Centro de Investigación Científica y de Educación Superior de Ensenada, Baja California

Maestría en Ciencias
en Ciencias de la Computación

Evaluación y diseño de predictores de interacción miARN-proteína

Tesis

para cubrir parcialmente los requisitos necesarios para obtener el grado de

Maestro en Ciencias

Presenta:

Nephtali Dicochea Moreno

Ensenada, Baja California, México

2018

Tesis defendida por

Nephtali Dicochea Moreno

y aprobada por el siguiente Comité

Dr. Carlos Alberto Brizuela Rodríguez

Director de tesis

Dr. Israel Marck Martínez Pérez

Dr. Sergio Andrés Águila Puentes

Dr. Jesús Favela Vara

Coordinador del Posgrado en Ciencias de la Computación

Dra. Rufina Hernández Martínez

Director de Estudios de Posgrado

Resumen de la tesis que presenta **Nephtali Dicochea Moreno** como requisito parcial para la obtención del grado de Maestro en Ciencias en Ciencias de la Computación.

Evaluación y diseño de predictores de interacción miARN-proteína

Resumen aprobado por:

Dr. Carlos Alberto Brizuela Rodríguez
Director de tesis

La estructura tridimensional de una proteína nos da una idea de su función biológica y de las posibles interacciones subyacentes con otras biomoléculas. Uno de los principales desafíos en los últimos años ha sido desentrañar las características ocultas que permiten y propician estas interacciones. Además, en las últimas dos décadas, con la creciente cantidad de datos de secuenciación de alto rendimiento se ha observado que el papel de las proteínas de unión a ARN (RBPs) implicadas en su procesamiento, es un factor clave en enfermedades mortales como el cáncer. Por lo tanto, tener un atlas completo de RBPs implicado en la vía de procesamiento del miARN se considera escencial. Además, con este propósito se han presentado enfoques con una amplia gama de técnicas, computacionales y de laboratorio húmedo, algunos de ellos con buenos resultados. En Sheng y Zhou (2013) se propone un modelo computacional para el problema de la clasificación de secuencias por su capacidad o no de acloparse a miARN. Siguiendo este modelo computacional, en esta tesis se analiza la eficacia de dos clasificadores basados en aprendizaje de máquina, uno basado en un esquema supervisado y el otro en uno semi-supervisado, los cuales podrían ser aplicados para identificar nuevas RBPs usando información de su estructura tridimensional. Así mismo, se comparan selectores de características con el propósito de mejorar la calidad de clasificación lograda en Sheng y Zhou (2013). Además de que los resultados experimentales de una combinación de datos, características y clasificador lograron mejorar todas las medidas de calidad de clasificación con respecto al trabajo de Sheng y Zhou (2013), en este trabajo se presenta evidencias de que un obstáculo prioritario que se debe resolver para avanzar en la predicción de RBPs es la escasez de ejemplos disponibles.

Palabras clave: bioinformatica, miARN, proteina, interaccion, aprendizaje de maquina

Abstract of the thesis presented by **Nephtali Dicochea Moreno** as a partial requirement to obtain the Master of Science degree in Name of the Degree.

Evaluation and Design of miRNA-protein interaction Predictors

Abstract approved by:

Dr. Carlos Alberto Brizuela Rodríguez
Thesis Director

The three-dimensional structure of a protein gives us an insight into its biological function and about the possible underlying interactions with other biomolecules. One of the main challenges in the last few years has been to unravel the hidden features that allow and encourage these interactions. In addition, over the last two decades, with the increasing amount of high-throughput sequencing data, it has been observed that the role of RNA-binding proteins (RBPs) involved in miRNA's processing is a key factor in deadly diseases such as cancer. Therefore, having a complete atlas of RBPs involved in the processing of the miRNA is considered essential. In addition, with this purpose, approaches have been presented on a wide range of technical, computational and laboratory techniques, some of them with good results. Sheng and Zhou (2013) proposed a computational model for the classification problem of sequences according their ability to bind to miRNAs.
Following this computational model, this thesis analyzes the effectiveness of two classifiers based on machine learning, one based on a supervised scheme and the other on a semi-supervised one, which could be applied to identify new RBPs using information from their three-dimensional structure. Likewise, feature selectors are compared with the purpose of improving the classification quality achieved in Sheng and Zhou (2013). In addition to the experimental results of a combination of data, features, and classifier, that show that all classification quality measures were improved with respect to Sheng and Zhou (2013), in this work we present evidence that a key obstacle that must be solved in order to advance in the prediction of RBPs is the lack of available examples.

Keywords: Bioinformatics, miRNA, protein, interaction, machine learning

Dedicatoria

A Consuelo y Obed, quienes aún más que padres, han sido mis más grandes amigos y soporte...

Agradecimientos

Al Dr. Carlos Brizuela, por su incanzable búsqueda de mi realización académica y personal...

A Armando Beltran, quien siempre con una sonrisa y vocación de maestro nunca vaciló en dar de sí lo que sabía...

Al CONACYT, por su enorme apoyo a la juventud Mexicana...

Tabla de contenido

Página

Tabla de contenido (continuación)

Lista de figuras

Figura Página

Lista de tablas

Tabla Página

Lista de tablas (continuación)

Capítulo 1. Introducción

1.1. Marco teórico y motivación

Los exosomas son pequeñas vesículas de entre 40 y 100 nm de diámetro, los cuales se encuentran en los cuerpos endosómicos o multivesiculares (MVB) (Zhang *et al.*, 2015). Su principal función es el intercambio de información entre células. Gracias al reciente avance en secuenciación de ADN y la consecuente generación de transcritos (e.g. RNA-Seq), diversos estudios (McKenzie *et al.*, 2016; Cha *et al.*, 2015; Silverman *et al.*, 2010), incluidos algunos relacionados con cáncer, han indicado que los exosomas están involucrados en el esparcimiento de enfermedades letales. Esa es la razón por la cual instituciones como “the National Institutes of Health (NIH)” actualmente hacen conciencia acerca del problema dentro de la comunidad científica. En 2007, un estudio realizado por la Universidad de Gotemburgo (Valadi *et al.*, 2007) descubrió que estas vesículas contienen secuencias de ARN no codificante llamadas microARN (miARN), dichas secuencias constituyen uno de los mecanismos naturales de las células para el control de expresión de secuencias de ARN mensajero (mARN). Gracias a las metodologías de purificación, el cargamento de exosomas ha podido ser recuperado para su secuenciación y finalmente obtenido su transcriptoma. En una estrategia que recuerda a la mitología Griega del caballo de Troya, los exosomas son utilizados para transportar mensajes codificados en ARN, provenientes de células enfermas, y una vez que alcanzan el medio extracelular, son engullidos por células objetivo para modificar su expresión génica. Por ejemplo: tomar el control del ciclo celular a través de la represión de proteínas involucradas en dicho ciclo y producir alteraciones en los puntos de control de dicho proceso.

Las secuencias de miARN dentro de exosomas desempeñan un papel importante en la progresión de diversas enfermedades, y se ha documentado (Ribeiro *et al.*, 2013) que estas pueden estimular la angiogénesis en el caso del cáncer y, posteriormente, de la metástasis. Hoy en día, la detección experimental de secuencias de miARN empaquetadas dentro de exosomas ha sido validada como un método viable para detectar enfermedades en el cuerpo humano. En investigaciones recientes (Lin *et al.*, 2015; An *et al.*, 2015) se sugiere que diferentes enfermedades favorecen diferentes empaquetamientos de secuencias de miARN en exosomas. Estas biomoléculas (e.g. miARN) relacionadas con una enfermedad en particular se llaman biomarcadores. Se sabe que

los cambios conformacionales postranscripcionales y diversos motivos en las secuencias de miARN juegan un papel clave en la interacción de proteínas con las secuencias para su posterior empaquetamiento dentro de exosomas. A pesar de grandes esfuerzos, los mecanismos que impulsan este empaquetamiento aún no son del todo claros, y todavía, son caros en términos de tiempo y recursos financieros cuando se estudian en un laboratorio húmedo.

Los enfoques computacionales han sido previamente mostrados como efectivos cuando un problema biológico comprende una gran cantidad de trabajo de laboratorio húmedo y una gran cantidad de datos (Markowetz, 2017), tal como el caso del problema de interacción miARN-proteína.

El Aprendizaje de Máquina (AM) puede ser definido como “el proceso que puede ejecutar una computadora para aprender de la experiencia previa con respecto a alguna clase específica de tareas y tener medidas de calidad del rendimiento en dicho proceso” (Liu *et al.*, 2013). Un problema de clasificación, como lo es la interacción miARN-proteína, puede ser modelado computacionalmente como un problema en el que las caracterśticas relevantes (las que permiten la discriminación entre clases pertenecientes de ejemplos) se extraen de las secuencias de proteínas (datos) y una computadora es entrenada para aprender de esas características y predecir más conocimiento a partir de observaciones previas.

Un trabajo reciente (Sheng y Zhou, 2013) sienta las bases para la clasificación de secuencias de proteína de acuerdo a su capacidad de acoplamiento con miARN. En este trabajo de investigación se destacan tres aspectos que motivan la continuación de la investigación acerca de las Proteńas que se acoplan a ARN (RBPs).

Primero, en dicho artículo, se describe un enfoque basado en una Máquina de Soporte Vectorial (SVM) laplaciana, aprovechando información de ejemplos no etiquetados (aprendizaje semi-supervisado). Tal enfoque considera la capacidad de asignar un peso a las clases, sin embargo, las medidas de calidad reportadas fueron 0.26 ±0.02 como valor-F, 0.63±0.09 como sensibilidad, 0.16±0.02 como precisión, 0.26±0.02 como MCC y 0.80±0.02 como exactitud.

Segundo, los autores no reportan el uso de ningún selector de características más allá de probar las distintas combinaciones de subconjuntos de características propuestas por ellos mismos.

Tercero, dado su modelo computacional y técnica de extracción de datos, solo reportan

61 ejemplos positivos provenientes de complejos de interacción miARN-proteína.

1.2. Definición del problema

Dado el modelo computacional de Sheng y Zhou (2013) para el problema de la clasificación de secuencias según su habilidad para acoplarse a miARN, dado secuencias de proteína en forma de 11-meros, las cuales, son etiquetadas como positivos, negativos o no etiquetados, y dado un universo de descriptores moleculares, el problema consiste en encontrar por medio de diversas propuestas de selectores de características, el subconjunto de estas que maximice las medidas de calidad de clasificación del modelo basado en aprendizaje semi-supervisado y utilizado por Sheng y Zhou (2013).

1.3. Objetivos generales

Determinar a través de experimentación computacional, el desempeño de diversos selectores de características usados en un clasificador SVM que busca discriminar secuencias de proteína con la habilidad de acoplarse a miARN de aquellas que no tienen dicha propiedad (**Figura 1**).

1.4. Objetivos especificos

1. Establecer criterios para la selección de ejemplos de entrenamiento.
2. Definir un conjunto de casos positivos y negativos a ser usados en el entrenamiento y validación.
3. Seleccionar criterios de comparación de calidad para la predicción de interacción miARN-proteína.
4. Determinar una jerarquía de calidad de los métodos del estado del arte para predecir interacciones miARN-proteína de acuerdo a los criterios establecidos en el objetivo anterior.
5. Determinar el modelo de aprendizaje de máquina que mejor se ajuste al problema basado en el objetivo anterior.
6. Determinar la efectividad de la metodología basada en MOEA/D descrita en Paul y Das (2015), la modificación de las funciones objetivo según Beltrán *et al.* (2017)

y la técnica de envoltura para selección de características descrita en Kohavi y John (1997) e impementada por Beltrán *et al.* (2017).

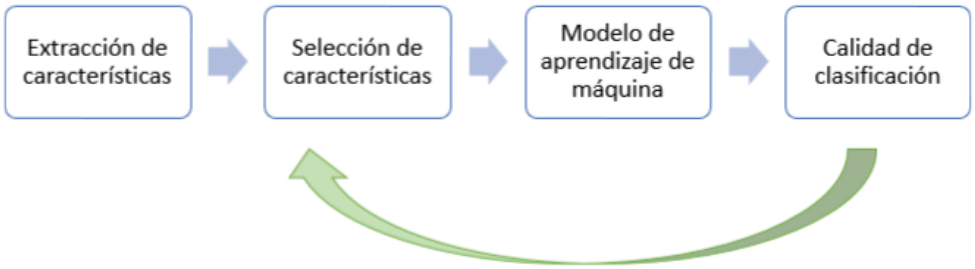

Figura 1. Flujo del modelado de aprendizaje de máquina tipo envoltura.

1.5. Metodología de solución

Para lograr el objetivo propuesto en este trabajo se propone la siguiente metodología: tomar como base el trabajo de Sheng y Zhou (2013) y replicarlo, este describe el problema de clasificación de secuencias de proteína de acuerdo a su capacidad de acoplarse o no a miARN. En dicha investigación se menciona el problema de la falta de ejemplos de entrenamiento etiquetados positivos, por lo que el primer paso en esta tesis es una búsqueda de estructuras de proteína en complejo con miARN en dos bases de datos públicas, RCSB PDB y UniProt, tal como se describe en dicho trabajo. El siguiente paso es crear un modelo supervisado de aprendizaje de máquina programado con la librería SciKit learn (Pedregosa *et al.*, 2011). A continuación se evaluan distintos trabajos de investigación acerca de descriptores moleculares para interacción ARN-proteína, tal como el de Jahandideh y Srinivasasainagendra (2012), con el objetivo de evaluar su aporte a los clasificadores usados en este trabajo. Después, se comparan tres métodos de selección de características: la metodología basada en MOEA/D descrita en Paul y Das (2015), la modificación de las funciones objetivo según Beltrán *et al.* (2017) y la técnica de envoltura para selección de características diseñada por Beltran (2015). El primero, define el problema de selección de características como un problema de optimización en el que sus dos funciones objetivo son la distancia intra-clase (distancia entre elementos de la misma clase), y la distancia inter-clase (distancia entre elementos de distintas clases). Se busca minimizar la distancia intra-clase y maximizar la distancia inter-clase, con el propósito de mejorar la clasificación por medio de hiperplanos en SVMs. El segundo, promete mejorar las me-

didas de calidad al ajustar el modelo de selección de características a un problema de clasificación de secuencias de proteína. Y el tercero, en el cual se utiliza un algoritmo de optimización, con el propósito de minimzar o maximizar funciones de calidad. En el caso de clasificación, esta función a maximizar puede ser el Coeficiente de Correlación de Mattews (MCC), el cual mide la calidad de clasificaciones binarias.
También, se obtienen los descriptores moleculares incluidos en el software ProtDCal (Ruiz-Blanco *et al.*, 2015) y se evalua su aporte a la calidad de clasificación, siguiendo la metodología usada para evaluar los demás descriptores.
Por último, se realiza una comparación de dichas soluciones y se hace un análisis de resultados en términos de calidad de clasificación.

1.6. Organización de la tesis

- El **Capítulo 1** comprende los antecedentes, la motivación y la definición del problema, así como los objetivos generales, específicos de este trabajo, la metodología y la organización de la tesis.
- El **Capítulo 2** presenta el marco teórico, este está dividido por la base biológica, la cual comprende el fundamento biológico, y el fundamento computacional, el cual comprende técnicas de aprendizaje de máquina y de selección de características.
- El **Capítulo 3** trata acerca de la metodología propuesta para abordar el problema definido, el proceso de obtención de conjuntos de datos, los conjuntos de características, el diseño experimental, los resultados y la discusión de las mismas.
- En el **Capítulo 4** se detallan los experimentos realizados y sus resultados.
  En el **Capítulo 5**, se presenta una discusión acerca de los resultados, el logro de objetivos y las conclusiones.
- Por último, en el **Apéndice A**, se listan las secuencias de entrenamiento utilizadas en este trabajo.

Capítulo 2. Marco teórico

2.1. Fundamento biológico

En una investigación reciente (Koppers *et al.*, 2014), en la cual se usaron células B como base para el estudio, se describen cambios postranscripcionales en secuencias de miARN, los cuales controlan la localización de secuencias de miARN y determinan que éstas se encuentren, ya sea en el medio celular o en exosomas. En dicho trabajo se sugiere que la localización de los miARNs no es el resultado de un proceso aleatorio de difusión y las diferencias entre las dos ubicaciones (exosoma y citoplasma) se identificaron en un panel de células B, seguido de un proceso de purificación para obtener sus exosomas y sus contenidos. Después, se identificó que la adenilación de la cola poli(A) en el extremo 3’ en las secuencias de miARN promueve su sobreexpresión en el citoplasma, mientras que el proceso de uridilación de una cadena poli(U) promueve su sobrerepresentación en los exosomas. Dada la importancia de los cambios conformacionales antes mencionados en la migración y localización de dichas secuencias, es necesario un método para la identificación del tipo de mecanismos de interacción entre los miARNs y sus transportadores.

2.1.1. Proteínas

Una proteína es un polímero lineal construido a partir de 20 aminoácidos diferentes. El tipo y la secuencia de aminoácidos en una proteína están especificados por el ADN en la célula que los produce. Esta secuencia de aminoácidos es esencial ya que determina la estructura general y la función de una proteína Mozo (2008a).

Una proteína tiene varias funciones, puede servir como material estructural, enzimas, transportadores, anticuerpos, o como reguladores de la expresión proteíca. Una proteína puede clasificarse según su forma y funciones principales: puede ser una proteína globular como la mayoría de las enzimas, proteínas fibrosas que tienen un papel estructural; y proteínas de membrana que sirven como receptores o canales para que la molécula polar o cargada pase a través de la membrana celular Mozo (2008a).

2.1.2. ARN codificante

2.1.2.1. ARN mensajero (mARN)

El ARN mensajero (ARNm) es una molécula, la cual transporta secuencias de ADN en del núcleo celular a los sitios de síntesis de proteína en el citoplasma (ribosomas) (Mozo, 2008b). La molécula que eventualmente se conocería como ARNm fue descrita por primera vez en 1956 por los científicos Elliot Volkin y Lazarus Astrachan. Cada molécula de ARNm codifica la información para una proteína (o más de una proteína en bacterias), con cada secuencia de tres bases que contienen nitrógeno en el ARNm, las cuales especifican la incorporación de un aminoácido particular dentro de una secuencia de proteína (péptido). Las moléculas de ARNm son transportadas a través de la membrana nuclear hacia el citoplasma, donde son traducidas por el ARNr de los ribosomas. En procariotas (organismos que carecen de un núcleo), los ARNm contienen una copia exacta transcrita de la secuencia de ADN original con un grupo terminal 5’-trifosfato y un residuo 3’-hidroxilo. En eucariotas (organismos que poseen un núcleo claramente definido), las moléculas de ARNm son más elaboradas. El residuo de 5’-trifosfato se esterifica adicionalmente, formando una estructura llamada cubierta. En los extremos 3’, los ARNm eucariotas típicamente contienen largos polimeros de residuos de adenosina (poliA) que no están codificados en el ADN pero se añaden enzimáticamente después de la transcripción con propósitos de preservación de la misma.

2.1.3. ARN no codificante

2.1.3.1. ARN no codificante largo

Los ARNs no codificantes largos forman complejos con proteínas modificadoras de la cromatina y reclutan actividad catalítica en sitios específicos del genoma (Mozo, 2008b), modificando así, los estados de la cromatina e influyendo en la expresión genética.

2.1.3.2. ARN pequeño de interferencia (siARN)

Los ARN interferentes cortos (siRNA) funcionan de forma similar a los miARN para mediar el silenciamiento génico postranscripcional (GTPT) como resultado de la degra-

dación del ARNm (Mozo, 2008b). Además de esta función, también se ha demostrado que los siRNAs inducen la formación de heterocromatina a través de un complejo de silenciamiento transcripcional inducido por ARN (RITS).

2.1.3.3. ARN que interacciona con Piwi (piARN)

Los ARN que interaccionan con Piwi (piRNA) reciben este nombre debido a su interacción con la familia de proteínas piwi (Mozo, 2008b). La función principal de estas moléculas de ARN implica la regulación de la cromatina y la supresión de la actividad del transposón en la línea germinal y las células somáticas. Los piRNAs que son antisentido para transposones expresados se dirigen y escinden el transposón en complejos PIWI-proteína. Esto genera piRNAs adicionales que se dirigen y cortan transposones adicionales.

