Predicción de interacciones proteína-proteína mediante un método basado en aprendizaje de máquina para el análisis de la proteína NS5A del virus GB tipo C.

Julián David Arango Rodríguez

Instituto Tecnológico Metropolitano
Facultad de Ingenierias
Medellín, Colombia
2019

Predicción de interacciones proteína-proteína mediante un método basado en aprendizaje de máquina para el análisis de la proteína NS5A del virus GB tipo C.

Julián David Arango Rodríguez

Requerimiento de tesis para recibir el título de:
Magíster en Automatización y Control Industrial

Director:
Ph.D Juan David Martínez Vargas

Director:
Ms.c Fabián Mauricio Cortés Mancera

Co-director:
Ph.D Jorge Alberto Jaramillo Garzón

Grupo de Investigación:
Automática, Electrónica y Ciencias Computacionales
Instituto Tecnológico Metropolitano
Facultad de Ingenierias
Medellín, Colombia
2019

Agradecimientos

Quiero expresar un profundo agradecimiento al Instituto Tecnológico Metropolitano por brindarme su apoyo por medio de las instalaciones, equipos y espacios que me ayudaron plenamente a llevar a cabo con éxitos este trabajo. A la facultad de Ingenierías y en especial al Grupo de Investigación en Automática, Electrónica y Ciencias Computacionales ya que por medio de su apoyo y docentes se adquirieron los conocimientos apropiados para finalizar el trabajo investigativo. A los profesores de la línea de Inteligencia Artificial: Andrés Castro, Andrés Giraldo, Leonardo Duque, Maria Constanza Torres, Gloria Díaz, Hermes Fandiño por su acogida y su atención en los momentos que fueron necesarios.

A mis compañeros de maestría: Andrés Pérez, Fredy Torres, Andrés Cardona, Carlos Duarte, Irving Solsol, Rubén Fonnegra, Juan Pablo Villegas y Oscar Ossa por su ayuda, enseñanzas y momentos extraclase que hicieron el día a día más ameno.

Finalmente quiero expresar mi mas grande y sincero agradecimiento al Mágister Fabian Cortés, uno de los directores de éste trabajo, por su atención y buena disposición, al Doctor Juan David Martínez que a pesar del poco tiempo de contacto fue de gran ayuda para la consecución de éste trabajo y al Doctor Jorge Jaramillo quien con su direccion, conocimiento, ensenanza, colaboración y buena disposición, hizo que además de concluir mi pregrado, me motivara a realizar y concluir mis estudios de posgrado.

Dedicatoria

Este trabajo está dedicado a la memoria de mi madre quien siempre estuvo orgullosa de mi y deseó verme como un profesional. A mi padre quien siempre estuvo a mi lado con mucho apoyo y a mi hermana Sandra quien ha sido y es incondicional en mi vida. A mi sobrina Dayana por su admiración y finalmente a la familia Rodríguez por el apoyo y ejemplo para mi motivación.

Resumen

La predicción de interacciones proteína-proteína ha sido una herramienta importante para demostrar la causa de una gran cantidad de enfermedades en los seres vivos. Para tal fin, se destacan los métodos basados en aprendizaje de máquina, resaltando aquellos en los cuales se pueden extraer la mayor cantidad de características. Estos métodos, a pesar de que pueden procesar una gran cantidad de datos en un tiempo más corto en comparación de los métodos físicos, pueden tardar una cantidad considerable de tiempo, además de que el uso de funciones kernel no son habitualmente optimizadas. Por tal motivo, en los últimos años se han desarrollado metodologías basadas en aprendizaje de máquina basadas en kernels con el fin de aumentar el rendimiento de las predicciones. En el transcurso del siguiente documento, se desarrolla una metodología de aprendizaje de máquina con múltiples kernel acompañada de un ajuste de parámetros por medio de programación cuadrática y optimización metaheurística, donde se extraen las interacciones positivas y negativas, luego se filtran las secuencias con repecto a su homología por medio de una herramienta llamada CD-HIT cumpliendo con un porcentaje de homología no mayor al 90 % . La caracterización consiste en el cálculo de frecuencias de aminoácidos que coinciden en características físico-químicas descritas en la base de datos AAindex. La implementación consiste en una combinación lineal que incluye hasta 10 kernels que cumplen con condiciones específicas en cuanto a los pesos, los cuales se encuentran por medio de la optimización cuadrática y que resultan en una matriz final a partir de la secuencia inicial de kernels. Finalmente se realiza la clasificación teniendo en cuenta la optimización por enjambres de partículas para sintonizar el parámetro C. Como resultado, se obtienen resultados consistentes y competentes con respecto a los predictores existentes en la literatura actual ya los supera en algunos indicadores, para lo cual en el presente documento se evidencian rendimientos que se encuentran alrededor del 80 %. Por lo anterior, se puede afirmar que el aprendizaje por múltiples kernel y la optimización de parámetros puede mejorar notablemente el clasificador para el caso de la predicción de interacciones proteína-proteína.

Palabras clave: Interacciones proteína-proteína, Máquina de vectores de soporte, Aprendizaje por múltiples kernel, Optimización metaheurística, Optimización heurística. .

Abstract

Prediction of protein-protein interactions has been an important tool show the cause of a large number of diseases in living beings. For it, the methods based on machine learning are highlighted, and those in which the greatest number of features can be extracted take advantage among others. Those methods, can process a large amount of data in a shorter time compared to physical methods. For this reason, methodologies based on machine learning have been developed in recent years in order to increase the performance of predictions. In the course of the following document, a machine learning methodology with multiple kernels is shown, together with an adjustment of

parameters by means of quadratic programming and metaheuristic optimization. These experiments are initially based on the extraction of interactions from the database called DIP (Database of interacting proteins), based on the sequences with respect to their homology are filtered through a tool called CD-HIT and continue to the next stage. The characterization consists in the calculation of amino acid frequencies that coincide in physical-chemical characteristics described in the AAindex database, and as such, taking into account that the amino acid sequences are comprised between 20 amino acids, 7 different groups are formed with respect to your properties. The implementation consists of a defined linear combination that includes up to 10 kernels that meet specific conditions that result in a final matrix of the initial sequence of kernels. As a result, consistent and valid results are obtained with respect to the existing predictors in the current literature, therefore, this document evidences performances around 80 %. Thus, It can be stated that multiple kernel learning and parameter optimization can significantly improve the classifier for the prediction of protein-protein interactions.

Keywords: Protein-protein interactions, Support vector machines, Multiple kernel learning, Metaheuristic optimization, Heuristic optimization

Contenido

1 Introducción

La predicción de interacciones proteína-proteína ha sido una herramienta importante para demostrar la causa de una gran cantidad de enfermedades en los seres vivos. Para esto, existen los métodos físicos ejecutados en laboratorios, por lo cual suelen ser complejos ya que consumen mucho tiempo y suelen tener ciertas limitaciones [1]. Por otro lado, existen los métodos computacionales, donde se destacan los métodos basados en aprendizaje de máquina, resaltando aquellos en los cuales se pueden extraer la mayor cantidad de características. Estos métodos, pueden procesar una gran cantidad de datos en un tiempo más corto en comparación de los métodos físicos, aunque para ello se requiere de una amplia cantidad de datos positivos y negativos con el fin de entrenar la máquina correctamente. Con respecto a los métodos basados en aprendizaje de máquina, los métodos cuyo principio son los kernels, son usados para que puedan mapear los datos a un espacio de mayor dimensión, lo que quiere decir que ayuda a trazar una mejor frontera de decisión ya que los datos pueden ser más separables. Por otro lado, se hace necesario controlar cada uno de los pesos en los kernels a combinar, ya que cada uno debe tener un aporte específico en una combinación lineal, por tanto se usó una optimización cuadrática para definir cuánto peso debe darse a cada una de las matrices y finalmente, es necesario realizar la optimización de los parámetros de los clasificadores tales como el parámetro C en Máquinas de Vectores de Soporte (Support Vector Machines - SVM's) ya que se ha vuelto primordial ya que se ha demostrado a través del tiempo que el uso de parámetros predeterminados pueden afectar el desempeño del clasificador.

En la literatura se pueden encontrar problemas relacionados con aprendizaje por múltiples kernel de acuerdo al problema de clasificación y al tipo de datos que se usan, por lo tanto es importante aclarar que están estrechamente relacionados con el problema específico. A pesar de la existencia de trabajos realizados en el área de Aprendizaje por múltiples kernels (Multiple Kernel Learning - MKL), se pueden encontrar artículos relacionados con el problema de interacciones entre proteínas donde la sintonización de parámetros se hace confusa, por tanto el experimento no se hace completamente reproducible a pesar de los buenos resultados que tiene [2], además de este, existen experimentos basados en la predicción de localización subcelular [3] donde se implementa el uso de múltiples kernels y que evidencia el problema de optimización de parámetros, finalmente el problema basado en la predicción de interaccion entre fármacos donde la sintonización de los pesos es hecha de modo predeterminado [4]. Por lo anterior, a pesar de existir una gran variedad de métodos computacionales para la predicción o exploración de interacciones, es imprescindible obtener un espacio en el cual las muestras puedan ser más separables por medio de la aplicación de funciones kernel que permita al clasificador trazar su frontera de decisión de una manera más

efectiva [4]. Por otro lado, no se evidencia en el estado del arte el uso de técnicas de optimización en la predicción de interacciones, lo que hace que el ajuste de parámetros sea netamente empírico y de esa manera, los resultados pueden verse afectados directamente. Dicho esto, se puede asumir que el uso de aprendizaje por múltiples kernels puede mejorar el desempeño de los predictores de interacciones entre proteínas basados en máquinas de vectores de soporte,ya que debido a la combinación lineal de matrices se llega a una sola compuesta por todas las anteriores con un peso específico dependiendo del aporte, seguidamente, la sintonización de parámetros heurística (Para los pesos que involucran cada matriz) y metaheurística (Para el parámetro C de la máquina de vectores de soporte) se usan para obtener los parámetros óptimos que podrían aumentar el rendimiento de la máquina. De esta manera, se propone un predictor de interacciones proteína-proteína mediante máquinas de vectores de soporte usando una sintonización de parámetros heurística y metaheuristica en conjunto con el aprendizaje basado en múltiples kernels orientado a la predicción de interacciones entre proteínas.

1.1. Motivación

El virus GB tipo C (GBV-C) es un virus linfotrópico, y diferentes estudios han planteado que la infección por GBV-C puede estar relacionada con el desarrollo de enfermedades linfoproliferativas al igual que el virus de la hepatitis C (HCV - Hepatitis C virus), como lo es el linfoma Hodgkin y no Hodgkin, sugiriendo que este virus puede estar relacionado con un mayor riesgo de desarrollar estos tipos de cáncer [5]. Desde otro punto de vista, han demostrado que la infección por GBV-C favorece el pronóstico clínico de individuos infectados con el virus de la inmunodeficiencia humana (HIV - Human Immunodeficiency Virus), y se observa un estado de mayor supervivencia en pacientes con coinfección, esto se debe a que ambos virus se replican en la misma célula blanco, y como consecuencia, demuestra que GB-C puede bloquear la replicación del HIV [6]. A raíz de ello, se ha propuesto que los estudios donde se estudien las proteínas del virus GBV-C, y más específicamente con la proteína NS5A y proteínas que tengan directa relación con procesos de proliferación celular, podrían establecer si existe una potencialidad oncogénica de tal virus. Todo ello se hace con el fin de controlar su transmisión, y a un futuro poder realizar pruebas de tamizaje que verifique cuando un paciente posee el virus en cuestión.

Del mismo modo, El virus GBV-C es un virus de transmisión parenteral que varía su prevalencia en donantes de sangre de acuerdo a la región geográfica. En países desarrollados se registran prevalencias entre 2-5 %, mientras que en los países en vía de desarrollo se reconoce que oscilan del 15-18 % como es el caso de los países africanos y latinoamericanos [7]. Se estimó que del 6-14 % están coinfectados con el virus de la Hepatitis B (HBV) y con el HCV hasta en un 20 % aproximadamente, seguidamente se han identificado prevalencias mucho mayores en individuos infectados con el virus del HIV que pueden variar entre 15 % -50 %, lo que indica un factor importante incidente en la coinfección con ambos virus [8].

En Colombia, los estudios acerca del GBV-C han sido muy pocos, en 1998 se reportó un estudio en el cual se describe una prevalencia del 1.5 % en donantes de sangre, y una prevalencia cercana al 8 % en poblaciones indígenas [9]. Un estudio más reciente reporta una prevalencia del 3.2 %y el 5.06 % en donantes de sangre infectados con HCV y HBV respectivamente; y una prevalencia del 7.7 % en población indígena de la región Amazónica [10], la escasez de estudios acerca de éste virus se debe a que no se le atribuía relación con ninguna patología. Este virus en específico tiene la característica de tener cierto lapso de persistencia dentro de un organismo, los análisis inmunológicos muestran que la mayoría de los individuos desarrollan anticuerpos protectores contra la glicoproteína E2 lo cual permite el aclaramiento de la infección un período no menor de dos años [11].
Por otro lado, las interacciones entre proteínas como contactos físico-químicos entre un par de moléculas tienen la capacidad de efectuar un cambio dentro de la célula, lo que usualmente produce efectos que se le atribuyen a funciones celulares [12]. Asimismo, como se ven comprometidas las funciones celulares, también puede verse involucrada toda la maquinaria celular. Al momento de efectuarse una interacción no conveniente, puede dar paso al origen de un cáncer por una anomalía en tal sistema, a lo cual se le debe adjudicar a la inhibición de una interacción predeterminada genéticamente [13]. El descubrimiento de posibles proteínas que interactúen con una proteína en específico, puede elucidar vías de señalización no descubiertas, inhibidores específicos en una ruta, o posibles fármacos que puedan detener una interacción no conveniente para el organismo [14]. Seguidamente, cabe mencionar que los métodos físicos para el descubrimiento de interacciones entre proteínas pueden llegar a ser costosos, y pueden consumir una gran cantidad de tiempo, con la condición de que sólo puede verificarse una a una variando la confiabilidad de un método en otro [15].

Por lo anterior, puede decirse que las interacciones proteína-proteína no solo han adquirido importancia en los últimos 10 años ya que por medio de su predicción se pueden descubrir efectos adversos en los seres vivos, ya sabiendo que las proteínas virales pueden contar con mecanismos para entrar a una célula humana, pueden modificar por medio de su modo de operación replicativo, pueden afectar el mecanismo de funcionamiento normal en una célula, dando lugar desde trastornos leves hasta enfermedades de gran magnitud [16]. Asimismo, al saber de una interacción específica en la cual se están obteniendo efectos adversos, se puede proceder a realizar estudios de inhibidores de interacciones realizados in vivo e in vitro [17].

Para el orden del fortalecimiento del departamento de Antioquia y las políticas de promoción en salud presentadas los últimos años, es conveniente plantear proyectos que tengan que ver con el diagnóstico y control de patologías desconocidas, ya que esto puede llevar a la implementación de planes de contingencia que incluyan el control de la transmisión del virus GBV-C en caso de tener relación con proteínas que tengan propiedades oncogénicas. Por lo tanto, este método para la predicción de interacciones proteína-proteína, puede contribuir al desarrollo del clúster de

servicios de medicina en cuanto a sus puntos estratégicos de gestión de conocimiento en conjunto con el punto relacionado con innovación y desarrollo, ya que coopera directamente en el desarrollo de nuevas estrategias para el diagnóstico, prevención y tratamiento de patologías.

1.2. Estado del arte

En cuanto al problema de predicción de interacciones proteína - proteína basado en aprendizaje de máquina, ha obtenido resultados positivos en los últimos años, y por otro lado, el avance de las metodologías basadas en múltiples kernels ha permitido incursionar en temas en los cuales el limitante era una sola función tal y como aún se usan muchas de las SVM u otros algoritmos de clasificación, lo cual hace que se limite a darle un peso específico a una sola función. De este modo, se ha generado mayor impacto con resultados mejor posicionados con respecto al uso de kernel simple.

