PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

FACULTAD DE CIENCIAS E INGENIERÍA

HERRAMIENTA PARA EL ANÁLISIS DE DIVERSIDAD CONFORMACIONAL EN ESTRUCTURAS DE PROTEÍNAS REPETIDAS

Tesis para obtener el título profesional de Ingeniero Informático

AUTOR:

Ronaldo Romario Tunque Cahui

ASESORA:

Dra. Layla Hirsh Martinez

CO-ASESOR:

Dr. Nicolás Palopoli

Lima, mayo, 2024

Informe de Similitud

Yo, Layla Hirsh Martinez docente de la Facultad de Ciencias e Ingeniería, especialidad de Ingeniería Informática de la Pontificia Universidad Católica del Perú, asesor(a) de la tesis/el trabajo de investigación titulado Herramienta para el Análisis de Diversidad Conformacional en Estructuras de Proteínas Repetidas del/de la autor(a)/ de los(as) autores(as) Ronaldo Romario Tunque Cahui dejo constancia de lo siguiente:

- El mencionado documento tiene un índice de puntuación de similitud de 19%.

Así lo consigna el reporte de similitud emitido por el software Turnitin el 18 / 07 / 2023.

- He revisado con detalle dicho reporte y la Tesis o Trabajo de Suficiencia Profesional, y no se advierte indicios de plagio.
- Las citas a otros autores y sus respectivas referencias cumplen con las pautas académicas.

Lugar y fecha: ...18 Julio 2023.....

Apellidos y nombres del asesor / de la asesora: Hirsh Martinez Layla		
DNI: 40329236	Firma:	
ORCID: 0000-0002-8215-6716		

Resumen

Las proteínas repetidas son conocidas por la característica particular de presentar repeticiones en su estructura y por ser un tipo de proteínas que se encuentran en organismos unicelulares y pluricelulares, por ejemplo, el organismo humano, las bacterias, entre otros.

Desde hace ya algunos años, las proteínas repetidas han cobrado interés debido a que están relacionadas a enfermedades humanas y a aplicaciones en ingeniería. Además, esta clase de proteínas presenta una fuente fundamental de información para explicar la diversidad estructural contemporánea. Sin embargo, la comprensión de las proteínas repetidas con respecto a sus estructuras, funciones y evolución, representa un desafío considerable.

Aunque desde un punto de vista estructural, es posible analizar las diferentes estructuras que una proteína cualquiera presenta en su estado nativo (análisis de diversidad conformacional). Una proteína cualquiera, dependiendo del entorno, puede adoptar una u otra conformación diferente. A este conjunto de estructuras alternativas se le denomina estado nativo de la proteína y los cambios de una conformación a otra se conocen como diversidad conformacional y es un concepto clave para la comprensión de las diversas propiedades esenciales de la proteína como su función biológica, el origen de nuevas funciones, entre otras.

No obstante, hasta el día de hoy, no hay registro ni publicación alguna que explique algún estudio de diversidad conformacional aplicado, específicamente, a las proteínas repetidas.

Por ello, se busca plantear un método y una herramienta bioinformática que permita calcular, evaluar y visualizar la información de diversidad conformacional de este tipo de proteínas. Con la finalidad de que los investigadores relacionados al área de bioinformática y/o afines tengan a su disposición una herramienta de acceso libre que les permita evaluar las características de las proteínas repetidas y, a la vez, entender un poco más sobre la estructura, función y evolución de las mismas.

Dedicatoria

A mis padres, Gloria Cahui y Alejandro Tunque, quienes han sido mi mayor fuente de apoyo, amor y motivación. Gracias por creer en mí y por brindarme la oportunidad de perseguir mis sueños.

A mi asesora, la Dra. Layla Hirsh, por su paciencia, confianza y sabiduría. Su orientación me ha ayudado a superar mi temor a dar una opinión, además, me ha brindado la oportunidad de desarrollar un enfoque analítico y crítico en el ámbito de la investigación.

A mis co-asesores, el Dr. Nicolás Palopoli y el Dr. Gustavo Parisi, por confiar en mi, por estar presente para aconsejarme en diversas disyuntivas y guiarme en este proyecto.

Finalmente, dedico este logro a mi persona, por la perseverancia y el compromiso que he demostrado a lo largo de este proceso. La presente tesis representa un hito importante en mi vida y será un recordatorio constante de que puedo conseguir mis metas si me lo propongo y creo en mi.

Tema FCI

FACULTAD DE
CIENCIAS E INGENIERÍA

PONTIFICIA **UNIVERSIDAD CATÓLICA** DEL PERÚ

TEMA DE TESIS

TEMA	: Herramienta para el análisis de diversidad conformacional en estructuras de proteínas repetidas
ÁREA	: Ciencias de la computación
ASESOR	: Dra. Layla Hirsh Martinez
CO-ASESOR	: Dr. Nicolás Palopoli
ALUMNO(S)	: Ronaldo Romario Tunque Cahuí - 20140755
FECHA	: 28/09/2021

DESCRIPCIÓN Y OBJETIVOS:

Las proteínas repetidas son conocidas por la característica particular de presentar repeticiones en su estructura y por ser un tipo de proteínas que se encuentran en organismos unicelulares y pluricelulares, por ejemplo, el organismo humano, las bacterias, entre otros.

Desde hace ya algunos años, las proteínas repetidas han cobrado interés debido a que están relacionadas a enfermedades humanas y a aplicaciones en ingeniería. Además, esta clase de proteínas presenta una fuente fundamental de información para explicar la diversidad estructural contemporánea. Sin embargo, la comprensión de las proteínas repetidas con respecto a sus estructuras, funciones y evolución, representa un desafío considerable.

Aunque desde un punto de vista estructural, es posible analizar las diferentes estructuras que una proteína cualquiera presenta en su estado nativo (análisis de diversidad conformacional). Una proteína cualquiera, dependiendo del entorno, puede adoptar una u otra conformación diferente. A este conjunto de estructuras alternativas se le denomina estado nativo de la proteína y los cambios de una conformación a otra se conocen como diversidad conformacional y es un concepto clave para la comprensión de las diversas propiedades esenciales de la proteína como su función biológica, el origen de nuevas funciones, entre otras.

No obstante, hasta el día de hoy, no hay registro ni publicación alguna que explique algún estudio de diversidad conformacional aplicado, específicamente, a las proteínas repetidas.

Por ello, se busca plantear un método y una herramienta bioinformática que permita calcular, evaluar y visualizar la información de diversidad conformacional de este tipo de proteínas. Con la finalidad de que los investigadores relacionados al área de bioinformática y/o afines tengan a su disposición una herramienta de acceso libre que les permita evaluar las características de las proteínas repetidas y, a la vez, entender un poco más sobre la estructura, función y evolución de las mismas.

OBJETIVO GENERAL:

El objetivo general del proyecto es desarrollar un método y una herramienta que permita analizar la diversidad conformacional de las proteínas repetidas.

Objetivos específicos:

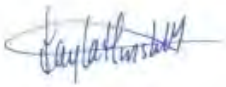
- Generar un conjunto de datos organizado de las proteínas repetidas, a partir de la base de datos RepeatsDB, que servirá como base para realizar el análisis de diversidad conformacional.
- Elaborar una propuesta de método específico para proteínas repetidas que permita analizar su diversidad conformacional.
- Desarrollar una herramienta de acceso libre a la comunidad científica donde los usuarios puedan evaluar y visualizar la diversidad conformacional de las diferentes proteínas repetidas.

ALCANCE:

El presente proyecto de fin de carrera es un proyecto de investigación enfocado en el área de bioinformática. Tiene como finalidad elaborar un método que permita analizar la diversidad conformacional de las proteínas repetidas, ya que esta clase de proteínas poseen la característica particular de tener repeticiones en su estructura. Asimismo, desarrollar una herramienta que permita evaluar y visualizar la diversidad conformacional de las proteínas repetidas o extraer y visualizar la información de diversidad conformacional de las proteínas repetidas obtenidas de la base de datos.

Para esto, se está delimitando a elaborar tres propuestas de métodos que permitirán analizar la diversidad conformacional de las proteínas repetidas. Estas propuestas se van a comparar entre sí junto con dos métodos existentes (métodos genéricos) con la finalidad de verificar cuál presenta los mejores resultados para poder analizar la diversidad conformacional de esta clase de proteínas.

Luego de elegir el método apropiado, este método se aplicará en el conjunto de datos de las proteínas repetidas y la información que se recolecte será guardada en una base de datos que se creará. Además, usando una interfaz de usuario, se visualizará la información extraída de la base de datos creada por medio de un servicio web que se desarrollará. Esta información consistirá en datos generales de la proteína repetida, en información estructural de la proteína repetida y las distintas conformaciones que la proteína repetida presenta. En caso, la información de la proteína insertada como dato no se encuentre en la base de datos, se procederá a calcular y evaluar el análisis de diversidad conformacional, utilizando el servicio web, para luego mostrar su información.



FACULTAD DE
CIENCIAS E INGENIERÍA

PONTIFICIA **UNIVERSIDAD CATÓLICA** DEL PERÚ

IMPORTANTE

1. Usted debe adjuntar un archivo conteniendo el tema de tesis en formato PDF con el visto bueno (firma) de su asesor (o asesores).
2. Usted no debe contar con un Tema de tesis asignado anteriormente. De darse el caso, deberá efectuar el trámite de cambio del tema de tesis en la Facultad.
3. Usted debe encontrarse matriculado o haber aprobado el primer curso de Tesis de su especialidad.
4. En caso de que el tema de tesis mencione a una organización, deberá adjuntar la autorización del representante legal de dicha organización.
5. Se recomienda que la extensión del documento final de tesis, incluyendo los anexos, esté comprendida entre 75 y 150 páginas. Asimismo, el archivo del documento final de tesis no deberá exceder los 15 MB. Revisar el instructivo para la elaboración de documentos académicos https://drive.google.com/open?id=15XqAM1J4YDk4wi_EAqVUQEJbGfaZihUr

En caso de alguna consulta adicional, puede contactarnos a la cuenta: titulacion-fci@pucp.edu.pe

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERU
Departamento de Ingeniería
Dra. LAYLA HIRSH M.
Coordinadora de Especialidad
Ingeniería Informática

Índice General

Índice de Figuras

Índice de tablas

Capítulo 1

Generalidades

1.1. Problemática

En esta sección se apreciará el problema de que no hay, hasta el día de hoy, registro alguno de un método y herramienta que permita analizar la diversidad conformacional en proteínas repetidas. Además, se desarrollará una breve contextualización indicando los problemas que dieron origen a esta problemática mencionada; asimismo de los efectos que produjo.

1.1.1. Árbol de problemas

A continuación, en la Tabla 1.1 se mostrará el problema central con sus respectivos problemas efectos y problemas causas.
Cabe mencionar, que estos problemas efectos y problemas causas nombradas en la Tabla 1.1 son de autoría propia debido a que no se encuentran registros de estos en alguna fuente primaria o secundaria.

1.1.2. Descripción

Las proteínas repetidas son conocidas por la característica particular de presentar repeticiones en su estructura (Kajava, 2012) y por ser una clase de proteína que prevalece en células eucariotas (Marcotte et al., 1999). Células que poseen núcleo y que se encuentran en organismos unicelulares y pluricelulares como los organismos del reino animal, vegetal, fungi y protistas (Lodish et al., 2003). Asimismo, esta clase de proteína, también se encuentra en procariotas, células que no poseen núcleo y que se encuentran en organismos unicelulares como las

Tabla 1.1

Árbol de problemas

	1	2	3
Problemas Efectos	No se conoce características de la diversidad conformacional en proteínas repetidas	No se conocen el estado nativo ni las diversas conformaciones de las proteínas repetidas	No se han identificado los casos particulares de las proteínas repetidas en las diversas bases de dato de diversidad conformacional
Problema Central	No existe un método ni una herramienta que permita el análisis de la diversidad conformacional en proteínas repetidas		
Problemas Causas	No se cuenta con un dataset organizado de proteínas repetidas que pueda ser usado para el análisis de diversidad conformacional	No se han realizado pruebas de métodos de diversidad conformacional genéricos en proteínas repetidas	No se conocen herramientas implementadas que evalúen específicamente la diversidad conformacional de las proteínas repetidas

bacterias y arqueas (Lodish et al., 2003). Sin embargo, este último tipo de células mencionado, presenta proteínas repetidas en menor proporción (Marcotte et al., 1999).

Por otro lado, cada patrón o región de repetición que se encuentra en la estructura de la proteína se denomina repeticion en tándem (TR), estas repeticiones se pueden describir por el número de aminoácidos, la longitud del patrón repetido y la similaridad estructural (Schaper et al., 2015). Además, cada región de repetición viene a ser un grupo de al menos tres unidades repetitivas (Di Domenico et al., 2014). Esta unidad repetitiva viene a ser el bloque estructural más pequeño que se repite para formar una región de repetición (Di Domenico et al., 2014). Asimismo, estas repeticiones abundan en gran cantidad en el proteoma humano, es decir, que existe un gran número de repeticiones dentro del conjunto de proteínas que un organismo humano posee; y se estima que el 14% de todas las proteínas conocidas contienen al menos una repetición en tándem (Marcotte et al., 1999). Además, existe una base de datos, llamada RepeatsDB, que contiene la información estructural de esta clase de proteínas (Di Domenico et al., 2014).

Por otra parte, las proteínas repetidas han cobrado interés desde hace ya algunos años, debido a que representan una fuente fundamental de información para explicar la diversidad estructural contemporánea y las propiedades fisicoquímicas de los pliegues (folding) que pueden adoptar muchas secuencias o funciones diferentes (Goncearenco and Berezovsky, 2015). Además,

están relacionadas a enfermedades humanas (Kajava and Steven, 2006). Por ejemplo, la enfermedad de Huntington, distrofia miotónica, entre otras enfermedades, las cuales la mayoría son trastornos del sistema nervioso, se deben a la expansión de las repeticiones en tándem dentro del organismo (Hannan, 2018). Asimismo, también tiene importancia en aplicaciones de ingeniería que hacen uso de proteínas, dado que las proteínas repetidas existentes ocupan solo una pequeña fracción de las posibles secuencias de esta clase de proteínas y que se pueden diseñar nuevas proteínas repetidas con geometrías específicas (Brunette et al., 2015). Asimismo, la importancia de las repeticiones en proteínas para la comprensión de la función biológica de estas, no solo reside en su alta frecuencia entre las secuencias conocidas, sino también en su capacidad para conferir múltiples enlaces y roles estructurales en las proteínas (Andrade et al., 2001).

Sin embargo, esta clase de proteínas todavía pertenece a la "materia oscura" del universo proteico que está caracterizada por relaciones no canónicas de secuencia-estructura (Hirsh et al., 2016). Es decir, que la relación entre la secuencia y estructura de una proteína repetida no siempre es de uno a uno. Por ejemplo, se puede tener una estructura idéntica para dos secuencias diferentes de proteínas repetidas. Además, la comprensión de las proteínas repetidas con respecto a sus estructuras, funciones y evolución, representa un desafío considerable (Andrade et al., 2001), ya que la duplicación de las unidades repetitivas generan una nueva protéina repetida, debido a que la repetición surge por cualquier lado de la misma (Andrade et al., 2001; Schaper et al., 2014).

Aunque desde un panorama estructural, es posible analizar las diferentes conformaciones que una proteína cualquiera presenta en su estado nativo (análisis de diversidad conformacional) (Parisi et al., 2015). Una proteína cualquiera puede prensentar diversas conformaciones o estructuras dependiendo del entorno y este conjunto de estructuras alternativas representan al estado nativo de esta proteína (Kumar et al., 2000; Tsai et al., 1999). Es así que, los cambios conformacionales o estructurales que experimenta una proteína en su estado nativo, es decir, los cambios de una estructura a otra dentro del estado nativo de la proteína, se conoce como diversidad conformacional y es un concepto clave para la comprensión de las diversas propiedades esenciales de las proteínas como su función biológica, su tasa evolutiva, el origen de nuevas funciones, entre otras (James and Tawfik, 2003). Asimismo, estos cambios conformacionales están relacionadas con las funciones de la proteína y su enfoque en estas tiene una extensión

que va desde fluctuaciones en las cadenas laterales a movimientos en bucle, estructuras secundarias y reordenamiento de la estructura terciaria (James and Tawfik, 2003).

No obstante, las proteínas repetidas son parte de las diversas clases de proteínas que tienen características particulares en el universo proteico (Andrade et al., 2001) y hasta el día de hoy, no hay registro ni publicación alguna que explique algún método o herramienta que permita evaluar y estimar la diversidad conformacional, específicamente, en esta clase de proteínas. Es así que, por la falta de un método y herramienta, no se pueden conocer las características de la diversidad conformacional ni las diversas estructuras de las proteínas repetidas. De la misma manera, no se pueden identificar los casos particulares de esta clase de proteínas en las diversas bases de datos de diversidad conformacional. Y esto se debe a tres causas, las cuales se detallarán a continuación.

En primer lugar, cuantificar las diferencias entre dos estructuras de la misma proteína no es trivial y continúa evolucionando en el tiempo (Burra et al., 2009); y más cuando estas estructuras tienen repeticiones internas (Andrade et al., 2001). Es así que, para estimar la diversidad conformacional en proteínas, las estructuras o también conocidas como confórmeros o conformaciones, tienen que superponerse estructuralmente (Kufareva and Abagyan, 2012). Para esto, existen diversos métodos que se basan en la comparación de estructuras que hacen uso de algún software que emplea algoritmos de alineamiento estructural que permiten la superposición de dos estructuras, tales como TM-Align (Zhang and Skolnick, 2005), FAST (Zhu and Weng, 2005), SANA (Wang et al., 2010), MICAN (Minami et al., 2013), MAMMOTH (Ortiz et al., 2002), entre otros. Luego, se tiene que calcular una medida estadística que permitirá ver el grado de diversidad conformacional. Es así que estos softwares mencionados, realizan diferentes cálculos de medidas estadísticas como el TM-score, desviación cuadrada media raíz (RSMD), GDT_TS y demás medidas. Pero, por lo general, la medida cuantitativa más utilizada para analizar la diversidad conformacional es la RMSD (Monzon et al., 2019), ya que es más sensible a los movimientos en bucles y colas (Palopoli et al., 2016). Mencionado esto, para las proteínas repetidas, no existe investigación ni publicación alguna que registre que se haya realizado pruebas con este procedimiento genérico descrito.

En segundo lugar, existen investigaciones que hacen uso de un conjunto de datos de proteínas que cumplen con ciertas características para analizar su diversidad conformacional. Este selecto conjunto de datos depende de la hipótesis de investigación, por ejemplo, se tiene que la

extensión de la diversidad conformacional en proteínas fue estudiada en un conjunto de datos de 5000 proteínas con más de 5 confórmeros por proteína (Monzon et al., 2019). Por lo tanto, para poder realizar el análisis de diversidad conformacional en cualquier tipo de proteína es necesario tener un conjunto de datos organizado de la clase de proteína, pero en el caso de las proteínas repetidas, a nuestro conocimiento, no se tiene uno.

Por último, existen herramientas que permiten evaluar o visualizar la información de la diversidad conformacional en proteínas, como por ejemplo el servicio web CoDNaS[1] (Monzon et al., 2016). A través de este servicio web, un usuario puede realizar consultas a la base de datos con la finalidad de visualizar la diversidad conformacional de las proteínas en su estado nativo (Monzon et al., 2016). Mientras que, en la actualidad, todavía no se conocen herramientas implementadas que evalúen y visualicen la información de la diversidad conformacional, específicamente, en proteínas repetidas, ya que las existentes evalúan la diversidad conformacional sin considerar la característica particular de las proteínas repetidas, la cual es presentar repeticiones en su estructura (Kajava, 2012).

1.1.3. Problema seleccionado

Hoy en día, no existe un método concreto ni una herramienta que permita analizar la diversidad conformacional en las proteínas repetidas.

1.2. Objetivos

En esta sección se presentan los objetivos a cumplir para elaborar un método y una herramienta que permita evaluar la diversidad conformacional de las proteínas repetidas con sus respectivos resultados esperados y sus medios de verificación de cumplimiento.

1.2.1. Objetivo General

Desarrollar un método y una herramienta que permita analizar la diversidad conformacional de las proteínas repetidas.

[1]. El servicio web CoDNaS tiene a su disposición una base de datos con la información de la diversidad conformacional en proteínas. Esta se puede encontrar en http://ufq.unq.edu.ar/codnas/

1.2.2. Objetivos específicos

O1. Generar un conjunto de datos organizado de las proteínas repetidas, a partir de la base de datos RepeatsDB, que servirá como base para realizar el análisis de diversidad conformacional.

O2. Elaborar una propuesta de método específico para proteínas repetidas que permita analizar su diversidad conformacional.

O3. Desarrollar una herramienta de acceso libre a la comunidad científica donde los usuarios puedan evaluar y visualizar la diversidad conformacional de las diferentes proteínas repetidas.

1.2.3. Resultados esperados

En la presente sección se presentan los resultados que se esperan obtener al alcanzar los objetivos específicos.

1. **Resultados esperados del objetivo específico 1 (O1)**

   R1. Estructura de datos organizado usado para representar el conjunto de datos de las proteínas repetidas.

   R2. Conjunto de datos de las proteínas repetidas que servirán como datos de entrada para el análisis de diversidad conformacional.

   R3. Conjunto de datos de prueba de las proteínas repetidas para evaluar la efectividad del método que analizará la diversidad conformacional de las proteínas repetidas.

2. **Resultados esperados del objetivo específico 2 (O2)**

   R4. Comparación de resultados obtenidos de dos métodos existentes (métodos genéricos) para analizar la diversidad conformacional sobre el conjunto de datos de prueba de proteínas repetidas con los resultados de la base de datos CoDNaS.

   R5. Tres propuestas de métodos a aplicar para analizar la diversidad conformacional de las proteínas repetidas.

   R6. Resultados obtenidos de los métodos elaborados sobre el conjunto de datos de prueba de las proteínas repetidas y la comparación con los resultados de los métodos genéricos.

R7. Aplicación del método seleccionado en todo el conjunto de datos de las proteínas repetidas.

3. **Resultados esperados del objetivo específico 3 (O3)**

R8. Modelamiento de la estructura de base de datos que contiene la información de diversidad conformacional de las proteínas repetidas.

R9. Servicio web que permita evaluar la diversidad conformacional de las proteínas repetidas o extraer la información de diversidad conformacional de esta clase de proteínas de la base de datos.

R10. Interfaz de usuario que permita evaluar y visualizar la información de diversidad conformacional de las proteínas repetidas utilizando el servicio web.

1.2.4. Mapeo de objetivos, resultados y verificación

A continuación, se mostrarán en la Tabla 1.2, Tabla 1.3 y Tabla 1.4 los resultados esperados, medios de verificación e indicadores objtetivamente verificables del objetivo específico 1 (O1), objetivo específico 2 (O2) y objetivo específico 3 (O3), respectivamente.

Tabla 1.2

Resultados esperados, medios de verificación e indicadores objetivamente verificables del objetivo específico 1

Objetivo específico 1 (O1): Generar un conjunto de datos organizado de las proteínas repetidas, a partir de la base de datos RepeatsDB, que servirá como base para realizar el análisis de diversidad conformacional.		
Resultado	**Medio de verificación**	**Indicador objetivamente verificable**
R1: Estructura de datos organizado para representar el conjunto de datos de las proteínas repetidas.	- Informe de la estructura de datos organizado.	- Conformidad al 100% del informe de la estructura de datos organizado por parte de 2 expertos.

R2: Conjunto de datos de las proteínas repetidas que servirán como datos de entrada para el análisis de diversidad conformacional.	- Reporte del conjunto de datos.	- Conformidad al 100% del archivo del conjunto de datos (6329 proteínas repetidas) por parte de 2 expertos.
R3: Conjunto de datos de prueba de las proteínas repetidas para evaluar la efectividad del método que analizará la diversidad conformacional de las proteínas repetidas.	- Reporte del conjunto de datos de prueba.	- Conformidad al 100% del reporte del conjunto de datos de prueba (30% del conjunto de datos de las proteinas repetidas) por parte de 2 expertos.

Tabla 1.3

Resultados esperados, medios de verificación e indicadores objetivamente verificables del objetivo específico 2

Objetivo específico 2 (O2): Elaborar una propuesta de método específico para proteínas repetidas que permita analizar su diversidad conformacional.		
Resultado	**Medio de verificación**	**Indicador objetivamente verificable**

R4: Comparación de resultados de dos métodos existentes (métodos genéricos) para analizar la diversidad conformacional sobre el conjunto de datos de prueba de proteínas repetidas con los resultados de la base de datos CoDNaS.	- Reporte de resultados de los dos métodos existentes.	- Conformidad al 100% del reporte de resultados de los 2 métodos existentes por parte de 3 expertos.
R5: Tres propuestas de métodos a aplicar para analizar la diversidad conformacional de las proteínas repetidas.	- Reporte de propuestas de los métodos a aplicar.	- Conformidad al 100% del reporte de propuestas de los métodos por parte de 3 expertos.
R6: Resultados de las frecuencias de los diversos valores de RMSD obtenidos de los métodos elaborados sobre el conjunto de datos de prueba de las proteínas repetidas y la comparación con los resultados de los métodos genéricos.	- Reporte de resultados de las tres propuestas de métodos.	- Conformidad al 100% del reporte de resultados de las tres propuestas de métodos por parte de 3 expertos.
R7: Aplicación del método seleccionado en todo el conjunto de datos de las proteínas repetidas.	- Reporte de resultados del método seleccionado.	- Conformidad al 100% del reporte de resultados del método seleccionado por parte de 3 expertos.

Tabla 1.4

Resultados esperados, medios de verificación e indicadores objetivamente verificables del objetivo específico 3

Objetivo específico 3 (O3): Desarrollar una herramienta de acceso libre a la comunidad científica donde los usuarios puedan evaluar y visualizar la diversidad conformacional de las diferentes proteínas repetidas.		
Resultado	**Medio de verificación**	**Indicador objetivamente verificable**
R8: Modelamiento de la estructura de base de datos que contiene la información de diversidad conformacional de las proteínas repetidas.	- Documento del modelamiento de la estructura de base de datos.	- Conformidad al 100% del documento del modelamiento de la estructura de base de datos por parte de 3 expertos.
R9: Servicio web que permita evaluar la diversidad conformacional de las proteínas repetidas o extraer la información de diversidad conformacional de esta clase de proteínas de la base de datos.	- Documento de arquitectura del servicio web. - Informe de pruebas funcionales del servicio web. - Repositorio del código fuente.	- Conformidad al 100% del documento de arquitectura del servicio web por parte de 3 expertos. - Informe de pruebas funcionales del servicio web aprobadas al 100% por parte de 3 expertos.

R10: Interfaz de usuario que permita evaluar y visualizar la información de diversidad conformacional de las proteínas repetidas utilizando el servicio web.	- Informe de prototipo de la interfaz de usuario. - Repositorio de código fuente. - Manual de uso.	- Conformidad al 100% del manual de uso por parte de 3 expertos. - Informe del prototipo de la interfaz de usuario aprobada al 100% por parte de 3 expertos.

1.3. Métodos y Procedimientos

En la presente sección, se detallan las herramientas y métodos a utilizar en el presente proyecto de tesis relacionados a los resultados esperados correspondientes. Además, se van a describir las características y el aporte que estas herramientas y métodos brindan en el desarrollo del proyecto.

1.3.1. Herramientas y métodos

A continuación, se presenta en la Tabla 1.5 los resultados esperados del proyecto con sus respectivas herramientas y métodos a utilizar.

Tabla 1.5

Herramientas y métodos a utilizar para cada resultado esperado.

Resultado Esperado	Herramientas	Métodos
R1: Estructura de datos organizado para representar el conjunto de datos de las proteínas repetidas.	- RCSB PDB - PyMOL	

R2: Conjunto de datos de las proteínas repetidas que servirán como datos de entrada para el análisis de diversidad conformacional.	- RCSB PDB - RepeatsDB - Python - Biopython - PyMOL	
R3: Conjunto de datos de prueba de las proteínas repetidas para analizar su diversidad conformacional.	- RCSB PDB - RepeatsDB - Python - Biopython - PyMOL	
R4: Comparación de resultados de dos métodos existentes (métodos genéricos) para analizar la diversidad conformacional sobre el conjunto de datos de prueba de proteínas repetidas con los resultados de la base de datos CoDNaS.	- Python - RStudio - CoDNaS - Mammoth - TM-align	- Método de similitud
R5: Tres propuestas de métodos a aplicar para analizar la diversidad conformacional de las proteínas repetidas.	- Overleaf	- Reuniones quincenales
R6: Resultados de las frecuencias de los diversos valores de RMSD obtenidos de los métodos elaborados sobre el conjunto de datos de prueba de las proteínas repetidas y la comparación con los resultados de los métodos genéricos.	- Python - RStudio - CD-HIT - Mammoth - TM-align	

R7: Aplicación del método seleccionado en todo el conjunto de datos de las proteínas repetidas.	- Python - CD-HIT - Mammoth - TM-align	
R8: Modelamiento de la estructura de base de datos que contiene la información de diversidad conformacional de las proteínas repetidas.	- MySQL Workbench	- Normalización de base de datos.
R9: Servicio web que permita evaluar la diversidad conformacional de las proteínas repetidas o extraer la información de diversidad conformacional de esta clase de proteínas de la base de datos.	- Flask - MySQL Workbench - Git - Postman - AWS - Visual Studio Code	- Reuniones quincenales
R10: Interfaz de usuario que permita evaluar y visualizar la información de diversidad conformacional de las proteínas repetidas utilizando el servicio web.	- Visual Studio Code - Git - React - Figma	- Reuniones quincenales

1.3.2. Descripción de Herramientas y Métodos

Como siguiente paso, en esta sección, se van a describir las herramientas y métodos mencionados en la sección anterior.

1. **Herramientas**

   En esta sección se describirán y se justificarán, en relación al proyecto, las herramientas mencionadas de la Tabla 1.5.

   - **AWS**

     AWS es una plataforma independiente del lenguaje y del sistema operativo. Además,

proporciona una infraestructura global y masiva en la nube que le permite innovar, experimentar e iterar con rapidez (Amazon Web Services, Inc, 2020). Esa herramienta se utilizará para preparar el entorno de desarrollo de la solución en la nube.

- **Biopython**

  Biopython es un conjunto de herramientas disponibles gratuitamente para computación biológica escritas en Python por un equipo internacional de desarrolladores (Biopython, 2020). Se utilizará esta herramienta para poder manejar los archivos en formato pdb que serán extraídos de la base de datos RCSB PDB.

- **CD-HIT**

  CD-HIT es un paquete que puede realizar varios trabajos como agrupar una base de datos de proteínas, agrupar una base de datos de ADN/ARN, entre otros (Huang et al., 2010). Esta herramienta se utilizará para poder identificar y separar en clústeres las proteínas repetidas que presenten similitud.

- **CoDNaS**

  CoDNaS es una base de datos de diversidad conformacional de las proteínas en su estado nativo, donde contiene información referente a las diversas conformaciones que una proteína cualquiera tiene (Monzon et al., 2013). Esta herramienta será utilizada en el proyecto con la finalidad de acceder a la información de las diferentes conformaciones que una proteína en su estado nativo puede poseer. De la misma manera, se utilizará para recolectar la información relacionada al RMSD y TM-score.

- **Figma**

  Figma es una herramienta que permite el diseño y creación de prototipos basados en web (Figma, 2020). Esta herramienta será utilizada para elaborar el prototipo de la interfaz de usuario a la cual los científicos podrán acceder para evaluar la diversidad conformacional de las proteínas repetidas.

- **Flask**

  Flask es un micro marco de trabajo para aplicaciones basadas en python (Flask MicroFramework, 2020). Esta herramienta se va a utilizar para el desarrollo del

servicio web y la conexión a la base de datos.

- **Git**

  Git es un sistema de control de versiones distribuido gratuito y de código abierto diseñado para manejar todo, desde proyectos pequeños hasta muy grandes, con rapidez y eficiencia (Git, 2020). Esta herramienta se va a utilizar para tener un control adecuado del progreso del proyecto.

- **Mammoth**

  Mammoth es un software para la alineación estructural independiente de la secuencia que permite la comparación entre estructuras de proteína (Ortiz et al., 2002. Se utilizará esta herramienta para alinear las estructuras de cada par de conformaciones y calcular el RMSD.

- **MySQL Workbench**

  Mysql Workbench es una herramienta gráfica de uso libre para trabajar con servidores y bases de datos MySql. Además, permite la conexión a base de datos de MySql, realizar consultas y procedimientos almacenados, realizar backups (Oracle, 2020). Esta herramienta se va a utilizar para poder realizar consultas a la base de datos MySQL y poder extraer la información de la diversidad conformacional de las proteínas repetidas. Asimismo, para el modelamiento de la base de datos.

- **Overleaf**

  Overleaf es una herramienta de publicación y redacción colaborativa en línea que hace que todo el proceso de redacción, edición y publicación de documentos científicos sea mucho más rápido y sencillo (Overleaf, 2020). Esta herramienta se utilizará para la elaboración del informe de las tres propuestas de métodos.

- **Postman**

  Postman es una herramienta qué facilita el diseño, simulación, depuración, prueba, documentación, monitoreo y publicación de API's (Postman, 2020). Se utiliza esta herramienta para monitorear y controlar los servicios que el servicio web va a proporcionar.

- **Pymol**

  Pymol es un sistema de visualización molecular de código abierto que permite visu-

alizar en tiempo real gráficos moleculares de alta calidad (Schrödinger LLC, 2020). Se va a utilizar para verificar que el conjunto de datos de proteínas repetidas es la adecuada.

- **Python**

  Python es un lenguaje de programación interpretado, orientado a objetos y de alto nivel. Asimismo, es de código abierto, simple y fácil de aprender. Además, es muy atractivo para el desarrollo rápido de aplicaciones (Python Software Foundation, 2020). Esta herramienta se utilizará para elaborar el conjunto de datos y datos de prueba de las proteínas repetidas. Además, nos servirá de apoyo para construir el servicio web que permitirá evaluar la diversidad conformacional de las proteínas repetidas.

- **RCSB PDB**

  RCSB PDB está impulsado por la información de archivo del Protein Data Bank sobre las formas 3D de proteínas, ácidos nucleicos y ensamblajes complejos que ayuda a estudiantes e investigadores a comprender todos los aspectos de la biomedicina y la agricultura, desde la síntesis de proteínas hasta la salud y la enfermedad (National Science Foundation et al., 2020b). Esta herramienta será utilizado en el proyecto, ya que, a partir de esta base de datos y repeatsDB, se van a generar el conjunto de datos organizado de las proteínas repetidas.

- **React**

  React es una librería de JavaScript para construir interfaces de usuario (Facebook, 2020). Esta librería se utilizará para construir la interfaz de usuario que permitirá estimar y visualizar la información de diversidad conformacional de proteínas repetidas.

- **RepeatsDB**

  RepeatsDB es una base de datos de estructuras de proteínas repetidas en tándem anotadas, es decir, contiene la información estructural de las proteinas repetidas (Di Domenico et al., 2014). Esta herramienta será utilizada en el proyecto, ya que, a partir de esta base de datos y RCSB PDB, se van a generar el conjunto de datos organizado de las proteínas repetidas.

- **RStudio**

  RStudio es un entorno de desarrollo integrado (IDE) para el lenguaje de programación R dedicado para ls computación estadística y gráficos. Además está disponible para plataformas UNIX, Windows y MacOS (RStudio, 2020). Esta herramienta se utilizará para elaborar gráficos en base a los cálculos obtenidos por el software Mammoth y TM-align.

- **TM-align**

  TM-align es un software que identifica la mejor alineación estructural entre pares de proteínas que combina la matriz de rotación TM-score y la programación dinámica (Zhang and Skolnick, 2005). Este software se utilizará para identificar la región de repetición en cada confómero que presente cada proteína repetida.

- **Visual Studio Code**

  Visual Studio Code es un editor de código disponible para Windows, macOS y Linux. Este cuenta con un ecosistema de extensiones para diferentes lenguajes de programación entre ellos, JavaScript (Visual Studio Code, 2020). Este herramienta será el editor de código de la interfaz de usuario que nos permitirá visualizar la información de diversidad conformacional de las proteínas repetidas.

2. **Métodos**

En esta sección se describirán y se justificarán, en relación al proyecto, los métodos mencionados en la Tabla 1.5.

- **Normalización de base de datos**

  La normalización de base de datos es una metodología que consiste en aplicar una serie de reglas a las relaciones obtenidas al modelo relacional con la finalidad de reducir la redundancia de datos (Codd, 2002).

  En el presente proyecto se va a utilizar esta metodología para facilitar la gestión del acceso a la base de datos que va a contener la información de diversidad conformacional. Asimismo, esto nos ayudará en tener un modelamiento más ordenado con mínima redundancia de datos.

- **Reuniones**

  Se van a establecer reuniones con la aseora y los co-asesores para mostrar las propuestas de métodos y el modelamiento planteado de la base de datos con la finalidad de obtener su aprobación. Además, también se acordarán reuniones para mostrar los avances del desarrollo del servicio web y la interfaz de usuario que permitirá la visualización de la información de diversidad conformacional de las proteínas repetidas.

Capítulo 2

Marco Conceptual

2.1. Introducción

En el Marco Conceptual se buscar describir conceptos relevantes para el entendimiento de la diversidad conformacional y las proteínas repetidas. Asimismo, este capítulo se dividirá en 2 secciones: Objetivo del Marco Conceptual y el Desarrollo del Marco Conceptual. La primera sección mostrará la finalidad de realizar el marco conceptual en el presente proyecto de tesis y la segunda sección detallará los conceptos de estado nativo de la proteína, diversidad conformacional, aminoácidos, proteínas y proteínas repetidas para tener una mejor comprensión del análisis de diversidad conformacional en proteínas repetidas.

2.2. Objetivo del Marco Conceptual

En el presente marco conceptual se busca definir los conceptos necesarios para comprender la importancia de elaborar un método y una herramienta que permita analizar la diversidad conformacional en proteínas repetidas.

2.3. Desarrollo del Marco Conceptual

2.3.1. Aminoácidos

Los aminoácidos son pequeñas moléculas y hay 20 tipos, entre las cuales tenemos a la Alanina, Arginina, Tirosina y demás. Además, dependiendo de la combinación en número y

secuencia distinta de estos aminoácidos se puede formar una u otra proteína (Lodish et al., 2003). Por otra parte, las propiedades distintivas de los aminoácidos, determinadas por sus cadenas laterales, permiten comprender las estructuras y funciones de las proteínas (Lodish et al., 2003). Estas cadenas laterales vienen a ser grupos químicos que se adhieren a la cadena principal y pueden variar en tamaño, forma, carga, hidrofobicidad y reactividad dependiendo del aminoácido (Lodish et al., 2003). Además, la estructura general de un aminoácido se establece por la presencia de un carbono alfa (Cα) unido a un ácido carboxílico, a un amino y a una cadena lateral (Lodish et al., 2003).

2.3.2. Proteínas

Las proteínas son macromoléculas generadas por la célula con la finalidad de cumplir diversas funciones dentro del organismo (Lodish et al., 2003). Además, estas se clasifican, dependiendo de la función que cumplen como las proteínas estructurales que dan rigidez a las células, las proteínas regulatorias que actúan como sensores y cambian la función genética, las proteínas de transporte que se encargan del control de flujo de materiales a lo largo de la membrana celular, las proteínas motoras que generan el movimiento y las proteínas receptoras que se encargan de transmitir señales externas al interior de la célula (Lodish et al., 2003).

Asímismo, la estructura de las proteínas se pueden describir en cuatro niveles de jerarquía de estructura como la estructura primaria que es el primer nivel de jerarquía y es un arreglo linear, o secuencia, de los residuos de aminoácidos que la componen; la estructura secundaria que es el segundo nivel de jerarquía y consiste en varios arreglos espaciales resultantes del plegamiento de partes localizadas de una cadena de polipéptidos; la estructura terciaria que es el tercer nivel de jerarquía y es la conformación general de una cadena de polipéptidos, es decir, es el arreglo tridimensional de todos los residuos del aminoácido; y la estructura cuartenaria que es el cuarto nivel de jerarquía y describe el número y las posiciones relativas de las subunidades en proteínas multiméricas (Lodish et al., 2003).

2.3.3. Proteínas Repetidas

Las proteínas repetidas son una clase de proteína generalizada que prevalece en eucariotas, pero que también están presentes en procariotas y arqueas (Marcotte et al., 1999), Estas proteinas tienen una característica particular de presentar repeticiones en su estructura, es decir

que contienen patrones de estructura de aminoácidos repetidos una al lado de la otra. Estos patrones son conocidos como repeticiones en tándem (TR) y no siempre son repeticiones exactas, sino que varían. Esto se debe a que durante la evolución de la proteína, una serie de mutaciones se acumularon a tal punto que modificaban la secuencia de patrones y esto generó que no sea tan fácil identificar algunas proteínas a simple vista (Kajava, 2012). Para ello, existe una clasificación estructural basada en la longitud de la repetición (Kajava, 2012) y son las siguientes.

1. **Clase I - Cristalinos:**

   Esta clase de estructuras incluye proteínas y péptidos con 1 o 2 repeticiones de residuos que forman diferentes tipos de cristalitos de tamaño ilimitado que son perjudiciales para los organismos vivos (Kajava, 2012).

