernoulli Naive Bayes	Physicochemical-Hydrophobicity	1
Bernoulli Naive Bayes	Physicochemical- Index	1
Bernoulli Naive Bayes	Physicochemical-Volume	1
Bernoulli Naive Bayes	Physicochemical-Secondary structure properties	1
Bernoulli Naive Bayes	Physicochemical-Energetic	1
	Total	413

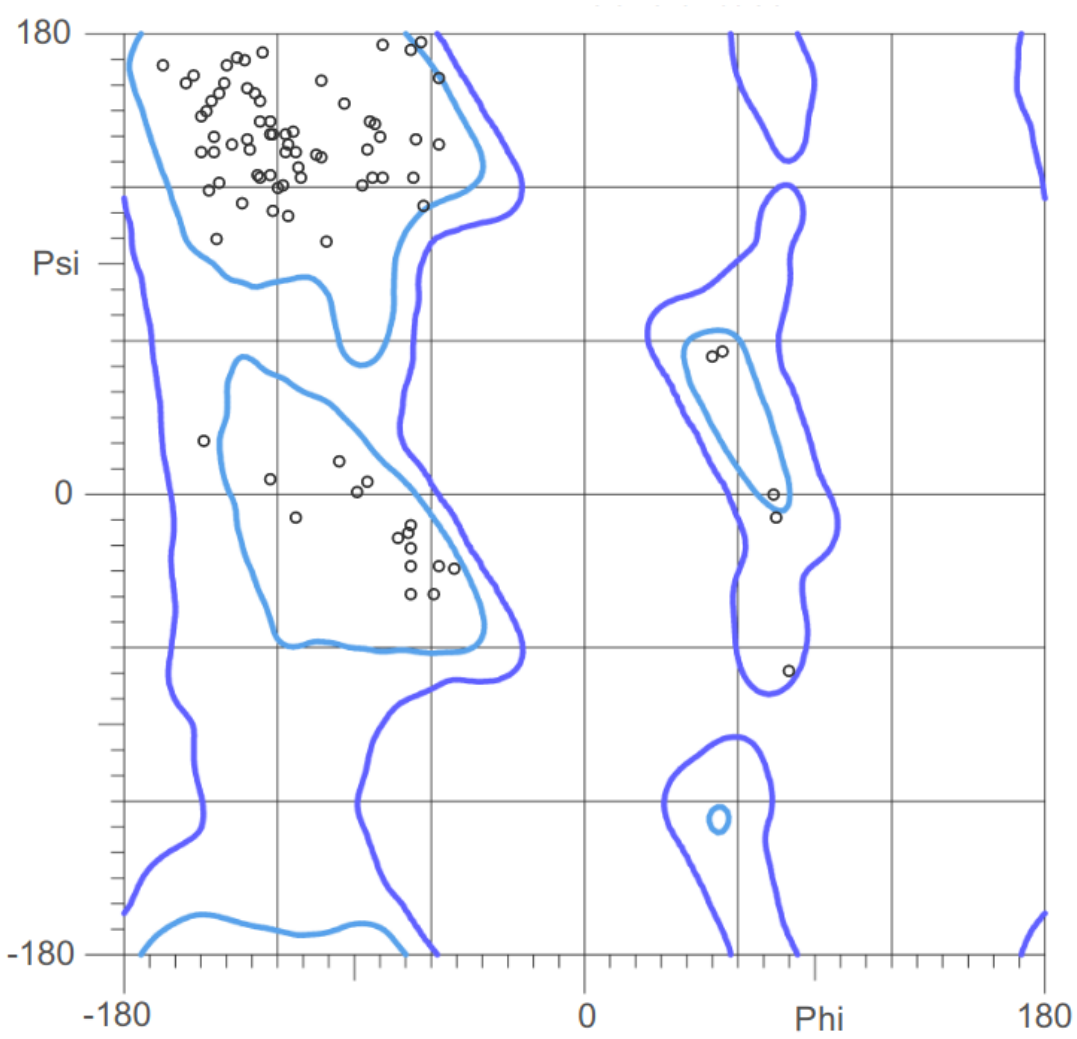


Figura A.0.1: Gráfico Ramachandran modelo A021. Se presenta un análisis de los enlaces phi y psi de los aminoácidos presentes en el modelo generado por Swiss model para el anticuerpo A021. Se observan a casi todos los aminoácidos en las regiones favorables para la disposición tridimensional. Sólo tres de los aminoácidos se presenta fuera de estas regiones, no obstante, se encuentra dentro de las regiones permitidas.