En la literatura se pueden encontrar funciones kernel de acuerdo al problema de clasificación y al tipo de datos que se usarán, por lo que es importante aclarar que están estrechamente relacionados con el problema específico, y por lo tanto no es recomendable usar aleatoriamente una función kernel determinada ya que causalmente llevaría a errores de clasificación. Aunque a pesar de tener trabajos en el área, no se registra concretamente un trabajo que se ubique dentro del análisis de múltiples kernel para la predicción de interacciones entre proteínas, y solo se destaca de manera relevante el trabajo realizado A continuación se realiza una breve explicación de la implementación y tipos de kernels usados por cada uno de los siguientes trabajos realizados por autores que se han resaltado en el desarrollo de herramientas para la predicción:

1.2.1. Localización subcelular de proteínas usando aprendizaje por múltiples kernels

Para este articulo [3] se realizan una clasificación por medio de máquinas de vectores de soporte (SVM's), en la cual se realiza de modomulti-etiqueta. Para este caso, se especifica la forma en la cual se realiza la combinación lineal de las funciones kernel, de modo que es fácil distinguir como lo hacen en su metodología:

$$\mathbf{K}_n = \sum_{m=1}^{P} n_m \mathbf{k}_m(\mathbf{x}_i^m, \mathbf{x}_j^m) \tag{1-1}$$

En la ecuación 3-13 n se comporta como los pesos, y el argumento toma P representaciones de caracteristicas. Del mismo modo, se debe agregar la descripción del kernel de base radial, el cual no se encuentra descrito en el trabajo expuesto y representa cierta importancia en la combinación lineal debido a que constantemente se hace una referencia al parámetro σ que se encuentra variante en la combinación lineal y cuya descripción aparece en la ecuación 1-2:

$$K(x, x^{'}) = exp(\frac{\left\|x - x^{'}\right\|^2}{2\sigma^2}) \tag{1-2}$$

El articulo realiza una mención acerca de la implementación realizada por ellos mismos, en la cual indican el proceso que debe seguirse en términos generales para poder recrear el procedimiento:

- Discretizar el parámetro sigma del kernel de base radial en 17 valores desde 2^{-8} hasta 2^{8}.
- Buscar los pesos de cada kernel respectivamente, teniendo en cuenta que son 17 valores del mismo modo.
- Realizar la combinación lineal de los 17 kernels con respecto a la ecuación 1-2 donde la variable m hace referencia al número de kernels.
- Entrenar las SVM's respectivamente para cada ubicación celular (Para este estudio son 9).
- Realizar la predicción.

Para este estudio obtuvieron en puntaje $F1$ aproximadamente $0{,}72$ y una precisión de $0{,}69$, donde lo comparan con otros predictores multi-etiqueta, debido a que es un problema donde una muestra puede pertenecer a varias localizaciones. Adicionalmente se realiza una comparación entre los resultados obtenidos para múltiples kernels, un solo kernel implementado por los mismos y otro tomado de la literatura donde sus implementaciones tanto en MKL (Multiple kernel learning) como en SG (Single kernel) tienen rendimientos de $0{,}81$ en puntaje $F1$ y $0{,}77$ en precisión. Adicionalmente,presentan resultados relacionados al tiempo de cómputo donde muestran la diferencia entre el método MKL con 30 horas, mientras que el SG en su implementación y en la existente en la literatura con alrededor de 75 horas.

1.2.2. Predicción computacional de interacciones proteína-proteína entre virus-humano

Primeramente, se debe esclarecer que el uso de ciertos parámetros en este artículo es confuso, esto se debe a que no hay una especificación de los parámetros de la máquina.
Del mismo modo, y como el estudio descrito en la subsección anterior, utilizan el kernel de base radial, con la diferencia que lo utilizan en 3 modos: Utilizan una función de pérdida en la cual simultáneamente se trata de aproximar S_v, S_h y S_{vh}, variables que estan directamente relacionadas con un criterio definido como real, por lo tanto será una aproximación por distancias, de modo que en la formulación, la matriz kernel se le restará la similitud [2]:

$$\ell = \sum_{\phi vh}(\mathbf{K}_{vh} - \mathbf{S}_{vh})^2 + \sum_{\phi vh}(\mathbf{K}_{v} - \mathbf{S}_{v})^2 + \sum_{\phi vh}(\mathbf{K}_{h} - \mathbf{S}_{h})^2 \tag{1-3}$$

Por otro lado, se especifica que son utilizados 4 spectrum kernels a modo de producto punto, donde los pesos son distribuidos de manera equitativa en la sumatoria. Este kernel está representado por:

$$\mathbf{K}(x, y) = \langle \phi_k(x)\phi_k(y) \rangle \tag{1-4}$$

En la ecuación 3-12 se describe el producto punto del ϕ_k kernel con respecto a otro ϕ_k kernel, teniendo en cuanta que son 4 En este estudio se verificó un $50\,\%$ de coincidencias en las interacciones que fueron validadas experimentalmente (Comúnmente llamadas curadas) y que estan directamente relacionadas con el HCV. Adicionalmente se realizó una prueba que tenía que ver con los datos filtrados, en los cuales encuentran una mejoría en los resultados, debido a que encuentran alrededor del $95\,\%$ en coincidencias con interacciones verificadas de modo experimental.

1.2.3. Algoritmo de múltiples kernel para predicción de interacción entre fármacos-objetivos

En éste artículo, se expone una mejora el algoritmo de Kron-RLS (Kronecker kernel RLS for pairwise data) [18] a manera de mejora [4] , en donde lo que se plantea fundamentalmente es una máquina de vectores de soporte cuyo fin es encontrar un equilibrio en el que se pueda minimizar el error en la predicción con respecto a la complejidad del modelo. Con respecto a la función de minimización de riesgo estructural, parte también de la representación dada el primer artículo expuesto en esta sección.
Para la definición del pairwise kernel, es necesario mencionar que es una construcción basada en el producto de dos kernels, con un kernel específico $\mathbf{K}_T$ correspondiente a los respectivos objetivos, este kernel en específico es descrito como el producto Kronecker, sin embargo, se describe una ventaja de algoritmo KronRLS sobre esta clase de producto debido a la velocidad que adquiere en el modelo de entrenamiento [4].

Por otro lado se especifica un trabajo realizado con respecto a tres tipos de características: Secuencia de aminoácidos, anotación funcional y proximidad en la relación de interacción proteína-proteína. Con respecto a la secuencia se considera el puntaje normalizado de Smith-Waterman, por otro lado, se utilizan diferentes parametrizaciones con respecto al mismatch y al spectrum kernel. Para el mismatch se evaluaron 4 combinaciones de distintos valores de longitud y el número de máxima incopatibilidad.

Por otro lado, en el artículo se hace una descripción metodológica de el trabajo realizado, donde se destaca una validación cruzada de 5 particiones repetida por 5 ocasiones, aquí cabe aclarar que la sintonización de los pesos son hechos por modo predeterminado, por lo tanto puede haber cierta aleatoriedad en los resultados ya que puede considerarse un paso importante para el desarrollo de una buena clasificación. El procedimiento utilizado fué el siguiente:

- Se simula un escenario donde se pretende tener un nuevo medicamento. En este escenario se

parte la base de datos en 5 partes donde 4 son usadas para el entrenamiento y 1 parte para la predicción.

- Escenario: Predicción de la interacción de los medicamentos para nuevos objetivos. Es similar al escenario anterior, sin embargo,se consideran 5 particiones para objetivos.
- Predicción de pares: Consiste en predecir interacciones entre medicamentos y objetivos que no han sido conocidas o registradas en la literatura. Las particiones también se realizan idénticamente a los puntos anteriores. Aunque hay que tener en cuenta que en algunos casos se debió realizar un sub-muestreo con el fin de equilibrar las muestras ya que son más la cantidad de interacciones negativas que las positivas.

En este estudio se infiere que cuando a los pesos de los kernels se les asigna números altos, se puede aumentar la calidad en los resultados, aunque solo sea una especulación de del estudio ya que no tienen evidencias claras de ello. Además, se hacen experimentos con kernels empleados de modo individual, en el que se señala que pueden funcionar similar a los MKL en situaciones muy específicas, por lo cual están directamente relacionadas con el problema específico.

1.3. Planteamiento del problema

A pesar de existir una gran variedad de métodos basados en aprendizaje de máquina para predicción de interacciones proteína-proteína, el rendimiento de estos predictores se ha visto comprometido por el uso de un espacio menos separable en ciertos casos. Esto se debe a que la selección de los parámetros de un kernel, aparte de ser un punto crucial, afectan la clasificación de modo directo debido a que en ocasiones el modo predeterminado o la selección aleatoria de tales parámetros confluye en una variación alta de los resultados. Usualmente se pueden percibir problemas en la literatura que son abordados por metodologías basadas en múltiples kernel, sin embargo no corresponden directamente a la predicción de interacciones entre proteínas, y aunque se evidencia frecuentemente y cada vez más el uso de la metodología por múltiples kernel, es común ver que prescinden de los métodos de optimización para acompañar a sus parámetros. Por otro lado, aunque estos clasificadores mantienen el nivel de desempeño alrededor del $80\,\%$, y a pesar de que estos resultados son aceptables, la gran mayoría de predictores presentan rendimientos similares o menores a éste, ya que las muestras puedan estar sujetas a un espacio de características en el cual no es suficientemente amplio para una clasificación con un mejor rendimiento. Para tener un mejor espacio de características es necesario hacer la selección de una o varias funciones kernel que aumente no solo espacio de características, sino que permita al clasificador trazar su frontera de decisión de una manera más efectiva. Además, se debe agregar que a pesar de la amplia gama de predictores publicados en los últimos años, existe una convergencia en el uso de pocas características, lo que hace limitar a cualquier predictor, debido a que las interacciones no se dan sólo de acuerdo a una característica específica sino por la confluencia de un gran número de las mismas. Por otro lado, no se evidencia de manera frecuente el uso de técnicas de optimización

en la predicción de interacciones proteína-proteína, lo que hace que el ajuste de parámetros sea netamente empírico y de esa manera, los resultados pueden verse afectados directamente, lo que implícitamente sugiere un conocimiento previo de las principales técnicas de optimización antes del diseño o implementación de un predictor.

1.4. Hipótesis

Debido a la existencia de diferentes kernels y a la dificultad en la selección de los parámetros para los mismos, se puede asumir que por medio del uso de aprendizaje por múltiples kernels puede mejorar el desempeño de los predictores de interacciones proteína-proteína basados en máquinas de vectores de soporte, esto se debe a que no se implementaría un kernel con ciertos parametros, sino que se emplearía un conjunto de kernels con buenos rendimientos en el problema de clasificación, del mismo modo se optimizarían los pesos de cada kernel mediante un método de optimización metaheurístico con el fin de obtener un predictor óptimo para predicción de interacciones en proteínas virales y así mejorando el rendimiento general del mismo.

1.5. Objetivos

1.5.1. Objetivo General

Proponer un predictor de interacciones proteína-proteína mediante SVM usando una sintonización de parámetros metaheurística y aprendizaje basado en múltiples kernels orientado al análisis de proteínas virales.

1.5.2. Objetivos Específicos

1. Diseñar una estrategia basada en aprendizaje por múltiples kernels en la cual se combinen diferentes tipos de funciones mediante una estrategia de optimización metaheurística, que permita encontrar el conjunto óptimo de pesos.

2. Integrar la estrategia basada en múltiples kernels sobre una máquina de vectores de soporte que permita hacer una selección de las características más discriminantes y al mismo tiempo escoger el conjunto de hiperparámetros óptimos para el modelo de clasificación.

3. Validar el clasificador mediante la extracción de medidas de desempeño propuestas para los predictores de interacciones proteína-proteína, con el fin de realizar un análisis comparativo del rendimiento de la metodología propuesta y los métodos de predicción actualmente utilizados en la literatura en el área de predicción de interacciones proteína-proteína.

2 Marco Teórico

En éste capítulo se realizarán las descripciones teóricas de los conceptos fundamentales para poder entender la tesis, es decir, se hará un breve recorrido por la parte biológica y se hará énfasis en los conceptos relacionados con aprendizaje de máquina y múltiples kernels.

2.1. Conceptos biológicos

2.1.1. Proteínas

Las proteínas son moléculas presentes tanto en organismos vivos como no vivos tales como: Humanos, animales, bacterias y virus, entre otros. Éstas están conformadas por pequeñas unidades llamadas aminoácidos, por lo tanto, una cadena de aminoácidos con determinada longitud y plegamientos puede considerarse una proteína y al mismo tiempo un polímero. Es así como los aminoácidos se unen uno al otro por medio de enlaces peptídicos por medio de un extremo $-COOH$ y otro NH_2 terminal, de modo que, un extremo amino, se une con otro extremo carboxilo con el fin de liberar una molécula de H_2O en la reacción [19], el correspondiente a R quiere decir el radical respectivo que finalmente será el que le da la distinción y la propiedad química diferenciadora.

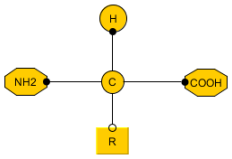

Figura 2-1: Estructura general del aminoácido.

De esta forma, como se describe en la imágen 2-1 cada uno de los aminoácidos que se adhieren a la cadena son ubicados al extremo derecho o izquierdo de la estructura. La diferencia de una proteína a otra radica en la secuencia de tales aminoácidos que conforman la cadena de caracteres, cada uno de ellos es diferenciado por su radical ubicado en la parte inferior de la molécula. Para su clara diferenciación, se pueden encontrar 20 aminoácidos que se pueden agrupar según su propiedad química más importante como la carga o solubilidad en agua [20]. Seguidamente, en las proteínas se pueden identificar cuatro estructuras importantes que consisten en una serie de etapas que las proteínas (Conformadas por una cadena de aminoácidos) por las cuales deben pasar para conseguir

una funcionalidad plena, estas estructuras son descritas a continuación basadas en la descripción de Creighton [21]:

- **Estructura primaria:** Determinada por la cadena de aminoácidos, es decir, por una cantidad específica moléculas (Caractéres) y por un orden específico de las mismas, lo cual diferencia una proteína de otra. Esta estructura se da como consecuencia de los enlaces peptídicos mencionados anteriormente, en la figura **2-2** se puede apreciar la adherencia aminoácido-aminoácido correspondiente y según lo expuesto en la teoría de proteínas.

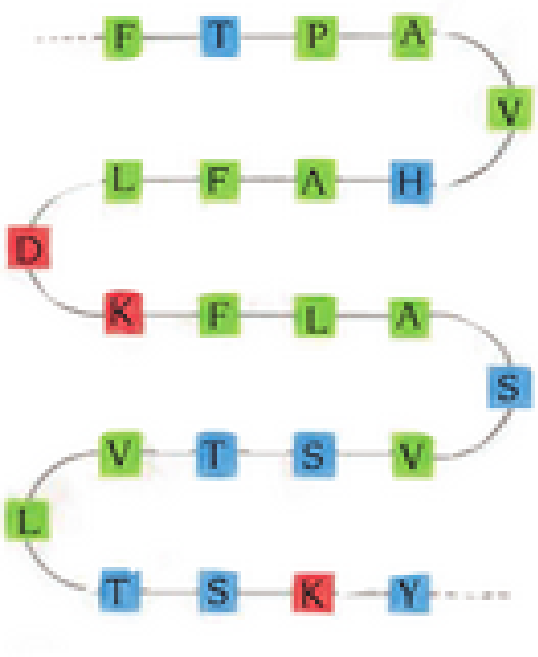

Figura 2-2: Estructura primaria de las proteínas

- **Estructura secundaria:** Luego de tener una cadena de aminoácidos en su estado lineal, tales moléculas deben sufrir plegamientos por causa de fuerzas intermoleculares como puentes de hidrógeno, puentes disulfuro e interacciones de Van der Waals, entre otras. A partir de estos plegamientos se generan las hélices alfa y las láminas beta. En la figura **2-3** se puede apreciar el doblamiento en forma de hélice por cuenta de los enlaces e interacciones.

- **Estructura terciaria:** En la figura **2-4** se puede apreciar la unión de los doblamientos que dan como resultado una estructura terciaria, en este tipo de estructura se comienza a forjar la estructura tridimensional que es la que le aporta las propiedades biológicas ya que al tener determinada posición sobre el espacio, puede proporcionar cierta disposición para la interacción con otras proteínas.