2. **Clase II - Estructuras Fibrosas:**

   Esta clase de estructuras está estabilizada por interacciones entre cadenas. Asimismo, tienen dimensiones bien definidas y el tamaño de sus unidades repetidas van de 3 a 4 residuos (Kajava, 2012).

3. **Clase III - Estructuras Alargadas:**

   En esta clase de estructuras las unidades repetitivas se requieren entre sí para mantener la estructura. Además, el tamaño de esta clase está entre 5 a 40 residuos. Asimismo, esta clase está dominada por proteínas solenoides (Kajava, 2012).

4. **Clase IV - Estructuras "Cerradas":**

   En esta clase de estructuras las unidades repetitivas se necesitan entre sí para tener estructura. Además, muchas proteínas en esta clase tienen un número fijo de repeticiones debido a sus estructuras circulares o "cerradas". Asimismo, las longitudes de repetición de las estructuras cerradas se superponen con las estructuras de clase III y V (Kajava, 2012).

5. **Clase V - Estructuras de "Beads on a string":**

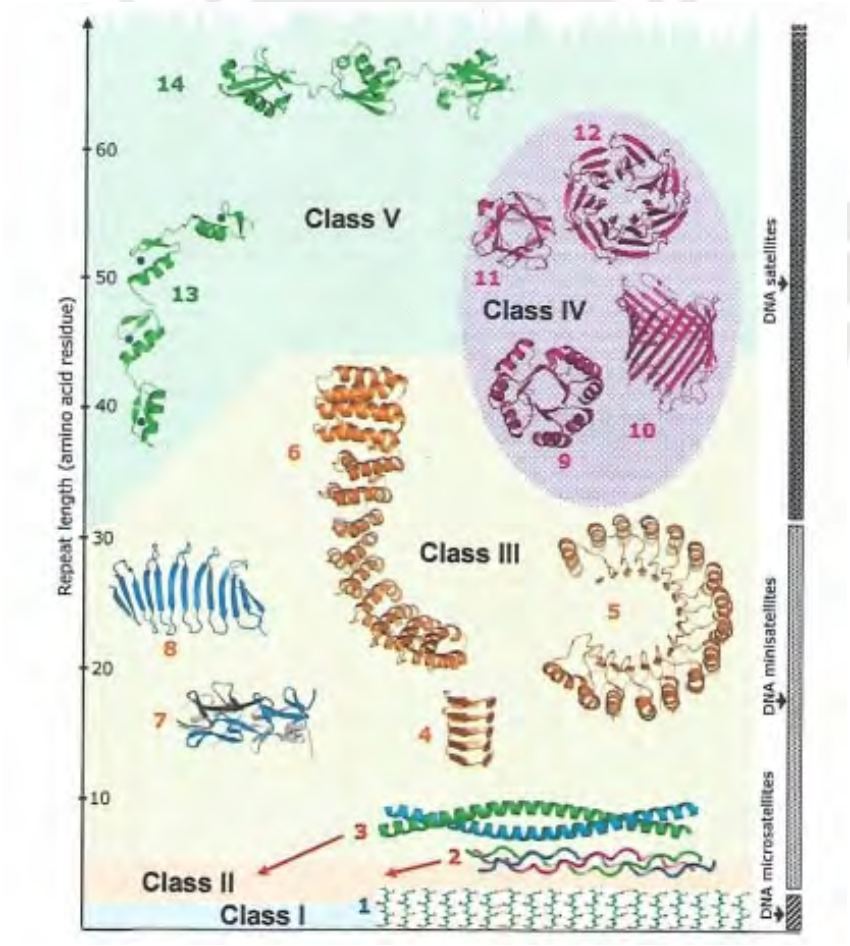
   Esta clase de estructuras incluye unidades repetitivas que ya son lo suficientemente grandes como para plegarse independientemente en dominios estables. Cabe señalar que el tamaño de esta clase es de más de 50 a 60 residuos (Kajava, 2012).

Asimismo, esta clasificación se puede apreciar en la Figura 2.1. Además, las proteínas repetidas no solo están relacionadas con la evolución sino también con la variabilidad del genoma (Kachroo et al., 1997) y los procesos de enfermedades (Djian, 1998), como la enfermedad de Huntington.

Un ejemplo de proteínas repetidas relacionadas a las enfermedades se puede observar en las estructuras fibrosas homotriméricas de las proteínas bacterianas. Estas proteínas con estructura β antiparalela con repeticiones más largas han sido encontradas en la proteína A transportadora de lipopolisacárido periplásmico (LPS) de bacterias gram-negativa. Dichas bacterias ocasionan enfermedades (Suits et al., 2008).

Figure 2.1

La clasificación estructural de las proteínas repetidas.



Nota: El gráfico representa las clasificaciones de las proteínas repetidas basado en la longitud de sus repeticiones. Extraído de Kajava, 2012.

2.3.4. Estado Nativo de la proteína

El concepto de estado nativo ha evolucionado a través del tiempo. Desde el año 1936, donde Mirsky y Pauling definen al estado nativo de una proteína como una cadena polipeptídica contínua y no ramificada que adopta un plegado o doblez con una estructura definida de forma única (Mirsky and Pauling, 1936); hasta, la definición que se conoce hoy por hoy, que el estado nativo de una proteína se describe como un ensamble de confórmeros. Estos confórmeros vienen a ser nada menos que alternativas de estructuras que puede adoptar una proteína dependiendo del ambiente en la que se encuentre (Kumar et al., 2000; Tsai et al., 1999).

Por ejemplo, si una célula se encuentra en un estado en el cual su pH es alta, la proteína que está en la célula adoptará un confórmero diferente a la que tendría si estuviera en una célula en un estado con pH baja. Entonces estos dos confórmeros diferentes vienen a ser las diferentes estructuras que esta proteína toma, por lo tanto, este conjunto de estructuras serían parte del estado nativo de esta proteína.

2.3.5. Diversidad Conformacional

Se le llama diversidad conformacional a las diferencias estructurales que una proteína presenta en su estado nativo y es por medio de este análisis que se puede relacionar la función de la proteína con los cambios conformacionales, es decir, con los cambios que una proteína puede presentar en su estructura dependiendo del entorno en la que se encuentre (James and Tawfik, 2003; Parisi et al., 2015). Asimismo, esta diversidad permite una diversificación en la capacidad funcional a través de un mecanismo, de tal manera que si una proteína adopta diferentes estructuras, esta podría presentar diferentes funciones también (James and Tawfik, 2003).

Por ejemplo, la proteína drk (Figura 2.2) puede adoptar aproximadamente 60 estructuras estables y diferentes, presentando significativas diferencias estructurales entre algunas de ellas (Choy and Forman-Kay, 2001). Asimismo, estas diferentes estructuras pueden presentar diferentes funciones (James and Tawfik, 2003).

Figure 2.2

Proteína drk

Nota: El gráfico presenta el dominio N-terminal SH_3 de la proteína drk de Drosophila. Extraído de National Science Foundation et al., 2020b.

2.3.6. Representación gráfica de estructuras de proteínas

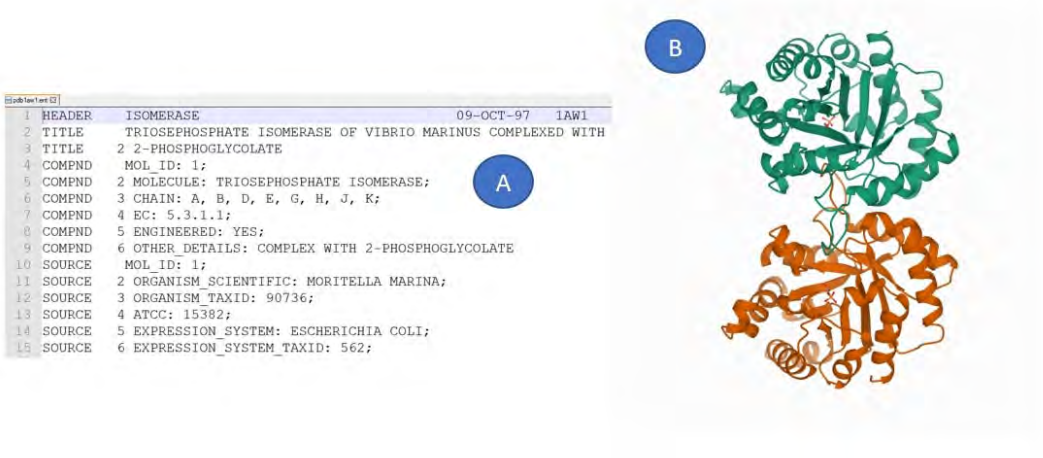
La representación gráfica de estructuras de proteínas está basada en archivos en formato pdb (pdb files). Estos archivos enumeran los átomos en cada proteína y su ubicación 3D en el espacio. Además, los pdb files incluyen una sección de encabezado, una sección de coordenadas atómicas y una sección de observaciones experimentales (National Science Foundation et al., 2020a).

En la primera sección está la información general de la proteína como el artículo de donde se le hizo estudio, el nombre, la clase y el organismo a la que pertenece. En la segunda sección se encuentran una lista de registros ATOM detallan las posiciones de los átomos en el espacio con su respectiva coordenada X, Y y Z; y los registros TER que indican el final de una lista de registros ATOM (National Science Foundation et al., 2020a).

Como ejemplo de lo mencionado previamente, a continuación, en la Figura 2.3, se observa el pdb file y la imagen de la estructura 3D de la proteína 1AW1 basada en el pdb file de la misma.

Figure 2.3

PDB File e imagen de la estructura 3D de la proteína 1AW1.



Nota: A: PDB File de la proteína 1AW1. B: Imagen de la estructura 3D de la proteína 1AW1 elaborada utilizando el software PyMOL. Elaboración Propia.

Capítulo 3

Marco Teórico

3.1. Introducción

En el presente capítulo se hará mención del objetivo del marco teórico y se describirá los conceptos teóricos que se manejarán en la presente tesis.

3.2. Objetivos del Marco Teórico

En el presente marco teórico busca describir detalladamente cada uno de los elementos de la teoría que serán directamente utilizados en el desarrollo de la tesis.

3.3. Desarrollo del Marco Teórico

3.3.1. Raíz de la desviación cuadrática media (RMSD)

La desviación cuadrada media de la raíz (RMSD), también conocida como la distancia matriz error (Maiorov and Crippen, 1994), es descrita como la medida que calcula la distancia entre dos coordenadas atómicas superpuestas (Kufareva and Abagyan, 2012) por cada amino ácido; generalmente, se utilizan los Cα de cada aminoácido de la proteína. Por ejemplo, la base de datos CoDNaS utiliza el algoritmo MAMMOTH para alinear las estructuras de la proteína y a su vez, realizar el cálculo del Cα RMSD (Monzon et al., 2016).

Por lo tanto, en la presente tesis, cada vez que se haga mención a un valor de RMSD, será aquel obtenido utilizando como referencia los Cα.

3.3.2. TM-align

TM-align es un algoritmo que identifica el alineamiento estructural entre pares de proteínas que combina la matriz de rotación de puntuación TM y la programación dinámica (DP). Asimismo, TM-align solo emplea las coordenadas backbone Cα de las estructuras proteicas dadas; sin embargo, se puede generalizar fácilmente a cualquier tipo de átomo (Zhang and Skolnick, 2005).

Es así que, para la presente tesis, cada vez que se mencione el algoritmo TM-align, este empleará las coordenadas backbone Cα.

Capítulo 4

Estado del Arte

4.1. Introducción

En el presente capítulo, se mostrarán las investigaciones disponibles ligadas a los temas de diversidad conformacional y proteínas repetidas. Estas investigaciones se usarán como referencia para la elaboración del proyecto de tesis utilizando la metodología de revisión sistemática. Esta revisión es un proceso iterativo que permitirá identificar, interpretar y evaluar toda investigación relevante relacionada con una pregunta de investigación (Kitchenham, 2004).

4.2. Objetivo de Revisión

Se realizará una revisión sistemática para conocer el estado del arte actual de la diversidad conformacional de las proteínas repetidas. Esto por medio de la identificación de publicaciones del tema, dentro de las diversas bases de datos utilizadas. El objetivo de esto, es recopilar información acorde a criterios establecidos para dar respuesta a la siguiente pregunta: *¿De qué manera se puede evaluar la diversidad conformacional en proteínas repetidas?*

4.3. Preguntas de Revisión

A partir del objetivo de revisión, se formularon las siguientes preguntas:

RQ1. ¿Qué métodos se utilizan para evaluar la diversidad conformacional en proteínas y cuáles son sus características?

RQ2. ¿Cuáles son los resultados de los métodos actuales que evaluan la diversidad conformacional en proteínas?

RQ3. ¿Qué características presentan las proteínas repetidas y las repeticiones en tándem dentro de una proteína?

4.4. Estrategia de búsqueda

4.4.1. Motores de búsqueda

Los motores de búsqueda que se utilizarán son:

1. Scopus
2. ScienceDirect
3. Pubmed

4.4.2. Cadenas de búsqueda

Se utilizarán dos cadenas de búsqueda que nos ayudarán a recopilar documentos con el fin de responder las preguntas de investigación. Asimismo, para la elaboración de estas cadenas se utilizaron los conectores AND y OR.

1. **Cadena de Búsqueda N° 1:**

   Esta cadena de búsqueda se encargará de filtrar documentos relacionados con la diversidad conformacional en proteínas. Esto con la finalidad de contestar la pregunta 1 (RQ1) y la pregunta 2 (RQ2) de investigación.

   TITLE-ABS-KEY (“conformational diversity” AND “protein” AND (“native state” OR “conformer” OR “evolution” OR “exploring”) AND NOT (“loop” OR “peptides” OR “network” OR “ions” OR “catalyzed” OR “cyclic”)) AND (AUTHLASTNAME (“Tawfik”) OR PUBYEAR $>$ 2012) AND (LIMIT-TO (SUBJAREA , “BIOC”))

2. **Cadena de Búsqueda N° 2:**

   Esta cadena de búsqueda se encargará de filtrar documentos relacionados con las proteínas repetidas y las repeticiones en tándem dentro de una proteína. Esto con la finalidad de

responder la pregunta 3 (RQ3) de investigación.

TITLE-ABS-KEY (“protein repeats” AND ((“tandem repeats” AND (“function” OR “structure”)) OR (“function” AND “structure”)) AND NOT (“molecule” OR “computational” OR “network”)) AND (AUTHLASTNAME (“Andrade” AND “Ponting”) OR PUBYEAR $>$ 2011) AND (LIMIT-TO (SUBJAREA , “BIOC”))

4.4.3. Documentos encontrados

A continuación, en la Tabla 4.1 se presenta la cantidad de documentos encontrados por cada motor de búsqueda.

Tabla 4.1

Cantidad de documentos encontrados

Motor de Búsqueda	**Cadena de Búsqueda**	**Artículos Encontrados**	**Artículos Duplicados**	**Artículos Relevantes**
Scopus	1	16	0	11
	2	15	0	10
Pubmed	1	12	11	1
	2	12	12	0
ScienceDirect	1	13	13	0
	2	11	11	0
Subtotal	1	41	24	12
	2	38	23	10
	Total	**79**	**47**	**22**

4.4.4. Criterios de inclusión/exclusión

Los criterios que se tomarán en cuenta son:

1. **Criterios de inclusión**

   - Se van a tomar en cuenta los artículos que pertenecen al área de ciencias de la computación, bioinformática y biología estructural, ya que los temas de diversidad conformacional y proteínas repetidas corresponden a estas áreas.

- Se van a considerar los artículos que desarrollan el tema de diversidad conformacional en proteínas, ya que la información rescatada de estos documentos ayudará a responder las preguntas de revisión.
- Se tomarán en cuenta los artículos que detallan un método para la evaluación de la diversidad conformacional, con el fin de recopilar información para responder las preguntas de revisión.
- Se considerarán los artículos que detallan las características de las proteínas repetidas y las repeticiones en tándem dentro de una proteína, con el fin de recopilar información para responder las preguntas de revisión.
- Se van a considerar los artículos publicados en inglés, ya que las revistas relacionadas al tema en cuestión están en dicho idioma.
- Se considerarán adicionalmente los artículos que tienen como autor a Dan S. Tawfik (h-index: 83) y a Edward M. Marcotte (h-index: 78), ya que son investigadores reconocidos en el área.

2. **Criterios de exclusión**

- Los artículos que tengan fecha de publicación de más de 8 años de antigüedad, exceptuando los que tienen como autor a Dan S. Twafik y a Edward M. Marcotte, no se van a considerar porque no son de relevancia al tema de investigación.
- Cualquier tipo de documento que no se difunde por los canales ordinarios de publicación, se van a considerar como literatura gris, por ejemplo los artículos que son publicados en revistas no científicas. Estos documentos no se van a considerar, porque no están respaldados por el método científico.
- Se considera la versión más reciente del artículo en caso de existir versiones de mayor antigüedad[1], ya que esta contiene mejoras de metodología, incremento de datos o nuevas metodologías implementadas.

[1]. Por ejemplo el artículo el artículo relacionado con la base de datos CoDNaS. El artículo "CoDNaS: a database of conformational diversity in the native state of proteins". La primera versión se publicó en el año 2013, mientras que en el año 2016 se publicó el artículo "CoDNaS 2.0: a comprehensive database of protein conformational diversity in the native state", en esta nueva versión se presentó una nueva actualización de la base de datos. Para nuestro caso particular se utilizará la versión 2016.

4.5. Estudios Primarios

Para los estudios primarios, teniendo en cuenta los criterios de inclusión y exclusión, se recolectaron un total de 79 documentos utilizando las cadenas de búsqueda en las bases de datos Scopus, Pubmed y ScienceDirect entre el 20/04/2020 y el 24/04/2020. De estos documentos, 47 son artículos duplicados y solo 22 documentos son relevantes para la revisión sistemática. A continuación, se mostrarán los artículos relevantes por cada cadena de búsqueda.

1. Los documentos identificados por la cadena de búsqueda N° 1 se encuentran en el Anexo B, Tabla B.1.

2. Los documentos identificados por la cadena de búsqueda N° 2 se encuentran en el Anexo B, Tabla B.2.

4.6. Formulario de extracción de datos

A continuación, en la Tabla 4.2 se presenta el diseño del formulario de extracción de datos.

Tabla 4.2

Diseño del formulario de extracción de datos

Campo	Descripción	RQ
Id	T[número]. P.ej: T01	General
Fecha de Extracción		General
Autores		General
Título		General
Tipo de Fuente	Revista, congreso, papers o tesis	General
Fuente	Nombre de la revista, congreso, papers o tesis	General
Año de publicación		General
Afiliación	Instituciones de afiliación de los investigadores	General
País	País(es) de afiliación de los investigadores	General
Método(s)	Qué métodos se utilizan para evaluar la diversidad conformacional en proteínas y cuáles son sus características.	RQ1

Resultado(s)	Cuáles son los resultados de los métodos actuales que evaluan la diversidad conformacional en proteínas.	RQ2
Característica(s)	Qué características presentan las proteínas repetidas y las repeticiones en tándem dentro de una proteína.	RQ3

4.7. Resultados de la revisión

4.7.1. Formulario de extracción

Se utilizará el diseño del formulario de extracción de datos en los artículos identificados previamente. Cabe señalar que hay dos conjuntos de artículos identificados, ya que se hace uso de dos cadenas de búsqueda. La cadena de búsqueda N° 1 corresponden a las preguntas N° 1 y N° 2 de investigación, y la la cadena de búsqueda N° 2 corresponde a la pregunta N° 3 de investigación.

Dicho esto, se presentarán dos formularios de extracción de datos llenados con la información de cada conjunto de documentos.

1. El formulario de extracción de datos para el conjunto de artículos identificados por la cadena de búsqueda N° 1 se encuentran en el Anexo C, Tabla C.1.

2. El formulario de extracción de datos para el conjunto de artículos identificados por la cadena de búsqueda N° 2 se encuentran en el Anexo C, Tabla C.2.

4.7.2. Respuestas a las preguntas de investigación

A continuación, se responden las preguntas de investigación a través de los artículos identificados previamente. En la Tabla 4.3 se muestra la cantidad de artículos relacionados a cada interrogante de investigación.

Tabla 4.3

Cantidad de artículos relacionados a cada interrogante de investigación

Preguntas de Investigación	**Número de artículos**	**Artículos**
¿Qué métodos se utilizan para evaluar la diversidad conformacional en proteínas y cuáles son sus características?	6	DC01 (Saldaño et al., 2020), DC02 (Monzon et al., 2019), DC07 (Monzon et al., 2016), DC08 (Palopoli et al., 2016), DC09 (Parisi et al., 2015), DC12 (James and Tawfik, 2003)
¿Cuáles son los resultados de los métodos actuales que evalúan la diversidad conformacional en proteínas?	7	DC02 (Monzon et al., 2019), DC03 (Rueda et al., 2018), DC04 (Monzon, Zea, Fornasari, et al., 2017), DC05 (Zea et al., 2016), DC06 (Monzon, Zea, Marino-Buslje, et al., 2017), DC10 (Javier Zea et al., 2013), DC11 (Juritz et al., 2013)
¿Qué características presentan las proteínas repetidas y las repeticiones en tándem dentro de una proteína?	10	PR01 (Kajava, 2012), PR02 (Di Domenico et al., 2014), PR03 (Brunette et al., 2015), PR04 (Andrade et al., 2001), PR05 (Hannan, 2018), PR06 (Hirsh et al., 2016), PR07 (Delucchi et al., 2020), PR08 (Paladin and Tosatto, 2015), PR09 (Schaper et al., 2015), PR10 (Marcotte et al., 1999)

1. **Pregunta 1: ¿Qué métodos se utilizan para evaluar la diversidad conformacional en proteínas y cuáles son sus características?**

   Para empezar, la diversidad conformacional está definida por las diferencias estructurales de confórmeros (Parisi et al., 2015). Asimismo, estas diferencias generalmente se miden por la raíz de la desviación cuadrática media, RMSD (Monzon et al., 2019), ya que es más sensible a los movimientos en bucles y colas (Palopoli et al., 2016). En adición, esta medición se utiliza para caracterizar la diversidad conformacional. Además, para realizar una estimación de estas diferencias estructurales, es importante tener en cuenta el número de confórmeros por proteína (Monzon et al., 2019).

   Por un lado, existen varios métodos para calcular el RMSD global entre un par de estructuras y la magnitud de este valor depende del algoritmo de alineación estructural utilizado (Monzon et al., 2019). Por ejemplo, CoDNaS usa MAMMOTH para alinear cada par de confórmero con el fin de estimar el RMSD global (Monzon et al., 2019). Además, CoDNaS estima el grado de diversidad conformacional utilizando diferentes medidas de similaridad estructural global y local (Monzon et al., 2016). Asimismo, esta base de datos permite al usuario explorar cómo cambian las diferencias estructurales entre los confórmeros en función de varias características estructurales que proporcionan más información biológica (Monzon et al., 2016).

   No obstante, el cálculo del RMSD tiene dependencia del análisis conformacional, ya que por medio de este, se puede evitar sesgos en los valores de RMSD si no se mezcla los confórmeros obtenidos por los métodos de resonancia magnética nuclear (NMR) y rayos X (Monzon et al., 2019).

   De igual forma, hacer uso del RMSD es poder dar un grado a la diversidad conformacional; sin embargo, estas diferencias estructurales se pueden simular utilizando métodos computacionales como las técnicas de Dinámica Molecular (MD), las cuales padecen de ineficiencia para alcanzar barreras de alta energía entre los confómeros (Saldaño et al., 2020); el análisis de modo normal de grano grueso (Parisi et al., 2015); y el método que se enfoca en tres pasos: el primer paso es la identificación de la red de interacción de residuos (RIN), el segundo paso es el análisis de modos normales (NMA) basado en este RIN y el tercer paso es la generación, selección y optimización de nuevas estructuras de proteínas desplazadas en la dirección de modos NMA seleccionados (Saldaño et al.,

2020). También, cabe señalar que desde la evidencia experimental de la diversidad conformacional, esta proviene del análisis de cristales de proteínas y resonancia magnética nuclear (NMR) de proteínas (Parisi et al., 2015).

Para terminar, además de los métodos ya mencionados, está la cinética de unión en estado preestable, esta fue la que proporcionó los primeros datos que indicaban isómeros preexistentes en equilibrio, es decir, este análisis reveló la existencia de la diversidad conformacional; sin embargo, es inalcanzable en muchos casos (James and Tawfik, 2003).

2. **Pregunta 2: ¿Cuáles son los resultados de los métodos actuales que evaluan la diversidad conformacional en proteínas?**

   Para empezar, haciendo uso de la raíz de la desviación cuadrática media (RMSD) máxima como medida estadística para evaluar la diversidad conformacional en proteínas, se realizó un estudio de la extensión de la diversidad conformacional sobre un conjunto de datos curados de 5000 proteínas con más de cinco confórmeros por proteína (Monzon, Zea, Fornasari, et al., 2017). Este estudio encontró tres clases de proteínas basadas en su comportamiento dinámico. Estas son las proteínas rígidas, maleables y parcialmente desordenadas (Monzon et al., 2019). Asimismo, se ha demostrado que las proteínas con gran diversidad conformacional muestran tasas evolutivas más bajas que las proteínas con confórmeros más similares (Javier Zea et al., 2013).

   Por un lado, cuando se tiene en cuenta la diversidad conformacional, la relación entre secuencia y divergencia estructural es más compleja (Monzon, Zea, Marino-Buslje, et al., 2017), ya que con una identidad de alrededor del 100%, varias proteínas muestran RMSD tan altas como las alcanzadas por la divergencia de secuencia durante la evolución. Esto significa que la divergencia estructural es un proceso complejo ya que una secuencia dada podría alcanzar varios angstroms de diversidad conformacional (Monzon et al., 2019). Teniendo una distribución general de RMSD entre confórmeros para todas las cadenas de proteínas contenidas en CoDNaS obtenidas por cristalografía de rayos X se pudo inferir que la mayoría de las proteínas requieren pequeños movimientos entre los confómeros para cumplir sus funciones biológicas (Monzon, Zea, Fornasari, et al., 2017).

   Por otra parte, las proteínas en solventes acuosos muestran mayores proporciones de diversidad conformacional medidas por la desviación cuadrada media de la raíz (RMSD)

máximo que aquellas en solventes no acuosos (Rueda et al., 2018). Asimismo, los confórmeros en medios no acuosos tienen cavidades más grandes, menos superficies expuestas a solventes y menos regiones desordenadas (Rueda et al., 2018).

También, se ha encontrado que las proteínas que muestran transiciones de desorden de orden entre confórmeros muestran valores de RMSD más altos que las proteínas que no muestran transiciones (Zea et al., 2016). Asimismo, las proteínas ordenadas tiene en general una baja diversidad conformacional (Monzon, Zea, Fornasari, et al., 2017).

Para terminar, es relevante considerar la diversidad conformacional en la comprensión de los mecanismos de evolución de proteínas. Asimismo, la diversidad conformacional y el sesgo derivado de sustitución de aminoácidos son aspectos esenciales a tener en cuenta para el desarrollo de nuevas herramientas bioinformáticas (Juritz et al., 2013).

3. **Pregunta 3: ¿Qué características presentan las proteínas repetidas y las repeticiones en tándem dentro de una proteína?**

   Para empezar, una cantidad considerable de proteínas contiene patrones de secuencia o estructura de aminoácidos repetidos de forma adayacente, es decir, los patrones o unidades repetidas de esta, se encuentran uno al lado del otro. Esto es conocido como repeticiones de tándem (TR) en proteínas y usualmente se presentan como repeticiones imperfectas (Delucchi et al., 2020). Además, estas son comunes en la naturaleza y se pueden encontrar de varias formas, por lo que son díficiles de reconocer (Paladin and Tosatto, 2015), porque la unidad repetida es relativamente corta y puede haber una considerable divergencia de secuencia entre las unidades de la misma TR (Andrade et al., 2001). Para esto, existe una clasificación estructural de repeticiones en tándem, la cual está implementada en la base de datos RepeatsDB (Kajava, 2012). Cabe señalar que las TR se describen por el número de unidades repetidas, la longitud del patrón repetido, y la similiridad estructural. (Schaper et al., 2015). En adición, una unidad repetida es definida como el bloque de construcción estructural más pequeño que forma una región repetida y este es un grupo de al menos tres unidades de repetición; además, esta región puede incluir inserciones, es decir, segmentos de estructuras no repetidas que ocurren dentro de una unidad de repetición o entre dos de ellas (Di Domenico et al., 2014).

   Por otro lado, las TR abundan en gran cantidad en el proteoma humano y se estima que

el 14% de las proteínas contienen al menos una TR (Marcotte et al., 1999). Sin embargo, un nuevo enfoque, menciona que un 50.9% de las proteínas contienen una TR, el cual suele estar ubicado en los flancos de secuencias (Delucchi et al., 2020). En adición, las TR muestran una diversa variación de tamaños, estructuras y funciones (Kajava, 2012). Asimismo, estas están asociadas con funciones y enfermedades relacionadas con la inmunidad (Hannan, 2018). Por ejemplo, la esclerosis lateral amiotrófica, la distrofia miotónica, la atrogia muscular espinobulbar, entre otras, son causadas por trastornos de repetición de tándem (Hannan, 2018).

Las proteínas repetidas son una clase de proteína generalizada que prevalece en eucariotas, pero que también están presente en bacterias y arqueas (Kajava, 2012). Además, estas realizan funciones únicas de las eucariotas (Paladin and Tosatto, 2015). También, se cree que las TRs surgieron de la duplicación intragénica y eventos de recombinación (Andrade et al., 2001). Por su parte, la clasificación de la proteína repetida se basa en la longitud de la repetición (Kajava, 2012), la cual puede variar de uno o dos repeticiones de residuos que forman diferentes tipos de cristalitos, Clase I; de cinco a cuarenta repeticiones de residuos que son dominadas por las estructuras alargadas llamadas solenoides, Clase III; y de más de cincuenta residuos en "beads on a string", Clase V (Kajava, 2012). Asímismo, están las estructuras cerradas donde las unidades repetidas se necesitan entre sí para tener una estructura, Clase IV; y las estructuras fibrosas estabilizadas por interacciones entre cadenas, Clase II (Kajava, 2012).

Por último, hay un interés creciente en proteínas repetidas en los últimos años, debido a su relevancia en la salud (Hannan, 2018) y sus aplicaciones para ingeniería (Brunette et al., 2015). Sin embargo, esta clase de proteínas todavía pertenece a la "materia oscura" del universo proteico que se caracteriza por relaciones no canónicas de secuencia-estructura (Hirsh et al., 2016).

4.8. Relación con productos similares

El presente proyecto no cuenta con alguna relación con productos similares en el mercado, debido a que es una nueva propuesta que se quiere realizar.

4.9. Conclusiones

Luego de revisar los artículos de investigación se llegaron a las siguientes conclusiones.

En primer lugar, la diversidad conformacional está relacionada con las funciones de la proteína y realizar este análisis abre las puertas de descubrir nuevas funciones de las proteínas, asi como, tener un mejor entendimiento de las funciones ya existentes (Monzon et al., 2019).

En segundo lugar, la medida más utilizada para calcular, sea cual sea el algoritmo de alineamiento estructural que se utilice, es la desviación cuadrada media raíz, RMSD (Parisi et al., 2015). Por lo que, esta medida nos permitirá evaluar y darle un grado a la diversidad conformacional a las proteínas (Palopoli et al., 2016).

En tercer lugar, las proteínas repetidas tienen la característica particular de tener repeticiones en su estructura (Andrade et al., 2001) y esta particularidad se puede encontrar en enfermedades humanas, por lo que esto generó un interés en los último años (Hannan, 2018).

Como última conclusión, hasta el momento no se ha realizado el análisis de diferencias estructurales en proteínas repetidas. Sin embargo, si se realiza este análisis de diversidad conformacional en un conjunto de proteínas repetidas, permitiría un mejor entendimiento no solo de las propiedades de esta clase de proteínas, sino también de las propiedades de todas las proteínas existentes, debido a que las proteínas repetidas tiene información fundamental para explicar las diversidades estructurales contemporáneas (Goncearenco and Berezovsky, 2015).

Capítulo 5

Conjunto de datos organizado de las proteínas repetidas

5.1. Introducción

En el presente capítulo se presenta el desarrollo del objetivo específico 1, el cual tiene como propósito generar un conjunto de datos organizado de las proteínas repetidas, a partir de la base de datos RepeatsDB, que servirá como base para realizar el análisis de diversidad conformacional. Para lograr este objetivo, primero se definió la estructura de datos organizada que va a representar a los conjuntos de datos de las proteínas repetidas. Posteriormente, teniendo como base esta estructura de datos, se generó el conjunto de datos de las proteínas repetidas, el cual será utilizado como dato de entrada para el análisis de diversidad conformacional y, finalmente, se generó el conjunto de datos de prueba de las proteínas repetidas, los cuales permitirán evaluar la efectividad de los 4 métodos (2 métodos genéricos y 2 propuestas de métodos) que analizarán la diversidad conformacional de las proteínas repetidas.

5.2. Resultados Alcanzados

5.2.1. Estructura de datos organizada para representar el conjunto de datos de las proteínas repetidas

La estructura de datos organizada es la estructura que va a representar al conjunto de datos de las proteínas repetidas que se van a utilizar como datos de entrada para el análisis de diver-

sidad conformacional; asimismo, va a representar al conjunto de datos de prueba que servirán para evaluar la efectividad de los 4 métodos (2 métodos genéricos y 2 propuestas de métodos) que analizarán la diversidad conformacional de las proteínas repetidas.

Además, esta estructura de datos organizada se definió en base a la sección de coordenadas atómicas de un archivo de estructura 3D de una proteína cualquiera (pdb file[1]) de la base de datos RCSB PDB, ya que esta sección se utiliza para identificar a la estructura 3D de la proteína. Es así que, la estructura de datos está representada por los parámetros del registro ATOM y del registro TER, los cuales se pueden ver en la Tabla 5.1 y en la Tabla 5.2, respectivamente.

Tabla 5.1

Estructura de datos organizado del registro ATOM

#	Tipo de Dato	Contenido
1	Nombre de registro	"ATOM"
2	Entero	Número secuencial del átomo
3	Átomo	Nombre del átomo
4	Caracter	Indicador de ubicación alternativa
5	Nombre del residuo	Nombre del residuo
6	Caracter	Identificador de la cadena
7	Entero	Número de la secuencia de residuo
8	Real	Coordenada X del átomo de la proteína
9	Real	Coordenada Y del átomo de la proteína
10	Real	Coordenada Z del átomo de la proteína
11	Real	Factor de ocupación
12	Real	Temperatura

[1]La representación gráfica de estructuras de proteínas se encuentra en la sección 2.3.6

Tabla 5.2

Estructura de datos organizado del registro TER

#	Tipo de Dato	Contenido
1	Nombre de registro	“TER”
2	Entero	Número secuencial del átomo[2]
3	Nombre del residuo	Nombre del residuo[3]
4	Caracter	Identificador de la cadena[4]
5	Entero	Número de la secuencia del residuo[5]

Por otro lado, la manera en cómo se definió esta estructura, la cual representará al conjunto de datos organizado de las proteínas repetidas, se puede apreciar en la Figura 5.1, la cual detalla brevemente los pasos para la creación de la misma. Asimismo, la descripción más a detalle de este resultado esperado se puede encontrar en el Anexo D.2: Informe de la estructura de datos organizada, en la sección Definición de la estructura de datos.

Figure 5.1

Pasos para definir la estructura de datos organizada

Nota: El gráfico muestra los pasos que se siguieron para definir la estructura de datos organizada. Elaboración Propia.

[2]El número secuencial del átomo es un parámetro opcional.
[3]El nombre del residuo es un parámetro opcional.
[4]El identificador de la cadena es un parámetro opcional.
[5]El número de la secuencia del residuo es un parámetro opcional.

Por otra parte, para verificar que la estructura definida es la adecuada se ha usado el software PyMOL, el cual nos permite ver la imagen 3D de la estructura de una proteína, pudiendo validar así que la estructura definida es la adecuada. Esta verificación se puede encontrar más a detalle en el Anexo D.3: Informe de la estructura de datos organizada, en la sección Verificación de la estructura de datos.

Finalmente, la definición y verificación de la estructura de datos organizada, las cuales se encuentran en el Anexo D: Informe de la estructura de datos organizada, han sido validadas al 100% por la Dra. Layla Hirsh, experta en el tema de proteínas repetidas; y el Dr. Nicolás Palopoli, experto en los temas relacionados a las proteínas. Esta validación se puede apreciar ingresando a la siguiente dirección: *https://www.drive.google.com/R1/acta_validacion.pdf*.

5.2.2. Conjunto de datos de proteínas repetidas que servirán como datos de entrada para el análisis de diversidad conformacional

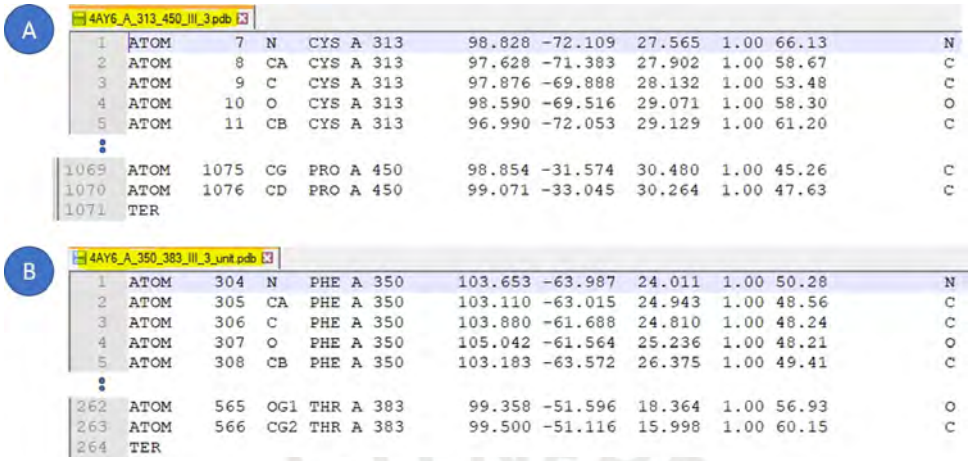
El conjunto de datos de proteínas repetidas está compuesto por archivos en formato pdb (pdb files) de la región repetida y de las unidades de repetición de esta clase de proteínas. Ejemplo de estos archivos generados se puede apreciar en la Figura 5.2. En adición, este conjunto será utilizado como dato de entrada para el análisis de diversidad conformacional. Asimismo, el conjunto de datos de proteínas repetidas se generó en base a la estructura de datos organizada previamente definida en la sección anterior. Además, para el presente conjunto de datos se elaboró un script usando python, las bases de datos RepeatsDB y RCSB PDB, y la librería biopython.

Por otro lado, el script elaborado está basado en una secuencia de pasos manuales que permiten generar los archivos correspondientes de una proteína repetida y esta secuencia se puede apreciar en la Figura 5.3. Asimismo, este script se puede encontrar a través de la siguiente dirección: *https://www.drive.google.com/file/generar_dataset_PRs.py*.

Por otra parte, para verificar que cada archivo del conjunto de datos de proteínas repetidas se generó de manera correcta se utilizó el software PyMOL, el cual permite ver la estructura de una proteína como imagen 3D, logrando validar así que el conjunto de datos es correcto. Esta verifcación se puede encontrar con más detalle en el Anexo E.3: Reporte del conjunto de datos de proteínas repetidas, en la sección Verificación del conjunto de datos.

Figure 5.2

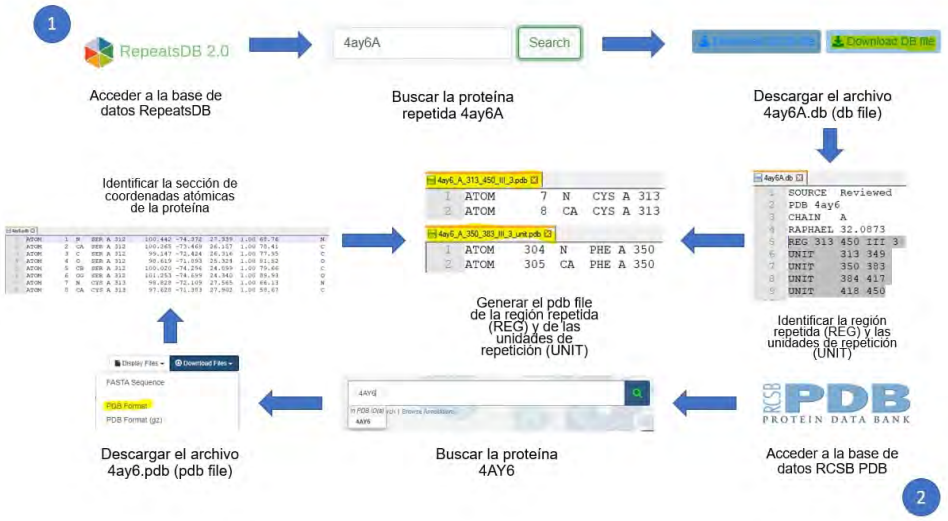
PDB File de la región repetida y de la unidad de repetición de la proteína 4AY6



Nota: A: El pdb file 4ay6_A_313_450_III_3.pdb representa la estructura de la región repetida de la proteína 4AY6 basado en los registros ATOM y el registro TER. B: El pdb file 4ay6_A_313_349_III_3_unit.pdb representa la estructura de la unidad de repetición de la proteína 4AY6 basado en los registros ATOM y el registro TER. Elaboración Propia.