2.1.3.4. microARN

Los microARN (miARN) representan una clase extensa de ARNs endógenos pequeños que regulan la expresón génica en el nivel postranscripcional mediante el corte del ARN mensajero (ARNm) o la inhibición de la traducción (Mestrovic, 2015) La mayoría de los organismos multicelulares codifican de docenas a cientos de genes miARN, los cuales desempeñan un papel importante en el control de los procesos biológicos. Dado que una gran fracción de los genes que codifican proteínas están bajo el control directo del miRNA, la producción apropiada de miRNAs específicos en el momento correcto y en el lugar correcto es fundamental para la mayoría de las rutas reguladoras de genes. Investigaciones recientes también han revelado la regulación del recambio maduro de miARN en el sistema inmune. La biogénesis de los miRNAs se encuentra sujeta a una regulación compleja tanto a nivel transcripcional como postranscripcional. Dado que pequeñas desviaciones en los niveles de miARN pueden alterar la regulación de diferentes genes blanco, un control adecuado de la biogénesis de miARN es esencial para el mantenimiento de la homeostasis celular normal.

La biogénesis de miRNA representa una serie de procesos secuenciales para generar miRNAs maduros. Los miRNAs primarios (pri-miRNAs) se transcriben inicialmente a partir de las regiones intergénicas o intragénicas mediante el RNA polimerasa II. Estos pri-miRNAs son posteriormente extirpados mediante ribonucleasas específica de ARN de doble hebra en el núcleo para producir pre-miRNAs con estructuras “hairpin”

(también llamadas precursores de horquilla).

Los “hairpins” se exportan desde el núcleo mediante la proteína Exportin-5, donde los pre-miRNAs se procesan adicionalmente mediante la proteína Dicer (endonucleasa RNasa III) en miARN dúplex de 21-14 nucleótidos de longitud. La cadena designada para ser la secuencia madura se carga luego en proteínas Argonauta, formando el complejo de silenciamiento inducido por miARN (RISC). Los miRNAs luego guían dichos complejos formados a mRNAs específicos usando un emparejamiento de bases imperfecto con el fin de regular negativamente su expresión a través de la desestabilización de mRNA o la represión traduccional. Es importante señalar que las proteínas accesorias pueden regular la biogénesis de miARN en cada uno de estos pasos, por lo que se puede garantizar la homeostasis de los miARN. Los mecanismos alternativos de represión de ARNm también pueden explotarse dependiendo de la estructura secundaria del ARN o de las proteínas efectoras asociadas con pares específicos de miARN y ARNm.

2.1.4. Motivos y su importancia en la interacción de proteínas

En genética (Squadrito *et al.*, 2014), un motivo es una secuencia de nucleótidos o aminoácidos, que restringe los patrones recurrentes que se presume tienen una función biológica. Gracias a la tecnología de secuenciación de ARN de alto rendimiento (RNA-Seq), se identificaron motivos específicos en las secuencias de miARN, incluido el tetranucleótido GGAG (Villarroya-Beltri *et al.*, 2013) y se observó que estaban sobrerepresentados en secuencias de miARN obtenidas de exosomas localizadas en células T. Este tetra-nucleótido ha sido reconocido junto con la proteína sumoilada hnRNPA2B1, la cual se sugiere (Cha *et al.*, 2015; Squadrito *et al.*, 2014; Villarroya-Beltri *et al.*, 2013) que dirige el tráfico de secuencias de miARN a cuerpos multivesiculares (MVBs). Se puede concluir que las secuencias específicas se unen a las proteínas “guía”, que tienen un papel importante en la dirección del flujo de miRNAs al endosoma. Esto sienta un precedente como metodología: buscar secuencias específicas que promuevan la unión de secuencias y proteínas implicadas en el tráfico de miRNAs a MVBs o endosomas con el objetivo de su posterior empaquetamiento dentro de exosomas y su liberación al medio extracelular. El trabajo de Batagov *et al.* (2011) describe los actuadores cis (motivos), llamados códigos zip y sus interacciones con los actuadores trans (proteínas que son parte de la maquinaria de transporte celular). Estas interacciones

pueden ocurrir a nivel de secuencia, estructura secundaria o ambas. Además, se ha encontrado que los códigos zip más conocidos se localizan predominantemente en las regiones 3’ UTR del ARNm. Sin embargo, a pesar de la intensa investigación experimental para su reconocimiento, se ha demostrado que los métodos computacionales tienen desventajas considerables debido a que la misma secuencia de ARN puede ser reconocida por múltiples proteínas que actúan en trans, y poseen múltiples modos de reconocimiento del objetivo.

2.1.5. La importancia de la proteína hnRNPA2B1

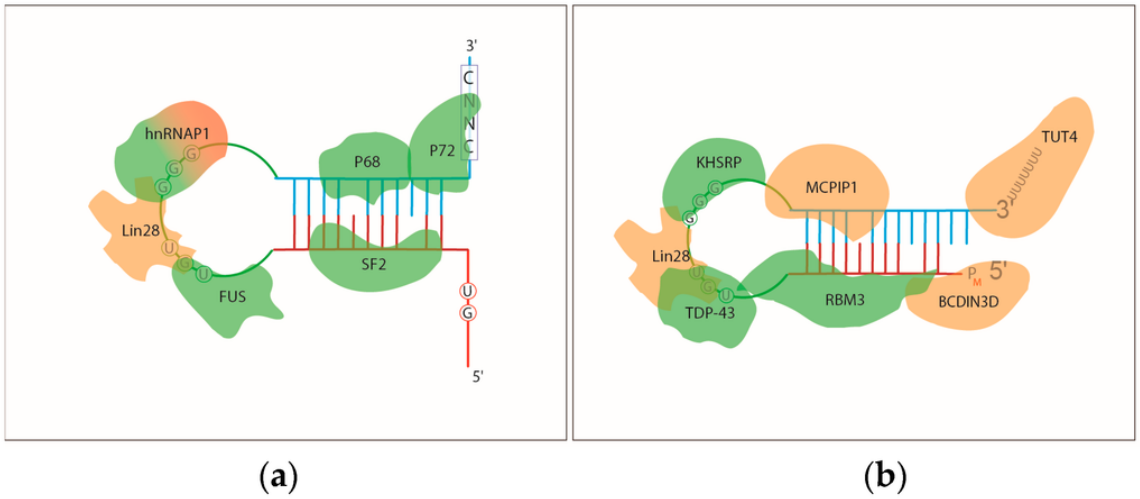

Figura 2. Interacción miRNA-proteína. **a)** Proteínas asociadas con pri-miRNAs) y textbfb) Proteínas asociadas con pre-miRNAs citepr13.

La proteína hnRNPA2B1 (**Figura 2a**) fue identificada experimentalmente como un transportador de miARN (Villarroya-Beltri *et al.*, 2013). En Villarroya-Beltri *et al.* (2013) se identificaron motivos, los cuales estaban sobrerepresentados en secuencias de miARN exosomal, estos tienen la función de guiar el empaquetamiento de secuencias hacia exosomas, y a su vez muestran que gracias a su mutagénesis permiten la modulación del miARN dentro de estas vesículas. La ribonucleoproteína nuclear heterogénea A2B1 (hnRNPA2B1), la cual es omnipresente, expresa la proteína de unión a ARN que controla el transporte y la localización subcelular de miRNA específicos en las neuronas y el tráfico del ARN genómico del Virus de Inmunodeficiencia Humana (VIH). De esta forma, se concluyó que la proteína hnRNPA2B1 se une a un subconjunto específico de miRNA a través de sus motivos que los empacan en exosomas (EXOmo-

tifs). Por otro lado, hnRNPA2B1 está sumoilado, y esta modificación postraduccional controla la unión de hnRNPA2B1-miRNA. Una forma en que se concluyó que existen miRNAs específicos que se empaquetan dentro de exosomas, fue con el análisis de datos de experimentos de microarreglos en dos estados celulares: producción activada y desactivada de miRNAs específicos. Posteriormente, los linfoblastos humanos T y sus exosomas se tomaron como punto de partida para el análisis (se pueden encontrar más detalles en Gene Expression Omnibus a través del número de acceso GSE50972 de la serie GEO). Este estudio mostró un mecanismo muy interesante en el que no importaba si los miRNAs estaban en un estado regulatorio positivo o negativo, había motivos en secuencias específicas que se encontraban en mayor o menor medida, dentro o fuera del endosoma o citoplasma. Esto nos lleva a la conclusión de que hay mecanismos por los cuales el estado celular no es relevante, existen tendencias de que ciertos miRNAs se empaqueten dentro de regiones específicas, y precisamente esta es la clave para decir que la proteína hnRNPA2B1 es responsable de llevar las secuencias al endosoma y, en consecuencia, a los exosomas para la posterior liberación al medio extracelular.

Esta tesis tiene como objetivo implementar y comparar diversas técnicas para la identificación de estos patrones y contribuir a la expansión del conocimiento en esta área, que promete contribuir al creciente impacto que esta ha tenido en los últimos tiempos en el campo de la biomedicina. Los mecanismos que promueven las interacciones miARN-proteína se pueden tratar con métodos y algoritmos desarrollados en el campo de la biología computacional. Específicamente, esta investigación contribuye a la expansión del conocimiento sobre la selección de descriptores moleculares relevantes para reconocer proteínas que se unen a miARN.

2.2. Aprendizaje de máquina

De acuerdo a Francois (2016), una definición general de Aprendizaje de Máquina (AM) es: el campo de estudio que da a las computadoras la habilidad de aprender sin ser programadas explícitamente. Por otro lado y de acuerdo con Sancho y García (2016), el aprendizaje de máquina es la rama de la Inteligencia Artificial (IA) que apuesta por desarrollar técnicas que permiten a las computadoras aprender. Más concretamente, se trata de crear algoritmos capaces de generalizar comportamientos y reconocer patrones a partir de información proporcionada en forma de ejemplos. AM

es, por lo tanto, un proceso de inducción del conocimiento, este permite generalizar conocimiento a partir de observaciones, y predecir comportamiento futuro de casos nuevos. Cuando se ha observado todo el conjunto de casos particulares, la inducción se considera completa y válida. Por lo general, en la mayoría de los casos es imposible obtener una inducción completa, por lo que el modelo está sujeto a un cierto grado de incertidumbre y, por consiguiente, no puede considerarse como un esquema de inferencia formalmente válido ni justificarse empíricamente.

A un nivel muy básico podría decirse que una de las tareas del AM es inducir conocimiento al generalizar propiedades de objetos previamente observados. Hay una gran cantidad de problemas que se engloban dentro del llamado aprendizaje inductivo. La principal diferencia entre ellos es el tipo de objetos que intentan predecir, a continuación se listan los más usados de acuerdo a Sancho y García (2016):

Regresión: predecir un valor real. Por ejemplo, predecir el precio de una casa en un cierto vecindario en un determinado mes. O predecir la estatura de una persona dependiendo de la nacionalidad, género, región, etc.

Clasificación: este tipo de aprendizaje predice la clase de objetos de entre un conjunto de clases previamente fijadas. Por ejemplo, la imagen de la silueta de un animal cuadrúpedo, se puede asociar a las clases *gato*, *perro*, *vaca*, *caballo*, *borrego*, *león*, entre otros. También, “la clase” podría ser la capacidad que tienen ciertos antibióticos para aniquilar bacterias gracias a sus múltiples propiedades fisicoquímicas. En algunos casos, *positivo* y *negativo* son las únicas clases, sin embargo, un problema de clasificación puede extenderse a un modelo multiclase, al final, las clases siempre dependen del tipo de objeto a clasificar.

Categorización: predecir el orden óptimo de un conjunto de objetos de acuerdo con un orden de relevancia dado. Por ejemplo, el índice de búsqueda de Google que es devuelto cuando se busca algo en la web como una respuesta al usuario.

Al resolver un problema de AM es importante encontrar una manera de medir la asertividad de la predicción. Dada la metodología aplicada, el problema de medir el éxito del aprendizaje se debe tratar para cada caso particular.

Por otro lado, de acuerdo con Sancho y García (2016), y dependiendo del tipo de salida que se está produciendo y cómo se maneja el tratamiento de los ejemplos, los diferentes algoritmos de AM pueden agruparse en:

Aprendizaje supervisado : es un tipo de aprendizaje en el que a partir de un con-

junto de ejemplos previamente etiquetados (conjunto de entrenamiento), se crea un modelo y se prueba con otro conjunto de ejemplos etiquetados (conjunto de pruebas). Para mayor detalle revisar Sancho y García (2016).

Aprendizaje no supervisado: en cambio, los modelos de clasificación no supervisados son aquellos en los que no existe un conjunto de ejemplos previamente etiquetados, pero sólo a partir de las propiedades de los ejemplos se intenta obtener un agrupamiento de los ejemplos según su similitud. Para mayor detalle revisar Sancho y García (2016).

Aprendizaje semi-supervisado: es una combinación de los dos algoritmos anteriores, teniendo en cuenta ejemplos etiquetados y no etiquetados. Para mayor detalle revisar Sancho y García (2016).

Aprendizaje por refuerzo : este tipo consiste en aprender a decidir ante una situación dada, qué acción es la más apropiada para lograr una meta. Consiste en dos componentes: un componente selectivo que implica realizar la mejor acción para ejecutar entre varias opciones y el componente asociativo, en el sentido de que las alternativas encontradas están asociadas con situaciones particulares en las que se tomaron. Para mayor detalle revisar Sancho y García (2016).

Encontramos los enfoques de AM prometedores para este tipo de problema de interacción miARN-proteína. El trabajo con el enfoque más directo y dedicado para resolver este problema en particular fue *Sequence-Based Prediction of microRNA-Binding Residues in Proteins Using Cost-Sensitive Laplacian Support Vector Machines* (Sheng y Zhou, 2013), donde los autores abordan el problema de la siguiente manera: “dada una secuencia de proteína x, encuentre si esta tiene o no la capacidad de acoplarse a una secuencia de miARN”, por supuesto, esto es una generalización y es bien conocido que una proteína posee múltiples mecanismos de acoplamiento. Sin embargo, en este trabajo tomamos la base computacional del enfoque de Sheng y Zhou (2013), para evaluar múltiples estrategias con el objeto de mejorar las posibilidades en términos de rendimiento, y dado el caso, sentar una mejor base para trabajos futuros en los cuales el fundamento biológico se pueda extender y, por consiguiente, acercar más el modelo comptuacional al problema biológico.

2.2.1. Selección de características

De acuerdo a Guyer (2016), la selección de características (FS) es, por varias razones, un paso fundamental en el aprendizaje de máquina para construir un buen modelo. Una de ellas es que implica algún grado de reducción de cardinalidad, ya que esta disminuye el número de características que se consideran para construir un modelo de clasificación. Los datos casi siempre contienen más información de la que se necesita para construir el modelo, mucha de la información es incorrecta o simplemente no aporta significativamente a la clasificación. Por ejemplo, es posible tener un conjunto de datos con 1,000 entradas que describen las características de 1,000 pianos, y por cada entrada 500 caracterśticas que la describen, sin embargo, si algunas de las caracterśticas no varian a lo largo de todas las entradas, se obtendría muy poco beneficio al agregarlas al modelo. No sólo la selección de características mejora la calidad del modelo, sino que también hace que el proceso de modelado sea más eficiente. Si se utilizan características innecesarias durante la construcción de un modelo, se requiere más poder de procesamiento y más memoria, así mismo, más espacio de almacenamiento para el modelo. Además, si los datos son ruidosos o redundantes esto haría más difícil descubrir patrones significativos. Para esta selección existen diversos algoritmos, tales como el método de envoltura (Kohavi y John, 1997), otros basados en algoritmos evolutivos (Paul y Das, 2015) y técnicas de filtrado como la descrita por Hall *et al.* (2009) y Mukaka (2012).

2.2.2. Modelos de validación no cruzada

2.2.2.1. Validación “holdout”

Este método es el más sencillo a la hora de validar el resultado de prueba estadística. El conjunto de datos se divide en dos: el conjunto de prueba y el de entrenamiento (Kuhn y Johnson, 2013).

2.2.3. Modelos de valicación cruzada

2.2.3.1. Validación cruzada de *k* pliegues

La validación cruzada mantiene una fracción de los datos como un conjunto de entrenamiento y otro para pruebas (Kuhn y Johnson, 2013), al mismo tiempo se busca

que todos los ejemplos funcionen para entrenamiento y para prueba, este proceso es descrito en la **Figura 3**. En una validación cruzada de k pliegues, k representa la cantidad de subconjuntos del conjunto de datos, $k-1$ es el tamaño del conjunto de entrenamiento y 1 el de pruebas, además, k es también el número de pliegues necesario para asegurar que todos los ejemplos funcionen tanto de entrenamiento como de prueba.

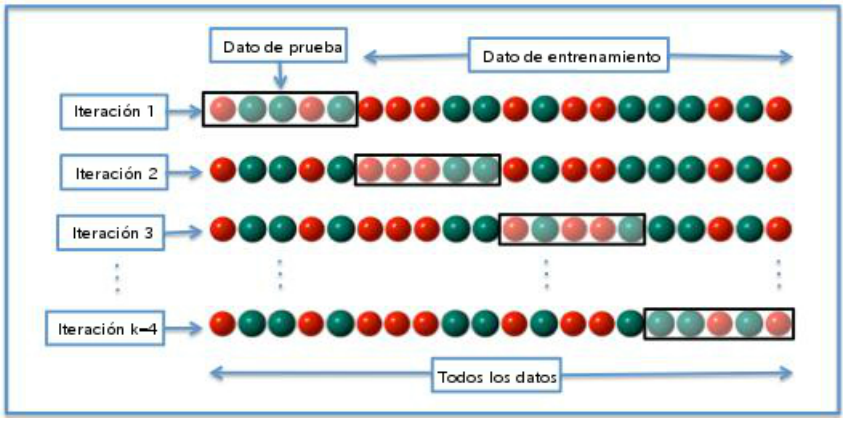

Figura 3. Validación cruzada de k pliegues[1]

[1] https://upload.wikimedia.org/wikipedia/commons/1/1c/K-fold_cross_validation_EN.jpg

Capítulo 3. Metodología de solución

En este capítulo se describe la metodología propuesta para resolver el problema planteado en este trabajo de investigación (**Figura 4**).

Primero, se extraen subcadenas de las cadenas peptídicas validadas como ejemplos de entrenamiento con actividad de acoplamiento a miARN según la metodoloía descrita en Sheng y Zhou (2013) (**Sección 3.1.5**).

Segundo, se extraen diversos descriptores moleculares de los conjuntos de datos descritos en la Sección 3.1.3.

Tercero, una vez extraidos los descriptores moleculares de dichos conjuntos de datos, y siguiendo la metodología descrita, se aplican diversas técnicas de selección de características, tales como los algoritmos evolutivos basados en descomposición multiobjetivo y una variante de la misma, además se aplican algoritmos de filtrado (**Sección 3.1.8**).

Cuarto, se alimentan las dos variantes de algoritmos de aprendizaje de máquina, entre ellos los descritos en Sheng y Zhou (2013) y el desarrollado en este trabajo, el cual fue desarrollado haciendo uso de la libreria SciKit Learn (Pedregosa *et al.*, 2011) (**Sección 1.5**).

Quinto, el resultado de los dos clasificadores anteriores es utilizado para aplicar métodos de validación que se describen a continuación en este trabajo.

Por último, con la combinación de conjuntos de datos, técnicas de selección de características y de los dos algoritmos de aprendizaje de máquina se obtienen variantes de propuesta de solución que se comparan con el trabajo original y base de Sheng y Zhou (2013) (**Capítulo 4**).