Figura 2-3: Estructura secundaria de las proteínas [22]

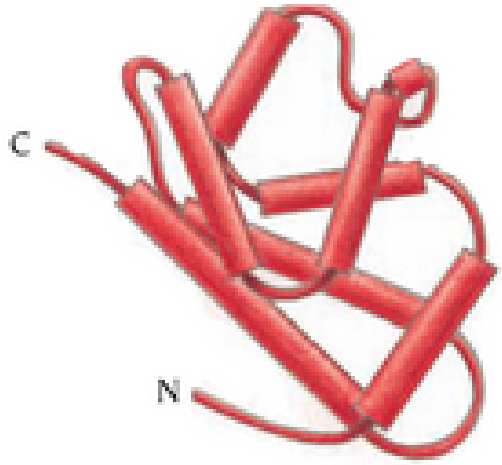

Figura 2-4: Estructura terciaria de las proteínas [22]

- **Estructura cuaternaria:** Las proteínas en este tipo de estructura se encuentran unidas con otros péptidos, esta disposición es denominada oligomérica. Para la ejecución de tal asociación es necesario que entre los monómeros exista un enlace no covalente y que exista una

relación directa entre las propiedades de cada uno de ellos. En la figura **2-5** se puede apreciar la unión de 4 estructuras terciarias formando una sola estructura cuaternaria y funcional.

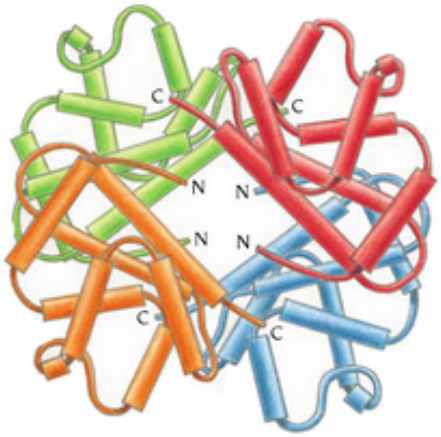

Figura 2-5: Estructura cuaternaria de las proteínas [22]

En la imágen **2-6**, se muestra la estructura $3D$ obtenida por cristalografía de rayos X y descargada de la base de datos "Protein Data Bank", donde se pueden encontrar una gran cantidad de proteínas en su estado funcional o ubicadas en interacción con ligandos. En la estructura terciaria como la apreciada en la figura anterior, la proteína puede ya cumplir funciones como interacciones y participación en rutas metabólicas donde son mayormente importantes.

2.1.2. Interacciones

En la gran mayoría de ocasiones, las proteínas deben asociarse con otras proteínas para desencadenar funciones biológicas en un organismo. Delmismo modo, las proteínas también pueden crear interacciones con moléculas que no son de su misma clase como es el caso de los ácidos nucléicos, carbohidratos o lípidos, así las proteínas lo realizan con el fin de desencadenar una reacción que puede ser benéfica si hay un correcto funcionamiento en las vias de señalización o destructivo, si existe una anomalía en el organismo.

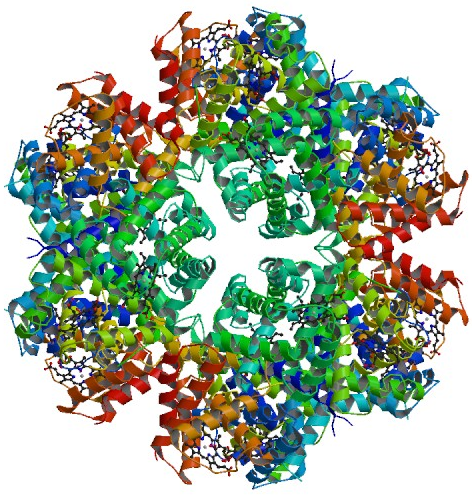

Figura 2-6: Estructura 3D de la hemoglobina extracelular. Tomado de: www.ncbi.nlm.gov (Pubmed)

En términos moleculares, las interacciones entre proteínas deben ser similares a los plegamientos que se realizan para la estructura secundaria como fuerzas de Van der Waals, puentes de hidrógeno o interacciones hidrofóbicas. De este modo, una parte de una proteína realiza contacto con otra proteína específica de acuerdo a criterios físico-químicos o estructurales. En cuanto a los criterios físico-químicos, existen una gran cantidad de características dadas por el orden específico de los aminoácidos en la cadena polipeptídica, esto hace que el potencial de contacto pueda variar de acuerdo a la conformación en su cadena y a su disposición en el espacio. En cuanto a los criterios estructurales, se pueden tener en cuenta los dominios, motivos y los puntos calientes (*Hostspots*) proporcionados por ciertos patrones en las cadenas de aminoácidos y que por tanto son conservados en ciertas familias de proteínas, cuando esto sucede se pueden conservar funciones entre esos patrones de caractéres, tales como puntos de interacción o funciones propias de una proteína como su integración en una vía de señalización específica. En la figura 2-6 se puede apreciar una interacción entre dos proteínas, proporcionado por Protein Data Bank (PDB), allí puede notarse que existe un punto de anclaje entre ambas proteínas y que por ende puede existir una interacción por fuerzas intermoleculares [12].

2.2. Predicción de interacciones proteína-proteína

Las interacciones entre proteínas juegan un papel fundamental dentro de la organización celular en cuanto a su estructura y función, estas interacciones pueden darse entre dominios de la misma proteína, dominios de diferentes cadenas polipeptidicas, y entre complejos de proteínas independientes. Del mismo modo, la investigación de las interacciones que se puedan producir dentro de un organismo, pueden llegar a dilucidar el método por el cual aparecen las anomalías en un sis-

tema determinado, ya que los organismos están gobernados por interacciones entre sus vías de señalización [12] [23]. Dicho esto, la predicción de interacciones entre proteínas juegan un papel fundamental en el conocimiento de enfermedades en las cuales no se ha dilucidado el mecanismo de acción o la vía de señalización a seguir.

Para el análisis de interacciones entre proteínas, existen una serie de métodos que pueden variar de fiabilidad de acuerdo a su principio de acción ya que existen métodos directos, indirectos, y celulares, tales como: Ensayos doble híbrido, Espectrometría de masas, co-inmunoprecipitación, Microscopía electrónica, Ensayos knock-out y Rayos X, por lo tanto, en la gran mayoria de ocasiones hoy por hoy, se hace necesaria la ayuda que puede brindar la bioinformática por medio de los métodos de predicción [24].

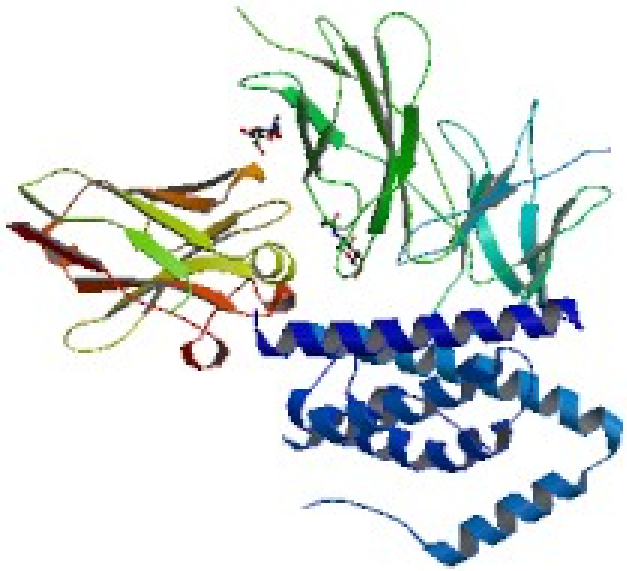

Figura 2-7: Estructura $3D$ de interacción entre anticuerpo y epítope. Tomado de: www.ncbi.nlm.gov (Pubmed)

Para la predicción de interacciones entre proteínas existen diferentes tipos de principios que se pueden tener en cuenta en el momento de caracterizar las proteínas en cuestión, a continuación se presentan algunos principios utilizados para la predicción de interacciones:

- **Perfiles filogenéticos:** Comprende el análisis desde que se comenzaron a construir los primeros árboles filogenéticos, que se hacían con el fin de descubrir la relación que podía guardar

un organismo con otro, y así podía explicarse la similaridad en cuanto a comportamiento y fenotipo [25]. Estos métodos eran basados en alineamientos de secuencias y muestran las primeras influencias del reconocimiento de patrones en la biología, luego se reconocerían los primeros dominios gracias a la conservación de ciertos fragmentos [26], y más adelante se descubriría el papel de los dominios y los Hotspots dentro de una proteína en el papel que desempeñan en las interacciones [23].

- **Patrones estructurales:** En este método se pretenden identificar los patrones de la secuencia en las cuales se tiene una probabilidad más alta de interacción con otras moléculas, esto se logra mediante la adquisición de una base de datos en la cual se especifiquen las secuencias para así buscar los patrones en cada secuencia. Estos métodos nacen a raíz del análisis de las mismas secuencias, pero esta vez no para identificar los dominios sino para distinguir las variaciones en la secuencia y los patrones que efectúan una interacción, como la estructura primaria, secundaria, terciaria y cuaternaria de una proteína, asimismo considerando sus transiciones [27], lo que puede mostrar claramente cuales fragmentos específicos son los potenciales interactores.

- **Métodos basados en aprendizaje de máquina y estadística bayesiana:** Este tipo de método incluyen por ejemplo las redes neuronales, que han sido utilizadas para la predicción de las interacciones hospedero-patógeno, en estas se debe tener en cuenta se debe realizar una sintonización efectiva entre el número de capas y neuronas. Por otro lado, la estadística bayesiana considera una hipótesis que se debe validar a partir de datos obtenidos a manera de evidencia. Para ello, es necesario el cálculo de probabilidades a partir de una referencia o base de datos [28].

2.2.1. Aprendizaje de máquina

Máquinas de vectores de soporte

Las máquinas de vectores de soporte (SVM) surgen como resultado de investigaciones en teoría del aprendizaje estadístico para la clasificación binaria, realizadas alrededor de los 90 [29], de esta manera se comenzaron a explicar problemas de clasificación de dos clases, aunque el método comenzó a migrarse a problemas de multietiqueta y agrupamiento donde han tenido una buena relevancia para diferentes aplicaciones referentes a clasificación de caracteres, proteínas y visión artificial entre otras. Las máquinas de vectores de soporte están definidas como un clasificador tipo estadístico para problemas de tipo binario, e incluidas dentro de los separadores lineales, cuya finalidad consiste en encontrar un hiperplano que pueda llegar a separar dos clases especificas a partir de unas muestras de entrada, resultando muy apropiado en problemas con atributos numéricos. Los datos pueden ser linealmente separables (Caso ideal) o no separables, cuando sucede esto, es necesario realizar una transformación a un espacio de caracteristicas donde debido a su alta dimensionalidad debe aplicarse una función Kernel.

Las SVM están enfocadas en minimizar el riesgo estructural, mapeando primero los puntos de entrada en un espacio de características en grandes dimensiones, todo ello con el objetivo de encontrar el hiperplano que separe y maximice el margen entre la línea que separa las muestras y los vectores de soporte, para separar las dos clases, se debe tener una máxima distancia entre los dos puntos más cercados al hiperplano de separación, por lo tanto estas muestras serán importantes en la clasificación ya que serán los vectores de soporte que definen el márgen mínimo [30]Èl hiperplano de caracteristicas toma la forma:

$$f(x) = \langle \mathbf{w}^T \mathbf{x} + b \rangle \tag{2-1}$$

En la ecuación 2-1 se puede apreciar el vector de pesos denotado como $\mathbf{w}$, también se puede evidenciar la presencia del vector de entrada denotado como $\mathbf{x}$, por último se encuentra el $\mathbf{b}$ llamado bias, el cual está directamente relacionado con el parámetro C ya que cuando éste último es más grande, el bias tiende a ser menor pero con mayor varianza. Al formarse el hiperplano que puede separar las muestras, *N* es la cantidad de las muestras, y *d* hace referencia a las caracteristicas. Cabe realizar la aclaración que esto es aplicable cuando las muestras sean linealmente separables.

Existen dos casos para poder separar las muestras en las SVM, cuando las muestras son linealmente separables, que consiste en que las muestras se encuentran considerablemente separadas y no mezcladas en el hiperplano, de manera que en el momento del entrenamiento se encuentra el hiperplano óptimo y por tanto la margen entre las proyecciones de los vectores de soporte es maximizado, por otro lado existen casos que no son linealmente separables, que permite alteraciones de la clasificación sin penalizarlas en el paso de su formulación. El hiperplano óptimo debe ser entonces reformulado para la solución del problema, ya que en ocasiones no basta con cierto número de dimensiones, sino que debe aumentarse para que puedan ser nuevamente linealmente separables. En estas máquinas de vectores de soporte existe un parametro de regularización C que puede llegar a ser sintonizado , el cual hace un balance entre la maximización del margen y la violación en la clasificación [31].

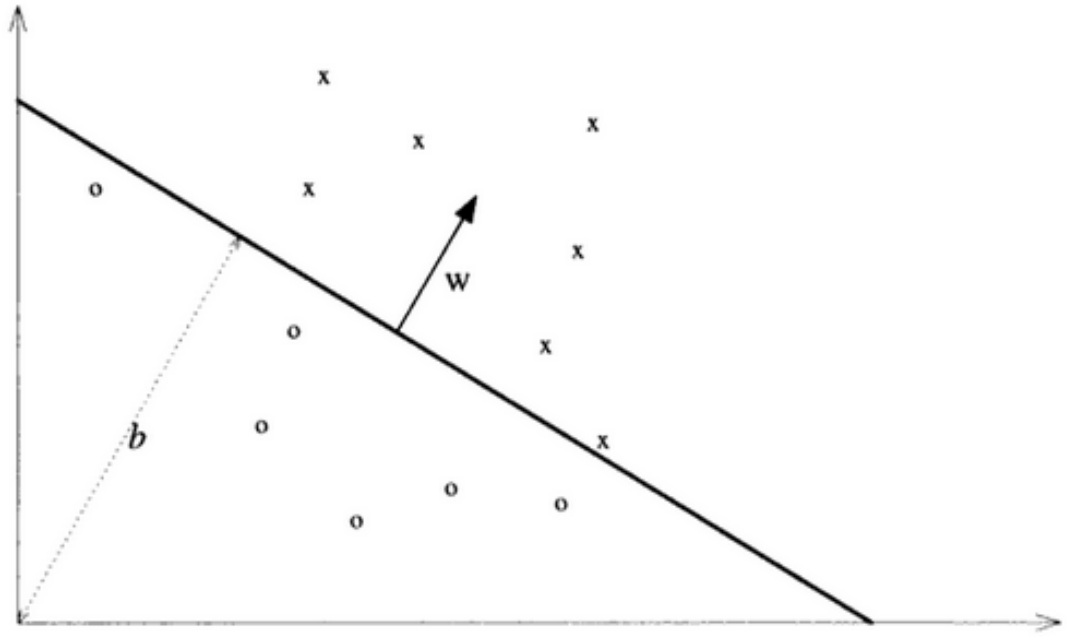

Figura 2-8: Esquema general de las máquinas de vectores de soporte [32]

En la figura 2-8 se puede observar la separación de dos clases con respecto a un espacio de caracteristicas (Hiperplano). Luego de esto, el problema de clasificación pasa a ser un problema de optimización debido a el siguiente planteamiento.

$$min \left\|w\right\|^2 + C \sum_{i}^{N} max(0, 1 - y_i f(x_i)) \tag{2-2}$$

En la ecuación 2-2 $w \in \mathbb{R}^d$ y y_i determinan la pertenencia a cada clase, debido a que cuando pertenece a la clase denotada como $+1$ la función deberá cumplir la condición $w^T x + b \geq +1$, de lo contrario, deberá pertenecer a la clase -1 , con la respectiva función $w^T x + b \leqslant -1$, tal problema de optimización puede resolverse por medio de los multiplicadores de Lagrange donde w se comporta como la combinación de los datos de entrenamiento $w = \sum_{j=1}^{N} \alpha_j x_j y_j$ en el cual si sustituimos y reformulamos los términos.

$$w = \sum_{i} \alpha_i - \frac{1}{2} \sum_{jk} \alpha_j \alpha_k y_j y_k (x_j^T x_k) \tag{2-3}$$

El cual está sujeto a $0 \leqslant \alpha \leqslant C$. Además de ello, los α son considerados los coeficientes de los multiplicadores de Lagrange y sólo los multiplicadores asociados a los vectores de soporte no con cero. Además, el hiperplano no es sensible a cualquier punto clasificado correctamente fuera del margen [33]. Cabe aclarar que una correcta sintonización de el parámetro C puede mejorar significativamente el clasificador, debido a que tiene la capacidad de suavizar o endurecer el márgen para los vectores de soporte y finalmente permitir o no hasta cierta cantidad de error.