Figure 5.3

Pasos para generar los pdb files de la región repetida y unidades de repetición



Nota: El gráfico muestra los pasos que se siguieron para generar el pdb file de la región repetida y los pdb files de las unidades de repetición de la proteína 4ay6A. Elaboración Propia.

Finalmente, la generación del conjunto de datos de proteínas repetidas y la verificación de este conjunto se pueden encontrar con mayor detalle en el Anexo E: Reporte del conjunto de datos de proteínas repetidas; y ha sido validado al 100% por la Dra. Layla Hirsh, experta en el tema de proteínas repetidas; y el Dr. Nicolás Palopoli, experto en los temas relaciona-

dos a las proteínas. Esta validación se puede apreciar ingresando a la siguiente dirección: *https://www.drive.google.com/R2/acta_validacion.pdf*.

5.2.3. Conjunto de datos de prueba de proteínas repetidas para evaluar la efectividad del método que analizará la diversidad conformacional de las proteínas repetidas

El conjunto de datos de prueba de proteínas repetidas es una lista de archivos en formato pdb (pdb files) que servirán para evaluar la efectividad de los 4 métodos (2 métodos genéricos y 2 propuestas de métodos) que analizarán la diversidad conformacional de las proteínas repetidas.

Además, este conjunto de datos de prueba de proteínas repetidas se generó aleatoriamente a partir del conjunto de datos elaborados previamente descritos en la sección 5.2.2 utilizando un script de elaboración propia. Asimismo, este script se puede encontrar a través de la siguiente dirección: *https://www.drive.google.com/file/generar_test_dataset_PRs.py*. En adición, cabe decir que este conjunto de datos está compuesto por el 30% de las proteínas repetidas.

Por otro lado, la manera en como se generó este conjunto de datos de prueba se puede apreciar en la Figura 5.4. Además, la descripción más a detalle de este resultado esperado se puede encontrar en el Anexo F.2: Reporte del conjunto de datos de prueba de proteínas repetidas, en la sección Generación del conjunto de datos de prueba.

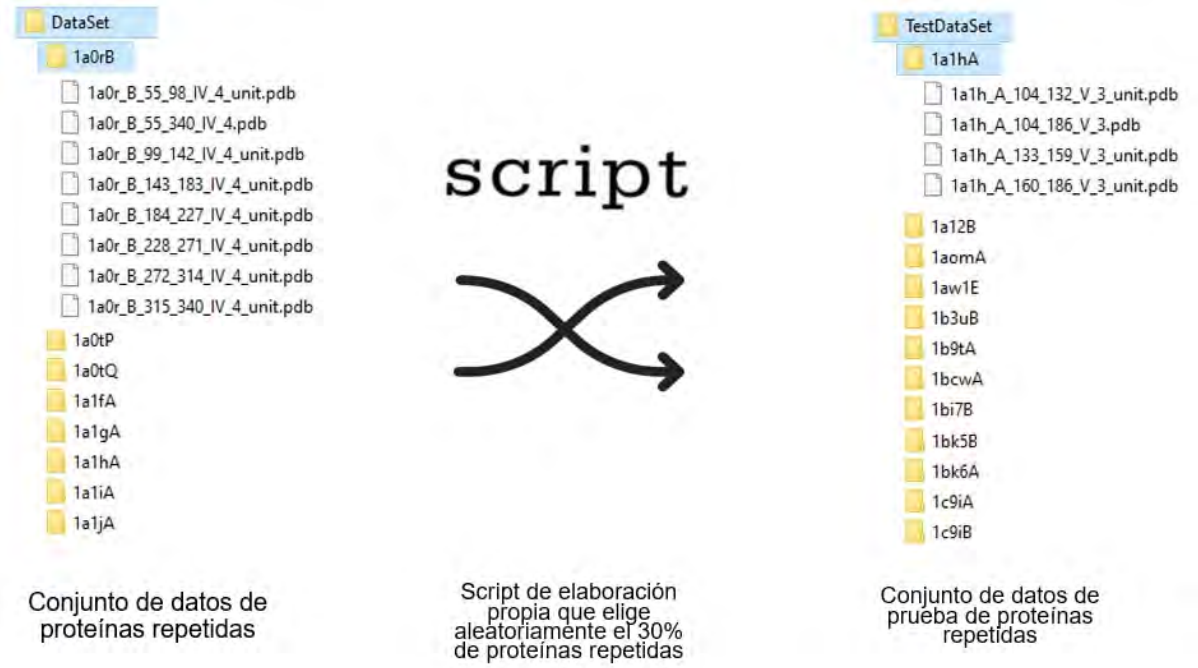
Por otra parte, para verificar que cada archivo del conjunto de datos de prueba de proteínas repetidas se generó de manera correcta se usó el software PyMOL, herramienta que permite visualizar la imagen 3D de una proteína, pudiendo validar así que el conjunto de datos de prueba está generado correctamente. Esta verificación se puede encontrar con más detalle en el Anexo F.3: Reporte del conjunto de datos de prueba de proteínas repetidas, en la sección Verificación del conjunto de datos de prueba.

Finalmente, la generación del conjunto de datos de prueba de proteínas repetidas y la verificación de este conjunto de datos se pueden encontrar con mayor detalle en el Anexo F: Reporte del conjunto de datos de prueba de proteínas repetidas; y ha sido validado al 100% por la Dra. Layla Hirsh, experta en el tema de proteínas repetidas; y el Dr. Nicolás Palopoli, experto en los temas relacionados a las proteínas. Esta validación se puede apreciar ingresando a la siguiente

dirección: *https://www.drive.google.com/R3/acta_validacion.pdf* .

Figure 5.4

Pasos para generar el conjunto de datos de prueba de proteínas repetidas



Nota: El gráfico muestra los pasos que se siguieron para generar el conjunto de datos de prueba de proteínas repetidas. Elaboración Propia.

5.3. Discusión

Se ha definido esta estructura de datos organizada para representar las coordenadas atómicas de las cadenas que posee una proteína cualquiera, ya que este conjunto de coordenadas nos permite obtener su estructura. Además, teniendo como base esta estructura de datos, se puede generar tanto el conjunto de datos de proteínas repetidas que servirán como dato de entrada para analizar la diversidad conformacional de las proteínas repetidas; como el conjunto de datos de prueba de proteínas repetidas que permitirá verificar la efectivad de los 5 métodos (2 métodos genéricos y 3 propuestas de métodos) que analizarán la diversidad conformacional.

Debemos considerar que definir una estructura de datos es necesario para toda investigación que esté relacionado al estudio de la estructura de la proteína, ya que esta será la base que representará a los conjuntos de datos que se vayan a utilizar. De la misma manera, la generación de los conjuntos de datos tienen que cumplir parámetros específicos dependiendo de lo que se

quiera investigar.

Por otra parte, en caso se requiera hacer un análisis de diversidad conformacional en otro tipo de proteína, bastaría con utilizar la misma estructura de datos descrita anteriormente. Asimismo, tanto el conjunto de datos como el conjunto de datos de prueba de proteínas repetidas estaría basado sobre esta estructura; sin embargo, para estos conjuntos no se tienen que considerar las estructuras de la región repetida y las unidades de repetición debido a que es una característica particular de las protéinas repetidas. Por ello, solo se debe considerar la información estructural que la base de datos RCSB PDB proporciona.

Finalmente, se limitó a identificar las secciones fundamentales para describir a la proteína repetida, ya que es información necesaria para realizar el análisis de diversidad conformacional. Por consiguiente, las secciones que brindan información adicional de las proteínas son prescindibles, ya que parte del objetivo del presente proyecto de tesis es analizar los cambios conformacionales que pueden presentar las regiones repetidas y las unidades de repetición de las proteínas repetidas.

Capítulo 6

Propuesta de método específico para proteínas repetidas que permita analizar la diversidad conformacional

6.1. Introducción

En el presente capítulo se presenta el desarrollo del objetivo específico 2, el cual tiene como propósito elaborar una propuesta de método específico para proteínas repetidas que permita analizar su diversidad conformacional. Para conseguir este objetivo, primero se tuvo que obtener los resultados de dos métodos existentes (métodos genéricos) que analizan la diversidad conformacional sobre el conjunto de datos de prueba de proteínas repetidas con la finalidad de comparar estos resultados con los resultados registrados en la base de datos CoDNaS. Luego, se procedió a plantear tres propuestas de métodos a aplicar para analizar la diversidad conformacional de las proteínas repetidas. Después, se procedió a comparar los resultados de estas tres propuestas de métodos elaborados, que están basados en los métodos genéricos, con los resultados obtenidos de los métodos genéricos con la finalidad de recaudar información que permitiera a los especialistas analizar y validar cual de estas tres propuestas es la más adecuada para usar específicamente en las proteínas repetidas. Y finalmente, aplicar el método seleccionado en todo el conjunto de datos de las proteínas repetidas.

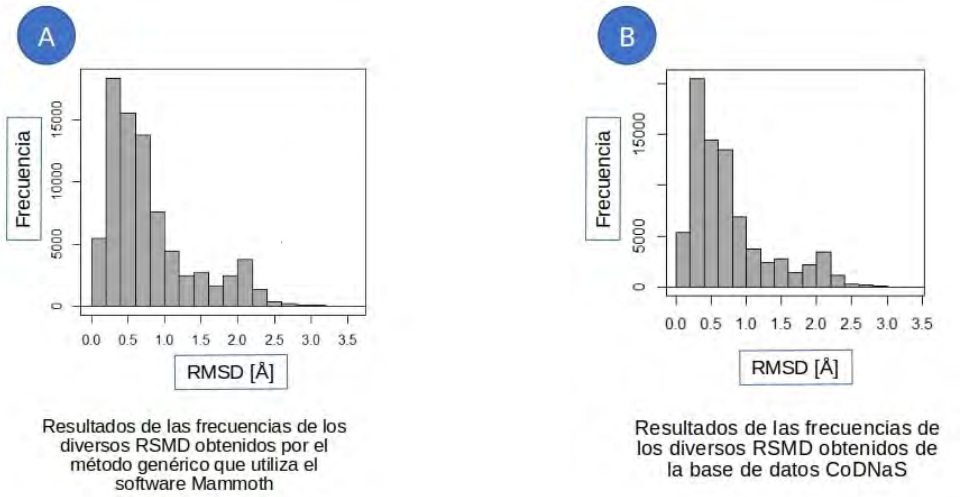
6.2. Resultados Alcanzados

6.2.1. Comparación de resultados obtenidos de dos métodos existentes (métodos genéricos) para analizar la diversidad conformacional sobre el conjunto de datos de prueba de proteínas repetidas con los resultados de la base de datos CoDNaS

Los resultados obtenidos de los dos métodos genéricos son resultados que se utilizarán para la comparación con los resultados conseguidos a través de las tres propuestas de métodos. Asimismo, se generaron dos histogramas, usando la herramienta RStudio, para cada método existente (Ver Figura 6.1 y Figura 6.2) que serán explicadas en el párrafo siguiente; y se generó un script para calcular la similitud (Ver Figura 6.3) entre los resultados calculados y los resultados que la base de datos CoDNaS[1] proporciona. Además, este script se puede ver más a detalle en la siguiente dirección: *https://drive.google.com/file/calcular_similitud.py* y en el Anexo G: Reporte de resultados de los dos métodos existentes.

Figure 6.1

Histogramas de los resultados de las frecuencias de los diversos valores de RMSD usando el software Mammoth y los extraídos de la base de datos CoDNaS

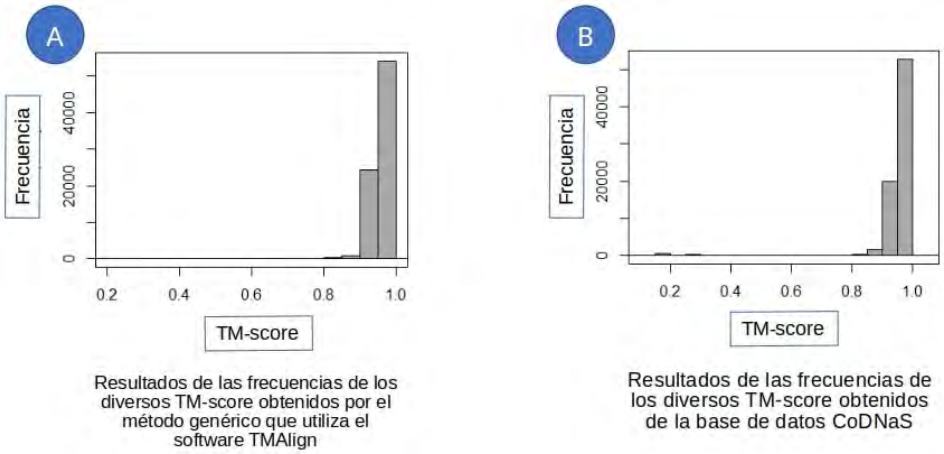


Nota: A: Histograma de los resultados de las frecuencias de los diversos valores de RMSD obtenidos usando el software Mammoth. B: Histograma de los resultados de las frecuencias de los diversos valores de RMSD extraídos de la base de datos CoDNaS. Elaboración propia.

[1]CoDNaS es una base de datos que contiene la información de diversidad conformacional de las proteínas en su estado nativo. Se puede acceder a esta base de datos a través de la siguiente dirección: http://ufq.unq.edu.ar/codnas/

Figure 6.2

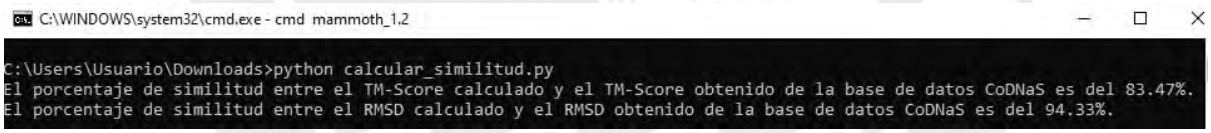
Histogramas de los resultados de las frecuencias de los diversos valores de TM-score usando el software TMAlign y los extraídos de la base de datos CoDNaS



Nota: A: Histograma de los resultados de las frecuencias de los diversos valores de TM-score obtenidos usando el software TMAlign. B: Histograma de los resultados de las frecuencias de los diversos valores de TM-score extraídos de la base de datos CoDNaS. Elaboración propia.

Figure 6.3

Similitud de los cálculos obtenidos con los resultados que proporciona CoDNaS



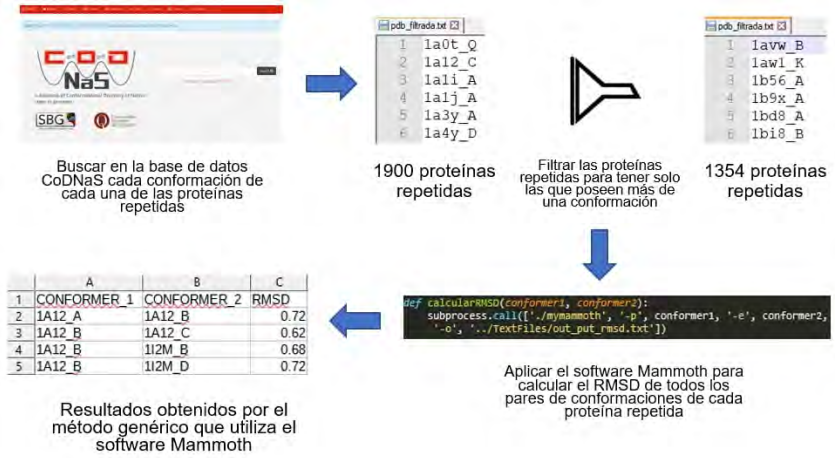
Nota: La presente figura muestra el cálculo de similitud de los dos métodos existentes con respecto a los resultados extraídos de la base de datos CoDNaS. Elaboración propia.

Los dos métodos existentes mencionados en el párrafo anterior vienen a ser procedimientos que conllevan a la obtención de las diferentes conformaciones de una proteína y en base a estas conformaciones superponer todos los pares de estructuras posibles de formar con la finalidad de medir la diferencia estructural que existe entre ambas conformaciones. Sin embargo, lo que diferencia al uno del otro, es el software que utiliza para poder calcular este grado de diversidad conformacional, el cual va a permitir a los científicos poder analizar la diversidad conformacional. Es así que uno utiliza el software Mammoth para calcular el RMSD y el otro, el software TMAlign para calcular el TM-score. Asimismo, la descripción más a detalle de estos dos métodos genéricos se puede encontrar en el Anexo G.2: Reporte de resultados de los dos métodos existentes, en la sección Métodos genéricos.

Por otra parte, antes de realizar la comparación con los resultados que se pueden obtener de la base de datos CoDNaS, se tuvieron que calcular los resultados correspondientes a cada método genérico. Es así que la manera en cómo se generaron estos resultados se pueden apreciar en la Figura 6.4 y en la Figura 6.5, respectivamente. Asimismo, la descripción a mayor detalle se encuentra en el Anexo G.3: Reporte de resultados de los dos métodos existentes, en la sección Generación de resultados.

Figure 6.4

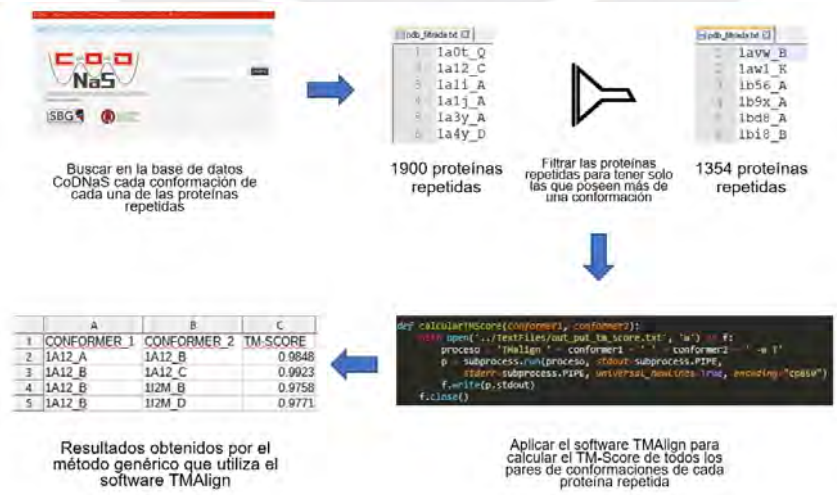
Pasos para obtener los resultados a través del método genérico que usa Mammoth



Nota: El gráfico muestra los pasos que se siguen para obtener los resultados a través del método genérico que utiliza el software Mammoth. Elaboración propia.

Figure 6.5

Pasos para obtener los resultados a través del método genérico que usa TMAlign



Nota: El gráfico muestra los pasos que se siguen para obtener los resultados a través del método genérico que utiliza el software TMAlign. Elaboración propia.

Finalmente, la comparación de los resultados obtenidos de los dos métodos genéricos con los resultados de la base de datos CoDNaS ha sido validada por la Dra. Layla Hirsh, experta en los temas relacionados con las proteínas repetidas; el Dr. Gustavo Parisi y el Dr. Nicolás Palopoli, expertos en los temas relacionados a la diversidad conformacional de las proteínas. Asimismo, esta validación se puede apreciar ingresando a la siguiente dirección: *https://www.drive.google.com/R4/acta_validacion.pdf*.

6.2.2. Tres propuestas de métodos a aplicar para analizar la diversidad conformacional de las proteínas repetidas

Las tres propuestas planteadas son propuestas basadas en métodos genéricos y en hipótesis que fueron trazadas con el asesoramiento de la Dra. Layla Hirsh, experta en el área de las proteínas repetidas y el Dr. Nicolás Palopoli, experto en el área de la diversidad conformacional en proteínas. Asimismo, estas propuestas están pensadas en permitir el análisis de diversidad conformacional en las proteínas repetidas; y a continuación, se explica una breve descripción de cada una.

Propuesta N° 1: Región de repetición

Toda proteína repetida tiene como mínimo una región de repetición formada por sus diferentes unidades de repetición. Teniendo en cuenta esto, la propuesta N° 1 consiste en determinar las diferentes conformaciones de cada una de las regiones de estas proteínas repetidas con la finalidad de generar un conjunto de confórmeros que incluya únicamente la región de repetición. Es así que, luego de tener estos nuevos conjuntos de estructuras alternativas para cada proteína repetida, se procede a generar el mayor número posible de pares de confórmeros para cada conjunto con la finalidad de superponerlos y calcular la diferencia estructural en cada uno de ellos a través de la medida estadística RMSD[2] usando el software Mammoth para obtener el grado de diversidad conformacional.

Propuesta N° 2: Unidades de repetición como confórmeros

Cada proteína repetida cuenta con una serie de unidades de repetición dentro de la región de repetición, es por ello que cada unidad se podría considerar como un confórmero de sí

[2]La desviación cuadrática media raiz (RMSD) es la unidad estadística que calcula la diferencia entre un par de estructuras.

mismo. De esta manera si una región de repetición posse 5 unidades consideraríamos para nuestro caso que el conjunto de confórmeros estaría formado por estas 5 unidades. Luego de identificar los nuevos conjuntos de confórmeros de cada proteína repetida, se procede a generar la mayor cantidad de pares posibles de confórmeros para cada conjunto con la finalidad de superponer las estructuras y calcular la diferencia estructural en cada uno de ellos a través de la medida estadística RMSD usando el software Mammoth para obtener el grado de diversidad conformacional.

Propuesta N° 3: Unidades de repetición de los confórmeros

Como se mencionó anteriormente, una cadena de proteína repetida puede tener diversas regiones de repetición y cada región de repetición puede tener diversos confórmeros, pero dentro de cada región de repetición tenemos un número diverso de unidades repetidas. Conociendo esto, se propone para esta opción formar el nuevo conjunto de confórmeros en base a todas las unidades de repetición de todos los confórmeros de una proteína repetida. Luego de formar los nuevos conjuntos de confórmeros por cada proteína repetida, se procede a formar el mayor número posibles de pares de confórmeros para superponer las estructuras de cada par y calcular la diferencia estructural a través de la medida estadística RMSD usando el software Mammoth con la finalidad de obtener el grado de diversidad conformacional.

Finalmente, para mayor detalle de estas propuestas se puede encontrar en el Anexo H: Reporte de propuestas de los métodos a aplicar y la validación de las tres propuestas de métodos a aplicar para analizar la diversidad conformacional de las proteínas repetidas han sido validadas por la Dra. Layla Hirsh, experta en los temas relacionados con las proteínas repetidas; el Dr. Gustavo Parisi y el Dr. Nicolás Palopoli, expertos en los temas relacionados a la diversidad conformacional de las proteínas. Asimismo, esta validación se puede apreciar ingresando a la siguiente dirección: *https://www.drive.google.com/R5/acta_validacion.pdf*.

6.2.3. Resultados obtenidos de los métodos elaborados sobre el conjunto de datos de prueba de las proteínas repetidas y la comparación con los resultados de los métodos genéricos

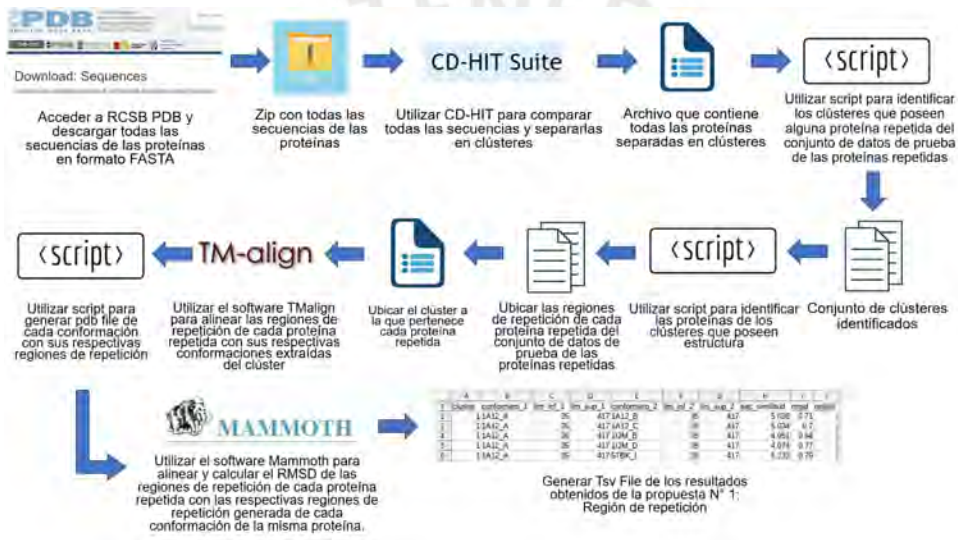
Los resultados obtenidos de las tres propuestas de métodos son resultados que se van a comparar con los resultados conseguidos de los dos métodos genéricos. Esta comparación va a

permitir a los expertos, en base a su conocimiento, poder seleccionar el método más apropiado que permita el análisis de diversidad conformacional de las proteínas repetidas.

Para el caso de la propuesta N° 1: Región de repetición, la manera en cómo se obtuvo los resultados se puede apreciar en la Figura 6.6, y la descripción más a detalle se puede encontrar en el Anexo I: Reporte de resultados de las propuestas de métodos, en la sección Generación de resultados.

Figure 6.6

Pasos para obtener los resultados a través de la propuesta N° 1: Región de repetición



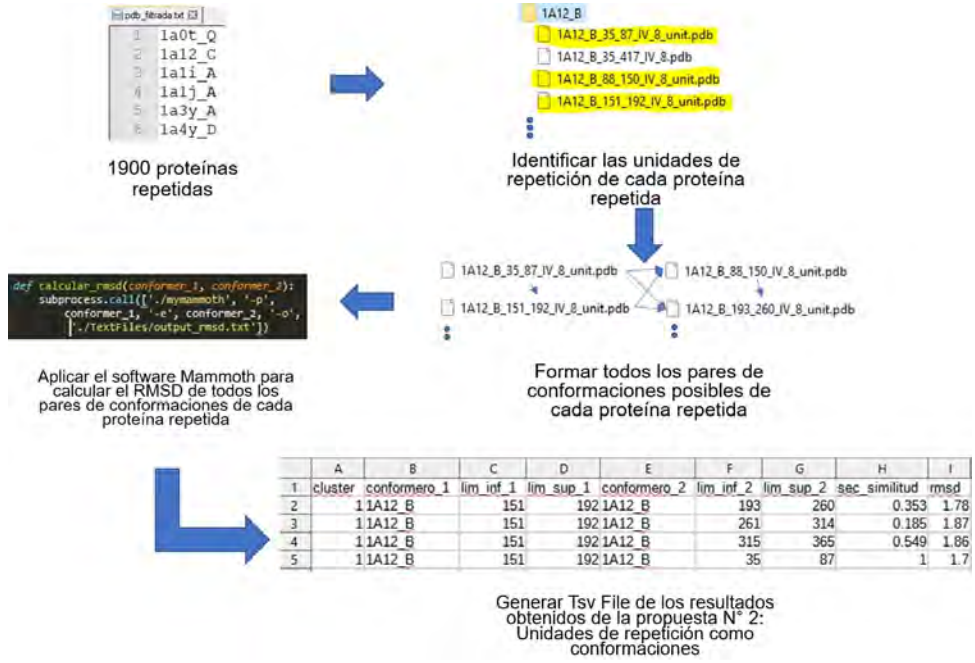
Nota: La presente figura muestra los pasos a seguir para obtener los resultados a través de la propuesta de método N° 1: Región de repetición. Elaboración propia.

Asimismo, en la Figura 6.7 se presenta los pasos que se realizaron para hallar los resultados respectivos a la propuesta N° 2: Unidades de repetición como confórmeros; y para mayor detalle de estos resultados se puede encontrar en el Anexo I: Reporte de resultados de las propuestas de métodos, en la sección Generación de resultados.

De la misma manera, se puede apreciar en la Figura 6.8 las actividades que se realizaron para poder obtener los resultados referentes a la propuesta N° 3: Unidades de repetición de los confórmeros; y la descripción más detallada se puede encontrar en el Anexo I: Reporte de resultados de las propuestas de métodos, en la sección Generación de resultados.

Figure 6.7

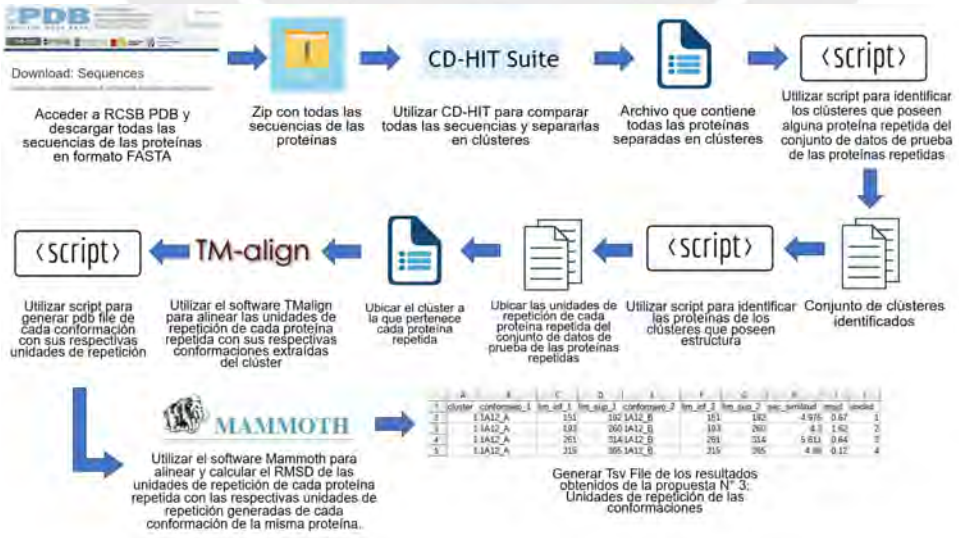
Pasos para obtener los resultados a través de la propuesta N° 2: Unidades de repetición como confórmeros



Nota: La presente figura muestra los pasos a seguir para obtener los resultados a través de la propuesta de método N° 2: Unidades de repetición como confórmeros. Elaboración propia.

Figure 6.8

Pasos para obtener los resultados a través de la propuesta N° 3: Unidades de repetición de los confórmeros

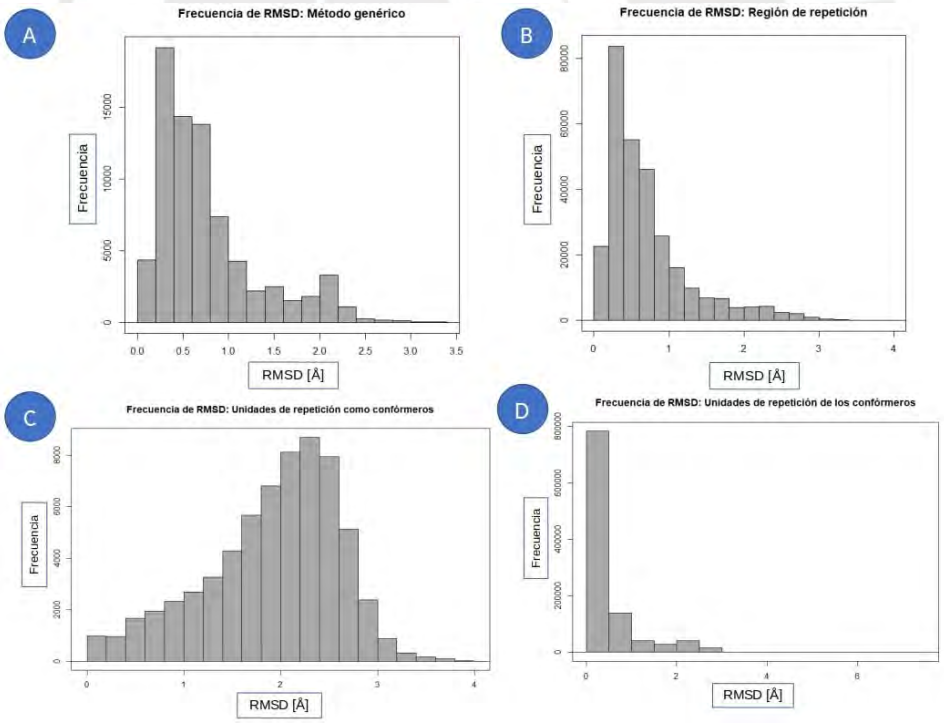


Nota: La presente figura muestra los pasos a seguir para obtener los resultados a través de la propuesta de método N° 3: Unidades de repetición de los confórmeros. Elaboración propia.

Teniendo estos resultados, se utilizó el software RStudio para generar histogramas (Ver Figura 6.9) que muestran las frecuencias de los diversos valores de RMSD obtenidos entre las tres propuestas de métodos y el método genérico que utiliza el software Mammoth para calcular la diferencia estructural a través del RMSD. Asimismo, cabe mencionar que, a través de un juicio crítico de la experta en protéinas repetidas, la Dra. Layla Hirsh, y de los expertos de la diversidad conformacional en proteínas, el Dr. Gustavo Parisi y el Dr. Nicolás Palopoli, se optó por no considerar los resultados de los diversos valores de la medida estadística TM-score, que nos proporciona el metodo genérico que utiliza el software TMAlign, para la comparación entre los resultados que se obtienen empleando los métodos mencionados previamente. Para mayor detalle de esta comparación se puede encontrar en el Anexo I: Reporte de resultados de las propuestas de métodos, en la sección Comparación entre los resultados obtenidos y los resultados calculados de los métodos genéricos.

Figure 6.9

Histogramas del método genérico y de las propuestas de métodos



Nota: A: Histograma de frecuencias RMSD obtenidas de CoDNaS. B: Histograma de frecuencias RMSD obtenidas empleando la propuesta N° 1: Región de repetición. C: Histograma de frecuencias RMSD obtenidas empleando la propuesta N° 2: Unidades de repetición como confórmeros. D: Histograma de frecuencias RMSD obtenidas empleando la propuesta N° 3: Unidades de repetición de los confórmeros. Elaboración propia.

Finalmente, para mayor detalle de los resultados obtenidos a través de las propuestas de métodos se pueden encontrar en el Anexo I: Reporte de resultados de las tres propuestas de métodos y la validación de estos resultados que determinan la estimación de la diversidad conformacional de las proteínas repetidas han sido validadas por la Dra. Layla Hirsh, experta en los temas relacionados con las proteínas repetidas; el Dr. Gustavo Parisi y el Dr. Nicolás Palopoli, expertos en los temas relacionados a la diversidad conformacional de las proteínas. Asimismo, esta validación se puede apreciar ingresando a la siguiente dirección: *https://www.drive.google.com/R6/acta_validacion.pdf* .

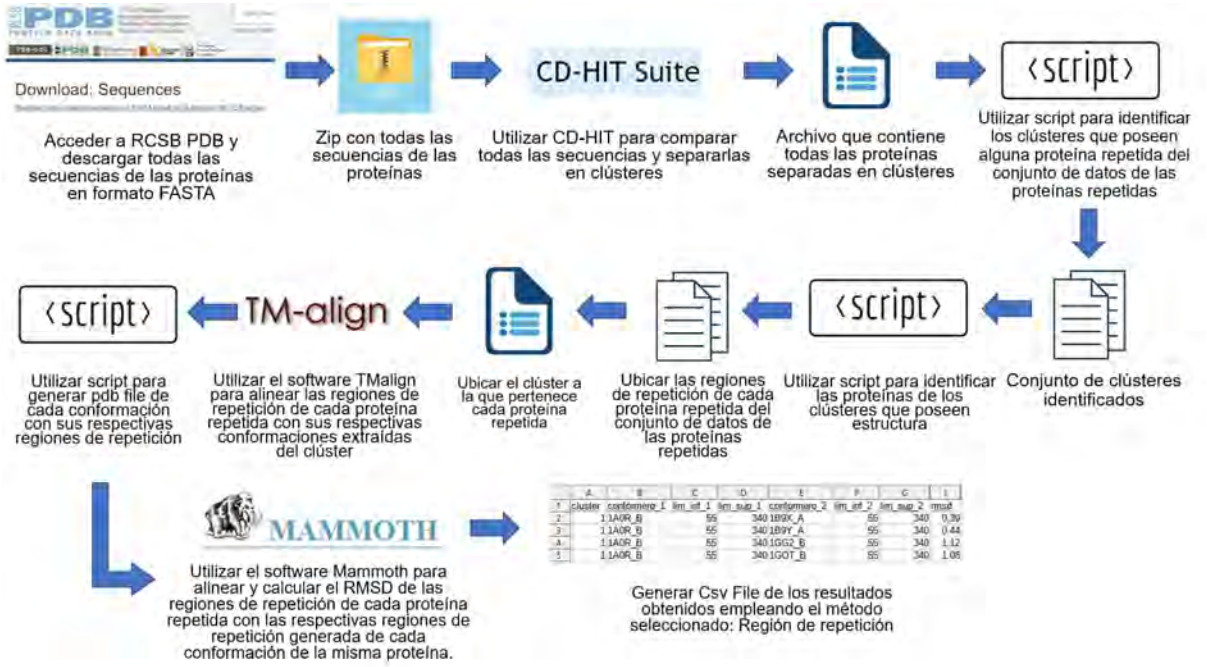
6.2.4. Aplicación del método seleccionado en todo el conjunto de datos de las proteínas repetidas

Se realizaron reuniones con la Dra. Layla Hirh, experta en el tema de las proteínas repetidas, con el Dr. Gustavo Parisi y el Dr. Nicolás Palopoli, ambos expertos en el tema de diversidad conformacional en proteínas, con la finalidad de presentar los resultados de las tres propuestas de métodos mencionadas en la sección anterior con sus respectivas comparaciones. Después del análisis de cada uno de los expertos se seleccionó la propuesta de método N° 1: Región repetida, como método para aplicar en todo el conjunto de datos de las proteínas repetidas, ya que determinar las regiones repetidas en las diversas conformaciones que presenta cada proteína repetida permite poder tomar como nuevo conjunto de confórmeros estas conformaciones con su respectiva región de repetición. Asimismo, con este nuevo conjunto de conformaciones se podrá analizar la diversidad conformacional con resultados prometedores, porque se tendría una aportación del análisis del dominio repetitvo, el cual es característica particular de las proteínas repetidas .

Es así que conociendo el método seleccionado, el cual permitirá el análisis de diversidad conformacional en estructuras de proteínas repetidas se procedió a aplicarlo en el conjunto de datos de esta clase de proteínas. Los pasos que se realizaron para la obtención de los resultados basados en la medida estadística RMSD se pueden apreciar en la Figura 6.10 , y la descripción con mayor detalle se puede encontrar en el Anexo J.2: Reporte de resultados del método seleccionado, en la sección Generación de Resultados.

Figure 6.10

Pasos para obtener los resultados a través del método seleccionado sobre el conjunto de datos de proteínas repetidas



Nota: La presente figura muestra los pasos a seguir para obtener los resultados a través del método seleccionado (Región de repetición). Elaboración propia.

Por último, para mayor detalle de los resultados obtenidos a través del método seleccionado se pueden encontrar en el Anexo J: Reporte de resultados del método seleccionado y la validación de estos resultados que determinan la estimación de la diversidad conformacional de las proteínas repetidas han sido validadas, basado en su juicio experto, por la Dra. Layla Hirsh, experta en los temas relacionados con las proteínas repetidas; el Dr. Gustavo Parisi y el Dr. Nicolás Palopoli, expertos en los temas relacionados a la diversidad conformacional de las proteínas. Asimismo, esta validación se puede apreciar ingresando a la siguiente dirección: *https://www.drive.google.com/R7/acta_validacion.pdf*.

6.3. Discusión

Se generaron los resultados de grados de diversidad conformacional de las proteínas repetidas empleando los métodos genéricos para compararlos con los resultados que se iban a obtener a través de las tres propuestas de métodos. Asimismo, con la finalidad de comprobar que estos resultados, tanto el RMSD como el TM-score a través del software Mammoth y del software

TMAlign, respectivamente; correspondian a la información registrada en la base de datos CoDNaS. Por tal motivo, se calculó la similitud que existía entre ambos resultados; sin embargo, no hubo un 100% de similitud. Y esto se debe a que la base de datos CoDNaS no ha sido actualizada desde abril del 2017, por lo que todavía no se han agregado a esta base de datos información que se han recolectado a través de las investigaciones sobre proteínas en los últimos años.

Además, haber realizado esta comprobación permite validar el entendimiento de los dos métodos existentes que evalúan la diversidad conformacional de las proteínas repetidas y el uso adecuado de las herramientas Mammoth y TMAlign para poder calcular el RMSD y TM-score, respectivamente. Asimismo, teniendo como base esto, se permitió tener un mejor enfoque al momento de plantear las tres propuestas de métodos que analizarán la diversidad conformacional en las proteínas repetidas.

Por otro lado, se limitó a utilizar dos métodos genéricos que calculan el TM-score y el RMSD, dado que estas son las unidades estadísticas que mayor información pueden proveer al científico para poder entender un poco más la naturaleza de las proteínas repetidas. Sin embargo, para el caso de las propuestas de métodos, estas consideran medir el grado de diversidad conformacional por medio de la unidad estadística RMSD, ya que es la medida más utilizada para calcular la diferencia estructural entre conformaciones.