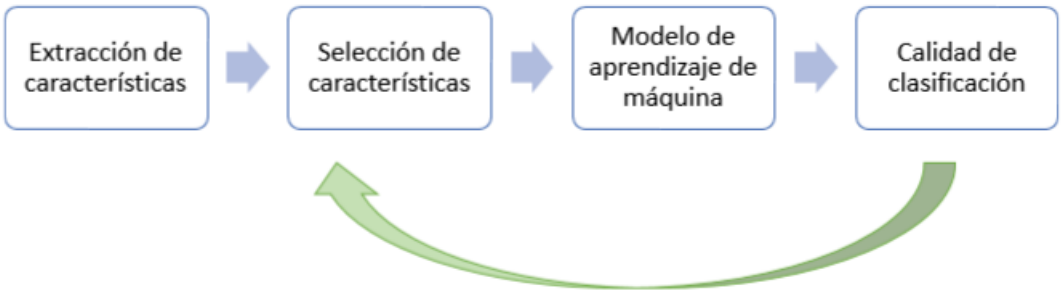

Figura 4. Diagrama de flujo de solución del modelo propuesto.

3.1. Metodología computacional

3.1.1. Descriptores moleculares

El propósito principal de los descriptores moleculares es ayudar a reconocer patrones específicos en secuencias que podrían ayudar a discriminar ejemplos entre clases dado un fenómeno específico. En el trabajo de Sheng y Zhou (2013) un ejemplo de entrenamiento es un conjunto de 11-meros provenientes de una secuencia de proteína. Sobre estos 11-meros se calculan varios conjuntos de descriptores. El primer conjunto describe la propensión a la interacción mutua con las purinas (adenina y guanina) y pirimidinas (citosina y timina) a lo largo de la secuencia. Estos valores se obtienen por tripletes de residuos de aminoácidos, la clase de los ejemplos de entrenamiento (positiva o negativa) se calcula utilizando los residuos de miARN cercanos espacialmente a dichos tripletes. Los nucleótidos se agrupan en purinas y pirimidinas, *i* y *j*, respectivamente. Se considera que un triplete interactúa cuando su residuo central se encuentra acoplado a miARN. Los 20 aminoácidos esenciales se agrupan en seis clases en función de la polaridad de los dipolos de los enlaces involucrados (el momento dipolar de enlace mide la polaridad de un enlace químico dentro de una molécula) y los volúmenes de las cadenas laterales de la siguiente manera (**Tabla 1**): Clase *a*: Ala, Gly, Val; Clase *b*: Ile, Leu, Phe, Pro; Clase *c*: Tyr, Met, Thr, Ser, Cys; Clase *d*: His, Asn, Gln, Tpr; Clase *e*: Arg, Lys y Clase *f*: Asp, Glu. El razonamiento detrás de esta clasificación de aminoácidos está dado en un trabajo de 2009 (Wu *et al.*, 2009) en el que los autores predicen residuos de proteína que se acoplan a ADN, esto a partir de secuencias de aminoácidos utilizando un modelo de bosques aleatorios con una característica híbrida, al mismo tiempo que tomaron los fundamentos de otro trabajo (SA *et al.*, 2007a), el cual describe una clasificación de aminoácidos en la que las interacciones electrostáticas e hidrofóbicas se consideran como las responsables de dirigir las interacciones proteína-proteína. Se cree que estos últimos dos tipos de interacciones se reflejan en los dipolos y volúmenes de las cadenas laterales de aminoácidos, respectivamente. Por lo tanto, estos dos parámetros se calcularon, respectivamente, utilizando el método de teoría de densidad funcional B3LYP/6-31G y un enfoque de modelado molecular (Wu *et al.*, 2009).

Un dipolo eléctrico es un sistema de dos cargas de signo opuesto e igual magnitud cercanas entre sí. De acuerdo con Sheng y Zhou (2013), la propensión de interacción

mutua de un triplete x con un nucleótido y, se entiende de la siguienta manera:

Escala de dipolos (en debyes (D)): -, Dipolo < 1.0D; +, 1.0D < Dipolo < 2.0D; ++, 2.0D < Dipolo < 3.0D; +++, Dipolo > 3.0D; +'+'+', Dipolo > 3.0D con orientación opuesta.

Escala de volumen ($Å^3$): -, Volumen<50$Å^3$; +, Volumen> 50$Å^3$.

Tabla 1. Clasificación de aminoácidos.

Clase.	Escala de dipolo (a)	Escala de volumen (b)	Clase
a	-	-	Ala, Gly, Val
b	-	+	Ile, Leu, Phe, Pro
c	+	+	Tyr, Met, Thr, Ser, Cys
d	++	+	His, Asn, Gln, Tpr
e	+++	+	Arg, Lys
f	+'+'+'	+	Asp, Glu

$$P(x,y) = \sum_{i,j} f_{ij}(x,y) log_2 \frac{f_{i,j}(x,y)}{f_i(x) f_j(y)}. \quad (1)$$

Donde,

$$f_{i,j}(x,y) = NRN(x,y) / \sum_{x,y} NRN(x,y), \quad (2)$$

$$f_i(x) = NR(x) / \sum_{x} NR(x) \quad (3)$$

y

$$f_j(y) = NN(y) / \sum_{x} NN(y); \quad (4)$$

donde x representa un triplete (z,z,z) ($z \in \{a,b,c,d,e,f\}$, $y \in \{i,j\}$, un nucleótido, $NRN(x,y)$ es el número de tripletes de residuos x acoplándose con el nucleótido y, $NR(x)$ es el número de tripletes de residuos x, y $NN(y)$ es el número de nucleótidos y. La ecuación 2 representa la propensión de interacción entre un triplete x y un nucleótido y, la ecuación 3 representa la relevancia de un triplete x, y la ecuación 4 representa la relevancia de un nucleótido y. Para cada secuencia de longitud 11, se extraen 9 tripletes, por cada triplete se calcula su valor de propensión de interacción mutua con ambas clases de nucleótidos (i,j).

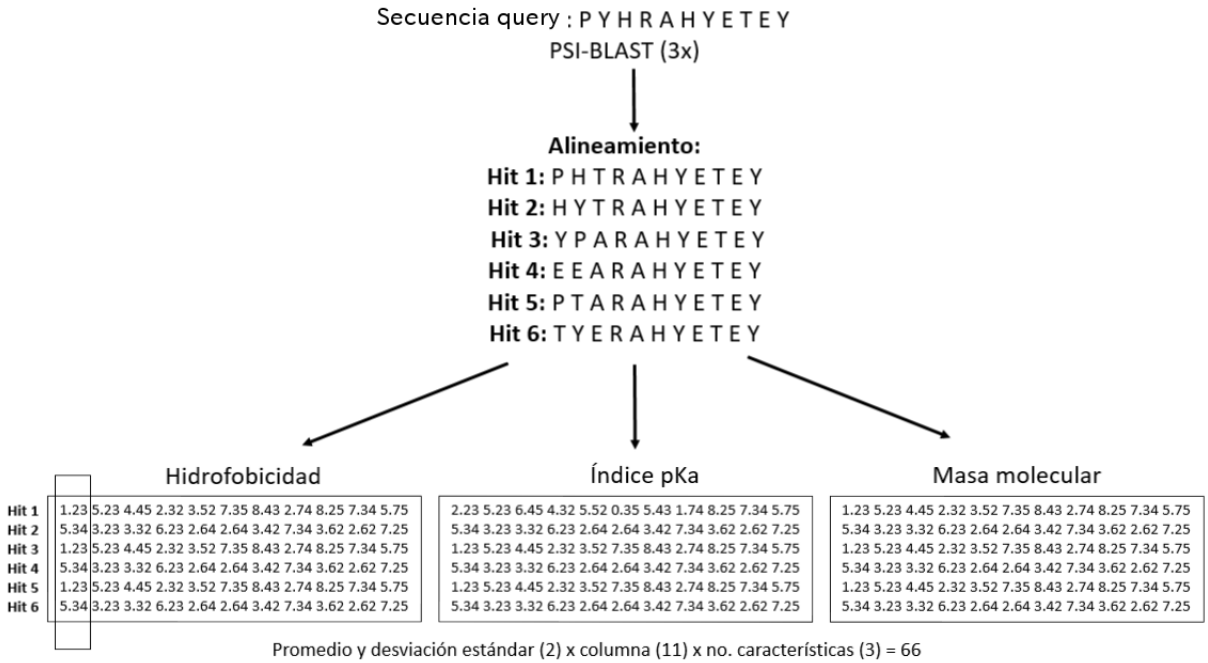

Figura 5. Ejemplo de cálculo de los 3 subcojuntos de características citepr18.

El segundo conjunto de características consiste en información evolutiva en forma de matrices de puntuación específica de posición (PSSMs), estas son generadas con el paquete PSI-BLAST (Altschul, 1997) contra el conjunto de datos no redundante (nr) de secuencias de aminoácidos en el NCBI. Este procedimiento se ejecuta tres iteraciones y se define un umbral de valor-E establecido en $1x10^{-3}$. Después de obtener los elementos de cada matriz, cada uno de ellos, r, se asigna al intervalo [0,1] utilizando la función logística estándar:

$$f(r) = \frac{1}{1 + exp(-r)}. \tag{5}$$

Se ha demostrado previamente que la información evolutiva contenida en PSSMs puede ser beneficiosa para la predicción de residuos de unión a ARN (Ma *et al.*, 2011; Carson *et al.*, 2010), sin embargo, esta puede generar pérdida de cierta información importante, por lo tanto, el tercer conjunto de características consiste en propiedades físico-químicas llamadas características *HKM*. Del mismo modo, como se hizo con el segundo conjunto de características, una PSSM se calcula haciendo uso del paquete PSI-BLAST con tres iteraciones y un umbral de valor-E establecido en $1x10^{-5}$. Posteriormente, cada una de las características se calcula dada su media y valores de desviación estándar por columna (**Figura 5**). *H* representa la hidrofobicidad, por lo que

se necesita una escala para definir los límites en los que estas características oscilan. Para dicho cálculo se utilizó la escala de hidrofobicidad de Kyte y Doolittle (Kyte y Doolittle, 1982), una escala usada normalmente para caracterizar e identificar la posible estructura o dominios de una proteína (consulte el Apéndice A.2 para los valores de la escala). *K* significa índice $P_k a$ (Altschul, 1997), el cual representa el estado de ionización de la molécula. Dado que los grupos fosfato en los nucleótidos están cargados negativamente, este estado de ionización tiene una influencia importante en las interacciones. *M* representa la masa molecular, la cual está estrechamente relacionada con el volumen ocupado por el 11-mero (Sheng y Zhou, 2013). En resumen, el primer conjunto comprende las primeras 18 características (9X2), el segundo 220 (11X20) características y el tercero 11X6 = 66 características, totalizando 304 características. En el trabajo de Sheng y Zhou (2013) se describe la metodología utilizada para la extracción del conjunto de características. El primer subconjunto de características (A) tiene que ver con la propensión de la interacción mutua de un triplete *x* con un nucleótido *y*, siguiendo los cálculos como se describe en Sheng y Zhou (2013). El segundo subconjunto de características (B) involucra la información evolutiva en matrices de puntaje de posición (PSSM) generada por el paquete PSI-BLAST (Altschul, 1997). Este paquete fue utilizado contra el conjunto de datos no redundantes (nr) de secuencias de aminoácidos del NCBI, y los parámetros se mantuvieron como se describen en el trabajo de Sheng y Zhou (2013), con tres iteraciones y utilizando un umbral de valor-E fijado a $1x10^{-3}$. Con el fin de obtener los mejores parámetros para el cálculo PSSMs, uno de los 11-meros tomado de los ejemplos de entrenamiento se utilizó en la versión en línea del NCBI PSI-BLAST, consiguiendo un wordsize = 2 y threshold = 20,000 con la ayuda de la herramienta de autoajustado de parámetros. El tercer subconjunto de características (C) tiene que ver con las propieadades físicoquímicas *HKM*. De forma similar al subconjunto B, se calculan las PSSMs con el uso del paquete PSI-BLAST, con tres pliegues y un umbral de valor-E fijado en $1x10^{-5}$. Para cada cálculo, cada una de las características (*HKM*) se obtuvo dada su media y valores de desviación estándar por columna (**Figura 5**).

Además, se agregaron 31 nuevas características propuestas en el trabajo de Jahandideh y Srinivasasainagendra (2012), las cuales son derivadas de secuencia: (i) composición de los 20 aminoácidos (20 características), (ii) composición de aminoácidos en 9 diferentes grupos físicoquimicos inclyendo muy pequeños (A,G,S), pequeños (N, D, C,

P y T), alifáticos (A, G, I, L, P y V), aromáticos (F, W e Y), polares (R, N, D, Q, E, H, K, S, T e Y), no-polares (A, C, G, I, L, M, F, P, W y V), cargados (D, E, R, H y K), acídicos (D y E), y básicos (R, H y K) (9 características), (iii) pI, el punto isoeléctrico (1 característica) y (iv) peso molecular (1 característica).

Finalmente, al conjunto de datos original (el que proviene de las estructuras de proteína del trabajo de Sheng y Zhou (2013)) se le llamó **D1**. Al conjunto de datos adicional (el que proviene de las estructuras de proteína encontradas en este trabajo) se le llamó **D2**.

Al conjunto de datos, cuyos ejemplos solo positivos provienen solo de **D1** se le llamó **D3**.

Al conjunto de datos, cuyos ejemplos solo positivos provienen solo de **D2** se le llamó **D4**.

Al conjunto de características original (el que proviene del trabajo de Sheng y Zhou (2013)) se le llamó **C1**.

Al conjunto de características adicional (el que proviene del trabajo de Jahandideh y Srinivasasainagendra (2012)) se le llamó **C2**.

Al conjunto de características resultantes de aplicar la metodología de Paul y Das (2015) se le llamó **C3**.

Al conjunto de características resultantes de aplicar selección la metodología de Beltrán *et al.* (2017) a **C2** se le llamó **C4**.

Al conjunto de características resultantes de aplicar selección mediante la técnica de envoltura de Kohavi y John (1997) en la implementación de Beltrán (2014) se le llamó **C5**.

Al conjunto de características resultantes de aplicar la técnica de filtrado se le llamó **C6**.

Al conjunto de características resultantes de aplicar la técnica de filtrado y la metodología de Beltrán *et al.* (2017) se le llamó **C7**.

Al conjunto de características resultantes de ProtDCal (Sección 3.1.5) se le llamó **C8**.

Al conjunto de características resultantes de aplicar la metodología de Paul y Das (2015) a **C1** se le llamó **C9**.

Al conjunto de características resultantes de ProtDCal y aplicar la técnica de filtrado se le llamó **C10**.

Al conjunto de características resultantes de ProtDCal y aplicar la técnica de filtrado y

la metodología de Beltrán *et al.* (2017) se le llamó **C11**.

3.1.2. Clasificador de Sheng y Zhou (2013)

Sheng y Zhou (2013) fueron los primeros en utilizar predictores de interacción miARN-proteína basados en aprendizaje de máquina. En un esfuerzo por sentar una base en esta novedosa área, ellos desarrollaron un clasificador para predecir secuencias de proteína que tengan la habilidad de acoplarse a miARN. En su trabajo, un residuo con dicha capacidad fue definido como aquel que cualquiera de sus átomos pesados se encuentra localizado a una distancia menor o igual a 3.5Å de cualquier átomo de los nucleótidos de la secuencia de miARN; con el fin de calcular las distancias entre los átomos de proteína y miARN, fueron utilizados complejos estructurales del tipo miARN-proteína. Una búsqueda en el banco de datos de proteínas (PDB RSCB) (para los complejos miARN-proteína, de los cuales los ejemplos de entrenamiento etiquetados proceden) y UniProt (para las secuencias, de las cuales los ejemplos no etiquetados proceden) [1] reveló sólo cinco estructuras en complejo válidas, de las cuales se tomaron ejemplos positivos y negativos, después de una eliminación de redundancia, en la que todas las secuencias con al menos 25 % de identidad fueron agrupadas utilizando el programa blastclust (**tablas 6-7**) del paquete BLAST (Altschul, 1997) de NCBI. Sólo permaneció la secuencia más larga de cada agrupación y sólo 4 cadenas procedentes de 3 complejos (**Tabla 7**) se mantuvieron y se utilizaron para propósitos de entrenamiento. En la **Tabla 8** se puede observar la similitud entre los centrómeros de dichos agrupamientos, algo importante a destacar en esta tabla es la baja similitud de secuencia entre ellos, algo que se buscaba con la eliminación de redundancia.

Con el fin de superar esta escasez en los ejemplos de entrenamiento, después de la eliminación de redundancia, se utilizaron 16 secuencias de proteínas no etiquetadas y previamente anotadas con capacidad de unión a miARN (todas proviniendo de UniProt) (**Tabla 6**), en otras palabras, ejemplos no etiquetados. Los ejemplos no etiquetados, los cuales también fueron procesados con la anterior eliminación de redundancia, se integran posteriormente para formar el clasificador de aprendizaje automático con el fin de aumentar el rendimiento de clasificación.

Un ejemplo de entrenamiento es definido como una secuencia de aminoácidos de tamaño $l = 11$. Considerando una proteína de n residuos, el número total de secuencias

[1] Revisado el 15 de marzo de 2012.

extraidas es calculada por $n - l + 1 - r$, donde l es el tamaño de la ventana deslizante y r es el número de residuos sin información de sus coordenadas atómicas en las entradas del PDB. Siguiendo esta metodología, los conjuntos de datos resultantes contienen 61 11-meros como ejemplos positivos, 1,298 11-meros como ejemplos negativos y 7,983 11-meros como ejemplos no etiquetados (para ver dichas secuencias, consulte el **Apéndice A (A.1))**.

Tabla 2. El conjunto de datos original y no redundante descargado del PDB y UniProt citepr18.

Original				
PDB(ID)	3A6P	3ADI	3TRZ	3TS0
	3TS2			
Uniprot(ID)	Q9XGW1	O04379	O04492	P92186
	Q8K3Y3	Q2KIA0	Q06413	F1LZC6
	Q8CFN5	A4UTP7	Q5R444	Q8TCS8
	Q8K1R3	Q5RCW2	Q01860	F7D1A4
	Q3MHX3	Q9BWF3	Q4R979	Q8C7Q4
	Q9BDY9	P48431	Q9GNA3	F1PAY8
	Q9U4F5	Q9GNA6	Q9GNJ2	Q9GND0
	Q8MRC7	Q9TW27	Q9TW12	E9Q6I7
	Q9NHW9	Q9W5S7	Q9NIH3	Q86LT0
	Q9U6N4			
No redundante				
PDB(ID)	3TS0_B	3ADI_A	3A6P_A	3A6P_C
Uniprot(ID)	Q9GNA6	Q8CFN5	Q4R979	Q5RCW2
	F7D1A4	O04379	Q4R979	Q5RCW2
	Q01860	P48431	Q9XGW1	Q86LT0
	F1LZC6	P92186	E9Q6I7	Q8K3Y3

Tabla 3. Ejemplos 11-meros positivos y negativos por cadena.

Cadena	Positivos	Negativos
3TS0:B	29	87
3ADI:A	6	55
3A6P:A	24	998
3A6P:C	2	158

En este capítulo y el siguiente, las cantidades de ejemplos positivos, negativos, su clasificación dada una solución, y sus medidas de calidad estarán descritos como,

PQ = La cantidad de ejemplos positivos.

NQ = La cantidad de ejemplos negativos.

TP = La cantidad de verdaderos positivos, aquellos ejemplos que fueron clasificados como positivos y pertenecen a la clase positiva.

Tabla 4. Similitud entre centrómeros.