Se debe aclarar que no existe un solo hiperplano de separación sino que pueden existir infinitos hiperplanos, por ello se debe tomar en cuenta la optimización del problema, denotado como la minimización de la distancia entre los vectores de soporte y el hiperplano, es decir, las dos muestras mas cercanas, lo cual implicaría una reducción de los posibles hiperplanos. La distancia entre las muestras y los hiperplanos está dada por:

$$\frac{\left|D(\mathbf{x}^{'})\right|}{\|\mathbf{w}\|} \tag{2-4}$$

Donde se describe la norma del vector w donde en conjunto con el parámetro b de la formulación de la máquina, definen el óptimo $D(x)$, lo que traduce que se debe encontrar el valor de w que maximice el margen pero que al mismo tiempo reduzca los hiperplanos posibles.
Para el caso de las muestras que no pueden ser separadas linealmente se debe agregar la definición de una variable de holgura que puede ir de 1 hasta n y que podrán cuantificar la cantidad de muestras admitibles al momento de separar (No lineales).

$$y_i(<\mathbf{w}, \mathbf{x}_i> +b) \geqslant 1 - \epsilon_i \tag{2-5}$$

La ecuación 2-5 demuestra que cuando la variable ε_i toma valor de 0 es que el problema resulta linealmente separable, pero cuando toma valores por encima de 0 es posible que aparte de no ser linealmente separables, también pueda existir un error de clasificación.

2.2.2. Kernels

Para explicar ampliamente el tema relacionado con los kernels es necesario recurrir al Teorema de Aronszajn, el cual dice que para cualquier función definida $K : \mathbb{X} \to \mathbb{R}$ debe ser simétrica y positiva, para la cual debe existir un espacio de Hilbert. Es importante saber que para construir una función kernel basta con tener la función que cumpla las dos condiciones [34]. De este modo, El uso de kernels en las máquinas de vectores de soporte ha impulsado su desempeño ya que amplía el hiperplano de características a uno mayor, de este modo, es necesario acudir a una función kernel para realizar un mapeo por medio de una función $k(x_1, x_2) = \langle \phi(x_1), \phi(x_2) \rangle$ que apunta hacia un espacio de características por medio de un producto punto [35], lo anterior se complementa mostrando la solución de una forma:

$$f_\theta * (X) = \sum_{i=1}^{L} \eta_i K(x, x_i) \tag{2-6}$$

En la ecuación 2-6 se puede notar claramente el peso que aporta el argumento η_i debido a que le da un peso específico al kernel con el fin de darle equilibrio a la proyección, el mismo que cumple un papel importante en una combinación de kernels y que más adelante será descrito. Para representar una función kernel, es necesario tener en cuenta que no es obligatorio realizarlo a partir de cualquier función existente, por lo dicho, puede ser suficiente con definir una función propia que cumpla las condiciones del teorema descrito y que además satisfaga las condiciones del problema.

Luego, para validar el correcto funcionamiento de la función, es necesario evaluar tal función. Para realizar lo anterior (Transformación no lineal) es necesario construir un estimador y los datos de entrada deben ser transformados de modo que tengan una equivalencia en un espacio de mayor dimensión descrito como:

$$x : \mathbb{R}^n x \mapsto \phi(x) : H \tag{2-7}$$

Las transformaciones lineales como la descrita en la ecuación 2-7 pueden permitir una mayor posibilidad de que los datos sean separables mediante un hiperplano, por lo anterior se describe la ecuación anterior con el fin de demostrar la correspondencia de las muestras en un espacio de Hilbert, el cual es un espacio de mayor dimensión donde las muestras pueden adquirir nuevas posiciones. Dentro de las mismas funciones kernels pueden existir parámetros de dispersión que pueden ser optimizados mediante alguna función de optimización cuadrática. Por otro lado, la fusión de kernels consiste en la combinación de diferentes kernels con el fin de conseguir la combinación optima en el caso lineal o no lineal, lo cual sugiere la obtención de una gran cantidad de características involucradas en la clasificación debido a que cada kernel se puede relacionar con la expansión del mapeo de características en un espacio determinado, por lo cual la sumatoria de otro kernel introduce una posibilidad más en cuanto a una cantidad de características [36] [37]
A continuación se describen los ejemplos más populares de funciones kernels:

- **Kernel lineal**

$$K(x, x^{'}) = \left\langle x, x^{'} \right\rangle \tag{2-8}$$

- **Kernel polinomial**

$$K_p(x, x^{'}) = \left\langle \gamma < x, x^{'} > +\Upsilon \right\rangle^p \tag{2-9}$$

- **Kernel gaussiano**

$$K(x, x^{'}) = exp(-\gamma \left\| x - x^{'} \right\|^2), \gamma > 0 \tag{2-10}$$

- **Kernel sigmoidal**

$$K(x, x^{'}) = tanh(\gamma < x, x^{'} > +\Upsilon) \tag{2-11}$$

Para los kernels, se hace necesario tener un caso que sea no linealmente separable, es conveniente aclarar que es posible formular una versión no lineal para un espacio de mayor dimensionalidad encontrando una transformación lineal $\phi(x)$, donde debe ser denotado como un producto escalar $K(x_i, x_j) = \phi(x_i)^T \phi(x_j)$ para el cual no serán necesarios los vectores especificos sino unicamente conocer el kernel.

Aprendizaje por múltiples kernels

Debido a que el uso de diferentes kernels o kernels compuestos, le asignan diferentes caracteristicas a los estimadores, es necesario tener en cuenta que las muestras pueden ser transformadas en diferentes espacios correspondientes al de Hilbert, o pueden ser transformadas en el resultante de varias funciones. Lo anterior implica que dado un vector x de debe realizar una transformación lineal del mismo lo cual implicaría que estaría compuesto por un espacio de Hilbert específico compuesto a su misma vez por n espacios dependientes del número de elementos del vector.

$$xx^{'} = \begin{pmatrix} [x_a \\ x_b] \end{pmatrix} \tag{2-12}$$

Con la transformación equivalente a:

$$\varphi(x) = \begin{pmatrix} \varphi^a(x^a) \\ \varphi^b(x^b) \end{pmatrix} \tag{2-13}$$

En lo anterior se demuestra que se obtienen dos espacios de Hilbert equivalentes a los literales **a** y **b** respectivamente, de igual forma se debe realizar un producto escalar compuesto entre la traspuesta de x_a y x_b por lo tanto el resultado del producto escalar de tales transformaciones resulta en $K(x_1^a, x_2^a) + K(x_1^b, x_2^b)$, lo que puede demostrar que al sumar dos kernels o realizar el producto escalar de las transformaciones lineales, equivale a un Kernel.
Los algoritmos basados en múltiples kernel usualmente utilizan diferentes métodos de aprendizaje que se pueden dividir según su categoría:

- **Reglas fijas:** Hacen referencia a funciones que no deben estar presenciadas por parámetros, para el caso de la suma o multiplicación de kernels es evidente que no necesitan de algún entrenamiento.

- **Enfoque heurístico:** Consiste en tener una función para la combinación que pueda ser parametrizada en la cual cada parámetro pueda ser medible. Tales medidas pueden ser calculadas por medio de matrices o un número como valores apropiados con el fin de comparar el rendimiento de cada kernel.

- **Optimización:** El parámetro puede ser buscado mediante una función de combinación parametrizada, con el fin de que se convierta en un problema de optimización. Con esto se podría formular un modelo diferente basado en la combinación de los parámetros obtenidos al resolver el problema de optimización.

- **Enfoques bayesianos:** Se deben interpretar los parámetros como variables aleatorias, de modo que se le pueda dar un énfasis a tales parámetros y finalmente realizar una inferencia para la etapa de aprendizaje.

- **Métodos impulsores:** Básicamente se encargan de evaluar el rendimiento constantemente, de modo que si el mismo no se encuentra a un nivel adecuado se le agrega iterativamente un nuevo kernel hasta que su rendimiento se vea mejorado.

Por otro lado, es necesario valorar la condición dada por la **función objetivo**, debido a que con esta misma se deben seleccionar los parámetros o la combinación de los mismos. Las funciones objetivo se pueden agrupar en tres categorias principales [34]:

- **Funciones basadas en similitud:** Calculan una métrica de similitud entre la matriz de kernels combinada y un kernel óptimo que resulta a partid de los datos de entrenamiento con el fin de seleccionar los parámetros que maximicen la métrica.

- **Funciones basadas en riesgo estructural:** Consiste en minimizar el riesgo estructural por medio de un parámetro de regularización que puede influir en el rendimiento del sistema. Las restricciones de los pesos de cada kernel pueden ser incluidas como las mismas restricciones del parámetro de regularización con el fin de darle uniformidad.

- **Funciones bayesianas:** Su objetivo consiste en medir el rendimiento de un kernel resultante a partir de dos candidatos posibles. Normalmente se usa la probabilidad a posteriori como función objetivo y finalmente relacionarla con la estimación de la máxima probabilidad a posteriori para la selección de parámetros.

Los algoritmos que se basan en múltiples kernel, se pueden categorizar de acuerdo a seis reglas específicas: El método de aprendizaje, debido a que dependiendo del mismo se sabe si es una sumatoria o una productoria, también existe la funcionalidad, la cual determina el tipo de combinación que se debe realizar (Lineal o no lineal), la función objetivo, el método de entrenamiento y la complejidad computacional. Para definir la suma de dos kernels primero se debe realizar la obtención de k_η el cual hace uso de f_η y su kernel canónico de entrenamiento con la matriz calculada $k_\eta(.,.)$, lo que quiere decir que para obtener un kérnel válido por suma o multiplicación de dos kernels válidos, se realiza según la ecuación 2-14 y 2-15:

$$k_\eta(\mathbf{x}_i, \mathbf{x}_j) = k_i(\mathbf{x}_i^1, \mathbf{x}_j^1) + k_i(\mathbf{x}_i^2, \mathbf{x}_j^2) \tag{2-14}$$

$$k_\eta(\mathbf{x}_i, \mathbf{x}_j) = k_i(\mathbf{x}_i^1, \mathbf{x}_j^1) k_i(\mathbf{x}_i^2, \mathbf{x}_j^2) \tag{2-15}$$

Finalmente si se aplica recursivamente para obtener la productoria o la sumatoria de kernels desde $m = 1$ hasta P, siendo este ultimo el número de kernels involucrados en la combinación quedaría de la siguiente forma:

$$k_\eta(\mathbf{x}_i, \mathbf{x}_j) = \sum_{m=1}^{P} k_m(\mathbf{x}_i^m, \mathbf{x}_j^m) \tag{2-16}$$

Y para la productoria:

$$k_\eta(\mathbf{x}_i, \mathbf{x}_j) = \prod_{m=1}^{P} k_m(\mathbf{x}_i^m, \mathbf{x}_j^m) \tag{2-17}$$

Aparte de la formulación inicial hecha para la productoria y la sumatoria, también se formula un "pairwise kernel"que se dice que funciona bien para el problema de predicción de interacciones entre proteínas y cuya formulación [34]

$$k_\eta(\mathbf{x}_i^a, \mathbf{x}_i^b)k_\eta(\mathbf{x}_j^a, \mathbf{x}_j^b) + k_\eta(\mathbf{x}_i^a, \mathbf{x}_j^b)k_\eta(\mathbf{x}_i^a, \mathbf{x}_j^b) \tag{2-18}$$

Para la ecuación anterior, se puede apreciar la formulación a partir de un k_η que corresponde inmediatamente a la formulación expresada para la combinación lineal, donde cumple el papel de regular el peso de cada uno de los kernels, además este ultimo se hace necesario cuando se realizauna comparación entre dos objetos, ya sean proteínas o genes.

2.2.3. Optimización

Los métodos de optimización consisten en fórmulas para minimizar o maximizar una variable involucrada en un proceso en el cual se tiene una función objetivo y un espacio de búsqueda determinado. Los primeros casos de optimización fueron detectados como problemas de programación lineal donde se pretendía minimizar o maximizar una función de tipo lineal. Una función objetivo hace referencia a la ecuación que será intervenida con el fin de encontrar uno de sus extremos por medio de programación lineal o no lineal, para ello hay que tener en cuenta que el problema debe tener ciertas restricciones que pueden limitar un espacio de búsqueda. Una forma para realizar optimización han sido los métodos numéricos, y aunque las soluciones se dan por medio de formulaciones matemáticas, se debe mencionar que todos los métodos numéricos tienen una caracteristica en común, todos ellos llegan a una solución aproximada. Por otro lado, con el avance de la computación se ha podido realizar la implementación de los algoritmos de optimización para realizar una búsqueda más rápida y precísa. A continuación se describen algunos métodos de optimización no lineal [38]:

- **Métodos de optimización directos:** Son métodos de optimización para funciones multivariables, es decir, que pueden tener mínimos locales y un mínimo global, por lo tanto debe realizarse una búsqueda más extensa con el fin de encontrar el mínimo (o máximo) más general y la búsqueda no se quede anclada en un mínimo local. Para esta clase de métodos se hace necesario un criterio de diferenciabilidad de la función objetivo. Entre los métodos directos podemos encontrar: Búsqueda aleatoria, búsqueda en rejilla y método simplex (geométrico) entre otros [39].

- **Métodos de optimización indirectos:** Para estos métodos se tiene en cuenta la diferencialidad para el direccionamiento de la búsqueda, lo que quiere decir que son dependientes de las derivadas. De este modo, la dirección de la búsqueda reduce la función objetido de tal

forma que se deba encontrar el máximo o mínimo. Para realizar algoritmos de optimización indirecta se debe tener en cuenta [39]:

Si x_0 es un punto inicial y x_1 es un punto próximo de la búsqueda

$$f(x_1) < f(x_0) \tag{2-19}$$

En cuanto a los métodos indirectos se encuentra el método de newton, el método del gradiente y el método de la secante.

- **Métodos metaheurísticos:** Los métodos de optimización metaheurísticos tratan de evadir los mínimos locales, y son súmamente utiles en problemas de optimización que usen combinatorias donde se usen variables continuas o discretas. El punto de inicio de las técnicas metaheuristicas, parten de una solución o un conjunto de soluciones que no suelen ser la óptima, de este modo se debe iterar de modo que se realicen comparaciones entre un estado actual y uno anterior. El proceso se lleva a cabo cierta cantidad de veces hasta que se cumpla una condición valorada previamente. Entre las técnicas más populares de éste tipo de optimización se encuentran los algoritmos genéticos, éste básicamente se ocupa de aplicar ciertos operadores genéticos a una población de individuos o muestras. Seguidamente, también se pueden encontrar técnicas basadas en recocido simulado, enjambres de partículas, búsqueda cuckoo, colonias de hormigas [40].

Optimización por Enjambres de Partículas

Usualmente los algoritmos metaheuristicos están basados en algoritmos bioinspirados, los cuales emulan el comportamiento de grupos biológicos o procesos determinados. Esta clase de algoritmos son usados para los casos donde no existe un método de resolución exacto, ya que los métodos heurísticos tienden a quedarse en óptimos locales en problemas en los cuales se pueden encontrar una gran cantidad de mínimos por no tener una componente exploratoria amplia. Los algoritmos metaheurísticos tienden a escaparse de los mínimos locales de modo que arroje la solución optima a partir de un conjunto de soluciones. Por ejemplo, para el caso del algoritmo por enjambres de partículas (Particle swarm optimization - PSO) se define como un algoritmo bioinspirado y desarrollado por Eberhart y Kennedy [41], usado para la sintonización de parámetros de la SVM. La optimización por enjambres de partículas es un algoritmo metaheurístico inspirado en el comportamiento de los enjambres de aves o peces introducido para optimizar funciones continuas no lineales. Este algoritmo realiza la búsqueda en el espacio de los parámetros de la función objetivo ajustando la trayectoria de las partículas.