Por otra parte, la comparación entre los resultados obtenidos empleando las tres distintas propuestas de métodos con los resultados obtenidos por medio de los métodos genéricos se limitó a comparaciones por medio de histogramas de frecuencia de RMSD, ya que esta puede dar una perspectiva amplia a los expertos en el tema de proteínas repetidas y diversidad conformacional. Además, esta comparación solo abarcó a los métodos que miden el grado de diversidad conformacional por medio del RMSD, es decir que la comparación solo abarcó a las tres propuestas de métodos y al método genérico que utiliza el software Mammoth para calcular el RMSD, ya que la medida estadística TM-score mide la similaridad estructural y no la diferencia estructural como lo hace la medida estadística RMSD.

Finalmente, basado en el juicio de los expertos se seleccionó la propuesta de método N° 1: Región de repetición, ya que el nuevo conjunto de confórmeros, que se forma en este método, puede ser tomado como conformaciones, debido a que existe una mayor diversidad de valores

de RMSD obtenidos y estas no se encuentran sesgadas hacia valores muy cercanos o muy lejanos de 0. Además, tomar cada unidad de repetición como conformación, teniendo en cuenta lo reflejado en la Figura 6.9, se puede notar que estas conformaciones tendrían un mayor impacto en el estudio de la divergencia estructural en proteínas repetidas, la cual intenta evaluar la diferencia estructural de una misma proteína presente en distintos organismos como resultado de la evolución; y esto es un enfoque diferente a lo que la presente tesis quiere desarrollar, el cual es la diversidad conformacional de proteínas repetidas.

Capítulo 7

Herramienta para el análisis de diversidad conformacional de las proteínas repetidas

7.1. Introducción

En el presente capítulo se muestra el desarrollo del objetivo específico 3, el cual tiene como propósito desarrollar una herramienta de acceso libre a la comunidad científica donde los usuarios puedan evaluar y visualizar la diversidad conformacional de las diferentes proteínas repetidas. Para lograr este objetivo, primero se tuvo que elaborar el modelamiento de la estructura de la base de datos, la cual va a contener la información de diversidad conformacional de las proteínas repetidas. Posteriormente, crear un servicio web que permita evaluar la diversidad conformacional de las proteínas repetidas o extraer la información de diversidad conformacional de esta clase de protéinas de la base de datos. Y finalmente, desarrollar una interfaz de usuario que permita evaluar y visualizar esta información de diversidad conformacional de las proteínas repetidas utilizando el servicio web.

7.2. Resultados Alcanzados

7.2.1. Modelamiento de la estructura de base de datos que contiene la información de diversidad conformacional de las proteínas repetidas

El modelamiento de la estructura de la base de datos consiste en definir el modelo relacional de la misma. Esta base de datos se va a llamar CoDNaS-PRs y va a contener los datos obtenidos luego del análisis de diversidad conformacional de las proteínas repetidas. Estos datos vienen a ser la información general, la información estructural y las diversas conformaciones de esta clase de proteínas que tienen la característica particular de tener repeticiones en su estructura.

Por otro lado, antes de elaborar el modelo relacional se tuvo que definir tres tablas con respecto a los datos que se van a obtener luego de este análisis de diversidad conformacional de la proteínas repetidas. Estas tablas se pueden apreciar en la Figura 7.1.

Figure 7.1

Definición de las tablas a utilizar en el modelo relacional

A

info_estructural	
Columna	**Tipo de dato**
cluster	int
region	int
num_conformaciones	int
rmsd_min	double
rmsd_max	double
rmsd_avg	double

B

info_general	
Columna	**Tipo de dato**
pdb_id	varchar(10)
cluster	int
nombre_proteina	varchar(300)
titulo	varchar(500)
organismo	varchar(300)
long_secuencia	int
clasificacion	varchar(300)
num_regiones	int

C

conformación	
Columna	**Tipo de dato**
conformero_1	varchar(10)
conformero_2	varchar(10)
lim_inf_1	int
lim_sup_1	int
lim_inf_2	int
lim_sup_2	int
sec_similitud	double
rmsd	double
region	int

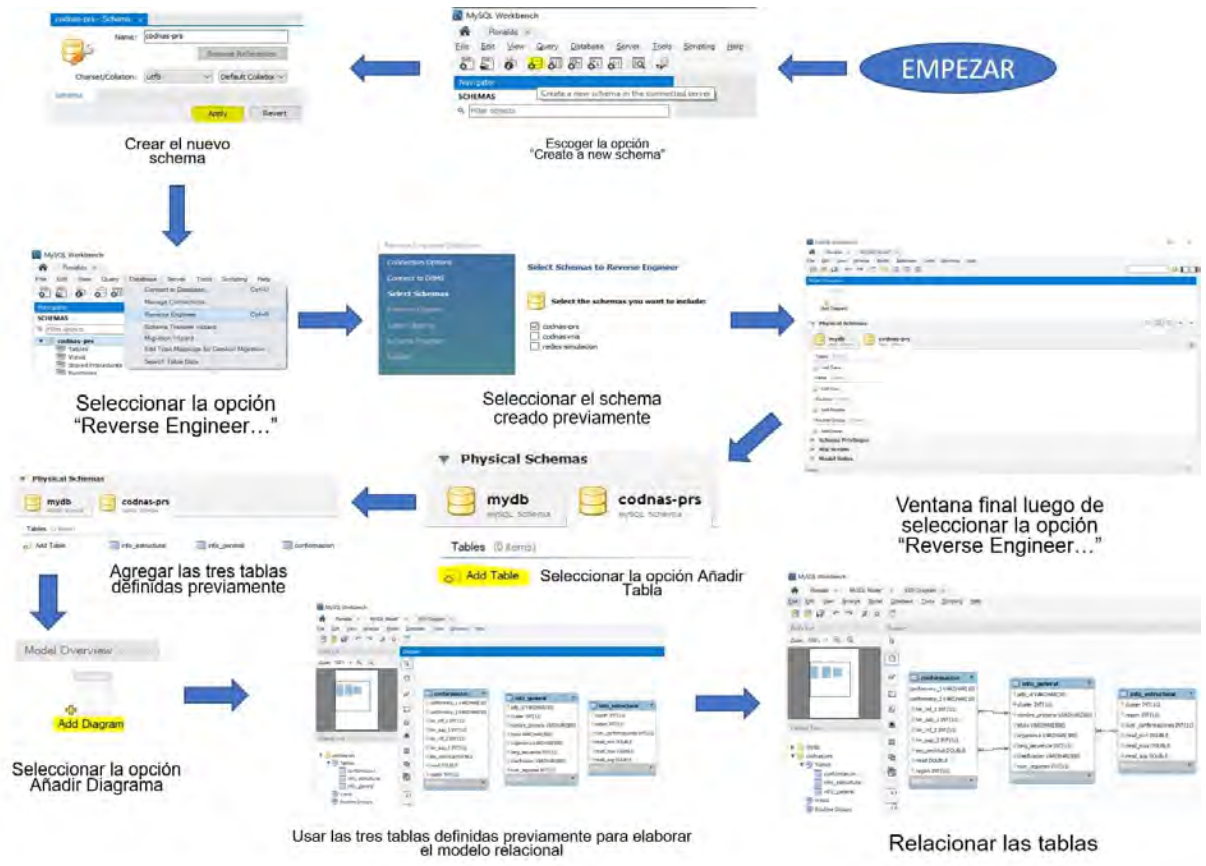
Nota: A: La tabla info_general contiene los datos de la información general de la proteína repetida. B: La tabla info_estructural contiene los datos de la información estructural de la proteína repetida. C: La tabla conformación contiene los datos de las diversas conformaciones de la proteína repetida. Elaboración propia.

Teniendo en cuenta estas tablas definidas, el modelo relacional de la base de datos se modeló utilizando la herramienta MySQL Workbench y la manera en cómo se hizo esto se puede apreciar en la Figura 7.2, la cual detalla brevemente los pasos para la definición de la misma. Asimismo, la descripción más a detalle de este resultado esperado se puede encontrar en el

Anexo K.3: Documento del modelamiento de la estructura de base de datos, en la sección Elaboración del modelo relacional de la estructura de base de datos.

Figure 7.2

Pasos para elaborar el modelo relacional de la base de datos



Nota: El gráfico muestra los pasos para elaborar el modelo relacional de la base de datos. Elaboración propia.

Además, al tener estructurado el modelo relacional de la base de datos se generó un script con la ayuda de la herramienta MySQL Workbench, ya que este software brinda una opción llamada "Forward Engineer" que permite generar una base de datos a partir de un modelo relacional (Ver Figura 7.4). Una parte de este script se puede apreciar en la Figura 7.3. Asimismo, el script completo se puede encontrar en el Anexo K.5: Documento del modelamiento de la estructura de base de datos, en la sección Script de creación de la base de datos y a través de la siguiente dirección: *https://www.drive.google.com/file/script_create_db.sql*.

Figure 7.3

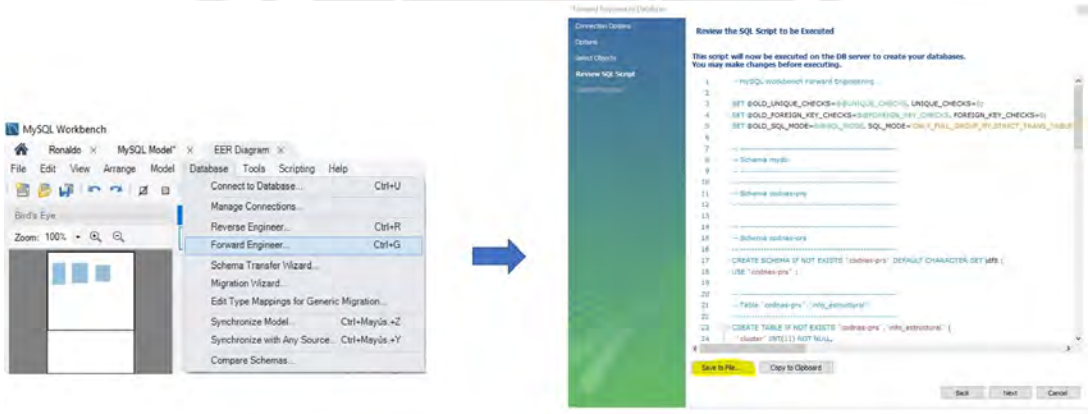
Script de creación de la base de datos

```
-- -----------------------------------------------------
-- Table `codnas-prs`.`info_estructural`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `codnas-prs`.`info_estructural` (
  `cluster` INT(11) NOT NULL,
  `region` INT(11) NOT NULL,
  `num_conformaciones` INT(11) NOT NULL,
  `rmsd_min` DOUBLE NOT NULL,
  `rmsd_max` DOUBLE NOT NULL,
  `rmsd_avg` DOUBLE NOT NULL,
  PRIMARY KEY (`cluster`, `region`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8;
```

Nota: El gráfico muestra una parte del script de creación de la base de datos. Elaboración propia.

Figure 7.4

Script de creación de la base de datos usando "Forward Engineer..."



Nota: El gráfico muestra los pasos para guardar el script de creación de la base de datos. Elaboración propia.

Finalmente, para mayor detalle del modelamiento de la estructura de base de datos que contiene la información de diversidad conformacional de las proteínas repetidas se puede encontrar en el Anexo K: Documento del modelamiento de la estructura de base de datos y la validación de este modelamiento ha sido realizada por la Dra. Layla Hirsh, el Dr. Gustavo Parisi y el Dr. Nicolás Palopoli; expertos con conocimento y participación en proyectos que hacen uso de una base de datos. Asimismo, esta validación se puede apreciar ingresando a la siguiente dirección: *https://www.drive.google.com/R8/acta_validacion.pdf*.

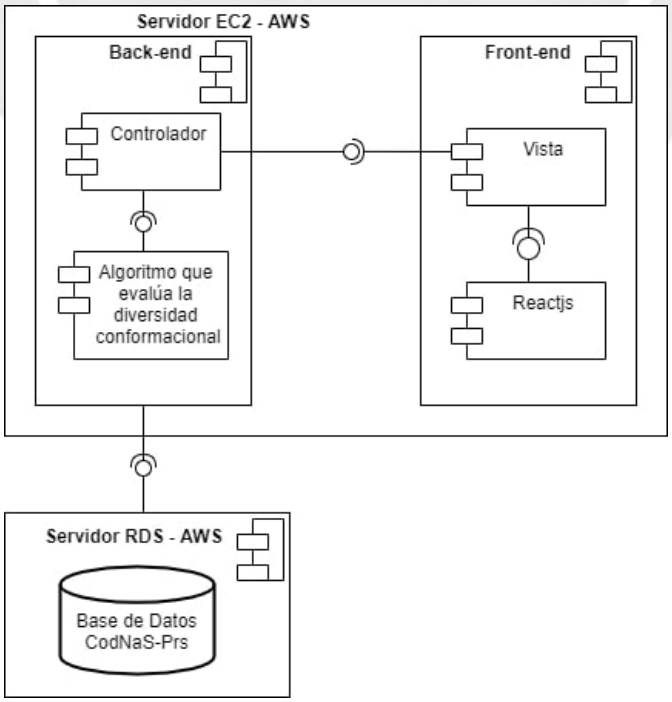
7.2.2. Servicio web que permita evaluar la diversidad conformacional de las proteínas repetidas o extraer la información de diversidad conformacional de esta clase de proteínas de la base de datos

El servicio web es un conjunto de funcionalidades que la interfaz de usuario va a poder utilizar para poder extraer la información del análisis de diversidad conformacional de las proteínas repetidas de la base de datos y en caso que no se encuentre esta información, la interfaz de usuario va a poder evaluar la diversidad conformacional de esta clase de proteínas usando este servicio.

Asimismo, este servicio está alojado en un servidor en la nube de AWS con acceso a la base de datos MySQL, versión 5.7.28, con la finalidad que los científicos puedan utilizarlo. Además, en la Figura 7.5 se puede apreciar la arquitectura basada en el diagrama de componentes, la cual representa la interacción que existe entre la interfaz de usuario, el servicio web, la base de datos y el servidor de AWS. Asimismo, la descripción más a detalle se puede encontrar en el Anexo L.2: Documento de arquitectura del servicio web, en la sección Elaboración de la arquitectura.

Figure 7.5

Diagrama de componentes

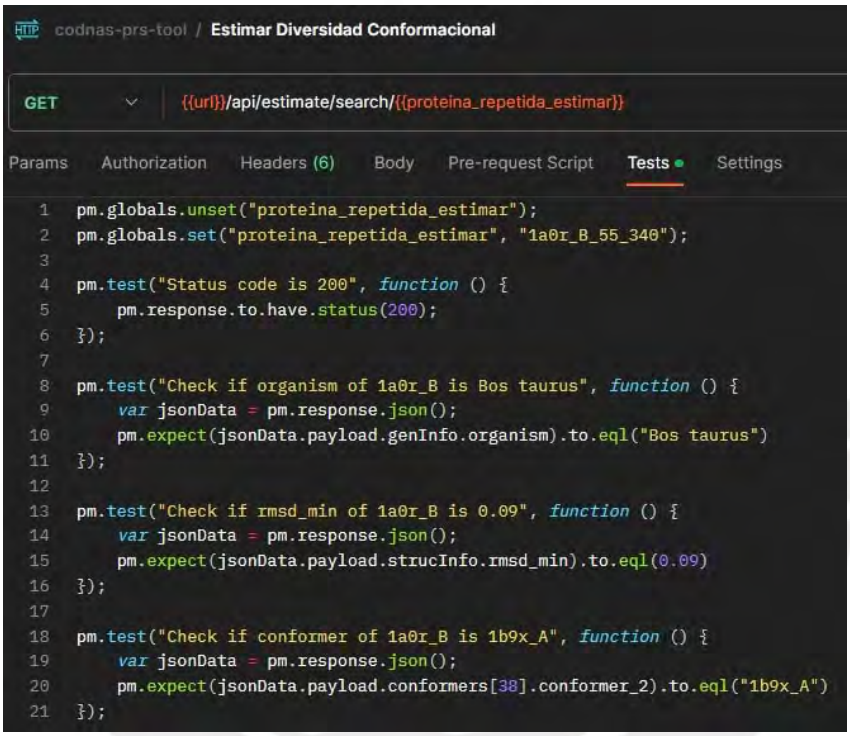


Nota: El gráfico muestra el diagrama de componentes del servicio web. Elaboración Propia.

Por otro lado, el servicio web ha sido elaborado usando el micro framework Flask y se utilizó la herramienta Postman para realizar pruebas del funcionamiento de las funcionalidades que cubre el servicio web. Asimismo, en la Figura 7.6 se puede apreciar un script que se usó para comprobar que el servicio Estimar funciona correctamente y para mayor detalle de las pruebas que se realizaron a las funcionalidades del servicio web se pueden encontrar en el Anexo M: Informe de pruebas funcionales del servicio web.

Figure 7.6

Script para comprobar la funcionalidad del servicio Estimar



Nota: El gráfico muestra el script que se utilizó para validar el correcto funcionamiento del servicio Estimar, el cual estima y da como resultado la información de la diversidad conformacional. Elaboración Propia.

Por otra parte, este servicio web está almacenado en un repositorio de GitHub, la cual se puede apreciar en la Figura 7.7; y se puede acceder al repositorio a través de la siguiente dirección: *https://www.github.com/servicio_web*.

Figure 7.7

Repositorio de github del servicio web

Nota: El gráfico muestra el repositorio de github del servicio web. Elaboración Propia.

Finalmente, el documento de arquitectura del servicio web y las pruebas funcionales aplicadas en el servicio web han sido validadas por la Dra. Layla Hirsh, y validados por el Dr. Gustavo Parisi y Dr. Nicolás Palopoli; expertos con conocimento y participiación en proyectos alojados en la nube AWS. Asimismo, esta validación se puede apreciar ingresando a la siguiente dirección: *https://www.drive.google.com/R9/acta_validacion.pdf*.

7.2.3. Interfaz de usuario que permita evaluar la información de diversidad conformacional de las proteínas repetidas utilizando el servicio web

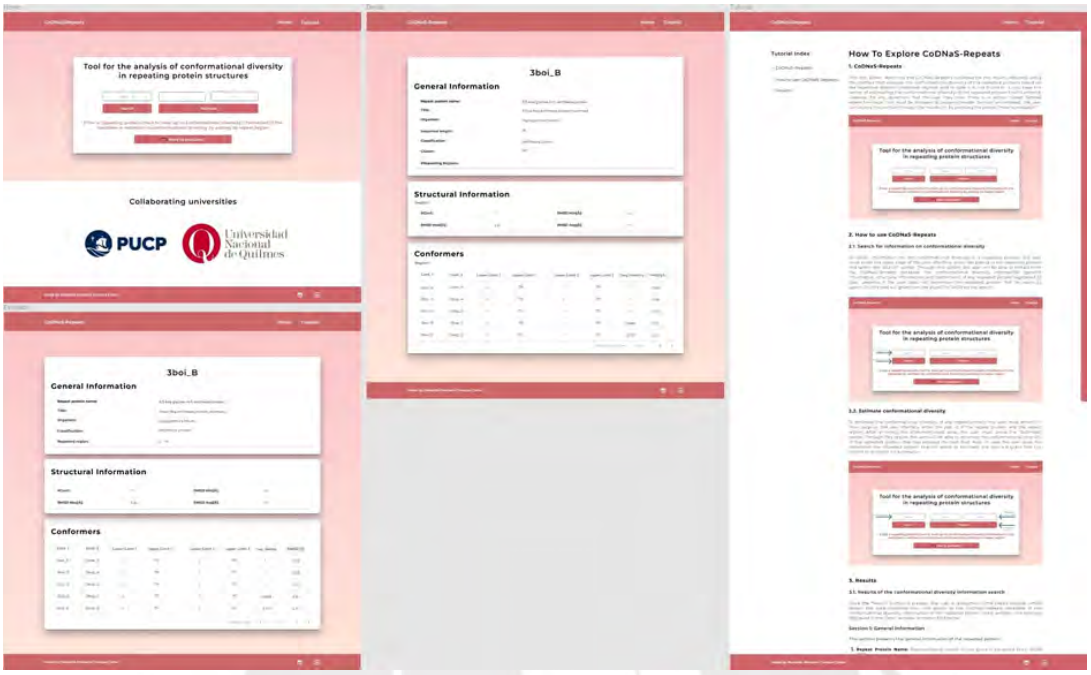
La interfaz de usuario es el medio en el cual los científicos que trabajan con proteínas repetidas van a poder acceder para buscar la información de diversidad conformacional de cualquier proteína repetida que se encuentre en la base de datos CoDNaS-PRs o, en caso no se encuentre esta información en la misma, van a poder evaluar la diversidad conformacional de este tipo de proteínas. Asimismo, la interfaz de usuario está compuesta por cuatro ventanas: Detalle, Estimación, Inicio y Tutorial.

Además, esta interfaz está basada en un prototipo de autoría propia hecho en Figma. Este prototipo se aprecia en la Figura 7.8 y se puede ver la interacción entre las diferentes ventanas a través de la siguiente dirección: *https://www.figma.com/prototype/CoDNaS-Repeats*. Asimismo, la descripción con mayor detalle del prototipo se puede encontrar en el Anexo N:

Informe del prototipo de la interfaz de usuario.

Figure 7.8

Prototipo de la interfaz de usuario



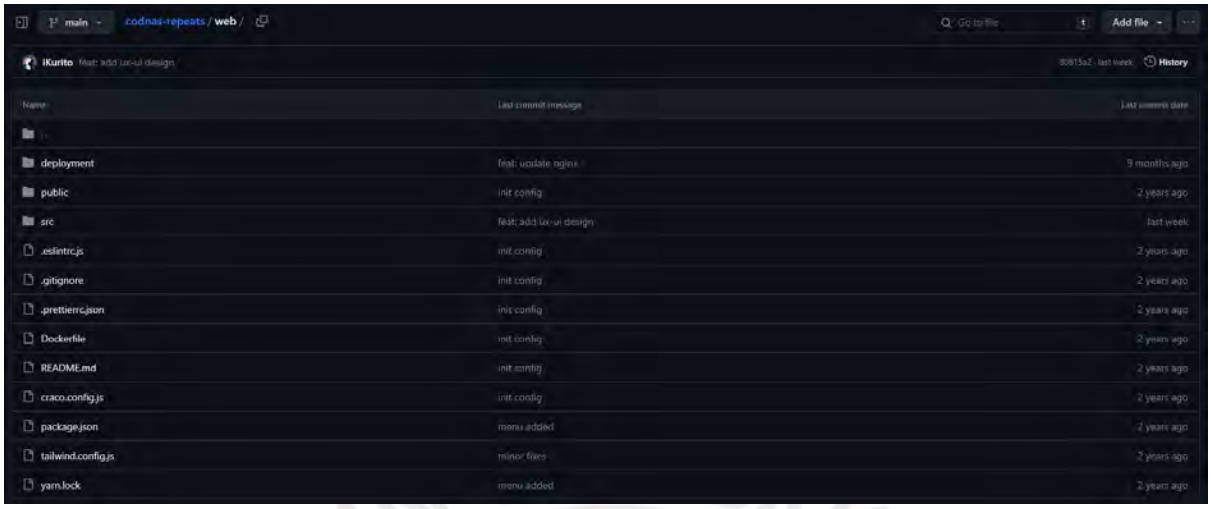
Nota: El gráfico muestra el prototipo de la interfaz de usuario hecho en Figma. Elaboración Propia.

Por otro lado, la interfaz de usuario está alojado en un servidor en la nube de AWS en una instancia EC2 con la finalidad de ser utilizada por los científicos y se utilizó la librería React.js para el desarrollo de la misma. Asimismo, se puede acceder a la interfaz de usuario a través de la siguiente dirección: *https://codnas-repeats.bioinformatica.org/home*. Además, se realizó un manual de uso para poder navegar en la interfaz de usuario, la cual se puede encontrar en el Anexo Ñ: Manual de uso.

Por otra parte, la interfaz de usuario está almacenado en un repositorio de GitHub, la cual se puede apreciar en la Figura 7.9; y se puede acceder al repositorio a través de la siguiente dirección: *https://github.com/iKurito/CoDNaS-Repeats*.

Figure 7.9

Repositorio de github de la interfaz de usuario



Nota: El gráfico muestra el repositorio de github de la interfaz de usuario. Elaboración Propia.

Finalmente, el prototipo de la interfaz de usuario y el manual de uso han sido validadas por la Dra. Layla Hirsh, el Dr. Gustavo Paris y Dr. Nicolás Palopoli; expertos con participación en proyectos de creación de páginas web alojados en la nube de AWS. Esta validación se puede apreciar ingresando a la siguiente dirección: *https://drive.google.com/R10/acta_validacion.pdf*.

7.3. Discusión

El modelamiento de la estructura de la base de datos es representado por un modelo relacional, ya que la información que se va a manejar es consistente y se quiere evitar duplicidad. Asimismo, como el volumen de los datos va a crecer poco a poco en el tiempo, utilizar una base de datos relacional es lo más adecuado.

Además, debemos considerar que elaborar una base de datos es necesaria para almacenar la información obtenida en el proceso del desarrollo de la herramienta que permite el análisis de diversidad conformacional en estructuras de proteínas repetidas, ya que así se tendría un acceso más rápido a dicha información y se evitaría un doble trabajo. Asimismo, en caso no se encuentre la información de diversidad conformacional de alguna proteína en la base de datos, se puede estimar la misma utilizando la herramienta.

Por otro lado, las pruebas funcionales que se realizaron eran necesarias para demostrar que

los resultados obtenidos que están almacenados en la base de datos son correspondientes a los resultados que se obtienen al momento de estimar la diversidad conformacional a través del algoritmo elaborado.

Por otro parte, en caso se requiera mostrar otro tipo de información de la diversidad conformacional de la misma clase de proteína u otra, se puede modificar el prototipo de la interfaz de usuario para que a partir de este diseño poder desarrollar la página web que representará a la herramienta.

Finalmente, se limitó a usar una instancia RDS para la base de datos, ya que la configuración es simple y se limitó a utilizar una instancia EC2 para desarrollar tanto la interfaz de usuario como el servicio web, ya que estando en el mismo servidor, la configuración es más simple y la comunicación entre ambos será más rápido y eficiente. Asimismo, se limitó a utilizar Flask para desarrollar el servicio web, ya que está basado en python y las librerías como biopython solo se podían utilizar en este lenguaje de programación. De la misma manera, se usó la librería React.js para la elaboración de la interfaz de usuario, porque se tiene conocimiento previo utilizando esta libreria.

Capítulo 8

Conclusiones y trabajos futuros

En este capítulo final se describen las conclusiones luego de alcanzar el objetivo de desarrollar una herramienta que permita analizar la diversidad conformacional en estructuras de proteínas repetidas. Asimismo, estas conclusiones detallan la manera en cómo se alcanzaron los objetivos planteados. Y finalmente, se detallan las recomendaciones y las propuestas que se van a tomar en consideración para posibles trabajos futuros, los cuales se van a poder realizar teniendo como base los resultados generados en este proyecto de tesis.

8.1. Conclusiones

Con el fin de lograr los objetivos planteados se elaboró un conjunto de datos y a partir de este se generó aleatoriamente un nuevo conjunto de datos de prueba, ya que se quiere comprobar, validar y experimentar sobre métodos existentes o propuestas de métodos correspondientes al estudio de las proteínas. Asimismo, este conjunto de datos debe estar descrito en base a una estructura de datos que lo representará. Por esta razón, se definió el conjunto de datos de las proteínas repetidas, la cual está representada por los parámetros de los registros ATOM y TER que trazan la estructura de la proteína. Y ya definida esta estructura, se logró elaborar un conjunto de 6329 cadenas de proteínas repetidas que se utilizaron para analizar la diversidad conformacional de las mismas. Asimismo, se escogieron aleatoriamente 1900 cadenas de este tipo de proteínas del conjunto de datos de la misma para formar el conjunto de datos de prueba, el cual fue utilizado para verificar la efectividad de los 5 métodos (2 métodos genéricos y 3 propuestas de métodos) utilizados para analizar la diversidad conformacional.

Por otro lado, plantear una propuesta de método que permita analizar específicamente la diver-

sidad conformacional de las proteínas repetidas no fue una tarea fácil, ya que para conseguir esto, se tuvo que tener una buena comprensión de los métodos existentes. Es así que se tomaron en cuenta dos métodos genéricos que usaban el software Mammoth y el software TM-Align con la finalidad de calcular la diferencia estructural y la similitud estructural a través de la unidad estadística RMSD y TM-score, respectivamente. Y en base a estos dos métodos, se realizó una comparación para verificar que los resultados que se habían obtenido por medio de estos métodos sobre el conjunto de datos de prueba de proteínas repetidas correspondían a la información existente que la base de datos CoDNaS brindaba. Y esto, con el fin de comprobar el buen entendimiento y aplicación de los métodos existentes, el uso correcto de los softwares Mammoth y TMAlign.

Estas tres propuestas de métodos fueron planteadas en base a la región de repetición, a las unidades de repetición y a los métodos genéricos. Es así que la información y los resultados obtenidos empleando estos métodos permitieron a los especialistas tener una visión amplia a verificar y validar qué propuesta de método resultaba la más adecuada para analizar la diversidad conformacional de las proteínas repetidas. Es así que, la propuesta de método seleccionada por los especialistas es la que se basa en generar las conformaciones de las regiones repetidas para luego medir la diferencia estructural que existe entre todas las conformaciones. Y, a partir de este método seleccionado, se procedió a su aplicación en todo el conjunto de datos de las proteínas repetidas.

Finalmente, luego de obtener los datos de diversidad conformacional de las proteínas repetidas utilizando la propuesta del método seleccionado se modeló una base de datos con la ayuda de la herramienta MySQL Workbench para poder almacenar esta información. Y teniendo la base de datos modelada en una instancia RDS junto con la información de diversidad conformacional se desarrolló un servicio web desplegado en la nube de AWS usando Flask y se diseñó un prototipo de cómo se vería la interfaz de usuario utilizando la herramienta Figma, posteriormente, utilizando la librería Reactjs se desarrolló esta interfaz de usuario conectada al servicio web, la cual también está desplegada en la nube de AWS. Es así que teniendo la base de datos, el servicio web y la interfaz de usuario se logró desarrollar la herramienta que permitiera al usuario extraer la información de la base de datos o evaluar la diversidad conformacional de una cadena de la proteína repetida en base al método seleccionado utilizando el servicio web. Y en caso el usuario tuviera alguna duda de cómo utilizar la herramienta desarrollada se elaboró un manual

de uso, en donde se detallan los pasos que se tienen que realizar para usarla de manera adecuada.

8.2. Trabajos futuros

La herramienta creada en el presente proyecto de tesis permite analizar la diversidad conformacional de las proteínas repetidas empleando el método seleccionado, el cual está basado en definir las regiones de repetición de las diferentes conformaciones que presenta la proteína repetida y a partir de estas medir la diferencia estructural. Asimismo, esto se limita a las conformaciones que presenta la proteína repetida y a la misma, y no incluye a otras proteínas pertenecientes a las mismas familias, ya que así es el estudio de la diversidad conformacional.

Sin embargo, incluir a otras estructuras de una misma familia es parte del estudio de la divergencia estructural y sería interesante evaluarla como trabajo futuro. Para esto, se tienen otras dos propuestas de métodos basados en las Unidades de repetición como confórmeros y Unidades de repetición de los confórmeros, ya que los resultados obtenidos de estas dos propuestas van a ser de utilidad para el estudio de la divergencia estructural en estructuras de proteínas repetidas.

Por otra parte, se puede proponer también como trabajo futuro integrar a la herramienta algún software que permita la visualización de la estructura en 3D de la proteína repetida como el software LiteMol, PyMOL, Chimera, entre otros. Asimismo, se puede incluir información sobre el par de confórmeros con el mayor grado de diversidad conformacional en la interfaz de usuario y se puede brindar al usuario reportes del análisis de diversidad conformacional de las proteínas repetidas por medio de gráficos estadísticos. Además, se podría añadir una funcionalidad que permita buscar en la base de datos CoDNaS-PRs o estimar la diversidad conformacional de más de una proteína repetida a la vez.

Finalmente, como último trabajo futuro se propone el desarrollo de un aplicativo móvil en las plataformas IOS y Android. De esta forma, los usuarios van a tener una vía más para acceder a la herramienta.

Referencias

Amazon Web Services, Inc. (2020). AWS - Amazon Web Services. Retrieved June 21, 2020, from https://aws.amazon.com/es/about-aws/

Andrade, M., Perez-Iratxeta, C., & Ponting, C. (2001). Protein repeats: Structures, functions, and evolution. *Journal of Structural Biology*, *134*(2-3), 117–131. https://doi.org/10.1006/jsbi.2001.4392

Biopython. (2020). Biopython - Python Tools for Computational Molecular Biology. Retrieved June 21, 2020, from https://biopython.org/

Brunette, T., Parmeggiani, F., Huang, P.-S., Bhabha, G., Ekiert, D., Tsutakawa, S., Hura, G., Tainer, J., & Baker, D. (2015). Exploring the repeat protein universe through computational protein design. *Nature*, *528*(7583), 580–584. https://doi.org/10.1038/nature16162

Burra, P., Zhang, Y., Godzik, A., & Stec, B. (2009). Global distribution of conformational states derived from redundant models in the PDB points to non-uniqueness of the protein structure. *Proceedings of the National Academy of Sciences of the United States of America*, *106*(26), 10505–10510. https://doi.org/10.1073/pnas.0812152106

Choy, W.-Y., & Forman-Kay, J. (2001). Calculation of ensembles of structures representing the unfolded state of an SH3 domain. *Journal of Molecular Biology*, *308*(5), 1011–1032. https://doi.org/10.1006/jmbi.2001.4750

Codd, E. F. (2002). A relational model of data for large shared data banks. *Software pioneers* (pp. 263–294). Springer.

Delucchi, M., Schaper, E., Sachenkova, O., Elofsson, A., & Anisimova, M. (2020). A New Census of Protein Tandem Repeats and Their Relationship with Intrinsic Disorder. *Genes*, *11*(4). https://doi.org/10.3390/genes11040407

Di Domenico, T., Potenza, E., Walsh, I., Gonzalo Parra, R., Giollo, M., Minervini, G., Piovesan, D., Ihsan, A., Ferrari, C., Kajava, A., & Tosatto, S. (2014). RepeatsDB: A database of

tandem repeat protein structures. *Nucleic Acids Research*, *42*(D1), D352–D357. https://doi.org/10.1093/nar/gkt1175

Djian, P. (1998). Evolution of simple repeats in DNA and their relation to human disease. *Cell*, *94*(2), 155–160. https://doi.org/10.1016/S0092-8674(00)81415-4

Facebook. (2020). Getting Started – React. Retrieved June 21, 2020, from https://reactjs.org/docs/getting-started.html

Figma. (2020). Figma. https://www.figma.com/

Flask MicroFramework. (2020). Flask MicroFramework. Retrieved June 21, 2020, from https://flask.palletsprojects.com/en/1.1.x/

Git. (2020). Git. Retrieved June 21, 2020, from https://git-scm.com/

Goncearenco, A., & Berezovsky, I. (2015). Protein function from its emergence to diversity in contemporary proteins. *Physical Biology*, *12*(4). https://doi.org/10.1088/1478-3975/12/4/045002

Hannan, A. (2018). Tandem repeats mediating genetic plasticity in health and disease. *Nature Reviews Genetics*, *19*(5), 286–298. https://doi.org/10.1038/nrg.2017.115

Hirsh, L., Piovesan, D., Paladin, L., & Tosatto, S. (2016). Identification of repetitive units in protein structures with ReUPred. *Amino Acids*, *48*(6), 1391–1400. https://doi.org/10.1007/s00726-016-2187-2

Huang, Y., Niu, B., Gao, Y., Fu, L., & Li, W. (2010). CD-HIT Suite: a web server for clustering and comparing biological sequences. *Bioinformatics*, *26*(5), 680–682. https://doi.org/10.1093/bioinformatics/btq003

James, L., & Tawfik, D. (2003). Conformational diversity and protein evolution - A 60-year-old hypothesis revisited. *Trends in Biochemical Sciences*, *28*(7), 361–368. https://doi.org/10.1016/S0968-0004(03)00135-X

Javier Zea, D., Miguel Monzon, A., Fornasari, M., Marino-Buslje, C., & Parisi, G. (2013). Protein conformational diversity correlates with evolutionary rate. *Molecular Biology and Evolution*, *30*(7), 1500–1503. https://doi.org/10.1093/molbev/mst065

Juritz, E., Palopoli, N., Fornasari, M., Fernandez-Alberti, S., & Parisi, G. (2013). Protein conformational diversity modulates sequence divergence. *Molecular Biology and Evolution*, *30*(1), 79–87. https://doi.org/10.1093/molbev/mss080

Kachroo, P., Ahuja, M., Leong, S., & Chattoo, B. (1997). Organisation and molecular analysis of repeated DNA sequences in the rice blast fungus Magnaporthe grisea. *Current Genetics*, *31*(4), 361–369. https://doi.org/10.1007/s002940050217

Kajava, A. (2012). Tandem repeats in proteins: From sequence to structure. *Journal of Structural Biology*, *179*(3), 279–288. https://doi.org/10.1016/j.jsb.2011.08.009

Kajava, A., & Steven, A. (2006). β-Rolls, β-Helices, and Other β-Solenoid Proteins. *Advances in Protein Chemistry*, *73*, 55–96. https://doi.org/10.1016/S0065-3233(06)73003-0

Kitchenham, B. (2004). Procedures for performing systematic reviews. *Keele, UK, Keele University*, *33*(2004), 1–26.

Kufareva, I., & Abagyan, R. (2012). Methods of protein structure comparison. *Methods in Molecular Biology*, *857*, 231–257. https://doi.org/10.1007/978-1-61779-588-6_10

Kumar, S., Ma, B., Tsai, C.-J., Sinha, N., & Nussinov, R. (2000). Folding and binding cascades: Dynamic landscapes and population shifts. *Protein Science*, *9*(1), 10–19.

Lodish, H., Berk, A., Matsudaira, P., Kaiser, C. A., Krieger, M., Scott, M. P., Zipursky, L., & Darnell, J. (2003). *Molecular Cell Biology* (5th). New York: W. H. Freeman.

Maiorov, V., & Crippen, G. (1994). Significance of root-mean-square deviation in comparing three-dimensional structures of globular proteins. *Journal of Molecular Biology*, *235*(2), 625–634. https://doi.org/10.1006/jmbi.1994.1017

Marcotte, E., Pellegrini, M., Yeates, T., & Eisenberg, D. (1999). A census of protein repeats. *Journal of Molecular Biology*, *293*(1), 151–160. https://doi.org/10.1006/jmbi.1999.3136

Minami, S., Sawada, K., & Chikenji, G. (2013). MICAN : A protein structure alignment algorithm that can handle Multiple-chains, Inverse alignments, Cα only models, Alternative alignments, and Non-sequential alignments. *BMC Bioinformatics*, *14*. https://doi.org/10.1186/1471-2105-14-24

Mirsky, A. E., & Pauling, L. (1936). On the structure of native, denatured, and coagulated proteins. *Proceedings of the National Academy of Sciences of the United States of America*, *22*(7), 439.