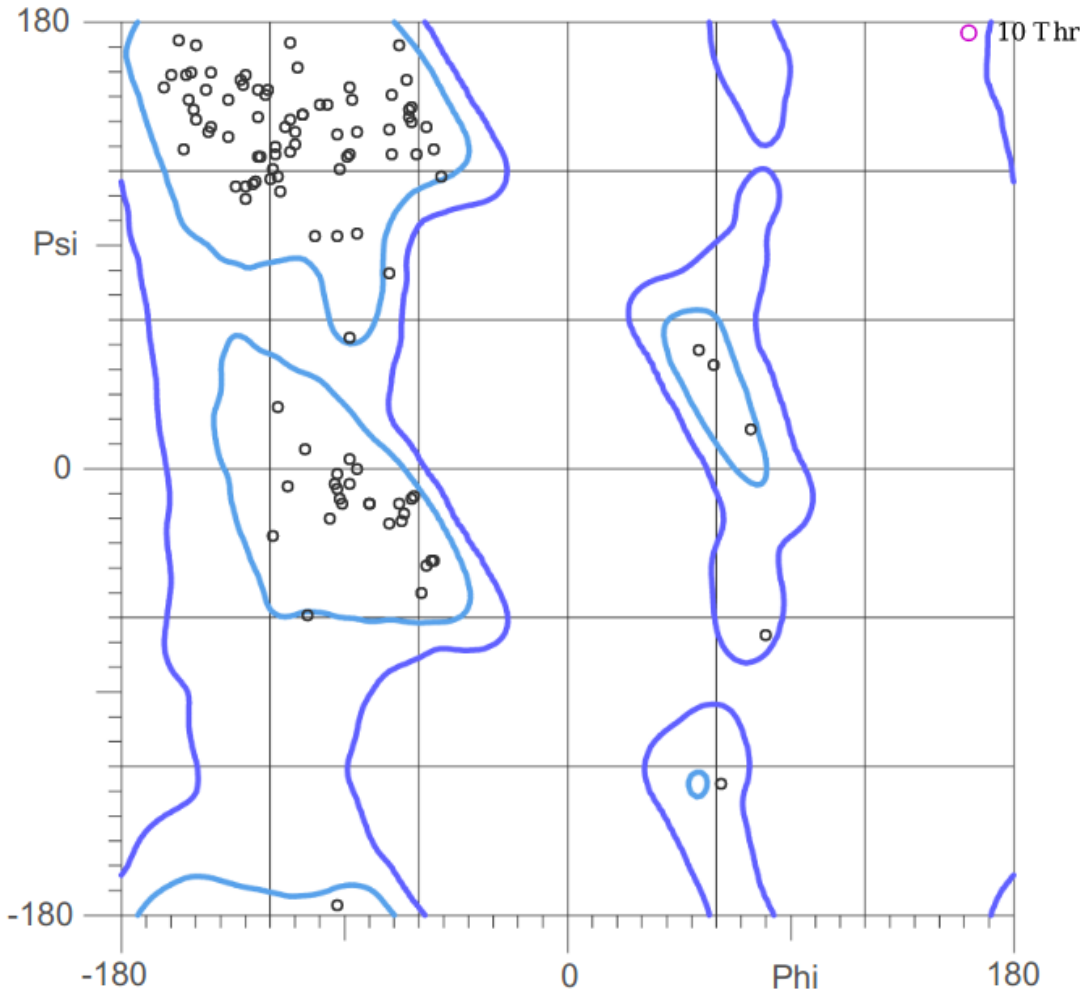


Figura A.0.2: Gráfico Ramachandran modelo A120. Se muestra un análisis de los enlaces phi y psi de los aminoácidos presentes en el modelo generado por Swiss model para el anticuerpo A120. Se observan a casi todos los aminoácidos en las regiones favorables para la disposición tridimensional. En este modelo se observan 4 aminoácidos fuera de las zonas favorables, pero aún dentro de las zonas permitidas para estos ángulos. Por otra parte, el residuo treonina se encuentra dentro de las regiones prohibidas de orientación.