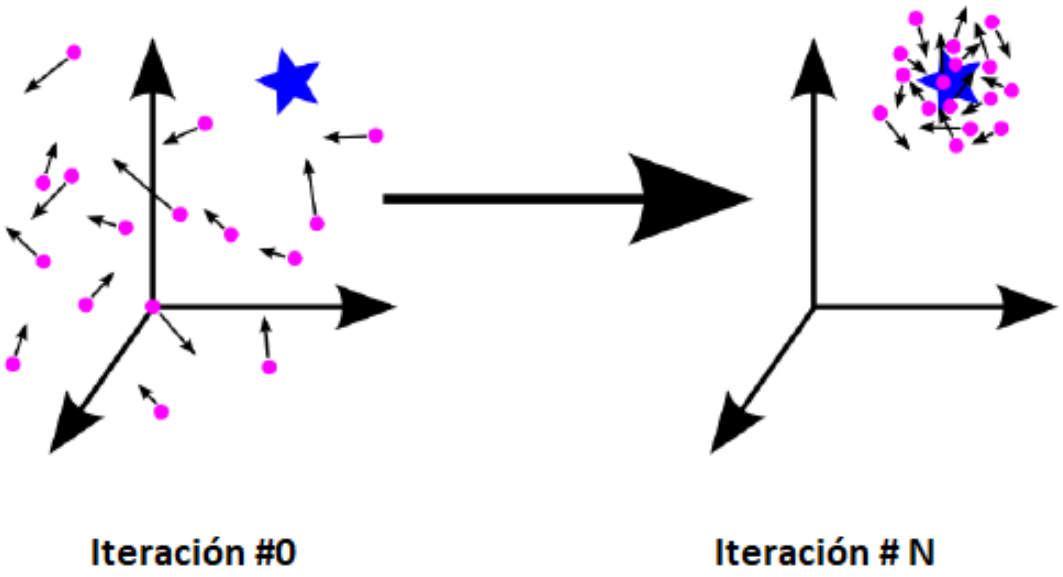

Figura 2-9: Enjambre de partículas. Tomado de: wireless tech thoughts

En la figura **2-9** se pueden evidenciar las partículas distribuidas en un espacio determinado, y luego muestra la convergencia de las partículas a un punto determinado por cuenta del método. El propósito de éste tipo de optimización es generar un conjunto de partículas que se ubican en el espacio de búsqueda de manera aleatoria como se puede observar en la figura, luego de ubicar las partículas en el espacio, se procede a reubicar cada una de las mismas con respecto a su mejor posición visitada individualmente y la mejor posición global (Entre todas las partículas), la finalidad es que a medida que pasen las iteraciones las partículas puedan converger a un mínimo global debido a su componente exploratorio. Una de las condiciones de parada es usualmente el número de iteraciones o de otro modo puede ser un criterio de tolerancia establecido desde el principio del ejercicio. Primero que todo se parte de una función $f(x, y)$ que es desconocida, agregando que la ventaja de esta técnica de optimización es que puede aplicarse en problemas de dos o más dimensiones [42].

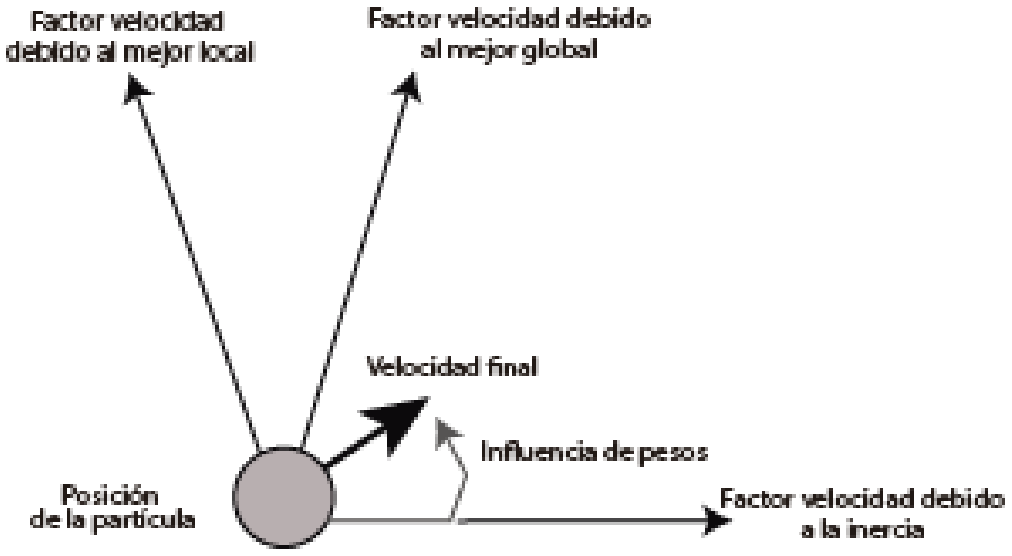

Figura 2-10: Partícula individual. Extraído de: www.cs.us.es

En la imágen **2-10** se puede apreciar una partícula y cuales son los factores que influyen en su movimiento tales como la **posición inicial**, esta puede ser influyente debido a que si en la próxima iteración no se encuentra una posición mejor que la actual, la partícula debe retroceder el paso realizado en tal iteración, se tiene una influencia debida a la mejor posición **local y global**, estas se pueden controlar por medio de velocidades asociadas a cada uno de estos componentes que están definidas por un α_1 y un α_2 que son local y global respectivamente, estos parámetros están entre 0 y 1, y entre más cerca de 1 se encuentra, más rápida será la exploración. Estos parámetros, descritos en la ecuación 2-20 y definidos anteriormente son usualmente llamados pesos ($c_i r_i$), los cuales le dan una influencia en la convergencia de un mínimo. Adicionalmente se tiene la **inercia**(w), por medio de esta se puede aumentar el componente exploratorio de cada una de las partículas, sin embargo se afecta la convergencia, lo que quiere decir que necesita de más iteraciones con un componente de inercia alto. Finalmente, la velocidad final estará condicionada por la influencia de los pesos que se le dan a los componentes locales, globales y de exploración. [43].

$$v_i(t+1) = wv_i(t) + c_1 r_1[\widehat{x}_i(t) - x_i(t)] + c_2 r_2[g(t) - x_i(t)] \tag{2-20}$$

En la ecuación 2-20 se pueden evidenciar tanto los componentes locales como globales involucrados en las velocidades de las partículas, donde también se notan los pesos que se le pueden dar a las conexiones locales o globales, esto quiere decir que el movimiento va a depender de su aporte en los dos sentidos regidos por el parametro c_i, y del mismo modo se le puede dar peso al valor que se define como inercia w el cual generalmente agrega un componente exploratorio. En la ecuación 2-21 se involucra una posición inicial de cada partícula, además se debe tener en cuenta cuál se encuentra mejor ubicada en el espacio de búsqueda con respecto a un mínimo global, esto se debe a que el resto de partículas deben procurar desplazarse hacia la mejor global. Además, se le debe

aplicar una velocidad a tales partículas, de esta dependerá también la rapidez en la exploración. En tal ecuación se nota la actualización de la posición $x_i(t+1)$ donde necesariamente influye la posición visitada anteriormente denotada como $x_i(t)$ y la velocidad que está explicada más a fondo en la ecuación 2-20.

$$x_i(t+1) = x_i(t) + v_i(t+1) \tag{2-21}$$

Optimización cuadrátrica

La optimización cuadrática consiste en un método para conseguir la minimización de una función de tipo cuadrática sujeta o no a ciertas restricciones. Esta forma de optimización se deriva de la programación lineal, debido a que es un problema de función objetivo cuadrática con restricciones lineales. De este modo, se encuentra dentro del grupo optimización llamados problemas no lineales. La programación cuadrática es un algoritmo perteneciente a la programación lineal, expresando el problema de modo simple con condiciones de desigualdad dentro del mismo. En la siguiente ecuación se presenta de modo general, la representación de un problema de tipo cuadrático:

$$c^t x + \frac{1}{2} x^t Q x \qquad s.t \qquad x \geq 0 \qquad Ax - b \geq 0 \tag{2-22}$$

Donde $\mathbf{C}$ es un vector de coeficientes que deben ser constantes, A es una matriz que debe ser $m \times n$ y para todos los casos se debe asumir el tamaño simétrico para la variable Q. Si la variable Q es positivo llevaría a un mínimo global, si éste no es positivo, es posible que el problema no se encuentre restringido y por lo tanto lleva a mínimos locales. Por otro lado, las desigualdades se resolverán como igualdades y se procede a resolver como un problema de optimización lineal.

3 Método propuesto

En esta sección se explicará detalladamente cada etapa experimental del trabajo realizado, en la que se resaltará cada uno de los pasos principales descritos en el esquema metodológico.

En la figura **3-1** se puede observar el flujo metodológico del trabajo donde se tiene una adquisición de interacciones positivas por medio de la base de datos DIP, para las interacciones negativas se implementa un algoritmo de permutaciones aleatorias. Seguidamente, ya con las características obtenidas se obtuvieron los kernels, con los mismos, se deben calcular los pesos de cada uno por medio de una optimización cuadrática que dió como resultado una matriz combinada final para usar en el paso de clasificación donde se optimizó el parámetro de regularización C (Por medio de optimización por enjambre de partículas) de la máquina de vectores de soporte para finalmente realizar una predicción de interacciones.

3.1. Adquisición de interacciones

3.1.1. Base de datos

La base de datos es llamada DIP (Database of interacting proteins), contiene aproximadamente 73.000 interacciones positivas, estas se obtuvieron experimentalmente por diferentes métodos físicos que pueden variar de efectividad según el principio de búsqueda de la interacción [44]. En esta base de datos se pueden obtener los identificadores de Uniprot de cada uno de los pares de proteinas que tienen una interacción positiva [45], en Uniprot se obtienen datos relevantes de cada proteína como la secuencia, interacciones, dominios reconocidos y relaciones filogenéticas. Luego de la obtención los identificadores de cada una de las proteínas, se procedió a operar por medio de un algoritmo de permutaciones aleatorias que se ejecutó por medio de los identificadores, todo ello con el fin de obtener las interacciones negativas. Luego de obtener tal lista, se pudo asociar cada uno de los identificadores de la lista con una secuencia .fasta conocida, esta secuencia tiene un formato conocido y se usó para identificar la secuencia de aminoácidos específica que fué usada para la filtración de datos y caracterización. Además de ello, se tomó en consideración la base de datos VirusMentha [46] con el fin de realizar las pruebas posteriores con proteínas virales con la misma cantidad que en la base de datos DIP, sin embargo cabe señalar que a pesar de ser proteínas virales, no garantiza de que estén directamente relacionadas con la NS5A.

3.1.2. Extracción y filtrado de interacciones

A partir de la creación de un archivo .fasta con las secuencias necesarias y extraído de la base de datos cuyos identificadores se conocen, se verificó que no hubieran secuencias repetidas, se realizó un filtrado por similitud por medio de CD-HIT [47], este software es utilizado para agrupar y comparar secuencias de proteínas o nucleótidos con el fin de obtener un grupo de proteinas donde no sean similares o tengan un umbral de distancia filogenética entre las mismas por medio de la reducción significativa analizando las secuencias en su estructura con el fin de corregir el sesgo en el conjunto de datos.

En este paso, se debió ingresar un umbral mínimo de 95 % de similitud, con el fin de garantizar un conjunto de secuencias no repetidas, esto se debe a que entre más homología tengan, hay una probabilidad más alta de obtener interacciones positivas o en casos no deseados; Falsos positivos. Luego de la obtención del conjunto de secuencias cuya interacción es positiva, se realizará un alineamiento por medio de ClustalW [48], donde se valorará el **Bitscore**, éste tiene que ver con el porcentaje de homología entre una A y B (Entre más alto sea el bitscore, existe una probabilidad más alta de interacción entre una proteína A y B), este $Bitscore$ hace uso del máximo y el mínimo resultado de similitud entre todas las secuencias para normalizar los resultados. Acorde con esto, se obtiene una matriz triangular de la interacción de todos contra todos normalizada con cada bitscore($S^{'}$), teniendo en cuenta el mínimo y el máximo y luego seleccionarse los que se encuentren por debajo de la media o un umbral propuesto previamente [49].

Es necesario aclarar que para el caso de la filtración y extracción de interacciones, que el hecho de que concuerde con otras proteínas en bajo porcentaje de homología, no garantiza que tengan una proximidad o relación, por el contrario, el método consiste en que tengan un porcentaje de similitud bajo.

$$S^{'} = \frac{\lambda S - ln(K)}{ln(2)} \tag{3-1}$$

$$Distance(x, y) = 1 - S^{'} \tag{3-2}$$

En la ecuación 3-1 y 3-2 se puede apreciar la normalización del Bitscore y el cálculo de cada una de las distancias entre las proteínas, las cuales sirvieron para extraer las interacciones negativas [49]. El calculo de las distancias de cada una de las interacciones involucra cada $Bitscore$ normalizado que se encuetra entre 0 y 1, lo que quiere decir que al calcular la distancia, entre más cerca se encuentre a 0 quiere decir que existe un gran porcentaje de homología entre las dos secuencias y por ende una probabilidad más alta de interacción.

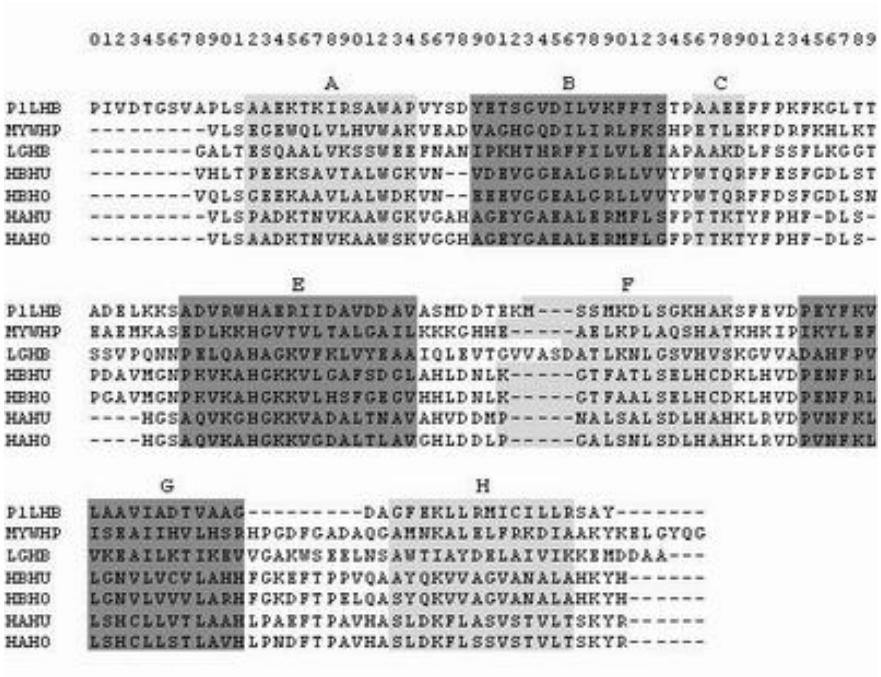

Figura 3-2: Alineamiento múltiple en ClustalW

En la figura **3-2** se puede ilustrar la visualización del software clustalW que permite observar el alineamiento global de secuencias [48], la escala de grises corresponde al nivel de coincidencia entre las secuencias dada una sección específica de cada proteína. Allí donde se efectúa un apareamiento de aminoácidos, pueden existir espacios vacíos debido a la longitud de cada cadena, del mismo modo pueden haber concordancias entre las cadenas o finalmente no coincidir. En conformidad con las condiciones anteriores, esto puede asignar un puntaje dependiente de la sumatoria de las coincidencias, tipos de sustituciones o no coincidencia de cada uno delos caracteres.

A partir de esta matriz de interacciones se obtuvieron las n primeras secuencias que cumplan con un umbral de distancia por debajo de 0.4, este valor es escogido debido a que tal distancia indica menos del 50 % de similitud entre una secuencia A y B, lo que indica que la probabilidad de interacción es mucho menor a las que se encuentran por encima de este umbral. Es importante aclarar que entre más bajo sea el umbral, se puede correr el riesgo de no satisfacer una cantidad n de interacciones que es igual a la cantidad de interacciones positivas a usar. A causa de esto se usaron 5000 interacciones positivas, lo que exige una cantidad igual de negativas para evitar el desbalance de clases.