Monzon, A., Fornasari, M., Zea, D., & Parisi, G. (2019). Exploring Protein Conformational Diversity. *Methods in Molecular Biology*, *1851*, 353–365. https://doi.org/10.1007/978-1-4939-8736-8_20

Monzon, A., Juritz, E., Fornasari, M., & Parisi, G. (2013). CoDNaS: A database of conformational diversity in the native state of proteins. *Bioinformatics*, *29*(19), 2512–2514. https://doi.org/10.1093/bioinformatics/btt405

Monzon, A., Rohr, C., Fornasari, M., & Parisi, G. (2016). CoDNaS 2.0: A comprehensive database of protein conformational diversity in the native state. *Database*, *2016*. https://doi.org/10.1093/database/baw038

Monzon, A., Zea, D., Fornasari, M., Saldaño, T., Fernandez-Alberti, S., Tosatto, S., & Parisi, G. (2017). Conformational diversity analysis reveals three functional mechanisms in proteins. *PLoS Computational Biology*, *13*(2). https://doi.org/10.1371/journal.pcbi.1005398

Monzon, A., Zea, D., Marino-Buslje, C., & Parisi, G. (2017). Homology modeling in a dynamical world. *Protein Science*, *26*(11), 2195–2206. https://doi.org/10.1002/pro.3274

National Science Foundation, US Department of Energy, National Cancer Institute, National Institute of Allergy and Infectious Diseases, & National Institute of General Medical Sciences. (2020a). Educational Portal of RCSB PDB. Retrieved June 21, 2020, from https://pdb101.rcsb.org/learn/guide-to-understanding-pdb-data/introduction

National Science Foundation, US Department of Energy, National Cancer Institute, National Institute of Allergy and Infectious Diseases, & National Institute of General Medical Sciences. (2020b). RCSB Protein Data Bank. Retrieved June 21, 2020, from https://www.rcsb.org/

Oracle. (2020). MySQL :: MySQL Workbench Manual. Retrieved June 21, 2020, from https://dev.mysql.com/doc/workbench/en/

Ortiz, A., Strauss, C., & Olmea, O. (2002). MAMMOTH (Matching molecular models obtained from theory): An automated method for model comparison. *Protein Science*, *11*(11), 2606–2621. https://doi.org/10.1110/ps.0215902

Overleaf. (2020). Overleaf. https://es.overleaf.com/

Paladin, L., & Tosatto, S. (2015). Comparison of protein repeat classifications based on structure and sequence families. *Biochemical Society Transactions*, *43*, 832–837. https://doi.org/10.1042/BST20150079

Palopoli, N., Monzon, A., Parisi, G., & Fornasari, M. (2016). Addressing the role of conformational diversity in protein structure prediction. *PLoS ONE*, *11*(5). https://doi.org/10.1371/journal.pone.0154923

Parisi, G., Zea, D., Monzon, A., & Marino-Buslje, C. (2015). Conformational diversity and the emergence of sequence signatures during evolution. *Current Opinion in Structural Biology*, *32*, 58–65. https://doi.org/10.1016/j.sbi.2015.02.005

Postman. (2020). Postman — The Collaboration Platform for API Development. Retrieved June 21, 2020, from https://www.postman.com/

Python Software Foundation. (2020). What is Python? Executive Summary. Retrieved June 21, 2020, from https://www.python.org/doc/essays/blurb/

RStudio. (2020). The RStudio. Retrieved June 21, 2020, from https://rstudio.com/

Rueda, A., Monzon, A., Ardanaz, S., Iglesias, L., & Parisi, G. (2018). Large scale analysis of protein conformational transitions from aqueous to non-aqueous media. *BMC Bioinformatics*, *19*(1). https://doi.org/10.1186/s12859-018-2044-2

Saldaño, T. E., Freixas, V. M., Tosatto, S. C. E., Parisi, G., & Fernandez-Alberti, S. (2020). Exploring Conformational Space with Thermal Fluctuations Obtained by Normal-Mode Analysis. *Journal of Chemical Information and Modeling*. https://doi.org/10.1021/acs.jcim.9b01136

Schaper, E., Gascuel, O., & Anisimova, M. (2014). Deep conservation of human protein tandem repeats within the eukaryotes. *Molecular Biology and Evolution*, *31*(5), 1132–1148. https://doi.org/10.1093/molbev/msu062

Schaper, E., Korsunsky, A., Pečerska, J., Messina, A., Murri, R., Stockinger, H., Zoller, S., Xenarios, I., & Anisimova, M. (2015). TRAL: Tandem repeat annotation library. *Bioinformatics*, *31*(18), 3051–3053. https://doi.org/10.1093/bioinformatics/btv306

Schrödinger LLC. (2020). The PyMOL Molecular Graphics System. Retrieved June 21, 2020, from https://pymol.org/2/

Suits, M., Sperandeo, P., Dehò, G., Polissi, A., & Jia, Z. (2008). Novel Structure of the Conserved Gram-Negative Lipopolysaccharide Transport Protein A and Mutagenesis Analysis. *Journal of Molecular Biology*, *380*(3), 476–488. https://doi.org/10.1016/j.jmb.2008.04.045

Tsai, C.-J., Kumar, S., Ma, B., & Nussinov, R. (1999). Folding funnels, binding funnels, and protein function. *Protein Science*, *8*(6), 1181–1190. https://doi.org/10.1110/ps.8.6.1181

Visual Studio Code. (2020). Documentation for Visual Studio Code. Retrieved June 21, 2020, from https://code.visualstudio.com/docs

Wang, L., Wu, L.-Y., Wang, Y., Zhang, X.-S., & Chen, L. (2010). SANA: An algorithm for sequential and non-sequential protein structure alignment. *Amino Acids*, *39*(2), 417–425. https://doi.org/10.1007/s00726-009-0457-y

Zea, D., Monzon, A., Gonzalez, C., Fornasari, M., Tosatto, S., & Parisi, G. (2016). Disorder transitions and conformational diversity cooperatively modulate biological function in proteins. *Protein Science*, *25*(6), 1138–1146. https://doi.org/10.1002/pro.2931

Zhang, Y., & Skolnick, J. (2005). TM-align: A protein structure alignment algorithm based on the TM-score. *Nucleic Acids Research*, *33*(7), 2302–2309. https://doi.org/10.1093/nar/gki524

Zhu, J., & Weng, Z. (2005). FAST: A novel protein structure alignment algorithm. *Proteins: Structure, Function and Genetics*, *58*(3), 618–627. https://doi.org/10.1002/prot.20331

Anexos

En esta sección se detallan los anexos relacionados al presente documento.

Anexo A

Plan de Proyecto

En el presente Anexo se desarrollará el plan de proyecto de tesis, el cual se detallará la justificación del proyecto mismo, se identificará las limitaciones y riesgos del proyecto. Además, se hará mención del alcance que tendrá el proyecto. Asimismo, se establecerá la estructura de descomposición del trabajo del proyecto. Junto a esto, se desarrollará una lista de tareas y el cronograma que se seguirá a pie para la ejecución del proyecto. También, se describirá la lista de recursos, la cual estará integrada por la lista de personas involucradras, los materiales, los estándares , el equipamiento y las herramientas requeridas para el proyecto. Por último se desarrollará el costeo del proyecto.

A.1. Justificación

En la presente sección, se justificará el proyecto, se planteará claramente el propósito del mismo, se explicará por qué es conveniente y cuáles son los beneficios de este.

Hoy en día, las proteínas repetidas representan una fuente fundamental de información para explicar la diversidad estructural contemporánea y las propiedades fisioquímicas de los pliegues altamente designables (Goncearenco and Berezovsky, 2015). Además, tienen gran importancia en relación a enfermedades humanas (Kajava and Steven, 2006) y aplicaciones de ingeniería que hacen uso de proteínas (Brunette et al., 2015). Sin embargo, esta clase de proteína todavía pertenece a la ”materia oscura“ del universo proteico (Hirsh et al., 2016) y, actualmente, no hay publicación alguna que indique que se haya implementado una herramienta o elaborado un método que permita evaluar la diversidad conformacional, específicamente, en las proteínas repetidas. Por tal motivo, el presente proyecto de tesis tiene la finalidad de poder plantear un

método que permita evaluar la diversidad conformacional en esta clase particular de proteínas y desarrollar una herramienta que permita calcular y visualizar esta información. Esto permitirá tener información nueva en las bases de datos de diversidad conformacional existentes y se conocerán las características de diversidad conformacional de las proteínas repetidas.

Por lo tanto, teniendo en cuenta lo descrito en el anterior párrafo, este proyecto se justifica como necesario de realizar y que presenta una solución al problema existente mencionado anteriormente.

A.2. Viabilidad

En la presente sección se detallará si se cuenta con la disponibilidad de los recursos y los conocimientos necesarios para el desarrollo del proyecto de tesis.

1. Viabilidad técnica

   Las herramientas que se van a utilizar para el desarrollo del proyecto son de acceso libre o educativo, por lo que no involucrará costo económico alguno. Asimismo, quien está a cargo de este proyecto cuenta con experiencia en el uso de estas herramientas. Además, con respecto al hardware que se usará, se cuenta con un ordenador personal. Este ordenador está acondicionado de acuerdo a las necesidades del proyecto. Por lo tanto, se concluye que el proyecto es técnicamente viable.

2. Viabilidad temporal

   La estimación del desarrollo total de las tareas del proyecto de fin de carrera tendrá una duración de 8 meses, por lo que el proyecto es temporalmente viable, ya que los objetivos planteados se van a poder realizar en este tiempo.

3. Viabilidad económica

   Este proyecto es económicamente viable debido a que por ser estudiante se cuenta con 50 dólares de créditos para el uso de AWS. Además, el resto de herramientas son de acceso libre y se cuentan con los equipos necesarios, por lo que no se necesitará de realizar algún gasto.

En conclusión, teniendo en cuenta lo anterior, el proyecto de tesis se justifica como viable para su ejecución.

A.3. Alcance del Proyecto

A continuación, se indicará el alcance del proyecto por medio de las actividades que se incluyen dentro del desarrollo del proyecto, de la misma manera, las actividades que no se incluyen.

El presente proyecto de fin de carrera es un proyecto de investigación enfocado en el área de bioinformática. Tiene como finalidad elaborar un método que permita analizar la diversidad conformacional de las proteínas repetidas, ya que esta clase de proteínas poseen la característica particular de tener repeticiones en su estructura. Asimismo, desarrollar una herramienta que permita evaluar y visualizar la diversidad conformacional de las proteínas repetidas o extraer y visualizar la información de diversidad conformacional de las proteínas repetidas obtenidas de la base de datos.

Para esto, se está delimitando a elaborar tres propuestas de métodos que permitirán analizar la diversidad conformacional de las proteínas repetidas. Estas propuestas se van a comparar entre sí junto con dos métodos existentes (métodos genéricos) con la finalidad de verificar cuál presenta los mejores resultados para poder analizar la diversidad conformacional de esta clase de proteínas.

Luego de elegir el método apropiado, este método se aplicará en el conjunto de datos de las proteínas repetidas y la información que se recolecte será guardada en una base de datos. Además, usando una interfaz de usuario, se visualizará la información extraída de la base de datos por medio de un servicio web que se desarrollará. Esta información consistirá en datos generales de la proteína repetida, en información estructural de la proteína repetida y las distintas conformaciones que la proteína repetida presenta. En caso, la información de la proteína insertada como dato no se encuentre en la base de datos, se procederá a calcular y evaluar el análisis de diversidad conformacional, utilizando el servicio web, para luego mostrar su información.

A.4. Restricciones

A continuación, se presentarán las restricciones identificadas en el proyecto que afectarían en la planificación del mismo:

1. El tiempo a realizar este proyecto es de dos ciclos regulares académicos que equivale a 8 meses aproximadamente, por lo tanto, el proyecto se debe desarrollar y estar listo en este tiempo como máximo.

2. El presente proyecto tiene la restricción de utilizar solo proteínas repetidas para analizar su diversidad conformacional.

A.5. Identificación de los riesgos del proyecto

En la presente sección, se muestra en la Tabla A.2 los riesgos identificados, el impacto y la severidad que afecta al proyecto de tesis; asimismo, la mitigación y contingencia que se realizarán ante los riesgos. Además, se presenta la Tabla A.1, la cual menciona la leyenda para la Tabla A.2.

Tabla A.1

Leyenda de la Tabla A.2.

Item	**Muy Bajo**	**Bajo**	**Medio**	**Alto**	**Muy Alto**
Probabilidad	1	3	5	7	9
Impacto	1.5	3.5	5.5	7.5	9.5
Severidad	Probabilidad X Impacto				

Tabla A.2

Riesgos identificados del proyecto.

Riesgo Identificado	**Probabilidad**	**Impacto**	**Severidad**	**Mitigación**	**Contingencia**

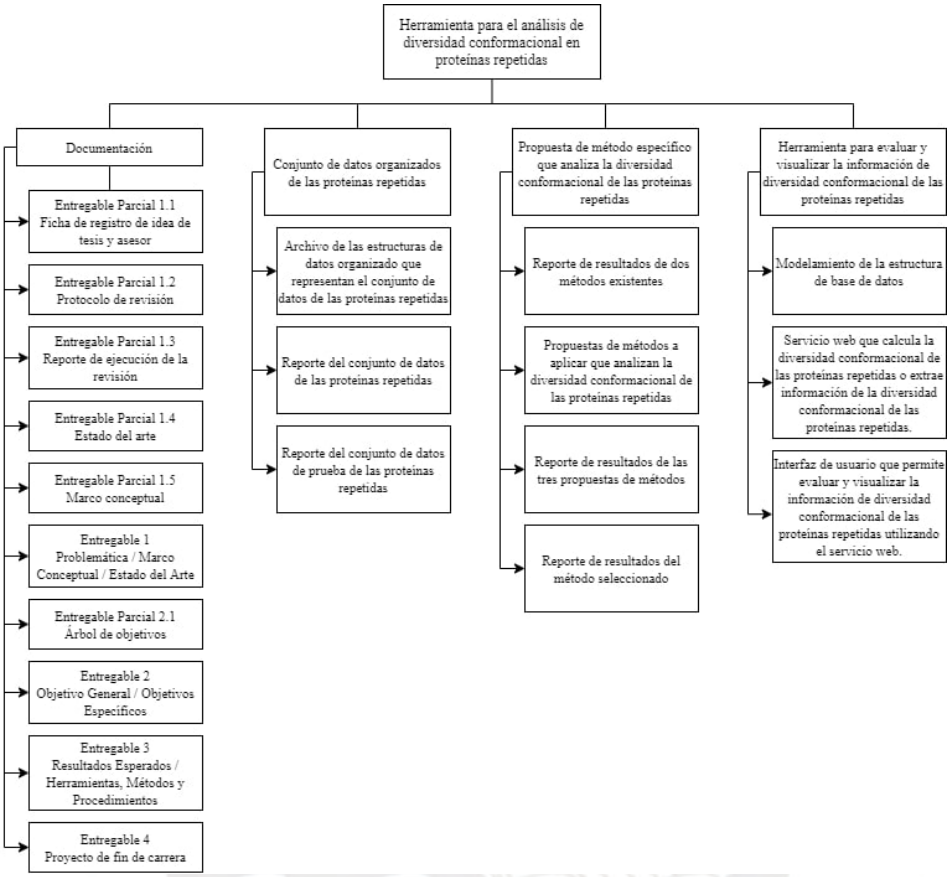
Dificultad con la curva de aprendizaje de los temas biológicos y bioinformáticos.	3	5.5	16.5	Llevar cursos virtuales de biología y bioinformática.	Consultar con expertos del tema. Para esto, se tiene contacto con toda la comunidad experta por medio de la asesora y los co-asesores
Poca disponibilidad de los expertos.	1	7.5	7.5	Coordinar reuniones con anticipación para separar tiempo en la agenda del experto.	Comunicación a través de correo o videollamada.
Caída de la base de datos de proteínas repetidas.	1	9.5	9.5	Comunicación con las organizaciones que tienen a su cargo la base de datos de proteínas repetidas para informarles del proyecto de tesis.	La asesora y los co-asesores de tesis tienen acceso a la base de datos donde se puede extraer información de las proteínas repetidas.

A.6. Estructura de Descomposición del Trabajo (EDT)

A continuación, en la figura A.1 se presenta la estructura de descomposición del trabajo (EDT). Esta gráfica muestra los entregables que se van a desarrollar en el presente proyecto de tesis.

Figure A.1

Estructura de descomposición del trabajo (EDT).



Nota: El gráfico representa la estructura de descomposición del trabajo. Elaboración propia.

A.7. Lista de Tareas

En la presente sección, se describe en la Tabla A.3 la lista de tareas que se realizarán durante el desarrollo del proyecto de tesis. Esta tabla incluye la duración estimada, el esfuerzo asociado y el costo estimado de cada tarea.

Tabla A.3

Lista de tareas del proyecto.

Lista de Tareas				
#	**Nombre de Tarea**	**Duración Estimada**	**Esfuerzo Estimado**	**Costo Estimado**
1	Reunión con la asesora	1 día	2 horas	S/. 400.00
2	Definir problemática	14 días	48 horas	S/. 2400.00

3	Reunión con la asesora	1 día	2 horas	S/. 400.00
4	Investigar el estado del arte	7 días	100 horas	S/. 5000.00
5	Reunión con la asesora	1 día	2 horas	S/. 400.00
6	Definir el marco conceptual y marco teórico	7 días	48 horas	S/. 2400.00
7	Reunión con la asesora	1 día	2 horas	S/. 400.00
8	Definir objetivos específicos	7 días	40 horas	S/. 2000.00
9	Reunión con la asesora	1 día	2 horas	S/. 400.00
10	Definir resultados esperados por cada objetivo	14 días	48 horas	S/. 2400.00
11	Reunión con la asesora	1 día	2 horas	S/. 400.00
12	Definir las herramientas y métodos a utilizar	14 días	48 horas	S/. 2400.00
13	Reunión con la asesora	1 día	2 horas	S/. 400.00
14	Justificar el proyecto	1 día	12 horas	S/. 600.00
15	Definir el alcance del proyecto	2 días	12 horas	S/. 600.00
16	Identificar las restricciones del proyecto	2 días	12 horas	S/. 600.00
17	Identificar los riesgos del proyecto	2 días	12 horas	S/. 600.00
18	Reunión con la asesora	1 día	2 horas	S/. 400.00
19	Justificar viabilidad	1 día	12 horas	S/. 600.00
20	Elaborar la estructura de descomposición de trabajos (EDT)	1 día	12 horas	S/. 600.00
21	Definir la lista de tareas	1 día	12 horas	S/. 600.00
22	Elaborar el cronograma del proyecto	2 días	12 horas	S/. 600.00
23	Definir los recursos y costos del proyecto	2 días	12 horas	S/. 600.00
24	Reunión con la asesora y los co-asesores	1 día	4 horas	S/. 4800.00
25	Preparar el ambiente de desarrollo	3 días	20 horas	S/. 1000.00

26	Reunión con la asesora y los co-asesores	1 día	4 horas	S/. 4800.00
27	Elaborar el archivo de la estructura de datos organizado que representa el conjunto de datos de las proteínas repetidas	6 días	12 horas	S/. 600.00
28	Redactar dentro del documento final del proyecto de tesis el resumen correspondiente al resultado esperado 1 (R1) sin incluir la descripción del IOV (indicador objetivamente verificable)	1 día	4 horas	S/. 200.00
29	Reunión con la asesora y los co-asesores	1 día	2 horas	S/. 2400.00
30	Elaborar el reporte del conjunto de datos de las proteínas repetidas	4 días	12 horas	S/. 600.00
31	Redactar dentro del documento final del proyecto de tesis la descripción el indicador objetivamente verificable (IOV) del resultado esperado 1 (R1)	1 día	1 hora	S/. 50.00
32	Redactar dentro del documento final del proyecto de tesis el resumen correspondiente al resultado esperado 2 (R2) sin incluir la descripción del IOV (indicador objetivamente verificable)	1 día	4 horas	S/. 200.00
33	Elaborar el reporte del conjunto de datos de prueba de las proteínas repetidas	1 día	4 horas	S/. 200.00

34	Redactar dentro del documento final del proyecto de tesis la descripción el indicador objetivamente verificable (IOV) del resultado esperado 2 (R2)	1 día	1 hora	S/. 50.00
35	Redactar dentro del documento final del proyecto de tesis el resumen correspondiente al resultado esperado 3 (R3) sin incluir la descripción del IOV (indicador objetivamente verificable)	1 día	4 horas	S/. 200.00
36	Reunión con la asesora y los co-asesores	1 día	4 horas	S/. 4800.00
37	Elaborar el reporte del prototipo de la interfaz de usuario	6 días	20 horas	S/. 1000.00
38	Redactar dentro del documento final del proyecto de tesis la descripción del indicador objetivamente verificable (IOV) del resultado esperado 3 (R3)	1 día	1 hora	S/. 50.00
39	Redactar dentro del documento final del proyecto de tesis un avance del resumen correspondiente al resultado esperado 10 (R10) sin incluir la descripción del indicador objetivamente verificable (IOV)	1 día	2 horas	S/. 100.00
40	Reunión con la asesora y los co-asesores	1 día	2 horas	S/. 2400.00
41	Elaborar el documento del modelamiento de la estructura de la base de datos	6 días	8 horas	S/. 400.00

42	Redactar dentro del documento final del proyecto de tesis el resumen correspondiente al resultado esperado 8 (R8) sin incluir la descripción del IOV (indicador objetivamente verificable)	1 día	4 horas	S/. 200.00
43	Reunión con la asesora y los co-asesores	1 día	2 horas	S/. 2400.00
44	Elaborar el documento de arquitectura del servicio web	1 día	4 horas	S/. 200.00
45	Redactar dentro del documento final del proyecto de tesis un avance del resumen correspondiente al resultado esperado 9 (R9) sin incluir la descripción del IOV (indicador objetivamente verificable)	1 día	2 horas	S/. 100.00
46	Reunión con la asesora y los co-asesores	1 día	1 hora	S/. 1200.00
47	Elaborar el reporte de resultados de los dos métodos genéricos que analizan la diversidad conformacional en proteínas	4 días	20 horas	S/. 1000.00
48	Redactar dentro del documento final del proyecto de tesis el resumen correspondiente al resultado esperado 4 (R4) sin incluir la descripción del IOV (indicador objetivamente verificable)	1 día	4 horas	S/. 200.00
49	Reunión con la asesora y los co-asesores	1 día	0.5 horas	S/. 600.00

50	Elaborar el reporte de propuestas de los métodos a aplicar para analizar la diversidad conformacional de las proteínas repetidas	6 días	20 horas	S/. 1000.00
51	Redactar dentro del documento final del proyecto de tesis la descripción del indicador objetivamente verificable (IOV) del resultado esperado 4 (R4)	1 día	1 hora	S/. 50.00
52	Redactar dentro del documento final del proyecto de tesis el resumen correspondiente al resultado esperado 5 (R5) sin incluir la descripción del IOV (indicador objetivamente verificable)	1 día	4 horas	S/. 200.00
53	Reunión con la asesora y los co-asesores	1 día	0.5 horas	S/. 600.00
54	Elaborar el reporte de resultados de las tres propuestas de métodos que van a permitir el análisis de diversidad conformacional de las proteínas repetidas	6 días	30 horas	S/. 1500.00
55	Redactar dentro del documento final del proyecto de tesis la descripción del indicador objetivamente verificable (IOV) del resultado esperado 5 (R5)	1 día	1 hora	S/. 50.00

56	Redactar dentro del documento final del proyecto de tesis el resumen correspondiente al resultado esperado 6 (R6) sin incluir la descripción del IOV (indicador objetivamente verificable)	1 día	4 horas	S/. 200.00
57	Reunión con la asesora y los co-asesores	1 día	1 hora	S/. 1200.00
58	Elaborar el reporte de resultados del método seleccionado	6 días	20 horas	S/. 1000.00
59	Redactar dentro del documento final del proyecto de tesis la descripción del indicador objetivamente verificable (IOV) del resultado esperado 6 (R6)	1 día	1 hora	S/. 50.00
60	Redactar dentro del documento final del proyecto de tesis el resumen correspondiente al resultado esperado 7 (R7) sin incluir la descripción del IOV (indicador) objetivamente verificable	1 día	4 horas	S/. 200.00
61	Reunión con la asesora y los co-asesores	1 día	1 hora	S/. 1200.00
62	Elaborar el repositorio del código fuente del servicio web en Github	3 días	20 horas	S/. 1000.00
63	Elaborar el informe de pruebas funcionales del servicio web	3 días	10 horas	S/. 500.00

64	Redactar dentro del documento final del proyecto de tesis la descripción del indicador objetivamente verificable (IOV) del resultado esperado 7 (R7)	1 día	1 hora	S/. 50.00
65	Redactar dentro del documento final del proyecto de tesis el avance restante del resumen correspondiente al resultado esperado 9 (R9) sin incluir la descripción del IOV (indicador objetivamente verificable)	1 día	4 horas	S/. 200.00
66	Reunión con la asesora y los co-asesores	1 día	1 hora	S/. 1200.00
67	Elaborar el repositorio del código fuente de la interfaz de usuario	3 días	20 horas	S/. 1000.00
68	Elaborar el manual de uso	3 días	10 horas	S/. 500.00
69	Redactar dentro del documento final del proyecto de tesis la descripción del indicador objetivamente verificable (IOV) del resultado esperado 9 (R9)	1 día	1 hora	S/. 50.00
70	Redactar dentro del documento final del proyecto de tesis el avance restante del resumen correspondiente al resultado esperado 10 (R10) incluyendo la descripción del IOV (indicador objetivamente verificable)	1 día	4 horas	S/. 200.00
71	Reunión con la asesora y los co-asesores	1 día	2 horas	S/. 2400.00

A.8. Cronograma del Proyecto

A continuación, se presenta los cronogramas respectivos a las fases de planeación y ejecución del proyecto de tesis.

1. Fase de planeación

   En la Tabla A.4 se describe el cronograma de las tareas a cumplir durante la fase de planeación del proyecto de tesis.

Tabla A.4

Cronograma de la fase de planeación del proyecto de tesis.

N°	Actividad	Duración (días)	Inicio	Fin	Predecesor
0	Herramienta para el análisis de la diversidad conformacional en estructuras de proteínas repetidas	216 días	06/04/20	08/11/20	-
1	**Problemática**	**14 días**	**06/04/20**	**19/04/20**	**-**
1.1	Definir la problemática	14 días	06/04/20	19/04/20	-
2	**Estado del arte**	**14 días**	**20/04/20**	**03/05/20**	**1**
2.1	Investigar el estado del arte	14 días	20/04/20	03/05/20	1.1
3	**Desarrollo del marco conceptual**	**7 días**	**04/05/20**	**10/05/20**	**1**
3.1	Definir el marco conceptual y el marco teórico	7 días	04/05/20	10/05/20	2.1
4	**Objetivos y resultados**	**21 días**	**11/05/20**	**31/05/20**	**2**
4.1	Definir objetivos específicos	7 días	11/05/20	17/05/20	3.1
4.2	Definir resultados esperados por cada objetivo	14 días	18/05/20	31/05/20	4.1
5	**Métodos y herramientas**	**14 días**	**01/06/20**	**14/06/20**	**3**
5.1	Definir herramientas y métodos a utilizar	14 días	01/06/20	14/06/20	4.2
6	**Plan de proyecto**	**14 días**	**28/06/20**	**11/07/20**	**4**

6.1	Justificar el proyecto	1 día	28/06/20	28/06/20	5.1
6.2	Definir el alcance del proyecto	2 días	29/06/20	30/06/20	6.1
6.3	Identificar las restricciones del proyecto	2 días	01/07/20	02/07/20	6.2
6.4	Identificar los riesgos del proyecto	2 días	03/05/20	04/07/20	6.3
6.5	Justificar la viabilidad	1 día	05/07/20	05/07/20	6.4
6.6	Elaborar la EDT	1 día	06/07/20	06/07/20	6.5
6.7	Definir la lista de tareas	1 día	07/07/20	07/07/20	6.6
6.8	Elaborar el cronograma del proyecto	2 días	08/07/20	09/07/20	6.7
6.9	Definir los recursos y costos del proyecto	2 días	10/07/20	11/07/20	6.8
Presentación del proyecto de tesis 1		**1 día**	**23/07/20**	**23/07/20**	**5**

2. Fase de ejecución

   En la Tabla A.5 se describe el cronograma de las tareas a cumplir durante la fase de ejecución del proyecto de tesis.

Tabla A.5

Cronograma de la fase de ejecución del proyecto de tesis.

SEM.	N°	Actividad	Duración (días)	Inicio	Fin	Predecesor
1	1	Preparar el ambiente de desarrollo	1 día	31/08/2020	31/08/2020	-
Objetivo Específico 1 (O1) Avance del 100%			**12 días**	**31/08/2020**	**11/09/2020**	**-**

1	2	Elaborar el archivo de la estructura de datos organizado que representa el conjunto de datos de las proteínas repetidas	4 días	31/08/2020	03/09/2020	1
1	3	Redactar dentro del documento final del proyecto de tesis el resumen correspondiente al resultado esperado 1 (Estructura de datos organizada para representar al conjunto de datos de las proteínas repetidas) sin incluir la descripción del indicador objetivamente verificable (IOV)	1 día	04/09/2020	04/09/2020	2
2	4	Exponer el avance del proyecto de fin de carrera (Exposición 1)	1 día	07/09/2020	07/09/2020	3
2	5	Elaborar el reporte del conjunto de datos de las proteínas repetidas	4 días	05/09/2020	08/09/2020	3

2	6	Redactar dentro del documento final del proyecto de tesis la descripción del indicador objetivamente verificable (IOV) del resultado esperado 1 (Estructura de datos organizada para representar al conjunto de datos de las proteínas repetidas	1 día	09/09/2020	09/09/2020	5
2	7	Redactar dentro del documento final del proyecto de tesis el resumen correspondiente al resultado esperado 2 (Conjunto de datos de las proteínas repetidas) sin incluir la descripción del indicador objetivamente verificable (IOV)	1 día	09/09/2020	09/09/2020	6
2	8	Elaborar el reporte del conjunto de datos de prueba de las proteínas repetidas	1 día	10/09/2020	10/09/2020	7

2	9	Redactar dentro del documento final del proyecto de tesis la descripción del indicador objetivamente verificable (IOV) del resultado esperado 2 (Conjunto de datos de las proteínas repetidas)	1 día	11/09/2020	11/09/2020	8
2	10	Redactar dentro del documento final del proyecto de tesis el resumen correspondiente al resultado esperado 3 (Conjunto de datos de prueba de las proteínas repetidas) sin incluir la descripción del indicador objetivamente verificable (IOV)	1 día	11/09/2020	11/09/2020	9
Objetivo Específico 3 (O3) Avance del 60%			**16 días**	**12/09/2020**	**27/09/2020**	**O1**
3	11	Exponer el avance del proyecto de fin de carrera (Exposición 2)	1 día	14/09/2020	14/09/2020	10
3	12	Elaborar el reporte del prototipo de la interfaz de usuario	6 días	12/09/2020	17/09/2020	10

3	13	Redactar dentro del documento final del proyecto de tesis la descripción del indicador objetivamente verificable (IOV) del resultado esperado 3 (Conjunto de datos de prueba de las proteínas repetidas)	1 día	18/09/2020	18/09/2020	12
3	14	Redactar dentro del documento final del proyecto de tesis un avance del resumen correspondiente al resultado esperado 10 (Interfaz de usuario que permita evaluar y visualizar la información de diversidad conformacional de las proteínas repetidas utilizando el servicio web) sin incluir la descripción del indicador objetivamente verificable (IOV)	1 día	18/09/2020	18/09/2020	13

4	15	Exponer el avance del proyecto de fin de carrera (Exposición 3)	1 día	21/09/2020	21/09/2020	14
4	16	Elaborar el documento del modelamiento de la estructura de base de datos	6 días	19/09/2020	24/09/2020	15
4	17	Redactar dentro del documento final del proyecto de tesis el resumen correspondiente al resultado esperado 8 (Modelamiento de la estructura de base de datos que contiene la información de diversidad conformacional de las proteínas repetidas) sin incluir la descripción del indicador objetivamente verificable (IOV)	1 día	25/09/2020	25/09/2020	16
5	18	Exponer el avance del proyecto de fin de carrera (Exposición 4)	1 día	28/09/2020	28/09/2020	17
5	19	Elaborar el documento de arquitectura del servicio web	1 día	26/09/2020	26/09/2020	17

5	20	Redactar dentro del documento final del proyecto de tesis un avance del resumen correspondiente al resultado esperado 9 (Servicio web que permita evaluar la diversidad conformacional de las proteínas repetidas o extraer la información de diversidad conformacional de las proteínas repetida) sin incluir la descripción del indicador objetivamente verificable (IOV)	1 día	27/09/2020	27/09/2020	19
Objetivo Específico 2 (O2) Avance del 100%			**27 días**	**28/09/2020**	**23/10/2020**	**O1**
5	21	Elaborar el reporte de resultados de los dos métodos genéricos que analizan la diversidad conformacional en proteínas	4 días	28/09/2020	01/10/2020	20

5	22	Redactar dentro del documento final del proyecto de tesis el resumen correspondiente al resultado esperado 4 (Comparación de resultados de dos métodos existentes para analizar la diversidad conformacional sobre el conjunto de datos de prueba de proteínas repetidas con los resultados de la base de datos CoDNaS) sin incluir la descripción del indicador objetivamente verificable (IOV)	1 día	02/10/2020	02/10/2020	21
6	23	Presentar el avance parcial del proyecto de tesis	1 día	05/10/2020	05/10/2020	22
6	24	Elaborar el reporte de propuestas de los métodos a aplicar para analizar la diversidad conformacional de las proteínas repetidas	6 días	03/10/2020	08/10/2020	22

6	25	Redactar dentro del documento final del proyecto de tesis la descripción del indicador objetivamente verificable (IOV) del resultado esperado 4 (Comparación de resultados de dos métodos existentes para analizar la diversidad conformacional sobre el conjunto de datos de prueba de proteínas repetidas con los resultados de la base de datos CoDNaS)	1 día	09/10/2020	09/10/2020	24

6	26	Redactar dentro del documento final del proyecto de tesis el resumen correspondiente al resultado esperado 5 (Tres propuestas de métodos a aplicar para analizar la diversidad conformacional de las proteínas repetidas) sin incluir la descripción del indicador objetivamente verificable (IOV)	1 día	09/10/2020	09/10/2020	25
7	27	Presentar el avance parcial del proyecto de tesis	1 día	12/10/2020	12/10/2020	26
7	28	Elaborar el reporte de resultados de las tres propuestas de métodos que van a permitir el análisis de diversidad conformacional de las proteínas repetidas	6 días	10/10/2020	15/10/2020	26

7	29	Redactar dentro del documento final del proyecto de tesis la descripción del indicador objetivamente verificable (IOV) del resultado esperado 5 (Tres propuestas de métodos a aplicar para analizar la diversidad conformacional de las proteínas repetidas)	1 día	16/10/2020	16/10/2020	28

7	30	Redactar dentro del documento final del proyecto de tesis el resumen correspondiente al resultados esperado 6 (Resultados obtenidos de los métodos elaborados sobre el conjunto de datos de prueba de las proteínas repetidas y la comparación con los resultados de los dos métodos genéricos) sin incluir la descripción del indicador objetivamente verificable (IOV)	1 día	16/10/2020	16/10/2020	29
8	31	Exponer el avance del proyecto de fin de carrera (Exposición 5)	1 día	19/10/2020	19/10/2020	30
8	32	Elaborar el reporte de resultados del método seleccionado	6 días	17/10/2020	22/10/2020	30

8	33	Redactar dentro del documento final del proyecto de tesis la descripción del indicador objetivamente verificable (IOV) del resultado esperado 6 (Resultados obtenidos de los métodos elaborados sobre el conjunto de datos de prueba de las proteínas repetidas y la comparación con los resultados de los dos métodos genéricos)	1 día	23/10/2020	23/10/2020	32

8	34	Redactar dentro del documento final del proyecto de tesis el resumen correspondiente al resultado esperado 7 (Aplicación del método seleccionado en todo el conjunto de datos de las proteínas repetidas) sin incluir la descripción del indicador objetivamente verificable (IOV)	1 día	23/10/2020	23/10/2020	33
Objetivo Específico 3 (O3) Avance del 40% restante			**14 días**	**24/10/2020**	**06/11/2020**	**O2**
9	35	Elaborar el repositorio del código fuente del servicio web en Github	3 días	24/10/2020	26/10/2020	34
9	36	Elaborar el informe de pruebas funcionales del servicio web	3 días	27/10/2020	29/10/2020	35

9	37	Redactar dentro del documento final del proyecto de tesis la descripción del indicador objetivamente verificable (IOV) del resultado esperado 7 (Aplicación del método seleccionado en todo el conjunto de datos de las proteínas repetidas)	1 día	30/10/2020	30/10/2020	36

9	38	Redactar dentro del documento final del proyecto de tesis el avance restante del resumen correspondiente al resultado esperado 9 (Servicio web que permita evaluar la diversidad conformacional de las proteínas repetidas o extraer la información de diversidad conformacional de esta clase de proteínas de la base de datos) sin incluir la descripción del indicador objetivamente verificable (IOV)	1 día	30/10/2020	30/10/2020	37
10	39	Exponer el avance del proyecto de fin de carrera (Exposición 6)	1 día	02/11/2020	02/11/2020	38
10	40	Elaborar el repositorio del código fuente de la interfaz de usuario en Github	3 días	31/10/2020	02/11/2020	38
10	41	Elaborar el manual de uso	3 días	03/11/2020	05/11/2020	40

10	42	Redactar dentro del documento final del proyecto de tesis la descripción del indicador objetivamente verificable (IOV) del resultado esperado 9 (Servicio web que permita evaluar la diversidad conformacional de las proteínas repetidas o extraer la información de diversidad conformacional de esta clase de proteínas de la base de datos)	1 día	06/11/2020	06/11/2020	41

10	43	Redactar dentro del documento final del proyecto de tesis el avance restante del resumen correspondiente al resultado esperado 10 (Interfaz de usuario que permita evaluar y visualizar la información de diversidad conformacional de las proteínas repetidas utilizando el servicio web) incluyendo la descripción del indicador objetivamente verificable (IOV)	1 día	06/11/2020	06/11/2020	42
Presentación Final			**49 días**	**09/11/2020**	**27/12/2020**	**O3**
11	44	Presentar el entregable final del proyecto de tesis	7 días	09/11/2020	15/11/2020	43
12	45	Revisión de parte del jurado	7 días	16/11/2020	22/11/2020	44
13	46	Revisión de parte del jurado	7 días	23/11/2020	29/11/2020	45
14	47	Levantar las observaciones recibidas de parte del jurado	7 días	30/11/2020	06/12/2020	46

15	48	Levantar las observaciones recibidas de parte del jurado	7 días	07/12/2020	13/12/2020	47
16	49	Exponer el proyecto de fin de carrera completo ante el jurado (Exposición Final)	7 días	14/12/2020	20/12/2020	48
17	50	Exponer el proyecto de fin de carrera completo ante el jurado (Exposición Final)	7 días	21/12/2020	27/12/2020	49

A.9. Lista de Recursos

A continuación, se detalla la lista de recursos del presente proyecto de tesis. Esta lista contendrá a las personas involucradas, a los materiales requeridos, a los estándares utilizados, a los equipos y herramientas requeridsa. Para esto, también se menciona una breve descripción del recurso, así como, la cantidad y la oportunidad de uso dentro del proyecto de tesis.

1. Personas involucradas

   A continuación, se describen las personas involucradas en el desarrollo del presente proyecto de tesis.

   - Jefe de Proyecto: La Dra. Layla Hirsh actuará como jefe de proyecto. Asimismo, será quien oriente y guíe al tesista Ronaldo Tunque Cahui durante el desarrollo del presente proyecto de tesis.
   - Desarrollador: El tesista Ronaldo Tunque Cahui será quien elabore una herramienta que permita evaluar la diversidad conformacional de las proteínas repetidas.
   - Especialistas: Se cuenta con el apoyo del Dr. Gustavo Parisi y del Dr. Nicolás Palopoli, quienes actuarán como co-asesores y especialistas para la aceptación, seguimiento y orientación de las diversas tareas que se van a realizar para el logro

de los objetivos planteados en el presente proyecto, lo que conlleva a la solución de la problemática planteada.

2. Materiales requeridos para el proyecto

   Para el presente proyecto de tesis, este recurso “No aplica”.

3. Estándares utilizados en el proyecto

   Para el presente proyecto de tesis, este recurso “No aplica”.

4. Equipamiento requerido

   Se detalla la lista de los equipos requeridos para el desarrollo del presente proyecto.

   - Computador: Se hará uso de dos computadoras para el desarrollo de la base de datos, la interfaz de usuario, el servicio web, y entre otras actividades del proyecto de tesis.
   - Servidor: Se utilizará un servidor donde estará alojado la base de datos, la interfaz de usuario y el servicio web del proyecto de tesis.

5. Herramientas requeridas

   Se presenta la lista de las herramientas requeridas para el desarrollo del proyecto.

   - Python
   - RCSB PDB
   - RepeatsDB
   - MySQL Workbench
   - Overleaf
   - Mammoth
   - Flask
   - Git
   - TM-align
   - Postman
   - PyMOL

- Visual Studio Code
- React
- Biopython
- Amazon Web Services
- CoDNaS
- Figma
- CD-HIT
- RStudio

A.10. Costeo del Proyecto

A continuación, se detalla en la Tabla A.6 el costo estimado del proyecto. En esta tabla se describe los costos por el equipo humano, por utilización de equipamiento, por pasajes y viáticos.

