Centro de Ciencias Básicas

Departamento de Ciencias de la Computación

Tesis

Búsqueda de motivos altamente conservados mediante una metaheurística multiobjetivo

Presenta

Jesús Alberto Correa Morales

Para obtener el grado de Maestro en Ciencias con Opción a la Computación

Tutores

Dr. Rogelio Salinas Gutiérrez

Dra. Eunice Esther Ponce de León Sentí

Integrante del comité tutoral

Dr. Julio César Ponce Gallegos

Aguascalientes, Ags, 12 de Abril del 2023

UNIVERSIDAD AUTÓNOMA DE AGUASCALIENTES

CARTA DE VOTO APROBATORIO INDIVIDUAL

MTRO. EN C. JORGE MARTÍN ALFÉREZ CHÁVEZ
DECANO DEL CENTRO DE CIENCIAS BÁSICAS
P R E S E N T E

Por medio del presente como **COTUTOR** designado del estudiante **JESÚS ALBERTO CORREA MORALES** con ID **205239** quien realizó **la tesis** titulada: **BÚSQUEDA DE MOTIVOS ALTAMENTE CONSERVADOS MEDIANTE UNA METAHEURÍSTICA MULTIOBJETIVO**, un trabajo propio, innovador, relevante e inédito y con fundamento en el Artículo 175, Apartado II del Reglamento General de Docencia doy mi consentimiento de que la versión final del documento ha sido revisada y las correcciones se han incorporado apropiadamente, por lo que me permito emitir el **VOTO APROBATORIO**, para que **él** pueda proceder a imprimirla así como continuar con el procedimiento administrativo para la obtención del grado.

Pongo lo anterior a su digna consideración y sin otro particular por el momento, me permito enviarle un cordial saludo.

A T E N T A M E N T E
"Se Lumen Proferre"
Aguascalientes, Ags., a 22 de mayo de 2023.

Dr. Rogelio Salinas Gutiérrez
Cotutor de tesis

c.c.p.- Interesado
c.c.p.- Secretaría Técnica del Programa de Posgrado

CARTA DE VOTO APROBATORIO
INDIVIDUAL

Mtro. En C. Jorge Martín Alférez Chávez
DECANO (A) DEL CENTRO DE CIENCIAS BÁSICAS

P R E S E N T E

Por medio del presente como **COTUTORA** designado del estudiante **JESÚS ALBERTO CORREA MORALES** con ID **205239** quien realizó **la tesis** titulada: **BÚSQUEDA DE MOTIVOS ALTAMENTE CONSERVADOS MEDIANTE UNA METAHEURÍSTICA MULTIOBJETIVO**, un trabajo propio, innovador, relevante e inédito y con fundamento en el Artículo 175, Apartado II del Reglamento General de Docencia doy mi consentimiento de que la versión final del documento ha sido revisada y las correcciones se han incorporado apropiadamente, por lo que me permito emitir el **VOTO APROBATORIO**, para que **él** pueda proceder a imprimirla así como continuar con el procedimiento administrativo para la obtención del grado.

Pongo lo anterior a su digna consideración y sin otro particular por el momento, me permito enviarle un cordial saludo.

A T E N T A M E N T E
"Se Lumen Proferre"
Aguascalientes, Ags., a 15 de MAYO de 2023.

Dra. Eunice Esther Ponce de León Sentí
Cotutora de *tesis*

c.c.p.- Interesado
c.c.p.- Secretaría Técnica del Programa de Posgrado

CARTA DE VOTO APROBATORIO
INDIVIDUAL

Mtro. En C. Jorge Martín Alférez Chávez
DECANO (A) DEL CENTRO DE CIENCIAS BÁSICAS

P R E S E N T E

Por medio del presente como **ASESOR** designado del estudiante **JESÚS ALBERTO CORREA MORALES** con ID **205239** quien realizó **la tesis** titulada: **BÚSQUEDA DE MOTIVOS ALTAMENTE CONSERVADOS MEDIANTE UNA METAHEURÍSTICA MULTIOBJETIVO**, un trabajo propio, innovador, relevante e inédito y con fundamento en el Artículo 175, Apartado II del Reglamento General de Docencia doy mi consentimiento de que la versión final del documento ha sido revisada y las correcciones se han incorporado apropiadamente, por lo que me permito emitir el **VOTO APROBATORIO**, para que **él** pueda proceder a imprimirla así como continuar con el procedimiento administrativo para la obtención del grado.

Pongo lo anterior a su digna consideración y sin otro particular por el momento, me permito enviarle un cordial saludo.

A T E N T A M E N T E
"Se Lumen Proferre"
Aguascalientes, Ags., a 17 de MAYO de 2023.

Dr. Julio César Ponce Gallegos
Asesor de *tesis*

c.c.p.- Interesado
c.c.p.- Secretaría Técnica del Programa de Posgrado

DICTAMEN DE LIBERACIÓN ACADEMICA PARA INICIAR LOS TRAMITES DEL EXAMEN DE GRADO

Fecha de dictaminación dd/mm/aaaa: 02/06/2023

NOMBRE: Jesús Alberto Correa Morales **ID** 205239

PROGRAMA: Maestría en Ciencias con opciones a la Computación, Matemáticas Aplicadas **LGAC (del posgrado):** Computación-Inteligencia Artificial

TIPO DE TRABAJO: (X) Tesis ( ) Trabajo Práctico

TITULO: Búsqueda de motivos altamente conservados mediente una metaheurística multiobjetivo

IMPACTO SOCIAL (señalar el impacto logrado): Se generó una herrramienta para el apoyo en la investigación de secuencias de proteínas, con la cual, a partir de una secuencia representativa de aminoácidos se puede generar un conjunto de secuencias artificiales para el apoyo de la caracterización de proteínas.

INDICAR SI NO N.A. (NO APLICA) SEGÚN CORRESPONDA:

	Elementos para la revisión académica del trabajo de tesis o trabajo práctico:
SI	El trabajo es congruente con las LGAC del programa de posgrado
SI	La problemática fue abordada desde un enfoque multidisciplinario
SI	Existe coherencia, continuidad y orden lógico del tema central con cada apartado
SI	Los resultados del trabajo dan respuesta a las preguntas de investigación o a la problemática que aborda
SI	Los resultados presentados en el trabajo son de gran relevancia científica, tecnologíca o profesional según el área
SI	El trabajo demuestra más de una aportación original al conocimiento de su área
NO	Las aportaciones responden a los problemas prioritarios del país
SI	Generó transferecia del conocimiento o tecnológica
SI	Cumple con la ética para la investigación (reporte de la herramienta antiplagio)
	El egresado cumple con lo siguiente:
SI	Cumple con lo señalado por el Reglamento General de Docencia
SI	Cumple con los requisitos señalados en el plan de estudios (créditos curriculares, optativos, actividades complementarias,estancia, predoctoral, etc)
SI	Cuenta con los votos aprobatorios del comité tutoral, en caso de los posgrados profesionales si tiene solo tutorpodrá liberar solo el tutor
N.A.	Cuenta con la carta de satisfacción del Usuario
SI	Coincide con el título y objetivo registrado
SI	Tiene congruencia con cuerpos académicos
SI	Tiene el CVU del Conacyt actualizado
N.A.	Tiene el artículo aceptado o publicado y cumple con los requisitos institucionales (en caso que proceda)
	En caso de Tesis por artículos científicos publicados
N.A.	Aceptación o Publicación de los articulos según el nivel del programa
N.A.	El estudiante es el primer autor
N.A.	El autor de correspondencia es el Tutor del Núcleo Académico Básico
N.A.	En los artículos se ven reflejados los objetivos de la tesis, ya que son producto de este trabajo de investigación.
N.A.	Los artículos integran los capítulos de la tesis y se presentan en el idioma en que fueron publicados
N.A.	La aceptación o publicación de los artículos en revistas indexadas de alto impacto

Con base a estos criterios, se autoriza se continúen con los trámites de titulación y programación del examen de grado:

Sí X

No

FIRMAS

Elaboró:

* NOMBRE Y FIRMA DEL CONSEJERO SEGÚN LA LGAC DE ADSCRIPCION: DR. HERMILO SÁNCHEZ CRUZ

NOMBRE Y FIRMA DEL SECRETARIO TÉCNICO: DR. HERMILO SÁNCHEZ CRUZ

*** En caso de conflicto de intereses, firmará un revisor miembro del NAB de la LGAC correspondiente distinto al tutor o miembro del comité tutoral, asignado por el Decano**

Revisó:

NOMBRE Y FIRMA DEL SECRETARIO DE INVESTIGACIÓN Y POSGRADO: DR. JUAN JÁUREGUI RINCÓN

Autorizó:

NOMBRE Y FIRMA DEL DECANO: M. EN C. JORGE MARTÍN ALFÉREZ CHÁVEZ

Nota: procede el trámite para el Depto. de Apoyo al Posgrado

En cumplimiento con el Art. 105C del Reglamento General de Docencia que a la letra señala entre las funciones del Consejo Académico: .... Cuidar la eficiencia terminal del programa de posgrado y el Art. 105F las funciones del Secretario Técnico, llevar el seguimiento de los alumnos.

Agradecimientos

Agradezco a CONACYT por el apoyo económico que me brindo durante el estudio del posgrado. Agradezco a la Universidad Autónoma de Aguascalientes (UAA) por darme la oportunidad de estudiar mi posgrado y recibirme con las puertas abiertas para consolidarme como una mejor persona tanto en el ámbito académico como en lo personal. Agradezco a los profesores que tuve la fortuna de conocer y la oportunidad de adquirir sus conocimientos. Agradezco a la Dra. Aurora Torres por escucharme y aconsejarme durante el posgrado, al Dr. Francisco Álvarez por diseminar todas mis dudas durante el posgrado y ayudarme a ser una mejor persona, a todo el personal administrativo que colaboró para mi admisión, estancia y conclusión del posgrado. Agradezco al C. a Dr. Mauricio Martín por el apoyo y orientación en el área de la bioinformática, al Dr. Rogelio Salinas por el apoyo incondicional que me brindo a lo largo del posgrado y siempre mantuvo una confianza incondicional en mí. Agradezco a mis compañeros que tuve el privilegio de compartir salón de clases, siempre los recordaré y conmemoraré los buenos momentos que pasamos, en especial al Ing. Mario Rosales por el apoyo incondicional que me brindo siempre. Quiero agradecer a la Dra. Eunice Esther Ponce de León Sentí que me apoyo en cada etapa del posgrado y fue la persona que me alentó para continuar con mi preparación, recordaré con un gran cariño todos los momentos que vivimos. Agradezco a mi mamá Ma. Dolores Morales Rangel, mi papá Efrén Correa Medina y a mis hermanos Sergio Efrén Correa Morales y Rafael Correa Morales que me apoyaron, me motivaron, me guiaron y me ayudaron a continuar con mis estudios a pesar de todas las adversidades. Agradezco a mis amigos Jorge Campos, Ángel Salmones y Emmanuel Avalos y a mi novia Rosa Hernández por siempre creer en mí y motivarme para continuar con mis objetivos. Agradezco a toda persona que no mencione anteriormente, pero tuve la fortuna de compartir momentos especiales.

Dedicatorias

A mi madre Ma. Dolores Morales Rangel que es la persona que me apoyo, ve por mi bien y me mostró que siempre hay manera de hacer las cosas.

A mi padre Efrén Correa Medina que es mi ejemplo para seguir, me mostró lo que es el trabajo duro y valorar las cosas que tengo.

A mi hermano Sergio Efrén Correa Morales que me mostró que los problemas que tenemos en realidad dependen de la perspectiva con la que los afrontemos.

A mi hermano Rafael Correa Morales que a pesar de todo las cosas que hago y digo no deja de ver por mi bien y procura que cada día que pasa sea una mejor persona.

A mis amigos que se convirtieron en mis hermanos que me han ayudado a caminar cuando yo creía que no podía seguir.

A mis amigos con los que he compartido buenos y malos momentos.

A mis profesores que son parte de mi formación personal y profesional.

A todas las personas que son parte de mi pasado, pero hoy ya no están con nosotros.

Índice General

Índice de Tablas

Índice de Figuras

Índice de Algoritmos y Pseudocódigos

Acrónimos

ACO Algoritmo de Colonia de Hormigas. 7, 44, 45

ADN Ácido desoxirribonucleico. 31

AEs Algoritmos Evolutivos. 7, 47, 49, 50

AMS Alineamiento Múltiple de Secuencias. 5, 31–34, 69–73, 76, 122

ARN Ácido ribonucleico. 31

BLAST Basic Local Alignment Search Tool. 29

BMDA Algoritmo con Distribución Marginal Bivariada. 52

BOA Algoritmo de Optimización Bayesiano. 53, 54

C(M) Matriz Consenso. 4, 32, 33

cGA Algoritmo Genético Compacto. 51

COMIT Optimización Combinatoria con Árboles de Información Mutua. 52

EcGA Algoritmo Genético Compacto Extendido. 53, 54

EDA Algoritmo de Estimación de la Distribución. 5, 18, 48–51, 55, 58–60, 66, 92, 94, 106, 107

EE Estrategias Evolutivas. 48

EHO Algoritmo de Pastoreo de Elefantes. 7, 45, 46

FCC Función de Compatibilidad por Carga. 65

FCH Función de Compatibilidad por Hidropaticidad. 65

FCP Función de Compatibilidad por Peso. 65

FDA Algoritmo de Distribución Factorizada. 53, 54

GA Algoritmo Genético. 5, 48, 50

GWO Algoritmo del Lobo Gris. 7, 46, 47

HM Momento de Hidropacidad. 78

Ic Índice de Conservación. 33, 34

IDE Entorno de Desarrollo Integrado. 66, 69, 73, 75, 79, 86

MCC Matriz de Compatibilidad por Carga. 4, 65, 67, 77, 78, 105

MCH Matriz de Compatibilidad por Hidropaticidad. 4, 65, 67, 77–79, 105

MCP Matriz de Compatibilidad por Peso. 4, 65, 67, 77–79, 105

MIMIC Algoritmos de Maximización de la Información para Clasificación. 52

MW Peso Molecular. 78

PBIL Aprendizaje Incremental Basado en Poblaciones. 51

PDM Problema de Descubrimiento de Motivos. 31, 79, 107

PE Programación Evolutiva. 48

pI Punto Isoeléctrico. 78

S(C(M)) Subsecuencias Conservadas. 4, 32, 34

Resumen

Vivimos una era de información, donde al paso del tiempo se genera grandes cantidades de información. Los investigadores descubren nueva información y es compartida en el menor tiempo posible. Un área que trabaja con información nueva es la bioinformática, que se encarga de la aplicación de métodos computacionales para el análisis de datos biológicos. El análisis de las proteínas es un problema complejo, las proteínas están formadas por una cantidad indeterminada de aminoácidos. Existen 20 aminoácidos estándar y no se tiene ninguna regla sobre la forma que los aminoácidos conforman las proteínas. La búsqueda e identificación de motivos tridimensionales es una tarea sumamente importante, ya que gracias a la localización de los motivos se pueden caracterizar subsecuencias de aminoácidos de una proteína y relacionarlos con una estructura y funcionalidad determinada. Por tal razón la búsqueda de motivos es una tarea importante, pero compleja porque las posibles combinaciones de aminoácidos que conforman una proteína dan origen a un problema combinatorio, ya que entre más grande es la secuencia para buscar el número de posibles combinaciones se incrementa exponencialmente. Por tal razón se implementa una metodología en la cual a partir de un conjunto de secuencias de proteínas de la familia *coronaviridae*, se obtiene un conjunto de secuencias con los mejores aciertos bidireccionales, a este conjunto se obtiene un subconjunto de subsecuencias conservadas. Por medio del algoritmo de estimación de la probabilidad se busca un conjunto de secuencias altamente compatibles por las propiedades fisicoquímicas para cada subsecuencia conservada. Esta metodología es aplicable a cualquier conjunto de proteínas. Los resultados obtenidos es un conjunto de subsecuencias conservadas y un conjunto de secuencia altamente compatible. Con este trabajo se deja una nueva metodología para la generación de secuencias altamente compatibles que se esperan que sean utilizadas para la caracterización de subsecuencias de proteínas.

Abstract

We live in an information age, where large amounts of information are generated over time. Researchers discover new information and it is shared in the shortest possible time. One area that works with new information is bioinformatics, which deals with the application of computational methods for the analysis of biological data. The analysis of proteins is a complex problem, proteins are made up of an indeterminate amount of amino acids. There are 20 standard amino acids and there are no rules for how amino acids make up proteins. The search and identification of three-dimensional motifs is an extremely important task, since thanks to the location of the motifs, amino acid subsequences of a protein can be characterized and related to a specific structure and functionality. For this reason, the search for motifs is an important, but complex task, because the possible combinations of amino acids that make up a protein give rise to a combinatorial problem, since the larger the sequence to search for, the number of possible combinations increases exponentially. For this reason, a methodology is implemented in which from a set of protein sequences of the family *coronaviridae*, a set of sequences with the best bidirectional hits is obtained, from this set a subset of conserved subsequences is obtained. By means of the probability estimation algorithm, a set of sequences highly compatible by physicochemical properties is searched for each conserved subsequence. This methodology is applicable to any set of proteins. The results obtained is a set of conserved subsequences and a highly compatible sequence set. With this work, a new methodology is left for the generation of highly compatible sequences that are expected to be used for the characterization of protein subsequences.

Capítulo 1

Introducción

En el presente capítulo se introducen los antecedentes de partida del trabajo de investigación, continuando con el planteamiento del problema de investigación, la justificación, así como el objetivo general de la investigación, los objetivos específicos, por último, pero igual de importante, las preguntas de investigación.

1.1. Antecedentes

El trabajo de investigación tiene un inicio en una metodología para la obtención de árboles filogenéticos mediante los mejores aciertos bidireccionales de un conjunto de organismos desarrollado en el trabajo de [Ponce de León Sentí et al., 2017]. La misma metodología se extendió para encontrar familias de proteínas basadas en la cantidad de mejores aciertos bidireccionales en el trabajo de tesina de [Gallegos, 2019] así como en [Ponce de León Sentí et al., 2021]. En la tesina de [Correa Morales, 2020] se desarrolló una metodología para el descubrimiento de motivos en familias de proteínas no reportadas en la literatura con el objetivo de profundizar en el estudio de las características y propiedades de los aminoácidos que conforman estas familias identificando motivos y dominios de interés. En el trabajo de [Galvis Motoa et al., 2021] se automatizaron varios procesos de adquisición de datos ómicos de la metodología reportada en [Ponce de León Sentí et al., 2017], con lo que se realizó el preprocesamiento de 68 virus de la familia *coronaviridae* así como la obtención del árbol filogenético a partir de una distancia basada en el conteo de los mejores aciertos bidireccionales que comparten los

organismos 1 a 1. Con el antecedente de diseño e implementación de las herramientas antes descritas la presente tesis tiene como objetivo estudiar desde el punto de vista multiobjetivo la búsqueda de motivos en proteínas en organismos de la familia *coronaviridae*. Para ello se cuenta con grupos de proteínas de virus de la familia *coronaviridae*, las cuales son de interés para caracterizar la función que estas proteínas realizan. Dichas proteínas se obtuvieron aplicando la metodología propuesta en la tesis de [Motoa, 2022].

La búsqueda de motivos se realiza tanto en secuencias de ADN como en secuencias de aminoácidos como podemos observar en el trabajo de [Álvarez and Rodríguez, 2013] que aborda el paradigma de multiobjetivo, pero para secuencias de ADN, otro trabajo que aborda la búsqueda de motivos en ADN es el trabajo de [Huo et al., 2010], pero para este trabajo utilizan una metaheurística con una función mono objetivo, en la investigación de [Koonin et al., 1994] se trabajó en la búsqueda de motivos en proteínas.

1.2. Planteamiento de problema de investigación

En la actualidad vivimos en la era de la información, donde cada segundo se generan grandes cantidades de información, pero ahora nos encontramos que faltan métodos para realizar un análisis más profundo de la basta cantidad de información generada.

Partiendo del problema de las grandes cantidades de información que se encuentra hoy en día, se puede encontrar grandes cantidades de proteínas que aún no se ha descrito su función o se han clasificado en familias que las representen. Además, con esta cantidad de información se puede generar nuevas familias que describan mejor a proteínas ya conocidas $^{y}/_{o}$ nuevas proteínas. Por tal razón, la búsqueda de motivos nos puede ayudar a describir su función. La estructura tridimensional de una proteína puede servir para describir motivos tridimensionales.

Por lo anterior, surge la necesidad de tener una herramienta que nos ayude encontrar motivos contenidos en proteínas, para su futura clasificación y asociación a una función

y estructura específica.

En la rama de biología molecular, existen muchos trabajos en la búsqueda de motivos lineales, pero poco trabajo en la descripción de motivos tridimensionales. Se tiene el problema de identificar motivos de las diferentes familias de proteínas, ya que hay evidencia de aquellas proteínas que comparten motivos, presentan la misma función y estructura, como se puede leer en [Arango et al., 2011] y [Xiong, 2012].

La búsqueda de motivos se clasifica como un problema NP-Completo [Li et al., 2002], ya que los motivos se componen por una secuencia de aminoácidos, los aminoácidos no tienen ninguna restricción sobre como conformar los motivos, por tal razón, el número de motivos de longitud n diferentes que se pueden generar con los 20 aminoácidos estándar se encuentra expresado por la Ecuación 1.1.

$$20^n \tag{1.1}$$

La búsqueda de motivos tridimensionales se puede realizar desde una secuencia lineal usando las propiedades de compatibilidad [Biro, 2006]. De igual manera, se puede buscar la optimización de más de una propiedad de los aminoácidos.

Al tratarse de un problema NP-Completo con múltiples objetivos, es abordado por métodos metaheurísticos para la búsqueda de una solución.

1.3. Justificación

El descubrimiento de motivos en secuencias biológicas es una tarea muy importante. Motivos similares pueden sugerir funciones similares lo que a su vez puede sugerir relaciones evolutivas. El descubrimiento de un motivo en un conjunto de secuencias biológicas se asocia a la conservación de una parte de la secuencia a la que se le puede asociar una funcionalidad determinada.

En la presente tesis se trabajará con grupos de proteínas de virus de la familia *coronaviridae*. Estos grupos de proteínas se encuentran en las bases de datos en las páginas de centros de investigación que se dedican a la actualización de las familias de proteínas y su caracterización funcional. De estos grupos se obtienen las proteínas con los mejores aciertos bidireccionales, para ser agrupadas por homología para buscar motivos tridimensionales conservados para poder definir una posible función. Debido a que las proteínas son palabras de un alfabeto de 20 letras y de cualquier longitud, el espacio de todas las posibles proteínas que se puede generar con este alfabeto es exponencial y la posibilidad de caracterizar incluso un subconjunto de este espacio es una tarea muy difícil.

Al finalizar el trabajo, se dejará una metodología para generar un conjunto de secuencias artificiales altamente compatible con una secuencia conservada de una familia de proteínas con una alta compatibilidad por sus propiedades fisicoquímicas descritas en [Biro, 2006]. A partir de una metaheurística multiobjetivo se genera un conjunto de secuencias artificiales que se utiliza como motivo para la caracterización de la familia de proteínas en estudio.

1.4. Objetivo

Diseñar e implementar una metaheurística multiobjetivo para el descubrimiento de motivos en un clúster de proteínas de virus de la familia *coronaviridae* basados en la metodología de los mejores aciertos bidireccionales.

1.4.1. Objetivos Específicos

- Seleccionar el o los grupos de proteínas aún no clasificados en la literatura.
- Elegir una metaheurística multiobjetivo a utilizar.

- Modelar el problema del descubrimiento de motivos como un problema multiobjetivo.
- Diseñar la metaheurística multiobjetivo para el problema de descubrimiento de motivos.
- Implementar la metaheurística multiobjetivo para el descubrimiento de motivos en clúster de proteínas basados en la metodología de los mejores aciertos bidireccionales.
- Evaluar la metaheurística multiobjetivo con criterios de eficiencia y efectividad.

1.5. Preguntas de investigación

- ¿Se pueden encontrar motivos con propiedades fisicoquímicas mediante el uso de una metaheurística multiobjetivo a partir de alineamientos múltiples de secuencias de proteínas agrupadas mediante un criterio específico?
- ¿Es posible describir motivos tridimensionales a partir de información no estructural?
- ¿Las propiedades fisicoquímicas son realmente excluyentes entre sí?
- ¿El número de iteraciones que realiza una metaheurística tendrá un impacto directo con la solución encontrada?
- De la información generada por la metaheurística, ¿Cuál es la información relevante?

Capítulo 2

Marco Teórico

Este capítulo es abordado en cinco secciones, las cuales se enfocan en dar los principales conceptos y métodos que se utilizan para abordar el problema de investigación. En este caso se trata del problema de descubrimientos de motivos para una familia de proteínas. En la sección 2.1 se dará la definición de aminoácido y proteína, su importancia, de qué están formadas, cómo se clasifican de acuerdo a sus estructuras y otros aspectos. Otros conceptos importantes que se tratará son el de motivo, dominios y familias que se refieren a la forma de agrupar subsecuencias de una proteína, a los cuales se les asocia una funcionalidad, una estructura 3D o alguna característica importante o relevante, además que en esta misma sección se dará una introducción al problema de descubrimiento de motivos.

En la sección 2.2 se expondrá la definición de problemas de optimización, los tipos de problemas de optimización que existen y qué características tiene cada tipo de problema de optimización.

Para la sección 2.3 se verá el concepto de heurísticas, metaheurísticas y varios algoritmos metaheurísticos que se utilizan para resolver problemas de optimización.

En relación con la Sección 2.4 se centrará en dar una explicación más detallada sobre el Algoritmo de Estimación de la Distribución (EDA), el cual es utilizado para la resolución del problema de investigación de la tesis y se abordará un planteamiento del problema de $OneMax$.

Para la sección 2.5 tenemos un análisis de trabajos semejantes con nuestro problema de investigación del descubrimiento de motivos.

2.1. Aminoácidos y proteínas

2.1.1. Aminoácidos

De acuerdo a [Feduchi et al., 2010], los aminoácidos "son un grupo heterogéneo de moléculas que poseen unas características estructurales y funcionales comunes. Existen veinte aminoácidos diferentes especificados en el código genético.".
Además [Mckee and MacKee, 2014] nos menciona que "existen 20 aminoácidos α estándar en las proteínas. Algunos de ellos tienen funciones únicas en los seres vivos." En la Tabla 2.1 se observan los 20 aminoácidos α estándar. Los aminoácidos α estándar se encuentran con regularidad en las proteínas y estos comparten una estructura general compuesta por un átomo de carbono central (el carbono α) al que están unidos un grupo amino, un grupo carboxilo, un átomo de hidrógeno y un grupo **R** (cadena lateral), ilustrada en la Figura 2.1. A excepción de la prolina, que tiene una estructura un poco diferente como podemos observar en la Figura 2.2. Los aminoácidos se pueden clasificar por sus propiedades químicas y físicas.

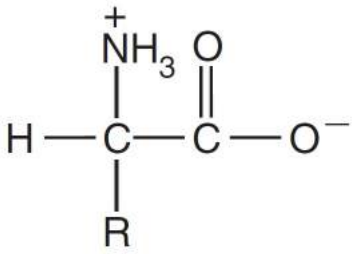


Figura 2.1: Estructura general de los aminoácidos. Obtenida de [Mckee and MacKee, 2014].

- Aminoácidos apolares: Interactúan poco con el agua, tienen un cometido importante en el mantenimiento de la estructura tridimensional de la proteína y carecen de carga positiva o negativa, en este grupo se encuentran dos tipos de hidrocarburos en la cadena **R**:

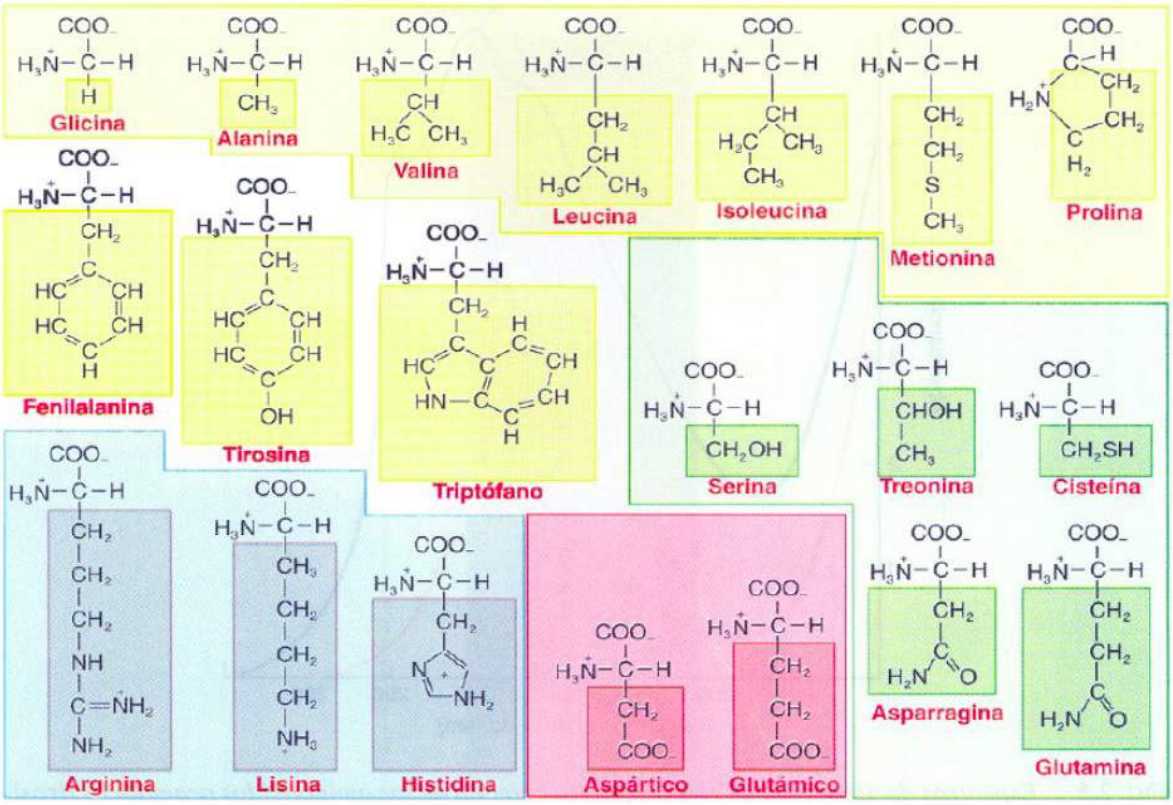


Figura 2.2: Clasificación de los 20 aminoácidos. Obtenida de [Calera and Sanz, 2003].

  - Aromáticos.
  - Alifáticos.

- Polares: Interactúan de manera sencilla con el agua y son capaces de formar enlaces por puentes de hidrógeno.
- Ácidos: Contiene carga negativa a pH fisiológico.
- Básicos: Contiene carga positiva a pH fisiológico.

2.1.2. Proteínas

Las proteínas son sustancias químicas que todo ser vivo necesita para llevar a cabo sus funciones elementales sin proteínas no hay vida. Las proteínas son sintetizadas y

Aminoácido	Abreviatura de tres letras	Abreviatura de una letra
Alanina	Ala	A
Cisteína	Cys	C
Ácido aspártico	Asp	D
Ácido glutámico	Glu	E
Fenilalanina	Phe	F
Glicina	Gly	G
Histidina	His	H
Isoleucina	Ile	I
Lisina	Lys	K
Leucina	Leu	L
Metionina	Met	M
Asparagina	Asn	N
Prolina	Pro	P
Glutamina	Gln	Q
Arginina	Arg	R
Serina	Ser	S
Treonina	Thr	T
Valina	Val	V
Triptófano	Trp	W
Tirosina	Tyr	Y

Tabla 2.1: Nombres y abreviaturas de los aminoácidos α estándar.

adquiridas. Las proteínas están compuestas de moléculas más pequeñas, conocidas como aminoácidos. Los aminoácidos están enlazados químicamente entre sí, la unión de estos aminoácidos forma una macromolécula. Se considera que existen 20 aminoácidos estándar, se pueden formar proteínas con propiedades fisicoquímicas diferentes. La estructura tridimensional de las proteínas es una propiedad emergente de las propiedades de los aminoácidos. El número de aminoácidos que puede contener una proteína no está limitado. De la inexistencia de alguna restricción sobre el número de aminoácidos o el orden de los aminoácidos que componen una proteína, surge la necesidad de clasificarlas. Esto da lugar a un número ilimitado de funciones y características.

En [Voet et al., 2007] menciona que "en los procesos biológicos las proteínas son el centro de acción. Prácticamente todas las transformaciones moleculares que definen el metabolismo celular están mediadas por catalizadores proteicos", de igual manera

en [Mckee and MacKee, 2014] define que "las proteínas son un grupo diverso de macromoléculas, esta diversidad está directamente relacionada con las posibilidades de combinación de los 20 aminoácidos", además nos menciona que las moléculas compuestas por menos de 50 aminoácidos, iguales o diferentes, se denominan **péptidos** y el término **proteína** describe específicamente las moléculas con un contenido de más de 50 aminoácidos.

2.1.3. Clasificación de las proteínas

La clasificación de las proteínas es una tarea sumamente importante, gracias a estas clasificaciones nos permite tener un mejor panorama de las características que conforman estas macromoléculas.

Actualmente se tienen muchas formas de clasificación de las proteínas, pero una de las clasificaciones que se utiliza actualmente es [Terfloth, 2009]:

- Su composición
  - Holoproteínas o proteínas simples: proteínas que al hidrolizarse producen únicamente aminoácidos.
  - Heteroproteínas o proteínas conjugadas: proteínas que al hidrolizarse producen aminoácidos y componentes orgánicos o inorgánicos.
- Su conformación: es la forma tridimensional que adquiere una proteína en el espacio.
  - Proteínas fibrosas: se componen por cadenas polipeptídicas alineadas de manera paralela, este tipo de proteínas conforman la estructura de los tejidos, son insolubles en agua.
  - Proteínas globulares: se conforman por cadenas polipeptídicas que se enrollan sobre sí misma, que se puede comparar con un nudo de hilo. La estructura

de este tipo de proteínas es esférica, la mayoría son solubles en agua y por lo general tienen funciones de transporte en el organismo.

- Su función: Cada proteína tiene una función en específico y muchas veces se necesita más de una proteína para realizar una función, algunos ejemplos de las funciones que puede presentar una proteína son:
  - Enzimas: utilización de reacciones bioquímicas.
  - Proteínas de transporte: se encarga de mover de un lugar a otro una sustancia determinada.
  - Proteínas de movimiento coordinado: modifican su estructura en relación con cambios en el ambiente que las rodea.
  - Proteínas de soporte o estructurales: componen los tejidos que soportan a un organismo, por ejemplo, huesos y tendones.
  - Anticuerpos: son proteínas específicas que tiene la capacidad de identificar y neutralizar sustancias extrañas al organismo.
  - Neurotransmisores: son partícipes en el proceso de recepción de impulsos nerviosos.
  - Proteínas represoras u hormonas: participan en la regulación de procesos metabólicos.

"Una de las claves para descifrar la función de una proteína dada es comprender su estructura" [Voet et al., 2007], la cual no es uniforme o regular, ya que depende totalmente en el orden en que se organizaron los aminoácidos, "la información sobre la secuencia de aminoácidos nos da una idea de las propiedades químicas y físicas de la proteína, sus relaciones con otra proteína" [Voet et al., 2007].

2.1.4. Estructura de las proteínas

Se puede encontrar como niveles estructurales o niveles de organización, "las proteínas pueden describirse en función de sus niveles de organización, en este caso sus estructuras primaria, secundaria, terciaria y cuaternaria" [Voet et al., 2007], como podemos ver en la Figura 2.3 se ilustran los 4 niveles de organización que a continuación son presentados.

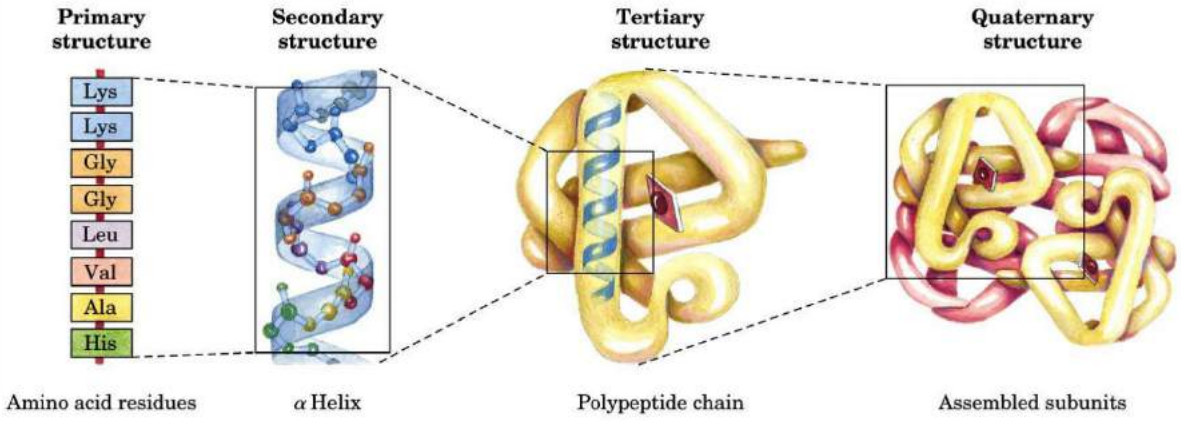


Figura 2.3: Niveles estructurales de las proteínas. Obtenida de [Amaru, 2019].

2.1.4.1. Estructura primaria

Consiste en el primer nivel de organización de la proteína, esta representación es utilizada al momento de analizar una proteína porque en ella se acomodan los aminoácidos de manera que, dependiendo el tipo de notación que se utilice, se coloca cada aminoácido de manera lineal, como podemos observar en la Figura 2.4, gracias a este nivel de organización es como se puede trabajar o procesar la información en equipos de cómputo o sistemas de información.

Para los siguientes niveles de organización **secundaria**, **terciaria** y **cuaternaria** se refiere a las formas tridimensionales que se pueden formar con las secuencias de aminoácidos que conforman o componen la proteína. "La estructura tridimensio-

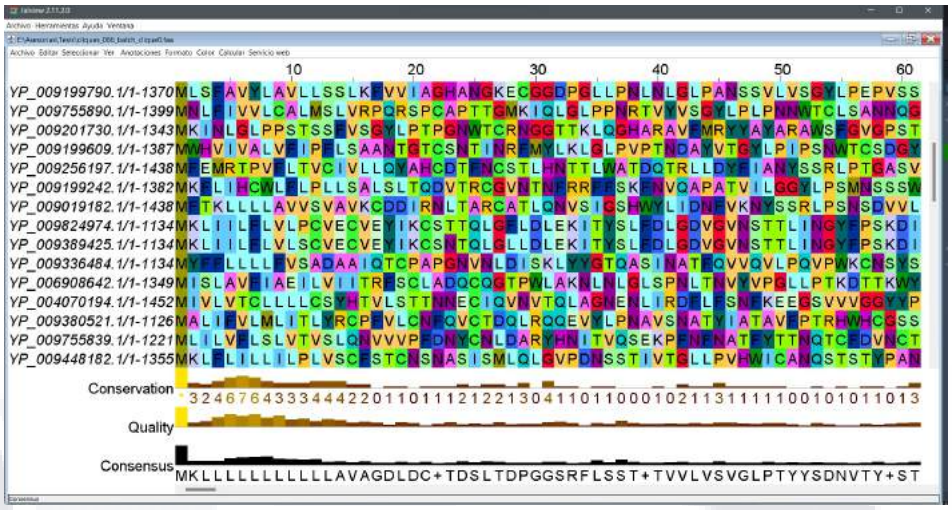


Figura 2.4: Secuencias de proteínas de coronavirus vistas con el software Jalview [Waterhouse et al., 2009]. Autoría propia.

nal de las proteínas está determinada por la identidad de los aminoácidos que la componen y el orden concreto en que estos aminoácidos se disponen en la cadena" [Feduchi et al., 2010].