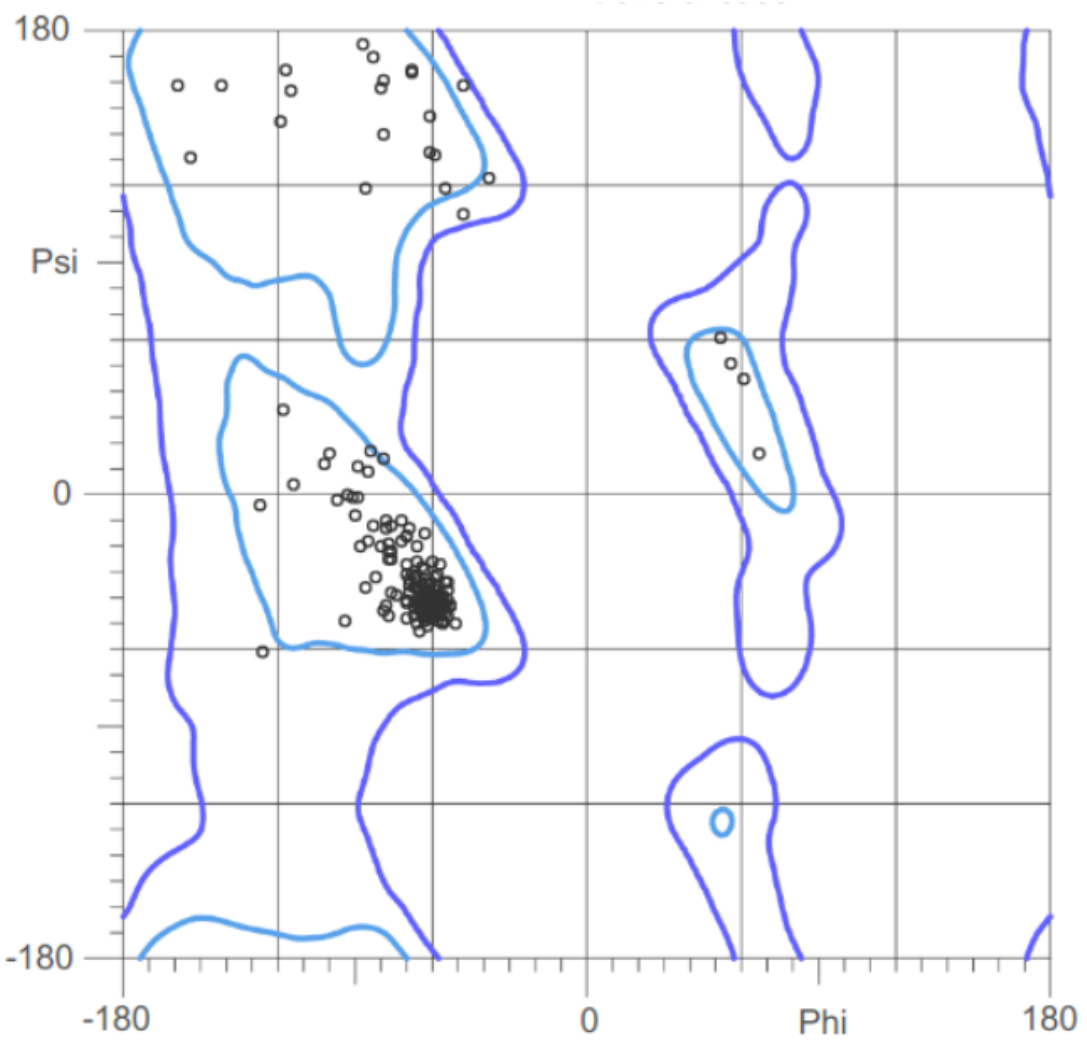


Figura A.0.3: Gráfico Ramachandran modelo uORF:IOH38079. Se observa un análisis de los enlaces phi y psi de los aminoácidos presentes en el modelo generado por Swiss model para el anticuerpo uORF:IOH38079. Se observan a casi todos los aminoácidos en las regiones favorables para la disposición tridimensional. En este modelo, al igual que con A120, se observan 4 aminoácidos fuera de las zonas favorables, pero aún dentro de las zonas permitidas para estos ángulos.