Tabla A.6

Costeo del proyecto

Costeo del Proyecto								
Item	**Descripción**	**Unid. 1**	**Cant. 1**	**Unid. 2**	**Cant. 2**	**Valor por unidad (S/.)**	**Monto Parcial (S/.)**	**Monto Total (S/.)**
0	**Costo del proyecto**	-	-	-	-	-	-	**76,200**
1	**Estudiante o tesistas**							**38,150**
1.1	Ronaldo Romario Tunque Cahui	Horas	763	-	-	50	38,150	-
2	**Otros participantes**							**31,150**
2.1	Layla Hirsh	Horas	41	-	-	150	6,150	-
2.2	Gustavo Parisi	Horas	25	-	-	500	12,500	-
2.3	Nicolás Palopoli	Horas	25	-	-	500	12,500	-
3	**Bienes y equipos**							**6,500**

3.1	Computadoras	Equipo	2	-	-	3,000	6,000	-
3.2	Servidores	Equipo	1	Horas	100	5	500	-
4	**Pasajes y viáticos**							**400**
4.1	Movilidad Local	Viajes / Día	2	Día	100	2	400	-

Anexo B

Estudios primarios al utilizar las cadenas de búsqueda

Tabla B.1

Artículos relevantes de la cadena de búsqueda N° 1

Id	Título	Referencia APA	Fuente
DC01	Exploring Conformational Space with Thermal Fluctuations Obtained by Normal-Mode Analysis	Saldaño, T. E., Freixas, V. M., Tosatto, S. C. E., Parisi, G., & Fernandez-Alberti, S. (2020). Ex-ploring Conformational Space with Thermal Fluctuations Obtained by Normal-Mode Analysis. *Journal of Chemical Information and Modeling.* https://doi.org/10.1021/acs.jcim.9b01136	Pubmed
DC02	Exploring Protein Conformational Diversity	Monzon, A., Fornasari, M., Zea, D., & Parisi, G. (2019). Exploring Protein Conformational Diversity. *Methods in Molecular Biology, 1851* , 353–365. https://doi.org/10.1007/978-1-4939-8736-8_20	Scopus

DC03	Large scale analysis of protein conformational transitions from aqueous to non-aqueous media	Rueda, A., Monzon, A., Ardanaz, S., Iglesias, L., & Parisi, G. (2018). Large scale analysis of protein conformational transitions from aqueous to non-aqueous media. *BMC Bioinformatics, 19*(1). https://doi.org/10.1186/s12859-018-2044-2	Scopus
DC04	Conformational diversity analysis reveals three functional mechanisms in proteins	Monzon, A., Zea, D., Fornasari, M., Salda˜no, T., Fernandez-Alberti, S., Tosatto, S., & Parisi, G. (2017). Conformational diversity analysis reveals three functional mechanisms in proteins. *PLoS Computational Biology, 13* (2). https://doi.org/10.1371/journal.pcbi.1005398	Scopus
DC05	Disorder transitions and conformational diversity cooperatively modulate biological function in proteins	Zea, D., Monzon, A., Gonzalez, C., Fornasari, M., Tosatto, S., & Parisi, G. (2016). Disorder transitions and conformational diversity cooperatively modulate biological function in proteins. *Protein Science, 25* (6), 1138–1146. https://doi.org/10.1002/pro.2931	Scopus
DC06	Homology modeling in a dynamical world	Monzon, A., Zea, D., Marino-Buslje, C., & Parisi, G. (2017). Homology modeling in a dynamical world. *Protein Science, 26* (11), 2195–2206. https://doi.org/10.1002/pro.3274	Scopus
DC07	CoDNaS 2.0: A comprehensive database of protein conformational diversity in the native state	Monzon, A., Rohr, C., Fornasari, M., & Parisi, G. (2016). CoDNaS 2.0: A comprehensive database of protein conformational diversity in the native state. *Database, 2016*. https://doi.org/10.1093/database/ baw038	Scopus

DC08	Addressing the role of conformational diversity in protein structure prediction	Palopoli, N., Monzon, A., Parisi, G., & Fornasari, M. (2016). Addressing the role of conformational diversity in protein structure prediction. *PLoS ONE, 11* (5). https://doi.org/10.1371/journal.pone.0154923	Scopus
DC09	Conformational diversity and the emergence of sequence signatures during evolution	Parisi, G., Zea, D., Monzon, A., & Marino-Buslje, C. (2015). Conformational diversity and the emergence of sequence signatures during evolution. *Current Opinion in Structural Biology, 32*, 58–65. https://doi.org/10.1016/j.sbi.2015.02.005	Scopus
DC10	Protein conformational diversity correlates with evolutionary rate	Javier Zea, D., Miguel Monzon, A., Fornasari, M., Marino-Buslje, C., & Parisi, G. (2013). Proteinconformational diversity correlates with evolutionary rate. *Molecular Biology and Evolution, 30* (7), 1500–1503. https://doi.org/10.1093/molbev/mst065	Scopus
DC11	Protein conformational diversity modulates sequence divergence	Juritz, E., Palopoli, N., Fornasari, M., Fernandez-Alberti, S., & Parisi, G. (2013). Protein conformational diversity modulates sequence divergence. *Molecular Biology and Evolution, 30* (1), 79–87. https://doi.org/10.1093/molbev/mss080	Scopus
DC12	Conformational diversity and protein evolution - A 60-year-old hypothesis revisited	James, L., & Tawfik, D. (2003). Conformational diversity and protein evolution - A 60-yearold hypothesis revisited. *Trends in Biochemical Sciences, 28* (7), 361–368. https://doi.org/10.1016/S0968-0004(03)00135-X	Scopus

Tabla B.2

Artículos relevantes de la cadena de búsqueda N° 2

Id	Título	Referencia APA	Fuente
PR01	Tandem repeats in proteins: From sequence to structure	Kajava, A. (2012). Tandem repeats in proteins: From sequence to structure. *Journal of Structural Biology, 179* (3), 279–288. https://doi.org/10.1016/j.jsb.2011.08.009	Scopus
PR02	RepeatsDB: A database of tandem repeat protein structures	Di Domenico, T., Potenza, E., Walsh, I., Gonzalo Parra, R., Giollo, M., Minervini, G., . . . Tosatto, S. (2014). RepeatsDB: A database of tandem repeat protein structures. *Nucleic Acids Research, 42* (D1), D352–D357. https://doi.org/10.1093/nar/gkt1175	Scopus
PR03	Exploring the repeat protein universe through computational protein design	Brunette, T., Parmeggiani, F., Huang, P.-S., Bhabha, G., Ekiert, D., Tsutakawa, S., . . . Baker, D. (2015). Exploring the repeat protein universe through computational protein design. *Nature, 528* (7583), 580–584. https://doi.org/10.1038/nature16162	Scopus
PR04	Protein repeats: Structures, functions, and evolution	Andrade, M., Perez-Iratxeta, C., & Ponting, C. (2001). Protein repeats: Structures, functions, and evolution. *Journal of Structural Biology, 134* (2-3), 117–131. https://doi.org/10.1006/jsbi.2001.4392	Scopus
PR05	Tandem repeats mediating genetic plasticity in health and disease	Hannan, A. (2018). Tandem repeats mediating genetic plasticity in health and disease. *Nature Reviews Genetics, 19* (5), 286–298. https://doi.org/10.1038/nrg.2017.115	Scopus

PR06	Identification of repetitive units in protein structures with ReUPred	Hirsh, L., Piovesan, D., Paladin, L., & Tosatto, S. (2016). Identification of repetitive units in protein structures with ReUPred. *Amino Acids, 48* (6), 1391–1400. https://doi.org/10.1007/s00726-016-2187-2	Scopus
PR07	A New Census of Protein Tandem Repeats and Their Relationship with Intrinsic Disorder	Delucchi, M., Schaper, E., Sachenkova, O., Elofsson, A., & Anisimova, M. (2020). A New Census of Protein Tandem Repeats and Their Relationship with Intrinsic Disorder. *Genes, 11* (4). https://doi.org/10.3390/genes11040407	Scopus
PR08	Comparison of protein repeat classifications based on structure and sequence families	Paladin, L., & Tosatto, S. (2015). Comparison of protein repeat classifications based on structure and sequence families. *Biochemical Society Transactions, 43* , 832–837. https://doi.org/10.1042/BST20150079	Scopus
PR09	TRAL: Tandem repeat annotation library	Schaper, E., Korsunsky, A., Pe˘cerska, J., Messina, A., Murri, R., Stockinger, H., . . . Anisimova, M. (2015). TRAL: Tandem repeat annotation library. *Bioinformatics, 31* (18), 3051–3053. https://doi.org/10.1093/bioinformatics/btv306	Scopus
PR10	A census of protein repeats	Marcotte, E., Pellegrini, M., Yeates, T., & Eisenberg, D. (1999). A census of protein repeats. *Journal of Molecular Biology, 293* (1), 151–160. https://doi.org/10.1006/jmbi.1999.3136	Scopus

Anexo C

Formularios de extracción de datos aplicados a los artículos identificados

Tabla C.1

Formulario de extracción de datos aplicado a los artículos que responden las preguntas N° 1 y N° 2 de investigación

Campo	Descripción	RQ
Id	DC01	General
Fecha de Extracción	27/04/2020	General
Autor(es)	Tadeo E. Saldaño, Victor M. Freixas, Silvio C. E. Tosatto, Gustavo Parisi1, Sebastian Fernandez-Alberti	General
Título	Exploring Conformational Space with Thermal Fluctuations Obtained by Normal-Mode Analysis	General
Tipo de Fuente	Revista	General
Fuente	Journal of Chemical Information and Modeling	General
Año de publicación	2020	General
Afiliación	Universidad Nacional de Quilmes, University of Padova	General
País	Argentina, Itaalia	General

Método(s)	Método que se enfoca en tres pasos: el primer paso es la identificación de la red de interacción de residuos (RIN), el segundo paso es el NMA basado en este RIN y el tercer paso es la generación, selección y optimización de nuevas estructuras de proteínas desplazadas en la dirección de modos NMA seleccionados	RQ1
Resultado(s)	No brinda información	RQ2
Característica(s)	No brinda información	RQ3
Id	DC02	General
Fecha de extracción	27/04/2020	General
Autores	Alexander Miguel Monzon, Maria Silvina Fornasari, Diego Javier Zea, Gustavo Parisi	General
Título	Exploring Protein Conformational Diversity	General
Tipo de Fuente	Revista	General
Fuente	Methods in Molecular Biology	General
Año de publicación	2019	General
Afiliación	Universidad Nacional de Quilmes	General
País	Argentina	General
Método(s)	La diferencias estructurales generalmente se miden por la desviación cuadrática media de la raíz de carbono alfa (RMSD). Para realizar una estimación de la diversidad conformacional, es importante tener en cuenta el número de confórmeros por proteína.	RQ1

Resultado(s)	Con una identidad de alrededor del 100% varias proteínas muestran RMSD tan altas como las alcanzadas por la divergencia de secuencia durante la evolución. Esto significa que la divergencia estructural es un proceso complejo ya que una secuencia dada podría alcanzar varios angstroms de diversidad conformacional.	RQ2
Carácteristica(s)	No brinda información	RQ3
Id	DC03	General
Fecha de extracción	27/04/2020	General
Autores	Ana Julia Velez Rueda, Alexander Miguel Monzon, Sebastián M. Ardanaz, Luis E. Iglesias, Gustavo Parisi	General
Título	Large scale analysis of protein conformational transitions from aqueous to non-aqueous media	General
Tipo de Fuente	Revista	General
Fuente	BMC Bioinformatics	General
Año de publicación	2018	General
Afiliación	Universidad Nacional de Quilmes	General
País	Argentina	General
Método(s)	No brinda información.	RQ1
Resultado(s)	Se encuentra que los conformadores en medios no acuosos tienen mucha menos diversidad conformacional que aquellos en medios acuosos; los conformadores en medios no acuosos también tienen cavidades más grandes, menos superficies expuestas a solventes y menos regiones desordenadas.	RQ2
Carácteristica(s)	No brinda información.	RQ3

Id	DC04	General
Fecha de extracción	27/04/2020	General
Autores	Alexander Miguel Monzon, Diego Javier Zea, María Silvina Fornasari, Tadeo E. Saldaño, Sebastian Fernandez-Alberti, Silvio C. E. Tosatto, Gustavo Parisi	General
Título	Conformational diversity analysis reveals three functional mechanisms in proteins	General
Tipo de Fuente	Revista	General
Fuente	PLoS Computational Biology	General
Año de publicación	2017	General
Afiliación	Universidad Nacional de Quilmes, University of Padova	General
País	Argentina, Italia	General
Método(s)	No brinda información.	RQ1
Resultado(s)	Teniendo una distribución general de RMSD entre confórmeros para todas las cadenas de proteínas contenidas en CoDNaS obtenidas por cristaalografía de rayos X se puede inferir que la mayoría de las proteínas requieren pequeños movimientos entre los confórmeros para cumplir sus funciones biológicas.	RQ2
Carácteristica(s)	No brinda información	RQ3
Id	DC05	General
Fecha de extracción	27/04/2020	General
Autores	Diego Javier Zea, Alexander Miguel Monzon, Claudia Gonzalez, Maria Silvina Fornasari, Silvio C. E. Tosatto, Gustavo Parisi	General

Título	Disorder transitions and conformational diversity cooperatively modulate biological function in proteins	General
Tipo de Fuente	Revista	General
Fuente	Protein Science	General
Año de publicación	2016	General
Afiliación	Universidad Nacional de Quilmes, University of Padova	General
País	Argentina, Italia	General
Método(s)	No brinda información.	RQ1
Resultado(s)	Se ha encontrado que las proteínas que muestran transiciones de desorden de orden entre confórmeros muestran valores de RMSD más altos que las proteínas que no muestran transiciones.	RQ2
Carácteristica(s)	No brinda información.	RQ3
Id	DC06	General
Fecha de extracción	27/04/2020	General
Autores	Alexander Miguel Monzon, Diego Javier Zea, Cristina Marino-Buslje, Gustavo Parisi	General
Título	Homology modeling in a dynamical world	General
Tipo de Fuente	Revista	General
Fuente	Protein Science	General
Año de publicación	2017	General
Afiliación	Universidad Nacional de Quilmes, Fundation Institute Leloir	General
País	Argetina	General
Método(s)	No brinda información.	RQ1

Resultado(s)	Cuando se tiene en cuenta la diversidad conformacional, la relación entre secuencia y divergencia estructural es más compleja.	RQ2
Carácteristica(s)	No brinda información.	RQ3
Id	DC07	General
Fecha de extracción	27/04/2020	General
Autores	Alexander Miguel Monzon, Cristian Oscar Rohr, Maria Silvina Fornasari, Gustavo Parisi	General
Título	CoDNaS 2.0: A comprehensive database of protein conformational diversity in the native state	General
Tipo de Fuente	Revista	General
Fuente	Database	General
Año de publicación	2016	General
Afiliación	Universidad Nacional de Quilmes, Universidad Nacional de Buenos Aires, Instituto de Ecología Genética y Evolución de Buenos Aires	General
País	Argentina	General
Método(s)	CoDNaS permite al usuario explorar cómo cambian la diversidad conformacional entre los confórmeros en función de varias características estructurales que proporcionan más información biológica.	RQ1
Resultado(s)	No brinda información.	RQ2
Carácteristica(s)	No brinda información	RQ3
Id	DC08	General
Fecha de extracción	27/04/2020	General
Autores	Nicolas Palopoli, Alexander Miguel Monzon, Gustavo Parisi, Maria Silvina Fornasari	General

Título	Addressing the role of conformational diversity in protein structure prediction	General
Tipo de Fuente	Revista	General
Fuente	PLoS ONE	General
Año de publicación	2016	General
Afiliación	Universidad Nacional de Quilmes	General
País	Argentina	General
Método(s)	Las diferencias estructurales se miden por el RMSD, ya que es más sensible a los movimientos en bucles y colas.	RQ1
Resultado(s)	No brinda información.	RQ2
Carácteristica(s)	No brinda información	RQ3
Id	DC09	General
Fecha de extracción	27/04/2020	General
Autores	Gustavo Parisi, Diego Javier Zea, Alexander Miguel Monzon, Cristina Marino-Buslje	General
Título	Conformational diversity and the emergence of sequence signatures during evolution	General
Tipo de Fuente	Revista	General
Fuente	Current Opinion in Structural Biology	General
Año de publicación	2015	General
Afiliación	Universidad Nacional de Quilmes, Fundación Insituto Leloir	General
País	Argentina	General

Método(s)	Aunque la diversidad conformacional puede estudiarse utilizando métodos computacionales, como la dinámica molecular y el análisis de modo normal de grano grueso, la evidencia experimental de la diversidad conformacional proviene del análisis de cristales de proteínas y resonancia magnética nuclear (RMN) de proteínas	RQ1
Resultado(s)	No brinda información.	RQ2
Carácteristica(s)	No brinda información	RQ3
Id	DC10	General
Fecha de extracción	27/04/2020	General
Autores	Diego Javier Zea, Alexander Miguel Monzon, Maria Silvina Fornasari, Cristina Marino-Buslje,Gustavo Parisi	General
Título	Protein conformational diversity correlates with evolutionary rate	General
Tipo de Fuente	Revista	General
Fuente	Molecular Biology and Evolution	General
Año de publicación	2013	General
Afiliación	Universidad Nacional de Quilmes, Fundación Insituto Leloir	General
País	Argentina	General
Método(s)	No brinda información.	RQ1
Resultado(s)	Las proteínas con gran diversidad conformacional muestran tasas evolutivas más bajas que las proteínas con confórmeros más similares.	RQ2
Carácteristica(s)	No brinda información.	RQ3
Id	DC11	General

Fecha de extracción	27/04/2020	General
Autores	Ezequiel Juritz, Nicolas Palopoli, Maria Silvina Fornasari, Sebastian Fernandez-Alberti, Gustavo Parisi	General
Título	Protein conformational diversity modulates sequence divergence	General
Tipo de Fuente	Revista	General
Fuente	Molecular Biology and Evolution	General
Año de publicación	2013	General
Afiliación	Universidad Nacional de Quilmes	General
País	Argentina	General
Método(s)	No brinda información.	RQ1
Resultado(s)	Los resultados muestran la relevancia de considerar la diversidad conformacional en nuestra comprensión de los mecanismos de evolución de proteínas y, en consecuencia, revelan la posibilidad de desarrollar mejores modelos evolutivos. Además, la consideración de la diversidad conformacional y su sesgo derivado de sustitución de aminoácidos son aspectos esenciales a tener en cuenta en el desarrollo de nuevas herramientas bioinformáticas.	RQ2
Carácteristica(s)	No brinda información.	RQ3
Id	DC12	General
Fecha de extracción	27/04/2020	General
Autores	Leo C. James,Dan S. Tawfik	General
Título	Conformational diversity and protein evolution - A 60-year-old hypothesis revisited	General
Tipo de Fuente	Revista	General
Fuente	Trends in Biochemical Sciences	General

Año de publicación	2003	General
Afiliación	Centre for Protein Engineering, The Weizmann Institute of Science	General
País	United King, Israel	General
Método(s)	La cinética de unión en estado preestable proporcionó los primeros datos que indicaban isómeros preexistentes en equilibrio, es decir, reveló la existencia de la diversidad conformacional. Sin embargo, es inalcanzable en muchos casos.	RQ1
Resultado(s)	No brinda información.	RQ2
Carácteristica(s)	No brinda información.	RQ3

Tabla C.2

Formulario de extracción de datos aplicado a los artículos que responden la pregunta N° 3 de investigación

Campo	Descripción	RQ
Id	PR01	General
Fecha de Extracción	27/04/2020	General
Autores	Kajava, A.V.	General
Título	Tandem repeats in proteins: From sequence to structure	General
Tipo de Fuente	Revista	General
Fuente	Journal of Structural Biology	General
Año de publicación	2012	General
Afiliación	Université Montpellier	General
País	Francia	General
Método(s)	No brinda información	RQ1
Resultado(s)	No brinda información	RQ2

Característica(s)	Las proteínas repetidas muestran una diversa variedad de tamaños, estructuras y funciones. Estas son una clase de proteína generalizada que prevalece en eucariotas, pero que también está presente en bacterias y arqueas. Por otro lado, la clasificación de las proteínas repetidas se basan en la longitud de la unidad repetida y están divididos en cinco clases.	RQ3
Id	PR02	General
Fecha de extracción	27/04/2020	General
Autores	Di Domenico, T. y Potenza, E. y Walsh, I. y Gonzalo Parra, R. y Giollo, M. y Minervini, G. y Piovesan, D. y Ihsan, A. y Ferrari, C. y Kajava, A.V. y Tosatto, S.C.E.	General
Título	RepeatsDB: A database of tandem repeat protein structures	General
Tipo de Fuente	Revista	General
Fuente	Nucleic Acids Research	General
Año de publicación	2014	General
Afiliación	University of Padua, Universidad de Buenos Aires, Institute of Information Technology, Institut de Biologie Computationnelle	General
País	Italia, Argentina, Pakistan, Francia	General
Método(s)	No brinda información.	RQ1
Resultado(s)	No brinda información.	RQ2

Carácteristica(s)	Las proteínas repetidas presentan regiones repetidas. Una región repetida es un grupo de al menos tres unidades de repetición; y una unidad repetida es definida como el bloque de construcción estructural más pequeño que forma una región repetida. Además, esta región puede incluir inserciones, es decir, segmentos de estructuras no repetidas que ocurren dentro de una unidad de repetición o entre dos de ellas.	RQ3
Id	PR03	General
Fecha de extracción	27/04/2020	General
Autores	Brunette, T.J. y Parmeggiani, F. y Huang, P.-S. y Bhabha, G. y Ekiert, D.C. y Tsutakawa, S.E. y Hura, G.L. y Tainer, J.A. y Baker, D.	General
Título	Exploring the repeat protein universe through computational protein design	General
Tipo de Fuente	Revista	General
Fuente	Nature	General
Año de publicación	2015	General
Afiliación	University of Washington, Universidad de California en San Francisco, University of Texas	General
País	Estados Unidos	General
Método(s)	No brinda información.	RQ1
Resultado(s)	No brinda información.	RQ2
Carácteristica(s)	Las repeticiones de tándem en proteínas son utilizadas para aplicaciones de ingeniería.	RQ3
Id	PR04	General
Fecha de extracción	27/04/2020	General

Autores	Andrade, M.A. y Perez-Iratxeta, C. y Ponting, C.P.	General
Título	Protein repeats: Structures, functions, and evolution	General
Tipo de Fuente	Revista	General
Fuente	Journal of Structural Biology	General
Año de publicación	2001	General
Afiliación	European Molecular Biology Laboratory, University of Oxford	General
País	Alemania, Reino Unido	General
Método(s)	No brinda información.	RQ1
Resultado(s)	No brinda información.	RQ2
Carácteristica(s)	Las repeticiones de tándem en proteínas son comunes en la naturaleza, por lo que se pueden encontrar de varias formas y son difíciles de reconocer, porque la unidad repetida es relativamente corta y puede haber una considerable divergencia de secuencia entre las unidades de la misma TR.	RQ3
Id	PR05	General
Fecha de extracción	27/04/2020	General
Autores	Hannan, A.J.	General
Título	Tandem repeats mediating genetic plasticity in health and disease	General
Tipo de Fuente	Revista	General
Fuente	Nature Reviews Genetics	General
Año de publicación	2018	General
Afiliación	University of Melbourn	General
País	Australia	General
Método(s)	No brinda información.	RQ1

Resultado(s)	No brinda información.	RQ2
Carácteristica(s)	Las repeticiones en tándem están asociadas con funciones y enfermedades relacionadas con la inmunidad. Además, las proteínas repetidas tienen relevancia en los último años en la salud.	RQ3
Id	PR06	General
Fecha de extracción	27/04/2020	General
Autores	Hirsh, L. y Piovesan, D. y Paladin, L. y Tosatto, S.C.E.	General
Título	Identification of repetitive units in protein structures with ReUPred	General
Tipo de Fuente	Revista	General
Fuente	Amino Acids	General
Año de publicación	2016	General
Afiliación	University of Padua, Pontificia Universidad Católica del Perú, CNR Institute of Neuroscience	General
País	Perú, Italia	General
Método(s)	No brinda información.	RQ1
Resultado(s)	No brinda información.	RQ2
Carácteristica(s)	Las proteínas repetidas pertenecen a la "materia oscura" del universo proteico que se caracteriza por relaciones no canónicas de secuencia-estructura.	RQ3
Id	PR07	General
Fecha de extracción	27/04/2020	General
Autores	Delucchi, M. y Schaper, E. y Sachenkova, O. y Elofsson, A. y Anisimova, M.	General

Título	A New Census of Protein Tandem Repeats and their Relationship with Intrinsic Disorder	General
Tipo de Fuente	Revista	General
Fuente	Genes	General
Año de publicación	2020	General
Afiliación	Swiss Institute of Bioinformatics, Stockholm University	General
País	Suecia, Suiza	General
Método(s)	No brinda información.	RQ1
Resultado(s)	No brinda información.	RQ2
Carácteristica(s)	Las repeticiones en tándem (TR) abundan en gran cantidad en el proteoma humano y se estima que el 50.9% de las proteínas contienen al menos un TR, el cual suele estar ubicado en los flancos de secuencia.	RQ3
Id	PR08	General
Fecha de extracción	27/04/2020	General
Autores	Paladin, L. y Tosatto, S.C.E.	General
Título	Comparison of protein repeat classifications based on structure and sequence families	General
Tipo de Fuente	Revista	General
Fuente	Biochemical Society Transactions	General
Año de publicación	2015	General
Afiliación	University of Padua	General
País	Itali	General
Método(s)	No brinda información.	RQ1
Resultado(s)	No brinda información.	RQ2

Carácteristica(s)	Las repeticiones en tándem se pueden encontrar de varias formas, por lo que son difíciles de reconocer. Además, las proteínas repetidas realizan funciones únicas de las eucariotas.	RQ3
Id	PR09	General
Fecha de extracción	27/04/2020	General
Autores	Schaper, E. y Korsunsky, A. y Pečerska, J. y Messina, A. y Murri, R. y Stockinger, H. y Zoller, S. y Xenarios, I. y Anisimova, M.	General
Título	TRAL: Tandem repeat annotation library	General
Tipo de Fuente	Revista	General
Fuente	Bioinformatics	General
Año de publicación	2015	General
Afiliación	Swiss Institute of Bioinformatics, ETH Zürich, Graz University of Technology, University of Zürich, Zürich University of Applied Sciences	General
País	Suiza, Austria	General
Método(s)	No brinda información.	RQ1
Resultado(s)	No brinda información.	RQ2
Carácteristica(s)	Las repeticiones en tándem se describen por el número de unidades repetidas, la longitud del patrón repetido, y la similitud entre sus unidades.	RQ3
Id	PR10	General
Fecha de extracción	27/04/2020	General
Autores	Marcotte, E.M. y Pellegrini, M. y Yeates, T.O. y Eisenberg, D.	General
Título	A census of protein repeats	General
Tipo de Fuente	Revista	General

Fuente	Journal of Molecular Biology	General
Año de publicación	1999	General
Afiliación	University of California	General
País	Estados Unidos	General
Método(s)	No brinda información.	RQ1
Resultado(s)	No brinda información.	RQ2
Carácteristica(s)	Se encuentra que las repeticiones de proteínas eucariotas tienen poca similitud con las repeticiones procariotas, lo que sugiere que la mayoría de las repeticiones surgieron después de que los linajes procariotas y eucariotas divergieran. Además, se estima que el 14% de las proteínas contienen al menos una repetición en tándem.	RQ3

Anexo D

Informe de la estructura de datos organizada

D.1. Introducción

La estructura de datos organizada es la estructura que va a representar al conjunto de datos de las proteínas repetidas que se van a utilizar como datos de entrada para el análisis de diversidad conformacional; asimismo, va a representar al conjunto de datos de prueba que servirán para evaluar la efectividad de los 4 métodos (2 métodos genéricos y 2 propuestas de métodos) que analizarán la diversidad conformacional de las proteínas repetidas.

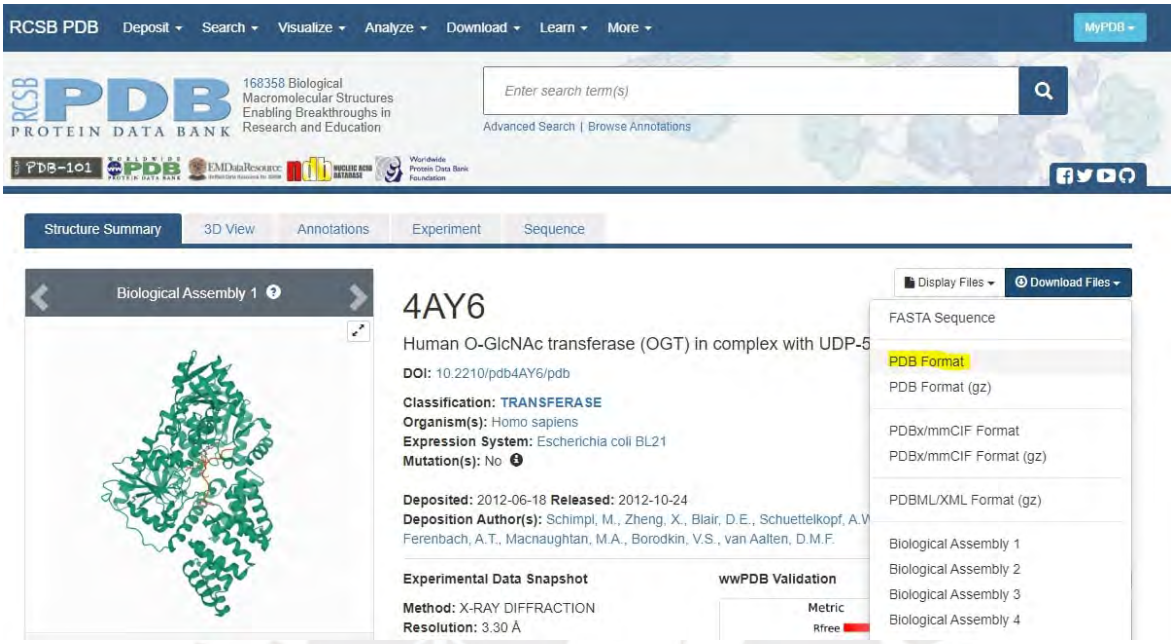
D.2. Definición de la estructura de datos

Para definir esta estructura de datos organizada, primero se tuvo que acceder a la base de datos RCSB PDB, ya que es donde se encuentra la información estructural de las proteínas. Luego, se tomó como ejemplo la proteína 4AY6, el cual es una proteína repetida cualquiera que la base de datos RCSB PDB nos proporciona, con la finalidad de descargar el archivo 4ay6.pdb (pdb file) que contiene la información de esta proteína (Ver Figura D.1).

Después, teniendo descargado el archivo 4ay6.pdb (Ver Figura D.2) se procedió a revisar la información del mismo y se optó por utilizar la información de la sección de coordenadas atómicas (Ver Figura D.3), ya que esta se usa para identificar a las proteínas. Esta sección contiene una lista de registros ATOM (Ver Figura D.3) por cada cadena que tiene la proteína y contiene registros TER (Ver Figura D.4) que indican el final de una lista de registros ATOM.

Figure D.1

Descarga de la proteína 4AY6



Nota: El gráfico muestra una captura de pantalla de la base de datos RSCB PDB (National Science Foundation et al., 2020b) donde se procede a realizar la descarga del archivo de la proteína 4AY6 en formato PDB. Elaboración Propia.

Figure D.2

PDB File de la proteína 4AY6

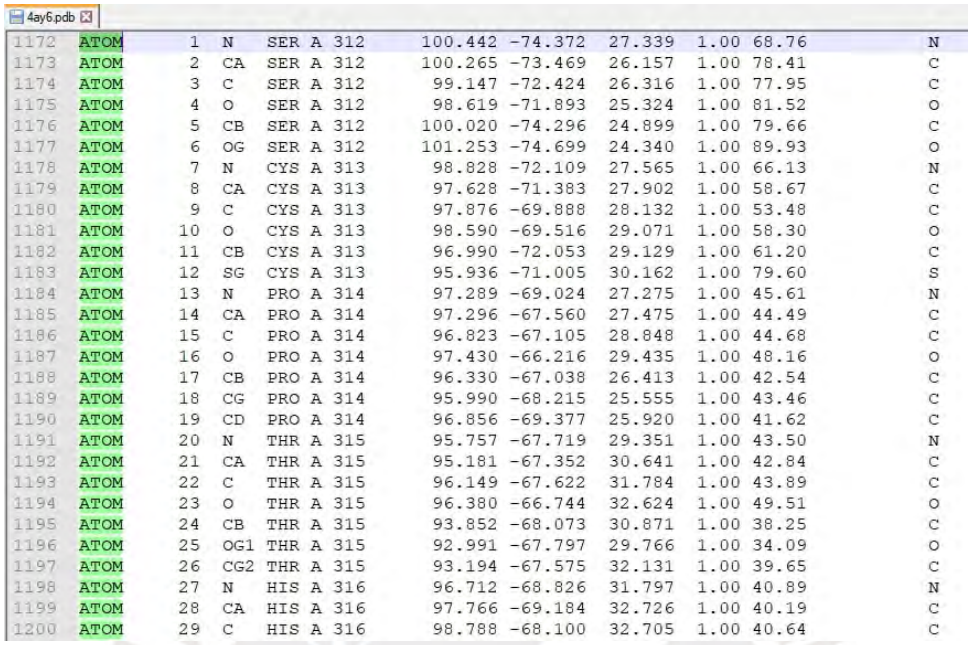
4ay6.pdb

```
HEADER    TRANSFERASE                             18-JUN-12   4AY6
TITLE     HUMAN O-GLCNAC TRANSFERASE (OGT) IN COMPLEX WITH UDP-5SGLCNAC
TITLE    2 AND SUBSTRATE PEPTIDE
COMPND    MOL_ID: 1;
COMPND   2 MOLECULE: UDP-N-ACETYLGLUCOSAMINE--PEPTIDE
COMPND   3  N-ACETYLGLUCOSAMINYLTRANS FERASE 110 KDA SUBUNIT;
COMPND   4 CHAIN: A, B, C, D;
COMPND   5 FRAGMENT: TPR (TRUNCATED) AND CATALYTIC DOMAIN, RESIDUES 197-915;
COMPND   6 SYNONYM: 2.4.1.255, O-GLCNAC TRANSFERASE SUBUNIT P110, O-LINKED
COMPND   7  N-ACETYLGLUCOSAMINE TRANSFERASE 110 KDA SUBUNIT, OGT;
COMPND   8 EC: 2.4.1.255;
COMPND   9 ENGINEERED: YES;
COMPND  10 MOL_ID: 2;
COMPND  11 MOLECULE: TGF-BETA-ACTIVATED KINASE 1 AND MAP3K7-BINDING PROTEIN 1;
COMPND  12 CHAIN: E, F, G, H;
COMPND  13 FRAGMENT: RESIDUES 389-401;
COMPND  14 SYNONYM: MITOGEN-ACTIVATED PROTEIN KINASE KINASE KINASE
COMPND  15  7-INTERACTING PROTEIN 1, TGF-BETA-ACTIVATED KINASE 1-BINDING
COMPND  16  PROTEIN 1, TAK1-BINDING PROTEIN 1;
COMPND  17 ENGINEERED: YES
```

Nota: El gráfico muestra una captura de pantalla del PDB file de la proteína 4AY6 descargada de la base de datos RCSB PDB (National Science Foundation et al., 2020b).

Figure D.3

Sección de coordenadas atómicas de la proteína 4AY6



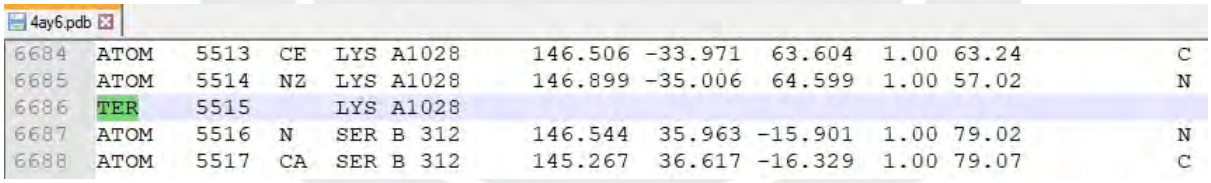
4ay6.pdb

1172	ATOM	1	N	SER A 312	100.442	-74.372	27.339	1.00	68.76	N
1173	ATOM	2	CA	SER A 312	100.265	-73.469	26.157	1.00	78.41	C
1174	ATOM	3	C	SER A 312	99.147	-72.424	26.316	1.00	77.95	C
1175	ATOM	4	O	SER A 312	98.619	-71.893	25.324	1.00	81.52	O
1176	ATOM	5	CB	SER A 312	100.020	-74.296	24.899	1.00	79.66	C
1177	ATOM	6	OG	SER A 312	101.253	-74.699	24.340	1.00	89.93	O
1178	ATOM	7	N	CYS A 313	98.828	-72.109	27.565	1.00	66.13	N
1179	ATOM	8	CA	CYS A 313	97.628	-71.383	27.902	1.00	58.67	C
1180	ATOM	9	C	CYS A 313	97.876	-69.888	28.132	1.00	53.48	C
1181	ATOM	10	O	CYS A 313	98.590	-69.516	29.071	1.00	58.30	O
1182	ATOM	11	CB	CYS A 313	96.990	-72.053	29.129	1.00	61.20	C
1183	ATOM	12	SG	CYS A 313	95.936	-71.005	30.162	1.00	79.60	S
1184	ATOM	13	N	PRO A 314	97.289	-69.024	27.275	1.00	45.61	N
1185	ATOM	14	CA	PRO A 314	97.296	-67.560	27.475	1.00	44.49	C
1186	ATOM	15	C	PRO A 314	96.823	-67.105	28.848	1.00	44.68	C
1187	ATOM	16	O	PRO A 314	97.430	-66.216	29.435	1.00	48.16	O
1188	ATOM	17	CB	PRO A 314	96.330	-67.038	26.413	1.00	42.54	C
1189	ATOM	18	CG	PRO A 314	95.990	-68.215	25.555	1.00	43.46	C
1190	ATOM	19	CD	PRO A 314	96.856	-69.377	25.920	1.00	41.62	C
1191	ATOM	20	N	THR A 315	95.757	-67.719	29.351	1.00	43.50	N
1192	ATOM	21	CA	THR A 315	95.181	-67.352	30.641	1.00	42.84	C
1193	ATOM	22	C	THR A 315	96.149	-67.622	31.784	1.00	43.89	C
1194	ATOM	23	O	THR A 315	96.380	-66.744	32.624	1.00	49.51	O
1195	ATOM	24	CB	THR A 315	93.852	-68.073	30.871	1.00	38.25	C
1196	ATOM	25	OG1	THR A 315	92.991	-67.797	29.766	1.00	34.09	O
1197	ATOM	26	CG2	THR A 315	93.194	-67.575	32.131	1.00	39.65	C
1198	ATOM	27	N	HIS A 316	96.712	-68.826	31.797	1.00	40.89	N
1199	ATOM	28	CA	HIS A 316	97.766	-69.184	32.726	1.00	40.19	C
1200	ATOM	29	C	HIS A 316	98.788	-68.100	32.705	1.00	40.64	C

Nota: El gráfico presenta la sección de coordenadas atómicas de la proteína 4AY6 extraídas del archivo 4ay6.pdb. Elaboración propia.

Figure D.4

Registro TER



4ay6.pdb

6684	ATOM	5513	CE	LYS A1028	146.506	-33.971	63.604	1.00	63.24	C
6685	ATOM	5514	NZ	LYS A1028	146.899	-35.006	64.599	1.00	57.02	N
6686	TER	5515		LYS A1028						
6687	ATOM	5516	N	SER B 312	146.544	35.963	-15.901	1.00	79.02	N
6688	ATOM	5517	CA	SER B 312	145.267	36.617	-16.329	1.00	79.07	C

Nota: El gráfico presenta el registro TER, el cual indica el final de una lista de registro ATOM para una cadena de la proteína 4AY6, extraída del archivo 4ay6.pdb (pdb file). Elaboración propia.

Por ello, la estructura de datos organizada está definida por los parámetros del registro ATOM (Ver Tabla D.1) y del registro TER (Ver Tabla D.2).

Tabla D.1

Estructura de datos organizado del registro ATOM

#	Tipo de Dato	Contenido
1	Nombre de registro	“ATOM”
2	Entero	Número secuencial del átomo
3	Átomo	Nombre del átomo
4	Caracter	Indicador de ubicación alternativa
5	Nombre del residuo	Nombre del residuo
6	Caracter	Identificador de la cadena
7	Entero	Número de la secuencia de residuo
8	Real	Coordenada X del átomo de la proteína
9	Real	Coordenada Y del átomo de la proteína
10	Real	Coordenada Z del átomo de la proteína
11	Real	Factor de ocupación
12	Real	Temperatura

Tabla D.2

Estructura de datos organizado del registro TER

#	Tipo de Dato	Contenido
1	Nombre de registro	”TER”
2	Entero	Número secuencial del átomo
3	Nombre del residuo	Nombre del residuo
4	Caracter	Identificador de la cadena
5	Entero	Número de la secuencia del residuo

D.3. Verificación de la estructura de datos

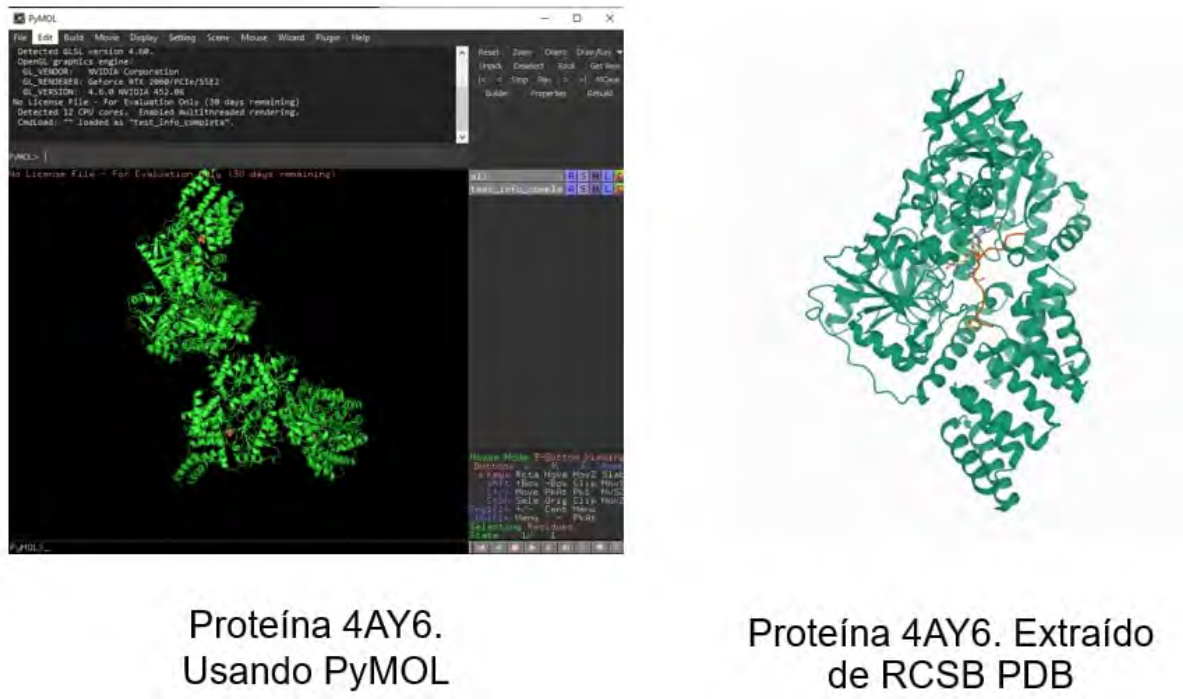
Para verificar que la estructura definida es la conveniente, se elaboraron 2 archivos de prueba, uno con la información de la proteína correspondiente a la estructura de datos descrita en la sección anterior, llamada test_info_completa.pdb y la otra con información incompleta,

llamada test_info_incompleta.pdb. Esto se hizo con la finalidad de mostrar la necesidad de cubrir todos los parámetros definidos en la estructura de datos para que se pueda observar de manera correcta la estructura de la proteína repetida.