	3TS0_B	3ADI_A	3A6P_A	3A6P_C	Q9GNA6	Q8CFN5	Q4R979	Q5RCW2	F7D1A4	O04379	Q9XGW1	O04492	Q01860	P48431	Q9TW12	Q86LT0	F1LZC6	P92186	E9Q6I7	Q8K3Y3
3TS0_B	100%	0%	0%	0%	0%	0%	0%	0%	0%	38%	0%	0%	0%	0%	0%	0%	0%	45%	0%	100%
3ADI_A	-	100%	0%	26%	0%	0%	50%	0%	50%	0%	0%	97%	0%	0%	0%	0%	0%	0%	0%	0%
3A6P_A	-	-	100%	0%	0%	0%	0%	0%	32%	29%	37%	0%	0%	0%	0%	0%	0%	0%	0%	0%
3A6P_C	-	-	-	100%	0%	50%	0%	0%	33%	36%	0%	0%	0%	22%	0%	0%	0%	0%	24%	0%
Q9GNA6	-	-	-	-	100%	0%	0%	0%	0%	0%	0%	0%	0%	0%	98%	98%	0%	0%	0%	0%
Q8CFN5	-	-	-	-	-	100%	0%	0%	0%	39%	29%	0%	0%	0%	0%	0%	99%	0%	0%	0%
Q4R979	-	-	-	-	-	-	100%	27%	0%	32%	30%	0%	0%	0%	0%	0%	0%	35%	0%	23%
Q5RCW2	-	-	-	-	-	-	-	100%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
F7D1A4	-	-	-	-	-	-	-	-	100%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
O04379	-	-	-	-	-	-	-	-	-	100%	75%	0%	0%	0%	0%	0%	39%	0%	0%	0%
Q9XGW1	-	-	-	-	-	-	-	-	-	-	100%	0%	0%	0%	0%	0%	0%	0%	0%	33%
O04492	-	-	-	-	-	-	-	-	-	-	-	100%	25%	0%	0%	0%	0%	0%	0%	0%
Q01860	-	-	-	-	-	-	-	-	-	-	-	-	100%	0%	0%	0%	0%	0%	0%	0%
P48431	-	-	-	-	-	-	-	-	-	-	-	-	-	100%	0%	0%	33%	0%	0%	0%
Q9TW12	-	-	-	-	-	-	-	-	-	-	-	-	-	-	100%	100%	0%	0%	44%	0%
Q86LT0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	100%	0%	0%	0%	0%
F1LZC6	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	100%	0%	0%	0%
P92186	-	-	-	-		-	-	-	-	-	-	-	-	-	-	-	-	100%	0%	36%
E9Q6I7	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	100%	0%
Q8K3Y3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	100%

FP = La cantidad de falsos positivos, aquellos ejemplos que fueron clasificados como positivos y pertenecen a la clase negativa.
TN = La cantidad de verdaderos negativos, aquellos ejemplos que fueron clasificados como negativos y pertenecen a la clase negativa.
FN = La cantidad de falsos negativos, aquellos ejemplos que fueron clasificados como negativos y pertenecen a la clase positiva.
Y las medidas de calidad,
Pre ($precisión$) = $TP/(TP + FP)$, la probabilidad de, dado un ejemplo clasificado como positivo, cuando este pertenezca a la clase positiva.
Sen ($sensibilidad$) = $TP/(TP + FN)$, la probabilidad de, dado un ejemplo positivo, este sea clasificado como tal.
Esp ($especificidad$) = $TN/(TN + FP)$, la probabilidad de, dado un ejemplo negativo, este sea clasificado como tal.
$Val{-}F$ ($valor{-}F$) = $2 \times precisión \times sensibilidad/(precisión + sensibilidad)$, la media armónica entre la precisión y sensibilidad.
$MCC = TP \times TN - FP \times FP/\sqrt{(TP + FP) \times (TP + FN) \times (TN + FP) \times (TN + FN)}$, coeficiente de correlación de Matthews, es una medida de clasificación binaria que mide la correlación entre lo clasificado y lo esperado.
AUC = Área bajo la curva, calculada con la librería Scikit Learn, la probabilidad de dado un ejemplo positivo, este reciba del clasificador un puntaje mayor que uno escogido de la clase negativa (Fawcett, 2006).

3.1.3. Análisis de calidad de clasificación de cadenas

Siguiendo la metodología de Sheng y Zhou (2013), cuatro cadenas de proteína permanecieron después de la eliminación de redundancia, lo que resultó en sólo cuatro cadenas de proteínas para entrenar: 3TS0, cadena B, 3ADI, cadena A, 3A6P, cadena A y 3A6P, cadena C. Con el propósito de dilucidar si alguna de las cadenas tiene una mayor relevancia en la clasificación, se corrió el software proporcionado por los autores con una validación cruzada de tres pliegues y utilizando el conjunto **DC1**, tal como especifican los autores. Los resultados se muestran por cadena (**tablas 2-5**), la cadena A de 3ADI y la cadena A de 3A6P obtuvieron mejores resultados en todas las medidas de calidad. La clasificación general y por cadena se comportó globalmente mejor que lo reportado por Sheng y Zhou (2013)

Tabla 5. Resultados de prueba de tres pliegues **S()** utilizando la secuencia 3TS0:B del PDB.

Iter	PQ	NQ	TP	FP	TN	FN	Val-F	Sen	Pre	MCC	AUC	Esp
1	9	31	9	28	3	0	0.39	1.00	0.24	0.15	0.55	0.10
2	11	30	11	15	15	0	0.59	1.00	0.42	0.46	0.75	0.50
3	14	31	14	20	11	0	0.58	1.00	0.41	0.38	0.68	0.35
Prom	11.33	30.67	11.33	21.00	9.67	0.00	0.52	1.00	0.36	0.33	0.66	0.32

Tabla 6. Resultados de prueba de tres pliegues **S()** utilizando la secuencia 3ADI:A del PDB.

Iter	PQ	NQ	TP	FP	TN	FN	Val-F	Sen	Pre	MCC	AUC	Esp
1	3	18	3	0	18	0	1.00	1.00	1.00	1.00	1.00	1.00
2	2	16	2	0	16	0	1.00	1.00	1.00	1.00	1.00	1.00
3	1	21	1	0	21	0	1.00	1.00	1.00	1.00	1.00	1.00
Prom	2.00	18.33	2.00	0.00	18.33	0.00	1.00	1.00	1.00	1.00	1.00	1.00

Tabla 7. Resultados de prueba de tres pliegues **S()** utilizando la secuencia 3A6P:A del PDB.

Iter	PQ	NQ	TP	FP	TN	FN	Val-F	Sen	Pre	MCC	AUC	Esp
1	5	333	5	0	333	0	1.00	1.00	1.00	1.00	1.00	1.00
2	10	330	10	0	330	0	1.00	1.00	1.00	1.00	1.00	1.00
3	6	335	6	0	335	0	1.00	1.00	1.00	1.00	1.00	1.00
Prom	7.00	332.67	7.00	0.00	332.67	0.00	1.00	1.00	1.00	1.00	1.00	1.00

Tabla 8. Resultados de prueba de tres pliegues **S()** utilizando la secuencia 3A6P:C del PDB.

Iter	PQ	NQ	TP	FP	TN	FN	Val-F	Sen	Pre	MCC	AUC	Esp
1	12	103	12	28	75	0	0.46	1.00	0.30	0.47	0.86	0.73
2	13	100	13	15	85	0	0.63	1.00	0.46	0.63	0.92	0.85
3	15	97	15	20	77	0	0.60	1.00	0.43	0.58	0.90	0.79
Prom	13.33	100.00	13.33	21.00	79.00	0.00	0.57	1.00	0.40	0.56	0.90	0.79

3.1.4. Kernel SVM Laplaciano

Sheng y Zhou (2013) describen una extensión sensible a costos al kernel SVM Laplaciano (LapSVM), llamado kernel Laplaciano sensible a costos (CS-LapSVM) (Sancho y García, 2016). LapSVM es un algoritmo semi-supervisado, basado en dos principios importantes. Uno es la suposición de colector, es decir, ejemplos similares dan resultados similares, lo cual lleva a una propagación de etiqueta, en la cual, un ejemplo etiquetado hereda su clase a los ejemplos no etiquetados similares a este (**Figura 6**). El otro es el principio del margen máximo, en el cual las distribuciones de dos clases distintas tienen un margen más grande entre hiperplanos. La extensión CS-LapSVM de LapSVM se utiliza en este trabajo para construir modelos de predicción de cadenas de aminoácidos con capacidad de acoplamiento a miARN. Esta extensión

posee la capacidad de utilizar ejemplos no etiquetados bajo la suposición de colector, la piedra angular de muchos algoritmos semi-supervisados para clasificación, este último indica que los datos de alta dimensionalidad se encuentran en un espacio dimensionalmente más pequeño. Esta suposición resuelve un problema bien conocido de muchos métodos estadísticos y algoritmos de aprendizaje, la llamada maldición de la dimensionalidad, la cual relaciona el hecho de que cuando un volumen crece exponencialmente con el número de dimensiones, un número cada vez mayor de ejemplos es necesario para tareas estadísticas tales como la estimación fiable de densidades. Si los datos se encuentran en un colector de baja dimensión, entonces el algoritmo de aprendizaje puede funcionar básicamente en un espacio de tal dimensión, evitando así la “maldición de la dimensionalidad”.

$$argmin\left\{\frac{1}{l}\sum_{i=1}^{l} V(\chi_i, y_i) + \gamma_A \|f\|_H^2 + \frac{\gamma_I}{(u+l)^2}\sum_{i,j=1}^{l+u}(f(\chi_i) - f(\chi_j))^2 \mathbf{W}_{ij}\right\}. \qquad (6)$$

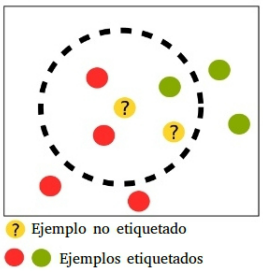

Figura 6. Propagación de etiqueta (David Przybilla (autor independiente), marzo de 2017).

Aquí, el primer término representa la función de pérdida, el segundo término representa la función de decisión de penalización de la complejidad del clasificador; el tercer término obliga a que ejemplos similares tengan salida similar de acuerdo con una matriz de ponderación de similitud **W** de todos los ejemplos de entrenamiento. γ_A y γ_I son dos parámetros que ponderan estos tres términos. Uno de los problemas más difíciles de la clasificación supervisada son los conjuntos de datos desequilibrados, esto significa una disparidad entre la cantidad de ejemplos para cada clase de datos. LapSVM no toma en cuenta un costo (peso) dada una clase específica. Sheng y Zhou (2013) propusieron extender LapSVM para escenarios sensibles a costo por clase, de

la siguiente manera:

$$\underset{f \in H}{argmin} \left\{ \frac{1}{l} \sum_{i=1}^{l} c(y_i)) V(\chi_i, y_i) + \gamma_A \|f\|_H^2 + \frac{\gamma_I}{(u+l)^2} \sum_{i,j=1}^{l+u} (f(\chi_i) - f(\chi_j))^2 \mathbf{W}_{ij} \right\}. \quad (7)$$

Aquí, $c(y_i)$ es calculado con la inversa de la cantidad de ejemplos de la clase i, es la proporción de ejemplos que corresponde al costo de clasificación errónea de la etiqueta y_i, esta fue la modificación propuesta en CS-LapSVM. Podría demostrarse que CS-LapSVM es un programa convexo cuya solución óptima global puede ser resuelta eficientemente. Además, el paquete LapSVM es público (`http://manifold.cs.uchicago.edu/manifold_regularization/data.html`), por lo que supone una ventaja para propósitos de modificación, la implementación fue sencilla, dado que CS-LapSVM sólo hace una modificación menor a LapSVM. Este kernel puede configurarse fácilmente para asignar pesos a las clases cuando estas no están equilibradas, esto, para penalizar más una solución que clasifica peor para las clases con menos entradas, esta mejoría se refleja en el incremento de medidas de calidad como valor-f, sensibilidad, precisión, MCC, área bajo la curva y especificidad. Estas propiedades tienen un fuerte impacto en este problema particular de clasificación, ya que los conjuntos de datos están significativamente desequilibrados con sólo unos pocos casos positivos, y la gran mayoría de los ejemplos no están etiquetados. Sheng y Zhou (2013) propusieron una validación cruzada de tres pliegues, con dos tercios de ejemplos etiquetados funcionando como un conjunto de datos de entrenamiento y un tercio para el conjunto de datos de prueba.

3.1.5. ProtDCal

Gracias a que en los últimos años se ha visto un gran incremento de bases de datos de secuencias y estructuras tridimensionales de proteínas, ahora estudiar la relación secuencia-estructura-función de una proteína es un área prometedora (Ruiz-Blanco *et al.*, 2015). En este sentido, las características estructurales de una proteína pueden ser transformadas en descriptores moleculares, los cuales pueden ser utilizados con múltiples propósitos en el estudio de dicha relación.

ProtDCal (Programa para el Cálculo de Descriptores de Proteína) es una nueva suite de software computacional que aborda esta necesidad (Ruiz-Blanco *et al.*, 2015). Este programa es capaz de generar decenas de miles de descriptores moleculares, considerando secuencia y estructuras tridimensionales. Dicho software fue desarrollado en

el lenguaje Java (JDK versión 1.7), con esto se proporciona compatibilidad entre plataformas para cualquier sistema en el que esté disponible una máquina virtual Java (JVM). ProtDCal ofrece tres tipos de descriptores: los basados en termodinámica, los cuales incluyen factores implicados en la estabilidad de las estructuras proteicas. Los topográficos, los cuales incluyen muchos de los descriptores relacionados con la velocidad de plegado de proteínas y el orden de contacto relativo. Además, los basados en índices, los cuales incluyen una serie de propiedades fisicoquímicas y estructurales de cada tipo de residuo, tales como hidrofobicidad, índice de carga electrónica, masa molar, volumen y superficie isotrópica. Al final, 6,466 características provenientes de ProtDCal fueron calculadas.

3.1.6. Selección de características.

En esta sección se habla acerca de los métodos para la selección de características usados en este trabajo; dos basados en algoritmos evolutivos (Paul y Das, 2015; Beltrán *et al.*, 2017), el método de envoltura (Beltran, 2015) y las técnicas de filtrado como la descrita en Hall *et al.* (2009) y Mukaka (2012).
En la siguiente sección haremos una descripción detallada de la metodología seguida por Sheng y Zhou (2013), los dos selectores de características basados en algoritmos evolutivos (Paul y Das, 2015; Beltrán *et al.*, 2017), y el método de envoltura (Beltran, 2015).

3.1.7. Algoritmo Evolutivo Basado en Descomposición (MOEA/D)

La descomposición es una estrategia bien conocida en optimización multiobjetivo. Zhang y Li (2007) presentaron un algoritmo evolutivo multiobjetivo basado en descomposición (MOEA/D). Este tiene la capacidad de descomponer un problema de optimización multiobjetivo en un conjunto de subproblemas de optimización escalar y se plantea resolver estos subproblemas simultáneamente. Posteriormente, cada subproblema se optimiza haciendo uso de la información de sus subproblemas vecinos, esto reduce la complejidad computacional de MOEA/D en cada generación. De hecho, más que otros enfoques como la Búsqueda Local Genética Multiobjetiva (MOGLS) (Tian, 2016) y el Algoritmo Genético de Ordenamiento No Dominado II (NSGA -II) (Deb *et al.*, 2002).
Las soluciones óptimas para dos subproblemas vecinos deben ser muy similares, por

lo que también abarca un vecindario de relaciones entre estos subproblemas, este vecindario se define sobre la base de las distancias entre sus vectores de coeficientes de agregación. Si bien MOEA/D trabaja resolviendo subproblemas dependientes, el resultado es una aproximación al frente de Pareto que contiene las soluciones no dominadas que cumplen con todos los objetivos. Al mismo tiempo, en MOEA/D se requiere que cada solución debe ser diferente de las otras tanto como sea posible (**Figura 7**).

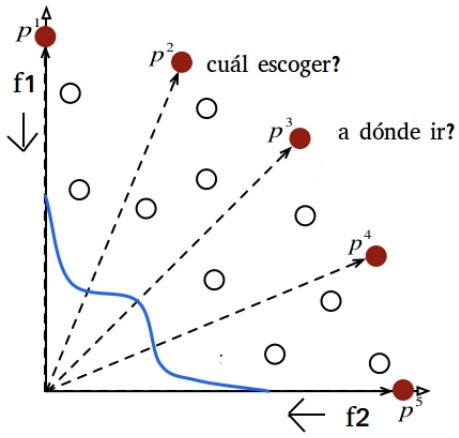

Figura 7. La selección en MOEA/D es un proceso de elección por agentes. Por agente, su solución seleccionada debería optimizar el problema subyacente tanto como se pueda, la solución debería ser diferente del resto de los agentes.

Tres puntos importantes para remarcar acerca de MOEA/D son:

1. MOEA/D comprende un marco para la optimización de problemas multiobjetivo (MOPs), una forma fácil y eficiente de introducir aproximaciones basadas en descomposición en Algoritmos Evolutivos (AE) para resolver MOPs.

2. Debido al hecho de que MOEA/D optimiza *n* problemas de optimización escalar en lugar de resolver todo el MOP, algunas problemáticas podrían surgir, tales como la asignación de aptitud y el mantenimiento de diversidad, estas problemáticas aparecen típicamente en los algoritmos de optimización multiobjetivo (MOEAs).

3. MOEA/D tiene una complejidad computacional menor en cada generación que la vista en enfoques similares, tales como los populares NSGA-II (Deb *et al.*, 2002) y MOGLS (Tian, 2016).

3.1.8. MOEA/D para selección de características

El problema de representar datos con la mínima cantidad de características importantes y discriminativas en grandes conjuntos de datos, esto como parte de un preprocesamiento, es un paso importante a realizar antes de la minería de datos (Paul y Das, 2015). En los últimos años, una enorme cantidad de experimentos biológicos procedentes de tecnologías de alto rendimiento ha llevado a la producción regular de una gran cantidad de grandes conjuntos de datos, los cuales son difíciles de manejar y clasificar. En los casos en los que se utilizan estos conjuntos de datos como entrada para la toma de decisiones, los algoritmos de aprendizaje o sistemas basados en conocimiento conllevan un problema: cuanto mayor es la dimensión de estos datos, mayor tiempo se requiere para clasificarlos. Comúnmente, estos datos son representados por características, algunas de las cuales pueden ser redundantes. La tarea de la selección de características (FS) es ayudar a resolver este problema mediante la eliminación de características no deseables y la selección de aquellas que aportan información valiosa, esto abre un camino para la reducción de tiempo de cálculo y una mejor precisión de los algoritmos de clasificación. Un trabajo de 2015 (Paul y Das, 2015) describe la posibilidad de aplicar el algoritmo MOEA/D para disminuir el número de características, teniendo así un fuerte impacto en el rendimiento de la clasificación. Una clasificación adecuada de entradas pertenecientes a distintas clases requiere que dichas entradas estén tan alejadas unas de otras como sea posible y las de la misma clase deben estar lo más cerca posible entre si. La capacidad de esta separación máxima de la distancia de los puntos inter-clase y la cercanía de los intra-clase depende de las características seleccionadas para representar las entradas. Los autores describen una técnica para minimizar la distancia entre entradas de la misma clase y maximizar la de los de distinta clase con el objetivo de mejorar la clasificación (**Figura 8**).

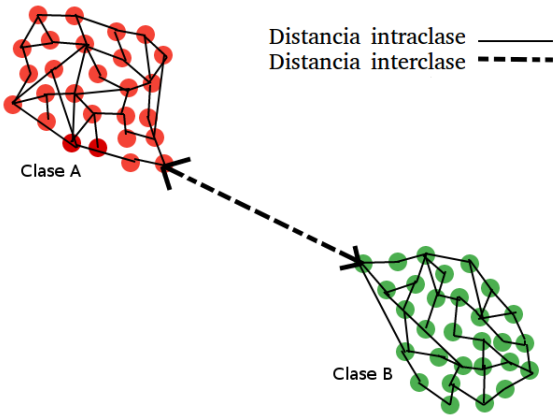

Figura 8. Distancias intra- e inter- clase dentro de un conjunto de datos.