En ciertos casos se usaron bases de dados compuestas por 2500 interacciones positivas y 2500 interacciones negativas, por lo tanto se compone de 5000 para los casos que el kernel es una combinación lineal de 5 resultantes. Y para los casos en los cuales no se usa combinación de kernels se usaron los mismos $10{,}000$ datos. La diferencia en cuanto a la cantidad de datos radica en que si se van a usar menos kernels pero la cantidad de datos es igual, existiría un desequilibrio.

3.2. Extracción de características

En cuanto a las características, se tuvieron en cuenta de AAindex, la cual es una base de datos donde se pueden tener características físico-quimicas por cada aminoácido, o en conjunto, esta es una fuente de carácterísticas netamente numéricas que se relacionan fuertemente con los tipos de interacciones de los aminoácidos [50].

$$f_i^{'} = \frac{f_i - min(F)}{max(F) - min(F)} \tag{3-3}$$

En la ecuación 3-3 se puede observar el cálculo de frecuencia para cada trío de aminoácidos donde también se tiene en cuenta la frecuencia mímima y máxima, de éste modo también se pretende normalizar los datos relacionados con la frecuencia. Asimismo, si se tiene $7^3 = 343$ combinaciones diferentes de tripletes, por lo tanto, $F = \{f_1, f_2, ..., f_{343}\}$ es un vector de características que se puede normalizar teniendo en cuenta que las secuencias de aminoácidos están comprendidas entre 20 aminoácidos, se forman 7 grupos diferentes con respecto a sus propiedades químicas, de este modo: {A, V, G}, {I, L, F, P}, {Y, M, T, S}, {H, N, Q, W}, {R, K}, {D, E}, y {C}. Así, las proteínas serán mapeadas de acuerdo al número del grupo, por ejemplo, si se tiene la cadena{AGHLCRLV} será mapeada como $\{11427521\}$ como un vector [49].

3.3. Optimización metaheurística

Inicialmente se realiza optimización por enjambres de partículas para la optimización de cada peso η descritos en la ecuación 2-6. De modo que cada peso $\eta_n = 1 - \eta$, para que finalmente cada peso aporte a la suma de los pesos igual a 1, por lo tanto se puede decir que el espacio de búsqueda de cada peso será reducido de acuerdo a la sumatoria teniendo en cuenta cada peso anterior. Es así como la sumatoria descrita anteriormente debe ser matemáticamente descrita como $\sum_{i=1}^{10} \eta_n = 1$. Por otro lado, de debe tener en cuenta que para la optimización de el parámetro C, se tiene un espacio de búsqueda de 1 a 100.
Esta sección tiene como objetivo maximizar el desempeño del clasificador por medio de un método de optimización metaheurístico, evaluable mediante medidas de desempeño destacadas en el estado del arte. El parámetro C es conocido como el parámetro de costo, este término se encarga de regular el compromiso de la máxima desviación aceptada.

Se debe tener en cuenta que usó una validación cruzada (Valor predeterminado de 5 particiones) con una proporción de datos de 80 % para el entrenamiento y 20 % para las predicciones [51] [52] Para la implementación genérica de la optimización por enjambres de partículas se debió encontrar el mínimo de una función específica y definida por el problema de interés. En el siguiente código realizado se ejecuta la optimización por enjambres de partículas definiendo los parámetros α en $0{,}9$ para aumentar la exigencia del algoritmo, donde se propuso establecer las iteraciones como condición de parada, lo que quiere decir que no necesariamente converge a un mínimo global en

todos los casos. La variable x_i en la ecuación 3-5 hace referencia al establecimiento de 10 partículas en el espacio de búsqueda aleatoriamente.
Además de la optimización del parámetro C, también se realizará una optimización cuadrática donde se optimizarán los pesos referentes a los 10 parámetros η correspondientes a cada uno de los kernel en cuestión. Para el primer parámetro es necesario denotar un conjunto de valores que puede ir de 0 a 100.

3.3.1. Alineamiento de kernels

Primeramente, se realizó una normalización de los datos correspondientes a las características. Luego de ello, se obtuvo un valor correspondiente a la mediana de todos los datos ya normalizados, éste valor es esencial en la combinación lineal de kernels, ya que se realizó con el fin de no escoger los valores de σ arbitrariamente. De este modo, cada valor de σ escogidos en rejilla fueron multiplicados por el valor resultante de la mediana de los datos.
Antes de realizar la combinación lineal de kernels, se debió tener en cuenta la cantidad de parámetros involucrados. En primer lugar se tienen los parámetros σ, y los parámetros correspondientes a los pesos que son denotados como η. Los primeros dependerán de la forma de los datos, por lo cual se normalizaron las caracteristicas y seguidamente se extrae la mediana de las mismas con el fin de multiplicar el valor del conjunto de σ por la rejilla de kernels propuesta, esto se debe a la gran variación de los datos, y siendo de otro modo, medidas como la mediana serían sesgadas. Con el parámetro η se realizó un conjunto de operaciones para efectos de optimización de pesos en cada kernel, y finalmente realizar una distribución equitativa de acuerdo con el aporte que realice cada kernel dentro de la combinación lineal. Es así como se debe proceder sabiendo que la sumatoria de los pesos debe ser igual a 1 lo cual infiere que cada peso puede tomarse como un porcentaje de importancia dentro de la operación. A continuación se describe primero la forma de realizar el alineamiento de kernels donde se establece una métrica entre el kernel combinado y el kernel óptimo o kernel de etiquetas [34]:

$$CA(K_1, K_2) = \frac{\langle K_1^c, K_2^c \rangle_F}{\sqrt{\langle K_1^c, K_1^c \rangle_F \langle K_2^c, K_2^c \rangle_F}} \tag{3-4}$$

Para la ecuación 3-4 se tiene que K^c corresponde a la versión centrada de K, el cual significa que se efectúa un alineamiento a modo de comparación, donde $\mathbf{K}_1$ corresponde a la combinación lineal de kernels y $\mathbf{K}_2$ corresponde al kernel de etiquetas denotado como $\mathbf{y}\mathbf{y}^T$. Es asi como finalmente se obtuvo un problema de optimización cuadrática en el cual se tuvo que optimizar el alineamiento de kernels donde se propone maximizar $CA(K_\eta, yy^T)$ con respecto a la condición $\sum_1^M \eta_i$ donde M es el número de kernels [53].

$$\mathbf{K}^c = \mathbf{K} - \frac{1}{N}\mathbf{1}\mathbf{1}^T\mathbf{K} - \frac{1}{N}\mathbf{1}\mathbf{1}^T - \frac{1}{N^2}(\mathbf{1}^T\mathbf{K}\mathbf{1})\mathbf{1}\mathbf{1}^T \tag{3-5}$$

Para la ecuación 3-5 se tiene que $\mathbf{1}$ corresponde a un vector de unos con dimensión propia. Para el cual se debe optimizar la versión centrada del kernel de modo que se debe maximizar $CA =$

$(\mathbf{K}_\eta \mathbf{y}\mathbf{y}^T)$ con la condición que $\eta \epsilon M$, además se debe realizar el cálculo de η de la siguiente forma:

$$\eta = \frac{\mathbf{M}^{-1}\mathbf{a}}{\|\mathbf{M}^{-1}\mathbf{a}\|_2} \tag{3-6}$$

Para el caso de la ecuación 3-6 se debe tener en cuenta que $\mathbf{M} = \{\langle \mathbf{K}_m^c, \mathbf{K}_h^c \rangle_F\}_{m,h=1}^P$ y $\mathbf{a} = \left\{\left\langle \mathbf{K}_m^c, \mathbf{y}\mathbf{y}^T \right\rangle_F\right\}_{m=1}^P$. Finalmente, los pesos para cada kernel deben ser valores no negativos cambiando la definicion de $\mathbf{M}$ por $\{\eta : \|\eta\|_2 = 1\}$ con la condición que η debe ser cualquier número real positivo y el problema de optimización cuadrática pasa a describirse según la siguiente ecuación con la condición que $\mathbf{v}$ deberá ser también un numero real positivo:

$$\mathbf{v}^T\mathbf{M}\mathbf{v} - 2\mathbf{v}^T\mathbf{a} \tag{3-7}$$

Por medio de lo anterior, se asegura unos pesos distribuidos de acuerdo a las condiciones propuestas y de paso se obtiene la mejor combinación que garantiza un aporte óptimo de cada uno.
Para tal paso resulta un conjunto de valores descritos como η =0.4041, 0.1711, 0.0659, 0.0012, 0, 0,0,0,0,0.3589

- La fusión de kernels consiste en la combinación de diferentes kernels con el fin de conseguir la combinación óptima en el caso lineal o no lineal, lo cual sugiere la obtención de una gran cantidad de características involucradas en la clasificación debido a que cada kernel se puede relacionar con la expansión del mapeo de características en un espacio determinado [54].

  A continuación, se describe el kernel que se utilizó para la prueba:

$$\mathbf{K}(\mathbf{x}, \mathbf{x}_i) = exp(\frac{\|\mathbf{x} - \mathbf{x}_i\|^2}{2\sigma}) \tag{3-8}$$

  En la ecuación 3-8 se menciona el kernel gaussiano usado para la prueba, por tanto se debe tener en cuenta que calcula la similitud en un espacio ya transformado con otras dimensiones. De modo que se calcula la distancia entre $\mathbf{x}$ y $\mathbf{x}_i$, para éste caso σ controlar la magnitud de la distancia entre los puntos.

- El éxito de una aplicación basada en aprendizaje de máquina en la cual se realice un aprendizaje por múltiples kernel, depende en gran medida el kernel a usar y los parámetros con los cuales se aplicará la función. Para este caso se tomará como primera opción el Kernel de base radial (RBF) ya que es la opción más razonable por cuenta de que no solamente depende de la distancia de origen o referencia, sino que toma la suma de infinitos kernels de forma polinomial. El ajuste de parámetros de las funciones kernel a utilizar es un problema dependiente de la cantidad de los mismos, por lo tanto, para este caso es necesaria realizar la discretización del espacio de búsqueda para el parámetro σ del kernel RBF, tomando en cuenta que son 10 kernels y se requiere de un conjunto asignado uno a uno dependendiente de las caracteristicas. .

$$\mathbf{K}_{\sigma_i} = \mathbf{K}_{\sigma_1}, \mathbf{K}_{\sigma_2} ... \mathbf{K}_{\sigma_n} \tag{3-9}$$

De este modo, se procede a encontrar un conjunto de valores $\mathbf{K}_{\sigma_n}$. Luego de ello, se realiza una combinación lineal de kernels

$$\mathbf{K}_n = \sum_{m=1}^{P} n_m k_m(\mathbf{x}_i^m, \mathbf{x}_j^m) \tag{3-10}$$

Esta combinación lineal tomará en cuenta solo las características, por lo tanto está entre la primera y la P-esima caracteristica para la combiandión lineal. Por lo tanto, la representación del espacio depende de la dimensionalidad de las características, las cuales serán reemplazadas por los kernels.

Seguidamente, puede decirse que para problemas de clasificación binario, la selección de pesos η_m se denota como:

$$\eta_m = \frac{A(\mathbf{K}_m \mathbf{y}\mathbf{y}^T)}{\sum_{h=1}^{p}(\mathbf{K}_h, \mathbf{y}\mathbf{y}^T)} \tag{3-11}$$

Del mismo modo, se realizó una comparación de pesos por medio de una medida compatible entre los kernels. Se implementará una técnica heurística con el propósito de obtener los pesos de los η referente a cada uno de los kernel, la cual será la optimización cuadrática.
Para la formulación de el alineamiento anterior es necesario formular $\langle K_1, K_2 ... K_n \rangle_F$ que es equivalente a la sumatoria desde 1 hasta la cantidad N de kernels de modo que:
Lo anterior se considera una medida de similitud que considera el coseno del ángulo que hay entre k_1 y k_2 dados únicamente dos kernel. El alineamiento entre los kernels es considerado una medida importante ya que la probabilidad de desviación de la media decae exponencialmente, indicando un alto grado de alineamiento entre los kernels [34].
Finalmente se propone una metodología de múltiples kernels basada en los siguientes pasos:

- Delimitación del parámetro C de la SVM, con valores desde 0 hasta 100 con una resulución de pasos en 1 es dependiente de la revisión de problemas referentes a predicción de interacciones, en los cuales los valores relativamente más altos, arrojan resultados menos relevantes. Los kernels serán escogidos de menor a mayor, variando en una unidad exponencial como lo indican los valores descritos.
- Realizar la obtención del valor de la mediana de los datos referentes a las características.
- Multiplicar el conjunto de valores de σ por la mediana ($X_{0,5}$) para así obtener cada uno de los valores de σ .
- Conseguir los valores de η por medio de la optimización cuadrática.

- Realizar la combinación lineal propuesta en la ecuación con los valores calculados.

- Se entrena la SVM y se validan los resultados de acuerdo a las medidas de desempeño propuestas como lo son: Especificidad, sensibilidad, puntaje F1, exactitud y precisión.

3.3.2. Implementación optimización

Debido a que el objetivo principal es aumentar el desempeño de un clasificador de interacciones, fué necesario acudir a un método de optimización heurístico y metaheurístico, con el fin de determinar cual es el que tendría mejor desempeño, ya que de este modo, se puede regularizarlos parámetros η y C con el fin de establecer una frontera que permita el menor número posible de errores.
En primer lugar, se realizan pruebas con un pairwise kernel, el cual no tiene combinación, las pruebas que le siguen las pruebas con reducción de caracteristicas por relevancia y redundancia donde las caracteristicas tienden a llegar de 460 a 243 caracteristicas a utilizar, lo anterior tomado como una primera parte ya que no tienen una relación directa con la combinación de kernels. Por otro lado, se realizan pruebas con variaciones del parámetro σ y poniendo pesos equilibrados, lo cual quiere decir que en este punto no se requiere aún de tener una métrica de los datos.
Por último, se realizarán las priebas involucrando al método con la combinación de kernels, tomando la mediana y relacionandola con el σ y luego de optimizar los pesos usando los mismos para la combinación lineal y finalmente obteniendo una matriz resultante y final.

3.4. Clasificación y predicción

3.4.1. Clasificación

Es necesario tener en cuenta que tanto el espacio de búsqueda como las restricciones del problema de optimización, deben estar en sintonía, esto se debe a que en ninguno de los casos para los parámetros libres deben tomar valores menores que cero. De este modo, se debe tener en cuenta que el uso de múltiples kernels y la sintonización de los parámetros no necesariamente deben ser iguales para el caso de los kernels y del mismo modo para el parámetro C de la SVM. Cabe mencionar que para este caso, se utilizan 10 kernels combinados linealmente con el fin de probar la variación de pesos en cada uno de ellos, se aclara que el parámetro σ se ajustó uniformemente en cada kernel, particularmente en este caso son calculados de acuerdo al conjunto con el cual se multiplica la mediana por el conjunto $\sigma = 0{,}1, 0{,}3, 0{,}5, 0{,}7, 0{,}9, 2, 5, 7, 10, 20$.
Por lo tanto puede tener una influencia directa sobre la clasificación ya que en este caso hay dos parámetros libres para el kernel y un parámetro libre de la SVM. Por otro lado, también se tiene que el espacio de características es inducido por el kernel, lo que quiere decir que el espacio de búsqueda puede variar de acuerdo a la dimensional del espacio de características.

$$Donde\ 0 < C \leqslant 100 \tag{3-12}$$

C es conocido como el parámetro de costo, este término se encarga de regular el compromiso entre la suavidad de la función y la máxima desviación aceptada. Lo que esto quiere decir es que las clasificaciones de los datos tienen un buen desempeño en el caso de que puedan ser separados linealmente, de modo contrario es necesario implementar una función Kernel de manera que pueda mapear los datos en un hiperplano de mayor dimensión.
Para el uso de kernels es necesario que el espacio de búsqueda como las restricciones del problema de optimización, deben estar dentro de límites similares. Además, se puede mencionar que los pesos de los kernels pueden adoptar valores diferentes, sin embargo se encuentran en un rango menor que 0.5 para cada uno debido a que en total deben sumar 1 [34].