Teniendo los 2 archivos ya generados, se procedió a usarlos como datos de entrada en el software PyMOL, el cual nos permite ver la imagen 3D de la estructura de una proteína y los resultados fueron los siguientes. Para el archivo test_info_completa.pdb la estructura de la proteína es igual a la que se puede observar en la base de datos RCSB PDB (Ver Figura D.5). Sin embargo, para el archivo test_info_incompleta.pdb la estructura está muy alejado a ser similar con la proteína extraída de la base de datos RCSB PDB (Ver Figura D.6).

Figure D.5

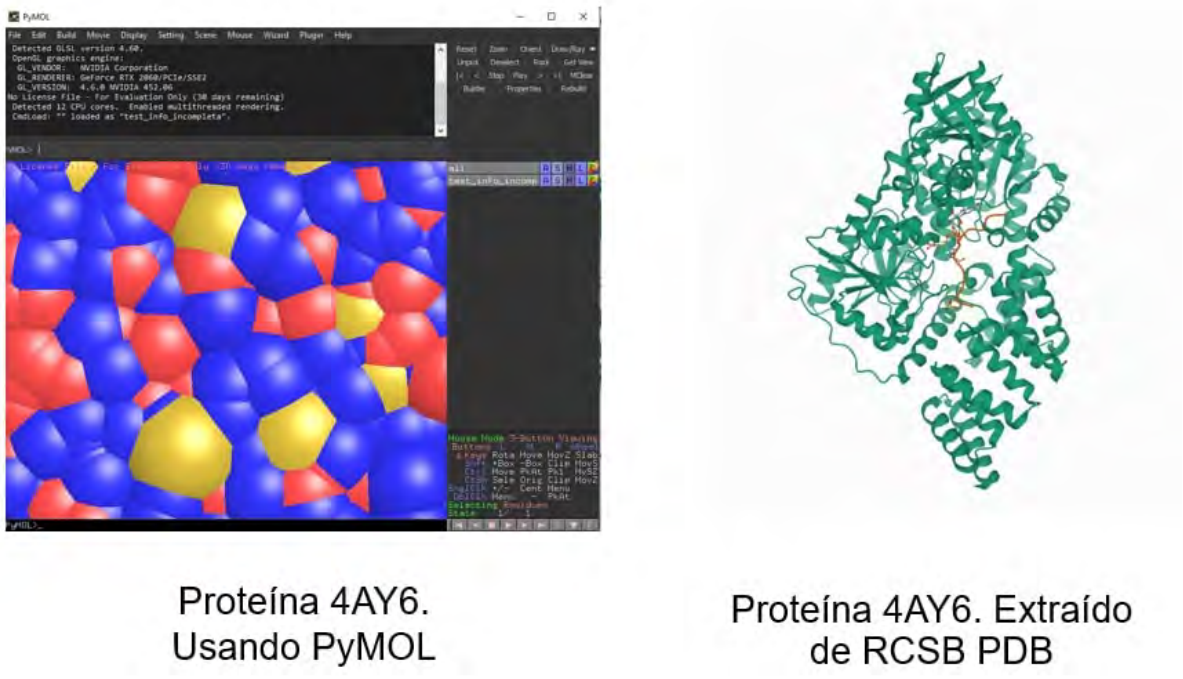
PDB File, basada en la estructura de datos descrita, completo



Nota: El gráfico muestra la comparación entre la estructura de la proteína 4AY6 basada en el test_info_completa.pdb usando el software PyMOL y la estructura de la misma proteína, pero extraída de la base de datos RCSB PDB. Elaboración propia.

Figure D.6

PDB File, basada en la estructura de datos descrita, incompleto



Nota: El gráfico muestra la comparación entre la estructura de la proteína 4AY6 basada en el test_info_incompleta.pdb usando el software PyMOL y la estructura de la misma proteína, pero extraída de la base de datos RCSB PDB. Elaboración propia.

Anexo E

Reporte del conjunto de datos de proteínas repetidas

E.1. Introducción

El presente reporte detalla la generación del conjunto de datos de proteínas repetidas, el cual será utilizado como dato de entrada para el análisis de diversidad conformacional. Posteriormente describe la verificación de este conjunto de datos por medio del software PyMOL. Y finalmente, se presenta una tabla con la cantidad de archivos en formato pdb (pdb files) que se generaron a partir de un script de autoría propia.

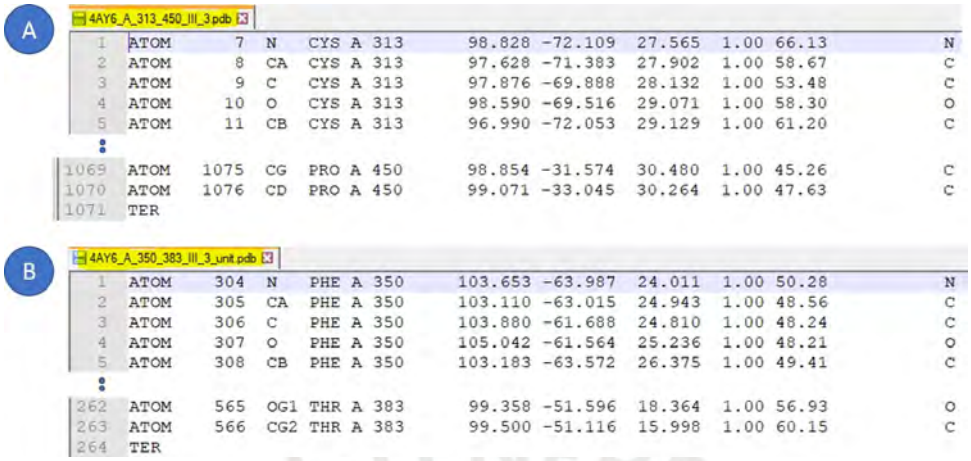
E.2. Generación del conjunto de datos

Para generar el conjunto de datos se elaboró un script usando python como lenguaje de programación y utlizando las bases de datos RepeatsDB y RCSB PDB, ya que contienen la información estructural de la región repetida y de las unidades de repetición de las proteínas repetidas; y la información estructural de las proteínas, respectivamente. Asimismo, se utilizó la librería biopython para poder acceder a la base de datos RCSB PDB.

Además, este script está basado en una secuencia de pasos manuales que permite generar el archivo en formato pdb (pdb file) de la región repetida y los pdb files de las unidades de repetición de una proteína repetida cualquiera (Ver Figura E.1).

Figure E.1

PDB File de la región repetida y de la unidad de repetición de la proteína 4AY6



Nota: A: El pdb file 4ay6_A_313_450_III_3.pdb representa la estructura de la región repetida de la proteína 4AY6. B: El pdb file 4ay6_A_313_349_III_3_unit.pdb representa la estructura de la unidad de repeticón de la proteína 4AY6. Elaboración Propia.

Esta secuencia de pasos manuales consiste en acceder a la base de datos RepeatsDB para descargar el archivo en formato db (db file) de una proteína repetida cualquiera (Ver Figura E.2). En esta oportunidad, se eligió, como ejemplo, la proteína repetida 4ay6A. Teniendo el db file de esta proteína repetida se procede a identificar la sección que detalla el inicio y fin de la región repetida y de las unidades de repetición (Ver Figura E.3). Después, se debe acceder a la base de datos RCDB PDB para descargar el pdb file de la proteína 4AY6 y teniendo este pdb file, se debe identificar la sección de coordenadas atómicas para poder segmentar esta información con respecto a la sección de la región repetida y de las unidades de repetición ya identificadas en el db file para generar sus respectivos pdb files.

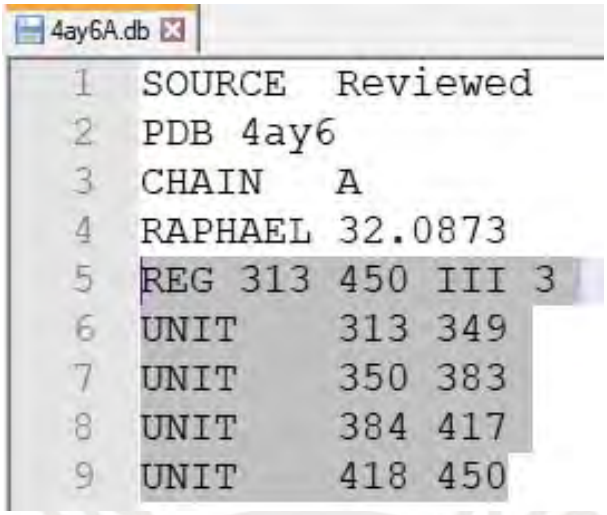
Figure E.2

Descarga de la proteína repetida 4ay6A

Nota: El gráfico muestra una captura de pantalla de la base de datos RepeatsDB (Di Domenico et al., 2014) donde se procede a realizar la descarga del archivo de la proteína repetida 4ay6A en formato DB. Elaboración Propia.

Figure E.3

Archivo en formato db de la proteína repetida 4ay6A

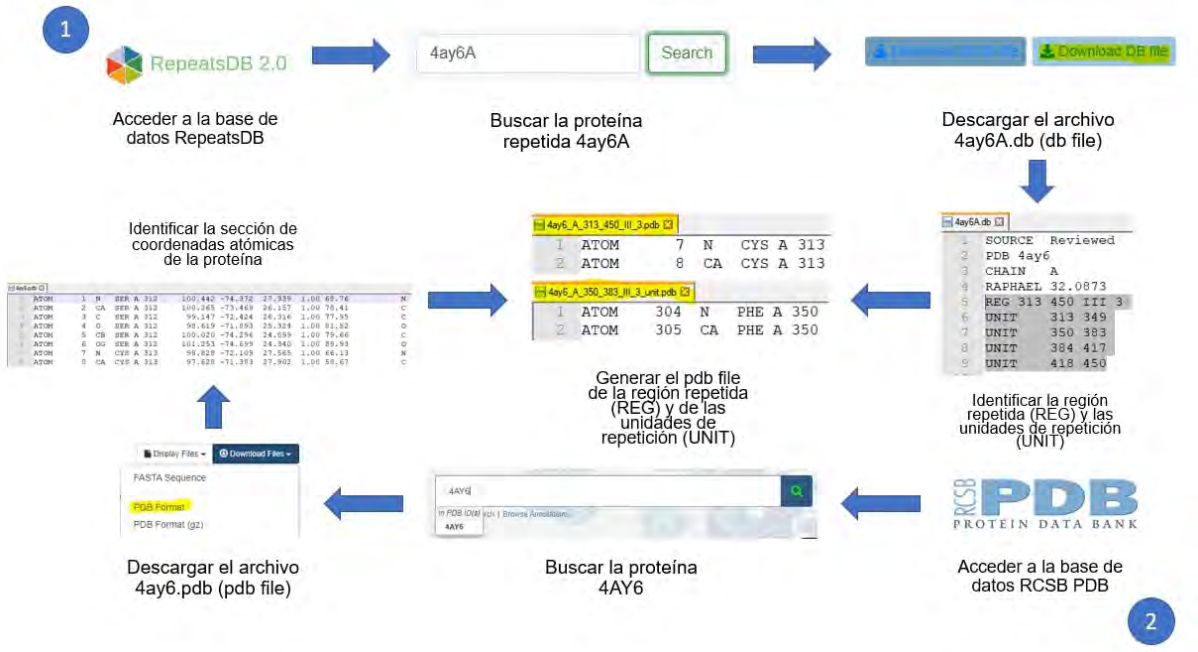


Nota: El gráfico muestra la sección de la región repetida (REG) y de las unidades de repetición (UNIT) de la proteína repetida 4ay6A. Elaboración Propia.

A modo de resumen, lo descrito en el párrafo anterior se puede apreciar en la Figura E.4.

Figure E.4

Pasos para generar los pdb files de la región repetida y unidades de repetición

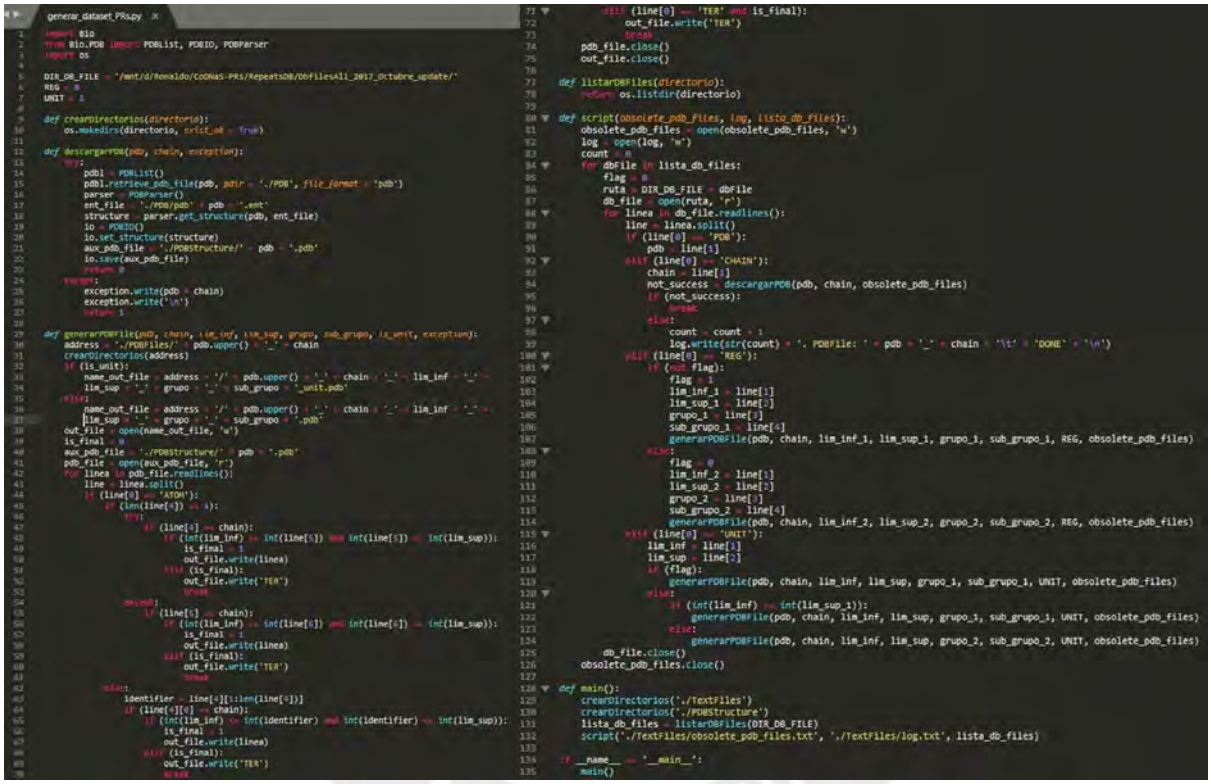


Nota: El gráfico muestra los pasos que se siguieron para generar el pdb file de la región repetida y los pdb files de las unidades de repetición de la proteína 4AY6. Elaboración Propia.

Por otro lado, dado que la base de datos RepeatsDB cuenta con la información estructural de las regiones repetidas de 6410 proteínas, realizar la secuencia manual para cada proteína repetida tendría un tiempo estimado muy alto. Por tal motivo, se elaboró un script (Ver Figura E.5) que realiza los pasos de la Figura E.4 de manera automática para todas las proteínas repetidas con la única diferencia que en lugar de descargar de db file en db file, se descarga un zip que contiene todos los db files y esto lo proporciona la misma base de datos RepeatsDB.

Figure E.5

Script para generar los pdb files de la región repetida y unidades de repetición



Nota: El gráfico muestra el script de elaboración propia que se utilizó para generar el pdb file de la región repetida y los pdb files de las unidades de repetición de las proteínas repetidas. Elaboración Propia.

E.3. Verificación del conjunto de datos

Para verificar que el conjunto de datos de proteínas repetidas es el adecuado, se elaboró una lista de imágenes con la ayuda del software PyMOL, el cual permite visualizar las estructuras de las proteínas como imagen 3D y permite exportarlas como imagen PNG. Cabe mencionar que cada archivo del conjunto de datos corresponde a una imagen de esta lista elaborada.

Luego, teniendo la lista de imágenes, se procedió a compararlas con las imágenes generadas por el software PyMOL usando como dato de entrada los pdb files de las proteínas correspondientes extraídas de la base de datos RCSB PDB. Esto con la finalidad de validar el conjunto de datos de proteínas repetidas. Ejemplo de esto, se puede apreciar en la Figura E.6.

Figure E.6

Imagen basada en el pdb file generado por el script de autoría propio vs Imagen basada en el pdb file extraído de la base de datos RCSB PDB

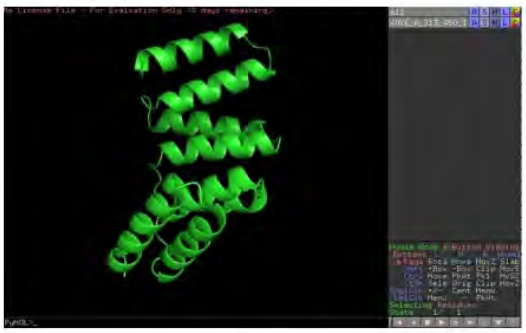


Imagen de la región repetida de la proteína 4AY6 generado por PyMOL usando como dato de entrada el pdb file generado por el script de autoría propia

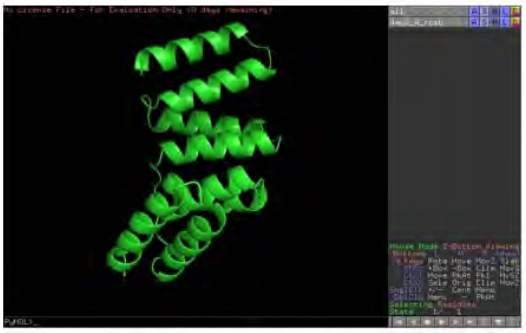


Imagen de la región repetida de la proteína 4AY6 generado por PyMOL usando como dato de entrada el pdb file extraído de la base de datos RCSB PDB

Nota: El gráfico muestra la comparación entre la estructura de la región repetida de la proteína 4AY6 basada en el pdb file generado por el script de autoría propia usando el software PyMOL y la estructura de la región repetida de la misma proteína, pero basada en el pdb file extraído de la base de datos RCSB PDB. Elaboración propia.

Finalmente, esta verificación ha sido validada al 100% por la Dra. Layla Hirsh, experta en el tema de las proteínas repetidas; y el Dr. Nicolás Palopoli, experto en el tema de diversidad conformacional en proteínas.

E.4. Resultados

A continuación, en la Tabla E.1 se presenta el total de pdb files de la región repetida y de las unidades de repetición de las proteínas repetidas generados por el script de autoría propia (Ver Figura E.5).

Tabla E.1

Cantidad de PDB Files generados de la región repetida y de las unidades de repetición de las proteínas repetidas

Clase	**Descripción**	**Cantidad de cadenas de proteínas repetidas**	**PDB Files de la región repetida**	**PDB Files de las unidades de repetición**
II	Estructura fibrosas	15	15	78
III	Estructuras alargadas	2503	2503	12672
IV	Estructuras cerradas	3595	3595	18438
V	Estructuras “Beads on a String”	216	216	1092
Total		**6329**	**6329**	**32280**

Anexo F

Reporte del conjunto de datos de prueba de proteínas repetidas

F.1. Introducción

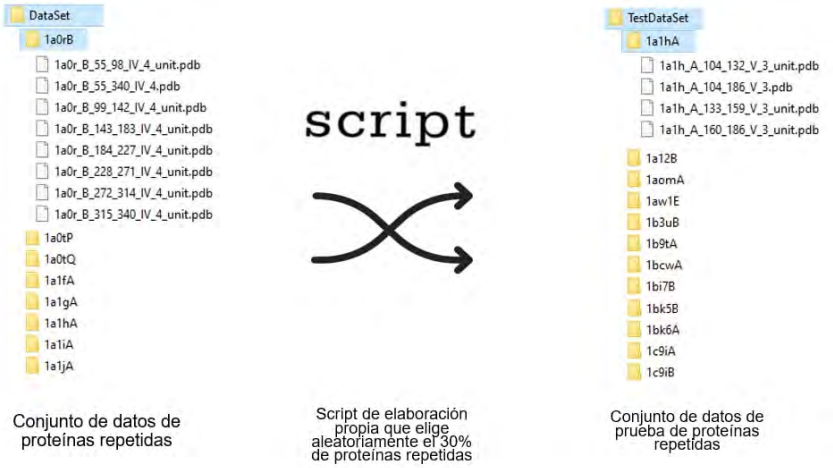
El presente reporte describe la generación del conjunto de datos de prueba de proteínas repetidas, el cual será utilizado para evaluar la efectividad de los 4 métodos (2 métodos genéricos y 2 propuestas de métodos) que analizarán la diversidad conformacional de las proteínas repetidas. Posteriormente, se describe la verificación de este conjunto de datos de prueba con la ayuda del software PyMOL. Y finalmente, se presenta una tabla con la cantidad de archivos en formato pdb (pdb files) que se generaron aleatoriamente teniendo como base al conjunto de datos de proteínas repetidas.

F.2. Generación del conjunto de datos de prueba

Para generar el conjunto de datos de prueba de proteínas repetidas se tuvo que escoger, de manera aleatoria, el 30% del total de cadenas de proteínas repetidas que se utilizaron para generar el conjunto de datos de proteínas repetidas que servirá como dato de entrada para el análisis de diversidad conformacional (Ver Figura F.1).

Figure F.1

Pasos para generar el conjunto de datos de prueba de proteínas repetidas



Nota: El gráfico muestra los pasos que se siguieron para generar el conjunto de datos de prueba de proteínas repetidas. Elaboración Propia.

Es así que, se elaboró un script que permite generar aleatoriamente este conjunto de datos de prueba y este script se puede apreciar en la Figura F.2.

Figure F.2

Script para generar aleatoriamente el conjunto de datos de prueba

generar_test_dataset_PRs.py

```python
import os
import random
import shutil
import subprocess

N = 1900
DIR_SOURCE = '/mnt/d/Ronaldo/CoDNaS-PRs/DataSet/All/PDBFiles'
DIR_TARGET = '/mnt/d/Ronaldo/CoDNaS-PRs/DataSet/Test/PDBFiles'

def crearDirectorios(directorio):
    os.makedirs(directorio, exist_ok = True)

def main():
    crearDirectorios('./PDBFiles')
    src_list = []
    #Listar
    print("LISTANDO ARCHIVOS...")
    for directorio in os.listdir(DIR_SOURCE):
        fullpath = os.path.join(DIR_SOURCE, directorio)
        if (os.path.isdir(fullpath)):
            src_list.append([directorio, fullpath])
    print("LISTADO SATISFACTORIO")
    #Copiar los archivos
    print("COPIANDO ARCHIVOS...")
    for file in random.sample(src_list, N):
        shutil.copytree(file[1], os.path.join(DIR_TARGET, file[0]))
        src_list.remove(file)
    print("GENERACIÓN SATISFACTORIA")

if __name__ == '__main__':
    main()
```

Nota: El gráfico muestra el script de elaboración propia que se utilizó para generar, de manera aleatoria, el conjunto de datos de prueba de proteínas repetidas. Elaboración Propia.

F.3. Verificación del conjunto de datos de prueba

Para verificar que el conjunto de datos de prueba de proteínas repetidas es el adecuado, se preparó una lista de imágenes utilizando el software PyMOL, el cual permite ver la estructura de una proteína como imagen 3D y tiene la opción de exportar esta estructura como imagen PNG. Cabe mencionar que cada archivo del conjunto de datos de prueba corresponde a una imagen de esta lista preparada.

Después, teniendo la lista de imágenes, se procedió a comparar estas imágenes generadas del conjunto de datos de prueba con las imágenes generadas por el software PyMOL usando como dato de entrada los pdb files de las proteínas correspondientes extraídas de la base de datos RCSB PDB. Esto con la finalidad de comprobar que el conjunto de datos de prueba de proteínas repetidas ha sido generada aleatoriamente de manera correcta. Ejemplo de esta comparación se puede ver, a continuación, en la Figura F.3.

Figure F.3

Imagen basada en el pdb file generado aleatoriamente por el script de autoría propia vs Imagen basada en el pdb file extraído de la base de datos RCSB PDB

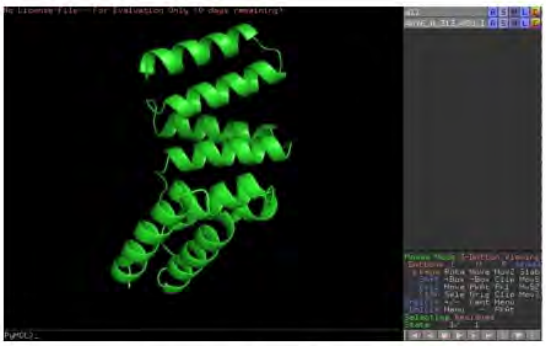


Imagen de la región repetida de la proteína 4AY6 generado por PyMOL usando como dato de entrada el pdb file generado aleatoriamente por el script de autoría propia

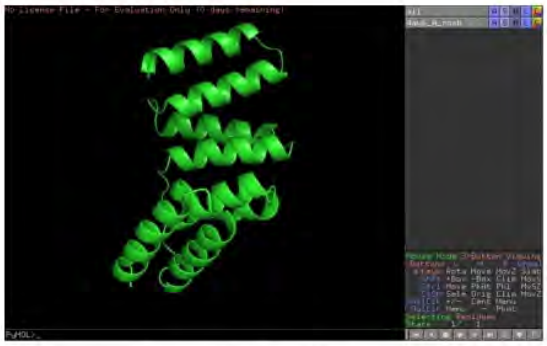


Imagen de la región repetida de la proteína 4AY6 generado por PyMOL usando como dato de entrada el pdb file extraído de la base de datos RCSB PDB

Nota: El gráfico muestra la comparación entre la estructura de la región repetida de la proteína 4AY6 basada en el pdb file generado aleatorimante por el script de autoría propia usando el software PyMOL y la estructura de la región repetida de la misma proteína, pero basada en el pdb file extraído de la base de datos RCSB PDB. Elaboración propia.

Finalmente, esta verificación ha sido validada al 100% por la Dra. Layla Hirsh, experta en el tema de las proteínas repetidas; y el Dr. Nicolás Palopoli, experto en el tema de diversidad conformacional en proteínas.

F.4. Resultados

A continuación, en la Tabla F.1 se presenta el total de pdb files de la región repetida y de las unidades de repetición de las proteínas repetidas generados aleatoriamente por el script de autoría propia (Ver Figura F.2).

Tabla F.1

Cantidad de PDB Files generados aleatoriamente de la región repetida y de las unidades de repetición de las proteínas repetidas

Clase	**Descripción**	**Cantidad de cadenas de proteínas repetidas**	**PDB Files de la región repetida**	**PDB Files de las unidades de repetición**
II	Estructura fibrosas	4	4	22
III	Estructuras alargadas	819	819	4127
IV	Estructuras cerradas	1000	1000	5158
V	Estructuras “Beads on a String”	77	77	393
Total		**1900**	**1900**	**9700**

Anexo G

Reporte de resultados de los dos métodos existentes

G.1. Introducción

El presente documento describe los dos métodos genéricos que se utilizarán sobre el conjunto de datos de prueba de las proteínas repetidas. Posteriormente, se detalla la manera en cómo se generaron los resultados correspondientes por cada método existente. Y finalmente, se presenta el resultado de la comparación entre los resultados de RMSD y TM-score obtenidos usando estos métodos genéricos y los resultados existentes que se encuentran en la base de datos CoDNaS.

G.2. Métodos genéricos

Los métodos genéricos vienen a ser procedimientos que conllevan a la obtención de las diferentes conformaciones que posee una proteína en su estado nativo y en base a estas estructuras alternativas formar todos los pares posibles con la finalidad de superponer cada par para medir la diferencia estructural que existe entre ambas conformaciones. Esto con la finalidad de poder obtener el grado de diversidad conformacional e información característica de cada confórmero, ya que con estos datos los científicos van a poder analizar los cambios conformacionales que presenten las diferentes proteínas.

Sin embargo, lo que diferencia de un método genérico al otro es la unidad estadística, el cual sirve para medir el grado de diversidad conformacional. Es así que los métodos existentes que

en esta oportunidad se están tomando en cuenta utilizan como unidad estadística el RMSD y el TM-score. En adición, el método genérico que usa la unidad estadística RMSD utiliza el software Mammoth para poder calcular esta unidad y el otro método existente que usa la unidad estadística TM-score utiliza el software TMAlign.

G.3. Generación de los resultados

Para generar los resultados, tanto del método que utiliza la unidad estadística TM-score como el método que utiliza la unidad estadística RMSD, primero se tiene que conocer las diferentes estructuras alternativas o conformaciones de cada una de las proteínas del conjunto de datos de prueba de las proteínas repetidas.

Por ello, se accedió a la base de datos CoDNaS, el cual almacena la información de diversidad conformacional de las proteínas en su estado nativo y parte de esta información contiene las diferentes conformaciones de cada una de las proteínas registradas hasta su última actualización, la cual es el 27 de Abril del 2017. Luego, una vez extraído las diferentes conformaciones del conjunto de prueba de las proteínas repetidas se procedió a filtrar esta información con la finalidad de tener una lista de proteínas repetidas que tengan más de una conformación (Ver Figura G.1) y para lograr esto se elaboró un script, el cual se puede apreciar en la Figura G.2 y en la siguiente dirección: *https://drive.google.com/file/filtrar_PRs_files.py*.

Figure G.1

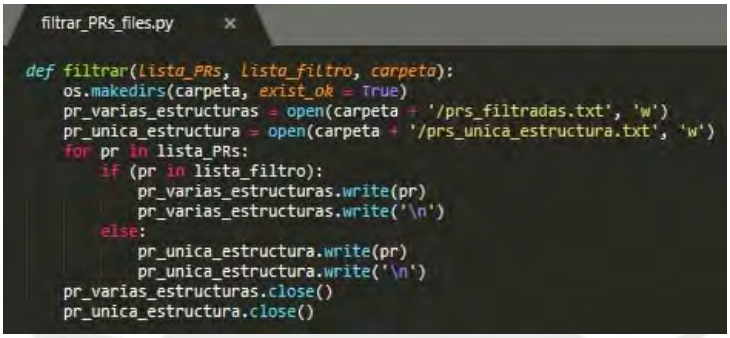
Filtrar el conjunto de prueba de las proteínas repetidas

Nota: El gráfico representa la lista filtrada de proteínas repetidas que poseen más de una conformación. Elaboración propia.

Luego de filtrar y tener la lista con las proteínas repetidas que poseen más de una conformación, se procedió a generar los archivos en formato pdb (pdb files) de las diferentes conformaciones. Para esto, se elaboró un script el cual se puede apreciar en la Figura G.3 y en la siguiente dirección: *https://drive.google.com/file/generar_conformers_files.py*.

Figure G.2

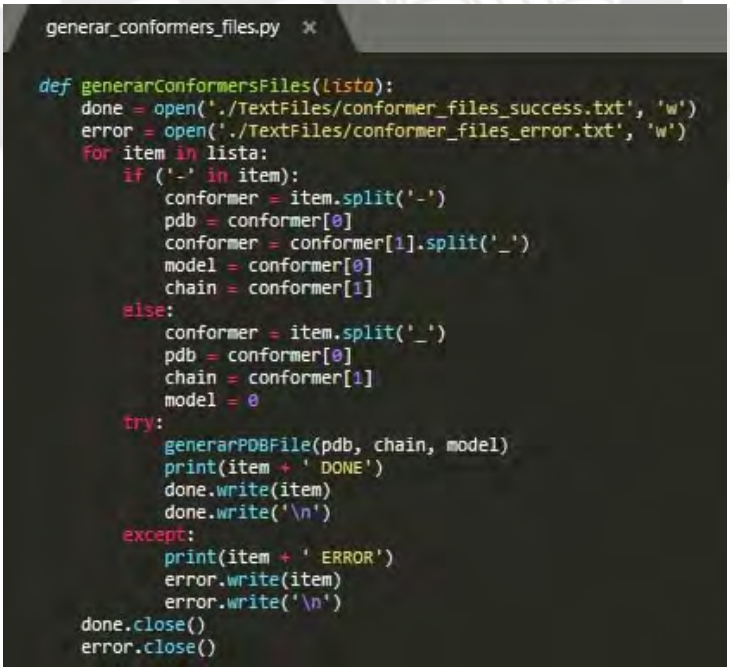
Script que filtra el conjunto de prueba de las proteínas repetidas



Nota: El gráfico presenta el script que filtra el conjunto de prueba de proteínas repetidas en una lista de proteínas repetidas que tienen más de una conformación. Elaboración propia.

Figure G.3

Script que genera los archivos en formato pdb de las diferentes conformaciones



Nota: El gráfico presenta el script que genera los pdb files de las diferentes conformaciones extraídas de la base de datos CoDNaS. Elaboración propia.

Finalmente, teniendo los pdb files de las diferentes conformaciones de cada proteína repetida, se realizó el cálculo del grado de diversidad conformacional para cada par posible. Para esto, se elaboraron dos scripts, uno que utiliza el software Mammoth (Ver Figura G.4), la cual se puede encontrar en la siguiente dirección: *https://drive.google.com/file/calc_rmsd.py*; y el otro que utiliza el software TMAlign (Ver Figura G.5), el cual se puede ubicar en la siguiente ruta: *https://drive.google.com/file/calc_tm_score.py*. Asimismo, los resultados se generaron en archivos con formato tsv y estos se pueden aprecia en la Figura G.6 y en la Figura G.7; además, se pueden encontrar en las siguientes direcciones: *https://drive.google.com/file/res_rmsd.tsv* y *https://drive.google.com/file/res_tm_score.tsv*

Figure G.4

Script que calcula el RMSD de las diferentes conformaciones

calcular_rmsd.py

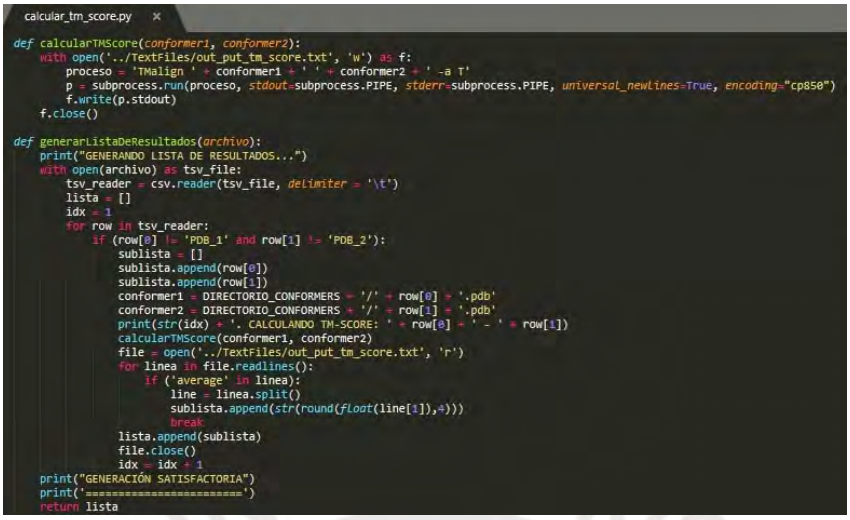
```
def calcularRMSD(conformer1, conformer2):
    subprocess.call(['./mymammoth', '-p', conformer1, '-e', conformer2, '-o', '../TextFiles/out_put_rmsd.txt'])

def generarListaDeResultados(archivo):
    print("GENERANDO LISTA DE RESULTADOS...")
    with open(archivo) as tsv_file:
        tsv_reader = csv.reader(tsv_file, delimiter = '\t')
        lista = []
        idx = 1
        for row in tsv_reader:
            if (row[0] != 'PDB_1' and row[1] != 'PDB_2'):
                sublista = []
                sublista.append(row[0])
                sublista.append(row[1])
                conformer1 = DIRECTORIO_CONFORMERS + '/' + row[0] + '.pdb'
                conformer2 = DIRECTORIO_CONFORMERS + '/' + row[1] + '.pdb'
                print(str(idx) + '. CALCULANDO RMSD: ' + row[0] + ' - ' + row[1])
                calcularRMSD(conformer1, conformer2)
                file = open('../TextFiles/out_put_rmsd.txt', 'r')
                for linea in file.readlines():
                    if ('Sstr(LG)' in linea):
                        line = linea.split()
                        sublista.append(line[7])
                        break
                lista.append(sublista)
                file.close()
                idx = idx + 1
    print("GENERACIÓN SATISFACTORIA")
    print('========================')
    return lista
```

Nota: El gráfico presenta el script que calcula el RMSD de las diferentes conformaciones de cada proteína repetida. Elaboración propia.

Figure G.5

Script que calcula el TM-score las diferentes conformaciones



```
calcular_tm_score.py
def calcularTMScore(conformer1, conformer2):
    with open('../TextFiles/out_put_tm_score.txt', 'w') as f:
        proceso = 'TMalign ' + conformer1 + ' ' + conformer2 + ' -a T'
        p = subprocess.run(proceso, stdout=subprocess.PIPE, stderr=subprocess.PIPE, universal_newlines=True, encoding="cp850")
        f.write(p.stdout)
    f.close()

def generarListaDeResultados(archivo):
    print("GENERANDO LISTA DE RESULTADOS...")
    with open(archivo) as tsv_file:
        tsv_reader = csv.reader(tsv_file, delimiter = '\t')
        lista = []
        idx = 1
        for row in tsv_reader:
            if (row[0] != 'PDB_1' and row[1] != 'PDB_2'):
                sublista = []
                sublista.append(row[0])
                sublista.append(row[1])
                conformer1 = DIRECTORIO_CONFORMERS + '/' + row[0] + '.pdb'
                conformer2 = DIRECTORIO_CONFORMERS + '/' + row[1] + '.pdb'
                print(str(idx) + '. CALCULANDO TM-SCORE: ' + row[0] + ' - ' + row[1])
                calcularTMScore(conformer1, conformer2)
                file = open('../TextFiles/out_put_tm_score.txt', 'r')
                for linea in file.readlines():
                    if ('average' in linea):
                        line = linea.split()
                        sublista.append(str(round(float(line[1]),4)))
                        break
                lista.append(sublista)
                file.close()
                idx = idx + 1
    print("GENERACIÓN SATISFACTORIA")
    print('======================')
    return lista
```

Nota: El gráfico presenta el script que calcula el TM-score de las diferentes conformaciones de cada proteína repetida. Elaboración propia.

Figure G.6

Resultados RMSD obtenidos del método genérico

	A	B	C
1	CONFORMER_1	CONFORMER_2	RMSD
2	1A12_A	1A12_B	0.72
3	1A12_B	1A12_C	0.62
4	1A12_B	1I2M_B	0.68
5	1A12_B	1I2M_D	0.72

Nota: El gráfico presenta los resultados RMSD obtenidos usando el método genérico que utiliza el software Mammoth. Elaboración propia.

Figure G.7

Resultados TM-score obtenidos del método genérico

	A	B	C
1	CONFORMER_1	CONFORMER_2	TM-SCORE
2	1A12_A	1A12_B	0.9848
3	1A12_B	1A12_C	0.9923
4	1A12_B	1I2M_B	0.9758
5	1A12_B	1I2M_D	0.9771

Nota: El gráfico presenta los resultados TM-score obtenidos usando el método genérico que utiliza el software TMAlign. Elaboración propia.