Esta implementación de MOEA/D para selección de características puede ser resumida de la siguiente manera:

Sea $\mathbb{Z}$ un conjunto de datos de tamaño $m \times n$, con m número de características por entrada y n número de entradas. Entonces, cada entrada $z_{i,j}$ representa la $i-ésima$ característica de la entrada $j-ésima$. Además, la etiqueta de cada entrada perteneciente a $\mathbb{Z}$ es representada por el vector $\mathbf{l}$, de tal manera que l_j representa la clase del dato $j-ésimo$. He aquí la importancia de remarcar que las diferentes características de una entrada puedan tener valores de diferente orden. Esa es la razón por la que las características son escaladas a un intervalo de [1, 10] de la siguiente manera:

$$x_{i,j} = 1 + 9\left(\frac{z_{i,j} - \min\limits_{1\leq k\leq n} z_{i,j}}{\max\limits_{1\leq k\leq n} z_{i,j} - \min\limits_{1\leq k\leq n} z_{i,j}}\right). \tag{8}$$

A partir de aquí cualquier operación subsecuente será realizada sobre la matriz $\mathbf{X}$. Ahora, considere la selección de características como un vector de pesos $\mathbf{w}$, de tal manera que w_i es la ponderación de la $i-ésima$ característica, esta última puede ser definida de la siguiente manera,

$$w_i = \begin{cases} 0, & \text{si la característica } x_i \text{ es rechazada,} \\ [1,a], & \text{si la característica } x_i \text{ es seleccionada.} \end{cases} \tag{9}$$

a es considerado como 10 en el algoritmo de Paul y Das (2015). Después de aplicar

los pesos, la $j-ésima$ entrada es representada como:

$$y_j = \mathbf{w} \bullet \mathbf{x_j}, \tag{10}$$

donde $\bullet$ representa el operador de la distancia de Manhattan entre dos vectores. Haciendo uso de las notaciones anteriores, la distancia de Manhattan entre las entradas $p-ésima$ y $q-ésima$ se calcula de la forma:

$$d(y_p, y_q) = \sum_{i=1}^{m} \left| y_{i,p} - y_{i,q} \right| \tag{11}$$

$$= \sum_{i=1}^{m} \left| w_i x_{i,p} - w_i x_{i,q} \right| = \sum_{i=1}^{m} w_i \left| x_{i,p} - x_{i,q} \right|. \tag{12}$$

Así,

$$d(y_p, y_q) = \mathbf{w}^T \left| \mathbf{x_p} - \mathbf{x_q} \right|. \tag{13}$$

Sea $\mathbb{D}$ el conjunto de entrenamiento, y considerando la ecuación anterior, entonces la distancia total intra-clase puede ser calculada de la forma:

$$d_{intra} = \sum_{p=1}^{n-1} \sum_{\substack{q=p+1 \\ \forall y_p = y_q}}^{n} d(y_p, y_q) \tag{14}$$

$$= \sum_{p=1}^{n-1} \sum_{\substack{q=p+1 \\ \forall y_p = y_q}}^{n} \mathbf{w}^T \left| \mathbf{x_p} - \mathbf{x_q} \right| = \mathbf{w}^T \sum_{p=1}^{n-1} \sum_{\substack{q=p+1 \\ \forall y_p = y_q}}^{n} \left| \mathbf{x_p} - \mathbf{x_q} \right|, \tag{15}$$

de este modo, $d_{intra} = \mathbf{w}^T \Delta^{intra}$, donde Δ^{intra} es un vector comprendiendo la distancia intra-clase. La distancia inter-clase puede ser calculada de forma similar de la siguiente manera:

$$d_{inter} = \sum_{p=1}^{n-1} \sum_{\substack{q=p+1 \\ \forall y_p \neq l_q}}^{n} d(y_p, y_q) \tag{16}$$

$$= \sum_{p=1}^{n-1} \sum_{\substack{q=p+1 \\ \forall y_p \neq y_q}}^{n} \mathbf{w}^T \left| \mathbf{x_p} - \mathbf{x_q} \right| = \mathbf{w}^T \sum_{p=1}^{n-1} \sum_{\substack{q=p+1 \\ \forall y_p \neq y_q}}^{n} \left| \mathbf{x_p} - \mathbf{x_q} \right|, \tag{17}$$

por lo que, $d_{inter} = \mathbf{w}^T \Delta^{inter}$, donde Δ^{inter} es un vector que representa la distancia intra-clase.

3.1.9. El mejoramiento de Beltrán *et al.* (2017) para MOEA/D-FS aplicado a clasificación

En el trabajo de Beltrán *et al.* (2017), el MOEA/D fue implementado para un problema de clasificación de secuencias de aminoácidos, específicamente péptidos antimicrobianos (AMPs). Los autores modelan el problema de selección de descriptores moleculares como un problema de optimización multiobjetivo con el fin de mejorar la representación molecular de secuencias, de esta manera mejorando el rendimiento de clasificación. La diferencia con la propuesta original (Paul y Das, 2015) se centró en el método de asignación de pesos a las características. El primero tiene como objetivo minimizar la distancia intra-clase para todas las clases, mientras que el enfoque del trabajo de Beltrán *et al.* (2017) sólo minimiza la distancia intra-clase para la clase positiva (AMPs). La justificación detrás de esto fue que el conjunto de los no AMPs puede contener péptidos con diferentes actividades biológicas, entonces tratar de minimizar la distancia intra-clase para los no-AMP sería contradictorio con el principio de propiedad de similitud. Dada la similitud con nuestro trabajo, hemos probado esta mejoría ya que este problema es similar, hablando en el contexto de las entradas de los conjuntos de datos, en el que las pertenecientes a la clase negativa (secuencias que no se acoplan a miARN) pueden no estar relacionadas en absoluto. Además, el número de pesos distintos de cero se utiliza como criterio de desempate para los vectores de peso con la misma distancia intraclase o interclase, respectivamente.
Por lo tanto, el problema de ponderación de características multiobjetivo se estableció como la minimización de las siguientes funciones:

$$f1(\mathbf{w}) = d_{intra}(\mathbf{w}, \mathbb{D}) + \frac{\left[min\left\{1, \mathbf{w}\right\}\right]^T \mathbf{1}}{m} \tag{18}$$

y

$$f2(\mathbf{w}) = -d_{inter}(\mathbf{w}, \mathbb{D}) + \frac{\left[min\left\{1, \mathbf{w}\right\}\right]^T \mathbf{1}}{m} \tag{19}$$

En estas dos últimas ecuaciones, $\mathbb{D}$ representan al conjunto de entrenamiento con n ejemplos y m el número de características candidatas de entrada. Los autores describen una mejoría sustancial con respecto al original. El término $\left[min\left\{1, w\right\}\right]^T \mathbf{1}$ es el número de pesos que son diferentes de cero. Este término da preferencia a un vector

de pesos con un número menor de características que cualquier otro vector de pesos con la misma distancia intra- o inter- clase.

3.1.10. Técnica de envoltura para selección de características

Por último, otra técnica para selección de características, la cual está ligada a la anterior, es la de envoltura. De acuerdo a Beltran (2015), en esta se utiliza algún algoritmo de optimización con el propósito de minimizar o maximizar funciones de calidad. En el caso de clasificación, esta función a maximizar puede ser el Coeficiente de Correlación de Mattews (MCC), el cual mide la calidad de un clasificador binario (dos clases). Entonces, si el objetivo es optimizar la calidad de clasificación, encontrando el vector de pesos óptimo para el conjunto de características, entonces por cada nueva solución resultante del proceso de optimización de la técnica anterior, se hace una corrida del clasificador utilizado y se mide la medida de calidad, en este caso el MCC, a diferencia de la distancia intra- e inter- clase como funciones objetivo de la técnica anterior.

3.1.11. Algoritmos de filtrado

Una forma de reducir la dimensionalidad de un conjunto de datos es mediante el uso de algoritmos de filtrado, dos de estos algoritmos parecen ser prometedores para este propósito. A continuaciń se describen ambos:

En primer lugar, el filtro llamado RemoveUseless integrado en el paquete Weka (Hall *et al.*, 2009), elimina las características que no varían en absoluto o que no varían según un umbral definido en 98 %, todos los atributos constantes se eliminan automáticamente junto con los que no exceden el parámetro mínimo de porcentaje de varianza fijado por el usuario.

Segundo, otro filtro plausible son los que eliminan las características correlacionadas. El propósito de esto último es calcular la correlación entre pares de atributos dentro de un conjunto de datos, e.g. si un par está correlacionado en un 98 %, entonces uno de ellos debe eliminarse para fines de reducción de dimensionalidad de datos. Para los cálculos de correlación, el índice de correlación de Spearman (Mukaka, 2012) es una buena opción, siendo una medida no paramétrica de correlación de rangos (dependencia estadística entre la clasificación de dos variables). Este último calcula el grado de relación entre dos variables que puedan ser descritas utilizando una función

monotónica (**Figura 9**).

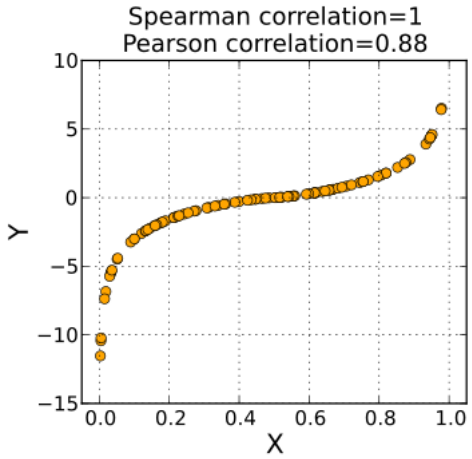

Figura 9. Una correlación de Spearman de 1 resulta cuando dos variables siendo comparadas están monotónicamente relacionadas, aun si su relación no es lineal

3.1.12. AutoDock Vina como método para incrementar el número de ejemplos de entrenamiento

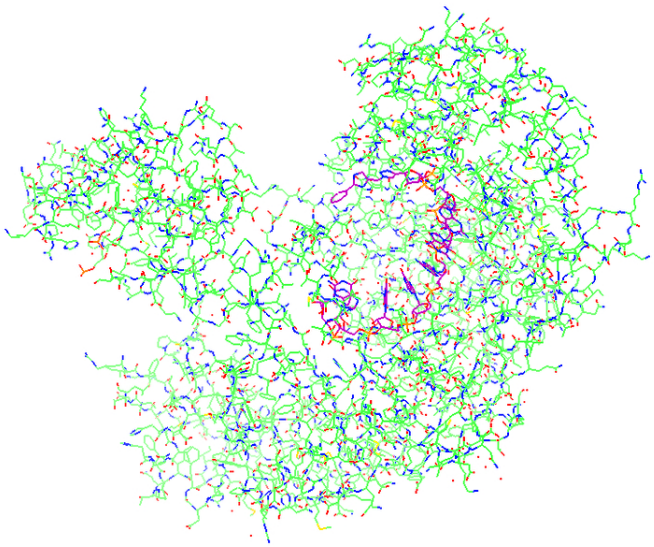

Figura 10. Acoplamiento de la estructura PDB 5KI6 (complejo proteico PDB miARN-proteína) realizado con AutoDock Vina.

AutoDock Vina (Trott y Olson, 2010) es un software de acoplamiento molecular y cribado virtual desarrollado en los laboratorios del Instituo de Investigacion Scripps en La Jolla, California. AutoDock Vina consigue un mejoramiento del doble en tiempo de

corrida en comparación con el software de acoplamiento molecular previamente desarrollado (AutoDock 4), al mismo tiempo que mejora significativamente la precisión de las predicciones según las pruebas realizadas por los autores con sus conjuntos de entrenamiento utilizados en el desarrollo de AutoDock 4. La aceleración adicional se logra a partir del paralelizado, utilizando “multithreading” en máquinas multi-hilo. AutoDock Vina calcula automáticamente los acoplamientos candidatos y agrupa los resultados de una manera transparente para el usuario.

Entonces, se propone la siguiente metodología para incrementar el número de ejemplos positivos: tomar las estructuras de proteínas previamente etiquetadas con la capacidad de unión a miARN, utilizar una técnica de acoplamiento ciego, en la que su respectiva secuencia de miARN sea utilizada para correr el acoplamiento molecular, se dividirá la estructura tridimensional en secciones equivalentes al tamaño máximo de una caja del tipo cuadrícula, que pueda contener a esa sección de la proteína y al miARN candidato, después, analizar las configuraciones con menor energía libre y tomar la mejor conformación. Después, utilizar los complejos miARN-proteína resultantes siguiendo la misma metodología de Sheng y Zhou (2013).

Para una validación más precisa se puede recurrir a simulaciones de dinámica molecular para estudiar la estabilidad de los complejos en las configuraciones de acoplamiento predichas por Autodock Vina (**Figura 10**).

Capítulo 4. Experimentos y resultados

Este capítulo describe los resultados obtenidos al seguir la metodología expuesta en el trabajo de Sheng y Zhou (2013) para la obtención del conjunto de datos y de los descriptores moleculares que pueden contener características con capacidad de discriminar péptidos con y sin capacidad de acoplamiento con miARN. Además se analizó el efecto en la calidad de clasificación de usar conjuntos de datos y características adicionales.

4.1. Conjunto de datos

Como se describió en el **Capítulo 3**, se siguió la metodología de Sheng y Zhou (2013) para la extracción del conjunto de datos. Sheng y Zhou (2013) sentaron las bases de los predictores de interacción miARN-proteína basados en aprendizaje de máquina. La regla para determinar si una secuencia de proteína es un ejemplo positivo (con capacidad de acoplamiento a miARN) es que cualquiera de los átomos pesados de su residuo central se encuentre dentro de un radio de 3.5Å de cualquier átomo del miARN. Para el cálculo de las distancias entre la proteína y los átomos de miARN se utilizan estructuras en complejo miARN-proteína. Siguiendo la metodología de Sheng y Zhou (2013), se realizó una búsqueda de dichas estructuras en complejo miARN-proteína en el Protein Data Bank (PDB RSCB)[1], resultando en 5 complejos miARN-proteína tal como reportan los autores. Después, se realizó la técnica de eliminación de redundancia descrita por los autores, en esta, todas las secuencias con al menos 25 % de identidad fueron agrupadas utilizando el programa blastclust (**tablas 2-3**) del paquete BLAST del NCBI (Altschul, 1997). Solo se mantuvo la secuencia más larga de cada grupo, y el resultado fue 4 cadenas provenientes de 3 complejos para propósitos de entrenamiento (**Tabla 2**). Como se describe en el **Capítulo 2** y después de la eliminación de redundancia, se utilizaron 7,983 11-meros provenientes de 16 secuencias de proteínas no etiquetadas provenientes de 16 proteínas con capacidad de unión a miARN (**Tabla 1**).

Con el objetivo de incrementar la calidad de clasificación se aumentó la cantidad de ejemplos positivos. Esto se conisiguió con una búsqueda adicional en el Protein Data Bank, donde se encontraron cinco nuevas estructuras 3D de complejos miARN-proteína, y después de la eliminación de redundancia se mantuvieron las proteínas:

[1] 1 de septiembre de 2016.

2N82, 4F3T y 5KI6 (identificadores de PDBs).

4.2. La replicación del trabajo de Sheng y Zhou (2013)

Siguiendo la metodología y nombres de los conjuntos de datos y características descritos en el **Capítulo 3**, los dos clasificadores de este trabajo son identificados de la siguiente forma: **S()** representa aquellas corridas con el clasificador supervisado y **SS()** representa aquellas corridas con el clasificador semi-supervisado. A los resultados del clasificador semi-supervisado reportado por Sheng y Zhou (2013) se lo identifica como **SH(D1,C1)**.

A manera de ejemplo: **SS(D1,D2,C2)** representa una corrida del clasificador semi-supervisado utilizando el conjunto de datos **D1** y los conjuntos de características **C1** y **C2**.

Siguiendo la metodología de Sheng y Zhou (2013) y nuestro enfoque supervisado, cuatro cadenas de proteína permanecieron después de la eliminación de redundancia, lo que resultó en sólo cuatro cadenas de proteínas para entrenar: 3TS0, cadena B, 3ADI, cadena A, 3A6P, cadena A y 3A6P, cadena C. Con el propósito de dilucidar si alguna de las cadenas tiene una mayor relevancia en la clasificación, se corrió el software proporcionado por los autores con una validación cruzada de tres pliegues y utilizando los conjuntos **D1** y **C1**, tal como especifican los autores. Los resultados se muestran por cadena en las **tablas 2-5**, la cadena A de 3ADI y la cadena A de 3A6P obtuvieron mejores resultados en todas las medidas de calidad. Nuestro clasificador supervisado tuvo mejores medidas de calidad en términos de valor-F, sensibilidad, precisión, MCC, AUC y especificidad, con respecto a **SH(D1,C1)** y la replicación semi-supervisada de su metodología. Después, y con el propósito de tener un punto de comparación entre el enfoque semi-supervisado de los autores y un enfoque supervisado donde solo se utilizaran los ejemplos etiquetados para entrenamiento, se desarrolló **S()**, el cual está basado en la API Scikit-learn (Sklearn) (Pedregosa *et al.*, 2011), la cual se utilizó para desarrollar un clasificador base: un SVM con un kernel laplaciano implementado con Sklearn para Python. El parámetro C fue fijado a 0.3 y otros parámetros fueron usados en sus valores predefinidos del programa. Sklearn proporciona una base para la asignación de peso a las diferentes clases, entonces siguiendo la metodología de Sheng y Zhou (2013), por cada iteración de validación cruzada se asignó un peso a las clases con el fin de superar el problema de desbalance entre clases (más ejemplos negativos

que positivos), tal peso se calculó utilizando la inversa de la cantidad de características por clase.

Con respecto a **SH(D1,C1)** y a **SS(D1,C1)**, **S(D1,C1)** logró un mayor rendimiento en todas las medidas de calidad, esto incluye el valor-F (**Tabla 9**), el cual permite obtener una relación entre la precisión y la especificidad, esto mide la capacidad de reconocer ejemplos positivos, una de las principales dificultades en este problema, además, una precisión superior (cuántos de los ejemplos clasificados positivos son relevantes), así también, un mejor coeficiente de correlación de Matthews (MCC), el cual evalúa la correlación entre lo predicho por el clasificador y lo que se observa en la realidad. Este clasificador tuvo un MCC de 0.70, mientras que **SH(D1,C1)** 0.27 (**Tabla 9**).

Tabla 9. Replicación del trabajo de Sheng y Zhou (2013) (validación cruzada de tres pliegues para el conjunto de prueba (aprendizaje semi-supervisado)).

	PQ	NQ	TP	FP	TN	FN	Val-F	Sen	Pre	MCC	AUC	Esp	No.C
SH(D1,C1)							0.26	0.63	0.17	0.27	0.81	-	304
SS(D1,C1)	20.33	432.67	12.67	22.67	410.00	7.67	0.46	0.64	0.38	0.45	0.79	0.89	304
SS(D1,D2,C1,C2)	45.33	432.67	31.00	45.67	387.00	14.33	0.51	0.69	0.41	0.46	0.79	0.89	335

Tabla 10. Replicación del trabajo de Sheng y Zhou (2013) (validación cruzada de diez pliegues para el conjunto de prueba (aprendizaje semi-supervisado)).

	PQ	NQ	TP	FP	TN	FN	Val-F	Sen	Pre	MCC	AUC	Esp	No.C
SS(D1,C1)	6.10	129.80	4.10	7.30	122.50	2.00	0.44	0.63	0.35	0.43	0.79	0.94	304
SS(D1,D2,C1,C2)	13.60	129.80	9.00	13.80	116.00	4.60	0.49	0.68	0.39	0.45	0.79	0.89	335

Además, se pudo observar un patrón en las corridas para una validación cruzada de tres y diez pliegues (**tablas 9-10**). Al comparar los enfoques semi-supervisados, los clasificadores utilizando los conjuntos **D2** y **C2** tuvieron mejores medidas de calidad en general (**tablas 12-13**). Por otro lado, al comparar los enfoques supervisados, los clasificadores que son entrenados con los conjuntos **D1** y **C2** tuvieron mejores medidas de calidad en general (**tablas 22-23**) (**Apéndice (A.5)**).

4.3. La aplicación de los algoritmos de selección de características

Después de la replicación del enfoque semi-supervisado de Sheng y Zhou (2013) y la creación de una solución de enfoque supervisado, se prosiguió con la aplicación de selección de características, según los tres métodos descritos en el **Capítulo 3**: abordar el problema de selección de características como un problema multi-objetivo con MOEA/D (Paul y Das, 2015), la variante de Beltrán *et al.* (2017) y la técnica de

envoltura (Kohavi y John, 1997), en la implementación de Beltrán (2014).