Plegamiento:

De acuerdo a [Quiroz and Scherer, 2004], el plegamiento "explica el mecanismo que adopta una proteína con una configuración tridimensional unívoca y termodinámicamente estable". Además, en [Murray et al., 2013], se comenta que "el plegado de proteínas por lo general ocurre mediante un proceso por pasos". El plegamiento cuenta con dos etapas principales, en la primera etapa segmentos cortos de aminoácidos se acercan a unidades estructurales secundarias, en la segunda etapa, las secciones hidrofóbicas se acercan al interior de la proteína. Este proceso es ordenado y en algunos casos es dirigido por proteínas chaperonas, pero en la mayoría de los casos no está dirigido. Lo que ayuda a la proteína en su proceso de plegamiento son los elementos de la estructura secundaria, estos elementos por lo regular facilitan el proceso de plegado al dirigir el proceso. Como podemos observar en la Figura 2.5 nos muestra el proceso de plegamiento de una proteína, donde comienza la proteína desdoblada y termina totalmente doblada

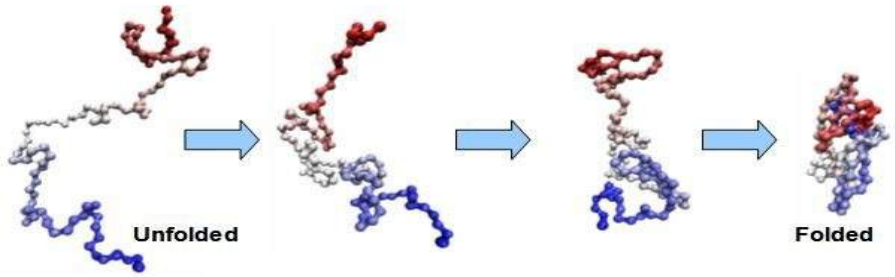


Figura 2.5: Plegamiento de proteína. Obtenida de [Flechsig, 2012].

o enrollada. El plegamiento se lleva a cabo cuando el péptido todavía está asociado al ribosoma y después del proceso de plegamiento, este se separa.

2.1.4.2. Estructura secundaria

La estructura secundaria se produce por el plegado de polipéptidos hacia motivos con enlaces de hidrógeno, como la hélice α, la hoja plegada β, flexiones β y asas. [Murray et al., 2013]. Estos ángulos se conocen como el ángulo fi (Φ) y el ángulo psi (Ψ). Φ es el ángulo alrededor del enlace $C_\alpha - N$ y Ψ es el ángulo que hay alrededor del enlace $C_\alpha - C_O$, donde C_α es el carbono α, C_O es el carbono carbonilo y N es el nitrógeno.

2.1.4.3. Estructura terciaria

En la estructura terciaria de acuerdo a [Feduchi et al., 2010] menciona que "en la estructura terciaria en la que las interacciones se pueden dar entre aminoácidos que están alejados en la secuencia primaria y que se encuentran en estructuras secundarias diferentes." Es decir, la estructura terciaria es utilizada para identificar las interacciones de los aminoácidos que se encuentran alejados, pero en realidad la estructura terciaria es la única que existe. La estructura secundaría y primaria se utilizan para su estudio. La estructura terciaria es la estructura real de una proteína.

2.1.4.4. Estructura cuaternaria

De acuerdo a [Feduchi et al., 2010] menciona que "existe la posibilidad de que varias cadenas proteicas o subunidades (que pueden ser iguales o diferentes) se asocien en complejos tridimensionales que componen la **estructura cuaternaria** de las proteínas". Es decir, la estructura cuaternaria es la interacción de dos o más proteínas iguales o diferentes, ya sea en su totalidad o en subestructuras que contienen las proteínas.

2.1.4.5. Motivos

Según [Feduchi et al., 2010] menciona que "existen combinaciones de segmentos con estructuras secundarias definidas que se asocian con una disposición espacial característica formando **motivos** que se repiten en diferentes proteínas", de igual manera este mismo autor menciona que "los elementos denominados **motivos** o **estructuras supersecundarias** son combinaciones de varios elementos con estructura secundaria definida con una disposición geométrica característica", además este mismo autor menciona que "algunos motivos tienen funciones biológicas específicas, pero otros simplemente forman parte de otras unidades estructurales y funcionales". Mientras otros autores como [Mckee and MacKee, 2014] definen a los **motivos estructurales** o **estructuras supersecundarias** como patrones. De igual manera [Voet et al., 2007] dice que "los **motivos** pueden tener tanto significado estructural como funcional". En [Feduchi et al., 2010] nos menciona que "Varios motivos se pueden asociar formando **dominios** que suelen ser unidades funcionales".

2.1.4.6. Dominios

Como se puede ver en [Mckee and MacKee, 2014] se define "los **dominios** son segmentos independientes en términos estructurales que poseen funciones específicas". Además este mismo autor menciona que "la estructura tridimensional central de un

dominio se denomina **pliegue**". Por su parte [Feduchi et al., 2010] dice "en algunas proteínas se encuentran estructuras globulares bien definidas formadas por la combinación de varios motivos que se denominan **dominios**. Estructuralmente, los dominios están formados por diferentes combinaciones de elementos con una estructura secundaria concreta o de diferentes motivos." Además menciona el autor que éstas estructuras también se definen como la parte de la cadena que se puede plegar de forma independiente formando una estructura estable y éstas unidades suelen estar también asociadas a una funcionalidad. Mientras que [Murray et al., 2013] define "Un **dominio** es una sección de estructura proteínica suficiente para desempeñar una tarea química o física particular, casi todos los dominios son de naturaleza modular". Por las definiciones anteriores, se interpreta que los dominios son una secuencia de aminoácidos de una longitud considerable, los cuales tiene una estructura $^{y}/_{o}$ funcionalidad definida. Además, una proteína puede estar constituida por múltiples dominios. Las proteínas que comparten grandes características se agrupan en familias.

Familias:

Con base a [Mckee and MacKee, 2014] se define que "las familias de proteínas están formadas por moléculas relacionadas por su similitud en la secuencia de aminoácidos, evidentemente, tales proteínas comparten un ancestro común". Además, el autor [Feduchi et al., 2010] añade que las familias de proteínas comparten precursores comunes. La clasificación de una proteína en una familia se realiza por homología, un procedimiento que se explica a continuación.

2.1.5. Homología de las proteínas

Como menciona [Murray et al., 2013] sobre la homología:

"Un método importante para la identificación de proteínas y productos de gen nuevos es por medio de comparación con proteínas de secuencia o estructura conocida.

Dicho de modo más sencillo, las búsquedas de homología y las comparaciones de secuencias múltiples operan con base en el principio de que las proteínas que desempeñan funciones similares compartirán dominios conservados u otras características de secuencias o **motivos**, y viceversa. De los muchos algoritmos creados para este propósito, el más ampliamente usado es el **BLAST**."

Además en [Mckee and MacKee, 2014] menciona que el "método llamado modulación de homología ha facilitado la predicción de estructuras de proteínas." En otras palabras la homología es un proceso de comparación entre dos o más secuencias de aminoácidos, por lo general las secuencias están alineadas, donde se compara cada aminoácido de la secuencia con los aminoácidos de la otra u otras secuencias por columna, como podemos observar en la Tabla 2.2, es un claro ejemplo de como se utiliza la homología en la práctica.

Idioma	Palabra	Alineación	Homología
Inglés	PHYSIOLOGICAL	PHYSIOLOGI-CAL -	-
Francés	PHYSIOLOGIQUE	PHYSIOLOGI-QUE -	10
Alemán	PHYSIOLOGISCH	PHYSIOLOGISCH- -	11
Holandés	FYSIOLOGISCH	F -YSIOLOGISCH- -	9
Español	FISIOLOGICO	F - ISIOLOGI -CO- -	8
Polaco	FIZJOLOGICZNY	F -IZJOLOGI -CZNY	6

Tabla 2.2: Representación de una alineación de secuencias múltiples. Los idiomas evolucionan de un modo que imitan la manera en que lo hacen los genes y las proteínas. Aquí se muestra la palabra physiological del inglés en varios idiomas. La alineación demuestra sus características conservadas. Las identidades con la palabra en inglés se muestran en color rojo oscuro; las similitudes lingüísticas en color rosa. Obtenida de [Murray et al., 2013].

En la Tabla anterior, observamos un análisis de homología que se realizó a la palabra fisiológico (PHYSIOLOGICAL, en inglés), en su manera de escribir en 5 idiomas diferentes al inglés, además de hacer la distinción de las coincidencias de la palabra en inglés con la letra de la palabra en otro idioma, se destacó la fonética similar de las letras que componen a la palabra, pero antes de hacer este análisis se realizó un proceso

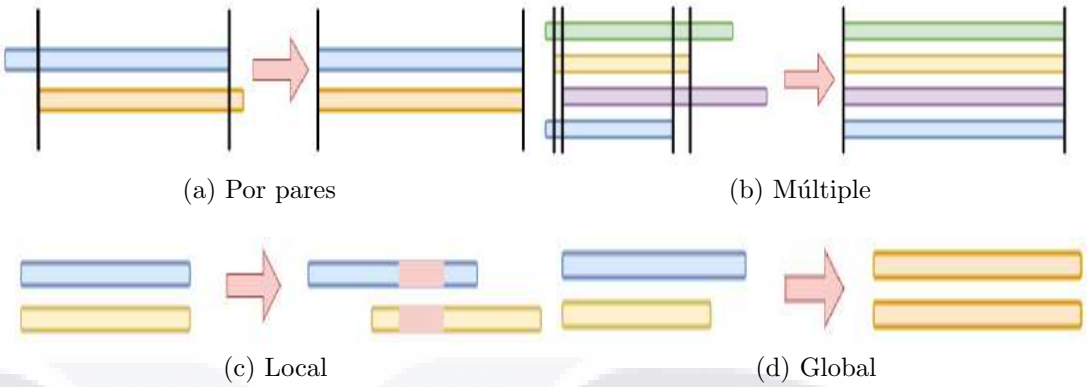
previo conocido como alineamiento de secuencias.

2.1.5.1. Alineamiento de secuencias

El alineamiento de secuencias es un proceso que se realiza a un conjunto de proteínas en bruto, el cual consiste en acomodar por columna a los aminoácidos que componen a las secuencias del conjunto con la intención de que se alcance el máximo número de coincidencias, además que se encarga de que todas las secuencias que integran al conjunto tengan el mismo número de columnas, esto se logra mediante la agregación de **gaps "-"** en las secuencias.

Existen diferentes tipos de alineamientos, como se puede observar en la Figura 2.6, los alineamientos se pueden clasificar de la siguiente manera:

- De acuerdo a las secuencias a alinear.
  - Por pares: solo se realiza el alineamiento entre dos secuencias. Observar la Figura 2.6a.
  - Múltiple: alineamiento que se realiza a partir de 3 secuencias. Observar la Figura 2.6b.
- De acuerdo a la región a alinear.
  - Local: se realiza el alineamiento en una región determinada de la secuencia, es decir en una subregión. Observar la Figura 2.6c
  - Global: se realiza el alineamiento en toda la región de la secuencia. Observar la Figura 2.6d



(a) Por pares (b) Múltiple

(c) Local (d) Global

Figura 2.6: Tipos de alineamientos. Autoría propia.

En la actualidad existen diferentes tipos de software con los cuales se pueden realizar alineamientos múltiples de secuencias, para nuestro caso se realizó el AMS por medio del software de **Clustal** ω [Larkin et al., 2007]. Gracias al AMS podemos abordar el Problema de Descubrimiento de Motivos (PDM) de una manera apropiada.

2.1.6. Problema de descubrimiento de motivos

El PDM es un problema típico en la bioinformática. Consiste en identificar zonas en las secuencias biológicas ya sean secuencias de ADN, ARN o aminoácidos que presenten un patrón o una zona conservada. Este patrón se puede identificar en la misma secuencia biológica o en diferentes secuencias. Para este trabajo de investigación el descubrimiento de motivos se realiza con una familia de proteína.

La manera computacional de cómo se aborda es:

Sea $M = \{M_i | i = 1, 2, \ldots, n\}$ un AMS el cual se aborda como una matriz de tamaño n x m, donde n es el número de proteínas en el alineamiento y m es el número de columnas que conforman el alineamiento, es decir, $|M_i| = m, \forall i \in \{1, 2, \ldots, n\}$. Sea $M_i = \left\{M_i^j | j = 1, 2, \ldots, m\right\}$ el $i - esimo$ renglón del AMS y M_i^j el valor de la celda i, j de la matriz asociada, M_i^j puede ser uno de los 20 aminoácidos o un **gap "-"**. La Figura 2.7 muestra un AMS M, en la parte izquierda se tiene los identificadores de las

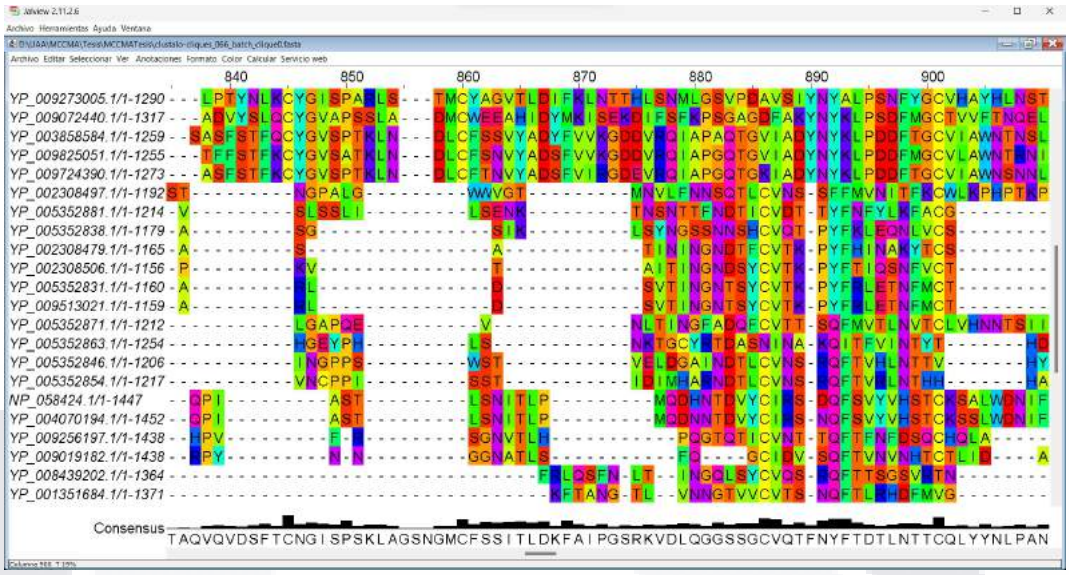


Figura 2.7: Alineamiento Múltiple de Secuencias. Autoría propia.

proteínas de la familia *coronaviridae* que conforma el AMS donde $n = 22$ proteínas y $m = 910$, visto con el software Jalview.

El siguiente paso es la construcción de una Matriz Consenso (C(M)) de tamaño 20xm que almacena la frecuencia que tiene cada uno de los 20 aminoácidos posibles por columna M^j.

$C(M) = \{C(M^1), (M^2), \ldots, C(M^m)\}$ donde $C(M^j) = \left\{C\left(M_b^j\right) | b \in B\right\}$

$\forall j = 1, 2, \ldots, m,\ B = \{A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y\}$, $|B| =$ 20 y los elementos de B representan a los aminoácidos en su codificación de una letra. $C\left(M_b^j\right)$ es la celda del renglón b y la columna j que representa la frecuencia de aparición del aminoácido b en la columna j del AMS. A partir de la Figura 2.7 se generó la C(M) contenida en la Tabla 2.3

Al obtenerse la Matriz Consenso C(M) se comienza la búsqueda de Subsecuencias Conservadas (S(C(M))) las cuales se definen con una longitud l determinada por el investigador. Para describir cada subsecuencia conservada se define un vector de longitud l donde cada valor se busca en la matriz C(M) siguiendo el criterio de poner aquel aminoácido que tenga la mayor frecuencia de aparición por columna hasta completar

A	5	0	0	0	0	1	1	0	1	1	0	8	0	2	0	1	1	0	0	1	0
C	0	0	0	0	2	7	0	1	0	0	0	4	4	0	2	1	0	0	0	0	1
D	2	0	0	0	0	3	0	0	0	0	0	8	0	1	0	3	4	0	0	0	1
E	0	0	9	0	0	1	0	0	0	1	0	1	0	5	0	0	3	1	0	0	1
F	2	0	0	0	0	0	0	0	0	0	0	0	1	1	0	5	9	3	0	0	1
G	0	0	1	0	2	5	1	2	0	3	0	0	0	0	0	1	0	1	0	6	0
H	1	16	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	2	0	0	0
I	0	0	0	0	1	0	0	7	0	0	0	0	0	0	0	0	0	14	0	0	0
K	4	0	2	0	0	0	0	0	0	0	0	5	0	2	2	1	6	0	0	0	0
L	1	0	3	0	0	0	0	1	5	0	0	0	0	0	0	7	4	1	0	0	0
M	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	17
N	0	0	1	0	0	0	0	0	1	0	0	7	5	0	3	3	2	0	0	0	0
P	0	0	0	0	1	0	0	0	0	0	0	0	0	10	0	0	0	0	0	11	0
Q	1	0	0	0	16	0	0	1	4	0	0	0	0	0	0	0	0	0	0	0	0
R	0	0	0	0	0	0	1	0	1	5	2	1	0	0	2	2	8	0	0	0	0
S	0	0	0	0	3	0	0	2	0	4	0	0	5	0	0	0	1	7	0	0	0
T	0	0	3	3	0	0	1	0	0	0	0	4	0	1	2	3	1	1	0	3	0
V	1	0	4	0	1	1	1	2	0	4	0	1	0	1	0	1	2	3	0	0	0
W	0	0	2	0	5	0	2	0	2	0	0	9	0	0	0	1	1	0	0	0	0
Y	0	0	0	0	6	0	1	0	0	1	2	0	0	0	0	3	5	2	0	2	0

Tabla 2.3: Matriz Consenso. Matriz generada a partir de la columna 880 hasta la columna 900 del AMS de la Figura 2.7.

los l elementos del vector. Para que una subsecuencia sea considerada como conservada, su Índice de Conservación (I_c) debe ser mayor a un porcentaje α también definido por el investigador. El I_c se calcula a partir de la Ecuación 2.1.

$$I_c = \frac{\sum_{j=1}^{l} \text{máx}_b \left(C \left(M_b^j \right) \right)}{nl} \tag{2.1}$$

Aplicando la Ecuación 2.1 en la Tabla 2.3 obtenemos su I_c contenido en la Tabla 2.4.

A partir de la Tabla 2.1 se obtuvieron las subsecuencias conservadas de tamaño $l = 5$ contenidas en la Tabla 2.5.

A	22.27	0.0	0.0	0.0	0.0	4.54	4.54	0.0	4.54	4.54	0.0	36.36	0.0	9.09	0.0	4.54	4.54	0.0	0.0	4.54	0.0
C	0.0	0.0	0.0	0.0	9.09	31.81	0.0	4.54	0.0	0.0	0.0	18.18	18.18	0.0	9.09	4.54	0.0	0.0	0.0	0.0	4.54
D	9.09	0.0	0.0	0.0	0.0	13.63	0.0	0.0	0.0	0.0	0.0	36.36	0.0	4.54	0.0	13.63	18.18	0.0	0.0	0.0	4.54
E	0.0	0.0	40.90	0.0	0.0	4.54	0.0	0.0	0.0	4.54	0.0	4.54	0.0	22.27	0.0	0.0	13.63	4.54	0.0	0.0	4.54
F	9.09	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	4.54	4.54	0.0	22.27	40.90	13.63	0.0	0.0	4.54
G	0.0	0.0	4.54	0.0	9.09	22.27	4.54	9.09	0.0	13.63	0.0	0.0	0.0	0.0	0.0	4.54	0.0	4.54	0.0	27.2	0.0
H	4.54	72.3	4.54	0.0	0.0	0.0	0.0	0.0	4.54	0.0	0.0	4.54	0.0	0.0	0.0	0.0	0.0	9.09	0.0	0.0	0.0
I	0.0	0.0	0.0	0.0	4.54	0.0	0.0	31.81	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	63.63	0.0	0.0	0.0
K	18.18	0.0	9.09	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	22.27	0.0	9.09	9.09	4.54	27.2	0.0	0.0	0.0	0.0
L	4.54	0.0	13.63	0.0	0.0	0.0	0.0	4.54	22.27	0.0	0.0	0.0	0.0	0.0	0.0	31.81	18.18	4.54	0.0	0.0	0.0
M	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	22.27	77.2
N	0.0	0.0	4.54	0.0	0.0	0.0	0.0	0.0	4.54	0.0	0.0	31.81	22.27	0.0	13.63	13.63	9.09	0.0	0.0	0.0	0.0
P	0.0	0.0	0.0	0.0	4.54	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	45.45	0.0	0.0	0.0	0.0	0.0	50.0	0.0
Q	4.54	0.0	0.0	0.0	72.3	0.0	0.0	4.54	18.18	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
R	0.0	0.0	0.0	0.0	0.0	0.0	4.54	0.0	4.54	22.27	9.09	4.54	0.0	0.0	9.09	9.09	36.36	0.0	0.0	0.0	0.0
S	0.0	0.0	0.0	0.0	13.63	0.0	0.0	9.09	0.0	18.18	0.0	0.0	22.27	0.0	0.0	0.0	4.54	31.81	0.0	0.0	0.0
T	0.0	0.0	13.63	13.63	0.0	0.0	4.54	0.0	0.0	0.0	0.0	18.18	0.0	4.54	9.09	13.63	4.54	4.54	0.0	13.63	0.0
V	4.54	0.0	18.18	0.0	4.54	4.54	4.54	9.09	0.0	18.18	0.0	4.54	0.0	4.54	0.0	4.54	9.09	13.63	0.0	0.0	0.0
W	0.0	0.0	9.09	0.0	22.27	0.0	9.09	0.0	9.09	0.0	0.0	40.90	0.0	0.0	0.0	4.54	4.54	0.0	0.0	0.0	0.0
Y	0.0	0.0	0.0	0.0	27.2	0.0	4.54	0.0	0.0	4.54	9.09	0.0	0.0	0.0	0.0	13.63	22.27	9.09	0.0	9.09	0.0

Tabla 2.4: Matriz de I_c. Matriz generada a partir de la columna 880 hasta la columna 900 del AMS de la Figura 2.7.

4	DTYCV	49.09
5	TYCVT	46.36
6	YCVTS	44.54
12	NYFTV	44.54
3	NDTYC	43.63
7	CVTSY	43.63
11	YNYFT	42.72
10	SYNYF	41.81
13	YFTVN	41.81
1	NGNDT	37.27

Tabla 2.5: Subsecuencias Conservadas. Secuencias de $L = 5$ generadas a partir del Ic contenido en la Tabla 2.4.

2.2. Problemas de optimización

Todo problema de optimización tiene el objetivo de maximizar o minimizar (dependiendo el caso), vea Ecuación 2.2. La calidad de una solución depende de un conjunto de variables, estas variables pueden ser continuas o discretas según sea la naturaleza del problema. El valor que pueden tomar las variables con las que se resuelve el problema de optimización pueden estar sujetas a restricciones del tipo de:

- Mayor que, ver ejemplo en Ecuación 2.3.

- Menor que, ver ejemplo en Ecuación 2.4.
- Igualdad, ver ejemplo en Ecuación 2.5.

En la Figura 2.8 se presentan los posibles valores que pueden tomar las variables x y y para cumplir una desigualdad. En la parte sombreada clara podemos encontrar los valores de las variables que cumplen con una de las dos restricciones ($4x + y > 5$ ó $-2x + 3y > 4$). La parte sombreada oscura representan los valores de las variables que cumplen con las dos restricciones. La parte sin sombrear son los valores de las variables que no cumple con ninguna desigualdad.

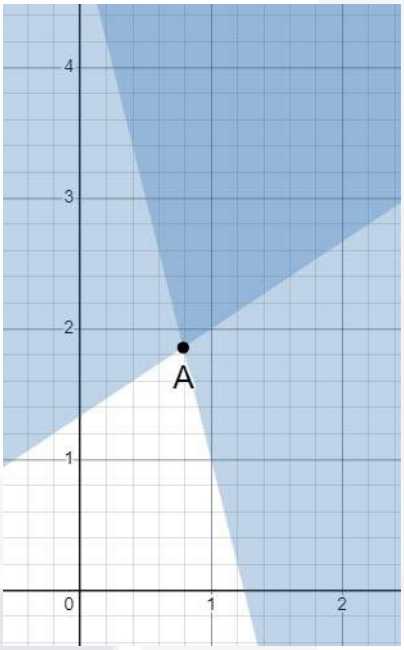


Figura 2.8: Posibles valores en una desigualdad. Autoría propia.

De manera formal un problema de optimización se puede definir como:

$$\text{máx}\left\{F\left(\vec{x}\right)\right\} \text{ ó } \text{mín}\left\{F\left(\vec{x}\right)\right\} \tag{2.2}$$

Sujeto a:

$$P_i(\vec{x}) > 0 \text{ para } i = 1, 2, 3, \ldots n \tag{2.3}$$

$$Q_i(\vec{x}) < 0 \text{ para } i = 1, 2, 3, \ldots m \tag{2.4}$$

$$R_i(\vec{x}) = 0 \text{ para } i = 1, 2, 3, \ldots k \tag{2.5}$$

Donde:

- $\vec{x}$ representa a un vector de una a más variables
- n representa el número de restricciones del tipo mayor que del problema
- m representa el número de restricciones del tipo menor que del problema
- k representa el número de restricciones del tipo igual que del problema

Desde su principio los problemas de optimización tienen la meta de encontrar aquellos valores que satisfaga de la mejor manera la función objetivo de nuestro problema. La mejor solución posible que tiene un problema de optimización es conocida como óptimo global, mientras que la mejor solución encontrada en una zona acotada de las posibles soluciones de nuestro problema, se le conoce como óptimo local. En la Figura 2.9 podemos observar un ejemplo de óptimo local y óptimo global. El valor marcado por el punto **G** es un óptimo global porque es el valor máximo que toma la función, mientras que los puntos **B** y **H** son óptimos locales, tomando en cuenta que el problema de optimización es de maximizar. Con el paso del tiempo surgieron aquellos problemas en los cuales no solo se tenía una función objetivo. Aquellos problemas que tienen más de un objetivo a optimizar. Al tener más de un objetivo de optimización un problema tiene la complicación de que los objetivos son excluyentes entre sí.

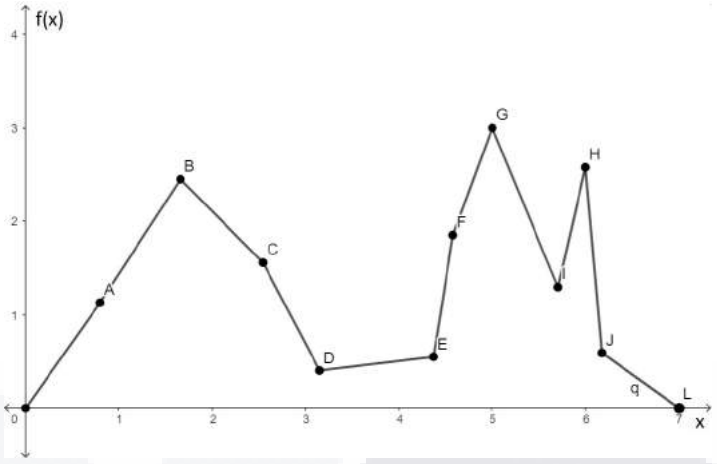


Figura 2.9: Optimización de una función. Autoría propia.

Un problema clásico de optimización con variable discreta es el Problema del Agente Viajero (TSP) (Traveling Salesman Problem)[Larrañaga et al., 1999]. El problema consiste en un vendedor ambulante que necesita recorrer un conjunto de n ciudades, en las cuales ofrece sus productos, pero no debe de pasar 2 veces por una misma ciudad, además debe regresar a la ciudad de la cual comenzó su viaje. Dicho problema a lo largo de su historia sufrió diferentes modificaciones o planteamientos. El TSP comenzó únicamente como un problema de minimizar el costo del camino, la distancia del recorrido o el tiempo necesario para realizar el recorrido. Con el paso del tiempo el planteamiento del problema cambio, ahora ya no es suficiente que el problema tenga únicamente una meta. Ahora el problema TSP debe de optimizar más de una meta, es decir mejorar el tiempo del recorrido, la distancia a recorrer o el costo del recorrido o cualquier combinación de dos o más metas. La implicación de que se quiera optimizar el TSP con más de una meta, es que estas metas son excluyentes entre sí, es decir si se quiere minimizar el tiempo de recorrido, posiblemente se debe de aumentar el costo del recorrido y viceversa. Por tal razón se comenzó a clasificar los problemas en **mono-objetivo** y **multi-objetivo**.

2.2.1. Problema de optimización mono-objetivo

Los problemas de optimización catalogados como mono-objetivo se caracterizan por tener o funcionar en base a una única función objetivo. La función objetivo se compone desde una a n variables diferentes. Un ejemplo de una función objetivo con una variable la podemos encontrar en la Ecuación 2.6. Un ejemplo de una función objetivo compuesta por dos o más variables, en este caso 4 variables, la podemos observar en la Ecuación 2.7.

$$F(x) = 2 - 2x - 0.8x^2 - x^4 \tag{2.6}$$

$$F(w, x, y, z) = \left[\frac{5 - 3w + 7x}{2y}\right]^z \tag{2.7}$$

En el caso de que nuestro problema tenga más de una función objetivo, se puede tomar las siguientes posturas:

- Seleccionar solo una función objetivo y buscar los valores de las variables para optimizar la función.
- Trabajar con una función de agregación. Ver Párrafo 2.2.2.2
- Tomarlo como un problema multi-objetivo.

2.2.2. Problema de multi-objetivo

Los problemas de optimización multi-objetivo se caracteriza por contener un número de objetivos (no) a optimizar. Por lo general, los objetivos son excluyentes entre sí. Es decir, para mejorar un objetivo, se debe de empeorar en al menos otro objetivo de nuestro problema. Estos problemas tienen un espacio de soluciones factibles (Ω). Se debe de encontrar un $\vec{x}$, donde $x_i \in \Omega, \forall i \in \{1, 2, \ldots, no\}$, que satisfagan cada

restricción, ver Ecuación 2.10, 2.11 y 2.12. Partiendo del $\vec{F}$ que $f_i, \forall i \in \{1, 2, \ldots, no\}$. Donde cada f_i es considerada como un objetivo independiente del problema y se busca su optimización, ver Ecuación 2.8 y 2.9. Al considerar cada f_i independiente se obtiene $\vec{X} = \{\vec{x}_i \mid i = 1, 2, \ldots, nu\}$, donde nu es el número de soluciones óptimas factibles encontradas en Ω ver Ecuación 2.13. En $\vec{X}$ se contienen las soluciones que conforman el frente de Pareto.

$$\text{máx}\left\{\vec{F}\left(\vec{x}\right)\right\} = \text{máx}\left\{f_1\left(\vec{x}\right), f_2\left(\vec{x}\right), \ldots, f_{no}\left(\vec{x}\right)\right\} \tag{2.8}$$

ó

$$\text{mín}\left\{\vec{F}\left(\vec{x}\right)\right\} = \text{mín}\left\{f_1\left(\vec{x}\right), f_2\left(\vec{x}\right), \ldots, f_{no}\left(\vec{x}\right)\right\} \tag{2.9}$$

Sujeto a:

$$P_i(\vec{x}) > 0 \text{ para } i = 1, 2, 3, \ldots n \tag{2.10}$$

$$Q_i(\vec{x}) < 0 \text{ para } i = 1, 2, 3, \ldots m \tag{2.11}$$

$$R_i(\vec{x}) = 0 \text{ para } i = 1, 2, 3, \ldots k \tag{2.12}$$

$$\vec{x_i} \in \Omega \text{ para } i = 1, 2, 3, \ldots no \tag{2.13}$$

Donde:

- $\vec{x}$ representa a un vector de una a más variables
- n representa el número de restricciones del tipo mayor que del problema
- m representa el número de restricciones del tipo menor que del problema
- k representa el número de restricciones del tipo igual que del problema
- Ω espacio de soluciones factibles

2.2.2.1. Frente de Pareto

Se conforma por aquellas soluciones que $\vec{x}_i \in \Omega, \forall i \in \{1, 2, \ldots, no\}$ que cumplen con el concepto de solución no dominada expresada a continuación.

Sea $f_1, f_2 \in \vec{F}$, $A, B \in \vec{X}$. Para que A, B sean soluciones no dominadas cumplen que $f_1(A) > f_1(B)$ y $f_2(A) < f_2(B)$. Un ejemplo de soluciones factibles de un problema multi-objetivo lo encontramos en la Figura 2.10, donde las soluciones dominadas son representadas por un cuadrado y las soluciones no dominadas son representadas por círculos. Las soluciones no dominadas conforman el frente de Pareto.

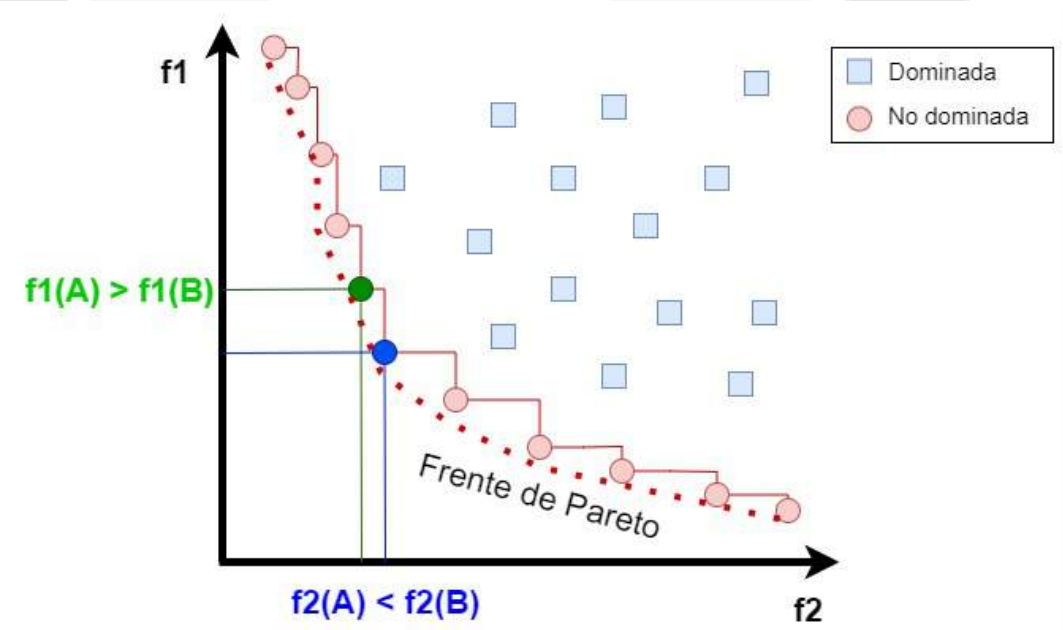


Figura 2.10: Frente de Pareto. Autoría propia.

Una alternativa para trabajar con los problemas multiobjetivo es mediante una función de agregación.

2.2.2.2. Función de agregación

Consiste que cada objetivo que compone nuestro problema de optimización tenga un peso (ω_i). Donde $\omega_i \in \vec{\omega}, \forall i \in \{1, 2, \ldots, no\}$. Además de que el $\vec{\omega}$ debe cumplir con

las propiedades expresadas en las Ecuaciones 2.14 y 2.15.

$$\forall \omega_i \in [0,1], \forall i \in \{1,2,\ldots,no\} \tag{2.14}$$

$$\sum \vec{\omega} = 1 \text{ es decir } \sum_{i=1}^{no} \omega_i = 1 \tag{2.15}$$

Supongamos el siguiente ejemplo, tenemos un problema de optimización de maximizar con 4 objetivos, entonces tenemos que $\vec{\omega} = (\omega_1, \omega_2, \omega_3, \omega_4)$ y cada ω tiene los siguientes valores $0.2, 0.3, 0.4, 0.1$, respectivamente. Nuestra función objetivo del problema a optimizar está representada por la Ecuación 2.16 y en la Ecuación 2.17 observamos la sustitución de los ω por su valor numérico, cada objetivo particular está representado por $f_i, \forall i \in \{1,2,3,4\}$.

$$\text{máx}\{\omega_1\ f_1(\vec{x}) + \omega_2\ f_2(\vec{x}) + \omega_3\ f_3(\vec{x}) + \omega_4\ f_4(\vec{x})\} \tag{2.16}$$

$$\text{máx}\{0.2\ f_1(\vec{x}) + 0.3\ f_2(\vec{x}) + 0.4\ f_3(\vec{x}) + 0.1\ f_4(\vec{x})\} \tag{2.17}$$

Los problemas de optimización por lo general llegan a ser un problema difícil de resolver. Esto es porque el Ω del problema es tan extenso que ni la mejor supercomputadora puede encontrar la mejor solución en un lapso de tiempo razonable. Por tal razón la comunidad científica ha desarrollado un conjunto de herramientas que nos permitan encontrar una muy buena solución en un tiempo aceptable. Estas herramientas se les conocen como metaheurísticas.

2.3. Heurísticas y metaheurísticas

En esta sección daremos una introducción sobre las heurísticas y metaheurísticas, las cuales son técnicas utilizadas para la resolución de problemas de optimización donde su espacio de soluciones factibles es inmenso. Estas técnicas tienen características particulares. Las metaheurísticas son las sucesoras de las heurísticas.

2.3.0.1. Heurísticas

Son técnicas clásicas que nos permiten resolver problemas de optimización. Estas técnicas tienen el problema de estancarse en un óptimo local y difícilmente escapar del mismo. Estas técnicas no garantizan encontrar la mejor solución del problema, pero en la mayoría de los casos encuentra una solución muy buena. Mientras que [Silver, 2004] define las heurísticas de la siguiente manera "método que, sobre la base de la experiencia o de un juicio, puede producir una solución razonable a un problema, pero que no se puede garantizar que produzca la solución matemática óptima".

2.3.0.2. Metaheurísticas

Son un conjunto de herramientas desarrolladas para resolver problemas con un espacio de posibles soluciones inmenso. Tienen su origen en las heurísticas. Se consideran plantillas a las cuales se les llena con información específica de cada problema a optimizar. Las metaheurísticas son un equilibrio entre diversificación e intensificación. Diversificación hace referencia a la propiedad de la herramienta de explorar diferentes soluciones de todas las posibles soluciones e intensificación se refiere a la búsqueda de soluciones en zonas pequeñas en comparación al espacio de posibles soluciones. Estas zonas pequeñas se conocen como vecindades. Para [Silver, 2004] define las metaheurísticas como "proceso maestro iterativo que guía y modifica las operaciones de heurísticas subordinadas para producir eficientes soluciones de alta calidad".

Las metaheurísticas tienen las siguientes propiedades:

- Son algoritmos no exactos y por lo general son no deterministas.
- Su estructura básica tiene una estructura predefinida.
- Su objetivo es una exploración eficiente del espacio de posibles soluciones.
- Son plantillas generales que guían el proceso de búsqueda.
- Pueden implementar métodos o funciones para evitar el estancamiento de zonas pocas prometedoras.
- Su esquema básico tiene una estructura predefinida.
- Hace uso del conocimiento del problema que se va a resolver, es moderado por una estrategia de más alto nivel.

Algunas de las estrategias implementadas por las metaheurísticas son:

Óptimo local:
Ya sea mínimo o máximo, el algoritmo explora zonas prometedoras del espacio de posibles soluciones, es decir el algoritmo explora los vecindarios de las posibles soluciones que se tienen.

Componente de aprendizaje:
El algoritmo comienza a reconocer la relación entre las variables del problema y el espacio de posibles soluciones, esto lo puede hacer de manera implícita o explícitamente.

2.3.1. Algoritmos metaheurísticos

Los algoritmos son un conjunto de instrucciones ordenadas, claras y finitas, con las cuales podemos llegar a un resultado o la solución de un problema en las mismas

condiciones. Es decir, un algoritmo siempre dará o llegará al mismo resultado mientras las condiciones en las que se aplique o utilice sean las mismas.

Mientras que un algoritmo metaheurístico es un método enfocado en la búsqueda del óptimo global en un problema de optimización con tiempo computacional aceptable. Por lo general estos algoritmos no encuentran el óptimo global del problema, pero encuentran soluciones muy buenas y que posiblemente se encuentren en la vecindad o zona del óptimo global.

Existen un gran número de algoritmos metaheurísticos que tienen características muy similares o parecidas, ya sea en la forma que se comporta el algoritmo o el origen de estos. Por estas razones se crearon clasificaciones para los algoritmos. En la sección 2.3.1.1 se aborda a más a detalles los algoritmos basados en poblaciones y se presentan algunos algoritmos de este tipo. Para la sección 2.3.1.2 se profundiza en los algoritmos evolutivos que tiene su origen en la teoría de la evolución de las especies [Darwin, 1859].