3.4.2. Validación cruzada

Se debe tener en cuenta que se usó una validación cruzada (Valor predeterminado de 5 particiones) con una proporción de datos de 80 % para el entrenamiento y 20 % para las predicciones, cuyo fin es confirmar que la predicción en todas las divisiones sea consistente, de ser así, quiere decir que el predictor tiene un buen comportamiento general y la medida de desempeño es acorde a los datos usados.
Con el fin de analizar el desempeño de un clasificador se utilizaron diferentes métricas como la sensibilidad, la especificidad. Para el caso de la predicción de interacciones entre proteínas se propone maximizar el rendimiento del clasificador y validando por medio de la exactitud descrita en la siguiente ecuación:

$$Acc = \frac{TP + TN}{TP + FP + TN + FN} \tag{3-13}$$

De este modo, se debe encontrar un parámetro C que sea apropiado para el problema de optimización, para este problema especifico se usó optimización por enjambre de partículas y finalmente verificando por medio de la ecuación 3-13.

3.4.3. Interacciones NS5A

Ya que la secuencia de la proteína NS5A es posible conseguirla por secuencia .fasta, es posible caracterizar fácilmente tal proteína, además no guarda alto porcentaje de homología con las demás. Para la predicción de las interacciones de la proteína en cuestión, se realiza primero un subconjunto de las proteínas de interés, las cuales tendrán que pasar por todo el proceso de caracterización combinación lineal en conjunto con NS5A. Después de ello se debe pasar por el clasificador teniendo en cuenta que se hace una validación de todos contra todos y que el clasificador tiende a ser no supervisado, por otro lado, como las interacciones entre las otras proteínas no son de gran importancia, de las interacciones de las demás proteínas unicamente seleccionamos las que interaccionaron con la proteína de importancia para nosotros.

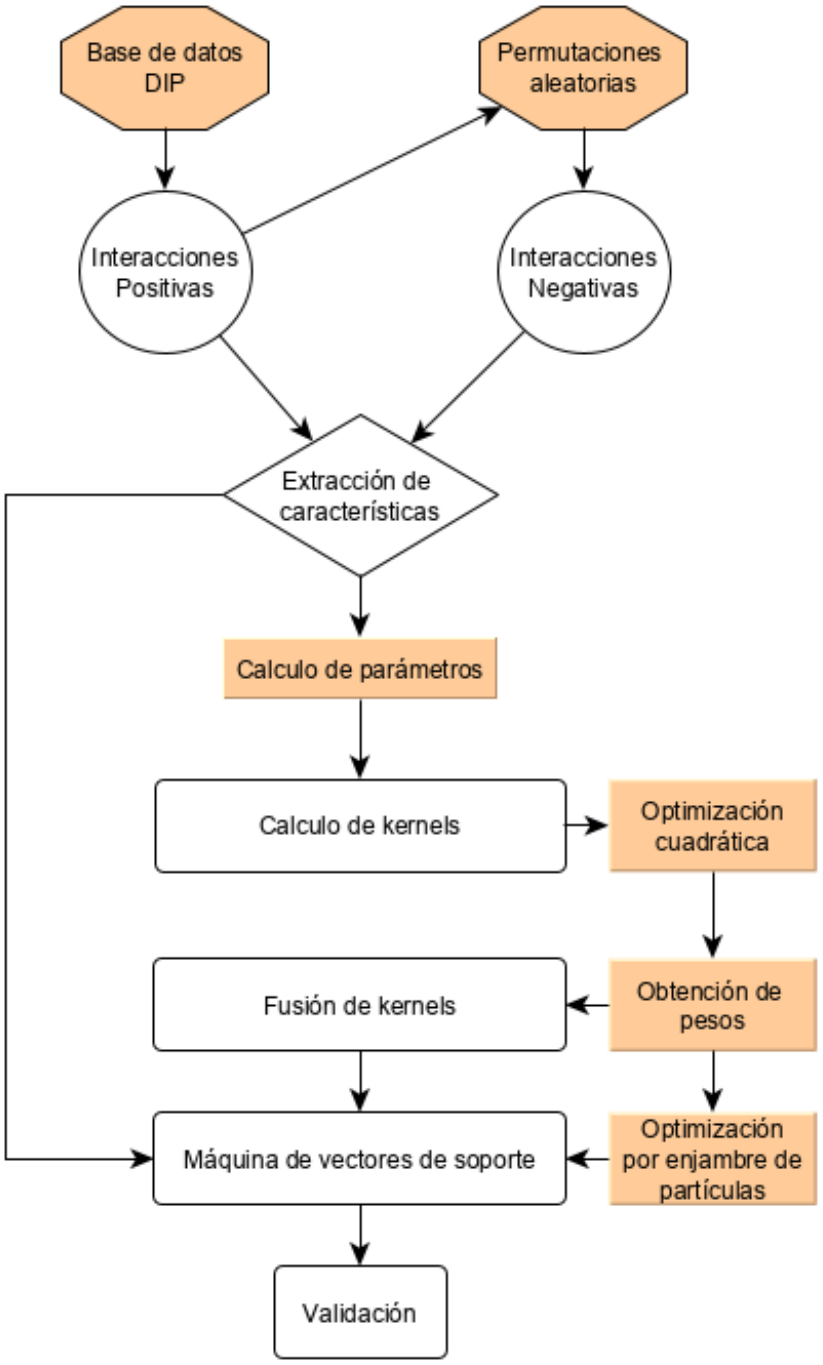

Figura 3-1: Esquema de trabajo para clasificación de interacciones entre proteínas por múltiples kernel

4 Resultados y discusión

4.1. Resultados

4.1.1. Resultados para optimización metaheurística

En primer lugar, se pueden apreciar los resultados para la optimización de parámetro η por medo de la optimización metaheurística, en esta parte se optimizaron los pesos de los kernels obtenidos de modo que la sumatoria de los pesos de 5 kernels diferentes debían sumar una totalidad de 1. Por otro lado se utsa un parámetro C de 10 debido a evaluaciones anteriores, siendo esta la mejor sintonización.

Tabla 4-1: Resultados de clasificación para 10.000 muestras y optimización de parámetro η por medio de optimización metaheurística

Partición	Sensibilidad	Especificidad	Media geométrica
1	0.32	0.28	0.31
2	0.28	0.30	0.30
3	0.25	0.24	0.24
4	0.26	0.16	0.20
5	0.31	0.29	0.31
Media	0.28 ± 0.02	0.25 ± 0.04	0.27 ± 0.04

4.1.2. Pairwise kernel

Para este caso, se muestran los resultados obtenidos en ensayos realizados según la metodología, en los cuales se emplea una máquina de vectores de soporte con una validación cruzada con particiones $80\,\%$ - $20\,\%$ respectivamente, para éste, se usa un pairwise kernel, donde se utilizan criterios de similitud, y adicionalmente, se debe afirmar que no se usan características físico químicas [55]. El uso del parámetro C se realiza de modo predeterminado, lo cual indica que no hay una sintonización.

Tabla 4-2: Resultados de clasificación para 10.000 muestras con pairwise kernel

Partición	Sensibilidad	Especificidad	Media geométrica
1	0.68	0.71	0.68
2	0.64	0.73	0.69
3	0.81	0.72	**0.73**
4	0.52	0.90	0.68
5	0.61	0.82	0.70
Media	0.65 ± 0.11	0.77 ± 0.08	0.70 ± 0.02

4.1.3. Reducción por relevancia y redundancia

Para la segunda instancia de resultados, se propuso una clasificación con $10{,}000$ muestras, sin embargo, se realiza con las características planteadas en el marco metodológico con una reducción de caracteristicas por relevancia y redundancia, para que finalmente se use un total de $10{,}000$ interacciones para cada clase [56], para éste se usan 260 características de 480 disponibles.

Tabla 4-3: Resultados para el uso de características físico-químicas con $10{,}000$ muestras

Partición & Sensibilidad	Especificidad	Media geométrica	Media geométrica
1	0.73	0.77	0.75
2	0.81	0.68	0.74
3	0.78	0.72	0.75
4	0.79	0.72	**0.76**
5	0.77	0.71	0.74
Media	0.77 ± 0.03	0.71 ± 0.04	0.74 ± 0.01

4.1.4. Optimización de parámetro C, sin alineamiento kernel

Los resultados obtenidos para la optimización del parámetro C de la SVM se encuentran entre 0 y 100 que se encuentran en las 5 particiones respectivas de la validación. En las siguiente tabla se obtienen los resultados de la clasificación de $5{,}000$ muestras, en las cuales no son variados los parámetros del kernel y no se usa alineamiento de kernels.

Parámetro	Valor
Media geométrica	0.571
Especificidad	0.594
Sensibilidad	0.688
Puntaje F1	0.563
Precisión	0.567
Exactitud	0.588

4.1.5. Alineación de kernels

Del mismo modo, se obtienen resultados para la variación de los 5 kernels con los parámetros descritos y con parámetros η seleccionados por medio del método del alineamiento de kernels. A pesar de ello, se puede notar una mejora de algunos indicadores en la variación de kernels. Cabe mencionar que los pesos de los kernels son logrados por medio del mismo alineamiento y la sumatoria de los mismos debe ser igual a 1. Para el caso de la siguiente tabla se usó un conjunto $\sigma = 0{,}1, 0{,}3, 0{,}5, 0{,}7, 0{,}9$.

Tabla 4-4: Resultados con variación de parámetros y $5{,}000$ muestras

Parámetro	Valor
Media geométrica	0.693
Especificidad	**0.711**
Sensibilidad	0.688
Puntaje F1	0.623
Precisión	0.648
Exactitud	0.660

4.1.6. Optimización cuadrática, metaheurística y combinación lineal

A continuación se muestran los resultados para $10{,}000$ datos y la combinación lineal descrita en el método propuesto, correspondiente a la combinación lineal de 10 kernels y la optimización de los pesos η por por optimización cuadrática. Hay que tomar en consideración que para esta parte de experimentos se usó un conjunto de pesos $\eta = 0{,}4041, 0{,}1711, 0{,}0659, 0{,}0012, 0, 0, 0, 0, 0, 0{,}3589$ debido a la optimización cuadrática.

Tabla 4-5: Resultados con parámetros $\sigma = 0{,}1, 0{,}3, 0{,}5, 0{,}7, 0{,}9, 2, 5, 7, 10$ multiplicados por el número de la mediana de los datos $14,342$ y $10{,}000$ muestras

Parámetro	Valor
Media geométrica	0.768
Especificidad	0.771
Sensibilidad	0.755
Puntaje F1	**0.793**
Precisión	0.765
Exactitud	0.762

En la tabla **4-5** se puede evidenciar la misma metodología trabajada con la base de datos Virus-Mentha [46] el cual se realizó con 5000 interacciones, se hace con el fin de obtener resultados para la metodología con ambas bases de datos.

Tabla 4-6: Resultados con parámetros $\sigma = 0{,}1, 0{,}3, 0{,}5, 0{,}7, 0{,}9, 2, 5, 7, 10$ con base de datos Virus Mentha para $5{,}000$ muestras

Parámetro	Valor
Media geométrica	0.723
Especificidad	0.662
Sensibilidad	0.772
Puntaje F1	0.693
Precisión	0.765
Exactitud	**0.782**

Finalmente en la tabla **4-6** se realiza una comparación de los métodos propuestos recientemente con respecto al problema de predicción de interacciones entre proteínas por medio de máquinas de vectores de soporte, descrito como método 1 [51], en el cual se usa una máquina de vectores de soporte con un kernel gausiano simple, lo comparan con el polinomial y finalmente lo contraponen con otros métodos, el de redes neuronales con kernel de base radial [57] se compone de un perceptrón multicapa donde usan kernel de base radial, en la tabla **4-7** con respecto al método propuesto en el presente documento.

Tabla 4-7: Comparación de resultados con otras metodologías

	Método 1	Método 2	Nuestra metodología
Puntaje F1	0.64	0.67	**0.76**
Precisión	0.77	0.80	0.79

4.1.7. Predicción interacción NS5A y proteínas hospederas.

Como parte final, se realizó un aislamiento de algunas proteínas humanas encargadas de el proceso biológico llamado fosforiliación, las cuales son las que usualmente activan ciertos procesos metabólicos y finalmente pueden demostrar que las interacciones causan una enfermedad determinada. Para ello se sacó un subconjunto de proteínas para fosforilación y se ponen en un escenario en conjunto con NS5A para un escenario hospedero-patógeno. En la tabla **4-8** se muestran los indicadores de uniprot de las proteínas potencialmente interactoras con NS5A.

Teniendo en cuenta la consideración anterior, se debe aclarar que debido a que no se pueden confirmar las interacciones por la no existencia de datos positivos, se procedió a realizar el trabajo de predicción de manera exploratoria en la cual se obtuvieron los siguientes resultados:

Tabla 4-8: Sobconjunto de proteínas relacionadas con fosforilación (Identificadores Uniprot)

Q86WV1	**Q9JM80**	Q9NWQ8	Q3UUV5	**P39687**
Q08050	Q13103	Q99567	P22059	Q9BXB4
Q15678	Q86UW1	O00443	Q92729	Q9Y6V0
Q99447	P12694	**Q15257**	Q9Y5N6	Q9Y6R0

4.1.8. Discusión

La diversidad de los resultados obtenidos en primera instancia se debe a la cantidad de ensayos realizados durante el proceso de ejecución de la investigación, estos comienzan necesariamente de la manera más simple en la cual pueda demostrar la veracidad del estudio. En primer lugar se muestran los resultados en la tabla **4-1**, los cuales muestran los resultados de la optimización metaheurística de los pesos, los cuales evidentemente no son lo suficientemente satisfactorios para seguir desarrollando la metodología en torno a una metaheurística, por tal motivo se se decidió realizar una optimización cuadrática de los pesos involucrados en los kernels, los cuales se ven reflejados en la tabla **4-5**, es así como en la presente metodología se propone una mejora por medio de la optimización cuadrática debido a los problemas obtenidos en la optimización metaheurística (Enjambres de partículas). A pesar de ello, se podría también enfocar la explicación teniendo en cuenta el parámetro **C** sin embargo, no es necesario debido a que la optimización de los pesos no tiene buenos resultados, por lo cual desde un principio podemos decir que el proceso tendrá dificultades. Con respecto a los resultados obtenidos y a los experimentos hechos, se puede evidenciar que en la tabla **4-2** se obtienen los resultados de los experimentos correspondientes a $5{,}000$ muestras y un conjunto de parámetros σ predefinidos, por lo cual se obtienen resultados que a pesar de ser consistentes, no satisfacen las necesidades en la predicción de interacciones entre proteínas. Uno de los posibles motivos es que el parámetro σ y η se están seleccionando de modo predeterminado y arbitrario, por lo tanto no es confiable.
Por otro lado, en la tabla **4-3** se obtuvieron resultados para el método correspondiente al método

que requiere un alineamiento de 5 kernels con parámetro σ fijo y $5{,}000$ datos, para lo cual se obtiene una mejora relativamente considerable con respecto a la asignación de parámetros de modo predeterminado, lo cual puede indicar que definitivamente se deben preparar los parámetros previamente con el fin de obtener mejores resultados en la matriz de confusión.
Adicionalmente, se obtienen los resultados para $10{,}000$ datos y un conjunto de parámetros σ seleccionados por medio de la multiplicación del conjunto de σ inicial descrito en la metodología y su multiplicación con la mediana ($X_{0,5}$) obtenida de los mismos datos. Los resultados que se evidencian en la tabla 4.1.4 son considerablemente consistentes debido a que los obtenidos en las pruebas anteriores resultan más varianbles y sensibles por cuenta de la variación de parámetros, lo que indica una clara diferencia entre las pruebas, hay una diferencia marcada entre las metodologías que no usan optimización cuadrática del parámetro η, que va directamente a la combinación lineal de los kernels. Por lo anterior, se puede decir que cada kernel realiza un aporte específico en la combinación lineal, y por lo tanto hay unos que deben tener un peso más alto que otros con el fin de obtener un kernel resultante equilibrado, para nuestro caso, los kernels que más aportan son los del medio siendo los sigma de $0{,}7, 0{,}9, 1, 3, 5$ los que más aportan en la combinación lineal.
Finalmente también se obtienen resultados referentes a otra base de datos llamada VirusMentha en la cual se destaca el valor de exactitud incluso por encima de las pruebas anteriores, esto podría probar que la metodología propuesta puede ser reproducible no solo con la base de datos inicial, si no también con otras bases de datos incluso con problemas similares al de la predicción de interacciones. También puede decir que el comportamiento del predictor no tiene una variación con respecto a la base de datos de entrada, lo cual hace el predictor más generalizable. También cabe aclarar que la cantidad de interacciones escogidas no fué de manera arbitraria, esto se debe a que cuando se escogía una cantidad mayor que 14.000 interacciones aproximadas, la memoria del equipo no podía continuar procesando los kernels. Con respecto a los pesos obtenidos $\eta = 0{,}4041, 0{,}1711, 0{,}0659, 0{,}0012, 0{,}0000, 0{,}0000, 0{,}0000, 0{,}0000, 0{,}0000, 0{,}3589$ se puede concluir que se le está dando más importancia a los primeros tres kernels y al último, por lo cual se evidencia un equilibrio llevando un gran peso al último kernel para realizar una compensación con los tres primeros

Por otro lado, los resultados relacionados con el subconjunto de 20 proteínas se destaca la interacción con 3 de las 20 proteínas, donde se resaltan $Q9JM80$, $P39687$, y $Q15257$ lo cual podría demostrar uno de los mecanismos de acción de la proteína NS5A del virus GBV-C, y probablemente sea aplicable a cualquier proteína de un virus. De este modo, las proteínas señaladas pueden tener una potencial interacción con la NS5A. De este modo, si queremos probar con cualquier proteína, basta con realizar un conjunto o subconjunto de proteínas de interés que tengan un potencial de interacción con la proteína de estudio y se podría realizar otro análisis exploratorio.