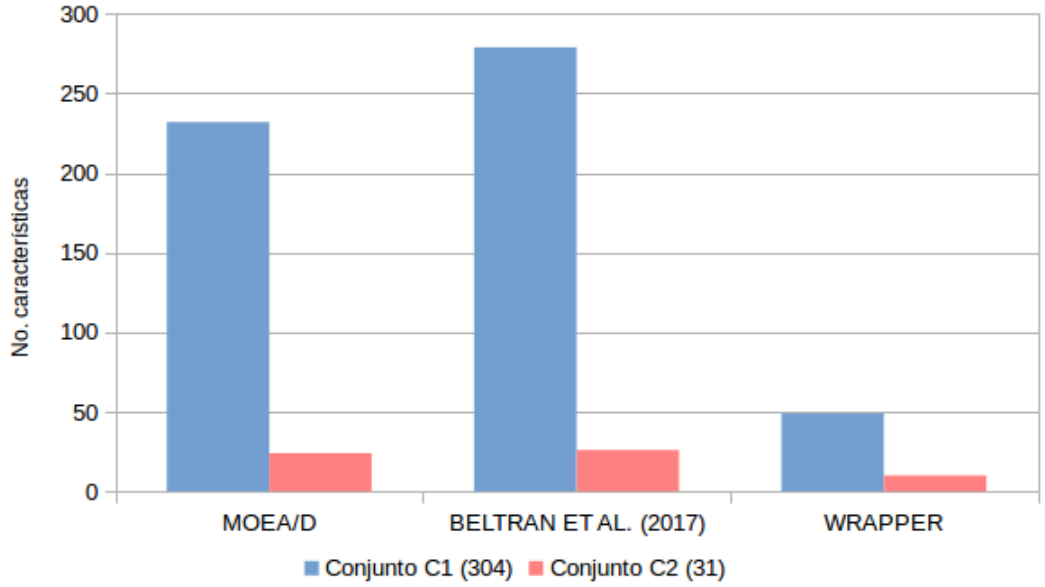

Figura 11. Distribucion del número de características después de la selección con las tres técnicas.

Al aplicar selección de características aplicando MOEA/D, se seleccionó 232 de 304 características (**76.32%**) del conjunto **C1** y 24 de 31 características (**77.42%**) del conjunto **C2** (**Figura 11**).

Al aplicar selección de características aplicando MOEA/D con la modificación de Beltrán *et al.* (2017), se seleccionó 279 de 304 características (**91.77%**) del conjunto **C1** y 26 de 31 características (**83.87%**) del conjunto **C2** (**Figura 11**).

Por último, al aplicar la técnica de envoltura se seleccionó 49 de 304 características (**16.12%**) de las caracterśiticas del conjunto **C1** y 10 de 31 características (**32.26%**) del conjunto **C2** (**Figura 11**).

Adicionalmente, se analizó la distribución de la selección de los tres algoritmos con el fin de buscar un patrón o firma que distinguiese subconjuntos de características relevantes o con poder discriminador, lo cual nos deja ver que no existe correlación alguna de preferencia entre los dos métodos de selección basados en MOEA/D y la técnica de envoltura, sin embargo, entre los dos primeros sí se observó la misma tendencia en la selección (**Tabla 11**).

Tabla 11. Distribución de características entre los algoritmos de selección del conjunto C2.

	MOEA/D	BELTRAN	WRAPPER
Propensión mutua (18)	11 (4.78%)	17 (5.68%)	1 (2.04%)
PSSMs (220)	146 (63.47%)	198 (66.22%)	7 (14.28%)
Índices HKM (66)	50 (21.73%)	58 (19.39%)	31 (63.26%)
Composición 20 aminoácidos (20)	13 (5.65%)	16 (5.35%)	7 (3.18%)
Composición de 9 grupos de aminoácidos (9)	9 (3.91%)	8 (2.67%)	3 (6.12%)
Punto isoeléctrico (1)	1 (0.43%)	1 (0.33%)	0 (0%)
Peso molecular (1)	0 (0%)	1 (0.33%)	0 (0%)
Total	**232 (100%)**	**279 (100%)**	**49 (100%)**

Lo que se puede observar en los enfoques semi-supervisados es que si bien no existió una técnica de selección de características que haya tenido resultados superiores en ambas validaciones cruzadas de tres y diez pliegues, **SS(D1,C3)** obtuvo resultados superiores para tres pliegues (**Tabla 12**) y **SS(D1,D2,C4)** para diez (**Tabla 13**), en ambos casos **SS(D1,C5)** logró resultados muy cercanos a los mejores valores para cada criterio con una menor cantidad de datos positivos y una mucho menor cantidad de características.

Tabla 12. Aplicación de selección de características (validación cruzada, tres pliegues para conjunto de prueba (aprendizaje semi-supervisado)).

	PQ	NQ	TP	FP	TN	FN	Val-F	Sen	Pre	MCC	AUC	Esp	No.C
SS(D1,C3)	20.33	432.67	19.00	43.00	389.67	1.33	0.60	1.00	0.51	0.69	0.98	0.95	232
SS(D1,C4)	20.33	432.67	16.33	36.00	396.67	4.00	0.45	0.79	0.31	0.46	0.86	0.92	279
SS(D1,C5)	45.33	432.67	41.33	65.00	367.67	4.00	0.59	0.89	0.44	0.58	0.89	0.88	49
SS(D1,D2,C3)	45.33	432.67	37.33	28.67	404.00	8.00	0.67	0.82	0.57	0.64	0.88	0.93	232
SS(D1,D2,C4)	45.33	432.67	37.33	30.67	402.00	8.00	0.66	0.83	0.55	0.63	0.88	0.93	279

Tabla 13. Aplicación de selección de características (validación cruzada, diez pliegues para conjunto de prueba (aprendizaje semi-supervisado)).

	PQ	NQ	TP	FP	TN	FN	Val-F	Sen	Pre	MCC	AUC	Esp	No.C
SS(D1,C3)	6.10	129.80	5.60	11.00	118.80	0.50	0.48	0.91	0.34	0.52	0.91	0.91	232
SS(D1,C4)	6.10	129.80	5.40	10.80	119.00	0.70	0.48	0.87	0.34	0.50	0.89	0.92	279
SS(D1,C5)	6.10	129.80	6.10	52.10	77.70	0.00	0.60	0.94	0.45	0.60	0.91	0.88	49
SS(D1,D2,C3)	13.60	129.80	11.2	8.40	121.40	2.40	0.67	0.82	0.57	0.64	0.88	0.93	232
SS(D1,D2,C4)	13.60	129.80	11.30	7.70	122.10	2.30	0.70	0.84	0.60	0.68	0.89	0.94	279

Por otra parte, en los enfoques supervisados se puede observar un claro patrón, en el que **S(D1,C4)** fue el clasificador con mejores medidas de calidad (**tablas 14-15**), tanto para tres como para diez pliegues, también **S(D1,C5)** logró resultados muy cercanos a las mejores valores con una menor cantidad de datos positivos y una mucho menor cantidad de características

Tabla 14. Aplicación de selección de características (validación cruzada, tres pliegues para conjunto de prueba (aprendizaje supervisado)).

	PQ	NQ	TP	FP	TN	FN	Val-F	Sen	Pre	MCC	AUC	Esp	No.C
S(D1,C3)	20.33	432.67	20.33	21.00	411.67	0.00	0.45	0.92	0.30	0.49	0.91	0.90	232
S(D1,C4)	20.33	432.67	20.33	21.00	411.67	0.00	0.66	1.00	0.50	0.69	0.98	0.95	279
S(D1,C5)	45.33	407.67	43.00	43.00	364.67	2.33	0.68	0.99	0.52	0.68	0.94	0.90	49
S(D1,D2,C3)	45.33	407.67	45.33	45.33	362.33	0.00	0.67	1.00	0.50	0.67	0.94	0.89	232
S(D1,D2,C4)	45.33	407.67	45.33	45.33	362.33	0.00	0.67	1.00	0.50	0.67	0.94	0.89	279

Tabla 15. Aplicación de selección de características (validación cruzada, diez pliegues para conjunto de prueba (aprendizaje supervisado)).

	PQ	NQ	TP	FP	TN	FN	Val-F	Sen	Pre	MCC	AUC	Esp	No.C
S(D1,C3)	6.10	129.80	6.10	49.70	80.10	0.00	0.20	1.00	0.11	0.26	0.81	0.62	269
S(D1,C4)	6.10	129.80	6.10	49.70	80.10	0.00	0.20	1.00	0.11	0.26	0.81	0.62	267
S(D1,C5)	13.60	122.30	13.60	100.50	21.80	0.00	0.21	1.00	0.12	0.15	0.59	0.18	49
S(D1,D2,C3)	13.60	122.30	13.60	108.80	13.50	0.00	0.20	1.00	0.11	0.11	0.56	0.11	256
S(D1,D2,C4)	13.60	122.30	13.60	108.80	13.50	0.00	0.20	1.00	0.11	0.11	0.56	0.11	305

Por lo tanto, se observa que ambas técnicas basadas en MOEA/D son efectivas y muestran resultados superiores a los enfoques sin la aplicación selección de características, pero también que la técnica de envoltura para selección de características resulta ser similar con una cantidad inferior de características.

4.4. La utilización de características provenientes de ProtDCal

En un esfuerzo por incrementar la cantidad de características, se hizo uso de la herramienta ProtDCal (**Capítulo 3**).

En total ProtDCal proporciona 6,466 diferentes descriptores moleculares basados en termodinámica, los cuales incluyen factores implicados en la estabilidad de las estructuras proteicas, los topográficos, los cuales incluyen muchos de los descriptores relacionados con la velocidad de plegado de proteínas y el orden de contacto relativo, y los basados en índices, los cuales incluyen una serie de propiedades fisicoquímicas y estructurales de cada tipo de residuo, tales como hidrofobicidad, índice de carga electrónica, masa molar, volumen y superficie isotrópica.

En el caso de los enfoques semi-supervisados se puede observar que para la validación cruzada de tres pliegues la mejor solución con una ligera ventaja frente a las demás fue **SS(D1,D2,C11)** (**Tabla 16**). Para la validación cruzada de diez pliegues la mejor solución con una ligera ventaja frente a las demás fue **SS(D1,D2,C10)** (**Tabla 17**). Para esta última el aporte de selección de características fue inexistente. En ambas

validaciones y en términos de todas las medidas de calidad a excepción de sensibilidad, estas fueron de baja calidad.

Tabla 16. Utilización de características provenientes de ProtDCal (validación cruzada de tres pliegues para el conjunto de prueba (aprendizaje semi-supervisado)).

	PQ	NQ	TP	FP	TN	FN	Val-F	Sen	Pre	MCC	AUC	Esp	No.C
SS(D1,D2,C7)	20.33	432.67	15.33	399.33	33.33	5.00	0.07	0.76	0.04	-0.12	0.42	0.08	6466
SS(D1,D2,C10)	20.33	432.67	15.00	402.33	30.33	5.33	0.07	0.74	0.04	-0.15	0.40	0.07	2786
SS(D1,D2,C11)	20.33	432.67	16.33	409.00	23.67	4.00	0.07	0.79	0.04	-0.13	0.42	0.05	2493

Tabla 17. Utilización de características provenientes de ProtDCal (validación cruzada de diez pliegues para el conjunto de prueba (aprendizaje semi-supervisado)).

	PQ	NQ	TP	FP	TN	FN	Val-F	Sen	Pre	MCC	AUC	Esp	No.C
SS(D1,D2,C7)	6.10	129.80	3.70	111.00	18.80	2.40	0.07	0.75	0.04	-0.12	0.42	0.08	6466
SS(D1,D2,C10)	6.10	129.80	3.70	111.00	18.80	2.40	0.07	0.76	0.04	-0.11	0.42	0.08	2786
SS(D1,D2,C11)	6.10	129.80	4.30	120.30	9.50	1.80	0.07	0.69	0.03	-0.17	0.38	0.07	2493

En el caso de los enfoques supervisados se puede observar que para ambas validaciones cruzadas de tres y diez pliegues la mejor solución con una ligera ventaja frente a las demás fue **S(D1,D2,C11)** (**tablas 18-19**). En este caso se observa el siguiente fenómeno: al aumentar la cantidad de pliegues para validación cruzada, **S(D1,D2,C11)** baja considerablemente de calidad en términos de precisión, y en consecuencia de valor-F.

Tabla 18. Utilización de características provenientes de ProtDCal (validación cruzada de tres pliegues para el conjunto de prueba (aprendizaje supervisado)).

	PQ	NQ	TP	FP	TN	FN	Val-F	Sen	Pre	MCC	AUC	Esp	No.C
S(D1,D2,C7)	20.33	432.67	20.33	21.00	411.67	0.00	0.66	1.00	0.50	0.68	0.98	0.95	6466
S(D1,D2,C10)	20.33	432.67	20.33	21.33	411.33	0.00	0.66	1.00	0.49	0.68	0.98	0.95	2786
S(D1,D2,C11)	20.33	432.67	20.33	21.00	411.67	0.00	0.66	1.00	0.50	0.69	0.98	0.95	2493

Tabla 19. Utilización de características provenientes de ProtDCal (validación cruzada de diez pliegues para el conjunto de prueba (aprendizaje supervisado)).

	PQ	NQ	TP	FP	TN	FN	Val-F	Sen	Pre	MCC	AUC	Esp	No.C
S(D1,D2,C7)	6.10	129.80	6.10	50.40	79.40	0.00	0.19	1.00	0.11	0.25	0.81	0.61	6466
S(D1,D2,C10)	6.10	129.80	3.70	111.00	18.80	2.40	0.19	1.00	0.11	0.25	0.81	0.61	2786
S(D1,D2,C11)	6.10	129.80	6.10	49.60	80.20	0.00	0.20	1.00	0.11	0.26	0.81	0.62	2493

4.5. Comparación de los mejores conjuntos de características

De todos los enfoques discutidos en las secciones anteriores, **SS(D1,D2,C4)** con una validación cruzada de diez pliegues fue el más prometedor, este enfoque semi-supervisado tuvo un mejor nivel y balance entre sensibilidad y precisión, dado el tipo

de problema a resolver, estos criterios son los más relevantes.

Dado que estos resultados fueron obtenidos utilizando los conjuntos de datos **D1** más **D2** y los conjuntos de características **C1** más **C2**, entonces con el propósito de dilucidar si estos resultados se debieron a los 75 ejemplos de entrenamiento positivos del conjunto **D2** o a las características del conjunto **C2**, se entrenó la SVM semi-supervisada solo agregando los 75 ejemplos positivos de **D2** (**SS(D1,D4,C4)**) y solo agregando las características de **C2** (**SS(D1,D2,C9)**) (**Tabla 20**).

Tabla 20. Comparación de **SH(D1,C1)** con las mejores soluciones.

	PQ	NQ	TP	FP	TN	FN	Val-F	Sen	Pre	MCC	AUC	Esp	No.C
SH(D1,C1)							0.26	0.63	0.17	0.27	0.81	-	304
SS(D1,C1)	6.10	129.80	5.40	10.80	119.00	0.70	0.48	0.87	0.34	0.50	0.89	0.92	279
SS(D1,D2,C4)	13.60	129.80	11.30	7.70	122.10	2.30	0.70	0.84	0.60	0.68	0.89	0.94	279
SS(D1,D3,C4)	6.10	129.80	11.60	7.40	122.40	2.00	0.70	0.84	0.61	0.68	0.89	0.94	305
SS(D1,D4,C4)	7.50	129.80	5.90	8.00	121.80	1.60	0.53	0.79	0.41	0.53	0.87	0.94	305
SS(D1,D2,C9)	13.60	129.80	10.80	9.00	120.80	2.80	0.65	0.81	0.55	0.62	0.87	0.93	279
SS(D1,C5)	6.10	129.80	6.10	52.10	77.70	0.00	0.60	0.94	0.45	0.60	0.91	0.88	49

En la **Tabla 20** se muestra la inexistencia de una significativa ventaja de **SS(D1,D2,C9)** con respecto a **SS(D1,D2,C4)**, pero sí una ligera mejora de **SS(D1,D3,C4)** con respecto a **SS(D1,D2,C4)**, y de manera significativa con respecto a **SH(D1,C1)**; por lo que se puede conjeturar que las 31 características adicionales del conjunto de características **C2** tuvieron un impacto positivo en la clasificación, pero no los 75 nuevos ejemplos positivos de **D2**. Esto sugiere que una cantidad mayor de ejemplos positivos de entrenamiento es necesaria para observar una mejora significativa en la calidad de clasificación o que el criterio de decisión del modelo de Sheng y Zhou (2013) no supone un buen modelo para decidir la clase de un ejemplo de entrenamiento.

4.6. Análisis de contribución del conjunto de datos D1 y D2 a la clasificación

Una de las preguntas que surge ahora es la contribución de los datos del conjunto **D1** y los datos del conjunto **D2**, por lo que se entrenó la SVM semi-supervisada, con una validación cruzada de diez pliegues, utilizando los 75 ejemplos de entrenamiento positivos del conjunto **D2** más sus negativos (los mismos que **D1** (**SS(D1,D4,C4)**), en este caso se muestra un comportamiento similar a **SS(D1,D3,C4)** en términos de valor-F, sensibilidad, precisión, MCC y AUC, sin embargo, ninguna medidad de calidad fue superada con respecto a **SS(D1,D3,C4)**, esto confirma la superioridad del conjunto de ejemplos de entrenamiento positivos de **D1** y la superioridad de **SS(D1,D3,C4)**,

y que el responsable del aumento en las medidas de calidad son los ejemplos de entrenamiento positivos del conjunto **D1**, no los de **D2**. Esto a su vez es una motivación para descubrir los motivos de este comportamiento, con este propósito se utilizó la herramienta de agrupamiento de secuencias CD-HIT (Li y Godzik, 2006). Esta última fue utilizada con los parámetros por omisión y un umbral de identidad de 40 %, los resultados pueden ser consultados en el **Apéndice A (A.3)**. El resultado del agrupamiento fue 49 grupos, los cuales todos con excepción de uno, contienen solo secuencias consecutivas con respecto a su orden de extracción de su proteína origen. Esto sugiere la inexistencia de una similitud evidente entre las secuencias positivas de cada uno de los conjuntos de entrenamiento (**D1** y **D2**). Entonces, con el propósito de corroborar esta afirmación se realizó un análisis de frecuencias relativas de los 20 aminoácidos escenciales y posibles motivos ocultos en los diferentes conjuntos de ejemplos positivos, los del conjunto **D1**, los del conjunto **D2** y los de la unión de **D1 y D2**. Para tal tarea se utilizó el paquete MEME Suite (Bailey *et al.*, 2009) en su versión online, el cual proporciona una interfaz de fácil uso. Se realizaron tres corridas con un corrimiento de un residuo por ventana, la primera, con los ejemplos positivos del conjunto **D1** como entrada, la segunda, con los del conjunto **D2**, y la tercera los de la unión de **D1** y **D2**. Los resultados mostraron un patrón de similitud muy alto en términos de composición de aminoácidos entre los tres conjuntos de datos (para consultar los motivos y sus valores-E, ver en **Apéndice (A.4)**) y no se reportó una desviación estándar considerable (Apéndice (A.6)). Como se mencionó, también se buscó encontrar motivos de secuencia, lo cuales pudiesen explicar los diferentes comportamientos de los diferentes conjuntos de ejemplos positivos, los resultados mostraron una total diferencia y ningún motivo se repitió entre los conjuntos (**figuras 14-16** del **Apéndice A**).