2.3.1.1. Algoritmos basados en poblaciones

En el trabajo de [Blum and Roli, 2003] se presentan algunas formas de clasificar los algoritmos metaheurísticos. La clasificación de los algoritmos basados en poblaciones se basan principalmente en el comportamiento de la población de una especie. De manera artificial se imita el comportamiento de la población de la especie o su forma de realizar una actividad, por ejemplo, la forma de recolectar comida por las hormigas. Algunos ejemplos de los algoritmos clasificados de esta forma se abordan a continuación.

Algoritmo de Colonia de Hormigas (ACO):

Este algoritmo se basa en el comportamiento que presentan las colonias de hormigas en su proceso de búsqueda y recolección de alimento fuera del hormiguero. Se conoce que las hormigas son ciegas, pero estas tienen una feromona que les permite seguir el camino que una hormiga recorrió. La feromona depositada en el camino que sigue una

hormiga dura un tiempo en el ambiente y después se evapora. Las hormigas al comenzar su búsqueda de alimento salen de forma aleatoria del hormiguero en búsqueda del alimento, cuando una hormiga encuentra el alimento regresa al hormiguero para después volver a salir por alimento. Los caminos que recorren las hormigas se les deposita una cantidad determinada de feromona. Después de un cierto número de hormigas que salen y entran al hormiguero, las hormigas encuentran un camino que les permite recolectar el alimento de manera óptima. En el Algoritmo 1 tenemos el pseudocódigo del ACO. La primera vez que fue propuesto el algoritmo ACO fue en la tesis doctoral de Marco Dorigo [Dorigo, 1992], además tenemos el trabajo [Dorigo et al., 1996] se tiene más información sobre el algoritmo.

Algoritmo 1 Pseudocódigo del ACO

Entrada: $evaporacion\_feromona, N\_hormigas, N\_ciclos$
Salida: $mejor\_hormiga$

Se crean las hormigas $\vec{x}$ de forma aleatoria
Se crea la matriz de feromonas Mf
for 1 hasta N_ciclos **do**
    Se evalúan $\vec{x}$ y se guarda en $\vec{f}$
    Se actualiza Mf
    $mejor\_hormiga \leftarrow$ mejor hormiga de $\vec{x}$ y
    Se actualizan $\vec{x}$ tomando en cuenta Mf
end for
Se evalúan $\vec{x}$ y se guarda en $\vec{f}$
return $mejor\_hormiga$

Algoritmo de Pastoreo de Elefantes (EHO):

El EHO se propuso por primera vez en [Wang et al., 2015]. El algoritmo se basa en la organización de los grupos de los elefantes. Los elefantes se organizan en clanes con números fijos de elefantes, cada clan de elefantes tiene un líder denominado matriarca, la cual representa la mejor solución encontrada hasta el momento y además influye en las posiciones de los demás miembros del clan. Otro elemento que se identifica en el algoritmo son los elefantes macho adultos, los cuales son los que tienen la peor solución

y son reemplazados del grupo. La separación que existe entre la matriarca y los elefantes machos adultos son lo que permiten la diversificación del algoritmo. En el Algoritmo 2 encontramos el pseudocódigo del EHO.

Algoritmo 2 Pseudocódigo del EHO

Entrada: $influencia\_matriarca, N\_clanes, influencia\_clan, n\_ciclo$
Salida: *matriarca*

```
Se crean los elefantes x⃗ de forma aleatoria
for 1 hasta n_ciclos do
    Evaluación de los elefantes
    División de clanes
    El mejor elefante se asigna a matriarca
    Se actualizan los clanes
    Se identifican los elefantes machos adultos
    Se separan los elefantes machos adultos
end for
return matriarca
```

Algoritmo del Lobo Gris (GWO):

Para el caso del algoritmo del lobo gris, fue propuesto por primera vez en el trabajo de [Mirjalili et al., 2014]. El algoritmo simula el método de casa de las manadas de lobos. Las manadas de lobos están conformadas en promedio de cinco a doce lobos. En cada manada de lobos se compone por cuatro niveles jerárquicos. En los niveles jerárquicos de la manada encontramos alfa α, beta β, delta δ y omega ω. Cada lobo que pertenece a un nivel jerárquico tiene una función importante. Los α se integran por un macho y una hembra, son los lobos que toman las decisiones más importantes del rebaño, en ellos se almacena la mejor solución. El lobo considerado β puede ser macho o hembra es el que ayuda en la toma de decisiones y a disciplinar al rebaño, es considerado el mejor candidato para reemplazar a un α en el caso que muera o esté muy viejo, en él se guarda la segunda mejor solución. Los δ son los lobos que no encaja en ninguna otra de las jerarquías, en ella se almacena la tercera mejor solución. En el caso de los ω, son considerados como chivos expiatorios y se someten a la voluntad de los otros lobos, en

ellos se almacenan el resto de las posibles soluciones. En el Algoritmo 3 observamos el pseudocódigo del GWO.

Algoritmo 3 Pseudocódigo del GWO

Entrada: n_ciclo
Salida: $lobo_\alpha$

```
Se crean la manada de lobos x⃗ de forma aleatoria
Se evalúa x⃗
lobo_α ← mejor solución
lobo_β ← segunda mejor solución
lobo_δ ← tercera mejor solución
for 1 hasta n_ciclos do
    Actualizar x⃗
    Se evalúa x⃗
    Se actualiza lobo_α, lobo_β y lobo_δ
end for
return lobo_α
```

2.3.1.2. Algoritmos Evolutivos (AEs)

Los algoritmos que son clasificados como AEs [Araujo and Cervigón, 2009] se caracterizan por simular los procesos evolutivos de las especies y la selección natural. Los individuos que mejor se adaptan en el entorno son los que tienen una mayor probabilidad de sobrevivir al paso de las generaciones o de trasmitir su contenido genético a las próximas generaciones. Los procesos evolutivos que se enfrentan los algoritmos de manera general son: evaluación, selecciones, cruzamiento, mutación y paso de generación. Cada proceso evolutivo es importante para los algoritmos, la evaluación determina su fitness o su adaptación al entorno de cada individuo de la población. La selección se encarga de elegir aquellos individuos que participaran en los siguientes procesos, aquellos individuos que tengan un mejor fitness tienen una mayor probabilidad de ser elegidos. El cruzamiento es el proceso donde se comparte el contenido genético de los individuos seleccionados para así dar origen a una nueva población, este proceso se puede comparar con el proceso de reproducción de las especies. Para el proceso de mutación es solamente

un cambio pequeño de manera aleatorio en su código genético de un pequeño grupo de individuos, este cambio aleatorio se presenta en la población generada por el proceso de cruzamiento. El paso generacional es el proceso en el cual se decide qué población o que individuos pasan a la siguiente generación para repetir el proceso evolutivo. Para cada etapa mencionada anteriormente existen un conjunto de métodos diferentes. En el Algoritmo 4 podemos observar un pseudocódigo de los algoritmos evolutivos de manera general, para cada algoritmo específico se debe adaptar los procesos evolutivos.

El algoritmo más representativo en esta categoría es el algoritmo genético, pero existen otros algoritmos en esta categoría como la programación evolutiva y las estrategias evolutivas. La mayor diferencia que existe entre los algoritmos anteriores es la forma de como representan los individuos que conforman a la población. Otro algoritmo que pertenece a los algoritmos evolutivos son los algoritmos de estimación de distribución, mejor conocidos como Algoritmo de Estimación de la Distribución (EDA) (Estimation of Distribution Algorithms).

Algoritmo Genético (GA):

Los valores con los que se componen los individuos de la población pueden ser del tipo binario (0's y 1's) o números enteros Z.

Programación Evolutiva (PE):

Los individuos que conforman a la población son ternas, las cuales representa un autómata finito. Cada terna contiene los siguientes valores: el valor del estado actual, un símbolo del alfabeto utilizado y el valor de nuestro estado. Por lo general este tipo de algoritmo es utilizado para generar autómatas.

Estrategias Evolutivas (EE):

Los individuos de la población están conformados por números reales R. Otras de sus principales diferencias de los otros algoritmos evolutivos son su forma de selección, ya

que esta la hace de manera imparcial, el proceso de cruzamiento, a partir de un conjunto de n padres, se generan un conjunto de m hijos, n y m no necesariamente deben ser iguales. EL proceso de mutación se realiza a partir de una distribución gaussiana. La nueva generación se conforma por los mejores individuos ya sea de la población o los hijos.

Algoritmo 4 Pseudocódigo general de los AEs

Entrada: *n_generaciones*, *t_poblacion*,*probabilidad_cruce*,*probabilidad_mutacion*
Salida: *mejor_individuo*

1: Crear la población inicial $\vec{x}$ de forma aleatoria
2: Se evalúa $\vec{x}$
3: *mejor_individuo* $\leftarrow$ mejor individuo de $\vec{x}$
4: **for** 1 hasta *n_generaciones* **do**
5: Se seleccionan individuos de $\vec{x}$ y se guarda en $\overrightarrow{x_{select}}$
6: Se crea una nueva población a partir de $\overrightarrow{x_{select}}$ por un método de cruzamiento y se almacena en $\overrightarrow{x_{cruz}}$
7: Se muta la población $\overrightarrow{x_{cruz}}$
8: Se realiza el paso de generación con $\overrightarrow{x_{cruz}}$ y se almacena en $\vec{x}$
9: Se evalúa $\vec{x}$
10: *mejor_individuo* $\leftarrow$ mejor individuo de $\vec{x}$
11: **end for**
12: **return** *mejor_individuo*

2.4. Algoritmo de Estimación de la Distribución

Los EDAs son algoritmos metaheurísticos clasificados como AEs, su primera introducción fue en el artículo de [Mühlenbein and Paaß, 1996], pero fue hasta en el artículo de [Alonso et al., 2003] que se da una mayor explicación, además en este mismo artículo se habla de la clasificación según su modelo probabilístico. Los EDAs tienen importantes diferencias de los AEs. Las similitudes que existen entre los AEs y los EDAs son el uso de poblaciones de individuos que almacenan la solución al problema. El uso de un método de selección y en su defecto los métodos de reemplazo de padres. Los métodos que difieren en su totalidad entre los algoritmos son el método de cruzamiento y de

mutación contenidos en los AEs y son remplazados por los métodos de estimación de la distribución de probabilidad y la generación de individuos. Como podemos observar en la Figura 2.11, muestra un diagrama comparativo entre la estructura general de los GAs y los EDAs. Los métodos de cruzamiento y mutación del los GAs son reemplazados por los métodos de estimación de la distribución y muestreo en los EDAs, dichos métodos son los que se encargan de generar los nuevos individuos de la población.

Los métodos de estimación de la distribución de probabilidad y la generación de individuos son el pilar principal de los EDAs. Ya que en estos métodos el algoritmo genera a la nueva población con la que se va a continuar trabajando. Dependiendo del tipo de variable con la que se está trabajando, ya sea variable discreta o variable continua, se pueden clasificar los EDAs. Además, el tipo de dependencia que exista entre las variables que componen al problema. Los EDAs se clasifican como: univariado, bivariado y multivariado.

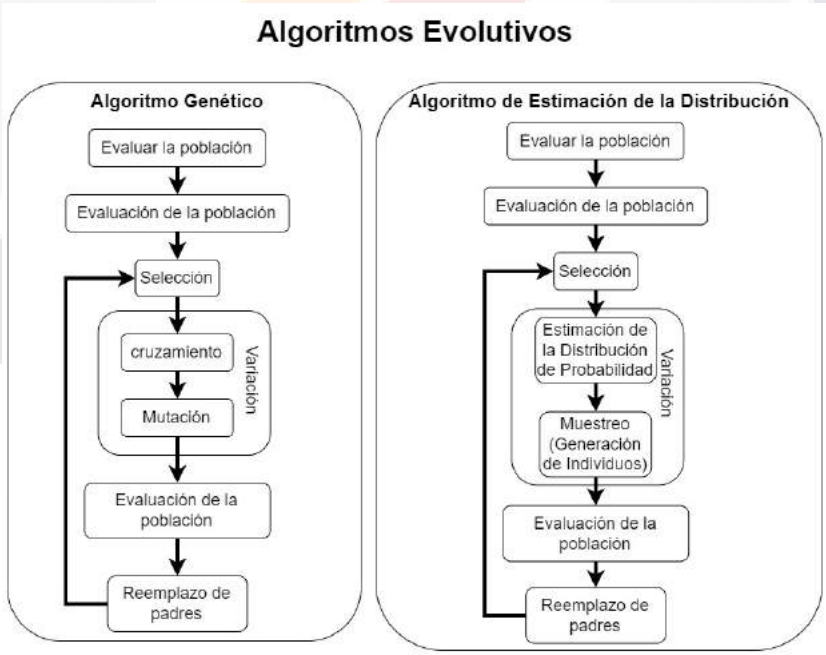


Figura 2.11: Diferencia entre la estructura general de los GAs y los EDAs. Autoría propia.

2.4.1. Clasificación según el modelo probabilístico

Dependiendo de la dependencia que se tiene en los modelos probabilísticos generados por los EDAs es el grupo o clasificación que se le asigne.

2.4.1.1. Dependencia univariada

Los algoritmos que pertenecen a esta categoría son aquellos en los cuales no existe una dependencia entre las variables, es decir, las variables son independientes entre sí y el valor que pueden tomar la variable es independiente. La distribución de probabilidad de un problema de n variables se factoriza como un producto de n distribuciones de probabilidad. La Ecuación 2.18 representa el modelo de probabilidad y en la Figura 2.12 observamos de manera gráfica la dependencia univariada. Los círculos representan variables del problema de optimización, pero entre las variables no existe ninguna dependencia.

$$p(X) = \prod_{i=1}^{n} p(x_i) \tag{2.18}$$

Algunos de los algoritmos más representativos de la dependencia univariada son:

- Algoritmo con Distribución Marginal Univariada (UMDA) (Univariate Marginal Distribution Algorithm).
- Aprendizaje Incremental Basado en Poblaciones (PBIL) (Population Based Incremental Learning).
- Algoritmo Genético Compacto (cGA) (Compact Genetic Algorithm).

2.4.1.2. Dependencia bivariada

Los algoritmos en los cuales existe una dependencia entre pares de variables son los que integran esta categoría. La distribución de probabilidad de los algoritmos de esta

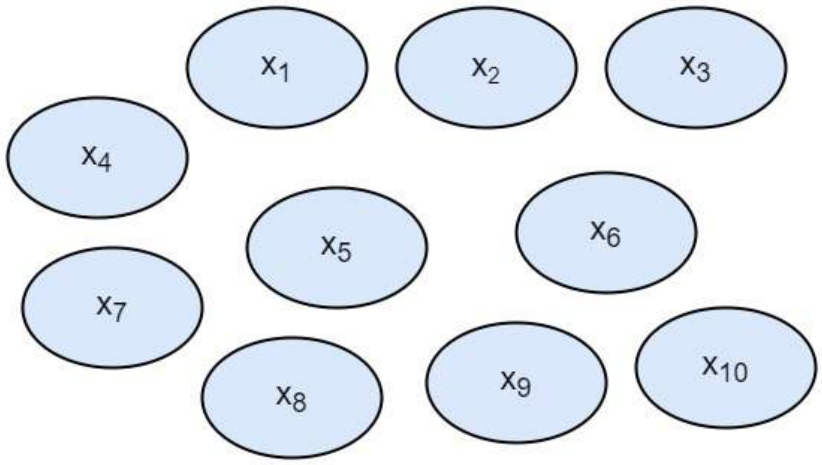


Figura 2.12: Dependencia univariada. Autoría propia.

categoría es bivariada. La Ecuación 2.19 es una representación del modelo de probabilidad. La Figura 2.13 es un ejemplo de dependencia de bivariada. La Figura 2.13a representación de la dependencia de variables del MIMIC. La Figura 2.13b representación de la dependencia de variables del COMIT. La Figura 2.13c representación de la dependencia de variables del BMDA.

$$p(X) = \prod_{i=1}^{n} p(x_{a_i}, x_{b_i}) \tag{2.19}$$

Los algoritmos que encontramos en esta categoría son:

- Algoritmos de Maximización de la Información para Clasificación (MIMIC) (Mutual Information Maximization for Input Clustering).
- Optimización Combinatoria con Árboles de Información Mutua (COMIT) (Combining Optimizers with Mutual Information Trees).
- Algoritmo con Distribución Marginal Bivariada (BMDA) (Bivariate Marginal Distribution Algorithm).

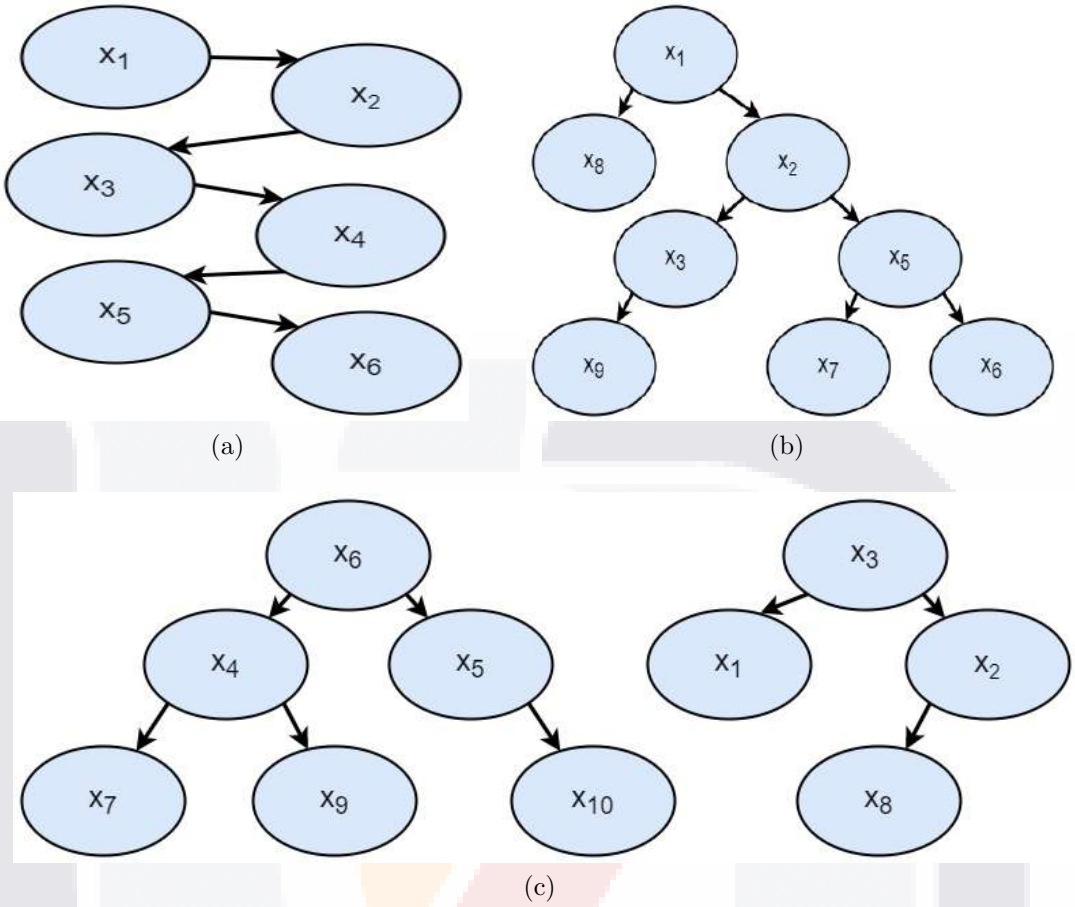


Figura 2.13: Dependencia bivariada. Autoría propia.

2.4.1.3. Dependencia multivariada

Los algoritmos que se encuentran en esta clasificación son los más complejos. Cualquier problema que tenga una dependencia por lo menos una cantidad de variables mayor a 2 es clasificado en esta categoría. La mayoría de los algoritmos clasificados en esta categoría usan redes Bayesianas. La Ecuación 2.20 es la representación del modelo de probabilidad. En la Figura 2.14 se observa un ejemplo de dependencia multivariada. La Figura 2.14a representación de la dependencia de variables del EcGA. La Figura 2.14b representación de la dependencia de variables del FDA. La Figura 2.14c representación de la dependencia de las variables del BOA.

$$p(x) = \prod_{i=1}^{n} \psi_i (X_{S_i}) \tag{2.20}$$

Algunos algoritmos clasificados en esta categoría son:

- Algoritmo Genético Compacto Extendido (EcGA) (Extended Compact Genetic Algorithm).
- Algoritmo de Distribución Factorizada (FDA) (Factorized Distribution Algorithm).
- Algoritmo de Optimización Bayesiano (BOA) (Bayesian Optimization Algorithm).

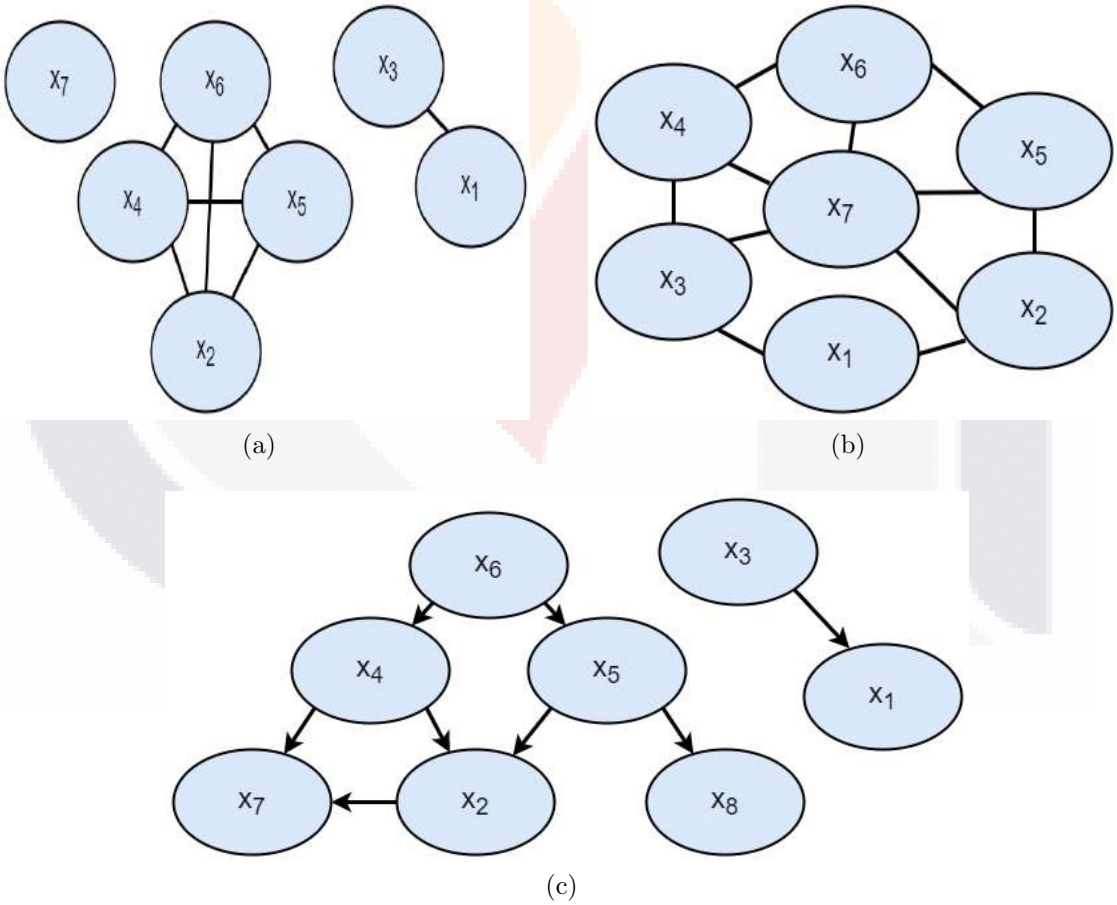


Figura 2.14: Dependencia Multivariada. Autoría propia.

2.4.2. Ejemplo de la implementación de un EDA

Consideremos que nuestro problema de optimización es el *One Max* que utiliza variables discretas. Tiene como objetivo llenar con puros $1s$ a $\vec{S} = \{s_i = \{0,1\} \,|i = 1,2,\ldots,n\}$ donde n es el número de elementos. Sea $D_g = \{d_k|k = 1,2,\ldots,m\}$ la matriz de tamaño mxn que almacena la población en la generación $g - esima$. $d_k = \{d_k^i|i = 1,2,\ldots,n\}$ representa al $k - esimo$ individuo de la población. $d_k^i = \{0,1\}$ representa el valor del $k - esimo$ individuo en la $i - esima$ columna. Sea x_i la $i - esima$ columna de D_g. La función objetivo del problema la vemos en la Ecuación 2.21.

$$F_{One\ Max}\left(\vec{S}\right) = \sum_{i=1}^{n} s_i \tag{2.21}$$

La Tabla 2.6 representa la primera generación D_0 de nuestro ejemplo. La distribución de probabilidad de D_0 es de 0.5, es decir, $p_0\left(\vec{x}\right) = \prod_{i=1}^{n}\left(x_i|D_0\right) = p_0\left(x_i = 1\right) = 0.5$. A partir de D_0 se selecciona una cantidad de d_k, para nuestro ejemplo seleccionaremos la mitad de la población, es decir $\frac{m}{2}$, para generar D_0^S, la cual podemos observar en la Tabla 2.7. Los individuos seleccionados son aquellos que tienen el mejor fitness de la población, a este método de selección se conoce como selección por truncamiento.

Ahora con D_0^S se obtiene $p_1\left(\vec{x}\right) = p_1\left(x_1, x_2, \ldots, x_n\right) = \prod_{i=1}^{n} p_1\left(x_i|D_0^S\right)$ que consiste en calcular la frecuencia con la que $d_k^i = 1, \forall k = 1,2,\ldots,m$ en x_i, observamos la Tabla 2.8 que contiene los valores de $p_1\left(\vec{x}\right)$. Después con $p_1\left(\vec{x}\right)$ generamos la nueva población que será D_1, que la encontramos en la Tabla 2.9. Los 3 pasos anteriores los repetimos hasta cumplir la condición de paro.

En otras palabras, la implementación de los EDAs consiste en los siguientes pasos:

1. Seleccionar una cantidad de individuos.
2. Estimar la distribución de la probabilidad de los individuos seleccionados.
3. Muestrear a la nueva población de la distribución de la probabilidad.

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	$F_{One\ Max}(\vec{x})$
1	1	0	1	0	1	0	0	1	4
2	1	1	1	0	0	0	1	0	4
3	1	1	0	1	0	1	1	1	6
4	1	1	0	0	0	1	0	1	4
5	1	0	1	0	0	0	0	1	3
6	0	1	0	0	1	1	1	0	4
7	1	0	1	1	1	0	0	1	5
8	0	1	0	1	0	0	0	0	2
9	0	1	0	1	1	0	1	1	5
10	0	1	0	0	1	1	0	0	3

Tabla 2.6: Primera generación D_0. Al ser la primera generación, los individuos se crean de forma aleatoria, ya que $p_0(\vec{x_i}) = 0.5, \forall i = 1, 2, \ldots, n$. Autoría propia.

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	$F_{One\ Max}(\vec{x})$
3	1	1	0	1	0	1	1	1	6
7	1	0	1	1	1	0	0	1	5
9	0	1	0	1	1	0	1	1	5
1	1	0	1	0	1	0	0	1	4
6	0	1	0	0	1	1	1	0	4

Tabla 2.7: Individuos de D_0 seleccionados por truncamiento. Los individuos con el mejor fitness son seleccionados por este método. Autoría propia.

2.5. Análisis de trabajos semejantes

En la siguiente sección hablaremos de los artículos encontrados en la literatura que abordan el descubrimiento de motivos tanto en nucleótidos como en aminoácidos, teniendo un mayor enfoque en los artículos que aborden la búsqueda de motivos en aminoácidos, especialmente donde la búsqueda la realicen mediante métodos heurísticos y metaheurísticos. Se toma como algo excepcional aquellos trabajos que realicen la búsqueda de motivos considerando más de una función objetivo, es decir sean mul-

i	1	2	3	4	5	6	7	8
$p_1\left(x_i \mid D_0^S\right)$	0.6	0.6	0.4	0.6	0.8	0.4	0.6	0.8

Tabla 2.8: Distribución de probabilidad de D_0^S. Obtenemos $p_1(x_i|D_0^S) = \frac{\sum_{k=1}^{m} d_k^i}{m}, \forall i = 1, 2, \ldots, n$. Autoría propia.

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	$F_{One\ Max}(\vec{x})$
1	1	1	1	0	1	0	0	1	5
2	1	1	1	0	1	0	1	1	6
3	0	1	0	1	0	1	1	1	5
4	1	1	0	1	1	0	1	1	6
5	1	1	1	1	1	0	0	1	6
6	0	1	0	0	1	1	1	0	4
7	1	1	1	1	1	1	0	1	7
8	0	1	0	1	0	0	1	0	3
9	1	0	1	0	1	1	0	1	5
10	1	1	0	1	1	1	1	1	7

Tabla 2.9: Segunda generación D_1. Los individuos se crean a partir de $p_1(\vec{x})$. Autoría propia.

tiobjetivo, ya que esto se aproxima en gran medida a nuestro trabajo. La fecha de la publicación de dichos artículos es importante, por lo que se le dará prioridad a los artículos nuevos (2023, 2022, 2021, 2020, 2019 y 2018). Aunque se revisaron algunos artículos relevantes un poco más viejos.

- Podemos encontrar en [Ashraf and Shafi, 2020] que trata de la búsqueda de motivos en nucleótidos y el artículo fue publicado en el 2020, en el artículo no habla de ninguna implementación de alguna metaheurística.

- En [Sarkar et al., 2021] encontramos un artículo sobre proteínas donde se propone una forma 3-D de las proteínas a partir de motivos, donde usan como función objetivo los ángulos que generan las proteínas de estudio, el artículo usa proteínas y no usa algún método metaheurístico.

- En [Lones and Tyrrell, 2005] encontramos la búsqueda de secuencias tanto en secuencias de nucleótidos como de aminoácidos, lo especial de este artículo es que habla sobre los algoritmos y computación evolutiva, en ella habla sobre su aplicación y describe de una manera concreta la parte de motivos y secuencias, a pesar de que ya tiene años la publicación, el aporte informativo es relevante.

- Como podemos leer en [Wang and Scott, 2005] nos habla de la búsqueda e implementación de un nuevo kernel para tener una mejor clasificación de los motivos encontrados en proteínas, describe la estructura de la búsqueda de motivos, la forma en que se clasificó y la función objetivo que utilizo para la clasificación.

- En [Sheth and Kim, 2005] nos muestra una metodología de siete secciones para la búsqueda de motivos de proteínas a partir de un clúster de subsecuencias, la primera parte consiste en seleccionar las secuencias con las que se va a trabajar, la segunda parte nos explica el proceso de como selecciono el conjunto de subsecuencias, para la tercera parte menciona la forma de como construyo el clúster, parta la cuarta parte habla de cómo busca y extrae las regiones conservadas, para la quinta parte habla sobre la extinción y unión de las regiones conservadas, en la sexta parte habla de cómo califico las regiones conservadas que se encontró y en la séptima parte nos menciona el método que uso para filtrar y seleccionar los motivos que encontró.

- En [Cordero et al., 2009] el artículo nos habla sobre la implementación de un framework para la identificación de motivos, en el cual realiza la utilización de una técnica de agrupación restringida para la localización de motivos entre grupos de proteínas. El nombre del framework es *de novo*.

- En [Li et al., 2010] habla sobre la búsqueda de motivos en nucleótidos, donde ellos desarrollan un método para la búsqueda y localización de los motivos, en el artículo describe de una manera detallada y formal la representación de la información, la función objetivo y la forma en que trabaja el algoritmo. Realiza una comparación entre los algoritmos existentes en la literatura con su método.

- En [Álvarez and Rodríguez, 2013] está enfocada en dar a conocer los EDAs con la variante de más de 1 objetivo, es decir, multiobjetivo, da una introducción

concreta y clara sobre los EDAs y de las partes que lo componen.

- En [Li et al., 2008] habla de la búsqueda de motivos con nucleótidos aplicando EDAs, en el artículo el autor describe de manera formal la forma en que se aborda el problema la modelación del problema y la solución, entre otros factores.

- En [Davey et al., 2010] explica sobre la búsqueda de motivos en proteínas mediante el cálculo de tres valores de probabilidad, el primer valor es p_{1+} que representa la probabilidad de tener 1 o más ocurrencias de un motivo en una proteína, el segundo valor es p que representa la probabilidad en un conjunto de datos de n proteínas de que un motivo dado ocurra por casualidad y el tercer valor es Sig que representa la probabilidad de que cualquier motivo alcance p o menos por casualidad, la explicación, demostración y utilización de cada uno de estos valores de probabilidad son explicados en el artículo.

- Para el caso de [Glasgow, 1998] es un documento donde se da de alta un proyecto que consiste en la clasificación de proteínas basándose en su estructura, utilizando un conjunto de elementos que se tiene en la literatura, esto con la intención de poder tener más elementos para poder clasificar las proteínas e identificar motivos en las proteínas.

- En [González Álvarez et al., 2015] realiza una búsqueda de motivos en nucleótidos multiobjetivo, lo mejor del artículo es su contenido informativo, donde se puede encontrar definiciones de conceptos, el planteamiento del problema, la modelación del problema, entre otros contenidos.

- En [González Álvarez and Vega Rodríguez, 2013] realiza todo un estudio sobre la escalabilidad que puede tener la resolución del problema de búsqueda de motivos, además que el artículo contiene una descripción a detalle del problema de búsqueda de motivos, además de que hace el análisis de la resolución de este problema

aplicando diferentes algoritmos.

- En [Calvet et al., 2022] hace un análisis amplio de los problemas bioinformáticos que se tienen en la actualidad y describe de una manera concreta en que consiste cada uno de los problemas, además hace un análisis sobre las metaheurísticas que se pueden implementar o utilizar, además nos solo se enfoca en un tipo de individuo (aminoácidos o nucleótidos) si no que utiliza las dos opciones.

- En [Jordán and Jordán, 2015] nos habla de la aplicación de los EDAs aplicado en la búsqueda de motivos en nucleótidos, en dicho artículo podemos encontrar el modelado del problema con su función objetivo, la representación de la solución y el ejemplo de cómo se ven las secuencias.

- En [Shao et al., 2009] trabaja la búsqueda de motivos en nucleótidos, pero en ella utiliza un método híbrido, en el cual combina la búsqueda tabú y la optimización por eliminación de bacterias, en el documento describe el algoritmo de estos métodos, su función objetivo y la forma en que este algoritmo trabaja.

- Para [Lones and Tyrrell, 2007] describe que, a partir de un clúster de secuencias de nucleótidos, es entrenado el algoritmo y en las próximas secuencias de nucleótidos enfoca la búsqueda de motivos en las zonas donde encontró en el clúster mayor conservación, describe en que consiste la búsqueda de motivos y como puede ser abordado por un algoritmo evolutivo, muestra algunos algoritmos que se utilizaron, además las data bases que se utilizaron.

- Para [Jackups and Liang, 2010] se realizaron búsqueda de secuencias pequeñas de aminoácidos en las cuales se buscarán que conformaran un motivo y estas mismas secuencias se buscaron en otras proteínas para tener una relación en función de la secuencia u otro factor, además menciona el método que se utilizó para realizar

dicha búsqueda, el análisis de la secuencia encontrada en otras secuencias y los modelos utilizados.

- En [Czeizler et al., 2017] nos habla de la búsqueda de motivos en proteínas utilizando la teoría de grafos, además nos da una comparativa de este método con otros dos métodos que existen en la literatura comparando el tiempo utilizado en cada uno de ellos, en el artículo nos habla de la manera de modelación del problema y la representación de la solución, en la cual es de suma importancia, ya que este tipo de implementación es poco común.

- Como vemos en [Saha et al., 2019] realiza una búsqueda de motivos en proteínas con la intención de asociar una función determinada del segmento de aminoácidos identificado, esto segmentando la secuencia o proteína en pequeñas zonas de interés, en el artículo encontramos conceptos de grafos, además de ejemplos y representaciones, la modelación del problema y la función con la que va a trabajar.

- En [Semwal et al., 2022] propone un nuevo método para la búsqueda de motivos en proteínas, además de que realiza la comparación con 2 métodos existentes en la literatura, en el artículo describe el problema y modela la función objetivo, muestra un algoritmo, el nuevo método propuesto está basado en el algoritmo de clasificación de K-means, donde una de las variables que afecta directamente los resultados es el tamaño del vecindario.

Capítulo 3

Metodología para el descubrimiento de motivos

En este capítulo se abordan las etapas que se realizaron para la consolidación de la búsqueda de motivos altamente conservados mediante una metaheurística multiobjetivo. Cada etapa abordada corresponde una Sección del capítulo.

En la Sección 3.1 se define el problema de investigación, el objetivo general, los objetivos específicos del problema de investigación, así como las preguntas de investigación y se comienza la revisión al estado del arte.

Continuando con la Sección 3.2 se detalla el proceso que se siguió para la selección de una herramienta para la solución del problema, para este caso particular se decidió utilizar una metaheurística.

Para la Sección 3.3 se indica la información que necesita MATEDA para trabajar y el proceso para obtener esta información. Además, se habla todos los pasos que se realizaron para la utilización de MATEDA, desde su descarga, configuración, creación de módulos, casos especiales de la información que necesita para la búsqueda de motivos y la forma de ejecución.

En la Sección 3.4 se habla sobre los experimentos que se realizaron, se habla de los procesos que sufren cada $\vec{SC}^i$ para ser utilizad por MATEDA, las variables utilizadas así como el valor que se les asigna y el lugar donde se guardan los resultados obtenidos.

Para concluir el capítulo se tiene la Sección 3.5 se mencionan los resultados obtenidos por MATEDA, se menciona los archivos generados por la herramienta, la información

que contiene cada archivo y la extensión del mismo, además se menciona el número de archivos generados en relación con el número de experimentos que se realizaron.

En la Figura 3.1 representa las actividades realizadas durante la metodología para su consolidación.

Figura 3.1: Metodología para el descubrimiento de motivos. Autoría propia.

3.1. Análisis y definición del problema de investigación

La primera actividad que se realizó fue la recopilación y análisis de artículos, documentos, memorias de congresos, etc., que tienen relación directa o indirectamente con la identificación de motivos en proteínas. La búsqueda se realizó por medio de diferentes medios digitales, de los cuales destacan navegadores de internet especializados, por ejemplo: Google Académico, bibliotecas especializadas como: Springer Link, ACM, IEEE, etc., repositorios bibliográficos de la UAA y libros especializados de Bioquímica, tales como: [Terfloth, 2009], [Mckee and MacKee, 2014], [Voet et al., 2007], entre otros.

Al término de la recopilación de la información se realizó el análisis de la misma. Se

clasificó la información analizada en diferentes categorías como: información importante, información complementaría e información sin relevancia.

La actividad siguiente que se realizó fue la formulación del problema de investigación. Se comenzó con la formulación del título del trabajo de investigación. Partiendo del título del trabajo, se continuó con la búsqueda de antecedentes del problema de investigación, la elaboración de la justificación, el planteamiento del objetivo general con sus respectivos objetivos específicos y se generaron algunas preguntas de investigación.

Retomando la información obtenida de la primera actividad, se creó una estructura general del marco teórico considerando el título, el planteamiento de investigación, la justificación, los objetivos y las preguntas de investigación. Además, se reunió información para aquellos conceptos necesarios para el entendimiento del trabajo y que presentaron alguno de los siguientes inconvenientes: no se tenía información, era escasa, estaba incompleta, era información muy especializada o técnica.

De la información recabada durante la conformación del marco teórico, se encontró que dicho problema es abordado mediante algoritmos metaheurísticos, además se encontró diversas formas para definir el problema de investigación.