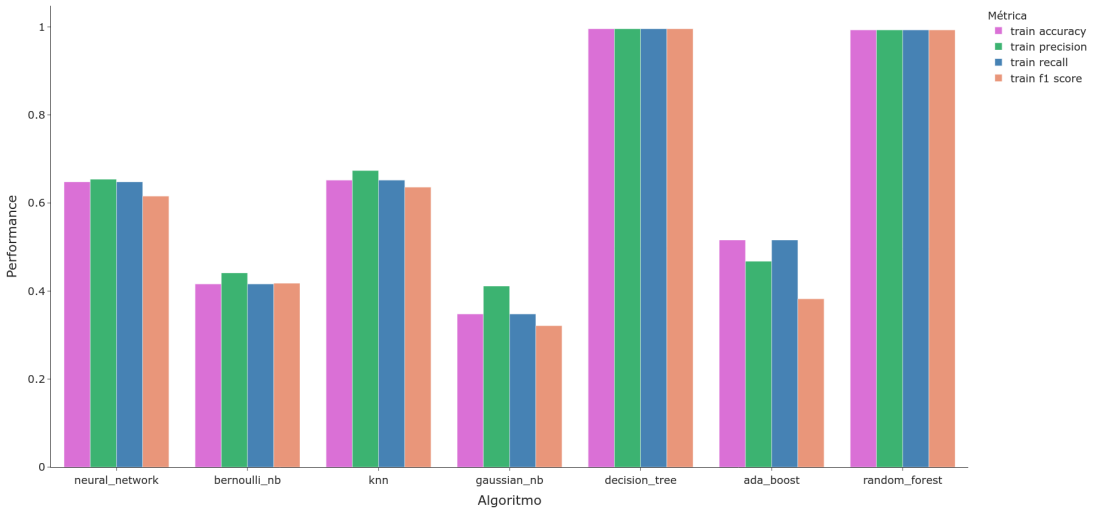


Figura A.0.4: Performance entrenamiento promedio algoritmos. Se presenta el valor promedio obtenido de performance para cada métrica seleccionada en etapa de entrenamiento, con respecto a los modelos generados por cada algoritmo. Se presenta en color rosa el accuracy, en verde precision, en azul el recall y en naranjo el f-score.

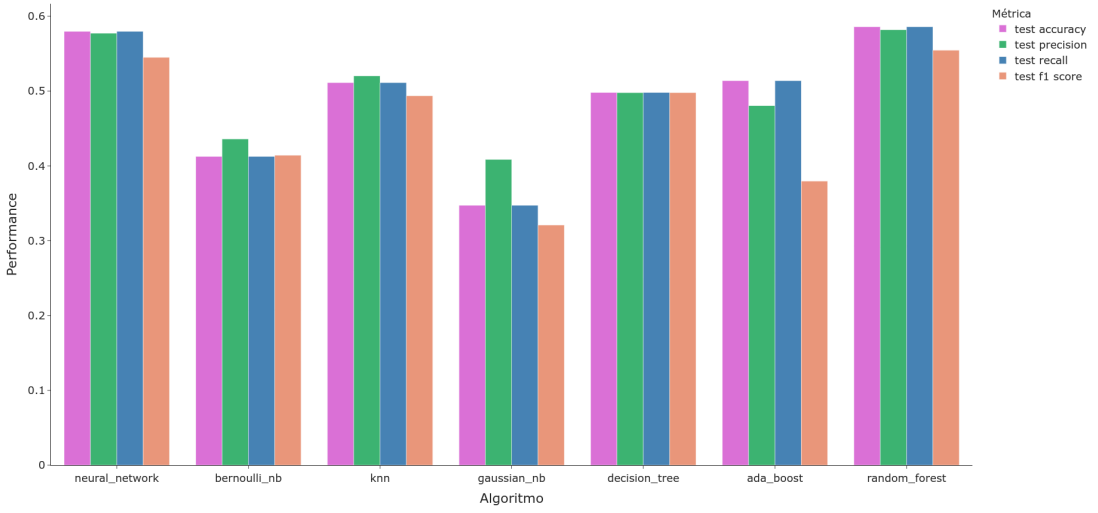


Figura A.0.5: Performance testeo promedio algoritmos. Se presenta el valor promedio obtenido de performance para cada métrica seleccionada en etapa de testeo, con respecto a los modelos generados por cada algoritmo. Se presenta en color rosa el accuracy, en verde precision, en azul el recall y en naranjo el f-score.