4.7. El uso de AutoDock Vina para incrementar el número de ejemplos de entrenamiento

Finalmente, y con el objetivo de analizar la viabilidad de AutoDock Vina para encontrar la región de acoplamiento de una estructura 3D de proteína no etiquetada y los miARNs reportados con capacidad de acoplarse a dicha estructura, se realizó un acoplamiento de los 4 archivos PDBs con los que se he venido trabajando en esta tesis, estos son: 2N82, 3A6P, 3ADI y 5KI6 (**Figura 8**). También desarrollado en el Instituto Scripps (La Jolla, California) se encuentra el software MGLTools, el cual es un visualiza-

dor con capacidad de preparar y transformar los archivos PDB a archivos PDBQT, los cuales son los datos de entrada de AutoDock Vina. Se realizó el acoplamiento de las 4 estructuras PDB mencionados anteriormente. Los parámetros fueron dejados en sus valores pre-establecidos, se eliminaron las moléculas de agua y se agregaron los átomos de hidrógeno a todas las moléculas incluidas en los complejos miARN-proteína.
El propósito de esta metodología es primero, comprobar que AutoDock Vina tiene la capacidad de predecir los blancos, para después realizar el mismo acoplamiento con las estructuras PDB que no están en complejo con sus respectivas secuencias de miARN (estructuras 3D de ejemplos no etiquetados). Para lograr esto se necesita realizar una búsqueda de los identificadores de las secuencias de miARN que tengan la capacidad de acoplarse a cada una de las estructuras no etiquetadas, para su posterior acoplamiento, y de este modo esas estructuras sean transformadas a más ejemplos positivos y negativos, algo que ha sido el principal problema durante la realización de este trabajo de investigación. Finalmente, se logró que los cuatro complejos miARN-proteína se acoplaran con las cavidades previamente documentadas como sitios de acoplamiento, sus afinidades pueden ser encontradas en la **Tabla 21**.

Tabla 21. Resultados de AutoDock Vina con los 4 complejos miARN-proteína

Complejo	Afinidad (kcal/mol)	Desviación RMSD con respecto al complejo original
2N82	-36.90	0.00Å
3A6P	-14.50	0.00Å
3ADI	-41.50	0.00Å
5KI6	-33.15	0.00Å

Capítulo 5. Discusión y conclusiones

5.1. Discusión

El desarrollo de clasificadores para la discriminación de proteínas que se unen a miARN de las que no, representa un reto importante por varias razones, desde la falta ejemplos positivos hasta preguntas más profundas concernientes al mismo método computacional. En esta sección se discute acerca de estos puntos.

5.1.1. Falta de ejemplos de entrenamiento

Sin duda la falta de ejemplos de entrenamiento positivos ha tenido un impacto en el desempeño general de la clasificación. En este caso solo cuatro complejos miARN-proteína pudieron ser recuperados de la literatura (Sheng y Zhou, 2013). En esta investigación se pudieron recolectar cuatro adicionales. En el capítulo anterior se demostró que los ejemplos positivos adicionales del conjunto de datos extendido no tuvieron un impacto positivo en la clasificación, incluso utilizando solo los 61 ejemplos positvos originales se tuvo un mejor desempeño del clasificador.

Con el objetivo de descifrar las causas subyacentes de este comportamiento se realizaron diversas pruebas tales como búsqueda de motivos, agrupamiento de secuencias y un análisis de frecuencias de los 20 aminoácidos escenciales, sin embargo, las frecuencias fueron similares, por lo tanto se puede concluir que la frecuencia relativa no es un patrón a evaluar para encontrar dichas diferencias o el porqué de la contribución diferente de los 61 ejemplos positivos originales y los 75 adicionales, en la clasificación. Por lo tanto, identificar nuevos complejos miARN-proteína en el PDB u otras bases de datos debe ser considerada por la comunidad científica como un paso fundametal para este tipo de estrategias basadas en aprendizaje de máquina.

Para este propósito, softwares como AutoDock Vina parecen prometedores para describir y estudiar las interacciones miARN-proteína. En una prueba sencilla utilizando AutoDock Vina, donde se tomaron estructuras 3D anotadas en complejo miARN-proteína, se aislaron sus moléculas por tipo (proteína y miARN), y se acoplaron con el software mencionado, se determinó que este es capaz de encontrar las cavidades de los complejos miARN-proteína originales, de manera precisa. De aquí que estas herramientas de acoplamiento molecular puedan ser utilizadas para discriminar las regiones reales de interacción de las que no lo son, y entrenar con ventanas con interacción comprobada y no con ventanas vecinas de otras que sean responsables de la cercanía de esa

subcadena. También, como fue descrito en el capítulo anterior, existe una gran cantidad de ejemplos positivos y estos pueden ser transformados a ejemplos etiquetados si se encuentran las secuencias de miARN que se conocen tienen interacción con dichas estructuras. El trabajo restante sería encontrar esas regiones y extraer ejemplos positivos. Al mismo tiempo, Diñamica Molecular es una solución que en los últimos años ha sido descrita como altamente confiable para estudiar cualquier tipo de interacciones físicoquímicas, y puede ser utilizada como una segunda prueba después de un cribado de posibilidades de interacción (posiciones de las moléculas, ángulos, etc) en estructuras no etiquetadas, i.e., las estructuras que se conocen por tener interacción con secuencias de miARN, pero no se conoce la región en particular. Entonces, una revisión más a fondo de los diferentes mecanismos de interacción miARN-proteína debe ser tomada en cuenta en la continuación de este trabajo de investigación. Sin un adecuado análisis de la exactitud de dicho modelo, no se tiene una certeza de si las técnicas computacionales que han sido aplicadas, tales como selección de características y algoritmos de filtrado, esten reflejando resultados relevantes.

5.1.2. Preguntas acerca del modelo computacional

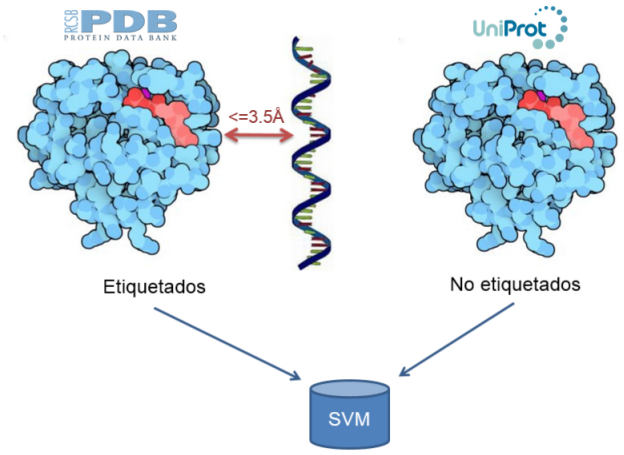

Figura 12. Distancia de 3.5Å como criterio de decisión.

Los modelos computacionales deben ser necesariamente tan apegados a la realidad como sea posible. En esta tesis el modelo computacional está basado en el trabajo de Sheng y Zhou (2013) y se supone que seguir su metodología para extracción de datos, con los cuales se entrena el modelo de aprendizaje de máquina, garantiza una alta confiabilidad en las clases a las que los datos pertenecen. Este es un punto impor-

tante a considerar al tratar de descifrar los motivos por los que se tuvo discrepancia al agregar al conjunto de datos **D1** los ejemplos de entrenamiento positivos del conjunto de datos **D2**, y las medidas de calidad que no fueron mejoradas.

Este modelo supone la siguiente premisa: si se divide una secuencia de una determinada proteína en ventanas, y si alguno de los átomos del residuo central se encuentran a una distancia de 3.5Å o menos de alguno de los átomos de la secuencia de miARN con la que interactua (**Figura 12**), entonces se considera que existe una interacción y esa ventana debe tomarase como un ejemplo positivo, de otra manera, como un ejemplo negativo. Esta es la forma de obtención de datos, sin embargo, esta premisa puede estar alejada de lo que ocurre en la naturaleza o no capturar todos los fenómenos involucrados en la interacción. De acuerdo a esta metodología se podrían estar capturando ejemplos positivos que no supongan una interacción real, ya que no existe garantía alguna de que una región que abarque varias ventanas positivas, por el motivo de estar cerca del miARN, todos sus residuos se encuentren en interacción, otras podrían ser las causas responsables de la interacción. De este modo existe la posibilidad de que una cantidad considerable de ejemplos positivos en realidad no lo sean, y dada la reducida cantidad de los mismos en relación con los negativos, existe la posibilidad de un sobreentrenamiento de ejemplos positivos falsos. En la misma línea, este fenómeno podría suceder de manera opuesta, y ejemplos negativos podrían ser considerados como positivos, y dado que los ejemplos negativos son la mayoría, entonces esto sería aún más plausible. De esta manera, se podría explicar la razón de por qué no se consiguió lograr una mejora al agregar al entrenamiento los ejemplos positivos adicionales del conjunto de datos extendido, aun cuando existe el principio de que, si se suministra una mayor cantidad de ejemplos de entrenamiento a un modelo de aprendizaje de máquina, mejor deberían ser las medidas de calidad de clasificación (fenómeno observado al comparar la solución **SS(D1,D3,C4)** con **SS(D1,D4,C4)**).

De manera adicional, se buscó características particulares que pudiesen explicar tal discrepancia (agrupramiento de secuencias, búsquedad de motivos y análisis de frecuencias de aminoácidos), desafortunadamente, dichas características no fueron observadas. Esta hipótesis de una incorrecta obtención de los datos de entrenamiento se encontraría sustentada al observar los diferentes mecanismos de interacción del ARN con proteínas.

En la biología molecular existe lo que se conoce como motivos de reconocimiento de

ARN (**Figura 13**), estos motivos son estructuras de aminoácidos que son bien conocidas por tener una forma particular, la cual es óptima para un acoplamiento con determinadas estructuras de ARN. Estos han sido bien estudiados y suponen una interacción directa con ARN (e.g. dedos de zinc (**Figura 13**)), de esta manera, existe la posibilidad de desechar ventanas a lo largo de la secuencia proteíca que no se encuentren dentro de estos motivos estructurales. El no tomar en cuenta los mismos es un punto que consideramos débil en la estrategía de Sheng y Zhou (2013), así como la seguida en este trabajo.

Otro punto débil reside en la diversidad de mecanismos de interacción, ya que no solo la cantidad de ejemplos positivos de entrenamiento es limitada, sino que también la cantidad de estructuras que originan esos ejemplos es limitada. Entonces, estaríamos frente al problema de que el modelo de aprendizaje de máquina no alcanza a aprenderlos, esto a su vez generaría la necesidad de cambiar totalmente el paradigma a uno multi-clase, en donde los ejemplos de entrenamiento no solo pertenezcan a la clase positiva y negativa, sino a grupos de interacción, para esto, la creación de clases nuevas y la organización de ejemplos de entrenamiento sería algo escencial.

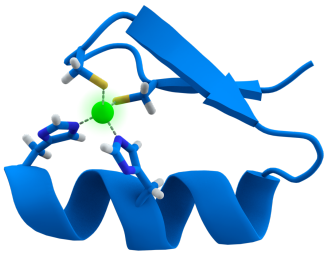

Figura 13. Dedo de zinc, motivo estructural de acoplamiento a ARN (Thomas Splettstoesser (www.scistyle.com)).

Por último, los resultados del experimento de selección de características con la tecnica de envoltura, resalta la debilidad del método de clasificación y de los datos de entrenamiento. En ambos experimentos semi-supervisados con una validación cruzada de tres y diez pliegues, y en el experimento supervisado (validación cruzada de tres pliegues), se observan medidas de calidad no inferiores a las producidas por **SS(D1,D3,C4)**, pero si muy cercanas. Por otra parte, en el ejemplo supervisado con validación cruzada de diez pliegues se observa una sensibilidad alta de 1.00, esto sugiere una tendencia a clasificar bien los ejemplos positivos, pero también a clasificar muchos ejemplos negativos como positivos, consiguiendo de esta manera una

baja precisión. Esto puede sugerir dos premisas: la primera, que estas 49 características obtenidas con el algoritmo de selección de características de envoltura fue muy bueno y concentró las características con mejor capacidad discriminatoria, o segundo, que en realidad se ha producido un sobreentrenamiento de los limitados ejemplos etiquetados. Esta última hipótesis no parece la más plausible ya que la desviación estandar del error no es muy alta en comparación de las medidas de calidad en tales experimentos. En este trabajo se exploraron diversas técnicas para explotar los datos disponibles, al mismo tiempo se llegó a preguntas y críticas, creemos que para un mayor entendimiento y respuesta de las preguntas anteriores acerca del modelo computacional empleado se necesita una mayor fidelidad en los datos de entrenamiento. Esto significa utilizar una técnica de decisión para la asignación de clases más apegada al fenómeno observado, como así también un número mayor de ejemplos de entrenamiento. De aquí surge el llamado a la comunidad científica acerca de la importancia del tema y de la necesidad de estos ejemplos de entrenamiento.

5.2. Conclusiones

- A pesar de que se realizaron esfuerzos, como búsquedas en las bases datos de estructuras de proteína RCSB PDB y Uniprot, y el uso de AutoDock Vina, para incrementar el número de ejemplos de entrenamiento, en especial, los positivos, este no pudo ser incrementado considerablemente, ya que sigue siendo insuficiente y dispar en relación al número de ejemplos negativos.

- La técnica de MOEA/D con la modificación de Beltrán *et al.* (2017) y la técnica de envoltura, aplicadas con el clasificador semi-supervisado resultaron ser superiores que la técnica de MOEA/D para selección de características original. MOEA/D con la modificación de Beltrán *et al.* (2017) fue la técnica utilizada en el clasificador que obtuvo mejores medidas de calidad (**SS(D1,D3,C4)**), por otro lado, la técnica de envoltura fue la utilizada en el clasificador con mejor balance entre medidas de calidad y menor cantidad de características (**SS(D1,C5)**). Por lo tanto, ambas técnicas demostraron ser efectivas, al mejorar los resultados del trabajo de Sheng y Zhou (2013).

- La corrida con mejores medidas de calidad (**SS(D1,D3,C4)**) fue entrenando el clasificador semi-supervisado con los datos del conjunto **D1**, utilizando las carac-

terísticas de los conjuntos **C1** y **C2**, y utilizando MOEA/D con la modificación de Beltrán *et al.* (2017) para selección de características.

- Basado en el punto anterior, se conjetura que el modelo computacional de Sheng y Zhou (2013) es el responsable de que al suministrar más ejemplos de entrenamiento, el clasificador no entregue mejores medidas de calidad (observado en **SS(D1,D3,C4)** vs **SS(D1,D4,C4)** (**Tabla 23**)). Esta afirmación se respalda al no encontrarse diferencias significativas en términos de frecuencias de aminoácidos entre los datos de entrenamiento positivos de **D1** y **D2** (**Figura 11**). Por lo tanto, el modelo computacional de Zheng y Zhou necesita ser revisado.

5.3. Trabajo futuro

Sin duda, incrementar la cantidad de complejos miARN-proteína sigue siendo un desafio a vencer, en este trabajo se hace un llamado a la comunidad científica, acerca de la importancia de hacer disponible esta información con el propósito de explotar la capacidad de estos clasificadores.

De igual manera, y tal como fue discutido previamente en este capítulo, un área de oportunidad aún existente es mejorar el modelo computacional en términos de su capacidad de reflejar la naturaleza de los fenómenos y mecanismos que permiten la interacción miARN-proteína.

Literatura citada

Altschul, S. (1997). Gapped blast and psi-blast: a new generation of protein database search programs. *Nucleic acids research*, **25**(17): 3389–3402.

An, Qin, Xu, Tang, Huang, Situ, Inal, y Zheng (2015). Exosomes serve as tumour markers for personalized diagnostics owing to their important role in cancer metastasis. *Journal of Extracellular Vesicles*, **4**(27522).

Bailey, T., Boden, M., Buske, F., Frith, M., Grant, C., Clementi, L., Ren, J., Li, W., y Noble, W. (2009). Meme suite: tools for motif discovery and searching. *Nucleic acids research*, **37**.

Batagov, A., Kuznetsov, V., y Kurochkin, I. (2011). Identification of nucleotide patterns enriched in secreted rnas as putative cis-acting elements targeting them to exosome nano-vesicles. *BMC Genomics*, **12**(3:S18).

Belkin, M., Niyogi, P., y Sindhwani, V. (2006). Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. **7**(Nov): 2399–2434.

Beltrán, A. (2014). *Métodos para la selección de características y clasificación de péptidos antimicrobianos. Tesis de Maestría en Ciencias. Centro de Investigación Científica y de Educación Superior de Ensenada, Baja California. 44 pp.*.

Beltrán, J., Aguilera-Mendoza, L., y Brizuela, C. (2017). Feature weighting for antimicrobial peptides classification: a multi-objective evolutionary approach. *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 276–283.

Carson, M., Langlois, R., y Lu, H. (2010). Naps: a residue-level nucleic acid- binding prediction server. *Nucleid acids research*, **38**: W431–W435.

Cha, D., Franklin, J., Dou, Y., Liu, Q., Higginbotham, J., Beckler, M., Weaver, A., Vickers, K., Prasad, N., Levy, S., Zhang, B., Coffey, R., y Patton, G. (2015). Kras-dependent sorting of mirna to exosomes. *eLife*, **4**(e07197).

Connerty, P., Ahadi, A., y Hutvagner, G. (2015). Rna binding proteins in the mirna pathway. *International Journal of Molecular Sciences*, **17**(1).

Deb, K., Pratap, A., Agarwal, S., y Meyarivan, T. (2002). A fastf and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, **6**(2): 182–197.

Fawcett, T. (2006). An introduction to roc analysis, pattern recognition letters. *Pattern Recognition Letters*, **27**(8): 861–874.

Francois, J. (2016). What Is Machine Learning? `https://www.ibm.com/developerworks/community/blogs/jfp/entry/What_Is_Machine_Learning?lang=en`. [Consultado el 2 de junio de 2018].

Guyer, C. (2016). Feature Selection (Data Mining). `https://www.biology-online.org/dictionary/index.php?title=Protein`. [Consultado el 2 de junio de 2018].

Ha, D., Yang, N., y Nadithe, V. (2016). Exosomes as therapeutic drug carriers and delivery vehicles across biological membranes: current perspectives and future challenges. *Acta Pharmaceutica Sinica B*, **6**(4): 287–296.

Hall, M., Geoffrey, E. F., Pfahringer, H. B., Reutemann, P., y Witten, I. H. (2009). The weka data mining software: an update. *ACM SIGKDD Explorations Newsletter*, **11**(1): 10–18.

Hornick, N., Huan, J., Doron, B., Lapidus, N. G. J., Chang, B., y Kurre, P. (2015). Serum exosome microrna as a minimally-invasive early biomarker of aml. *Science Reports*, **5**(11295).

Jahandideh, S. y Srinivasasainagendra, V. (2012). Comprehensive comparative analysis and identification of rna-binding protein domains: Multi-class classification and feature selection. *Journal of theoretical biology*, **312**: 65–75.

Kohavi, R. y John, G. (1997). Wrappers for feature subset selection. *Artificial Intelligence*, **97**(1-2): 273–324.

Koppers, D., M Hackenberg, I Bijnsdorp, M. v., Sadek, P., Sie, D., Zini, N., Middeldorp, J., Ylstra, B., deMenezes, R., Wurdinger, T., Meijer, G., y Pegtel, D. (2014). Nontemplated nucleotide additions distinguish the small rna composition in cells from exosomes. *Cell Reports*, **8**(6): 649–1658.

Kuhn, M. y Johnson, K. (2013). *Applied Predictive Modeling*.

Kyte, J. y Doolittle, R. (1982). A simple method for displaying the hydropathic character of a protein. *Journal of Molecular Biology*, **157**(1): 105–132.

Li, W. y Godzik, A. (2006). Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics (Oxford, Journal)*, **22**(13): 1658–1659.

Lin, J., Li, J., Huang, B., Liu, J., Chen, X., Chen, X., Xu, Y., Huang, L., Wang, X., Lin, J., Li, J., Huang, B., Liu, J., Chen, X., Chen, M., Xu, Y., Huang, L., y Wang, X. (2015). Exosomes: Novel biomarkers for clinical diagnosis. *The Scientific World Journal*, **2015**(657086).

Liu, C., Che, D., Liu, X., y Song, Y. (2013). Applications of machine learning in genomics and systems biology. **2013**.

Ma, X., Guo, J., Wu, J., Liu, H., Yu, J., Xie, J., y Sun, X. (2011). Prediction of rna-binding residues in proteins from primary sequence using an enriched random forest model with a novel hybrid feature. *Proteins*, **79**(4): 1230–1239.