3.1.1. Definición de problema de optimización

Se definió el problema de investigación de la siguiente manera: Encontrar una secuencia de aminoácidos $\vec{R} = \{r_i \in B | i = 1, 2, \ldots, L\}$ que maximicen su compatibilidad por propiedades fisicoquímicas con una secuencia de aminoácidos conservada $\vec{SC} = \{sc_i \in B | i = 1, 2, \ldots, L\}$ que se obtiene de los mejores aciertos bidireccionales. Donde L es la longitud de la secuencia. La secuencia conservada se refiere a una secuencia altamente homóloga obtenida de un conjunto de proteínas. La función objetivo del problema de investigación está expresada por la Ecuación 3.1 y las Ecuaciones 3.2, 3.3 y 3.4 son el desglose de cada elemento que compone la Ecuación 3.1.

$$F\left(\vec{R},\vec{SC}\right) = \text{máx}\left[FCC\left(\vec{R},\vec{SC}\right), FCP\left(\vec{R},\vec{SC}\right), FCH\left(\vec{R},\vec{SC}\right)\right] \tag{3.1}$$

$$FCC\left(\vec{R},\vec{SC}\right) = \sum_{i=1}^{L} MCC\left(r_i, sc_i\right) \tag{3.2}$$

$$FCP\left(\vec{R},\vec{SC}\right) = \sum_{i=1}^{L} MCP\left(r_i, sc_i\right) \tag{3.3}$$

$$FCH\left(\vec{R},\vec{SC}\right) = \sum_{i=1}^{L} MCH\left(r_i, sc_i\right) \tag{3.4}$$

Donde:

- $\vec{R}$: Secuencia solución
- $\vec{SC}$: Secuencia conservada de los mejores aciertos bidireccionales
- FCC: Función de Compatibilidad por Carga
- MCC: Matriz de Compatibilidad por Carga
- $MCC\left(a,b\right)$: Compatibilidad por carga del aminoácido a con el aminoácido b
- FCP: Función de Compatibilidad por Peso
- MCP: Matriz de Compatibilidad por Peso
- $MCP\left(a,b\right)$: Compatibilidad por peso del aminoácido a con el aminoácido b
- FCH: Función de Compatibilidad por Hidropaticidad
- MCH: Matriz de Compatibilidad por Hidropaticidad
- $MCH\left(a,b\right)$: Compatibilidad por hidropaticidad del aminoácido a con el aminoácido b

3.2. Selección de herramienta

Partiendo de la información encontrada en la Sección 3.1 sobre la forma de abordar el problema de búsqueda de motivos. La primera actividad que se realizó en esta etapa de la metodología fue la búsqueda, recolección y la síntesis de la información de los diferentes algoritmos metaheurísticos utilizados para abordar el problema. Con la información resumida, se continuó con la actividad de analizar los diferentes algoritmos para seleccionar el algoritmo con el cual se aborda el problema de investigación.

Después de que se analizaron los algoritmos encontrados durante la investigación, se decidió trabajar con los EDAs. Definido el algoritmo con el que se trabajó, la siguiente actividad que se hizo fue decidir si se programa el algoritmo desde cero o se utiliza alguna librería que ya tenga implementado el algoritmo. Se optó por utilizar una librería. Por tal razón. La siguiente actividad realizada fue la búsqueda e identificación de una librería que tenga los EDAs implementados.

Se investigó sobre librerías en las cuales se tenga implementada los EDAs, se decidió la utilización de MATEDA [Santana et al., 2009] [Santana et al., 2010] [Irurozki et al., 2018]. MATEDA es un ToolBox desarrollado en *M* para el Entorno de Desarrollo Integrado (IDE) conocido como MATLAB.

Después de que se seleccionó la herramienta con la que se va a trabajar el problema de investigación, se continúa con la etapa de diseño e implementación de una metaheurística para el descubrimiento de motivos.

3.3. Diseño e implementación de una metaheurística para el descubrimiento de motivos

Esta Sección está conformada por 2 actividades fundamentales que se realizaron para el correcto funcionamiento de MATEDA. Partimos con las actividades realizadas en la Sección 3.3.1, para obtener toda la información necesaria para trabajar con MATEDA. Es decir, el preprocesamiento de la información, mientras que en la Sección 3.3.2 se aborda la descarga, configuración y creación de módulos que necesitaba MATEDA para su correcto funcionamiento.

3.3.1. Preprocesamiento de la información

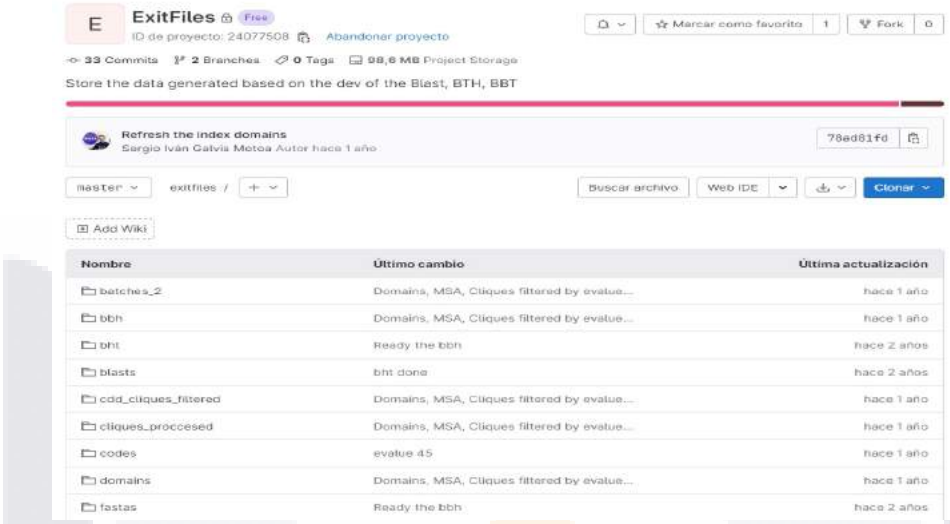
La información que necesitaba MATEDA para trabajar con el problema de optimización es:

- Conjunto de Secuencias Conservadas $C\vec{S}C$
- Matrices de compatibilidad MCC, MCP y MCH

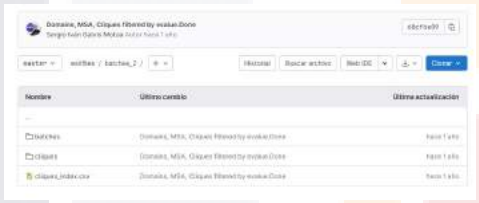
3.3.1.1. Obtención de $C\vec{S}C$

Para la obtención de $C\vec{S}C$, la primera actividad que se realizó es la obtención del Data Set de proteínas con los mejores aciertos bidireccionales del repositorio en GitLab de Sergio Iván Galvis Motoa propuesto en [Galvis Motoa et al., 2021] y [Motoa, 2022]. En la Figura 3.2 podemos observar las ventanas que se siguieron para la descarga del Data Set elegido. Estando en la página principal del repositorio de GitLab, Figura 3.2a, en esa pantalla se da clic en la carpeta **batches_2** para que se muestre la ventana de la Figura 3.2b, en esta pantalla se da clic en la carpeta **batches** para que se muestre la pantalla de la Figura 3.2c, en esta pantalla se busca la secuencia **cli-**

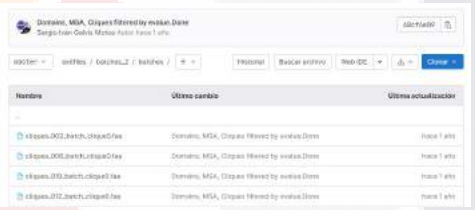
ques_066_batch_clique0.faa, la Figura 3.2d muestra la secuencia y descargamos el archivo.



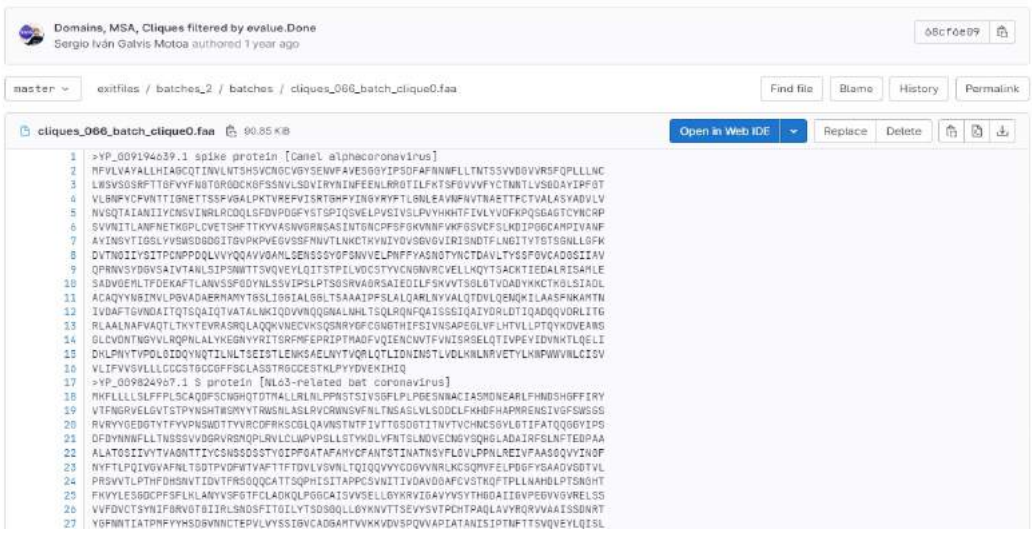
(a) Página principal del repositorio de GitLab.



(b) Carpeta **batches_2**.



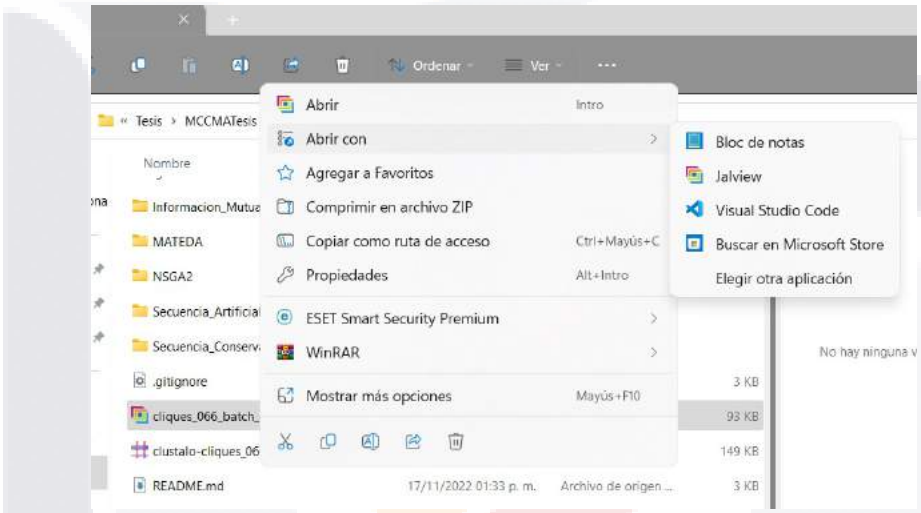
(c) Carpeta **batches**.



(d) Archivo **cliques_066_batch_clique0.faa**.

Figura 3.2: Descarga del Data Set. Autoría propia.

El archivo **cliques_066_batch_clique0.faa** contiene un pequeño error por lo cual se realizó la modificación del mismo, en la Figura 3.3 se pueden observar los pasos que se siguieron para corregir el error del archivo. Se abre el archivo **cliques_066_batch_clique0 .faa** en un bloc de notas o con un IDE, Figura 3.3a, se dirigió a la línea 504 del archivo y en la parte donde se tiene el símbolo de mayor que >, Figura 3.3b, se da un salto de línea antes del >, Figura 3.3c y se guarda el cambio realizado.



(a) Abriendo el archivo **cliques_066_batch_clique0.faa** con un bloc de texto o IDE.

(b) Archivo **cliques_066_batch_clique0.faa** entre las líneas 501-506.

```
501  SIQAIYDRLDSIQADQQVDRLITGRLAALNAFVSQVLNKYTEVRSSRRLAQQKINECVKSQSNRYGFCGNGTHIFSIVNS
502  APDGLLFLHTVLLPTDYKNVKAWSGICVDGIYGYVLRQPNLVLYSDNGVFRVTSRVMFQPRLPVLSDFVQIYNCNVTFVN
503  ISRVELHTVIPDYVDVNKTLQEFAQNLPKYVKPNFDLTPFNLTYLNLSSELKQLEAKTASLFQTTVKLQGLIDQINSTYV
504  DLKLLNRFENYIKWPWWVWLIISVVFVVLLSLLVFCCLSTGCCGCCNCLTSSMRGCCDCGSTKLPYYEFEKVHVQ
505  >NP_045300.1 spike protein [Murine hepatitis virus]
506  MLFVFILFLPSCLGYIGDFRCIQLVNSNGANVSAPSISTETVEVSQGLGTYYVLDRVYLNATLLLTGYYPVDGSKFRNLA
507  LTGTNSVSLSWFQPPYLSQFNDGIFAKVQNLKTSTPSGATAYFPTIVIGSLFGYTSYTVVIEPYNGVIMASVCQYTICLL
```

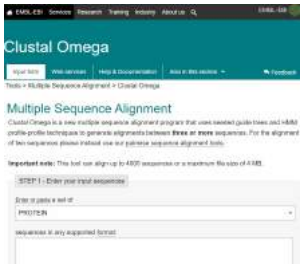
(c) Separación de la línea 504 antes del símbolo >.

Figura 3.3: Modificación del Data Set. Autoría propia.

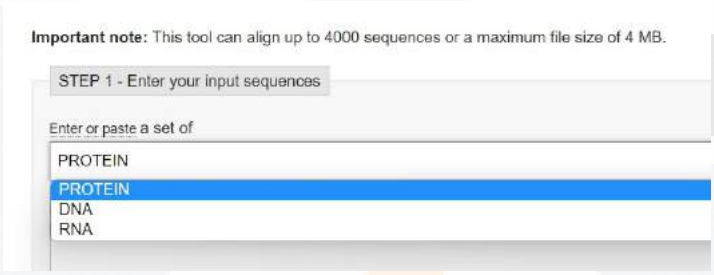
La siguiente actividad para obtener $C\vec{S}C$, es someter el Data Set a un AMS, el cual se realizó con el software *Clustal*Ω [Sievers et al., 2011] [Sievers and Higgins, 2018] en

su versión web, en la Figura 3.4 muestra de manera generar los pasos realizados. Lo primero que se hizo fue buscar en un navegador web la palabra clustal o, ver Figura 3.4a, se ingresó a la página de *Clustal*Ω, ver Figura 3.4b. Dentro de la página, lo primero que se tiene que hacer es elegir el tipo de secuencias, en nuestro caso son secuencias de proteínas e ingresamos las secuencias que se procesaran, en nuestro caso es el archivo **cliques_066_batch_clique0.faa**, ver Figuras 3.4c y 3.4d. El siguiente paso que marca la página es la selección de formato de salida del AMS, para este caso es el formato Pearson/FASTA, ver Figura 3.4e, para finalizar la página pregunta que si desean que se les notifique por correo electrónico cuando termine el AMS, ver Figura 3.4f, después de que se termine el AMS la página web manda un correo con una liga para descargar el AMS, a partir de este momento se considera como Data Set el AMS. La página web solo conserva 7 días el AMS, después de este tiempo se debe hacer nuevamente el proceso.

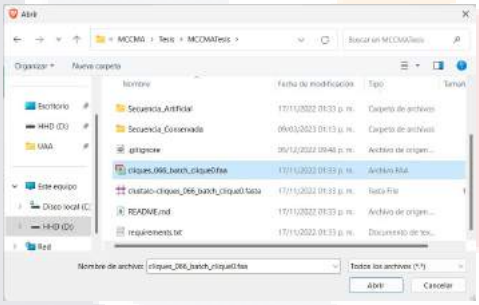
(a) Abrimos el navegador web y buscamos clustal o.



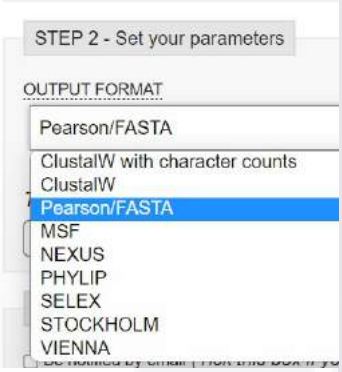
(b) Página principal de *ClustalΩ*



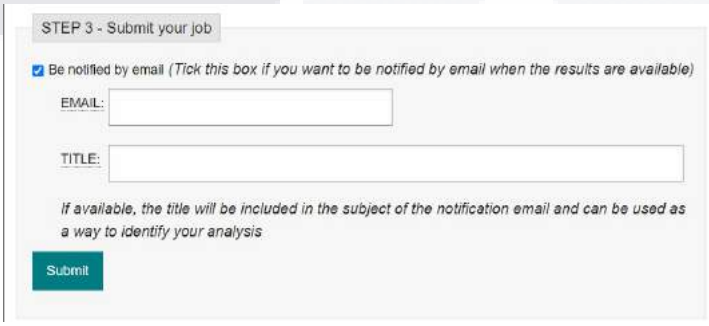
(c) Se selecciona el tipo de secuencias de entrada



(d) Se selecciona el archivo **cliques_066_batch_clique0.faa**



(e) Se selecciona el formato de salida del archivo AMS

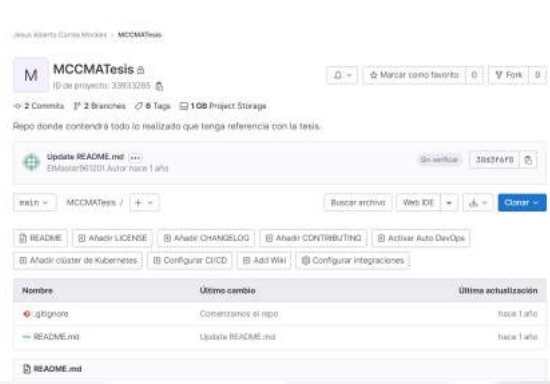


(f) Se selecciona la opción de notificación por correo y se llena los datos requeridos.

Figura 3.4: Alineamiento Múltiple de Secuencias. Pasos que se siguieron para hacer un AMS del archivo **cliques_066_batch_clique0.faa**. Autoría propia.

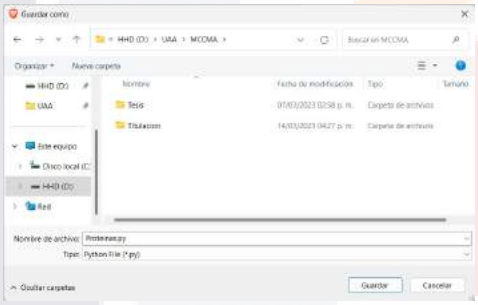
Después de que se tiene el AMS, se obtuvo un conjunto de secuencias altamente conservadas utilizando un módulo de la metodología descrita en [Correa Morales, 2020]. La utilización del módulo mencionado se realizó en tres etapas principales. La primera etapa se realizó la descarga del módulo de la metodología, ver Figura 3.5. La segunda etapa se realizaron las modificaciones o configuraciones que se tienen que hacer al módulo para su funcionamiento con nuestro Data Set, ver Figura 3.6. La tercera etapa es la ejecución del módulo, ver Figura 3.7.

Para la primera etapa se dirigen al repositorio de GitLab de CorreaJesus, Figura 3.5a, en el repositorio se cambia a la rama **develop**, Figura 3.5b, después se accede a la carpeta **Secuencia_Conservada**, Figura 3.5c, de la carpeta **Secuencia_Conservada** se descarga el archivo **Proteinas.py** y **Secuencia_Conservada.py**, Figura 3.5d y 3.5e, dichos archivos corresponden al módulo de la metodología que se necesitan.

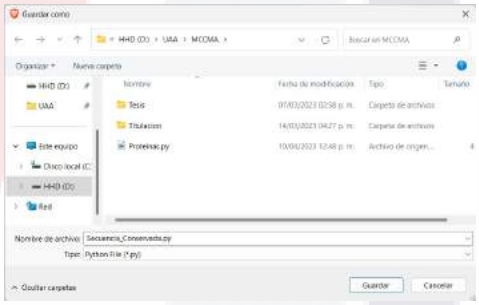


(a) Repositorio de GitLab de CorreaJesus. (b) Cambio de rama del repositorio

(c) Carpeta **Secuencia_Conservada**.



(d) Descarga del archivo **Proteinas.py**.



(e) Descarga del archivo **Secuencia_Conservada.py**.

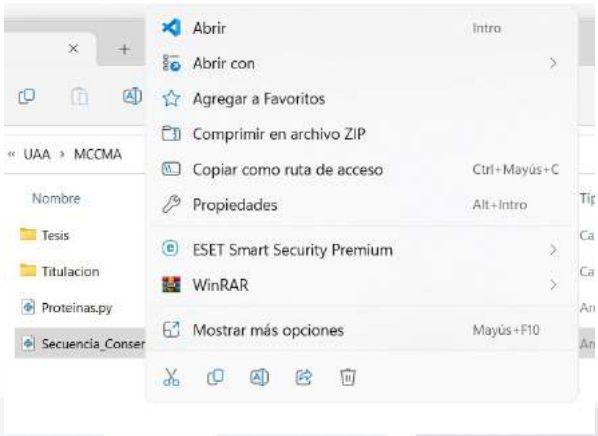
Figura 3.5: Descarga de un módulo de la metodología. Autoría propia.

Teniendo el módulo para la obtención de secuencias conservadas de un AMS, se comienza la segunda etapa, que consiste en realizar los cambios pertinentes al módulo para que funcione con el Data Set. En la Figura 3.6 se observan los pasos necesarios para el funcionamiento del módulo. Se abre el archivo **Secuencia_Conservada.py** con un procesador de texto o un IDE, para este caso se abrió con VSCode, Figura 3.6a, se dirige al bloque de código que comprende desde las líneas 168 - 177, Figura 3.6b,

el bloque de código corresponde a la asignación de valores de las variables del módulo. Cada variable modifica el comportamiento del módulo, solo se modificarán las variables:

- **nombre_Data_Set**: Nombre del Data Set
- **ruta**: Nombre de la carpeta donde se guardan las secuencias
- **nombre_archivo**: Nombre del archivo donde se guardan las secuencias
- **tamano_secuencias**: Longitud de las secuencias a buscar
- **S**: Se indica que se busquen secuencias de longitud fija
- **S2**: Se indica que se busquen secuencias de longitud variada
- **tam_min**: Longitud mínima de las secuencias con longitud variada
- **tam_max**: Longitud máxima de las secuencias con longitud variada
- **block**: Tamaño del bloque de búsqueda de las secuencias con longitud variada

Para este caso se utilizaron los valores que se pueden observar en la Figura 3.6c, después de que se realizaron las modificaciones correspondientes, se guarda el archivo y está listo para utilizarse.



(a) Se abre el archivo **Secuencia_Conservada.py**.

```
168  nombre_Data_Set = ""
169  sec_con = SecuenciaConservada(nombre_Data_Set)
170  ruta = ""
171  nombre_archivo = ""
172  tamano_secuencias = []
173  S = True
174  S2 = True
175  tam_min = 10
176  tam_max = 50
177  block = 5
```

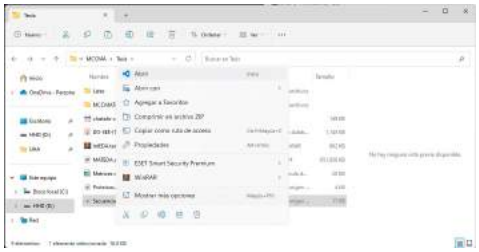
(b) Código entre las líneas 168-177

```
168  nombre_Data_Set = "clustalo-cliques_066_batch_clique0.fasta"
169  sec_con = SecuenciaConservada(nombre_Data_Set)
170  ruta = "Resultados"
171  nombre_archivo = "Secuencia_Conservada"
172  tamano_secuencias = [10]
173  S = True
174  S2 = True
175  tam_min = 10
176  tam_max = 50
```

(c) Valores que se asignaron a las variables

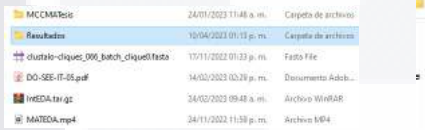
Figura 3.6: Configuración del módulo de la metodología. Autoría propia.

La tercera etapa es la forma en que se puede utilizar el módulo que se descargó y se configuró depende totalmente del usuario, ya que se puede usar desde un IDE o desde una consola de comandos, en la Figura 3.7 se tienen los paso a seguir y donde se deben buscar los resultados, para este caso se utilizó por medio de un IDE, se abre el módulo **Secuencia_Conservada.py** en el IDE, Figura 3.7a y únicamente se da clic al botón de ejecución y el programa se ejecuta, Figura 3.7b, se revisa la carpeta donde se guardó el módulo, Figura 3.7c, se ingresa a la carpeta con el nombre del Data Set, Figura 3.7d, ya que en esta se guardaron los resultados generados por el módulo, Figura 3.7e, el archivo que importa es el que tiene el nombre de **Secuencia_Conservada_[10, 50].txt**, ya que en este archivo se guardaron los resultados generados por el módulo de forma compacta, Figura 3.7f. Para que el módulo funcione correctamente, el Data Set debe estar en la misma carpeta.

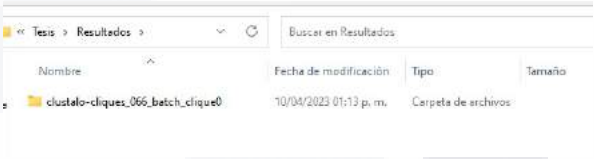


(a) Se abre el archivo **Secuencia_Conservada.py** con VSCode.

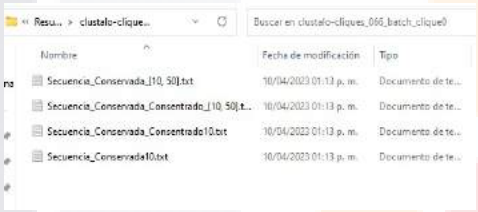
(b) Se ejecuta el archivo.



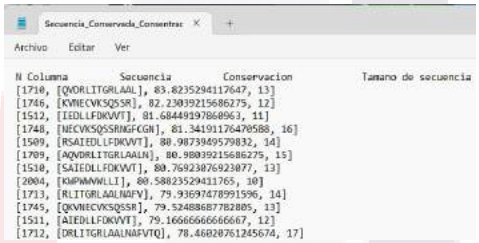
(c) Carpeta donde se encuentra el archivo.



(d) Carpeta con el nombre del Data Set.



(e) Carpeta con los resultados obtenidos.



(f) Archivo **Secuencia_Conservada_[10, 50].txt** con los resultados de forma compacta.

Figura 3.7: Ejecución del módulo de la metodología. Autoría propia.

El archivo **Secuencia_Conservada_[10, 50].txt** contiene los resultados de la ejecución del módulo de forma compacta. Los resultados obtenidos se encuentran en un tipo de tabla compuesta por 4 columnas. En la primera fila contiene los nombres de las columnas que componen la tabla. Los datos están contenidos entre un par de corchetes "[ ]" y se encuentran separados por comas ",". La primera columna contiene el número de la columna del AMS que se tomó de referencia para obtener la $\vec{SC}$, en la segunda columna se contiene la $\vec{SC}$, la cual se encuentra dentro de corchetes "[ ]". Para la tercera columna contiene el porcentaje de conservación de $\vec{SC}$. En la última columna contiene la longitud de la $\vec{SC}$. De este archivo se obtiene $C\vec{S}C = \left\{\vec{SC}^i | i = 1, 2, \ldots, ns\right\}$ y

$L\vec{S}C = \{lsc^i | i = 1, 2, \ldots, ns\}$, donde lsc^i es la longitud de la $\vec{SC}^i$ y ns es el número de secuencias que se toman del archivo **Secuencia_Conservada_[10, 50].txt** tomando un criterio de selección, por ejemplo un valor mínimo de conservación de $\vec{SC}$. En la Sección 3.3.2.4 se explica cómo se introduce $C\vec{S}C$ en MATEDA para su procesamiento en los posibles casos de que $ns = 1$ o $ns \geq 2$.

3.3.1.2. Obtención de las matrices de compatibilidad por carga, peso e hidropaticidad

Las matrices de compatibilidad de los aminoácidos se obtuvieron del artículo de [Biro, 2006]. En dicho artículo se presentan las fórmulas con las que se generaron las diferentes matrices de compatibilidad. La Ecuación 3.5 es utilizada para obtener la Matriz de Compatibilidad por Carga (MCC), ver Tabla 3.2. Mientras que la Ecuación 3.6 es utilizada para obtener la Matriz de Compatibilidad por Peso (MCP), ver Tabla 3.3. Para la Ecuación 3.7 es utilizada para obtener la Matriz de Compatibilidad por Hidropaticidad (MCH), ver Tabla 3.4, los valores utilizados por las ecuaciones se encuentran en la Tabla 3.1

Las Tablas 3.2, 3.3 y 3.4 son utilizadas por MATEDA. Las Tablas se almacenan en un archivo de texto plano (.txt) y se colocan en la carpeta **Modulos**, se explica en la Sección 3.3.2.3, con los siguientes nombres:

- Para la Tabla 3.2 se guardó con el nombre de MCC.txt
- Para la Tabla 3.3 se guardó con el nombre de MCP.txt
- Para la Tabla 3.4 se guardó con el nombre de MCH.txt

$$MCC\,(A,B) = 11 - [pI\,(A) - 7] \cdot [pI\,(B) - 7] \cdot \frac{19}{33.8} \tag{3.5}$$

$$MCP\,(A,B) = 20 - \left| [MW\,(A) + MW\,(B) - 123] \cdot \frac{19}{135} \right| \tag{3.6}$$

$$MCH\,(A,B) = 20 - \left| [HM\,(A) - HM\,(B)] \cdot \frac{19}{10.6} \right| \tag{3.7}$$

Donde:

- A y B: Representan los aminoácidos
- pI: Punto Isoeléctrico
- MW: Peso Molecular
- HM: Momento de Hidropacidad

	Ala, A	Cys, C	Asp, D	Glu, E	Phe, F	Gly, G	His, H	Ile, I	Lys, K	Leu, L	Met, M	Asn, N	Pro, P	Gln, Q	Ard, R	Ser, S	Thr, T	Val, V	Trp, W	Tyr, Y
pI	6.00	5.07	2.77	3.22	5.48	5.97	7.59	6.02	9.74	5.98	5.74	5.41	6.30	5.65	10.76	5.68	5.60	5.96	5.89	5.66
HM	1.00	0.20	-3.00	-2.60	2.50	0.70	-1.70	3.10	-4.60	2.20	1.10	-2.70	-0.30	-2.90	-7.50	-1.10	-0.80	2.30	1.50	0.10
MW	71.08	103.14	115.09	129.11	147.17	57.05	137.14	113.16	128.17	113.16	131.20	114.10	97.12	128.13	156.19	87.08	101.10	99.13	186.21	163.17

Tabla 3.1: Propiedades Fisicoquímicas de los aminoácidos. Obtenida de [Biro, 2006].

	Ala, A	Cys, C	Glu, E	Asp, D	Phe, F	Gly, G	His, H	Ile, I	Lys, K	Leu, L	Met, M	Asn, N	Pro, P	Gln, Q	Ard, R	Ser, S	Thr, T	Val, V	Trp, W	Tyr, Y
Ala, A	10.44	9.92	8.62	8.88	10.15	10.42	11.33	10.45	12.54	10.43	10.29	10.11	10.61	10.24	13.11	10.26	10.21	10.42	10.38	10.25
Cys, C	9.92	8.91	6.41	6.90	9.35	9.88	11.64	9.94	13.97	9.89	9.63	9.28	10.24	9.54	15.08	9.57	9.48	9.87	9.80	9.55
Glu, E	8.62	6.41	0.94	2.01	7.39	8.55	12.40	8.67	17.52	8.57	8.00	7.22	9.34	7.79	19.94	7.86	7.67	8.53	8.36	7.81
Asp, D	8.88	6.90	2.01	2.97	7.77	8.81	12.25	8.92	16.82	8.83	8.32	7.62	9.51	8.13	18.99	8.20	8.03	8.79	8.64	8.15
Phe, F	10.15	9.35	7.39	7.77	9.70	10.12	11.50	10.16	13.34	10.13	9.92	9.64	10.40	9.85	14.21	9.87	9.80	10.11	10.05	9.86
Gly, G	10.42	9.88	8.55	8.81	10.12	10.40	11.34	10.43	12.59	10.41	10.27	10.08	10.59	10.22	13.18	10.24	10.19	10.40	10.36	10.22
His, H	11.33	11.64	12.40	12.25	11.50	11.34	10.80	11.33	10.09	11.34	11.42	11.53	11.23	11.45	9.75	11.44	11.46	11.34	11.37	11.44
Ile, I	10.45	9.94	8.67	8.92	10.16	10.43	11.33	10.46	12.51	10.44	10.31	10.12	10.61	10.26	13.07	10.27	10.23	10.43	10.39	10.26
Lys, K	12.54	13.97	17.52	16.82	13.34	12.59	10.09	12.51	6.78	12.57	12.94	13.45	12.08	13.08	5.21	13.03	13.16	12.60	12.71	13.06
Leu, L	10.43	9.89	8.57	8.83	10.13	10.41	11.34	10.44	12.57	10.42	10.28	10.09	10.60	10.23	13.16	10.24	10.20	10.40	10.36	10.23
Met, M	10.29	9.63	8.00	8.32	9.92	10.27	11.42	10.31	12.94	10.28	10.11	9.87	10.50	10.04	13.66	10.07	10.01	10.26	10.21	10.05
Asn, N	10.11	9.28	7.22	7.62	9.64	10.08	11.53	10.12	13.45	10.09	9.87	9.58	10.37	9.79	14.36	9.82	9.75	10.07	10.01	9.80
Pro, P	10.61	10.24	9.34	9.51	10.40	10.59	11.23	10.61	12.08	10.60	10.50	10.37	10.72	10.47	12.48	10.48	10.45	10.59	10.56	10.47
Gln, Q	10.24	9.54	7.79	8.13	9.85	10.22	11.45	10.26	13.08	10.23	10.04	9.79	10.47	9.98	13.85	10.00	9.94	10.21	10.16	9.98
Ard, R	13.11	15.08	19.94	18.99	14.21	13.18	9.75	13.07	5.21	13.16	13.66	14.36	12.48	13.85	3.05	13.79	13.96	13.20	13.35	13.83
Ser, S	10.26	9.57	7.86	8.20	9.87	10.24	11.44	10.27	13.03	10.24	10.07	9.82	10.48	10.00	13.79	10.02	9.96	10.23	10.18	10.01
Thr, T	10.21	9.48	7.67	8.03	9.80	10.19	11.46	10.23	13.16	10.20	10.01	9.75	10.45	9.94	13.96	9.96	9.90	10.18	10.13	9.95
Val, V	10.42	9.87	8.53	8.79	10.11	10.40	11.34	10.43	12.60	10.40	10.26	10.07	10.59	10.21	13.20	10.23	10.18	10.39	10.35	10.22
Trp, W	10.38	9.80	8.36	8.64	10.05	10.36	11.37	10.39	12.71	10.36	10.21	10.01	10.56	10.16	13.35	10.18	10.13	10.35	10.31	10.16
Tyr, Y	10.25	9.55	7.81	8.15	9.86	10.22	11.44	10.26	13.06	10.23	10.05	9.80	10.47	9.98	13.83	10.01	9.95	10.22	10.16	9.99

Tabla 3.2: Matriz de Compatibilidad por Carga (MCC). Se elaboró a partir de la Ecuación 3.5 y los datos de la Tabla 3.1.

	Ala, A	Cys, C	Glu, E	Asp, D	Phe, F	Gly, G	His, H	Ile, I	Lys, K	Leu, L	Met, M	Asn, N	Pro, P	Gln, Q	Ard, R	Ser, S	Thr, T	Val, V	Trp, W	Tyr, Y
Ala, A	18.23	15.83	14.94	13.89	12.54	19.28	13.29	15.08	13.96	15.08	13.73	15.01	16.28	13.96	11.86	17.03	15.98	16.13	9.61	11.34
Cys, C	15.83	13.43	12.54	11.49	10.14	16.88	10.89	12.68	11.56	12.68	11.33	12.61	13.88	11.56	9.46	14.63	13.58	13.73	7.22	8.94
Glu, E	14.94	12.54	11.64	10.59	9.24	15.99	9.99	11.79	10.66	11.79	10.44	11.72	12.99	10.67	8.57	13.74	12.69	12.84	6.32	8.05
Asp, D	13.89	11.49	10.59	9.54	8.19	14.94	8.94	10.74	9.61	10.74	9.39	10.67	11.94	9.62	7.52	12.69	11.64	11.79	5.27	7.00
Phe, F	12.54	10.14	9.24	8.19	6.84	13.58	7.59	9.39	8.26	9.39	8.04	9.32	10.59	8.27	6.17	11.34	10.29	10.44	3.92	5.64
Gly, G	19.28	16.88	15.99	14.94	13.58	19.67	14.34	16.13	15.01	16.13	14.78	16.06	17.33	15.01	12.91	18.08	17.03	17.18	10.66	12.39
His, H	13.29	10.89	9.99	8.94	7.59	14.34	8.34	10.14	9.01	10.14	8.79	10.07	11.34	9.02	6.92	12.09	11.04	11.19	4.67	6.40
Ile, I	15.08	12.68	11.79	10.74	9.39	16.13	10.14	11.93	10.81	11.93	10.58	11.86	13.13	10.81	8.71	13.88	12.83	12.98	6.47	8.19
Lys, K	13.96	11.56	10.66	9.61	8.26	15.01	9.01	10.81	9.68	10.81	9.46	10.74	12.01	9.69	7.59	12.76	11.71	11.86	5.34	7.07
Leu, L	15.08	12.68	11.79	10.74	9.39	16.13	10.14	11.93	10.81	11.93	10.58	11.86	13.13	10.81	8.71	13.88	12.83	12.98	6.47	8.19
Met, M	13.73	11.33	10.44	9.39	8.04	14.78	8.79	10.58	9.46	10.58	9.23	10.51	11.78	9.46	7.36	12.53	11.48	11.63	5.12	6.84
Asn, N	15.01	12.61	11.72	10.67	9.32	16.06	10.07	11.86	10.74	11.86	10.51	11.79	13.06	10.74	8.64	13.81	12.76	12.91	6.40	8.12
Pro, P	16.28	13.88	12.99	11.94	10.59	17.33	11.34	13.13	12.01	13.13	11.78	13.06	14.33	12.01	9.91	15.08	14.03	14.18	7.67	9.39
Gln, Q	13.96	11.56	10.67	9.62	8.27	15.01	9.02	10.81	9.69	10.81	9.46	10.74	12.01	9.69	7.59	12.76	11.71	11.86	5.35	7.07
Ard, R	11.86	9.46	8.57	7.52	6.17	12.91	6.92	8.71	7.59	8.71	7.36	8.64	9.91	7.59	5.49	10.66	9.61	9.76	3.25	4.97
Ser, S	17.03	14.63	13.74	12.69	11.34	18.08	12.09	13.88	12.76	13.88	12.53	13.81	15.08	12.76	10.66	15.83	14.79	14.93	8.42	10.14
Thr, T	15.98	13.58	12.69	11.64	10.29	17.03	11.04	12.83	11.71	12.83	11.48	12.76	14.03	11.71	9.61	14.79	13.74	13.88	7.37	9.09
Val, V	16.13	13.73	12.84	11.79	10.44	17.18	11.19	12.98	11.86	12.98	11.63	12.91	14.18	11.86	9.76	14.93	13.88	14.03	7.52	9.24
Trp, W	9.61	7.22	6.32	5.27	3.92	10.66	4.67	6.47	5.34	6.47	5.12	6.40	7.67	5.35	3.25	8.42	7.37	7.52	1.00	2.72
Tyr, Y	11.34	8.94	8.05	7.00	5.64	12.39	6.40	8.19	7.07	8.19	6.84	8.12	9.39	7.07	4.97	10.14	9.09	9.24	2.72	4.45

Tabla 3.3: Matriz de Compatibilidad por Peso (MCP). Se elaboró a partir de la Ecuación 3.6 y los datos de la Tabla 3.1.