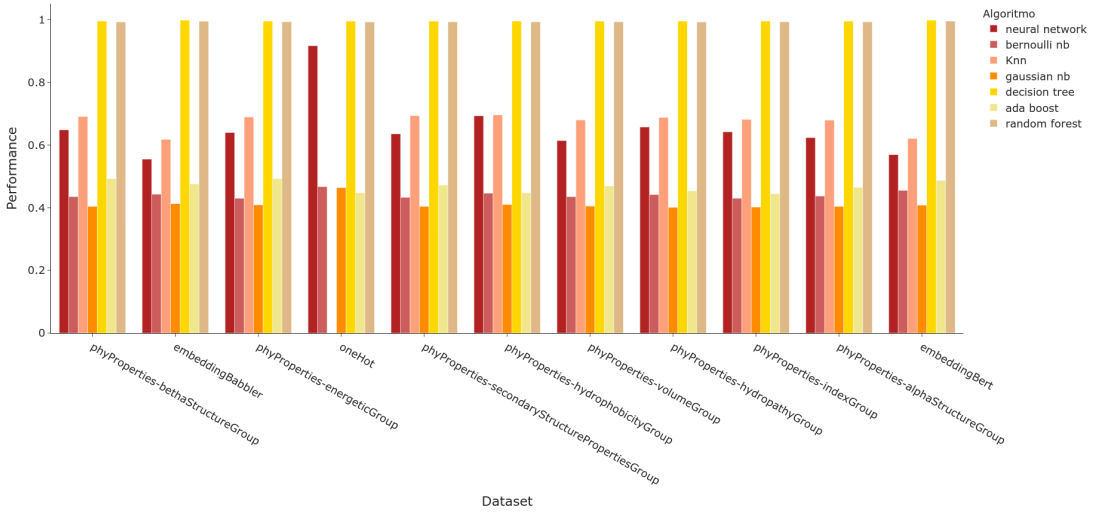


Figura A.0.6: Performance entrenamiento promedio dataset. Se presenta el valor promedio de performance obtenido para cada set de datos utilizado, con respecto a los algoritmos explorados.

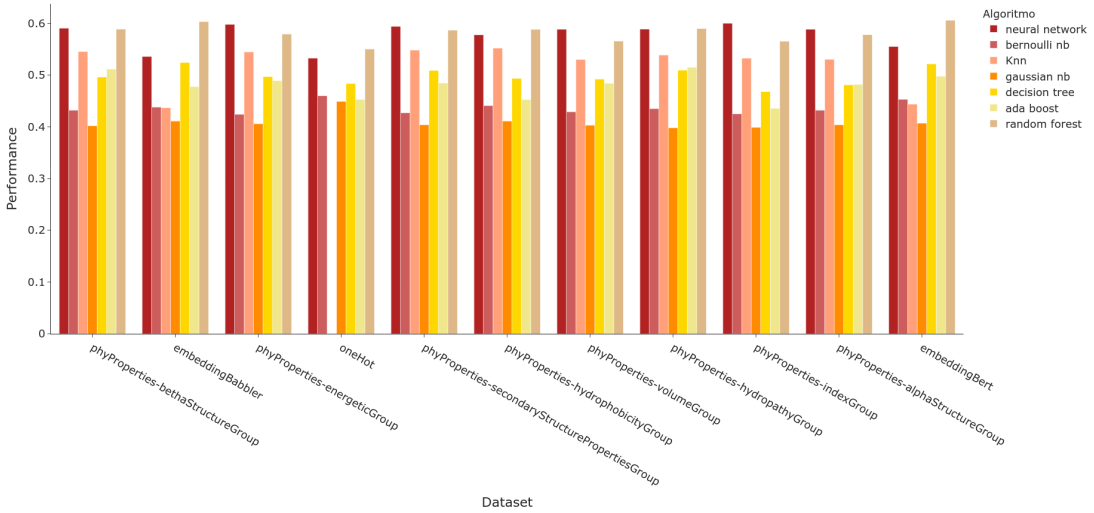


Figura A.0.7: Performance testeo promedio datasets. Se presenta el valor promedio de performance obtenido para cada set de datos utilizado, con respecto a los algoritmos explorados.