Markowetz, F. (2017). All biology is computational biology. **15**(3).

McKenzie, A., Hoshino, D., N Hong, D Cha, J. F. J. C., Patton, J., y Weaver, A. (2016). Kras-mek signaling controls ago2 sorting into exosomes. *Cell Reports*, **15**(5): 978–987.

Mestrovic, T. (2015). MicroRNA Biogenesis. `https://www.news-medical.net/life-sciences/MicroRNA-Biogenesis.aspx`. [Consultado el 2 de junio de 2018].

Mozo, V. (2008a). Protein. `https://www.biology-online.org/dictionary/index.php?title=Protein`. [Consultado el 2 de junio de 2018].

Mozo, V. (2008b). Protein. `https://www.biology-online.org/dictionary/index.php?title=Protein`. [Consultado el 2 de junio de 2018].

Mukaka, M. (2012). Statistics corner: A guide to appropriate use of correlation coefficient in medical research. *Malawi Medical Journal: the journal of Medical Association of Malawi*, **24**(3): 69–71.

Ohno, S., Takanashi, M., Sudo, K., Ueda, S., Ishikawa, A., Matsuyama, N., Fujita, K., Mizutani, T., Ohgi, T., Ochiya, T., Gotoh, N., y Kuroda, M. (2012). Systemically injected exosomes targeted to egfr deliver antitumor microrna to breast cancer cells. *Molecular Therapy: The journal of the American Society of Gene Therapy*, **21**(1).

Paul, S. y Das, S. (2015). Simultaneous feature selection and weighting, an evolutionary multi-objective optimization approach. *Journal Pattern Recognition Letters*, **65**(C): 51–59.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., y Duchesnay, E. (2011). Scikit-learn: Machine learning in python. *The Journal of Machine Learning Research*, **12**(2/1/2011): 2825–2830.

Properzi, F., Logozzi, M., y Fais, S. (2013). Exosomes: the future of biomarkers in medicine. *Biomarkers in medicine*, **7**(5): 769–778.

Przybilla, D. (2013). Label Propagation. `https://wwwslidesharenet/dav009/label-propagation-semisupervised-learning-with-applications-to-nlp`. [Consultado el 2 de junio de 2018].

Ribeiro, M., Zhu, H., Millard, R., y Fan, G. (2013). Exosomes function in pro- and anti-angiogenesis. *PubMed Central*, **2**(1): 54–59.

Ruiz-Blanco, Y., Paz, W., Green, J., y Marrero-Ponce, Y. (2015). Protdcal: A program to compute general-purpose-numerical descriptors for sequences and 3d-structures of proteins. *BMC Bioinformatics*, **6**(162).

SA, C., DG, P., KA, P., y AG, P. (2007a). Hydrogen bonds in protein-dna complexes: where geometry meets plasticity. *Biochimie*, **89**(11): 1291–1303.

SA, C., DG, P., KA, P., y AG, P. (2007b). Hydrogen bonds in protein-dna complexes: where geometry meets plasticity. *Biochimie*, **89**(11): 1291–1303.

Sancho, F. y García, J. C. (2016). NetLogo: A Modeling Tool / Una herramienta de modelado. `http://www.cs.us.es/~fsancho/?e=128`. [Consultado el 2 de junio de 2018].

Shen, J., Zhang, J., Luo, X., Zhu, W., Yu, K., Chen, K., Li, Y., y Jiang, H. (2007). Predicting protein–protein interactions based only on sequences information. *Proceedings of the National Academy of Sciences of the United States of America*, **104**(11): 4337–4341.

Sheng, J. y Zhou, Z.-H. (2013). Sequence-based prediction of microrna-binding residues in proteins using cost-sensitive laplacian support vector machines. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, **10**(3).

Silverman, J., Clos, J., deOliveira, C., Shirvani, O., Fang, Y., Wang, C., Foster, L., y Reiner, N. (2010). An exosome-based secretion pathway is responsible for protein export from leishmania and communication with macrophages. *Journal of Cell Science*, **123**(6): 842–852.

Squadrito, M., Baer, C., Burdet, F., C Maderna, G. G., Lyle, R., Ibberson, M., y DePalma, M. (2014). Endogenous rnas modulate microrna sorting to exosomes and transfer to acceptor cells. *Cell Reports*, **8**(5): 1432–1446.

Tian, D. (2016). A multi-objective genetic local search algorithm for optimal feature subset selectio. *2016 International Conference on Computational Science and Computational Intelligence (CSCI)*, pp. 1089–1094.

Trott, O. y Olson, A. J. (2010). Autodock vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization and multithreading. *Journal of Computational Chemistry*, **31**(2): 455–461.

Valadi, H., Ekstrom, K., Bossios, A., Sjostrand, M., Lee, J., y Lotvall, J. (2007). Exosome-mediated transfer of mrnas and micrornas is a novel mechanism of genetic exchange between cells. *Nature Cell Biology*, **9**: 654–659.

Villarroya-Beltri, C., Gutiérrez-Vázquez, C., Sánchez-Cabo, F., Pérez-Hernández, Vázquez, J., Martin-Cofreces, N., Martinez-Herrera, D., Pascual-Montano, A., Mittelbrunn, M., y Sánchez-Madrid, F. (2013). Sumoylated hnrnpa2b1 controls the sorting of mirnas into exosomes through binding to specific motifs. *Nature Communications*, **4**(2980).

Wang, L. y Brown, S. (2006). Bindn: A web-based tool for efficient prediction of dna and rna binding sites in amino acid sequences. *Nucleic acids research*, **34**.

Wu, J., Liu, H., Duan, X., Ding, Y., Wu, H., Bai, Y., y Sun, X. (2009). Prediction of dna-binding residues in proteins from amino acid sequences using a random forest model with a hybrid feature. *Bioinformatics (Oxford, England)*, **25**(1): 30–35.

Zhang, J., Li, S., Li, L., Li, M., Guo, C., Yao, J., y Mi, S. (2015). Exosome and exosomal microrna: Trafficking, sorting, and function. *Genomics, Proteomics & Bioinformatics*, **13**(1): 17–24.

Zhang, Q. y Li, H. (2007). Moea/d: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation*, **11**(6).

Apéndice

.1. Secuencias de entrenamiento

La secuencias de entrenamiento son públicas y pueden ser encontradas en `https://drive.google.com/drive/folders/0B5u85HD5zG7OeUlMMVl3Y1IwTU0?usp=sharing`

.2. Valores para el cálculo de las características descritas en el trabajo de Sheng y Zhou (2013)

La referencia para los valores de la escala Kyte y Doolittle puede ser encontrada en Paul y Das (2015).

Residuo	Kyte-Doolittle	Valor pkA (cadena lateral)	Peso molecular (g/mol)
ALA	1.8	0	89.1
ARG	-4.5	12.48	174.2
ASN	-3.5	0	132.1
ASP	-3.5	3.65	133.1
CYS	2.5	0	121.2
GLN	-3.5	0	146.2
GLU	-3.5	4.25	147.1
GLY	-0.4	0	75.1
HIS	-3.2	6	155.2
ILE	4.5	0	131.2
LEU	3.8	0	132.2
LYS	-3.9	10.53	146.2
MET	1.9	0	149.2
PHE	2.8	0	165.2
PRO	-1.6	0	115.1
SER	-0.8	0	105.1
THR	-0.7	0	119.1
TRP	-0.9	0	204.2
TYR	-1.3	0	181.2
VAL	4.2	0	117.1

.3. Agrupamiento de ejemplos positivos

>Cluster 0

0 11aa, >seq1... *

1 11aa, >seq2... at 90.91%

2 11aa, >seq3... at 81.82%

3 11aa, >seq4... at 72.73%

4 11aa, >seq5... at 63.64%

5 11aa, >seq6... at 54.55%

6 11aa, >seq7... at 45.45%

>Cluster 1

0 11aa, >seq9... *

1 11aa, >seq10... at 81.82%

2 11aa, >seq11... at 63.64%

3 11aa, >seq12... at 54.55%

4 11aa, >seq13... at 45.45%

5 11aa, >seq54... at 45.45%

>Cluster 2

0 11aa, >seq18... *

1 11aa, >seq19... at 81.82%

2 11aa, >seq20... at 63.64%

3 11aa, >seq21... at 54.55%

4 11aa, >seq36... at 45.45%

5 11aa, >seq37... at 45.45%

>Cluster 3

0 11aa, >seq69... *

1 11aa, >seq70... at 81.82%

2 11aa, >seq71... at 72.73%

3 11aa, >seq72... at 63.64%

4 11aa, >seq73... at 54.55%

5 11aa, >seq74... at 45.45%

>Cluster 4

0 11aa, >seq75... *

1 11aa, >seq76... at 90.91%

2 11aa, >seq77... at 81.82%

3 11aa, >seq78... at 72.73%

4 11aa, >seq79... at 63.64%

5 11aa, >seq80... at 45.45%

>Cluster 5

0 11aa, >seq84... *

1 11aa, >seq85... at 90.91%

2 11aa, >seq86... at 81.82%

3 11aa, >seq87... at 72.73%

4 11aa, >seq88... at 63.64%

5 11aa, >seq89... at 45.45%

>Cluster 6

0 11aa, >seq62... *

1 11aa, >seq63... at 81.82%

2 11aa, >seq64... at 63.64%

3 11aa, >seq65... at 54.55%

4 11aa, >seq66... at 45.45%

>Cluster 7

0 11aa, >seq25... *

1 11aa, >seq26... at 81.82%

2 11aa, >seq27... at 72.73%

3 11aa, >seq28... at 54.55%

>Cluster 8

0 11aa, >seq90... *

1 11aa, >seq91... at 90.91%

2 11aa, >seq92... at 81.82%

3 11aa, >seq93... at 54.55%

>Cluster 9

0 11aa, >seq101... *

1 11aa, >seq102... at 90.91%

2 11aa, >seq103... at 81.82%

3 11aa, >seq104... at 72.73%

>Cluster 10

0 11aa, >seq115... *

1 11aa, >seq116... at 90.91%

2 11aa, >seq117... at 72.73%

3 11aa, >seq118... at 45.45%

>Cluster 11

0 11aa, >seq130... *

1 11aa, >seq131... at 81.82%

2 11aa, >seq132... at 72.73%

3 11aa, >seq133... at 54.55%

>Cluster 12

0 11aa, >seq30... *

1 11aa, >seq31... at 90.91%

2 11aa, >seq32... at 54.55%

>Cluster 13

0 11aa, >seq33... *

1 11aa, >seq34... at 90.91%

2 11aa, >seq35... at 81.82%

>Cluster 14

0 11aa, >seq39... *

1 11aa, >seq40... at 90.91%

2 11aa, >seq41... at 54.55%

>Cluster 15

0 11aa, >seq55... *

1 11aa, >seq56... at 72.73%

2 11aa, >seq57... at 63.64%

>Cluster 16

0 11aa, >seq81... *

1 11aa, >seq82... at 54.55%

2 11aa, >seq83... at 45.45%

>Cluster 17

0 11aa, >seq94... *

1 11aa, >seq95... at 90.91%

2 11aa, >seq96... at 81.82%

>Cluster 18

0 11aa, >seq98... *

1 11aa, >seq99... at 63.64%

2 11aa, >seq100... at 54.55%

>Cluster 19

0 11aa, >seq105... * 1 11aa, >seq106... at 81.82 %

2 11aa, >seq107... at 54.55 %

>Cluster 20

0 11aa, >seq108... *

1 11aa, >seq109... at 90.91 %

2 11aa, >seq110... at 63.64 %

>Cluster 21

0 11aa, >seq111... *

1 11aa, >seq112... at 81.82 %

2 11aa, >seq113... at 54.55 %

>Cluster 22

0 11aa, >seq123... *

1 11aa, >seq124... at 90.91 %

2 11aa, >seq125... at 54.55 %

>Cluster 23

0 11aa, >seq126... *

1 11aa, >seq127... at 63.64 %

2 11aa, >seq128... at 45.45 %

>Cluster 24

0 11aa, >seq14... *

1 11aa, >seq15... at 45.45 %

>Cluster 25

0 11aa, >seq16... *

1 11aa, >seq17... at 63.64 %

>Cluster 26

0 11aa, >seq23... *

1 11aa, >seq24... at 90.91 %

>Cluster 27

0 11aa, >seq44... *

1 11aa, >seq45... at 63.64 %

>Cluster 28

0 11aa, >seq46... *

1 11aa, >seq47... at 90.91%

>Cluster 29

0 11aa, >seq50... *

1 11aa, >seq51... at 54.55%

>Cluster 30

0 11aa, >seq52... * 1 11aa, >seq53... at 90.91% >Cluster 31

0 11aa, >seq58... *

1 11aa, >seq59... at 63.64%

>Cluster 32

0 11aa, >seq60... *

1 11aa, >seq61... at 81.82%

>Cluster 33

0 11aa, >seq67... *

1 11aa, >seq68... at 90.91%

>Cluster 34

0 11aa, >seq119... *

1 11aa, >seq120... at 63.64%

>Cluster 35

0 11aa, >seq134... *

1 11aa, >seq135... at 45.45%

>Cluster 36

0 11aa, >seq8... *

>Cluster 37

0 11aa, >seq22... *

>Cluster 38

0 11aa, >seq29... *

>Cluster 39

0 11aa, >seq38... *

>Cluster 40

0 11aa, >seq42... *

>Cluster 41

0 11aa, >seq43... *

>Cluster 42

0 11aa, >seq48... *

>Cluster 43

0 11aa, >seq49... *

>Cluster 44

0 11aa, >seq97... *

>Cluster 45

0 11aa, >seq114... *

>Cluster 46

0 11aa, >seq121... *

>Cluster 47

0 11aa, >seq122... *

>Cluster 48

0 11aa, >seq129... *

>Cluster 49

0 11aa, >seq136... *

.4. Motivos obtenidos de MEME

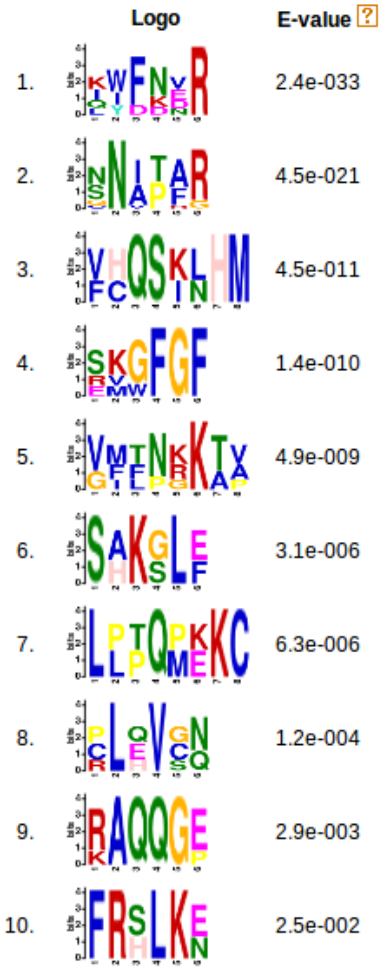

Figura 14. Motivos de los ejemplos positivos del conjunto extendido.

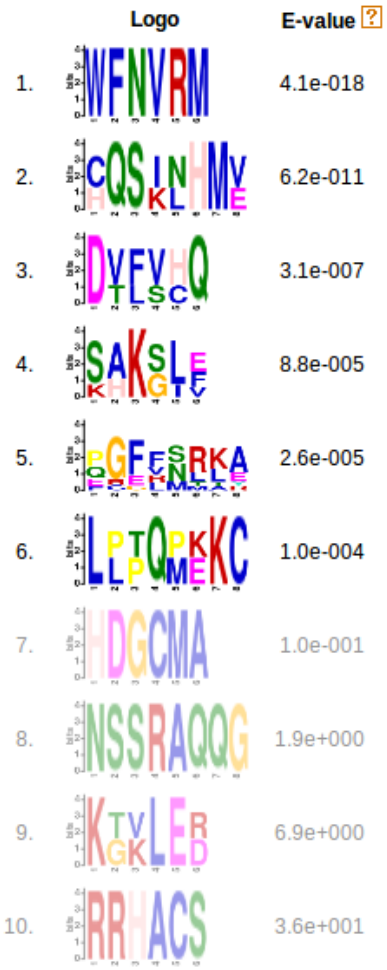

Figura 15. Motivos de los ejemplos positivos del conjunto original.

Figura 16. Motivos de los ejemplos positivos de los ejemplos adicionales del conjunto extendido.

.5. Replicación del trabajo de Sheng y Zhou (2013) (validación cruzada de tres y diez pliegues para el conjunto de prueba (aprendizaje supervisado))

Tabla 22. Replicación del trabajo de Sheng y Zhou (2013) (validación cruzada de tres pliegues para el conjunto de prueba (aprendizaje supervisado)).

	PQ	NQ	TP	FP	TN	FN	Val-F	Sen	Pre	MCC	AUC	Esp	No.C
S(D1,C1)	20.33	432.67	20.33	18.67	414.00	0.00	0.67	1.00	0.51	0.70	0.98	0.95	304
SS(D1,D2,C1,C2)	45.33	407.67	45.00	41.67	366.00	0.33	0.68	0.99	0.53	0.68	0.95	0.90	334

Tabla 23. Replicación del trabajo de Sheng y Zhou (2013) (validación cruzada de diez pliegues para el conjunto de prueba (aprendizaje supervisado)).

	PQ	NQ	TP	FP	TN	FN	Val-F	Sen	Pre	MCC	AUC	Esp	No.C
S(D1,C1)	6.10	129.80	6.10	47.90	81.90	0.00	0.20	1.00	0.12	0.27	0.82	0.63	304
SS(D1,D2,C1,C2)	13.60	122.30	13.60	103.40	18.90	0.00	0.21	1.00	0.12	0.13	0.58	0.15	334

.6. Frecuencias relativas de los 20 aminoácidos escenciales por conjuntos de datos

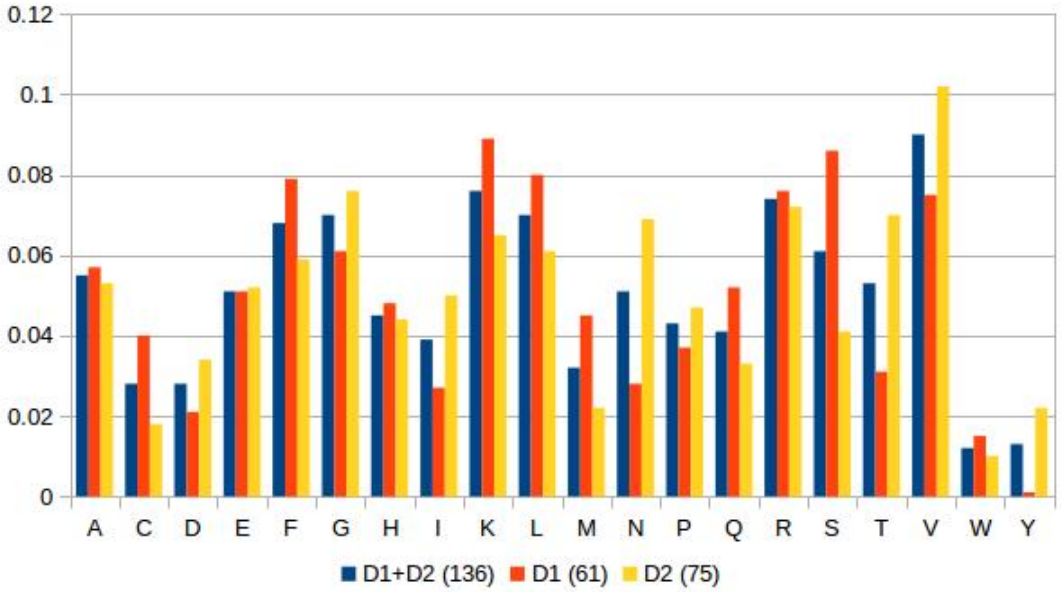

Figura 17. Frecuencias relativas de los 20 aminoácidos escenciales por conjuntos de datos.