Se debe acotar que los resultados para el predictor pueden ser exclusivos debido al modo de escoger las proteínas que posiblemente podrían interactuar con la NS5A, esto se debe a que se escogió un subset de 20 proteínas, sin embargo se podría escoger un subset más grande en otros caso, aquí se puede confirmar la posibilidad de interacción con proteínas que tienen que ver con lin-

foma, pero se prevé que si el subset es más grande, se podrían tener muchas más interacciones. Asimismo, es necesario tener en cuenta que debido a la poca investigación de la interacciones de la proteína NS5A del virus GBV-C, no fué posible realizar una validación de interacciones reales, sino un trabajo exploratorio en el cual se propone un predictor que comienza de lo general a lo más específico.

5 Conclusiones

Se logró diseñar e implementar una metodología basada en aprendizaje por múltiples kernels en la cual a partir de las características de los datos se calcularon los kernels y se realizó una combinación lineal con un conjunto de parámetros planteada en la metodología propuesta, del mismo modo permite hallar el conjunto óptimo de pesos por medio de la optimización cuadrática ya que la optimización metaheurística no logra arrojar resultados satisfactorios para superar otras metodologías, lo que quiere decir que hay una influencia directa del modo de escoger los pesos sobre el resultado final, de este modo se obtienen los mejores resultados encontrados para nuestra metodología en la predicción de interacciones entre proteínas.

Como resultado de la implementación de los múltiples kernels se obtiene una matriz kernel que resulta de la combinación de los anteriores y sus respectivos pesos. La matriz resultante va a la integración con la máquina de vectores de soporte. Allí se realiza la optimización por enjambres de partículas con el fin de obtener el hiperparámetro C óptimo para el modelo de clasificación.

Finalmente, por medio de las medidas de desempeño asumidas en el presente documento y correspondientes a las medidas encontradas en la literatura para la predicción de interacciones proteína-proteína, el predictor de la metodología presente se encuentra concordante y competente con las metodologías actuales e incluso puede superar en ciertos casos algunas de las métricas de desempeño planteadas por otros predictores.

5.0.1. Trabajo futuro

A partir de los resultados obtenidos y a el predictor planteado, se pretende realizar la predicción a modo exploratorio con proteínas que tienen que ver no solamente con la proteína NS5A, sino extrapolar el estudio en problemas más generalizables como la búsqueda de potenciales proteínas interactoras con un organismo específico. Del mismo modo, se propone seguir trabajando con la metodología en diferentes modos de optimización para los hiperparámetros y preprocesamiento de los datos, con el fin de realizar una selección de las características más relevantes y resultantes de ésta metodología.

Bibliografía

[1] O. Keskin, N. Tuncbag, and A. Gursoy, "Predicting protein–protein interactions from the molecular to the proteome level," *Chemical reviews*, vol. 116, no. 8, pp. 4884–4909, 2016.

[2] E. Nourani, F. Khunjush, and S. Durmuş, "Computational prediction of virus–human protein–protein interactions using embedding kernelized heterogeneous data," *Molecular BioSystems*, vol. 12, no. 6, pp. 1976–1986, 2016.

[3] M. A. M. Hasan, S. Ahmad, and M. K. I. Molla, "Protein subcellular localization prediction using multiple kernel learning based support vector machine," *Molecular BioSystems*, vol. 13, no. 4, pp. 785–795, 2017.

[4] A. C. Nascimento, R. B. Prudêncio, and I. G. Costa, "A multiple kernel learning algorithm for drug-target interaction prediction," *BMC bioinformatics*, vol. 17, no. 1, p. 46, 2016.

[5] C. M. Chang, J. T. Stapleton, D. Klinzman, J. H. McLinden, M. P. Purdue, H. A. Katki, and E. A. Engels, "Gbv-c infection and risk of nhl among us adults," *Cancer research*, vol. 74, no. 19, pp. 5553–5560, 2014.

[6] J. C. Arroyave, F. H. Pujol, M. C. Navas, and F. M. Cortés-Mancera, "Interacción entre el virus de la inmunodeficiencia humana y el virus gb tipo-c durante el estado de co-infección," *Revista chilena de infectología*, vol. 30, no. 1, pp. 31–41, 2013.

[7] E. L. Mohr and J. T. Stapleton, "Gb virus type c interactions with hiv: the role of envelope glycoproteins," *Journal of viral hepatitis*, vol. 16, no. 11, pp. 757–768, 2009.

[8] R. Alcalde, A. Nishiya, J. Casseb, L. Inocêncio, L. A. Fonseca, and A. J. Duarte, "Prevalence and distribution of the gbv-c/hgv among hiv-1-infected patients under anti-retroviral therapy," *Virus research*, vol. 151, no. 2, pp. 148–152, 2010.

[9] Y. Tanaka, M. Mizokami, E. Orito, K. Ohba, T. Nakano, T. Kato, Y. Kondo, X. Ding, R. Ueda, S. Sonoda *et al.*, "Gb virus c/hepatitis g virus infection among colombian native indians." *The American journal of tropical medicine and hygiene*, vol. 59, no. 3, pp. 462–467, 1998.

[10] M. V. Alvarado-Mora, L. Botelho, A. Nishiya, R. A. Neto, M. S. Gomes-Gouvêa, M. F. Gutierrez, F. J. Carrilho, and J. R. Pinho, "Frequency and genotypic distribution of gb virus c (gbv-c) among colombian population with hepatitis b (hbv) or hepatitis c (hcv) infection," *Virology journal*, vol. 8, no. 1, p. 345, 2011.

[11] J. T. Stapleton, S. Foung, A. S. Muerhoff, J. Bukh, and P. Simmonds, "The gb viruses: a review and proposed classification of gbv-a, gbv-c (hgv), and gbv-d in genus pegivirus within the family flaviviridae," *Journal of general virology*, vol. 92, no. 2, pp. 233–246, 2011.

[12] S. Jones and J. M. Thornton, "Principles of protein-protein interactions," *Proceedings of the National Academy of Sciences*, vol. 93, no. 1, pp. 13–20, 1996.

[13] M. D. Leiserson, F. Vandin, H.-T. Wu, J. R. Dobson, J. V. Eldridge, J. L. Thomas, A. Papoutsaki, Y. Kim, B. Niu, M. McLellan *et al.*, "Pan-cancer network analysis identifies combinations of rare somatic mutations across pathways and protein complexes," *Nature genetics*, vol. 47, no. 2, pp. 106–114, 2015.

[14] M. R. Arkin, Y. Tang, and J. A. Wells, "Small-molecule inhibitors of protein-protein interactions: progressing toward the reality," *Chemistry & biology*, vol. 21, no. 9, pp. 1102–1114, 2014.

[15] V. S. Rao, K. Srinivas, G. Sujini, and G. Kumar, "Protein-protein interaction detection: methods and analysis," *International journal of proteomics*, vol. 2014, 2014.

[16] R. Halehalli and H. A. Nagarajaram, "Molecular principles of human virus protein-protein interactions," *Bioinformatics*, p. btu763, 2014.

[17] A. J. Wilson, "Inhibition of protein–protein interactions using designed molecules," *Chemical Society Reviews*, vol. 38, no. 12, pp. 3289–3300, 2009.

[18] T. Pahikkala, A. Airola, S. Pietilä, S. Shakyawar, A. Szwajda, J. Tang, and T. Aittokallio, "Toward more realistic drug–target interaction predictions," *Briefings in bioinformatics*, vol. 16, no. 2, pp. 325–337, 2014.

[19] G. E. Schulz and R. H. Schirmer, *Principles of protein structure*. Springer Science & Business Media, 2013.

[20] S. J. Lippard and J. M. Berg, *Principles of bioinorganic chemistry*. University Science Books, 1994.

[21] T. E. Creighton, *Proteins: structures and molecular properties*. Macmillan, 1993.

[22] C.-I. Brändén and J. Tooze, *Introduction to protein structure*. Taylor & Francis, 1999.

[23] O. Keskin, A. Gursoy, B. Ma, R. Nussinov *et al.*, "Principles of protein-protein interactions: what are the preferred ways for proteins to interact?" *Chemical reviews*, vol. 108, no. 4, p. 1225, 2008.

[24] E. M. Phizicky and S. Fields, "Protein-protein interactions: methods for detection and analysis." *Microbiological reviews*, vol. 59, no. 1, pp. 94–123, 1995.

[25] E. M. Marcotte, I. Xenarios, A. M. Van der Bliek, and D. Eisenberg, "Localizing proteins in the cell from their phylogenetic profiles," *Proceedings of the National Academy of Sciences*, vol. 97, no. 22, pp. 12 115–12 120, 2000.

[26] W. S. Valdar and J. M. Thornton, "Protein–protein interfaces: analysis of amino acid conservation in homodimers," *Proteins: Structure, Function, and Bioinformatics*, vol. 42, no. 1, pp. 108–124, 2001.

[27] J. R. Bock and D. A. Gough, "Predicting protein–protein interactions from primary structure," *Bioinformatics*, vol. 17, no. 5, pp. 455–460, 2001.

[28] R. Jansen, H. Yu, D. Greenbaum, Y. Kluger, N. J. Krogan, S. Chung, A. Emili, M. Snyder, J. F. Greenblatt, and M. Gerstein, "A bayesian networks approach for predicting protein-protein interactions from genomic data," *science*, vol. 302, no. 5644, pp. 449–453, 2003.

[29] V. Vapnik, *The nature of statistical learning theory*. Springer science & business media, 2013.

[30] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[31] V. N. Vapnik and V. Vapnik, *Statistical learning theory*. Wiley New York, 1998, vol. 1.

[32] N. Cristianini, J. Shawe-Taylor *et al.*, *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press, 2000.

[33] I. Steinwart and A. Christmann, *Support vector machines*. Springer Science & Business Media, 2008.

[34] M. Gönen and E. Alpaydın, "Multiple kernel learning algorithms," *Journal of Machine Learning Research*, vol. 12, no. Jul, pp. 2211–2268, 2011.

[35] B. Scholkopf and A. J. Smola, *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2001.

[36] D. Li, J. Wang, X. Zhao, Y. Liu, and D. Wang, "Multiple kernel-based multi-instance learning algorithm for image classification," *Journal of Visual Communication and Image Representation*, vol. 25, no. 5, pp. 1112–1117, 2014.

[37] S. Qiu and T. Lane, "A framework for multiple kernel support vector regression and its applications to sirna efficacy prediction," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 6, no. 2, pp. 190–199, 2009.

[38] D. P. Bertsekas, *Nonlinear programming*. Athena scientific Belmont, 1999.

[39] O. Von Stryk and R. Bulirsch, "Direct and indirect methods for trajectory optimization," *Annals of operations research*, vol. 37, no. 1, pp. 357–373, 1992.

[40] E.-G. Talbi, *Metaheuristics: from design to implementation*. John Wiley & Sons, 2009, vol. 74.

[41] S. Intelligence, "Particle swarm optimization," *MCCAFFREY, James.[online].[cit. 2014-05-20]. Dostupné z: http://msdn. microsoft. com/en-us/magazine/hh335067. aspx*, 2007.

[42] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium on*. IEEE, 1995, pp. 39–43.

[43] K.-L. Du and M. Swamy, "Particle swarm optimization," in *Search and Optimization by Metaheuristics*. Springer, 2016, pp. 153–173.

[44] L. Salwinski, C. S. Miller, A. J. Smith, F. K. Pettit, J. U. Bowie, and D. Eisenberg, "The database of interacting proteins: 2004 update," *Nucleic acids research*, vol. 32, no. suppl 1, pp. D449–D451, 2004.

[45] R. Apweiler, A. Bairoch, C. H. Wu, W. C. Barker, B. Boeckmann, S. Ferro, E. Gasteiger, H. Huang, R. Lopez, M. Magrane *et al.*, "Uniprot: the universal protein knowledgebase," *Nucleic acids research*, vol. 32, no. suppl 1, pp. D115–D119, 2004.

[46] A. Calderone, L. Licata, and G. Cesareni, "Virusmentha: a new resource for virus-host protein interactions," *Nucleic acids research*, vol. 43, no. D1, pp. D588–D592, 2014.

[47] W. Li and A. Godzik, "Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences," *Bioinformatics*, vol. 22, no. 13, pp. 1658–1659, 2006.

[48] J. D. Thompson, T. Gibson, D. G. Higgins *et al.*, "Multiple sequence alignment using clustalw and clustalx," *Current protocols in bioinformatics*, pp. 2–3, 2002.

[49] F.-E. Eid, M. ElHefnawi, and L. S. Heath, "Denovo: virus-host sequence-based protein–protein interaction prediction," *Bioinformatics*, vol. 32, no. 8, pp. 1144–1150, 2016.

[50] S. Kawashima and M. Kanehisa, "Aaindex: amino acid index database," *Nucleic acids research*, vol. 28, no. 1, pp. 374–374, 2000.

[51] H. Guo, B. Liu, D. Cai, and T. Lu, "Predicting protein–protein interaction sites using modified support vector machine," *International Journal of Machine Learning and Cybernetics*, vol. 9, no. 3, pp. 393–398, 2018.

[52] G. Taherzadeh, Y. Yang, T. Zhang, A. W.-C. Liew, and Y. Zhou, "Sequence-based prediction of protein–peptide binding sites using support vector machine," *Journal of computational chemistry*, vol. 37, no. 13, pp. 1223–1229, 2016.

[53] C. Cortes, M. Mohri, and A. Rostamizadeh, "Two-stage learning kernel algorithms." in *ICML*, 2010, pp. 239–246.

[54] D. Korkinof and Y. Demiris, "Multi-task and multi-kernel gaussian process dynamical systems," *Pattern Recognition*, vol. 66, pp. 190–201, 2017.

[55] J. A. Rodríguez, J. Jaramillo-Garzón, and J. Arroyave-Ospina, "Prediction of protein-protein interactions through support vector machines," in *Signal Processing, Images and Computer Vision (STSIVA), 2015 20th Symposium on*. IEEE, 2015, pp. 1–5.

[56] J. Arango-Rodriguez, A. Cardona-Escobar, J. Jaramillo-Garzon, and J. Arroyave-Ospina, "Machine learning based protein-protein interaction prediction using physical-chemical representations," in *Signal Processing, Images and Artificial Vision (STSIVA), 2016 XXI Symposium on*. IEEE, 2016, pp. 1–5.

[57] Y. Chen, J. Xu, B. Yang, Y. Zhao, and W. He, "A novel method for prediction of protein interaction sites based on integrated rbf neural networks," *Computers in biology and medicine*, vol. 42, no. 4, pp. 402–407, 2012.