	Ala, A	Cys, C	Glu, E	Asp, D	Phe, F	Gly, G	His, H	Ile, I	Lys, K	Leu, L	Met, M	Asn, N	Pro, P	Gln, Q	Ard, R	Ser, S	Thr, T	Val, V	Trp, W	Tyr, Y
Ala, A	20.00	18.57	12.83	13.55	17.31	19.46	15.16	16.24	9.96	17.85	19.82	13.37	17.67	13.01	4.76	16.24	16.77	17.67	19.10	18.39
Cys, C	18.57	20.00	14.26	14.98	15.88	19.10	16.59	14.80	11.40	16.42	18.39	14.80	19.10	14.44	6.20	17.67	18.21	16.24	17.67	19.82
Glu, E	12.83	14.26	20.00	19.28	10.14	13.37	17.67	9.07	17.13	10.68	12.65	19.46	15.16	19.82	11.93	16.59	16.06	10.50	11.93	14.44
Asp, D	13.55	14.98	19.28	20.00	10.86	14.08	18.39	9.78	16.42	11.40	13.37	19.82	15.88	19.46	11.22	17.31	16.77	11.22	12.65	15.16
Phe, F	17.31	15.88	10.14	10.86	20.00	16.77	12.47	18.92	7.27	19.46	17.49	10.68	14.98	10.32	2.08	13.55	14.08	19.64	18.21	15.70
Gly, G	19.46	19.10	13.37	14.08	16.77	20.00	15.70	15.70	10.50	17.31	19.28	13.91	18.21	13.55	5.30	16.77	17.31	17.13	18.57	18.92
His, H	15.16	16.59	17.67	18.39	12.47	15.70	20.00	11.40	14.80	13.01	14.98	18.21	17.49	17.85	9.60	18.92	18.39	12.83	14.26	16.77
Ile, I	16.24	14.80	9.07	9.78	18.92	15.70	11.40	20.00	6.20	18.39	16.42	9.60	13.91	9.25	1.00	12.47	13.01	18.57	17.13	14.62
Lys, K	9.96	11.40	17.13	16.42	7.27	10.50	14.80	6.20	20.00	7.81	9.78	16.59	12.29	16.95	14.80	13.73	13.19	7.63	9.07	11.58
Leu, L	17.85	16.42	10.68	11.40	19.46	17.31	13.01	18.39	7.81	20.00	18.03	11.22	15.52	10.86	2.61	14.08	14.62	19.82	18.75	16.24
Met, M	19.82	18.39	12.65	13.37	17.49	19.28	14.98	16.42	9.78	18.03	20.00	13.19	17.49	12.83	4.58	16.06	16.59	17.85	19.28	18.21
Asn, N	13.37	14.80	19.46	19.82	10.68	13.91	18.21	9.60	16.59	11.22	13.19	20.00	15.70	19.64	11.40	17.13	16.59	11.04	12.47	14.98
Pro, P	17.67	19.10	15.16	15.88	14.98	18.21	17.49	13.91	12.29	15.52	17.49	15.70	20.00	15.34	7.09	18.57	19.10	15.34	16.77	19.28
Gln, Q	13.01	14.44	19.82	19.46	10.32	13.55	17.85	9.25	16.95	10.86	12.83	19.64	15.34	20.00	11.75	16.77	16.24	10.68	12.11	14.62
Ard, R	4.76	6.20	11.93	11.22	2.08	5.30	9.60	1.00	14.80	2.61	4.58	11.40	7.09	11.75	20.00	8.53	7.99	2.43	3.87	6.38
Ser, S	16.24	17.67	16.59	17.31	13.55	16.77	18.92	12.47	13.73	14.08	16.06	17.13	18.57	16.77	8.53	20.00	19.46	13.91	15.34	17.85
Thr, T	16.77	18.21	16.06	16.77	14.08	17.31	18.39	13.01	13.19	14.62	16.59	16.59	19.10	16.24	7.99	19.46	20.00	14.44	15.88	18.39
Val, V	17.67	16.24	10.50	11.22	19.64	17.13	12.83	18.57	7.63	19.82	17.85	11.04	15.34	10.68	2.43	13.91	14.44	20.00	18.57	16.06
Trp, W	19.10	17.67	11.93	12.65	18.21	18.57	14.26	17.13	9.07	18.75	19.28	12.47	16.77	12.11	3.87	15.34	15.88	18.57	20.00	17.49
Tyr, Y	18.39	19.82	14.44	15.16	15.70	18.92	16.77	14.62	11.58	16.24	18.21	14.98	19.28	14.62	6.38	17.85	18.39	16.06	17.49	20.00

Tabla 3.4: Matriz de Compatibilidad por Hidropaticidad (MCH). Se elaboró a partir de la Ecuación 3.7 y los datos de la Tabla 3.1.

3.3.2. MATEDA

La implementación de MATEDA se divide en 4 puntos importantes, en la Sección 3.3.2.1 se explica todo el proceso de descarga de la herramienta MATEDA. La Sección 3.3.2.2 contiene la explicación de su configuración para ser usada en MATLAB, se toma por hecho que se tiene instalado el IDE de MATLAB en el equipo donde se realizan los experimentos, para la Sección 3.3.2.3 se explican los módulos que se generaron para el funcionamiento de MATEDA para el PDM y en la Sección 3.3.2.4 se explican los posibles casos del $C\vec{S}C$ y la ejecución de MATEDA.

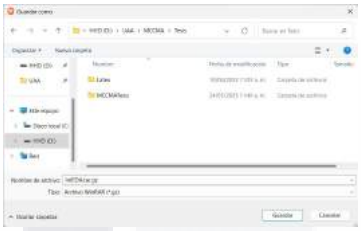
3.3.2.1. Descarga de MATEDA

La descarga de MATEDA es sencilla como se observa en la Figura 3.8 que contiene las diferentes ventanas para la descarga de MATEDA. Primero se abre el navegador de su preferencia, ver Figura 3.8a, se escribe en la barra de búsqueda MATEDA y se ingresa en el primer link que se muestra, Figura 3.8b, en la página que se desplegó, se da clic en el enlace Download, Figura 3.8c, se muestra una ventana de descarga, se selecciona la dirección de descarga y se da clic en aceptar, Figura 3.8d, una vez descargado MATEDA, se dirige a la carpeta donde se guardó, se observa que es un paquete **tar.gz**, Figura 3.8e, se revisa su contenido dando doble clic en el archivo, Figura 3.8f, para extraer a MATEDA se da clic derecho sobre el archivo y se elige la opción de extraer aquí, Figura 3.8g, una vez extraído el contenido del paquete se accede a la carpeta generada, Figura 3.8h y se revisa el contenido de MATEDA, Figura 3.8i.

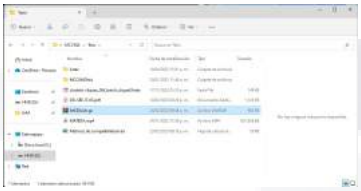
(a) Se abre el navegador.

(b) Se busca MATEDA e ingresa al primer link.

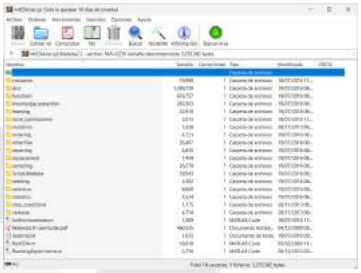
(c) Se da clic en el enlace Download.



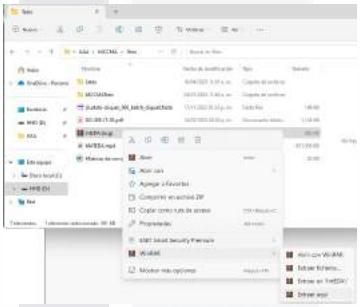
(d) Se selecciona la carpeta y se guarda.



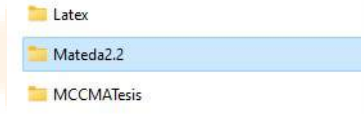
(e) Se ingresa a la carpeta donde se guardo.



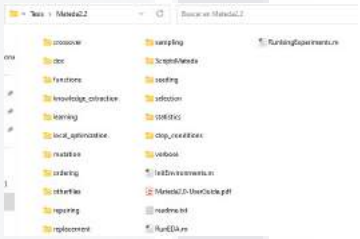
(f) Se revisa el contenido del paquete.



(g) Descomprimimos el paquete.



(h) Ingresamos a la carpeta que se genera.

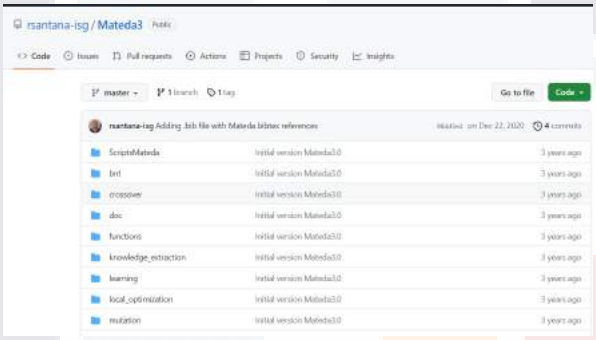


(i) Se observa el contenido de MATEDA.

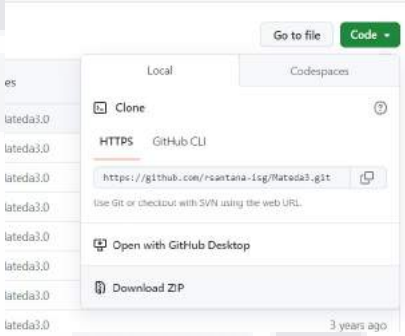
Figura 3.8: Descarga de MATEDA. Autoría propia.

Para descargar la última versión de MATEDA, se descarga directamente del GitHub de Roberto Santana, el proceso de descarga se ilustra en la Figura 3.9. Se comienza abriendo un navegador, se busca MATEDA y se ingresa al repositorio de GitHub de Roberto Santana, Figura 3.9a, dentro del repositorio se da clic en el botón verde que dice **Code**, Figura 3.9b, se da clic en el enlace **Download ZIP**, Figura 3.9c, después de aquí se repiten los pasos de la Figura 3.8 desde la Figura 3.8d. Después de que se tiene descargado MATEDA, se procede a su configuración para su uso.

(a) Se abre el navegador, se busca MATEDA y se ingresa al repositorio de GitHub.



(b) Dentro del repositorio se da clic en el botón verde que dice **Code**.

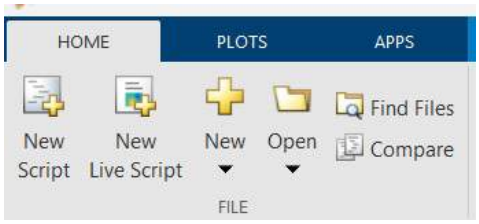


(c) Se da clic en el enlace **Download ZIP**.

Figura 3.9: Descarga de MATEDA última versión. Autoría propia.

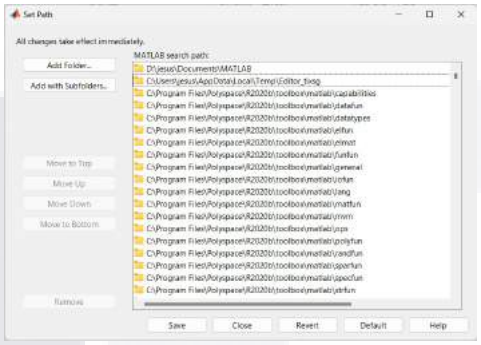
3.3.2.2. Configuración de MATEDA

Se comienza colocando la carpeta que contiene a MATEDA en una ruta definitiva para su uso y almacenamiento de toda información relacionada con la herramienta. Con MATLAB abierto, se procede agregar la carpeta de MATEDA a MATLAB, en la Figura 3.10 se observan los pasos. Primero se ubica la barra superior en la pestaña de **HOME**, Figura 3.10a, en la sección de **ENVIRONMENT** se selecciona la opción de **Set Path**, Figura 3.10b, en la ventana que se muestra se da clic en el botón de **Add Folder**, Figura 3.10c, en la ventana que se muestra se busca y selecciona la carpeta que contiene a MATEDA, Figura 3.10d, se verifica que la carpeta se añadió a la lista y se da clic en el botón de **Save** y se cierra la ventana, Figura 3.10e.

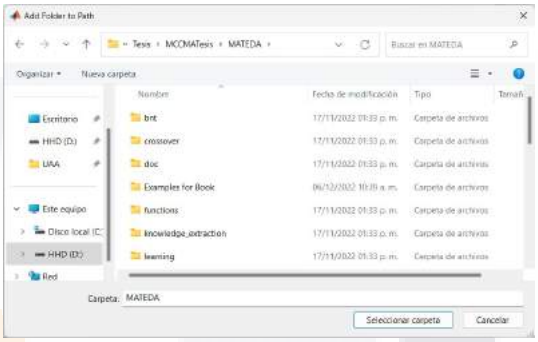


(a) Pestaña **HOME** de MATLAB.

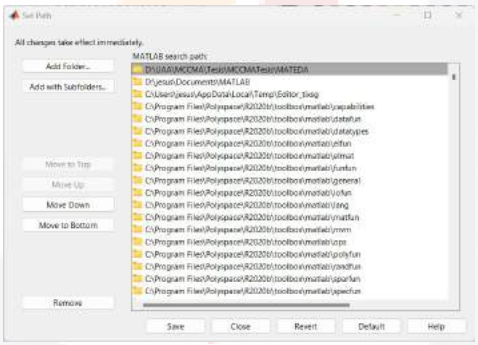
(b) Se da clic en la opción de **Set Path**.



(c) Se da clic en el botón de **Add Folder**.



(d) Se busca la carpeta donde se almaceno MATEDA.

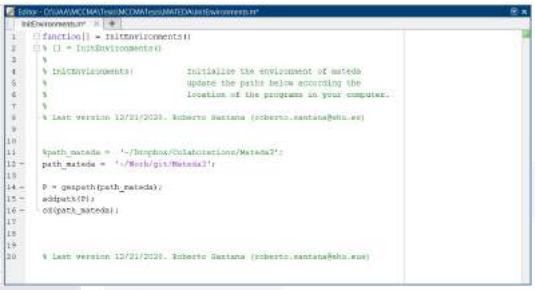


(e) Se verifica que se añadió la carpeta seleccionada, se da clic en **Save** y se cierra la ventana.

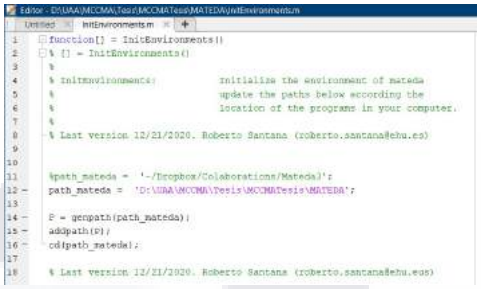
Figura 3.10: Añade la carpeta de MATEDA a MATLAB. Autoría propia.

Ahora se continúa con los archivos internos de MATEDA, en la Figura 3.11 se observa los pasos para la configuración, se abre el archivo **InitEnvironment.m**, Figura 3.11a, se modifica la línea 12 del archivo que corresponde a la variable **path_mateda**, en ella se ingresa la ruta donde se encuentra la carpeta de MATEDA, Figura 3.11b, se guarda la modificación realizada y se cierra el archivo. Para verificar que se configuró de forma correcta la herramienta, en la línea de comandos de MATLAB se ejecuta el

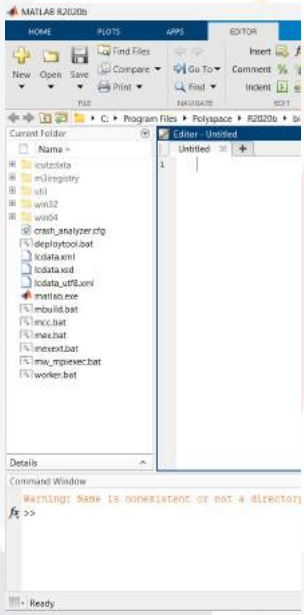
comando **InitEnvironments**, en la Figura 3.11c se observan las carpetas antes de la ejecución del comando, si no se muestra ningún mensaje de error, la configuración es correcta y en la Figura 3.11d se observa el contenido de la carpeta de MATEDA.



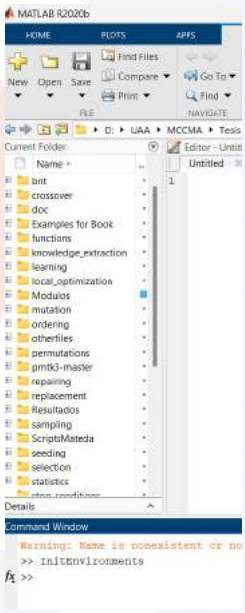
(a) Archivo **InitEnvironments.m**.



(b) Archivo **InitEnvironments.m** modificado.



(c) Carpetas abiertas antes de ejecutar el comando **InitEnvironments**.



(d) Carpetas abiertas después de ejecutar el comando **InitEnvironments**.

Figura 3.11: Configuración de MATEDA. Autoría propia.

Después de que se verifica que MATEDA está correctamente configurado, se crean los módulos necesarios para su funcionamiento con nuestro problema de investigación y el archivo principal para ejecutar MATEDA.

3.3.2.3. Módulos para MATEDA

En la carpeta de MATEDA se crea una carpeta y se le asigna un nombre, para este caso se nombro **Modulos**. La carpeta contiene los siguientes archivos:

- **Compatibilidad.m**: Módulo con el que se obtiene el porcentaje del fitness de la población de cada objetivo del problema de optimización.
- **Compatibilidad2.m**: Módulo con el que se obtiene el fitness de la población de cada objetivo del problema de optimización.
- **Convertidor.m**: Módulo que se encarga de convertir una secuencia de aminoácidos a un vector de números o convertir un vector de números a una secuencia de aminoácidos.
- **Crear_Poblacion.m**: Módulo donde se crea la población inicial con la que trabajara MATEDA.
- **Graficacion.m**: Módulo que genera una gráfica con el comportamiento de cada objetivo del mejor individuo de cada generación.
- **Guardar_Informacion.m**: Módulo que se encarga de generar 2 archivos txt donde se guarda cada generación del algoritmo y el mejor individuo de cada generación, además de generar un archivo fasta.
- **MCC.txt**: Archivo que contiene los valores de compatibilidad por carga de la Tabla 3.2
- **MCH.txt**: Archivo que contiene los valores de compatibilidad por peso de la Tabla 3.3
- **MCP.txt**: Archivo que contiene los valores de compatibilidad por hidropaticidad de la Tabla 3.4

- **Nombre.m**: Módulo que se encarga de generar nombre único para guardar los archivos.

El código completo del trabajo de investigación lo encontramos en el repositorio de GitLab de Jesús Correa `https://gitlab.com/CorreaJesus/MCCMATesis/-/tree/develop/MATEDA`.

Además de los archivos contenidos en la carpeta **Módulos**, se necesita generar dos archivos principales con los cuales se ejecutara MATEDA, para este caso se generó **Principal.m** y **Principal-Test.mlx**. A partir del archivo **RunEDA.m** se generó un nuevo archivo para modificaciones, para este caso se nombró **RunEDATesis.m**.

Tanto el archivo **Principal.m** y **Principal-Test.mlx** se dividen en 8 partes diferentes las cuales se hablaran a continuación y en la Figura 3.12 se muestra las partes más importantes del archivo **Principal-Test.mlx**.

En la primera parte se encuentra el nombre de la tesis junto con el nombre del autor. Para la segunda parte está compuesta por el comando **InitEnvironments** con el cual se activa el módulo de MATEDA. La tercera parte está compuesta por comandos para limpiar la consola y limpiar las variables del IDE. La cuarta parte se dedica a la declaración de las variables globales que necesita MATEDA para trabajar, Figura 3.12a. La quinta parte se inicializan las variables declaradas en la cuarta parte, Figura 3.12b. La sexta parte se dedica para ingresar $C\vec{S}C$ o $\vec{SC}$, dependiendo el caso, Figura 3.12c. En la séptima parte se declaran las variables que necesitan MATEDA, Figura 3.12d. En la octava parte es donde se ejecuta MATEDA, Figura 3.12e.

Variables Globales

```
global secuencia; % Secuencia base con la cual se buscara un conjunto de
                  % secuencias artificiales que maximicen su compatibilidad.
global longitud; % Longitud de la secuencia.
global mC; % Matriz que alacena la compatibilidad por carga.
global mP; % Matriz que alacena la compatibilidad por peso.
global mH; % Matriz que alacena la compatibilidad por hidropaticidad.
global maxmC; % Valor maximo que puede tomar compatibilidad por carga.
global maxmP; %Valor maximo que puede tomar compatibilidad por peso.
global maxmH; % Valor maximo que puede tomar compatibilidad por hidropaticidad.
global nombre; % Contiene el nombre del archivo donde se guardaran las secuencias
               % y sus valores de compatibilidad por generacion.
global nombre2; % Contiene el nombre del archivo donde se guardaran las secuencias
                % la mejor compatibilidad por cada objetivo y cada genmeracion.
global carpeta; % Contiene el nombre de la carpeta donde se guardara los resultados.
```

(a) Declaración de variables globales.

Inicializamos variables importantes

```
mC =importdata('MCC.txt'); % Archivo donde se almacena la matriz.
mP =importdata('MCP.txt'); % Archivo donde se almacena la matriz.
mH =importdata('MCH.txt'); % Archivo donde se almacena la matriz.
nombre_Original = 'Experimento'; % Contiene el nombre donde de los archivos donde se guardaran los resultados.
maxmC = max(max(mC)); % Valor maximo de la compatibilidas por carga.
maxmP = max(max(mP)); % Valor maximo de la compatibilidas por peso.
maxmH  = max(max(mH)); % Valor maximo de la compatibilidas por hidropaticidad.
```

(b) Inicialización de las variables globales.

Matriz de secuencias conservadas

Añadimos una matriz con un conjunto de secuencias conservadas.

```
% Se cuencias de aminoacidos conservados.
secuenciaAminoacidos = ['Q' 'V' 'D' 'K' 'L' 'I' 'I' 'G' 'R' 'L' 'A' 'A' 'L' '-' '-' '-'
                        'K' 'V' 'N' 'E' 'C' 'V' 'K' 'S' 'Q' 'S' 'S' 'R' '-' '-' '-' '-'
                        'I' 'E' 'D' 'L' 'L' 'F' 'D' 'K' 'V' 'T' '-' '-' '-' '-' '-' '-'
                        'N' 'E' 'C' 'V' 'K' 'S' 'Q' 'S' 'S' 'R' 'N' 'G' 'F' 'C' 'G' 'N'
                        'R' 'S' 'A' 'S' 'E' 'D' 'L' 'L' 'F' 'D' 'K' 'V' 'V' 'T' '-' '-'
                        'A' 'Q' 'V' 'D' 'N' 'L' 'I' 'T' 'G' 'R' 'L' 'A' 'A' 'L' 'N' '-'
                        'S' 'A' 'I' 'E' 'D' 'L' 'L' 'F' 'D' 'K' 'V' 'V' 'T' '-' '-' '-'
                        'K' 'W' 'P' 'W' 'V' 'N' 'L' 'L' 'I' '-' '-' '-' '-' '-' '-' '-'];
```

(c) Ingreso de $C\vec{S}C$ o $S\vec{C}$ dependiendo el caso

Declaración de variables fijas

Son aquellas variables que, a lo largo de la ejecución del programa, no cambian su valor.

```
PopSize = 100;
F = 'Compatibilidad2';
cache = [1,1,1,1,1];
maxgen = 2000;
numeroExperimentos = 30;
```

(d) Declaración de variables fijas.

Ejecución

Es la parte más importante del programa, en ella se realizan lo siguiente:

- Se obtiene cada secuencia conservada de la matriz.
- En el caso que la secuencia contenga gaps, son eliminados.
- Se convierten las secuencias a sus equivalentes numéricos.
- Se crean la carpeta donde se almacenarán.
- Se genera el nombre único con el que se guardaran los resultados.
- Se asignan los valore necesarios para el funcionamiento de MATEDA.
- Se ejecuta MATEDA.

```
for i=1:size(secuenciaAminoacidos,1)
    aux = split(secuenciaAminoacidos(i,:),'-');
    auxString=char(aux(1,:));
    secuencia = Convertidor(auxString);
    longitud = size(secuencia,2);
    n = longitud;
    Card = 20*ones(1,n); % Valor maximo que puede tomar nuestra variable.
    carpeta = strcat('Resultados','\',auxString); % Contiene el nombre de la carpeta donde se guardara los resu

    if ~exist(carpeta)
        mkdir(carpeta)
    end
    for j=1:numeroExperimentos
        disp([sprintf('Numero de Experimento: %d\nSecuencia Conservada %s',j,auxString)]);
        [nombre,nombre2] = Nombre(nombre_Original);

        selparams(1:2) = {0.5,'ParetoRank_ordering'};
        edaparams{1} = {'stop_cond_method','max_gen',{maxgen}};
        edaparams{2} = {'selection_method','truncation_selection',selparams};
        edaparams{3} = {'replacement_method','best_elitism',{'ParetoRank_ordering'}};
        edaparams{4} = {'seeding_pop_method','Crear_Poblacion', {}};
        edaparams{5} = {'verbose_method','none',{}}; % Para evitar que nos muestre informacion  el algoritmo.
        [AllStat,Cache] = RunEDATesis(PopSize,n,F,Card,cache,edaparams);
    end
end
```

(e) Código principal donde se utiliza MATEDA

Figura 3.12: Segmentos de código de **Principal-Test.mlx**. Autoría propia.

En el archivo **RunEDATesis.m** es una copia del archivo **RunEDA.m**, pero en el cual se modificó desde la línea 333 del código, en la Figura 3.13 se observa la modifi-

cación. En la Figura 3.13a se aprecia el código de **RunEDA.m** original de MATEDA. En la Figura 3.13b se observa el código de **RunEDATesis.m** que es el código de **RunEDA.m**.

```
331 -       time_operations(k,8) = cputime - previous_t;  % Time spent in the whole generation
332 -       previous_t = cputime;
333         % Statistics are computed
334 -       [AllStat] = eval([statistics_method,'(k,Pop,FunVal,time_operations,number_evaluations,All
335
336
337 -       if(strcmp(verbose_method,'none') ~= 1)  %  Statistics information about the run is printe
338 -         eval([verbose_method,'(k,AllStat,verbose_params,auxedaparams)']);
339 -       end
340
341 -        k=k+1;
342 -   end
343
344
345 -   return
```

(a) Código original de **RunEDA.m**.

```
331 -       time_operations(k,8) = cputime - previous_t;  % Time spent in the whole generation
332 -       previous_t = cputime;
333          % Codigo propio
334 -       secuencias =Convertidor(Pop);
335 -       Guardar_Informacion(nombre,nombre2,secuencias,FunVal, k);
336         % Codigo propio
337
338         % Statistics are computed
339 -       [AllStat] = eval([statistics_method,'(k,Pop,FunVal,time_operations,number_evaluations,All
340
341 -       if(strcmp(verbose_method,'none') ~= 1)  %  Statistics information about the run is printe
342 -         eval([verbose_method,'(k,AllStat,verbose_params,auxedaparams)']);
343 -       end
344 -       k=k+1;
345 -   end
346 -   Graficacion(nombre2);
347 -   return
```

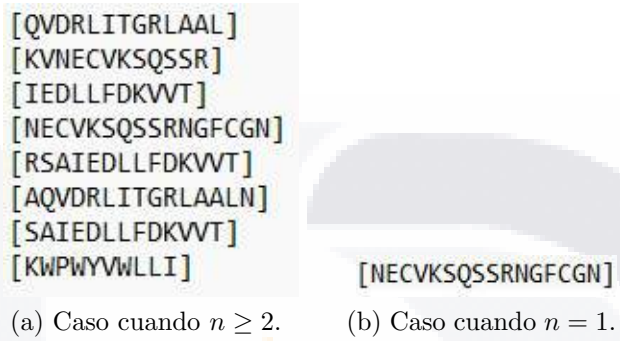
(b) Código de **RunEDATesis.m**.

Figura 3.13: Comparación entre **RunEDA.m** y **RunEDATesis.m**. Autoría propia.

3.3.2.4. Casos de $C\vec{S}C$

MATEDA necesita un formato específico para trabajar con $C\vec{S}C$, además existen dos casos que se presentan en relación con ns. El primer caso es cuando $ns = 1$ y el segundo caso es cuando $ns \geq 2$, es decir el primer caso ocurre cuando $C\vec{S}C$ es integrado únicamente por una secuencia $C\vec{S}C = \left\{\vec{SC}^1\right\}$. El segundo caso es cuando $C\vec{S}C$ es integrado por más de una $\vec{SC}$. Al momento de obtener $C\vec{S}C$ del archivo **Secuen-**

cia_Conservada_[10, 50].txt se obtiene algo semejante a lo que se muestra en la Figura 3.14, para el caso $n \geq 2$ tenemos la Figura 3.14a y para el caso $ns = 1$ tenemos la Figura 3.14b. Para ambos casos a continuación se muestra la forma en la que debe ser convertido nuestro $C\vec{S}C$ para ser utilizado por MATEDA.



(a) Caso cuando $n \geq 2$. (b) Caso cuando $n = 1$.

Figura 3.14: Posibles casos de ns. Autoría propia.

Cuando $ns = 1$:

Para convertir una $\vec{SC}$ cada elemento de la secuencia debe ir entre comillas simples “ ’ ’ ” y separado por coma “,” o por un espacio en blanco “ ”. Por ejemplo, se tiene $\vec{SC} = [KWPWYVWLLI]$ para que sea utilizado por MATEDA se debe declarar de la siguiente manera:

Utilizando espacios en blanco.

```
1 secuenciaAminoacidos = ['K' 'W' 'P' 'W' 'Y' 'V' 'W' 'L' 'L' 'I'];
```

Utilizando comas.

```
1 secuenciaAminoacidos = ['K', 'W', 'P', 'W', 'Y', 'V', 'W', 'L', 'L', 'I'];
```

Cuando $ns \geq 1$:

Para convertir $C\vec{S}C$, primero se convierte cada $\vec{SC}^i$ como se muestra anteriormente, solo que para esta conversión se omiten los corchetes “[ ]” para cada secuencia y son separadas por un salto de línea. Existen dos casos que se pueden presentar cuando ya se

convirtieron las $\vec{SC}$. El primero de ellos es que $\forall lsc^i = lsc^j \mid i, j = 1, 2, \ldots, ns$, es decir, todas las secuencias son de la misma longitud. El segundo caso es cuando $\exists lsc^i \neq lsc^j \mid i, j = 1, 2, \ldots, ns$, es decir, existe por lo menos una $\vec{SC}^i$ de una longitud diferente. Para las secuencias con una longitud menor, llenar al final de la secuencia con **gaps** "**-**" hasta completar la longitud. Para que $C\vec{SC}$ sea utilizado por MATEDA se debe declarar de la siguiente manera:

Para los casos que $lsc^i = lsc^j$ Utilizando espacios en blanco.

```
1  secuenciaAminoacidos = ['Q' 'V' 'D' 'R' 'L' 'I' 'T' 'G' 'R' 'L'
2                          'Y' 'I' 'K' 'W' 'P' 'W' 'Y' 'V' 'W' 'L'
3                          'K' 'V' 'N' 'E' 'C' 'V' 'K' 'S' 'Q' 'S'
4                          'A' 'Q' 'V' 'D' 'R' 'L' 'I' 'T' 'G' 'R'
5                          'S' 'A' 'I' 'E' 'D' 'L' 'L' 'F' 'D' 'K'
6                          'K' 'W' 'P' 'W' 'Y' 'V' 'W' 'L' 'L' 'I'
7                          'N' 'E' 'C' 'V' 'K' 'S' 'Q' 'S' 'S' 'R'
8                          'R' 'S' 'A' 'I' 'E' 'D' 'L' 'L' 'F' 'D'
9                          'D' 'R' 'L' 'I' 'T' 'G' 'R' 'L' 'A' 'A'
10                         'R' 'L' 'I' 'T' 'G' 'R' 'L' 'A' 'A' 'L'
11                         'A' 'I' 'E' 'D' 'L' 'L' 'F' 'D' 'K' 'V'
12                         'I' 'E' 'D' 'L' 'L' 'F' 'D' 'K' 'V' 'V'];
```

Utilizando comas.

```
1  secuenciaAminoacidos = ['Q','V','D','R','L','I','T','G','R','L'
2                          'Y','I','K','W','P','W','Y','V','W','L'
3                          'K','V','N','E','C','V','K','S','Q','S'
4                          'A','Q','V','D','R','L','I','T','G','R'
5                          'S','A','I','E','D','L','L','F','D','K'
6                          'K','W','P','W','Y','V','W','L','L','I'
7                          'N','E','C','V','K','S','Q','S','S','R'
```

```
8                                   'R','S','A','I','E','D','L','L','F','D'
9                                   'D','R','L','I','T','G','R','L','A','A'
10                                  'R','L','I','T','G','R','L','A','A','L'
11                                  'A','I','E','D','L','L','F','D','K','V'
12                                  'I','E','D','L','L','F','D','K','V','V'];
```

Para los casos que $lsc^i \neq lsc^j$ Utilizando espacios en blanco.

```
1 secuenciaAminoacidos = ['A' 'C' 'D' 'G' 'F' '-' '-' '-' '-' '-'
2                         'K' 'W' 'P' 'W' 'Y' '-' '-' '-' '-' '-'
3                         'Q' 'V' 'D' 'R' 'L' 'A' 'A' 'L' '-' '-'
4                         'Y' 'I' 'K' 'W' 'P' 'L' '-' '-' '-' '-'
5                         'K' 'V' 'N' 'E' 'C' 'S' 'R' '-' '-' '-'
6                         'I' 'E' 'D' 'L' 'L' 'T' '-' '-' '-' '-'
7                         'K' 'W' 'P' 'W' 'Y' '-' '-' '-' '-' '-'
8                         'R' 'S' 'A' 'I' 'E' 'K' 'V' 'V' 'T' '-'
9                         'A' 'Q' 'V' 'D' 'R' 'L' 'A' 'A' 'L' 'N'
10                        'S' 'A' 'I' 'E' 'D' 'V' 'V' 'T' '-' '-'];
```

Utilizando comas.

```
1 secuenciaAminoacidos = ['A','C','D','G','F','-','-','-','-','-'
2                         'K','W','P','W','Y','-','-','-','-','-'
3                         'Q','V','D','R','L','A','A','L','-','-'
4                         'Y','I','K','W','P','L','-','-','-','-'
5                         'K','V','N','E','C','S','R','-','-','-'
6                         'I','E','D','L','L','T','-','-','-','-'
7                         'K','W','P','W','Y','-','-','-','-','-'
8                         'R','S','A','I','E','K','V','V','T','-'
9                         'A','Q','V','D','R','L','A','A','L','N'
10                        'S','A','I','E','D','V','V','T','-','-'];
```

Para este caso, el $C\vec{S}C$ antes de la aplicación del formato que acepta MATEDA se observa en la Figura 3.14a y en la Figura 3.15 se observa el $C\vec{S}C$ con el formato aceptado por MATEDA. El $C\vec{S}C$ está conformado por 8 secuencias, es decir $ns = 8$.

```
['Q' 'V' 'D' 'R' 'L' 'I' 'T' 'G' 'R' 'L' 'A' 'A' 'L' '-' '-' '-'
'K' 'V' 'N' 'E' 'C' 'V' 'K' 'S' 'Q' 'S' 'S' 'R' '-' '-' '-' '-'
'I' 'E' 'D' 'L' 'L' 'F' 'D' 'K' 'V' 'V' 'T' '-' '-' '-' '-' '-'
'N' 'E' 'C' 'V' 'K' 'S' 'Q' 'S' 'S' 'R' 'N' 'G' 'F' 'C' 'G' 'N'
'R' 'S' 'A' 'I' 'E' 'D' 'L' 'L' 'F' 'D' 'K' 'V' 'V' 'T' '-' '-'
'A' 'Q' 'V' 'D' 'R' 'L' 'I' 'T' 'G' 'R' 'L' 'A' 'A' 'L' 'N' '-'
'S' 'A' 'I' 'E' 'D' 'L' 'L' 'F' 'D' 'K' 'V' 'V' 'T' '-' '-' '-'
'K' 'W' 'P' 'W' 'Y' 'V' 'W' 'L' 'L' 'I' '-' '-' '-' '-' '-' '-']
```

Figura 3.15: $C\vec{S}C$ en formato aceptado por MATEDA. Autoría propia.

3.3.2.5. Ejecución de MATEDA

Para ejecutar MATEDA, se debe asegurar que ya se tiene el $C\vec{S}C$ que se ingresa en la sexta parte del archivo **Principal.m** o **Principal-Test.mlx** en el formato indicado. Para la séptima parte del archivo se declaran las variables utilizadas por MATEDA. Estas variables a continuación se listan y se explica su función.

- **PopSize**: Indica el número de individuos que conforma la población.
- **F**: Contiene el nombre del módulo de la función objetivo.
- **cache**: Indica las partes de los EDAs que se guarda a lo largo de la ejecución del programa.
- **maxgen**: Indica el número máximo de generaciones que ejecuta el algoritmo.
- **numeroExperimentos**: Indica el número de experimentos que ejecutara el archivo.

Para este caso los valores que se le dieron son:

```
1 PopSize = 100;
2 F = 'Compatibilidad2';
3 cache = [1,1,1,1,1];
4 maxgen = 2000;
5 numeroExperimentos = 30;
```

La octava parte del código es donde se llama a ejecutar MATEDA. Para cada secuencia contenida en la variable **secuenciaAminoacidos** se realiza lo siguiente:

Primeramente se almacena la $\vec{SC}^i$ en la variable **aux** eliminando los **gaps** que posiblemente contiene dicha secuencia, después los aminoácidos que contiene la secuencia se transforma en números y se obtiene la longitud de la secuencia. Para realizar esto se utiliza el siguiente código:

```
1 aux = split(secuenciaAminoacidos(i,:),'-');
2 auxString=char(aux(1,:));
3 secuencia = Convertidor(auxString);
4 longitud = size(secuencia,2);
```

Después se continúa declarando más variables, como las siguientes:

```
1 n = longitud;
2 Card = 20*ones(1,n);
```

Ahora se genera el nombre de la carpeta donde se almacenan los resultados generados, en el siguiente código se muestra el cómo se genera el nombre de la carpeta y en el caso de que no exista la carpeta se genera:

```
1 carpeta = strcat('Resultados','\',auxString);
2 if ~exist(carpeta)
3    mkdir(carpeta)
4 end
```

Una vez creada la carpeta que contiene los resultados, se ejecuta MATEDA el número de veces que se le asignó a la variable **numeroExperimentos**:

```
1 disp([sprintf('Numero de Experimento: %d\nSecuencia Conservada %s
      ',j,auxString)]);
2 [nombre,nombre2] = Nombre(nombre_Original);
3 selparams(1:2) = {0.5,'ParetoRank_ordering'};
4 edaparams{1} = {'stop_cond_method','max_gen',{maxgen}};
5 edaparams{2} = {'selection_method','truncation_selection',
      selparams};
6 edaparams{3} = {'replacement_method','best_elitism',{'
      ParetoRank_ordering'}};
7 edaparams{4} = {'seeding_pop_method','Crear_Poblacion', {}};
8 edaparams{5} = {'verbose_method','none',{}};
9 [AllStat,Cache] = RunEDATesis(PopSize,n,F,Card,cache,edaparams);
```

La primera línea muestra en la consola el número de experimento que se está ejecutando y la secuencia conservada con la que se está trabajando. Con la segunda línea se obtienen los nombres de los archivos que almacena los resultados. En la tercera línea se crea una variable que contiene modificadores para el método de selección con el que trabaja el EDA. En la cuarta línea se establece el método de parada del EDA. En la quinta línea se establece el método de selección con el que trabajara el EDA. En la sexta línea se establece el método de remplazo con el que trabajara. En la séptima línea se establece el método con el que se crea la primera generación. En la octava línea solo se le indica a MATEDA para que no muestre ninguna información en la consola. La novena línea se encarga de ejecutar MATEDA.

3.4. Experimentación

Para la experimentación cada secuencia que contiene el $C\vec{S}C$, ver Figura 3.15, se realizó lo siguiente:

1. Se aseguró que las $\vec{SC}^i$ no contuviera **gaps** en su interior.
2. Se separa cada aminoácido contenido en $\vec{SC}^i$.
3. Se convierten los aminoácidos en sus equivalentes numéricos.
4. Se obtiene la $\vec{lsc}^i$ de $\vec{SC}^i$.
5. Se genera el nombre de la carpeta donde se almacenan los resultados.
6. Se inicializa la variable **Card**, variable utilizada por MATEDA.
7. Se verifica que existe la carpeta donde se almacenan los resultados y en caso de que no exista la crea.

Después para cada $\vec{SC}^i$ se realiza el número de experimentos indicados en la variable **numeroExperimentos**. Los valores de las variables que se utilizaron para los experimentos son:

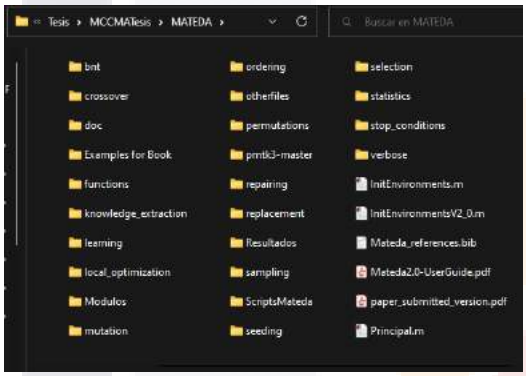
- **PopSize** = 100
- **F** = 'Compatibilidad2'
- **maxgen** = 2000
- **numeroExperimentos** = 30

Los resultados obtenidos de la experimentación los podemos encontrar en su carpeta, en la Figura 3.16 se observa la carpeta donde se almacenan los resultados y como acceder

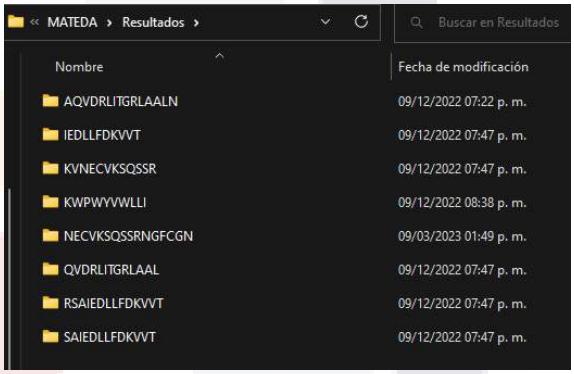
a la carpeta. Se observa el nombre que se le asignó a la carpeta, esto se encuentra cuando se le da valor a la variable **carpeta**, Figura 3.16a, se dirige a la carpeta que contiene MATEDA, Figura 3.16b, se localiza la carpeta que almacena los resultados y se accede a ella, Figura 3.16c, en esta carpeta se observan carpetas con el nombre de cada $\vec{SC}^{i}$, para este caso se accede a la carpeta $AQVDRLITGRLAALN$ y se observan los archivos que contiene los resultados, Figura 3.16d.

```
carpeta = strcat('Resultados','\',auxString);
```

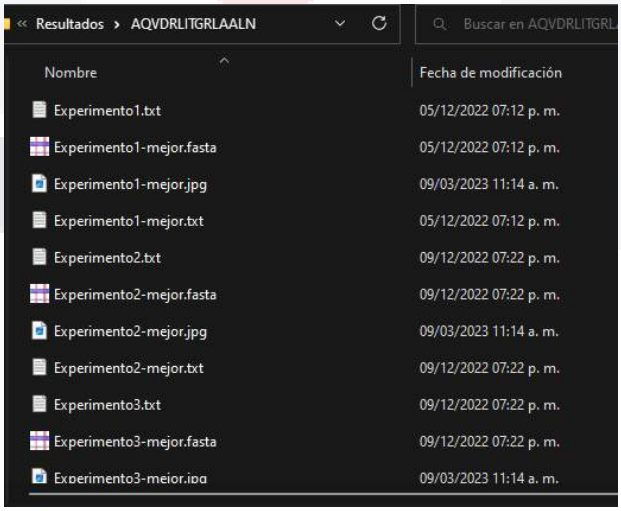
(a) Asignación del nombre de la carpeta, el nombre de la carpeta es el texto que se encuentra entre comillas simples.



(b) Carpeta principal de MATEDA.



(c) Interior de la carpeta que contiene los resultados.



(d) Interior de la carpeta $AQVDRLITGRLAALN$.

Figura 3.16: Resultados obtenidos de MATEDA. Autoría propia.

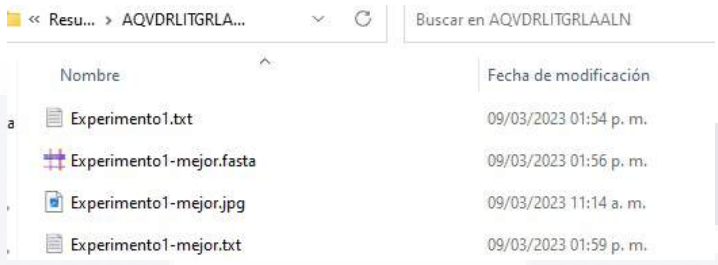
3.5. Resultados

Se ubica la carpeta que almacena los resultados generados por MATEDA. Dentro de la carpeta se verifica que contenga las carpetas referentes a las $\vec{SC}^i$ del $C\vec{S}C$. El número de archivos que contienen estas carpetas, dependen directamente del número de experimentos que se realizaron. Por cada experimento que se realiza se generan 4 archivos con la misma nomenclatura y cada uno contiene información específica. La asignación del nombre a los archivos que almacenan los resultados se asigna en la quinta parte del archivo **RunEDA.m** o **RunEDATesis.m** en la variable **nombre_Original**. El valor que se le asigna a la variable **nombre_Original** no es el nombre final que contendrá los resultados, ya que se necesita diferenciar cada archivo que se genera y además se debe de diferenciar el nombre de cada experimento. En la Figura 3.17 se observan cada uno de los archivos generados por MATEDA para la secuencia $AQVDRLITGRLAALN$. La forma en que se genera cada nombre para cada archivo y la información que contiene se muestra a continuación:

- **nombre_Original** + numero_Experimento + .txt: Archivo .txt que almacena la población de cada generación del experimento, además de los valores de compatibilidad por Carga, Peso e Hidropaticidad. Ver Figura 3.17b.
- **nombre_Original** + numero_Experimento + -mejor.fasta: Archivo tipo fasta que contiene el mejor individuo por cada objetivo de compatibilidad por generación. Ver Figura 3.17c.
- **nombre_Original** + numero_Experimento + -mejor.jpg: Contiene un gráfico del comportamiento del mejor individuo de cada objetivo por generación. Ver Figura 3.17d.
- **nombre_Original** + numero_Experimento + -mejor.txt: Contiene el mejor indi-

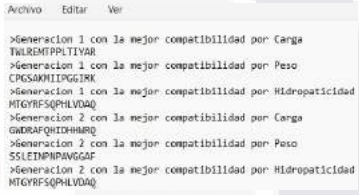
viduo de cada objetivo por generación, junto con sus valores de compatibilidad. Ver Figura 3.17e.

En la Figura 3.17 se observa el contenido de la carpeta $AQVDRLITGRLAALN$, además del interior de los 4 archivos generados por un experimento, ver Figura 3.17a.

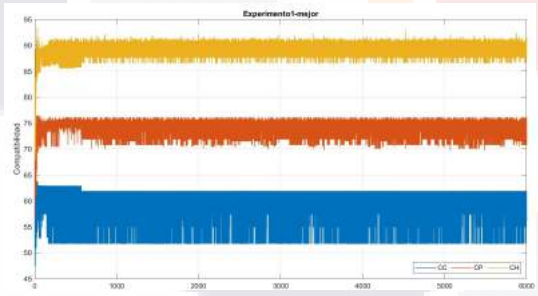


(a) Interior de la carpeta $AQVDRLITGRLAALN$.

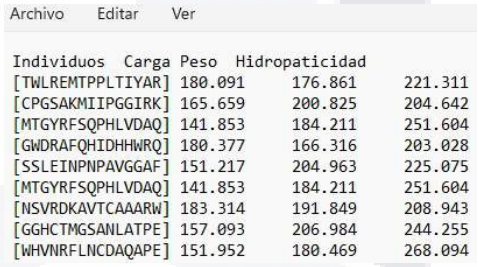
(b) Interior del archivo **nombre_Original** + numero_Experimento + .txt.



(c) Interior del archivo **nombre_Original** + numero_Experimento + -mejor.fasta.



(d) Gráfica **nombre_Original** + numero_Experimento + -mejor.jpg.



(e) Interior del archivo **nombre_Original** + numero_Experimento + -mejor.txt.

Figura 3.17: Archivos de la carpeta $AQVDRLITGRLAALN$. Autoría propia.

El número de archivos que se obtienen depende directamente del ns que contiene el $C\vec{S}C$, ya que por cada experimento que se realice a una $\vec{SC}^i$ se generan 4 archivos, mencionados anteriormente. En la Tabla 3.5 se observa un resumen de los experimentos realizados, así como los archivos generados.

$\vec{SC}^i$	lsc^i	Número de experimentos	Archivos generados
$QVDRLITGRLAAL$	13	30	120
$KVNECVKSQSSR$	12	30	120
$IEDLLFDKVVT$	11	30	120
$NECVKSQSSRNGFCGN$	16	30	120
$RSAIEDLLFDKVVT$	14	30	120
$AQVDRLITGRLAALN$	15	30	120
$SAIEDLLFDKVVT$	13	30	120
$KWPWWVWLLI$	10	30	120
Total	104	240	960

Tabla 3.5: Resultados. Concentrado del $C\vec{S}C$ que se utilizaron en MATEDA, el número de experimentos que se realizaron y el número de archivos que se generaron.

Como se observa en la Tabla 3.6, se tiene en la 1° columna las $\vec{R}$, en la 2°, 3° y 4° columna se contiene la compatibilidad de $\vec{R}$ con $\vec{CS}$, pero este valor de compatibilidad no está en porcentaje, para calcular el porcentaje de compatibilidad se deben aplicar las siguientes ecuaciones, para calcular el porcentaje de compatibilidad por carga se utiliza la Ecuación 3.8, para calcular el porcentaje de compatibilidad por peso se utiliza la Ecuación 3.9 y para calcular el porcentaje de compatibilidad por hidropaticidad se utiliza la Ecuación 3.10.

$$CompatibilidadCarga\,\% = \frac{CompatibilidadCarga \times 100}{19.9406 \times \vec{lsc}} \tag{3.8}$$

$$CompatibilidadPeso\,\% = \frac{CompatibilidadPeso \times 100}{19.6723 \times \vec{lsc}} \tag{3.9}$$

$$CompatibilidadHidropaticidad\,\% = \frac{CompatibilidadHidropaticidad \times 100}{20 \times \vec{lsc}} \tag{3.10}$$

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
AQVKDAGPGDAAALK	185.080	208.874	268.094	1
GNASDGGPGDGGGGQ	171.521	224.546	262.538	1
GQVKRAGPGDGGALK	168.142	208.948	274.547	1
AHGKDGGKADGAGPK	189.415	212.979	250.528	2
GSGKDGGGGDGGGVK	184.753	224.389	257.340	2
ANGKDGIGGDVAGVQ	180.948	212.925	271.142	2
GHLRDAVPGDAAGLK	188.950	204.005	262.358	3
GPGKDGGGGDAAGGK	185.263	224.687	254.472	3
GHLKDAVPGDAAGLK	186.525	206.101	267.557	3

Tabla 3.6: $\vec{R}$ altamente compatibles con la $\vec{SC} = AQVDRLITGRLAALN$. Se muestran los $\vec{R}$ más compatibles de la última generación del experimento 1, 2 y 3.

La Tabla 3.6 completa se encuentra en los Anexos.

Capítulo 4

Discusión de resultados

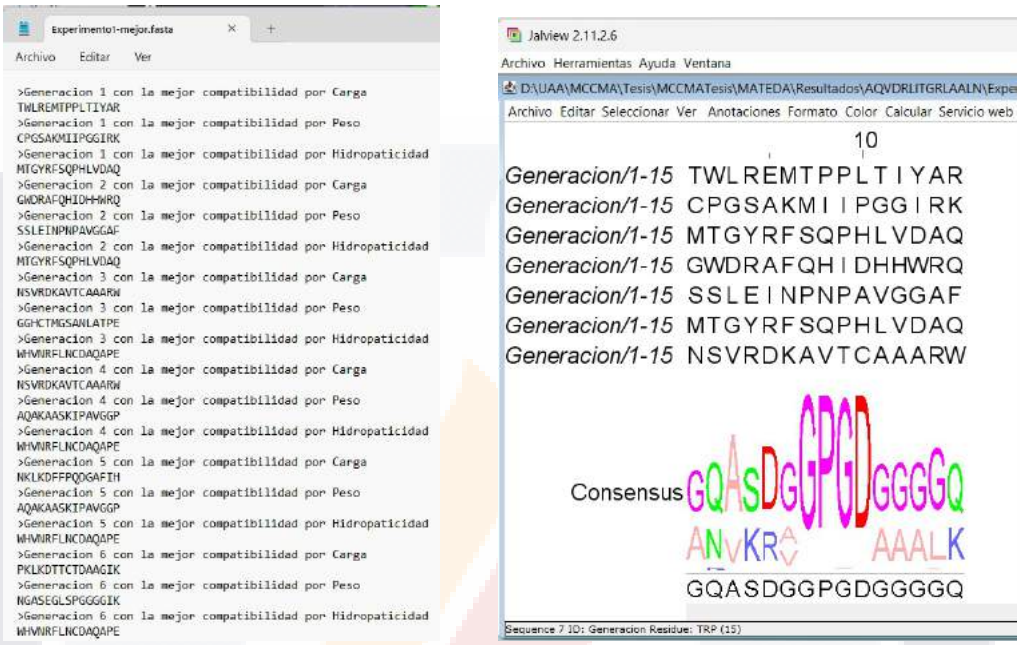
En este capítulo se habla sobre los resultados obtenidos por MATEDA, se analiza los resultados obtenidos $\forall \vec{SC}^i \in C\vec{S}C, i = 1, 2, \ldots, ns$ que se le ingresó a MATEDA. Se analizan todos los archivos generados por MATEDA agrupándose en 4 categorías. Cada categoría en la que se agrupan los datos corresponde al tipo de archivo generado. Para realizar el análisis, las categorías se asignan de la siguiente manera:

- **Resultados.txt**: En esta categoría se tienen los archivos que almacena la población de cada generación del algoritmo. Ver Figura 3.17b que ilustra el contenido del archivo.

- **Resultados-mejor.fasta**: En esta categoría se agrupan los archivos en formato fasta. Estos archivos almacenan la mejor secuencia generada por objetivo de cada generación que tiene el algoritmo. Ver Figura 3.17c que ilustra el contenido del archivo, en la Figura 4.1 se compara el archivo **Experimento1-mejor.fasta** de la carpeta $AQVDRLITGRLAALN$ que se abrió con dos programas diferentes, en la Figura 4.1a se observa el archivo abierto con la aplicación de bloc de notas y en la Figura 4.1b se observa el archivo abierto con Jalview [Waterhouse et al., 2009].

- **Resultados-mejor.jpg**: En esta categoría se analizan las imágenes que se generaron. Cada imagen es una gráfica elaborada con los valores de compatibilidad de las mejores secuencias por generación, el número de secuencias que integran a esta gráfica depende directamente del número de generaciones que realizo MATEDA. Ver Figura 3.17d.

- **Resultados-mejor.txt**: En esta categoría se colocan los archivos que contienen el mejor individuo de cada objetivo por generación. Ver Figura 3.17e que ilustra el contenido del archivo.



(a) Archivo **Experimento1-mejor.fasta** abierto con bloc de notas.

(b) Archivo **Experimento1-mejor.fasta** abierto con Jalview.

Figura 4.1: Comparación de vista de archivo. Autoría propia.

Resultados.txt:

En estos archivos se observa el comportamiento de la población con la que trabajo MATEDA. $\forall \vec{SC}^i \in C\vec{S}C, i = 1, 2, \ldots, ns$ se observa que la primera generación que se obtiene es de forma aleatoria por lo cual la población es variada sin mostrar alguna preferencia a un aminoácido. Esto a primera vista no representa un problema, pero si lo es, ya que en la naturaleza no se han encontrado combinaciones específicas de ciertos aminoácidos, por lo cual comenzar la primera generación en su totalidad de forma aleatoria no es muy recomendable, ya que existe una pequeña probabilidad de que se generen secuencias irreales.

Como se observan en los archivos resultantes de los experimentos, al paso que van avanzando las generaciones en el experimento, la población comienza a presentar subsecuencias de 3 a 4 aminoácidos en diferentes individuos de la población. Además, se observa que para algunas columnas se comienza a tener una tendencia a un aminoácido en específico. En las últimas generaciones, se observa que la variación entre la población de una generación con la población antecesora es poca, es decir ya no hay un cambio significativo entre los individuos que conforman la población.

En la Figura 4.2 se observan los valores de compatibilidad obtenidos por la población de individuos en la 100^{ma} generación del primer experimento de la $\vec{SC}^{i} = AQVDRLITGRLAALN$, donde se observan las soluciones no dominadas representadas por círculos rojos, mientras que las soluciones dominadas son representadas por cuadrados azules.

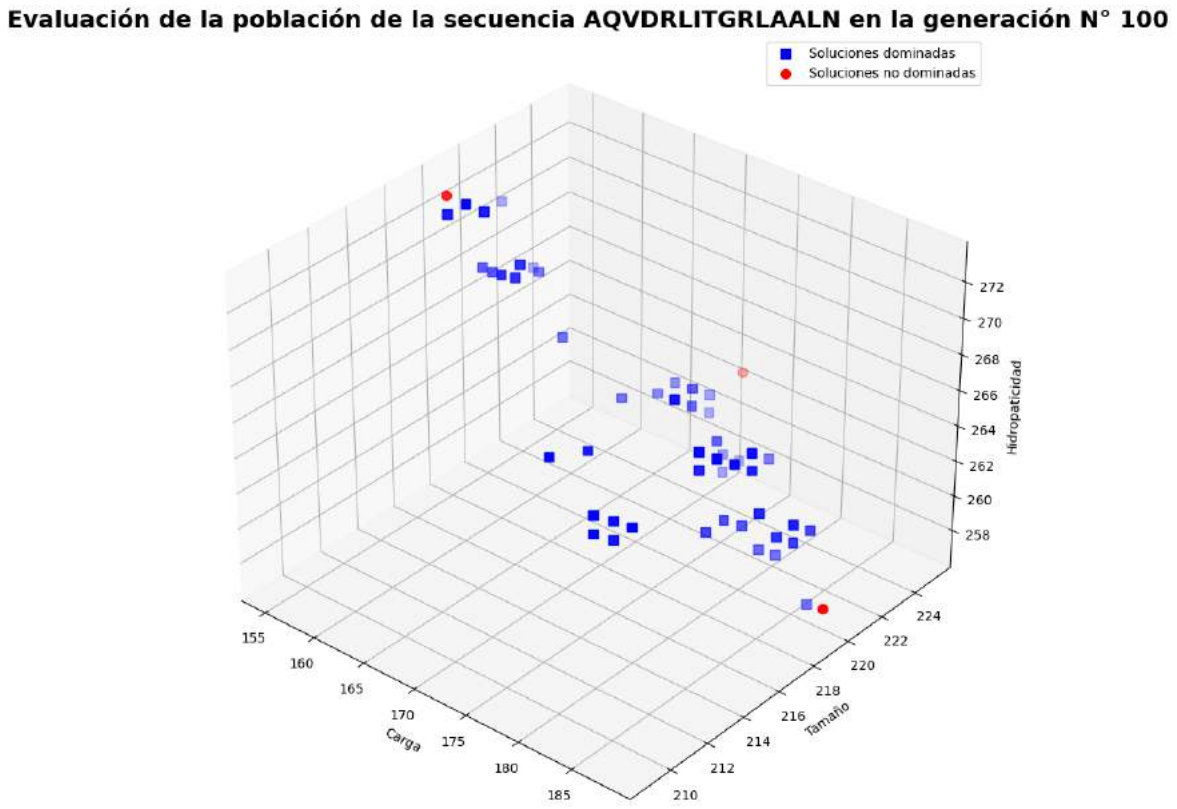


Figura 4.2: Evaluación de la población. Autoría propia.

Resultados-mejor.fasta:

Para los archivos .fasta son abiertos a través del software de Jalview. Se observa una misma subsecuencias de aminoácidos en los diferentes archivos generados a partir de una $\vec{SC}^i$. Se analizó los aminoácidos contenidos en cada archivo y se observa que, para cada archivo, los aminoácidos que conforman la columna tienen poca variación, donde se encuentran aminoácidos con una homología menor al 5 %, mientras que los aminoácidos predominantes en la columna, llegaban a tener una homología cercana al 99 %. Por lo general la homología del aminoácido predominante por columna ronda entre $30\,\% \pm 20$. Las primeras secuencias que se tienen en estos archivos pueden ser no tan compatibles con $\vec{SC}^i$, ya que estas secuencias fueron generadas de forma aleatoria. Por tal razón, al analizar la homología por columna de los aminoácidos, sean estas secuencias las responsables de que se encuentren homologías menores al 5 %. Las secuencias contenidas en los archivos posiblemente son las secuencias que tiene la mejor compatibilidad con $\vec{SC}^i$ en uno de sus tres posibles compatibilidades, pero esto no significa que la secuencia encontrada tenga una veracidad biológica, es decir que en la naturaleza no se encuentra la secuencia igual a como el algoritmo la generó.

Resultados-mejor.jpg:

Las imágenes generadas por MATEDA, presentan una dificultad para su análisis por el número de secuencias que se utilizaron para generar el gráfico. La ventaja de tener de esta forma la información es que se puede observar el comportamiento que tiene el mejor individuo a lo largo de las generaciones. Se observa en el gráfico una oscilación de los valores de compatibilidad de las secuencias. Estos valores se ven de manera general que el valor de compatibilidad por hidropaticidad es mayor a los valores de compatibilidad por cargas y por peso. Si para la generación del gráfico se tomaran un menor número de secuencias, se podrá analizar de mejor manera la compatibilidad de las secuencias.

Resultados-mejor.txt:

Las secuencias que conforman este archivo se observan que sus valores de compatibilidad presentan un patrón que se repite cada 3^{ra} secuencia. Cada bloque de 3 secuencias contenidas en el archivo representa la secuencia con la mayor compatibilidad con una de sus tres propiedades fisicoquímicas. La primera secuencia del bloque tiene la mejor compatibilidad por carga, la segunda secuencia del bloque tiene la mejor compatibilidad por peso y la tercera secuencia del bloque tiene la mejor compatibilidad por hidropaticidad. Es decir, el primer bloque contiene las mejores secuencias de la primera generación, el segundo bloque contiene las mejores secuencias de la segunda generación y si sucesivamente hasta que se complete el número de generaciones.

De manera general, se observa que sin importar la $\vec{SC}$ o el valor de lsc que se ingresó a MATEDA, la compatibilidad por hidropaticidad es la mayor. Esto ocurre porque al ser analizadas las matrices MCC, MCP y MCH se encontró que las matrices MCP MCH no son tan excluyentes entre sí y la combinación de algunos aminoácidos en específicos dan valores de compatibilidad muy buenos para ambas matrices. Si se extrae una de estas matrices y se añade otra condicional que se utilice como función objetivo y esta sea excluyente de las dos matrices de compatibilidad restantes, probablemente los resultados que se obtienen de MATEDA tengan un comportamiento más homogéneo.

Capítulo 5

Conclusiones

En este capítulo se abordan las conclusiones que se llegaron con la consolidación del trabajo de investigación. En la Sección 5.1 se habla de los objetivos cumplidos y una pequeña explicación del porqué se cumplió el objetivo. Para la Sección 5.2 se detalla las contribuciones que deja el cumplimiento de esta investigación a la sociedad en general. Para terminar, en la Sección 5.3 se habla del trabajo a futuro que se queda pendiente por realizar.

Las conclusiones generales a las que se llegaron al finalizar el trabajo de investigación se listan a continuación.

- MATEDA es una excelente alternativa si se desea trabajar con EDAs y no se requiere programar en su totalidad el algoritmo.
- Al analizar los archivos de resultados, se encontró que el algoritmo presenta una convergencia antes de la generación número 200, por lo cual se pueden realizar experimentos con un número de generaciones que ronden entre los 150 a 250 generaciones y obtener resultados muy similar.
- La metodología puede trabajar con cualquier secuencia de aminoácidos, solo se necesite que esté en el formato especificado en la Sección 3.3.2.4.
- La generación de los individuos que conforman la primera generación con la que trabaja MATEDA se le debe añadir reglas para que no genere individuos tan aleatorios y se generen individuos con secuencias de aminoácidos reales.

- Los objetivos utilizados en el problema multiobjetivo se deben de replantear o modificar porque provocan un sesgo en los resultados, ya que dos de los tres objetivos utilizados no son completamente excluyentes entre sí y comparten objetivos, es decir, una determinada combinación de aminoácidos es más compatibles en dos objetivos del problema, mientras otras combinaciones solo son altamente compatible en un solo objetivo del problema.

5.1. Objetivos Cubiertos

Objetivos específicos:

- La familia de *coronaviridae* se encuentra poco reportada en la literatura y fue nuestro grupo de estudio.
- La metaheurística con la que se trabajó son los EDAs, en el paradigma multiobjetivo, utilizando la herramienta de MATEDA.
- El problema se modeló con un enfoque multiobjetivo, lo cual permitió generar soluciones con MATEDA.
- A partir de MATEDA, se generaron módulos auxiliares, que permitieron abordar el PDM con un enfoque multiobjetivo.
- La implementación de la metodología fue correcta, ya que se logró obtener secuencias altamente compatibles con una secuencia conservada.
- Las secuencias obtenidas de la metodología en diferentes experimentos teniendo la misma $\vec{SC}$ presentan una homología alta.

5.2. Contribuciones

Se hace una aportación de una metodología funcional con la cual se generan secuencias de aminoácidos altamente compatibles por sus propiedades fisicoquímicas, en la cual sin importar la longitud de la secuencia realizara la búsqueda. La guía desde su descarga, hasta su utilización están documentadas en el Capítulo 3 del presente documento.

5.3. Trabajo a futuro

Las actividades que se quedan pendientes por realizar se listan a continuación.

- Implementación de reglas para la generación de la población inicial.
- Cambio o sustitución de las funciones objetivos para evitar el sesgo en los resultados.
- Añadir una función de parada si se detecta que después de un número determinado de generaciones la mejor solución no presenta una mejora significativa.
- Automatizar el proceso de obtención del $C\vec{S}C$.
- Realizar una comparativa del tiempo utilizado por la metodología en contraste de otra metodología para la búsqueda de motivos.

Glosario

fitness Calidad del individuo. 47, 55, 56, 85

matriz Arreglo bidimensional de tamaño nxm, donde n y m son dos números naturales cualquiera.. 31, 32, 45, 55

precursor Es una sustancia que se necesita para obtener otra diferente a partir de una reacción química. 28

terna Conjunto de tres cosas relacionadas entre sí. 48

variable continua Variable que puede tomar cualquier valor dentro de un intervalo. Su medición es difícil o imprecisa. 50

variable discreta Variable en la cual solo puede tomar un valor de un conjunto de valores finito, no puede tomar un valor intermedio entre dos valores consecutivos. 37, 50

Bibliografía

[Alonso et al., 2003] Alonso, J. L., Mühlenbein, H., and Múgica, P. L. (2003). Algoritmos de estimación de distribuciones en problemas de optimización combinatoria. *Inteligencia artificial: Revista Iberoamericana de Inteligencia Artificial*, 7:149–168.

[Amaru, 2019] Amaru, R. (2019). Capitulo 2.2 aminoácidos y péptidos.

[Arango et al., 2011] Arango, G., Garzón, J. J., Rothlisberger, S., and Dominguez, G. C. (2011). Classification of unaligned sequences based on prototype motifs representation. *2011 6th Colombian Computing Congress, CCC 2011*.

[Araujo and Cervigón, 2009] Araujo, L. and Cervigón, C. (2009). *Algoritmos Evolutivos: un enfoque práctico*. Alfaomega Grupo Editor, S.A. de C.V., primera edición edition.

[Ashraf and Shafi, 2020] Ashraf, F. B. and Shafi, M. S. R. (2020). Mfea: An evolutionary approach for motif finding in dna sequences. *Informatics in Medicine Unlocked*, 21.

[Biro, 2006] Biro, J. (2006). Amino acid size, charge, hydropathy indices and matrices for protein structure analysis. *Theoretical Biology and Medical Modelling*, 3:15.

[Blum and Roli, 2003] Blum, C. and Roli, A. (2003). Metaheuristics in combinatorial optimization. *ACM Computing Surveys*, 35:268–308.

[Calera and Sanz, 2003] Calera, C. G. M. and Sanz, J. S. (2003). *Estructura de Proteinas*. Ariel, Editorial S.A.

[Calvet et al., 2022] Calvet, L., Benito, S., Juan, A. A., and Prados, F. (2022). On the role of metaheuristic optimization in bioinformatics. *International Transactions in Operational Research*.

[Cordero et al., 2009] Cordero, F., Visconti, A., and Botta, M. (2009). A new protein motif extraction framework based on constrained co-clustering. In *Proceedings of the 2009 ACM Symposium on Applied Computing*, pages 776–781. ACM.

[Correa Morales, 2020] Correa Morales, J. A. (2020). Diseño, implementación y prueba de una metodología para el descubrimiento de motivos altamente conservados de proteínas utilizando una metaheurística. Tesina, Universidad Autónoma de Aguascalientes.

[Czeizler et al., 2017] Czeizler, E., Hirvola, T., and Karhu, K. (2017). A graph-theoretical approach for motif discovery in protein sequences. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 14:121–130.

[Darwin, 1859] Darwin, C. (1859). *The Origin of The Species*. John Murray.

[Davey et al., 2010] Davey, N. E., Edwards, R. J., and Shields, D. C. (2010). Estimation and efficient computation of the true probability of recurrence of short linear protein sequence motifs in unrelated proteins. *BMC Bioinformatics*, 11:14.

[Dorigo, 1992] Dorigo, M. (1992). *Optimization, Learning and Natural Algorithms*. PhD thesis, Politecnico di Milano, Italy.

[Dorigo et al., 1996] Dorigo, M., Maniezzo, V., and Colorni, A. (1996). Ant system: optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 26:29–41.

[Feduchi et al., 2010] Feduchi, E., Blasco, I., Romero, C., and Yáñez, E. (2010). *Bioquimica Conceptos esenciales*. Médica Panamericana, 1 edition.

[Flechsig, 2012] Flechsig, H. (2012). *Structurally Resolved Coarse-Grained Modeling of Motor Protein Dynamics*. PhD thesis, Technische Universität Berlin, Fakultät II - Mathematik und Naturwissenschaften.

[Gallegos, 2019] Gallegos, J. E. R. (2019). Algoritmo eficiente para la agrupación de proteínas en familias basado en mejores aciertos bidireccionales y el árbol filogenético. Tesina, Universidad Autónoma de Aguascalientes.

[Galvis Motoa et al., 2021] Galvis Motoa, S. I., Ponce de León Sentí, E. E., Álvarez Tostado, E. M. M., and Cuellar Garrido, L. D. (2021). Acquisition and preprocessing of proteomic data for bidirectional best hits methodology: a study case in the coronaviridae family. *Research in Computing Science*, 150.

[Glasgow, 1998] Glasgow, J. (1998). Classification of protein structure motif conformations. *SIGBIO Newsl.*, 18:12.

[González Álvarez and Vega Rodríguez, 2013] González Álvarez, D. L. and Vega Rodríguez, M. A. (2013). Analysing the scalability of multiobjective evolutionary algorithms when solving the motif discovery problem. *Journal of Global Optimization*, 57:467–497.

[González Álvarez et al., 2015] González Álvarez, D. L., Vega Rodríguez, M. A., and Álvaro Rubio Largo (2015). Multiobjective optimization algorithms for motif discovery in dna sequences. *Genetic Programming and Evolvable Machines*, 16:167–209.

[Huo et al., 2010] Huo, H., Zhao, Z., Stojkovic, V., and Liu, L. (2010). Optimizing genetic algorithm for motif discovery. *Mathematical and Computer Modelling*, 52:2011–2020.

[Irurozki et al., 2018] Irurozki, E., Ceberio, J., Santamaria, J., Santana, R., and Mendiburu, A. (2018). Algorithm 989: perm_mateda: A matlab toolbox of estimation of

distribution algorithms for permutation-based combinatorial optimization problems. *ACM Transactions on Mathematical Software*, 44:1–13.

[Jackups and Liang, 2010] Jackups, R. and Liang, J. (2010). Combinatorial analysis for sequence and spatial motif discovery in short sequence fragments. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 7:524–536.

[Jordán and Jordán, 2015] Jordán, C. I. and Jordán, C. J. (2015). *MBMEDA: An Application of Estimation of Distribution Algorithms to the Problem of Finding Biological Motifs*, volume 9107, pages 39–46. Springer Verlag.

[Koonin et al., 1994] Koonin, E. V., Mushegian, A. R., Tatusov, R. L., Altschul, S. F., Bryant, S. H., Bork, P., and Valencia, A. (1994). Eukaryotic translation elongation factor l y contains a glutathione transferase domain-study of a diverse, ancient protein superfamily using motif search and structural modeling. *Protein Science*, 3:2045–2054.

[Larkin et al., 2007] Larkin, M., Blackshields, G., Brown, N., Chenna, R., McGettigan, P., McWilliam, H., Valentin, F., Wallace, I., Wilm, A., Lopez, R., Thompson, J., Gibson, T., and Higgins, D. (2007). Clustal w and clustal x version 2.0. *Bioinformatics*, 23:2947–2948.

[Larrañaga et al., 1999] Larrañaga, P., C.M.H., K., Murga, R., Inza, I., and Dizdarevic, S. (1999). Genetic algorithms for the travelling salesman problem: A review of representations and operators. *Artificial Intelligence Review*, 13:170.

[Li et al., 2008] Li, G., Chan, T.-M., Leung, K.-S., and Lee, K.-H. (2008). An estimation of distribution algorithm for motif discovery. In *2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)*, pages 2411–2418. IEEE.

[Li et al., 2010] Li, G., Chan, T.-M., Leung, K.-S., and Lee, K.-H. (2010). A cluster refinement algorithm for motif discovery. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 7:654–668.

[Li et al., 2002] Li, M., Ma, B., and Wang, L. (2002). Finding similar regions in many sequences. *Journal of Computer and System Sciences*, 65:73–96.

[Lones and Tyrrell, 2005] Lones, M. A. and Tyrrell, A. M. (2005). The evolutionary computation approach to motif discovery in biological sequences. In *Proceedings of the 7th Annual Workshop on Genetic and Evolutionary Computation*, GECCO '05, page 1–11, New York, NY, USA. Association for Computing Machinery.

[Lones and Tyrrell, 2007] Lones, M. A. and Tyrrell, A. M. (2007). Regulatory motif discovery using a population clustering evolutionary algorithm. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 4:403–414.

[Mckee and MacKee, 2014] Mckee, T. and MacKee, J. R. (2014). *Bioquimica : las bases moleculares de la vida*. McGraw-Hill Education.

[Mirjalili et al., 2014] Mirjalili, S., Mirjalili, S. M., and Lewis, A. (2014). Grey wolf optimizer. *Advances in Engineering Software*, 69:46–61.

[Motoa, 2022] Motoa, S. I. G. (2022). Clusterización de proteínas mediante metaheurísticas multiobjetivo en la familia coronaviridae. Master's thesis, Universidad Autónoma de Aguascalientes.

[Murray et al., 2013] Murray, R. K., Bender, D. A., Botham, K. M., Kennelly, P. J., Rodwell, V. W., and Anthony, W. P. (2013). *Harper Bioquimica Ilustrada*. McGraw-Hill Interamericana de España S.L., 28 edition.

[Mühlenbein and Paaß, 1996] Mühlenbein, H. and Paaß, G. (1996). From recombination of genes to the estimation of distributions i. binary parameters. *Parallel Problem Solving from Nature — PPSN IV*, 1141:178–187.

[Ponce de León Sentí et al., 2017] Ponce de León Sentí, E. E., Diaz Diaz, E., Guardado Muro, H., Cuellar Garrido, L. D., Martinez Guerra, J. J., Torres Soto, A., Torres Soto, M. D., and Hernandez Aguirre, A. (2017). A distance measure for building phylogenetic trees: A first approach. *Research in Computing Science*, 139:149–162.

[Ponce de León Sentí et al., 2021] Ponce de León Sentí, E. E., Reyes Gallegos, J. E., Cuellar Garrido, L. D., Álvarez Tostado, E. M. M., Díaz Díaz, E., Torres Soto, A., Torres Soto, M. D., and Martínez Guerra, J. (2021). Metodología eficiente para obtener cliques de proteínasmediante los mejores aciertos bidireccionale. In *Memorias del Seminario de Investigación UAA*.

[Quiroz and Scherer, 2004] Quiroz, L. O. and Scherer, L. G. C. (2004). Plegamiento de las proteínas: Un problema interdisciplinario. *Rev. Soc. Quím. Méx*, 48:95–105.

[Saha et al., 2019] Saha, T. K., Katebi, A., Dhifli, W., and Hasan, M. A. (2019). Discovery of functional motifs from the interface region of oligomeric proteins using frequent subgraph mining. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 16:1537–1549.

[Santana et al., 2009] Santana, R., Echegoyen, C., Mendiburu, A., Bielza, C., Lozano, J., Larranaga, P., Armañanzas, R., and Shakya, S. (2009). Mateda: A suite of eda programs in matlab. Technical report, University of the Basque Country.

[Santana et al., 2010] Santana, R., Echegoyen, C., Mendiburu, A., Bielza, C., Lozano, J. A., Larrañaga, P., Armañanzas, R., and Shakya, S. (2010). Mateda-2.0: A matlab package for the implementation and analysis of estimation of distribution algorithms. *Journal of Statistical Software*, 35.

[Sarkar et al., 2021] Sarkar, T., Raghavan, V. V., Chen, F., Riley, A., Zhou, S., and Xu, W. (2021). Exploring the effectiveness of the tsr-based protein 3-d structural comparison method for protein clustering, and structural motif identification and discovery of protein kinases, hydrolases, and sars-cov-2's protein via the application of amino acid grouping. *Computational Biology and Chemistry*, 92.

[Semwal et al., 2022] Semwal, R., Aier, I., Raj, U., and Varadwaj, P. K. (2022). Pr[m]: An algorithm for protein motif discovery. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 19:585–592.

[Shao et al., 2009] Shao, L., Chen, Y., and Abraham, A. (2009). Motif discovery using evolutionary algorithms. In *2009 International Conference of Soft Computing and Pattern Recognition*, pages 420–425. IEEE.

[Sheth and Kim, 2005] Sheth, H. A. and Kim, S. (2005). Motif discovery for proteins using subsequence clustering. In *Proceedings of the 5th International Workshop on Bioinformatics*, pages 3–6. ACM.

[Sievers and Higgins, 2018] Sievers, F. and Higgins, D. G. (2018). Clustal omega for making accurate alignments of many protein sequences. *Protein Science*, 27:135–145.

[Sievers et al., 2011] Sievers, F., Wilm, A., Dineen, D., Gibson, T. J., Karplus, K., Li, W., Lopez, R., McWilliam, H., Remmert, M., Söding, J., Thompson, J. D., and Higgins, D. G. (2011). Fast, scalable generation of high-quality protein multiple sequence alignments using clustal omega. *Molecular Systems Biology*, 7:539.

[Silver, 2004] Silver, E. A. (2004). An overview of heuristic solution methods. *Journal of the Operational Research Society*, 55:936–956.

[Terfloth, 2009] Terfloth, A. (2009). *Las proteínas: composición química*. [El Cid Editor — apuntes].

[Voet et al., 2007] Voet, D., Voet, J. G., and Pratt, C. W. (2007). *Fundamentos de Bioquímica La vida a nivel molecular*. Medica Panamericana, 2 edition.

[Wang and Scott, 2005] Wang, C. and Scott, S. D. (2005). New kernels for protein structural motif discovery and function classification. In *Proceedings of the 22nd International Conference on Machine Learning*, pages 940–947. ACM Press.

[Wang et al., 2015] Wang, G.-G., Deb, S., and dos S. Coelho, L. (2015). Elephant herding optimization. In *2015 3rd International Symposium on Computational and Business Intelligence (ISCBI)*, pages 1–5. IEEE.

[Waterhouse et al., 2009] Waterhouse, A. M., Procter, J. B., Martin, D. M. A., Clamp, M., and Barton, G. J. (2009). Jalview version 2—a multiple sequence alignment editor and analysis workbench. *Bioinformatics*, 25:1189–1191.

[Xiong, 2012] Xiong, J., editor (2012). *Protein Motifs and Domain Prediction*. Cambridge University Press.

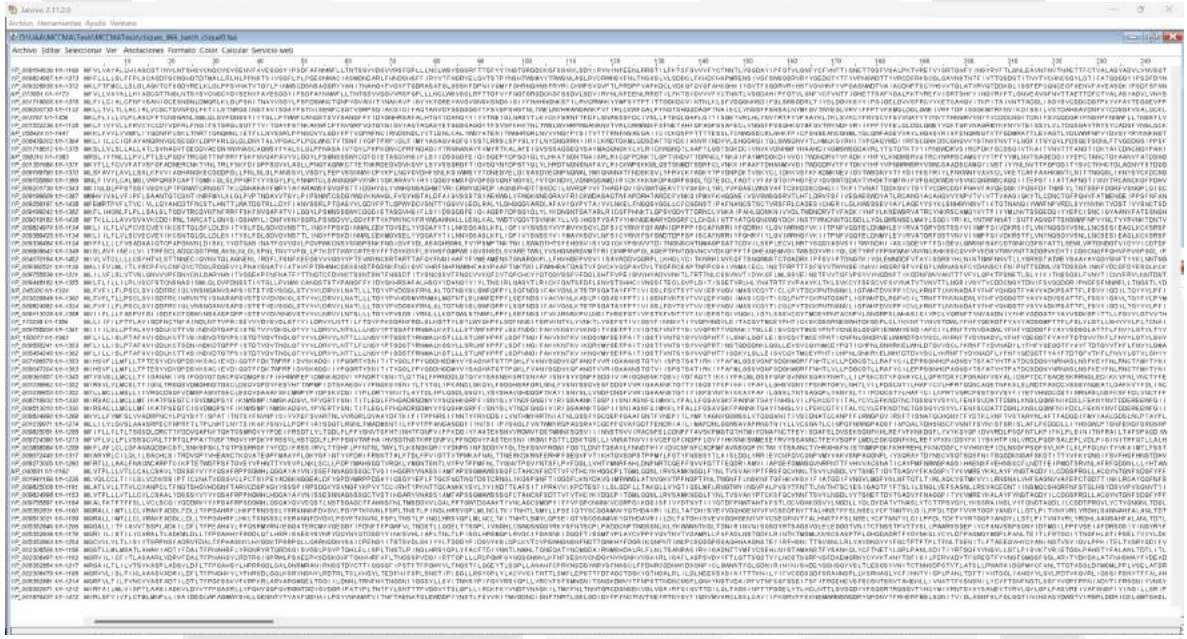
[Álvarez and Rodríguez, 2013] Álvarez, D. L. G. and Rodríguez, M. A. V. (2013). Hybrid multiobjective artificial bee colony with differential evolution applied to motif finding. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 7833 LNCS:68–79.

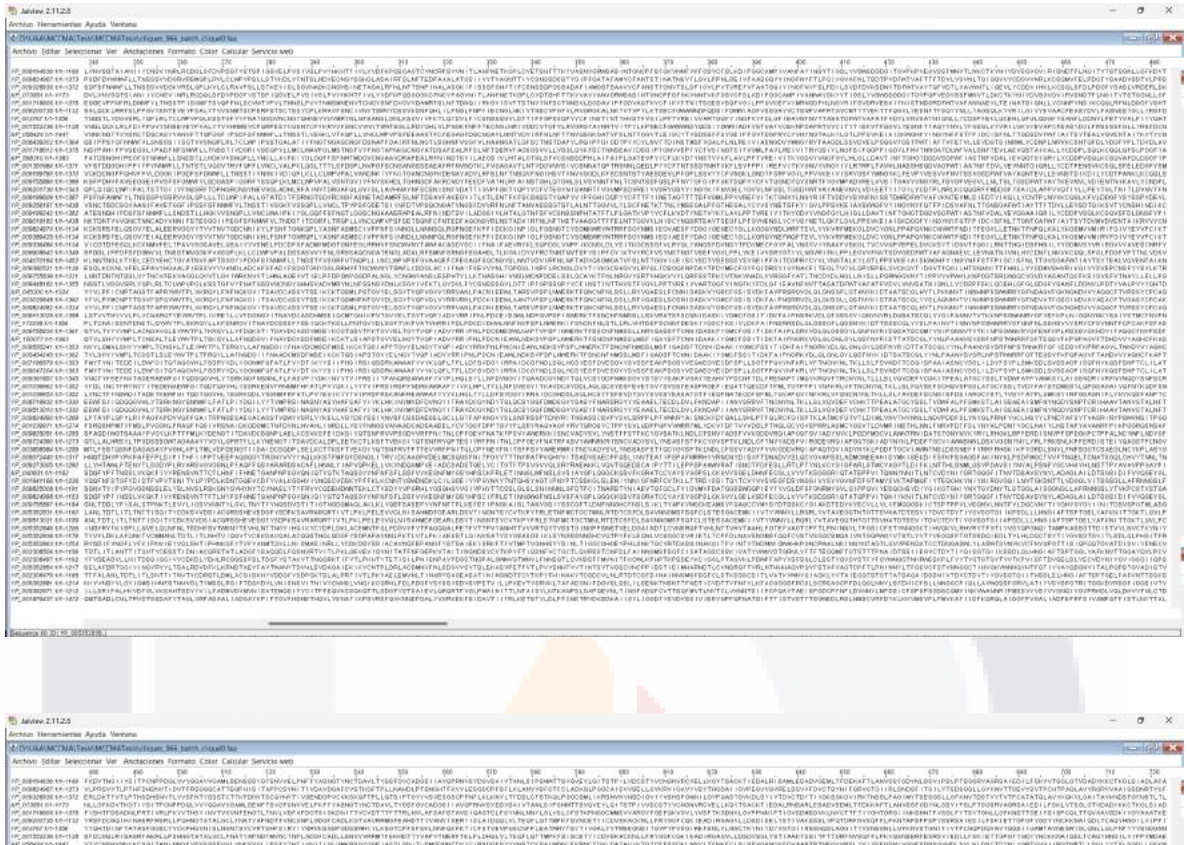
Anexos

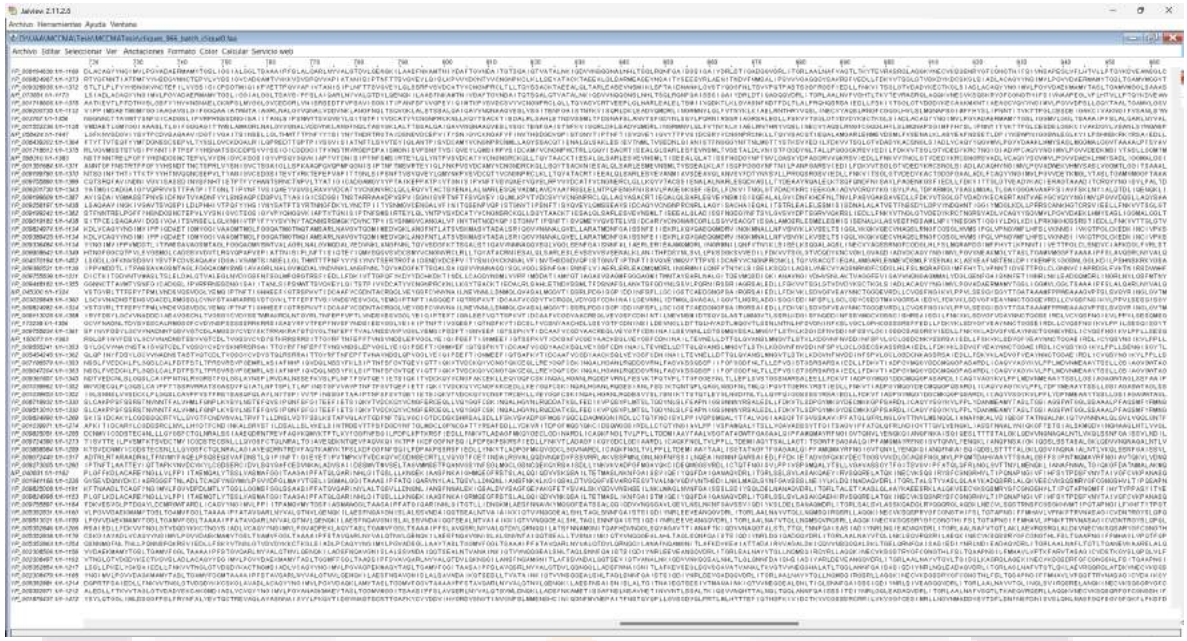
En este capítulo se presentarán anexos que completan el trabajo de investigación.

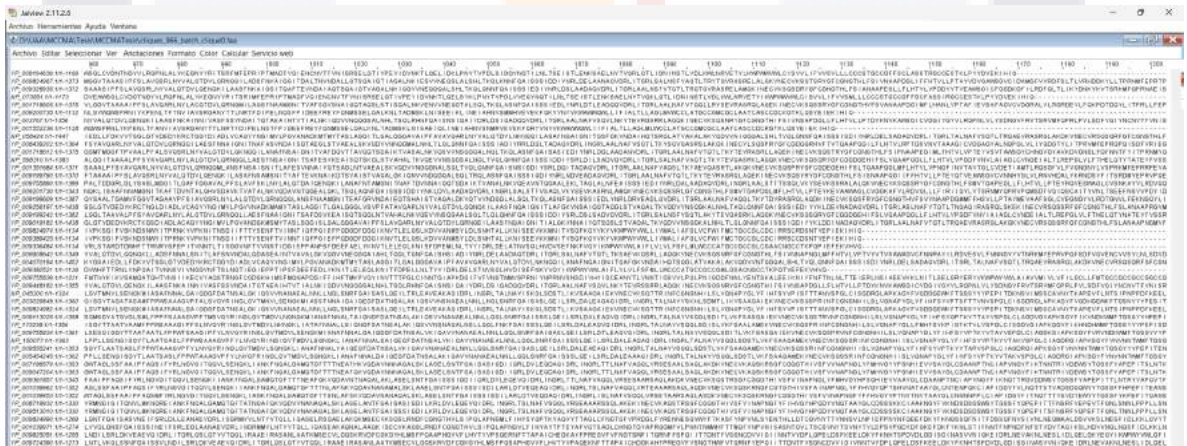
Anexo A

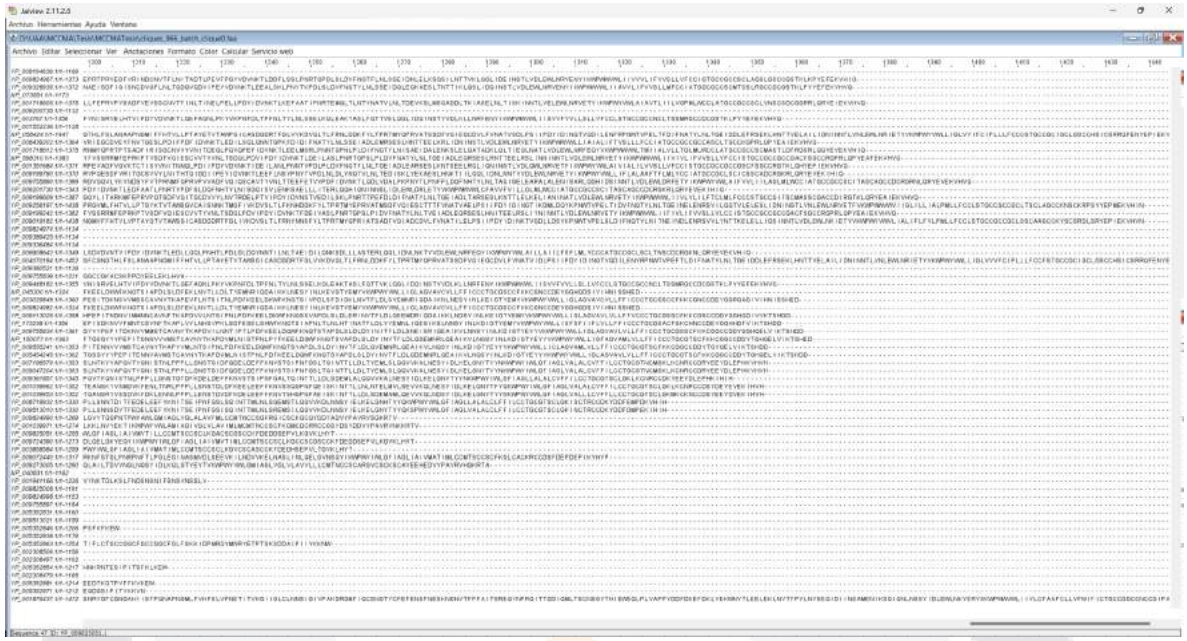
Archivo **cliques_066_batch_clique0.faa.** obtenido del repositorio GitLab abierto con Jalview.





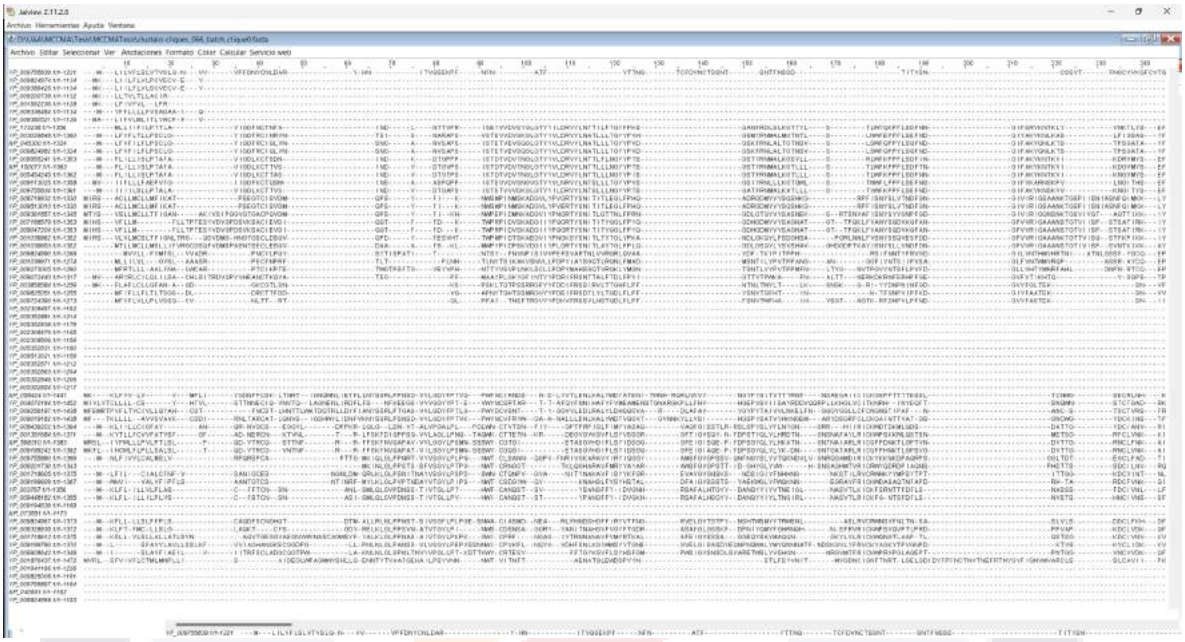


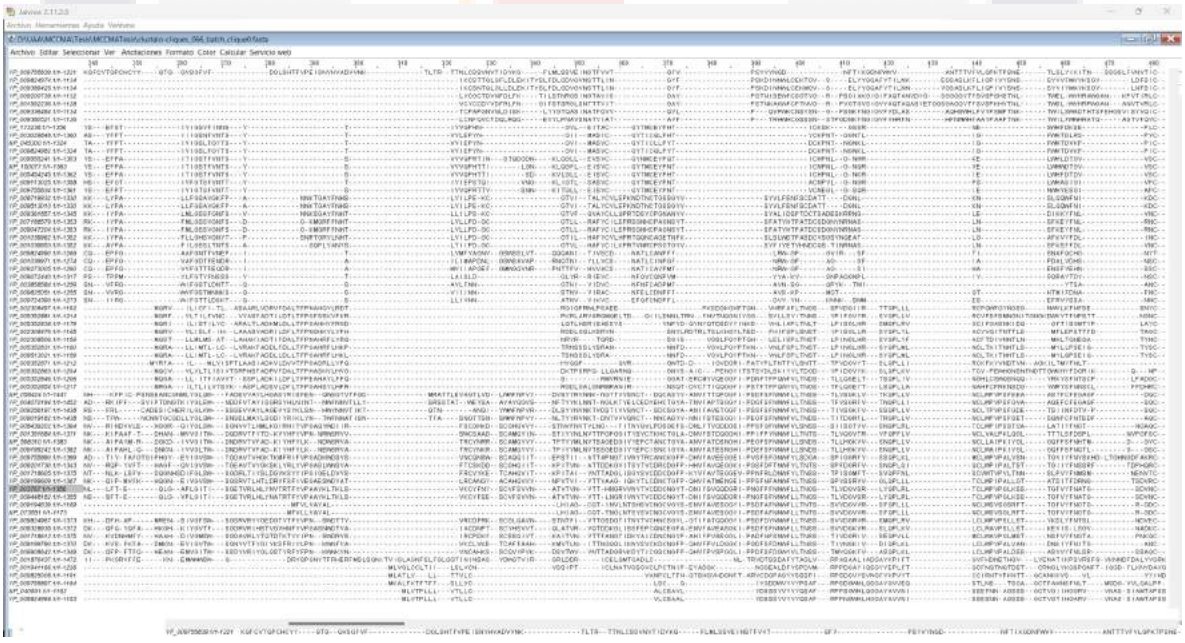


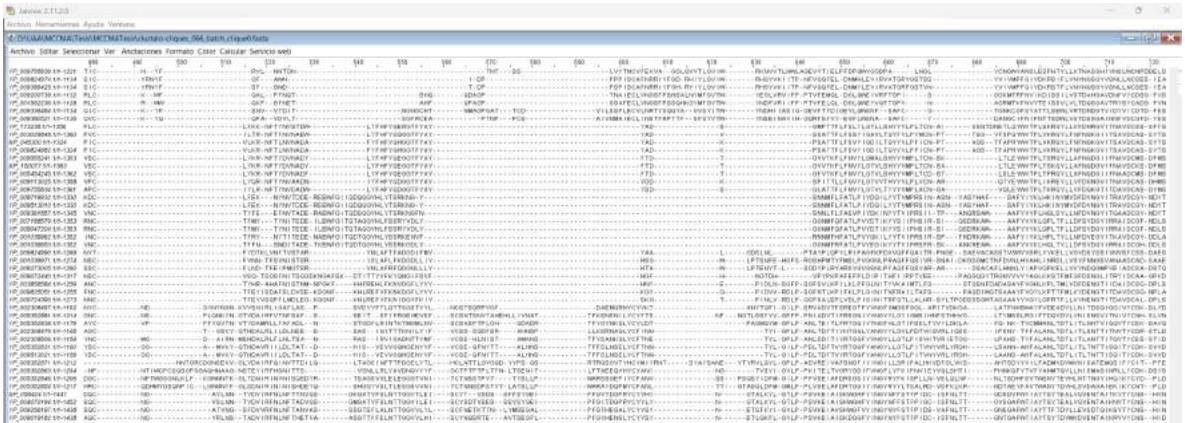


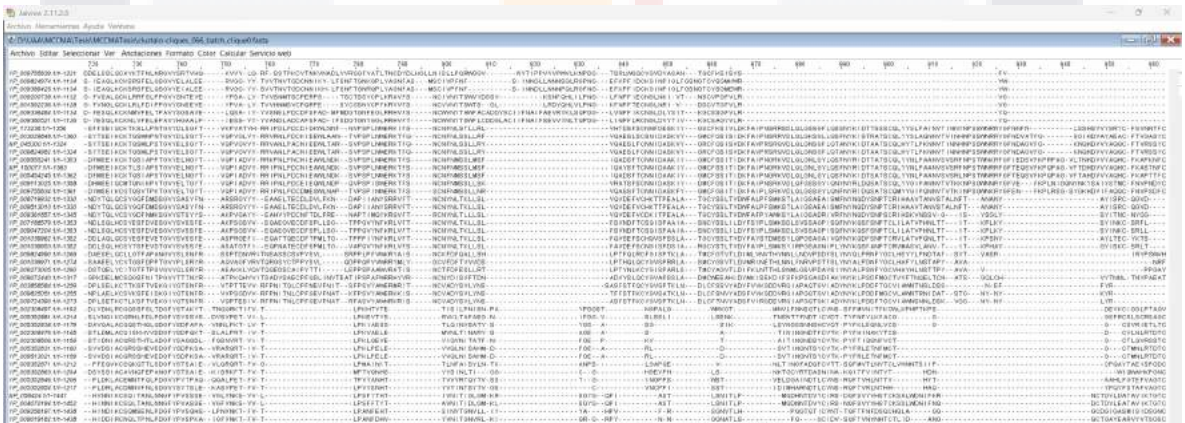
Anexo B

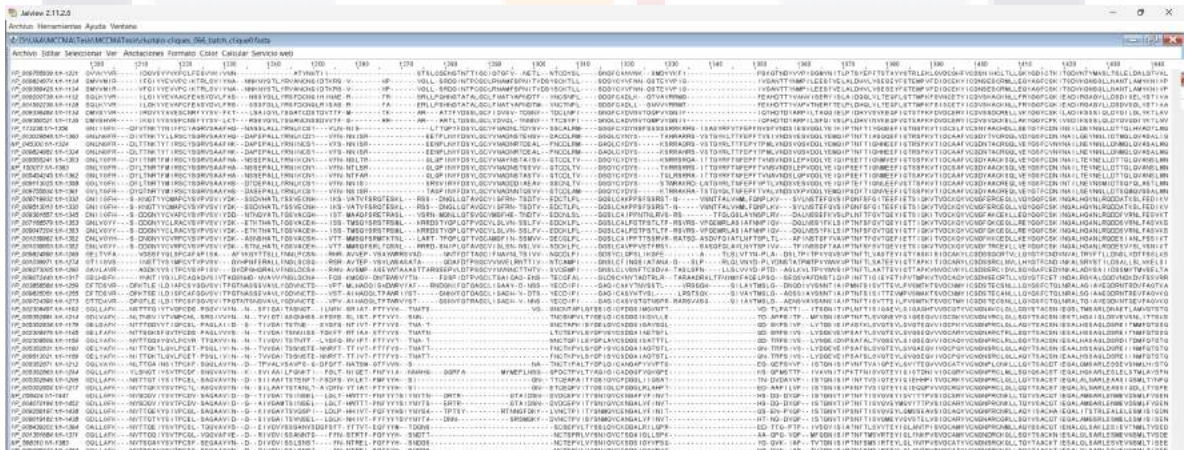
Contenido del archivo **cliques_066_batch_clique0.faa.** aplicado el AMS con ClustalΩ y abierto con Jalview.

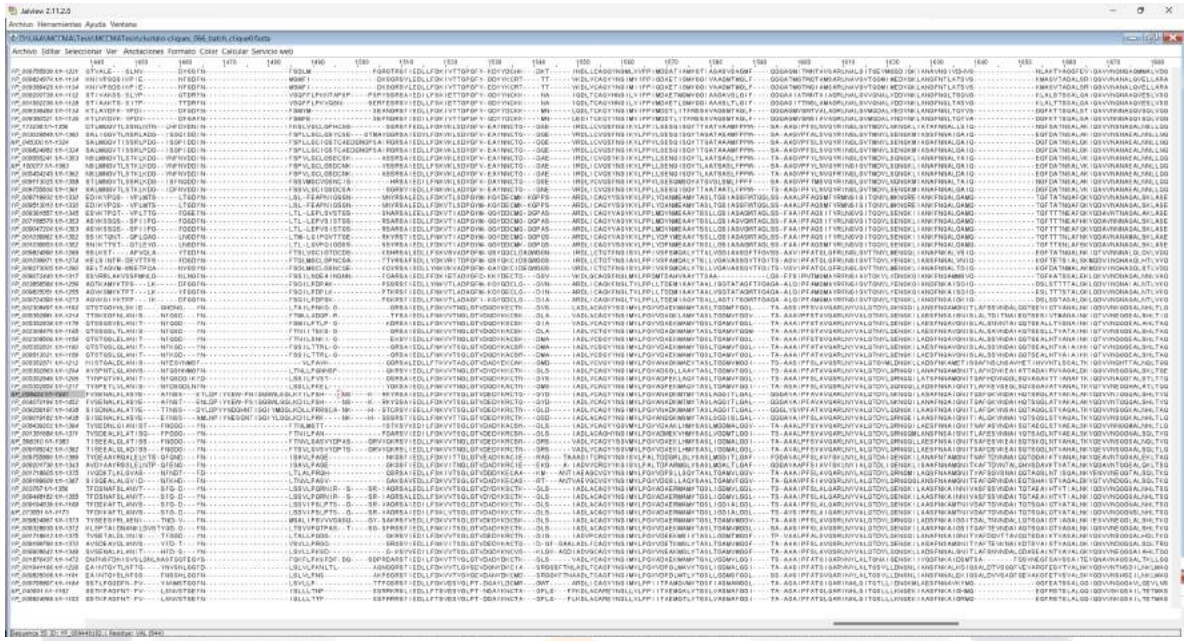


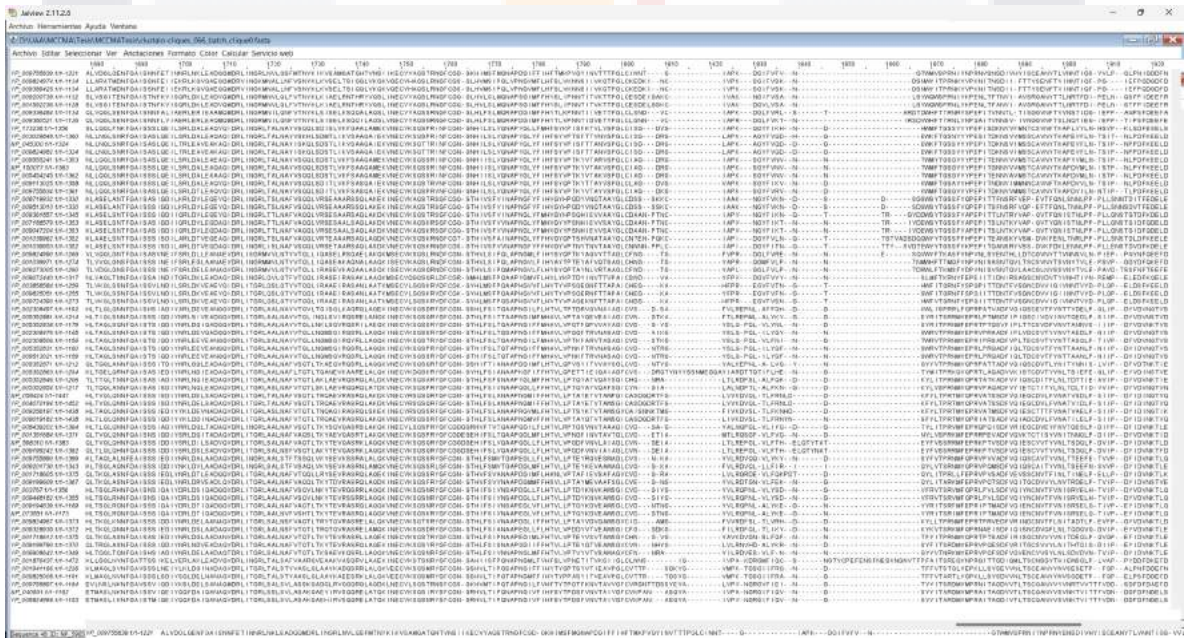


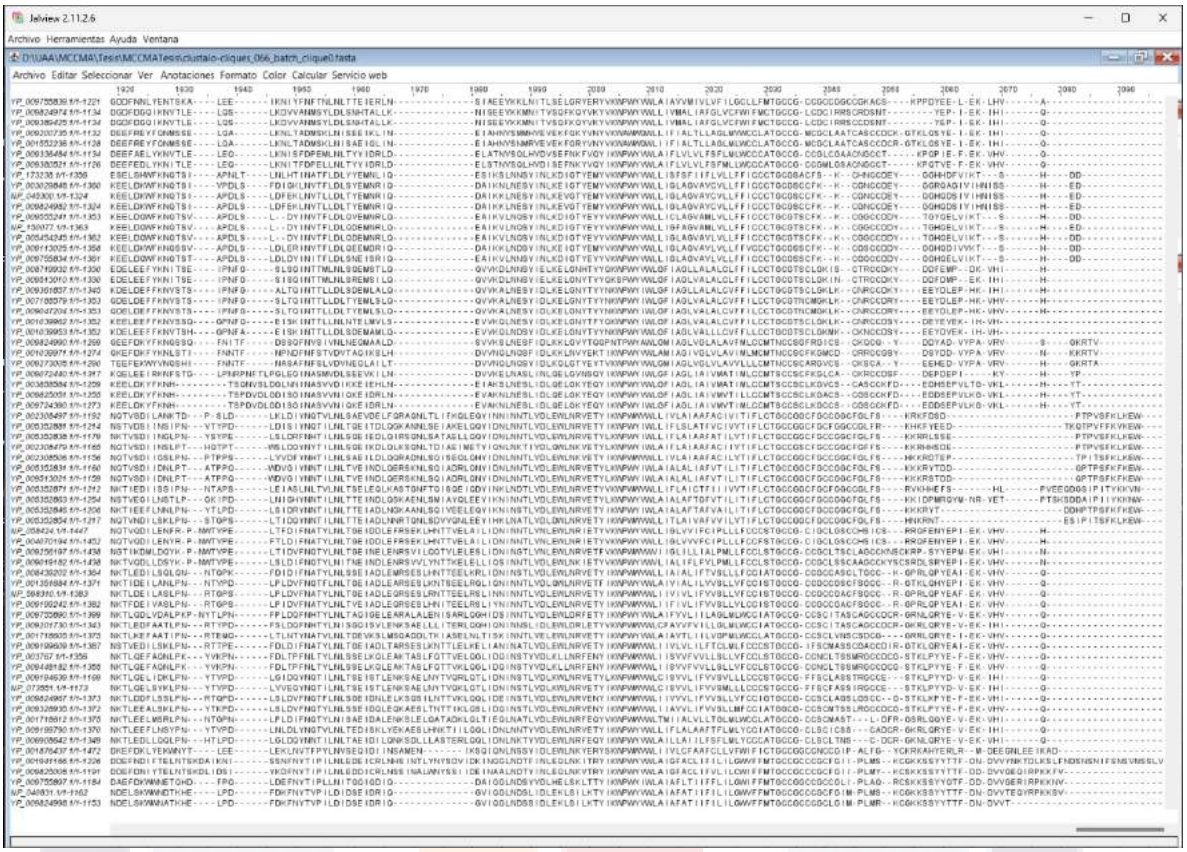












Anexo C

$\vec{R}$ altamente compatibles con la $\vec{SC} = AQVDRLITGRLAALN$.

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
AQVKDAGPGDAAALK	185.080	208.874	268.094	1
GNASDGGPGDGGGGQ	171.521	224.546	262.538	1
GQVKRAGPGDGGALK	168.142	208.948	274.547	1
AHGKDGIKADGAGVK	189.248	208.631	259.132	2
ASGKDGGGGDGAGPK	184.981	222.441	254.113	2
ANGKDVIGGDGAGVQ	180.948	212.925	271.142	2
GHLRDAVPGDAAGLK	188.950	204.005	262.358	3
GPGKDGGGGDAAGGK	185.263	224.687	254.472	3
GHLKDAVPGDAAGLK	186.525	206.101	267.557	3
ARARDGIGGDAAGGK	191.135	211.875	251.066	4
APGKDGGGGDGAGGK	185.263	224.687	254.472	4
AQAKDVIHGDAGGGK	186.084	207.978	267.557	4
AKARDVIPGDGAGGH	188.676	208.203	261.642	5
GKAGDGGGGDGAGAH	177.005	225.964	254.472	5
GKGQDGIPGDGGGGN	174.531	218.323	267.198	5
AKARDAIPGEGGAVK	189.664	206.775	259.849	6
ANAKDAGGGEGGGGS	180.025	224.392	259.132	6
ANLKDAIPGEGGGVK	183.924	207.825	269.349	6
GKGRDGGSGEGGGGR	189.997	216.974	246.047	7
GKGKDGGPGEGGGGS	183.519	223.490	256.623	7
GKVKDIVPGELAGVK	187.182	201.524	268.632	7
AGARDGGPADAAAGR	188.678	216.953	245.151	8
GGAKDGGPGDGAGGK	185.273	224.687	254.472	8
GQVKDGGPADAAAGK	185.058	214.777	263.792	8
GHARDAIPGDVAGAK	188.984	207.153	261.462	9
GGANDGIGGDGGGAS	171.116	227.614	259.849	9
GQAKDGIPGDVGGVS	181.400	212.998	270.245	9
AKARDAIPGDLAGAH	188.722	205.055	262.896	10
GKGKDGVGGDGAGAH	185.946	218.544	260.566	10
AKAKDAIPGDLAGAH	186.296	207.151	268.094	10

$\vec{R}$ altamente compatibles con la $\vec{SC} = AQVDRLITGRLAALN$. Parte 1

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
GHARDGIPGDAGALK	188.978	207.154	261.104	11
GGGKDGGGGDGGAGK	184.995	228.734	252.142	11
GHGKDGIPGDVGALK	186.512	208.200	267.736	11
GHARDAIPGDGAGAK	188.989	210.302	258.953	12
GPGKDAGGGDGAGAK	185.280	223.638	255.009	12
GNLKDAIGGDGAGVN	180.745	212.925	271.679	12
APARDAIPADGGAAK	188.045	211.853	256.443	13
GGAKDAGPGDGGGGN	181.403	225.740	257.877	13
APAKDAIPGDGGAAN	181.732	215.396	265.585	13
AKARDGIGPDGAGGK	190.535	212.678	253.934	14
GSAKDGIGPDGGGGK	184.995	219.948	257.877	14
ADAKDGIGPDVAGVN	178.938	210.509	270.425	14
APGRDGGKGDAGGGP	187.581	219.594	244.255	15
GPGKDGGGGDAGGGP	182.171	228.060	253.038	15
GQVKDGIGGDAGGAG	181.422	220.342	263.613	15
GKLKELIHGDAGAAK	188.254	201.678	266.660	16
GKGKKGIPGDAGGGK	173.412	215.238	264.509	16
GQVKELIPADVGAAK	184.117	203.233	272.038	16
AKVKDIVPADAGAGK	188.185	207.427	264.689	17
AQGQDGGPGDAGAGS	171.692	221.398	264.689	17
AQVKDIVPGDAGAVK	185.058	204.676	270.783	17
AKAKDGIPGDGAGLK	188.184	209.920	265.406	18
GSAKDGGPGDGAGGS	181.424	225.515	258.236	18
ASVNDGIGGRGAGGS	154.001	219.144	274.009	18
AKAKELIPGDAGGGK	187.233	208.871	264.689	19
GQAKEGGGGDGGGGN	179.932	223.419	261.283	19
AQAKELIPGDLGGVK	184.112	202.577	272.396	19
GKGKDGIPGDAGAAK	188.178	214.118	262.717	20
GSGSDGGPGDAGGAN	171.528	226.568	260.566	20
GNVKDVIPGDAGGAN	180.993	210.976	273.651	20

$\vec{R}$ altamente compatibles con la $\vec{SC} = AQVDRLITGRLAALN$. Parte 2

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
GKAKDAVAGDGAGAK	187.926	216.067	259.491	21
GGASDAGGGDGGGAN	171.506	230.763	256.085	21
GKAKRVVGGDGAAVK	170.986	208.794	272.575	21
AHLRDAAHADAAGGK	190.016	205.864	256.623	22
AHGRDAGPADAGGGK	188.962	215.155	253.575	22
AHVKDAVPADAAGVK	186.536	206.756	267.557	22
AKIKDAAPGDAAALK	188.236	206.772	264.151	23
AKGKDGAPGDAAAGG	184.814	221.539	256.802	23
ADGKRALPGDAGALG	162.632	212.096	272.575	23
AKVKDLIGADAAAAK	187.969	207.277	267.019	24
GKGKDGGGGDAAAAS	184.279	223.339	257.698	24
ANVKDLIGGDAAAAS	181.037	211.798	270.783	24
GKARDVIPGDAAAAH	188.710	206.104	262.717	25
GQAKDGGGGDGAGAA	181.444	225.589	256.443	25
GQAKDVIGGDGAAAH	182.904	212.250	268.632	25
GKARDVIGADGGGGK	190.321	213.576	256.623	26
ANAKDLGGGDGGGGN	180.724	220.271	264.868	26
ANAKDVIGGDVGAGN	180.751	212.925	272.038	26
AKAKDAIAPDAAAAK	188.201	209.526	260.745	27
AGAKDGGGPDAGGAK	185.238	223.244	251.962	27
AGAKRAIGPDAGGAN	164.525	215.974	268.274	27
GKARDGAPADGGAGK	190.593	215.826	252.679	28
GNAHDGAGADGGAGN	175.639	222.354	262.717	28
GNVHDLAGADGGAVN	175.616	212.909	270.245	28
AKVRDGIPGDAGGAK	190.597	208.874	260.387	29
AQGKDGGGGDGGGGN	180.882	224.469	262.000	29
AKVKDAIPGDAAAAK	188.223	207.821	267.198	29
GHARDVIPGDAAGAR	189.895	205.057	256.264	30
GGGKDGAGGDAGGGR	185.924	225.589	247.481	30
GHAKDVIPGDLAGAR	187.458	204.005	263.613	30

$\vec{R}$ altamente compatibles con la $\vec{SC} = AQVDRLITGRLAALN$. Parte 3

Anexo D

$\vec{R}$ altamente compatibles con la $\vec{SC} = IEDLLFDKVVT$

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
AKRGGGKEGGH	144.748	144.783	182.179	1
AKHGGGKEAGG	135.953	151.151	187.377	1
IKHVVVKEVAP	136.198	132.411	203.689	1
GKRAGAKDGAP	144.470	146.728	184.689	2
VSQGGANDGAG	113.114	153.854	196.340	2
VKHLLAKDVLP	136.903	131.361	203.151	2
GRRGGGKDGGG	146.317	150.778	176.085	3
GNDGGAKDGGG	115.976	155.953	193.292	3
INKAAGKDVVP	132.834	140.431	202.792	3
AKKGAARDAGH	145.501	142.684	184.509	4
GKKGAAKDAGG	141.785	151.822	188.094	4
AKKVLVKDILH	143.042	129.038	200.283	4
IKKGGGRDAAP	144.472	143.580	188.453	5
IQKGGGNDGAG	122.783	150.779	196.698	5
IQKGLVKDLAP	133.341	135.184	204.406	5
AKRGGAKDGGG	144.192	150.775	182.358	6
GNKAGGNDGGG	122.245	156.026	192.755	6
ANKALANDLGG	122.299	145.531	199.208	6
IKRAGGKDGVP	144.448	141.481	190.783	7
GSKGGGKDGAG	133.115	156.996	187.915	7
ISDAGINDGVP	106.569	144.485	204.226	7
IKKAGAKDAAP	142.089	143.577	194.726	8
ISNGGADDAAP	106.576	149.733	200.283	8
ISKLVVKDVVP	133.364	134.058	207.274	8
VKKGAIKDGAG	141.796	145.525	192.575	9
VNQGGGDDGGG	106.219	153.857	198.311	9
VNQGAGDDVVG	106.225	146.510	204.585	9
GKRGGAKDAAG	144.210	149.726	182.896	10
GNNGGAKDAGG	122.271	154.977	193.292	10
VKKLVIKDLAG	141.785	135.030	199.925	10

$\vec{R}$ altamente compatibles con la $\vec{SC} = IEDLLFDKVVT$. Parte 1

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
GRKGGGRDGGG	146.317	150.778	176.085	11
GNDGGGDDGGG	99.377	157.981	195.623	11
INDGGGDDVGP	99.658	147.637	204.585	11
GKRGAGKDGAG	144.184	150.775	182.358	12
GGKGAGKDGGP	133.990	156.245	186.481	12
ISDGVGDDGVP	100.226	146.511	204.585	12
GKRGAGKDAGA	144.208	149.726	181.821	13
GKGGAGKDAGG	132.795	158.193	183.792	13
VKKLAIKDVLG	141.784	135.030	200.283	13
IKRAAAKEGGP	143.804	141.481	188.274	14
ISKAGGKEGGP	132.709	148.750	193.292	14
ISKLGAKEVVP	132.711	138.256	201.717	14
IKKGAAKDAAG	141.830	146.574	192.934	15
INKGGGKDAAG	132.586	149.726	195.264	15
IKKLVVKDVVG	141.731	133.980	204.585	15
IKKGAARDGAA	144.261	144.478	186.660	16
GDQGAGQDGGG	107.481	154.906	195.264	16
IDQGAGQDAVG	107.520	146.510	202.972	16
IKKLAARDVVP	144.474	133.085	196.877	17
GNKAGGKDAGG	132.558	153.924	190.962	17
INQAAAKDVGP	123.140	142.534	203.151	17
GKRGAGKDAAG	144.202	149.726	182.896	18
GSKAAGKDAAG	133.167	153.847	189.528	18
ISNAAAKDVLG	122.889	144.405	201.179	18
IKKALAKDIAP	142.107	136.230	198.311	19
ISQAGGQDGAS	113.481	149.508	200.283	19
ISKALAQDAVP	123.720	140.358	203.330	19
IKRLAAKDVVH	145.489	130.090	196.160	20
GSNGAAKDAAG	122.879	154.900	191.858	20
IKKLGGKDVVG	141.746	140.277	199.208	20

$\vec{R}$ altamente compatibles con la $\vec{SC} = IEDLLFDKVVT$. Parte 2

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
IKKAAAKDLAH	143.110	136.384	196.698	21
IGKGAAKDGVS	133.550	148.600	194.547	21
IKKAAANDLVP	131.776	138.332	202.075	21
AKKGAARDGAP	144.486	145.679	185.226	22
AKKGGGKDGAS	141.530	150.625	189.708	22
VKKALIKDVAP	142.056	135.181	199.925	22
GRRGAGKDGGG	146.334	149.729	176.623	23
GSKGAGKDGGP	133.374	153.998	189.708	23
IKKAVVKDVVP	142.003	134.131	204.226	23
IKRGGAKDGAP	144.480	143.580	188.453	24
GKQGGGKDGGG	131.999	154.974	189.170	24
INQGGGKDAVP	123.097	144.633	202.075	24
IRGGGAKDAAG	135.015	150.849	183.434	25
IPGGGGKDAAG	125.513	156.318	187.557	25
INKLVAKDALG	132.600	138.182	203.151	25
GHKVAARDAAG	139.653	144.857	187.915	26
GHQVGGNDGAG	117.146	151.157	196.519	26
VHKVAIKDVLG	137.204	135.409	202.075	26
IKRGGGRDAGP	146.880	142.533	182.717	27
GSKGAAKDGGP	133.400	152.948	190.245	27
IKKALIKDALP	142.100	133.082	201.179	27
IKKGGGRDGAP	144.454	144.629	187.915	28
GKKGGGKDGGG	141.724	154.970	186.481	28
IQKAAAKDVAP	133.392	140.431	200.642	28
GRKAGARDGAP	146.637	144.632	179.491	29
GSKAGAKDAAA	133.199	152.798	188.991	29
ISKALANDAVP	123.149	141.407	202.972	29
IRKAGAKDGGP	144.222	143.580	188.453	30
GNKAGGKDGAG	132.558	153.924	190.962	30
IHKVIAKDVLP	137.492	131.361	204.226	30

$\vec{R}$ altamente compatibles con la $\vec{SC} = IEDLLFDKVVT$. Parte 3

Anexo E

$\vec{R}$ altamente compatibles con la $\vec{SC} = KVNECVKSQSSR$.

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
DGHRAADGGGPD	157.386	162.875	194.651	1
DGSKAADGGGSD	153.052	169.468	200.208	1
DGHKPADSQSSD	154.411	153.962	214.726	1
DGKRGGDGKGGD	161.874	163.321	194.651	2
DGKKGGDGGGGD	156.846	170.738	196.443	2
DANKGLDSNPSD	152.389	154.784	217.415	2
DVKKGLDSKHSD	160.478	147.585	214.009	3
DGGKGGDSSSSD	152.611	167.073	206.660	3
DGSKGADSSSSR	135.481	160.702	218.491	3
DAHRGGDGPGGD	157.359	163.924	195.189	4
DGGRGGDGGGGD	155.644	173.964	188.557	4
DAHKGVDSPSSD	154.541	156.132	212.934	4
DGKKPADPAPGD	157.734	159.647	200.028	5
DGSNAGDGAPGD	144.317	169.769	201.104	5
DVSNAGDGNPGK	129.131	162.423	213.472	5
DVHRAVDSKSSD	159.328	148.564	211.142	6
DAGGAGDSKGSD	147.946	169.468	201.283	6
DANGAVDSNSGR	127.266	160.028	221.000	6
DGKKGADAAPGD	157.153	164.592	197.698	7
DGSSGGDSNGGD	143.950	170.373	207.198	7
DGKKGVDSNPSD	156.229	155.830	213.651	7
DGKKGADGKPGD	159.969	161.370	202.179	8
DGKKGGDGGGGD	156.846	170.738	196.443	8
DVNKGVDSKPSD	155.639	152.682	217.236	8
DAKKGADSAPGD	156.933	162.346	202.000	9
DGKKGGDSAGGD	156.653	167.442	199.132	9
DAKKGVDSNPSD	156.247	154.781	214.189	9
DVKKAADAKSPE	158.852	152.826	206.481	10
DGGKAGDSGGSE	152.127	169.468	198.953	10
DGGKAADSKGSR	139.069	161.071	211.679	10

$\vec{R}$ altamente compatibles con la $\vec{SC} = KVNECVKSQSSR$. Parte 1

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
DAHKPGDHKSGD	159.148	152.460	207.377	11
DGPKGGDGKGGD	156.632	167.741	198.953	11
DGHEGGDSQSSD	140.182	157.939	217.774	11
DVHRGGDSHPSD	158.130	151.340	208.274	12
DVGKGGDGNGGD	153.045	168.643	202.717	12
DVNNGGDSNGSD	142.914	160.934	218.670	12
DGKKGGDPKGGD	159.952	162.419	201.642	13
DGSKGGDSGSGD	152.787	169.320	203.434	13
DVNKGADSHSSD	153.571	154.861	217.236	13
EGKKAADGKHGD	160.266	156.276	201.283	14
EGSKGADSNSGD	151.686	162.952	209.349	14
ELKKGLDSNHSD	156.511	146.539	215.802	14
DAKKPADGKSGD	159.885	158.074	204.151	15
DVSKAADGGSGD	153.046	166.319	203.075	15
DVSNPADSQSGD	143.713	157.859	216.698	15
DVHKGVDHHHHD	159.748	139.801	214.547	16
DGHKGGDGHGGD	156.154	164.075	202.358	16
DVNSGVDSNHHD	146.101	150.069	220.104	16
DGKRGVDPQGHD	160.211	151.186	204.509	17
DGGRGADGSGGD	155.441	170.668	192.321	17
DAKHGADPQSHD	153.301	151.414	213.113	17
DGHKAGDGKAGD	157.840	162.647	200.387	18
DGNSGGDGNSSD	143.493	166.104	213.292	18
DVHKGVDSQSSD	154.024	151.713	219.925	18
DVKKGGDSKHSD	160.473	151.783	211.321	19
DGGKGGDGKGGD	156.337	170.738	197.160	19
DVKSGVDSQSSD	147.319	155.458	219.208	19
DAGRPGDGKGPD	159.125	161.598	194.292	20
DANGGGDGNGPD	144.802	168.797	205.943	20
DGNKGVDPKSSD	155.645	155.830	214.368	20

$\vec{R}$ altamente compatibles con la $\vec{SC} = KVNECVKSQSSR$. Parte 2

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
DGPKGGDHKGGD	157.834	161.748	201.104	21
DGGKGGDSNGGD	152.836	169.545	203.075	21
DGQKGADSNSSD	152.137	158.684	215.802	21
EAKKPGDGKGGD	159.389	160.320	199.670	22
EGKKAGDGSGGD	155.965	166.393	198.415	22
EGQKAVDSKSSD	154.739	153.433	214.189	22
DGKRAGDSHSHD	161.047	151.115	203.613	23
DGKKAGDGGGGD	156.878	169.689	195.906	23
DANKAADSNSGD	152.188	159.881	212.934	23
DAKKPADGHGPD	158.713	156.652	203.613	24
DVSKPGDGNSGD	152.929	161.152	209.170	24
DVKKPVDSNSPD	156.582	149.684	216.519	24
DVKRGVDSHSSD	159.585	149.613	210.962	25
DGGKGGDSGSGD	153.046	171.567	200.208	25
DVKKGVDSNSSD	155.764	153.433	217.953	25
DGGRGGDSKHGD	159.491	160.404	197.340	26
DGGKGADSNGGD	152.854	168.495	203.613	26
DVNKGLDSKHPD	157.068	147.887	215.981	26
DAKKPGDAKGGD	160.105	160.320	199.849	27
DGSKPADGNGGD	153.168	165.497	203.613	27
DASKPVDGNGSD	152.947	160.102	209.708	27
DLHKGVDSKPHD	159.017	146.163	214.189	28
DGSKGGDGKGGD	156.078	168.492	200.387	28
DGNNGVDSKSSD	145.985	157.634	219.208	28
DVGKGADSKHGE	156.385	157.252	205.226	29
DGGSGGDGNGGR	127.537	171.791	208.811	29
DANSGGDGNGSR	126.838	164.226	218.670	29
DGKKGGDGKGPD	159.952	162.419	201.642	30
DGSSGGDSNGSD	143.735	168.126	210.425	30
DVSSGADSQSSD	143.713	160.632	217.415	30

$\vec{R}$ altamente compatibles con la $\vec{SC} = KVNECVKSQSSR$. Parte 3

Anexo F

$\vec{R}$ altamente compatibles con la $\vec{SC} = KWPWYYVWLLI$.

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
DAGAHAPGAA	112.570	123.704	176.160	1
DGGAGAAGAA	111.144	132.694	180.104	1
DGGAGVAVVV	111.070	123.249	189.245	1
DAPAKGAAAI	114.143	120.177	178.849	2
DGGAPAGGGG	111.340	132.845	178.849	2
DAPAPAAAAI	111.569	121.451	187.094	2
DGPAKAALIA	114.131	117.029	177.774	3
DGPGGGAGVG	111.198	130.746	182.255	3
DGPAGAAVVV	111.223	122.350	188.708	3
DAPAKAAGGI	114.126	121.227	178.311	4
DAGGPGAGGG	111.341	132.845	178.849	4
DAPAPVAGGI	111.512	121.451	188.349	4
DAHAKGAALI	114.639	114.034	178.491	5
DGGAGGAAGG	111.110	134.793	179.028	5
DGGAPLALLI	111.386	116.053	191.217	5
DAPAKGALAI	114.132	117.029	181.000	6
DGPGKGGGGG	114.025	129.623	171.858	6
DAPAPVALAI	111.535	116.204	191.575	6
DGPGKAGPGG	114.232	125.575	170.604	7
DGGGAGGPGG	111.267	133.894	175.085	7
DGPGAAGLAI	111.276	123.400	186.736	7
DPGGKAAAAI	114.199	121.227	174.726	8
DPGGGAAAAG	111.332	130.746	177.774	8
DAPGGVAAVI	111.256	121.301	190.500	8
DGPGKAAAAI	114.123	121.227	178.311	9
KGPGGGGGAA	100.483	131.866	183.151	9
KGPGGAGVAI	100.506	124.519	189.962	9
DAPAKIAALI	114.161	112.831	182.434	10
DGGGGAAAGG	111.109	134.793	179.028	10
DAPAGIAALI	111.321	118.152	189.783	10

$\vec{R}$ altamente compatibles con la $\vec{SC} = KWPWYYVWLLI$. Parte 1

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
DGPGRGGGGG	114.793	127.527	166.660	11
DGGAPGGGGG	111.323	133.894	178.311	11
DAPAHVAAAI	112.518	116.358	186.915	11
DGPAPAAALI	111.539	119.352	188.708	12
DGPGGAAAVG	111.233	128.647	183.330	12
DGPAPVAALI	111.516	117.253	191.038	12
DAPAKLAAVI	114.126	113.880	183.509	13
DGGAGAAAGG	111.127	133.744	179.566	13
DAPAGLAAVI	111.286	119.202	190.858	13
DPGGKGAAGA	114.153	126.474	169.887	14
DAGGPGGAGA	111.356	131.795	179.387	14
DPPAPVAAAV	111.700	118.454	185.660	14
DGPKGGGKGG	115.699	124.301	160.208	15
DGGAGGGVGG	111.068	133.744	180.462	15
DGPAPVAVGI	111.487	119.352	190.321	15
DPGGKAGGGG	114.119	128.573	168.811	16
DGPGGGAGGG	111.204	133.894	179.745	16
DPPAGVAVAI	111.462	118.303	188.708	16
DGPAKAAAAI	114.142	120.177	178.849	17
DGGAPGGGGG	111.323	133.894	178.311	17
DGPAPAAAAI	111.551	122.501	186.557	17
DAPGKVAAAI	114.118	118.078	181.179	18
DGPGGGGAAG	111.220	132.845	180.283	18
DGPGGVGGLI	111.213	123.400	189.066	18
DAGPKGGGGG	114.120	128.573	168.811	19
DAGGGGAGAG	111.110	134.793	179.028	19
DAGAHVALAV	112.343	117.256	185.840	19
DGPKPAGAAI	113.866	119.278	175.981	20
DGGPGGGAGG	111.278	133.894	176.160	20
DGPPPVAAAI	111.714	118.454	186.557	20

$\vec{R}$ altamente compatibles con la $\vec{SC} = KWPWYYVWLLI$. Parte 2

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
DAGGKGAGPG	114.122	127.524	169.349	21
DGGGGGAGPG	111.263	133.894	176.160	21
DAPGGAAVVV	111.223	122.350	188.708	21
DAGGKPAGAG	114.143	126.474	169.887	22
DGGGGVAGGG	111.068	133.744	180.821	22
DAGAPVAAAI	111.416	122.350	187.632	22
DKGAHAGGGG	114.664	124.529	166.840	23
DAGAGAGAGG	111.127	133.744	179.566	23
DAPAPLGGLI	111.510	117.253	190.321	23
DGPAKGGPGG	114.233	125.575	170.604	24
DGGAPGGAGG	111.340	132.845	178.849	24
DAPGPAAALI	111.539	119.352	188.708	24
DGHAKVAALI	114.615	111.935	180.821	25
DGGGGAAAAA	111.142	132.694	180.104	25
DAHAAVAVLI	111.793	113.058	190.142	25
DGPPKAGGAG	114.266	124.526	171.142	26
DGGGGGAGAG	111.091	135.843	178.491	26
DGPAHAALLI	112.499	113.209	188.349	26
DPPGGAGKGG	113.570	125.575	168.453	27
DPGGGAGGGG	111.279	133.894	176.160	27
DAPGGVAVAI	111.256	121.301	190.500	27
DAGGPAKGGG	113.692	127.524	169.349	28
DGGGGGAGGG	111.074	136.892	177.953	28
DAPGPAAVAI	111.528	120.402	188.528	28
DAGGRAGAAG	114.734	126.326	167.019	29
DGGGGAGAAG	111.107	134.793	179.028	29
DAGGPLGALL	111.357	120.251	186.915	29
DGPGKAAGPA	114.267	123.476	171.679	30
DAPGGAGGAA	111.255	130.746	181.358	30
DAPGGVALAV	111.234	121.301	189.245	30

$\vec{R}$ altamente compatibles con la $\vec{SC} = KWPWYVWLLI$. Parte 3

Anexo G

$\vec{R}$ altamente compatibles con la $\vec{SC} = NECVKSQSSRNGFCGN$.

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
SKPADSQPPDKAVPGK	193.786	213.106	287.915	1
SKGGDGSGSDKAGGGK	192.594	232.366	277.698	1
SKGVDSQPPDKAVPGN	189.534	215.058	293.651	1
HKGADSKGGDKGGGGK	197.382	224.890	280.028	2
PKGADGGGGDKGGGGK	193.583	235.453	270.887	2
NKGVDSKSSDKGAGGN	191.135	220.025	294.547	2
KKGGDPGGPDKGAGGK	197.156	227.134	275.368	3
KKGGDSGGPDKGGGGG	193.301	234.256	273.575	3
NKGLDSQSPDNGGGGG	185.108	224.599	292.755	3
HKGADGGGGDKGAPGK	195.120	228.411	273.934	4
GKGADGGGGDSGAGGK	189.685	240.476	270.170	4
HKGGDSGGSRSGAGGN	169.927	229.018	291.858	4
KKGVDSKGGDHGAPGK	197.742	218.744	282.896	5
NKGGDGNGGDGGAAGQ	185.378	234.188	281.104	5
NKGGDSNGSDHGGAGQ	186.370	224.752	291.321	5
KKKVDSKGGDKGGGAH	201.466	217.076	274.113	6
KKAGDPKGGDKGGGGS	196.150	227.885	276.443	6
SKAVDSNPSDKGVGGH	190.498	217.477	292.934	6
GKPADKAGGDKAAGAK	196.527	227.245	264.972	7
GKPADPAGGDKGAGAK	193.957	229.962	270.349	7
QKPADHSGSDKGAGAK	194.170	218.206	283.434	7
KKKGDSKGGDGGGGGK	200.007	226.611	267.481	8
GKPGDSNGGDGGGGGN	185.749	236.361	278.594	8
KKPADSNSSDKGVGGK	195.937	215.975	290.425	8
KKGVDPKGSDKGIGAK	199.585	215.873	284.151	9
KKGVDGGGSDGGGGGK	193.050	234.105	274.651	9
NKGGDPNSSDKGGGGN	188.290	224.525	292.396	9
KKGVDPKPSDPGLGGK	196.704	215.593	286.123	10
NKGGDGKGGDGGGGGK	192.262	235.232	275.368	10
KKGVDPKSSDSGLGGH	193.768	216.424	290.604	10

$\vec{R}$ altamente compatibles con la $\vec{SC} = NECVKSQSSRNGFCGN$. Parte 1

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
KKPADGKGGDHGGAAK	198.005	222.696	275.726	11
GKPVDSQSGDSGGGAS	185.710	229.298	284.868	11
KKPVDSQSSDHGVGAK	194.191	211.761	294.189	11
NKGVDSKSSDKGLGAH	193.084	214.759	294.368	12
NSGGDSKGSDSGGGGS	179.317	234.641	286.481	12
NSGVDSKSSRSGGGGS	162.208	226.171	300.642	12
KRGGDGKGGDKAGGAP	198.629	228.297	267.481	13
NKGADGKGGDKAGGAG	192.314	233.394	274.830	13
KKGLDHKSPDNAGGAS	193.275	216.762	287.377	13
KKPGDHKGSDKGVPGH	199.274	210.804	284.509	14
KKAGDGQSSDSGGGGN	188.501	227.445	287.557	14
KKAGDHQSSDKGVGGN	193.323	215.230	292.038	14
KKPVDGGPGDKGAPGK	197.621	220.988	276.443	15
KKAVDGGSGDKGAGGS	192.849	229.760	277.877	15
NKPVDSQSSDKGAGGS	188.631	219.053	294.726	15
KKGVDSKSHDKGGPGK	200.625	212.226	286.123	16
KKGGDGGGPDKGGPGS	193.615	231.258	273.575	16
KKGVDSNSHDNGVPGK	193.461	211.183	295.085	16
KKPVDGKSSDKGVGGK	199.415	215.070	286.840	17
KKPGDGSGGDKGGGGS	193.150	232.009	275.009	17
HKPVDGKSSDNGVGGK	193.623	215.451	291.858	17
KKGVDSKPPDKGAGGK	199.796	216.418	284.868	18
KKGGDSGGGDGGGGGK	193.056	237.254	271.783	18
NKGVDSQGGDSGGGGN	184.808	228.646	291.142	18
KKGGDSKGKDGGAGAK	198.757	225.167	272.142	19
KKGGDSKGGDGGGGAK	195.934	231.538	274.651	19
KKGVDSKSSDNGAGAS	191.394	221.653	291.142	19
HKGVDSKHGDKAVGAK	198.587	212.862	286.302	20
GKGGDSNGGDNAGGAN	184.925	234.302	283.613	20
NKGADSNGSDNAGGAN	184.227	226.737	293.472	20

$\vec{R}$ altamente compatibles con la $\vec{SC} = NECVKSQSSRNGFCGN$. Parte 2

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
KKGVDPHGSDKGGGGK	197.894	219.794	283.434	21
KKGGDGGGGDKGGGGK	196.640	234.179	271.245	21
KKGVDPHSSDKGLGGN	193.817	214.402	292.755	21
KKGADHGSGDKGGGGK	197.645	224.890	277.160	22
KKGADSGGGDGGGGGK	193.073	236.204	272.321	22
KKGVDHNSSDKGVGGN	193.103	214.180	295.085	22
KKGGDGKSGDKAAGGK	199.329	225.167	277.877	23
GKGGDGGSSDKGGGGK	192.841	235.007	275.009	23
KKGVDSNSSDNGGGGK	191.694	221.075	293.292	23
KKGLDSKGSDHGAGGK	197.181	218.446	285.943	24
KKGADGAGGDGGGGGN	189.441	238.454	271.962	24
NNGADGKGSDNGAGGN	178.519	228.723	292.575	24
KKGANSKPGDKGGGGK	195.482	222.637	279.670	25
KKGGNGGGGDKGGGGN	188.704	235.305	274.113	25
KKGVNSKPSDQGVGGK	191.580	215.147	291.142	25
KKGADGKGPDHAGGGK	197.860	223.745	278.057	26
KKGGDGKGSDGGGGGK	195.917	231.932	275.189	26
NKGVDGNSSDNGGGGK	188.039	224.374	293.472	26
KKGADGKGADPGGGAK	196.484	228.688	273.217	27
KKGGDGSGSDPGGGAS	189.519	234.690	276.802	27
NKGADGNSSDNGGGAS	184.451	229.154	291.142	27
KKGVDSKPSDKGAGGK	199.336	217.169	286.302	28
NKGVDGGGSDNGAGGS	185.076	232.914	285.226	28
NKGVDSSGSDNGVGGS	184.607	226.322	294.009	28
KKGADGKGGDKAGGGK	199.536	227.414	274.651	29
KKGGDGGGGDKGGGGK	196.640	234.179	271.245	29
NKGVDSGSGDQGGGGN	184.808	228.646	290.425	29
KKGADGKGGDKGGGGK	199.519	227.808	275.189	30
GKGGDGKGGDKGGGGK	196.132	234.179	271.962	30
HKGGDSNSSDSGGGGN	186.149	226.627	292.575	30

$\vec{R}$ altamente compatibles con la $\vec{SC} = NECVKSQSSRNGFCGN$. Parte 3

Anexo H

$\vec{R}$ altamente compatibles con la $\vec{SC} = QVDRLITGRLAAL$.

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
KAKDAIHGDAGAL	165.346	175.860	229.170	1
KAKDGGHGDAGGG	165.279	186.355	221.104	1
KVKDVIHGDLGAL	165.289	168.514	235.623	1
HARDGGGADAGAA	164.849	187.088	216.264	2
QGKDGGGGDGGAA	160.900	192.351	223.075	2
HVKDGGSGDVGAV	162.109	181.034	230.425	2
HAKDGIHGDAGAA	163.709	179.387	227.377	3
HAKDGGPGDGGGG	162.615	189.729	222.179	3
NAKDGIPGRAGGG	144.118	183.130	236.877	3
KARDGGGGDGGGG	166.412	191.301	214.292	4
NGQDGGGGDGGGG	150.958	195.503	224.330	4
NGDDGVPGRGGGG	127.477	187.257	237.236	4
KARDAIPGDAGGV	166.728	178.858	223.972	5
GAKDAAPGDGGGG	161.419	193.622	218.953	5
KAKDAIPGRAAGG	147.438	179.978	235.264	5
KAKDLIPPDAAAA	164.539	175.463	228.632	6
NAKDGGPGDAAGG	160.995	189.353	225.047	6
NAKDGIPGRAAGA	144.152	181.031	237.953	6
KGRDGIGGDGGGG	166.422	188.153	218.057	7
PGHDGGGGDGGGG	156.247	196.099	217.877	7
QAHDGIGGDVGAV	155.804	181.185	232.934	7
KVKDAPGGDAAAG	164.213	184.103	222.179	8
SVKDGIGGDGGGG	160.910	190.175	225.943	8
NVQDGIGGDLGAG	151.003	182.909	234.726	8
HARDAAGADLAAA	164.889	180.791	220.028	9
HAKDGAGGDGAAG	162.406	189.578	222.000	9
HAKDLAGGDLAAA	162.434	180.132	227.915	9
KAKDLGPADAAAL	164.326	178.461	227.736	10
SAKDGGPGDGAGG	161.182	192.425	221.642	10
NVKDLVPGDAAGL	160.977	175.710	235.623	10

$\vec{R}$ altamente compatibles con la $\vec{SC} = QVDRLITGRLAAL$. Parte 1

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
PARDAIPGDAAGA	164.158	182.231	220.925	11
GGKDAIGGDAGGA	161.188	192.422	221.462	11
QGNDAIPGDAGGA	150.909	185.158	232.038	11
KARDGGGGDGAAG	166.446	189.202	215.368	12
QASDGGGGDGGAG	151.246	195.426	222.538	12
QASDGIGGRGGAA	134.403	187.103	235.443	12
KGKDAIPADLAAA	164.348	178.461	229.349	13
KGKDAGGADVGAA	164.032	187.756	222.538	13
NGKDAIPGDVAAA	161.033	180.957	232.396	13
KLKDVLHADAAAA	165.335	172.318	229.528	14
QGNDGGAADGAGG	150.628	193.010	223.792	14
QGNDGGPARAAGA	134.010	185.887	235.264	14
KGRDAGPADAGAA	166.740	184.761	217.160	15
NGKDGGGGDAGGA	160.718	193.400	222.717	15
KLKDILPADAGAA	164.338	175.313	228.274	15
KGRDAIPGDLAAA	166.756	176.759	224.689	16
KGKDAGGGDGGAG	164.003	192.348	220.028	16
KVKDAIPGDLAAA	164.325	175.707	232.755	16
RAKDAIPAEAAAL	164.188	174.266	223.972	17
RAKDGGGGEAAGG	163.844	188.153	214.651	17
HAKDAIPGEAAAL	161.765	176.085	230.604	17
KAKDGIHPDIAAV	165.537	171.419	227.736	18
KAKDGASGDVGAG	163.786	185.903	225.226	18
KVKDVISGDVGAL	163.774	173.309	236.519	18
KARDGIPADLGAL	166.745	174.266	225.764	19
SANDGIGADGGGG	150.655	192.933	225.406	19
SVNDVIPADGGGG	150.886	184.687	232.038	19
KARDGAPADGPAA	166.925	181.763	215.368	20
KAKDGAGGDGPGA	164.206	188.300	218.774	20
KAKDVAPGDVPAL	164.460	174.808	228.274	20

$\vec{R}$ altamente compatibles con la $\vec{SC} = QVDRLITGRLAAL$. Parte 2

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
KARDAGGGEGGGI	165.507	185.004	215.189	21
NAQDAGGGEGGGV	150.036	189.206	227.198	21
KVKDLVPGELGGL	163.284	171.509	233.830	21
KARDGIHADAGAG	167.766	178.618	220.208	22
KARDGGGGDAGGG	166.430	190.252	214.830	22
KAKDVIHGDAGAV	165.312	174.811	230.962	22
HARDVVPGDAAAG	165.080	178.187	223.972	23
SGRDGAGGDGAGG	163.347	193.326	214.651	23
HVHDGVPGDAAAL	157.531	176.464	232.217	23
HARDAVPGDAAAA	165.120	179.237	222.538	24
AGKDGGAGDGGAA	161.189	195.570	215.547	24
PGKDGVPGRAAAA	144.793	183.351	232.217	24
KGRDLIHGDAAGA	167.754	174.814	223.434	25
GGRDLGGGDAAGA	163.591	190.326	214.651	25
NGHDLIGGDGAGL	155.627	181.185	232.396	25
KAREAGPADGAGG	165.772	184.761	215.906	26
NGKEGGGGDAGGG	159.750	193.400	221.462	26
KAKEVVPGDAAGA	163.336	179.905	227.557	26
KLKDAIPADAAAA	164.365	177.412	229.887	27
SGKDAAPGDGAGA	161.216	190.326	222.717	27
SGKRAIPGDAAAA	144.373	182.003	235.623	27
RGREGGGGDGGGG	166.218	189.205	207.840	28
KGKEGGGGDGGGG	163.018	193.397	218.236	28
KVKEAAPGDLAAV	163.340	175.707	230.245	28
HVRDAIKADAAAG	167.837	174.420	219.311	29
GAKDGGPGDAAGG	161.420	193.622	218.953	29
HAKDAIPGDAAAG	162.711	181.333	228.632	29
KAKEGIAGDAGAG	163.121	185.001	223.613	30
NGSEGGAGDAGGG	150.119	195.426	220.387	30
NVKEGISGDVGGA	159.537	180.659	233.292	30

$\vec{R}$ altamente compatibles con la $\vec{SC} = QVDRLITGRLAAL$. Parte 3

Anexo I

$\vec{R}$ altamente compatibles con la $\vec{SC} = RSAIEDLLFDKVVT$.

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
DGGGKRGGGKEAGP	184.331	193.705	231.066	1
DGGGQKGGGNEAAA	162.700	197.756	239.849	1
DGGIQQGGGNEAGS	152.733	193.413	248.991	1
DKGGKRGGAKDAGH	188.862	185.389	228.557	2
DSAGSKGGAKDAGG	173.539	198.577	240.387	2
DSAIKKGVAKDALH	183.469	177.966	250.066	2
DGGIKRAGGKDGAA	184.833	191.455	234.292	3
DGAAGRGGGKDGAG	176.787	200.974	228.736	3
DSAIKKAVGKDVAG	182.174	185.008	249.170	3
DGGARRGGGKDAAG	186.966	193.557	225.868	4
DGGANKGGGKDGGG	173.137	200.901	238.594	4
DHGINKGVGKDGVP	174.599	182.465	251.679	4
DPGAKKALGRDAAP	185.326	184.410	237.877	5
DGGANKAVGKDAGG	173.166	195.653	242.179	5
DPGINKALGKDAAP	173.711	184.411	250.245	5
DHGIKKVLAKDAAP	183.871	177.215	249.708	6
DGGGNKGGAKDGAP	173.424	196.854	240.925	6
RGGINNVGAKDAAP	146.279	186.435	258.670	6
DHAIKKAVVKDLVH	184.846	168.973	254.189	7
DGGINKGGGKDGGG	173.148	197.752	242.358	7
RSGINNGGGKDGVG	145.744	190.335	258.849	7
DPAIRRVAGKDVVH	188.478	171.973	240.745	8
DPGGKKGAGKDGGG	182.584	196.850	236.981	8
DSAIKKGVGKDVAG	182.157	186.057	248.632	8
DAAIKKAAAKDAAG	182.483	189.353	241.642	9
DGAGNKGVASDAAG	163.539	197.679	242.179	9
DGAINKAAIKDVVG	173.231	184.109	251.858	9
DPGVKRAAAKDVLP	185.306	179.163	243.075	10
DSGVSNAAAKDGGG	163.238	196.482	245.585	10
DPGVKKALVKDVLP	182.835	176.011	252.755	10

$\vec{R}$ altamente compatibles con la $\vec{SC} = RSAIEDLLFDKVVT$. Parte 1

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
DPGGKRLAVKDAVG	185.018	183.210	240.745	11
DSAGGPGAGKDAGG	165.968	203.148	235.189	11
DSAGKNLAVKDAVG	171.853	186.060	250.245	11
DHAIKKLAARDAAG	186.077	179.166	241.283	12
DPAAKKGAGKDAGG	182.634	193.702	238.594	12
DPGIKKLAAKDVAG	182.654	183.207	247.915	12
DPGGKKAAARDGAA	185.093	190.556	232.858	13
DPGVGKGAAKDGAA	174.634	195.875	238.057	13
DPGVKKAAAKDVLA	182.644	183.207	245.943	13
DHAIKRVAAKDVVP	186.278	173.020	247.557	14
DGGGKKAGGKDGGG	182.339	199.848	235.189	14
RGGINKAGGKDGGP	156.538	190.630	252.755	14
DGAVKKAVPKDVVP	182.874	180.209	246.840	15
DGGGSKAGGKDAGP	173.989	198.876	238.415	15
DSGVSKAVGKDVVP	173.733	185.085	252.217	15
DPAVKKLAIKDAGG	182.661	183.207	245.764	16
DSAGSNGGGKDAGG	163.218	200.680	242.179	16
RSGGSNGGAKDAVG	146.333	194.456	253.113	16
DHAIKRAAVKDAVP	186.291	175.119	245.585	17
DSGIKHGGGKDAAG	177.057	191.683	243.792	17
DSGIKHAGGKDVVP	177.287	183.438	250.783	17
DGAGKRGGGKDAGP	185.041	193.705	232.321	18
DGGGGKGGGKDGGP	174.570	203.221	234.113	18
DSAIKKLGVKDVVP	182.396	176.762	255.802	18
DHAIKKLVAKEAVH	184.187	170.022	251.142	19
DHGIKKGAAKEAGG	182.918	186.509	242.000	19
DHAIKKLVVKEVVH	184.130	165.824	255.802	19
DGGGRKALAKDGAP	184.815	188.457	235.547	20
DGGGKKGGGKDGAP	182.599	196.850	236.981	20
DGGGKKALAKDVVP	182.618	185.306	245.943	20

$\vec{R}$ altamente compatibles con la $\vec{SC} = RSAIEDLLFDKVVT$. Parte 2

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
DGAGRKGGAKDAAG	184.566	194.604	231.604	21
RGAGGNGGAKDAAG	147.205	199.999	244.868	21
DSAVKKVLAKDLAT	181.875	177.514	252.934	21
DPGIKKAGAKDAAP	182.931	186.506	244.689	22
DPGAGSAGGKDGGS	164.707	201.951	236.802	22
RPGIKKVGGKDGAS	165.490	184.182	254.009	22
DGAVKKALAKDALG	182.405	185.155	245.047	23
DGGGDQAGGSDGGG	148.149	203.905	240.208	23
RSAGKKALAKDGLG	165.290	184.032	252.934	23
DGAIKRGGAKDAVH	186.103	182.314	239.311	24
DGGGNKGGAKDGGG	173.146	200.901	238.594	24
DGAINKGAAKDAVG	173.220	190.406	247.377	24
DGGIRRGGGKDGGP	187.201	189.510	230.349	25
DGGGRKGGGKDGGG	184.489	198.802	229.453	25
DPGIKKGGAKDVAP	182.891	185.457	246.481	25
DPGAKKAGARDVGG	185.045	189.507	235.726	26
DGGASKAGGKDVVG	173.717	195.576	242.358	26
DPAAKKAVAKDVVG	182.625	184.257	246.840	26
DHAGKKGGAKDAGP	183.843	188.759	240.208	27
DGAGGKGGAKDAGP	174.631	200.073	235.726	27
RHAVKKGVVKDAVP	166.904	174.140	258.849	27
DGAAKKAVIKDGAP	182.686	186.356	243.255	28
DGGVGKAVGKDGAP	174.594	194.825	240.566	28
DSAVKKGVADDVAP	165.836	184.038	252.396	28
DGAIKKAAARDVAP	185.123	183.210	241.104	29
DGGGKKGAGKDVAP	182.610	192.652	240.387	29
DGAIKKAAAKDVVP	182.674	183.207	248.632	29
DSAAKRLAIKEAAG	183.973	181.862	239.491	30
DSGANKVGAKEAAG	172.284	191.308	245.226	30
DSAVNKLGVKEAVG	172.233	182.912	252.934	30

$\vec{R}$ altamente compatibles con la $\vec{SC} = RSAIEDLLFDKVVT$. Parte 3

Anexo J

$\vec{R}$ altamente compatibles con la $\vec{SC} = SAIEDLLFDKVVT$.

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
HAIKKGAARDLGP	166.361	169.699	230.066	1
HGGKKGAGRDGGP	166.285	180.194	222.000	1
HAIKKVAAKDLVP	163.924	165.498	240.642	1
PGIKKAAGKDGAP	162.965	178.988	232.217	2
GGASKGAGKDGAP	154.065	189.258	227.019	2
PGIKKLVVKDVAP	162.916	167.443	242.075	2
HAGKKGGARDGAP	166.328	178.095	223.075	3
GGGNKGGAQDGAP	143.758	188.289	231.679	3
HAVKKVGGQDVVP	154.111	169.699	241.000	3
GGGKRGGARDAAG	167.292	184.989	213.934	4
GGGSKGAAKDAAG	153.832	191.206	225.764	4
SGVKKGAAKDAVP	162.474	177.640	235.085	4
PGGKRAGGRDAGG	167.511	183.041	215.189	5
GGGKKGGGKDAGG	162.398	191.280	223.255	5
HGIKKGVVKDGVG	163.590	172.694	237.415	5
GGGRKAAGKDAGG	164.600	187.085	219.132	6
GAIGNAGANDAGG	133.883	191.361	231.500	6
GAIKKVLAKDVVG	162.439	172.389	239.028	6
PAIKRAGVKEGAP	164.688	172.694	229.170	7
PGGKKGGVKEGGG	161.924	185.134	226.660	7
PAIKKAGVKEVAP	162.257	171.641	237.236	7
KGGRKGAAKDAGG	167.406	181.764	216.085	8
PGGSKAAAKDGGG	154.059	189.258	227.019	8
PGISKVAVKDAGG	154.047	179.813	236.160	8
HGAKKAGAKDAAG	163.677	181.090	227.557	9
TGASNAGVKDAAG	143.244	185.815	233.651	9
TGVSNVVVKDAAG	143.193	178.468	240.462	9
GGGKRAGGKDGAG	164.841	188.135	218.594	10
GGGNKAGGKDAAG	153.233	190.234	227.736	10
SGINKALVKDAAP	153.302	173.445	242.613	10

$\vec{R}$ altamente compatibles con la $\vec{SC} = SAIEDLLFDKVVT$. Parte 1

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
HAVKRVAARDVAP	168.754	165.504	226.660	11
GGGKKGAAKDGAP	162.701	186.184	226.123	11
GAVKKVAAKDVAP	162.701	175.689	234.906	11
PGIKKLVARDGAP	165.382	170.595	231.679	12
GGIGKGVVNDGGG	144.088	188.209	232.396	12
PAIKKLVVKDVAP	162.933	166.394	242.613	12
PAGRKGGGKDAGH	166.102	179.145	221.462	13
GGGNKGGGKDAGS	152.970	190.086	228.811	13
PGIHKGGLKDVVP	158.341	170.971	241.000	13
PAIKKAAARDAAG	165.191	176.741	226.840	14
GGGNKGAGKDAAG	153.233	190.234	227.736	14
PAINKAAGKDVAG	153.516	178.840	237.236	14
GAGRRAGGKDGGP	167.267	183.041	215.189	15
GAGDKAGGKDGGG	147.605	191.210	226.660	15
GAIDQAGGKDVGP	138.161	180.869	238.311	15
HGIKRVAGRDGAP	168.750	169.702	224.151	16
GGGKRGGGRDGGG	167.232	188.138	212.321	16
HGIKRVAVKDGVP	166.293	166.551	234.547	16
PGIRKGAIKEGLP	164.453	169.545	230.066	17
GGGKKGVVKEGGG	161.674	184.983	227.377	17
SGIKKGVVKEGGP	161.746	175.541	236.698	17
HGVKKLVAKDAVK	166.582	165.274	232.755	18
GGGKKGGGKDAVK	165.360	182.810	222.000	18
HAVKKLLVKDVVA	163.609	163.248	241.717	18
GGGKKLGIKDAAA	162.488	180.785	228.094	19
GGGGKLGGKDAAA	154.435	190.305	223.613	19
SGVKKVVGKDAVS	161.932	174.193	239.387	19
GAGKKAGGKDAGP	162.692	186.184	226.123	20
GAGSKAGGKDAGG	153.806	192.256	225.226	20
SGGSKVLVKDLGP	153.796	175.467	239.387	20

$\vec{R}$ altamente compatibles con la $\vec{SC} = SAIEDLLFDKVVT$. Parte 2

$\vec{R}$	Carga	Peso	Hidropaticidad	N° Experimento
HGIKRAGAKDVVP	166.327	168.650	232.575	21
GGGQKGGGNDAGP	143.672	189.338	230.425	21
HGIKKGGVKDVVH	164.865	166.702	238.849	21
HAIRKGGARDVAG	168.257	171.650	223.255	22
GAAKKGGGKDGAG	162.432	189.181	224.330	22
HAIKKLLAKDGAG	163.682	170.595	236.160	22
GAIKRAAGKDAGA	164.926	180.788	223.434	23
GAISNAAGKDAGG	143.554	188.061	232.396	23
GAIKKLLIKDAVG	162.491	170.290	238.491	23
KGAKRAAGKDAAP	167.949	176.667	218.953	24
GGAKRAGGKDAAG	164.875	186.036	219.670	24
GAAKKLAGKDVVP	162.685	175.689	234.547	24
SGGRRGGAKDGGP	167.044	181.844	217.877	25
SGIGKGGGKDGAG	154.200	190.157	228.453	25
SGIKKLGAKDVAP	162.496	173.442	238.670	25
GGGKRAGAKDAAP	165.144	183.038	221.462	26
GGIGNAGGKDAGP	144.396	189.409	229.887	26
SGIKKLGGKDAVP	162.470	174.491	238.132	26
PGIKKLAAKDAAG	162.737	176.738	233.651	27
SGGNKGGGKDAAG	153.000	189.037	230.425	27
PGINKLAGKDAVG	153.487	176.741	238.849	27
GGIKRLVAKDGGH	166.134	171.647	228.632	28
GGGKKGGGKDGGA	162.405	191.280	222.179	28
SAIKKLGVKDGGH	163.482	171.497	237.415	28
PAAKKAAIKDGGP	162.996	177.938	230.604	29
PAAKKGAGKDGGP	162.936	183.186	227.915	29
PAVKKLVIKDVVP	162.928	164.295	242.792	29
HAGKKAAGKDAGP	163.911	179.142	228.811	30
SAVSNAAGNDAGP	133.270	184.919	238.311	30
SAVSNVAVNDVVP	133.209	174.424	248.349	30

$\vec{R}$ altamente compatibles con la $\vec{SC} = SAIEDLLFDKVVT$. Parte 3