G.4. Comparación entre los resultados obtenidos y los resultados existentes en CoDNaS

Se realizó una comparación entre los resultados obtenidos utilizando los métodos genéricos descritos en la sección G.2 con los resultados extraídos de la base de datos CoDNaS. Para esto, se elaboró un script (Ver Figura G.8) que calcula la similitud y el resultado se puede apreciar en la Figura G.9. Esta comparación permitirá decidir si se realizó de manera correcta los métodos genéricos; de la misma manera de comprobar que se utilizaron adecuadamente el software Mammoth y el software TMAlign.

Figure G.8

Script para calcular la similitud

```python
import csv

TSV_TM_SCORE = '/mnt/d/Ronaldo/CoDNaS-Prs/Resultados/Test/TsvFiles/resultados_tm_score.tsv'
TSV_TM_SCORE_CODNAS = '/mnt/d/Ronaldo/CoDNaS-Prs/Resultados/Test/TsvFiles/tm_score_codnas.tsv'
TSV_RMSD = '/mnt/d/Ronaldo/CoDNaS-Prs/Resultados/Test/TsvFiles/resultados_rmsd.tsv'
TSV_RMSD_CODNAS = '/mnt/d/Ronaldo/CoDNaS-Prs/Resultados/Test/TsvFiles/rmsd_codnas.tsv'

def listarTSV(archivo):
    with open(archivo) as tsv_file:
        tsv_reader = csv.reader(tsv_file, delimiter = '\t')
        lista = []
        for row in tsv_reader:
            condicion1 = row[0] != 'PDB_1' and row[1] != 'PDB_2'
            condicion2 = row[0] != 'CONFORMER_1' and row[1] != 'CONFORMER_2'
            if (condicion1 and condicion2):
                sublista = []
                sublista.append(row[0])
                sublista.append(row[1])
                sublista.append(row[2])
                lista.append(sublista)
    return lista

def calcularSimilitud(file1, file2):
    lista_resultados = listarTSV(file1)
    lista_codnas = listarTSV(file2)
    count = 0
    lista = []
    for i in range(len(lista_resultados)):
        if (lista_resultados[i][2] == lista_codnas[i][2]):
            count = count + 1
    similitud_porcentaje = round(count*100/len(lista_resultados), 2)
    return similitud_porcentaje

def main():
    sim_porc_rmsd = calcularSimilitud(TSV_RMSD, TSV_RMSD_CODNAS)
    sim_porc_tm_score = calcularSimilitud(TSV_TM_SCORE, TSV_TM_SCORE_CODNAS)
    print("El porcentaje de similitud entre el TM-Score calculado y el TM-Score obtenido de la base de datos CoDNaS es del " + str(sim_porc_tm_score) + '%.')
    print("El porcentaje de similitud entre el RMSD calculado y el RMSD obtenido de la base de datos CoDNaS es del " + str(sim_porc_rmsd) + '%.')

if __name__ == '__main__':
    main()
```

Nota: El gráfico presenta el script del cálculo de similitud entre los resultados obtenidos y los existentes que se encuentran en la base de datos CoDNaS. Elaboración propia.

Figure G.9

Cálculo de la similitud entre los resultados obtenidos y los existentes en CoDNaS

```
C:\WINDOWS\system32\cmd.exe - cmd mammoth_1.2

C:\Users\Usuario\Downloads>python calcular_similitud.py
El porcentaje de similitud entre el TM-Score calculado y el TM-Score obtenido de la base de datos CoDNaS es del 83.47%.
El porcentaje de similitud entre el RMSD calculado y el RMSD obtenido de la base de datos CoDNaS es del 94.33%.
```

Nota: El gráfico presenta el cálculo de similitud entre los resultados obtenidos y los existentes que se encuentran en la base de datos CoDNaS. Elaboración propia.

Anexo H

Reporte de propuestas de los métodos a aplicar

H.1. Introducción

El presente reporte describe la definición de las tres propuestas de métodos a aplicar que permiten el análisis de diversidad conformacional de las proteínas repetidas, las cuales son: Región de repetición, Unidades de repetición como confórmeros y Unidades de repetición de los confórmeros.

H.2. Definición de las propuestas de métodos

A continuación, se definirán las tres propuestas de métodos a aplicar para analizar la diferencia conformacional de las proteínas repetidas. Asimismo, estas propuestas están basadas en los métodos genéricos y en hipótesis trazadas con la Dra. Layla Hirsh, experta en los temas relacionados con las proteínas repetidas; el Dr. Gustavo Parisi y el Dr. Nicolás Palopoli, expertos en evaluar la diversidad conformacional de las proteínas repetidas.

H.2.1. Propuesta N° 1: Región de repetición

Toda proteína repetida cuenta con una o varias regiones de repetición. Teniendo en cuesta esto, el conjunto de confórmeros podría ser representado por las regiones de repetición de cada confórmero que presenta la misma proteína. Es decir, si una proteína repetida tiene 5 conformaciones y una región de repetición, entonces el conjunto de confórmeros estaría formado por

estas 5 conformaciones basados en la estructura de la región de repetición.

Es así que, partiendo como base la hipótesis de definir al conjunto de confórmeros de una proteína repetida como las conformaciones determinadas de cada región de repetición, la propuesta N° 1, Región de repetición, consiste, en primer lugar, determinar todas las conformaciones de las regiones de repetición que presenta cada proteína repetida. Para esto, se utilizará la base de datos RCSB PDB para descargar las secuencias de las proteínas en formato FASTA con la finalidad de utilizar este archivo descargado en el software CD-HIT para obtener un archivo que contenga todas las proteínas separadas en clústeres. Cada clúster estará conformado por n proteínas que son similares entre sí. Es así que teniendo este archivo de clústeres se va a poder ubicar las diferentes conformaciones de cada proteína repetida. Y, una vez identificadas, se usará el software TM-align para alinear las regiones de repetición de cada proteína repetida con las conformaciones de la misma para generar el nuevo conjunto de conformaciones basados en la regiones de repetición.

En segundo lugar, luego de tener los nuevos conjuntos de confórmeros para cada proteína repetida, se formará todos los pares posibles de confórmeros con la finalidad de superponer las conformaciones de cada par para medir la diferencia que existe entre las estructuras y obtener el grado de diversidad conformacional. Cabe señalar que esta diferencia estructural va a estar representada por la desviación cuadrática media raíz (RMSD). Asimismo para la superposición del par de confórmeros y el cálculo de esta medida estadística se utilizará el software Mammoth.

Finalmente, empleando esta propuesta de método se va a tener un conjunto de resultados del análisis completo sobre las diversas conformaciones que presenta cada proteína repetida.

H.2.2. Propuesta N° 2: Unidades de repetición como confórmeros

Cada proteína repetida cuenta con una serie de unidades de repetición dentro de la región de repetición. Teniendo en cuesta esto, cada unidad se podría considerar como un confórmero de la misma proteína. Es decir, si una proteína repetida tiene 5 unidades de repetición, entonces el conjunto de confórmeros estaría formado por estas 5 unidades.

Es así que, partiendo como base la hipótesis de definir al conjunto de confórmeros de una proteína repetida como las unidades de repetición que posee la misma, la propuesta N° 2,

Unidades de repetición como confórmeros, consiste, en primer lugar, identificar las unidades de repetición de cada proteína repetida, ya que estas unidades formarán los nuevos conjuntos de conformaciones de cada proteína repetida.

En segundo lugar, luego de tener los nuevos conjuntos de confórmeros para cada proteína repetida, se formará todos los pares posibles de confórmeros con la finalidad de superponer las conformaciones de cada par para medir la diferencia que existe entre las estructuras y obtener el grado de diversidad conformacional. Cabe señalar que esta diferencia estructural va a estar representada por la desviación cuadrática media raíz (RMSD). Asimismo para la superposición del par de confórmeros y el cálculo de esta medida estadística se utilizará el software Mammoth.

Finalmente, empleando esta propuesta de método se va a tener un conjunto de resultados del análisis completo sobre las diversas conformaciones que presenta cada proteína repetida.

H.2.3. Propuesta N° 3: Unidades de repetición de los confórmeros

Conociendo que una cadena de proteína repetida puede tener diversas regiones de repetición y cada región puede tener diversas conformaciones, pero dentro de cada región de repetición se tiene un número diverso de unidades repetidas. Entonces, el conjunto de confórmeros de cada proteína repetida estaría formada por las unidades de repetición de cada confórmero de cada proteína repetida. Es decir, si una proteína repetida tiene 5 unidades de repetición y 5 conformaciones, entonces el conjunto de confórmeros estaría formado por estas 5 unidades ubicadas en cada conformación y esto haría un total de 25 confórmeros.

Es así que, partiendo como base la hipótesis de definir al conjunto de confórmeros de una proteína repetida como las unidades de repetición de cada conformación que posee la misma, la propuesta N° 3, Unidades de repetición de las conformaciones, consiste, en primer lugar, identificar las unidades de repetición de cada proteína repetida. En segundo lugar, determinar todas las conformaciones de las proteínas repetidas y para esto, se utilizará la base de datos RCSB PDB para descargar las secuencias de las proteínas en formato FASTA con la finalidad de utilizar este archivo descargado en el software CD-HIT para obtener un archivo que contenga todas las proteínas separadas en clústeres. Cada clúster estará conformado por n proteínas que son similares entre sí. Es así que teniendo este archivo de clústeres se va a poder ubicar las diferentes conformaciones de cada proteína repetida.

En tercer lugar, teniendo identificadas los confórmeros, se usará el software TM-align para alinear las unidades de repetición de cada proteína repetida con las conformaciones de la misma para generar el nuevo conjunto de conformaciones basados en las unidades de repetición. En cuarto lugar, luego de tener los nuevos conjuntos de confórmeros para cada proteína repetida, se formará todos los pares posibles de confórmeros con la finalidad de superponer las conformaciones de cada par para medir la diferencia que existe entre las estructuras y obtener el grado de diversidad conformacional. Cabe señalar que esta diferencia estructural va a estar representada por la desviación cuadrática media raíz (RMSD). Asimismo para la superposición del par de confórmeros y el cálculo de esta medida estadística se utilizará el software Mammoth.

Finalmente, empleando esta propuesta de método se va a tener un conjunto de resultados del análisis completo sobre las diversas conformaciones que presenta cada proteína repetida.

Anexo I

Reporte de resultados de las tres propuestas de métodos

I.1. Introducción

El presente reporte describe la generación de los resultados de las tres propuestas de métodos a aplicar para el análisis de diversidad conformacional de las proteínas repetidas, las cuales son: Región de repetición, Unidades de repetición como confórmeros y Unidades de repetición de los confórmeros. Asimismo, estas propuestas de métodos fueron son aplicadas en el conjunto de datos de prueba de las proteínas repetidas. Y finalmente, se presenta la comparación, en caso exista, entre los resultados obtenidos y los resultados calculados de los métodos genéricos.

I.2. Generación de resultados

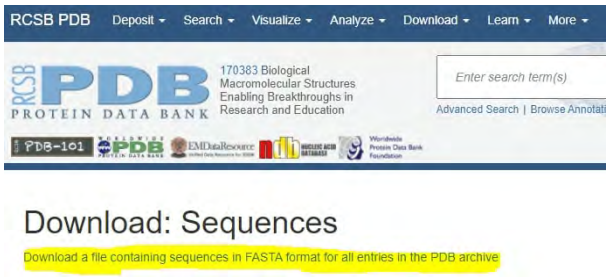
I.2.1. Propuesta N° 1: Región de repetición

Para generar los resultados de la presente propuesta, primero se tuvo que acceder a la base de datos RCSB PDB para descargar todas las secuencias de las proteínas repetidas en formato FASTA (Ver Figura I.1) a través de la siguiente dirección: *www.rcsb.org/downloads/fasta*.

Al realizar la descarga se obtuvo un archivo en formato zip con las secuencias de todas las proteínas. Este archivo fue utilizado como dato de entrada para el software CD-HIT Suite (Ver Figura I.2), el cual compara todas las secuencias de las proteínas insertadas para poder separar cada proteína en un conjunto, denominado clúster, cuyos elementos son similares entre sí.

Figure I.1

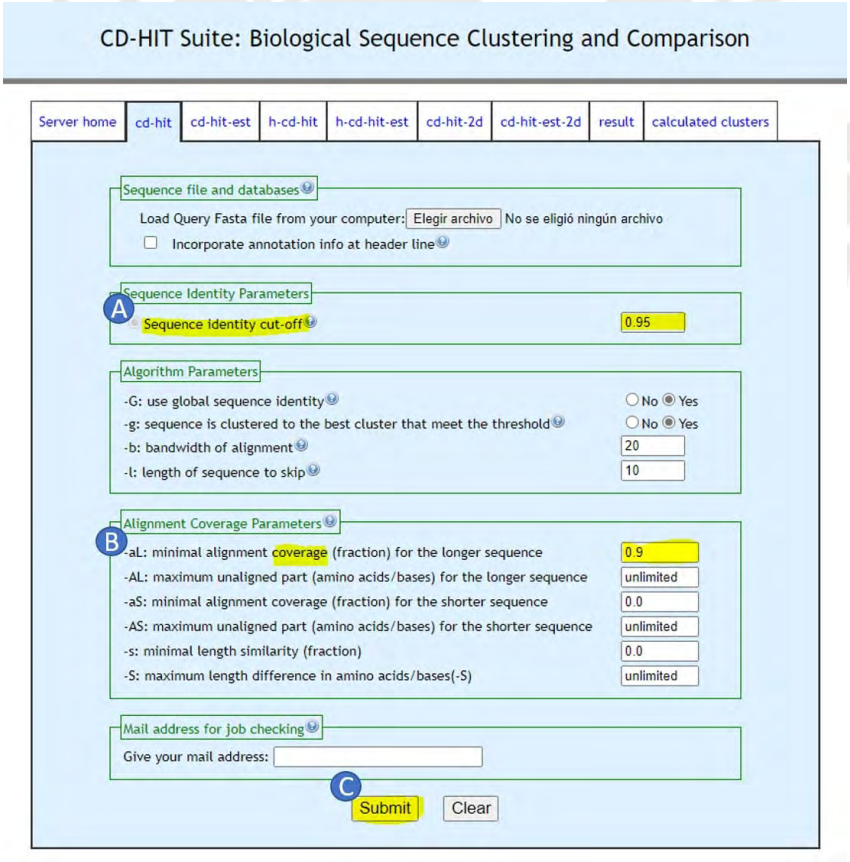
RCSB PDB: Descarga de secuencias de proteínas



Nota: El gráfico presenta la sección de la base de datos RCSB PDB para poder descargar las secuencias de las proteínas en formato FASTA. Elaboración propia.

Figure I.2

Pasos para generar el archivo de clústeres



Nota: A: Se utiliza 95% de secuencia de identidad. B: Se utilizar el 90% de coverage. C: Se presiona submit luego de establecer la secuencia de identidad y el coverage. Elaboración propia.

Luego de utilizar el software CD-HIT Suite, se obtuvo un archivo el cual contiene todas las proteínas separadas en clústeres. Este archivo se puede encontrar a través de la siguiente dirección: *https://drive.google.com/file/full.out.clstr*. Después, se elaboró un script, el cual se puede ubicar en la siguiente dirección: *https://drive.google.com/m1/filtrar_proteinas_repetidas.py*, para identificar los clústeres que poseen alguna proteína repetida del conjunto de datos de prueba y para filtrar las proteínas repetidas que tienen más de una conformación. Asimismo, parte de este script mencionado se puede apreciar en la Figura I.3.

Figure I.3

Script para identificar clústeres y filtrar proteinas repetidas

filtrar_proteinas_repetidas.py

```
import os

DIR_PRS = '/mnt/d/Ronaldo/CoDNaS-PRs/DataSet/Test/PDBFiles'
FILE_CLUSTERS = '/mnt/d/Ronaldo/CoDNaS-PRs/Resultados/Test/Metodo1/ClstrFiles/full.out.clstr'
DIR_TEXT_FILES = '/mnt/d/Ronaldo/CoDNaS-PRs/Resultados/Test/Metodo1/TextFiles'

def listarClusters(archivo):
    lista = []
    with open(archivo, 'r') as cluster_file:
        cluster = ''
        lista_pdb = []
        for linea in cluster_file.readlines():
            identificador = linea.split()
            if ('>' in identificador[0]):
                if (cluster != ''):
                    tupla = cluster, lista_pdb
                    lista.append(tupla)
                    cluster = identificador[0][1:] + ' ' + identificador[1]
                    lista_pdb = []
                else:
                    cluster = identificador[0][1:] + ' ' + identificador[1]
            else:
                line = linea.split()
                pdb = line[2][1:len(line[2]) -3]
                lista_pdb.append(pdb)
        tupla = cluster, lista_pdb
        lista.append(tupla)
    return lista

def listarPRs(directorio):
    return os.listdir(directorio)

def listarPRsFiltradas(lista_prs, lista_clusters):
    lista = []
    for pr in lista_prs:
        pr_minus = pr[:4].lower() + pr[4:len(pr)]
        for cluster in lista_clusters:
            if (pr_minus in cluster[1]):
                if (pr not in lista):
                    lista.append(pr)
                break
    return lista
```

Nota: La presente figura muestra parte del script que identifica clústeres con alguna proteína repetida del conjunto de datos de prueba y que filtra a las proteínas repetidas que tienen más de una conformación. Elaboración propia.

Al tener las proteínas filtradas y los clústeres identificadas, se utilizó el software TM-align para alinear las regiones de repetición de cada proteína repetida con sus respectivas conformaciones extraídas del clúster a la que pertenece. Esto se hizo con la finalidad de identificar las regiones repetidas de la proteína repetida en cada conformación y para esto se elaboró un script, el cual se puede apreciar en la Figura I.4 y se puede encontrar en la siguiente dirección: *https://drive.google.com/m1/identificar_regiones.py*

Figure I.4

Script para identificar las regiones repetidas de las conformaciones de la proteína repetida

identificar_regiones.py

```
def alinearEstructuras(path_conformer_1, path_conformer_2, conformer, i):
    ubicacion = './SupFiles/' + conformer + '/' + conformer + '_reg_' + str(i) + '.sup'
    subprocess.call(['./TMalign', path_conformer_1, path_conformer_2, '-a', 'T', '-o', ubicacion])

def limpiar(conformer):
    directorio = './SupFiles/' + conformer
    lista = os.listdir(directorio)
    l = []
    for item in lista:
        l.append('./SupFiles/' + conformer + '/' + item)
    if (len(l) > 15):
        subprocess.call(['rm', l[1], l[2], l[3], l[4], l[5], l[6], l[7], l[8], l[9], l[10], l[11],
            l[12], l[13], l[14], l[16], l[17], l[18], l[19], l[20], l[21], l[22], l[23], l[24], l[25], l[26], l[27], l[28], l[29]])
    else:
        subprocess.call(['rm', l[1], l[2], l[3], l[4], l[5], l[6], l[7], l[8], l[9], l[10], l[11],
            l[12], l[13], l[14]])

def identificarRegiones(lista_conformeros_filtrados):
    archivo = open('./TextFiles/conformers_sin_regiones_ubicadas.txt', 'w')
    for tupla in lista_conformeros_filtrados:
        lista = os.listdir(DIR_PROTEINAS_REPETIDAS + '/' + tupla[0])
        lista_reg = []
        for reg in lista:
            if ('unit' not in reg):
                lista_reg.append(reg)
        for conformer in tupla[2]:
            #if (conformer != tupla[0]):
            crearDirectorios('./SupFiles/' + conformer)
            i = 1
            for reg in lista_reg:
                conformer_1 = DIR_PROTEINAS_REPETIDAS + '/' + tupla[0] + '/' + reg
                conformer_2 = DIR_PDB_FILES + '/' + conformer + '.pdb'
                alinearEstructuras(conformer_1, conformer_2, conformer, i)
                i = i + 1
            try:
                limpiar(conformer)
            except:
                archivo.write(conformer)
                archivo.write('\n')
    archivo.close()
```

Nota: La presente figura muestra parte del script que identifica las regiones repetidas de la proteína repetida en las conformaciones de la misma. Elaboración propia.

Después de identificar las regiones repetidas de cada confórmero de cada proteína repetida se elaboró un script que permite generar los archivos en formato pdb (pdb files) de los mismos. Parte de este script se puede apreciar en la Figura I.5 y se puede visualizar en la siguiente dirección: *https://drive.google.com/m1/generar_regiones_conformers.py*

Figure I.5

Script para generar las regiones repetidas en las conformaciones de las proteínas repetidas

generar_regiones_conformers.py

```
def generarConformacionesRegiones(lista_regiones):
    for tupla in lista_regiones:
        generarPDBFile(tupla[0], tupla[1], tupla[2])

def crearDirectorios(directorio):
    os.makedirs(directorio, exist_ok = True)

def generarPDBFile(conformer, lim_inf, lim_sup):
    directorio = DIR_CONFORMERS + '/' + conformer
    crearDirectorios(directorio)
    name_out_file = directorio + '/' + conformer + '_' + lim_inf + '_' + lim_sup + '.pdb'
    out_file = open(name_out_file, 'w')
    is_final = 0
    archivo = open(DIR_PDB_FILES + '/' + conformer + '.pdb', 'r')
    for linea in archivo.readlines():
        line = linea.split()
        if (line[0] == 'ATOM'):
            if (len(line[4]) == 1):
                try:
                    if (line[4] == conformer[5:6]):
                        if (int(lim_inf) <= int(line[5]) and int(line[5]) <= int(lim_sup)):
                            is_final = 1
                            out_file.write(linea)
                        elif (is_final):
                            out_file.write('TER')
                            break
                except:
                    if (line[5] == conformer[5:6]):
                        if (int(lim_inf) <= int(line[6]) and int(line[6]) <= int(lim_sup)):
                            is_final = 1
                            out_file.write(linea)
                        elif (is_final):
                            out_file.write('TER')
                            break
            else:
                identifier = line[4][1:len(line[4])]
                if (line[4][0] == conformer[5:6]):
                    if (int(lim_inf) <= int(identifier) and int(identifier) <= int(lim_sup)):
                        is_final = 1
                        out_file.write(linea)
                    elif (is_final):
                        out_file.write('TER')
                        break
        elif (line[0] == 'TER' and is_final):
            out_file.write('TER')
            break
    archivo.close()
    out_file.close()
```

Nota: La presente figura muestra parte del script que genera los pdb files de las regiones repetidas de las proteínas repetidas en las conformaciones de la misma. Elaboración propia.

Es así que, teniendo los pdb files del nuevo conjunto de confórmeros de cada proteína repetida delimitada por sus regiones de repetición. Se procedió a formar la máxima cantidad de

pares de confórmeros por cada proteína repetida para luego utilizar el software Mammoth para superposicionar las estructuras de cada par y calcular la diversidad conformacional a través de la medidad estadística RMSD. Y para realizar lo mencionado recientemente, se elaboró un script que se puede encontrar en la siguiente ruta: *https://drive.google.com/m1/gen_result.py* y parte del código se puede apreciar en la Figura I.6.

Figure I.6

Script para generar los resultados de diversidad conformacional de cada proteína repetida

generar_resultados.py

```
def calcularRMSD(conformer_1, conformer_2):
  subprocess.call(['./mymammoth', '-p', conformer_1, '-e', conformer_2, '-o', './TextFiles/output_rmsd_metodo1.txt'])

def listarResultados(lista_conformers):
  lista = []
  i = 0
  num_conf = 0
  laxus = os.listdir(DIR_CONFORMERS_FILES)
  for tupla in lista_conformers:
    num_conf = num_conf + 1
    sublista = []
    pares = list(itertools.combinations(tupla[1], 2))
    for pair in pares:
      i = i + 1
      conf_1, conf_2 = pair
      if (conf_1 in laxus and conf_2 in laxus):
        l_conf_1 = os.listdir(DIR_CONFORMERS_FILES + '/' + conf_1)
        l_conf_2 = os.listdir(DIR_CONFORMERS_FILES + '/' + conf_2)
        for j in range(len(l_conf_1)):
          conformer_1 = DIR_CONFORMERS_FILES + '/' + conf_1 + '/' + l_conf_1[j]
          arr_1 = l_conf_1[j].split('_')
          lim_inf_1 = arr_1[2]
          arr_1 = arr_1[3].split('.')
          lim_sup_1 = arr_1[0]
          conformer_2 = DIR_CONFORMERS_FILES + '/' + conf_2 + '/' + l_conf_2[j]
          arr_2 = l_conf_2[j].split('_')
          lim_inf_2 = arr_2[2]
          arr_2 = arr_2[3].split('.')
          lim_sup_2 = arr_2[0]
          print(str(i) + '. ' + conf_1 + ' - ' + conf_2)
          calcularRMSD(conformer_1, conformer_2)
          rmsd_file = open('./TextFiles/output_rmsd_metodo1.txt', 'r')
          for linea in rmsd_file.readlines():
            if ('Sstr(LG)' in linea):
              line = linea.split()
              rmsd = line[7]
            if ('Seq.sim=' in linea):
              line = linea.split()
              num_region = j + 1
              par = conf_1, lim_inf_1, lim_sup_1, conf_2, lim_inf_2, lim_sup_2, line[3], rmsd, num_region
              sublista.append(par)
              break
    cluster = tupla[0].split()
    cluster = cluster[1]
    item = cluster, sublista
    lista.append(item)
  return lista
```

Nota: La presente figura muestra parte del script que permite generar los resultados de diversidad conformacional a través del RMSD de cada proteína repetida. Elaboración propia.

Finalmente, se generó un archivo Tsv que registra los cálculos de la diversidad conformacional sobre el conjunto de proteínas repetidas empleando el método seleccionado. Asimismo, este archivo se puede encontrar en la siguiente ruta: *https://drive.google.com/m1/res_rmsd.tsv*

y se puede apreciar en la Figura I.7.

Figure I.7

Resultados obtenidos de la propuesta de método N° 1: Región de repetición

	A	B	C	D	E	F	G	H	I	J
1	cluster	conformero_1	lim_inf_1	lim_sup_1	conformero_2	lim_inf_2	lim_sup_2	sec_similitud	rmsd	region
2	1	1A12_A	35	417	1A12_B	35	417	5.028	0.71	1
3	1	1A12_A	35	417	1A12_C	35	417	5.034	0.7	1
4	1	1A12_A	35	417	1I2M_B	35	417	4.951	0.64	1
5	1	1A12_A	35	417	1I2M_D	35	417	4.974	0.77	1
6	1	1A12_A	35	417	5TBK_I	35	417	5.232	0.79	1

Nota: La presente figura muestra el archivo tsv que contiene la información de diversidad conformacional del conjunto de datos de prueba de las proteínas repetidas empleando la propuesta de método N° 1: Región de repetición. Elaboración propia.

I.2.2. Propuesta N° 2: Unidades de repetición como confórmeros

Para generar los resultados de la presente propuesta, primero se tuvo que identificar las unidades de repetición de cada proteína repetida para elaborar su respectivo pdb file y considerar a estas unidades de repetición como el nuevo conjunto de conformaciones de la repectiva proteína repetida. Para esto, se accede al conjunto de datos de prueba de las proteínas repetidas elaborado previamente, ya que en este conjunto se encuentran identificados los archivos pdb de las unidades repetitivas.

En segundo lugar, teniendo los pdb files de las unidades de repetición como confórmeros de cada proteína repetida, se realizó el cálculo del grado de diversidad conformacional para cada par posible a través de la unidad estadística RMSD usando el software Mammoth . Para esto, se elaboraró un script (Ver Figura I.9), el cual se puede puede encontrar en la siguiente dirección: *https://drive.google.com/m2/generar_resultados.py*.

Finalmente, los resultados obtenidos se generaron en un archivo Tsv por medio de un script (Ver Figura I.9), y estos resultados se pueden apreciar en la Figura I.8; además, se pueden ubicar en la siguiente dirección: *https://drive.google.com/m2/resultados_rmsd.tsv*.

Figure I.8

Resultados obtenidos de la propuesta de método N° 2: Unidades de repetición como confórmeros

	A	B	C	D	E	F	G	H	I
1	cluster	conformero_1	lim_inf_1	lim_sup_1	conformero_2	lim_inf_2	lim_sup_2	sec_similitud	rmsd
2	1	1A12_B	151	192	1A12_B	193	260	0.353	1.78
3	1	1A12_B	151	192	1A12_B	261	314	0.185	1.87
4	1	1A12_B	151	192	1A12_B	315	365	0.549	1.86
5	1	1A12_B	151	192	1A12_B	35	87	1	1.7

Nota: La presente figura muestra el archivo tsv que contiene la información de diversidad conformacional del conjunto de datos de prueba de las proteínas repetidas empleando la propuesta de método N° 2: Unidades de repetición como confórmeros. Elaboración propia.

Figure I.9

Script para calcular el RMSD de las diferentes conformaciones y generar el conjunto de resultados

generar_resultados.py

```
def calcularRMSD(conformer_1, conformer_2):
    subprocess.call(['./mymammoth', '-p', conformer_1, '-e', conformer_2, '-o', './TextFiles/output_rmsd.txt'])

def generarTSV(lista):
  with open('./TsvFiles/resultados_rmsd.tsv', 'w') as out_file:
    tsv_writer = csv.writer(out_file, delimiter = '\t', lineterminator = '\n')
    tsv_writer.writerow(['cluster', 'conformero_1', 'lim_inf_1', 'lim_sup_1', 'conformero_2', 'lim_inf_2',
      'lim_sup_2', 'sec_similitud', 'rmsd'])
    for row in lista:
      cluster, sublista = row
      for tupla in sublista:
        tsv_writer.writerow([cluster, tupla[0], tupla[1], tupla[2], tupla[3], tupla[4], tupla[5], tupla[6],
          tupla[7]])

def listarResultados(lista_pares):
  lista = []
  i = 0
  idx = 1
  for pr in lista_pares:
    sublista =[]
    pares = list(itertools.combinations(pr, 2))
    for par in pares:
      i = i + 1
      arr = par[0].split('/')
      arr = arr[len(arr)-1].split('.')
      aux = arr[0].split('_')
      conformero_1 = aux[0] + '_' + aux[1]
      lim_inf_1 = aux[2]
      lim_sup_1 = aux[3]
      print(str(i) + ', ' + arr[0] + ' - ', end = '')
      arr = par[1].split('/')
      arr = arr[len(arr)-1].split('.')
      aux = arr[0].split('_')
      conformero_2 = aux[0] + '_' + aux[1]
      lim_inf_2 = aux[2]
      lim_sup_2 = aux[3]
      print(arr[0])
      calcularRMSD(par[0], par[1])
      archivo = open('./TextFiles/output_rmsd.txt', 'r')
      for linea in archivo.readlines():
        if ('Sstr(LG)' in linea):
          line = linea.split()
          rmsd = line[7]
        if ('Seq.sim=' in linea):
          line = linea.split()
          tupla = conformero_1, lim_inf_1, lim_sup_1, conformero_2, lim_inf_2, lim_sup_2, line[3], rmsd
          sublista.append(tupla)
          break
    item = idx, sublista
    lista.append(item)
    idx = idx + 1
  return lista
```

Nota: El gráfico presenta el script que calcula el RMSD de las diferentes conformaciones de cada proteína repetida y genera el conjunto de resultados. Elaboración propia.

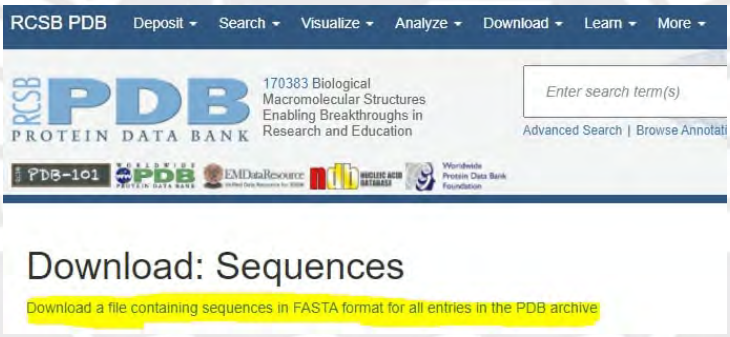
I.2.3. Propuesta N° 3: Unidades de repetición de los confórmeros

Para generar los resultados de la presente propuesta, primero se tuvo que acceder a la base de datos RCSB PDB para descargar todas las secuencias de las proteínas repetidas en formato FASTA (Ver Figura I.10) a través de la siguiente dirección: *www.rcsb.org/downloads/fasta*.

Al realizar la descarga se obtuvo un archivo en formato zip con las secuencias de todas las proteínas. Este archivo fue utilizado como dato de entrada para el software CD-HIT Suite (Ver Figura I.11), el cual compara todas las secuencias de las proteínas que se le insertan. Esto se realizó con la finalidad de separar cada proteína en un conjunto, denominado clúster, cuyos elementos son similares entre sí.

Figure I.10

RCSB PDB: Descarga de secuencias de proteínas

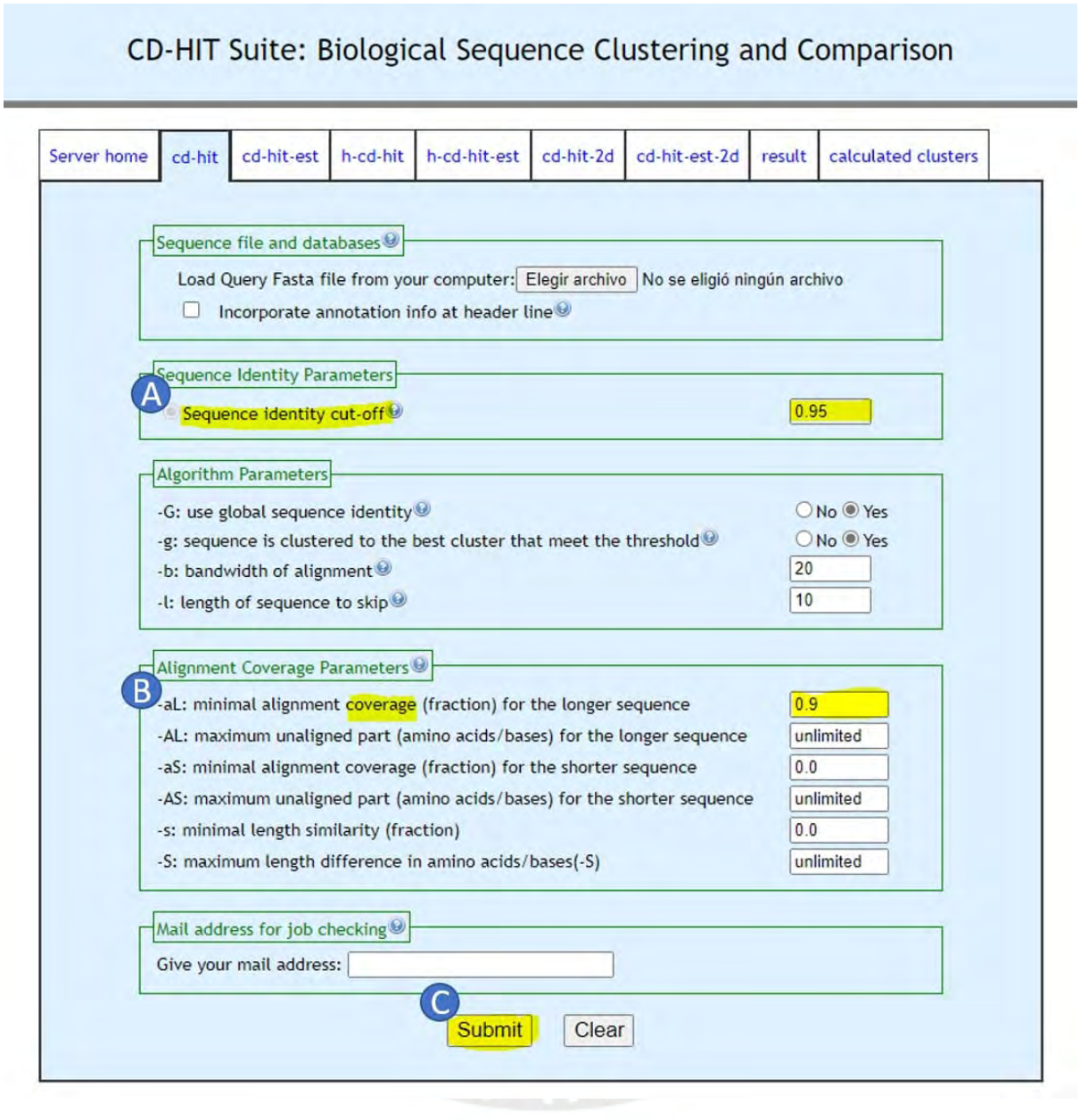


Nota: El gráfico presenta la sección de la base de datos RCSB PDB para poder descargar las secuencias de las proteínas en formato FASTA. Elaboración propia.

Luego de utilizar el software CD-HIT Suite, se obtuvo un archivo el cual contiene todas las proteínas separadas en clústeres. Este archivo se puede encontrar a través de la siguiente dirección: *https://drive.google.com/file/full.out.clstr*. Después, se elaboró un script, el cual se puede ubicar en la siguiente dirección: *https://drive.google.com/m3/filtrar_proteinas_repetidas.py*, para identificar los clústeres que poseen alguna proteína repetida del conjunto de datos de prueba y para filtrar las proteínas repetidas que tienen más de una conformación. Asimismo, parte de este script mencionado se puede apreciar en la Figura I.12.

Figure I.11

Pasos para generar el archivo de clústeres



Nota: A: Se utiliza 95% de secuencia de identidad. B: Se utilizar el 90% de coverage. C: Se presiona submit luego de establecer la secuencia de identidad y el coverage. Elaboración propia.

Figure I.12

Script para identificar clústeres y filtrar proteinas repetidas

filtrar_proteinas_repetidas.py

```
import os

DIR_PRS = '/mnt/d/Ronaldo/CoDNaS-PRs/DataSet/Test/PDBFiles'
FILE_CLUSTERS = '/mnt/d/Ronaldo/CoDNaS-PRs/Resultados/Test/Metodo3/ClstrFiles/full.out.clstr'
DIR_TEXT_FILES = '/mnt/d/Ronaldo/CoDNaS-PRs/Resultados/Test/Metodo3/TextFiles'

def listarClusters(archivo):
    lista = []
    with open(archivo, 'r') as cluster_file:
        cluster = ''
        lista_pdb = []
        for linea in cluster_file.readlines():
            identificador = linea.split()
            if ('>' in identificador[0]):
                if (cluster != ''):
                    tupla = cluster, lista_pdb
                    lista.append(tupla)
                    cluster = identificador[0][1:] + ' ' + identificador[1]
                    lista_pdb = []
                else:
                    cluster = identificador[0][1:] + ' ' + identificador[1]
            else:
                line = linea.split()
                pdb = line[2][1:len(line[2])-3]
                lista_pdb.append(pdb)
        tupla = cluster, lista_pdb
        lista.append(tupla)
    return lista

def listarPRs(directorio):
    return os.listdir(directorio)

def listarPRsFiltradas(lista_prs, lista_clusters):
    lista = []
    for pr in lista_prs:
        pr_minus = pr[:4].lower() + pr[4:len(pr)]
        for cluster in lista_clusters:
            if (pr_minus in cluster[1]):
                if (pr not in lista):
                    lista.append(pr)
                break
    return lista
```

Nota: La presente figura muestra parte del script que identifica clústeres con alguna proteína repetida del conjunto de datos de prueba y que filtra a las proteínas repetidas que tienen más de una conformación. Elaboración propia.

Al tener las proteínas filtradas y los clústeres identificadas, se utilizó el software TM-align para alinear la unidades de repetición de cada proteína repetida con sus respectivas conformaciones extraídas del clúster a la que pertenece. Esto se hizo con la finalidad de identificar las unidades de repetición de la proteína repetida en cada conformación y para esto se elaboró un script, el cual se puede apreciar en la Figura I.13 y se puede encontrar en la siguiente dirección: *https://drive.google.com/m3/identificar_unidades.py*

Figure I.13

Script para identificar las unidades de repetición de las conformaciones de la proteína repetida

identificar_unidades.py

```
def identificarUnidades(lista_conformeros_filtrados):
  archivo = open('./TextFiles/conformers_sin_unidades_ubicadas.txt', 'w')
  for tupla in lista_conformeros_filtrados:
    lista = os.listdir(DIR_PROTEINAS_REPETIDAS + '/' + tupla[0])
    lista_unit = []
    for unit in lista:
      if ('unit' in unit):
        lista_unit.append(unit)
    for conformer in tupla[2]:
      crearDirectorios('./SupFiles/' + conformer)
      i = 1
      for unit in lista_unit:
        conformer_1 = DIR_PROTEINAS_REPETIDAS + '/' + tupla[0] + '/' + unit
        conformer_2 = DIR_PDB_FILES + '/' + conformer + '.pdb'
        alinearEstructuras(conformer_1, conformer_2, conformer, i)
        i = i + 1
      try:
        limpiar(conformer)
      except:
        archivo.write(conformer)
        archivo.write('\n')
  archivo.close()

def alinearEstructuras(path_conformer_1, path_conformer_2, conformer, i):
  ubicacion = './SupFiles/' + conformer + '/' + conformer + '_unit_' + str(i) + '.sup'
  subprocess.call(['./TMalign', path_conformer_1, path_conformer_2, '-a', 'T', '-o', ubicacion])

def limpiar(conformer):
  directorio = './SupFiles/' + conformer + '/'
  lista = os.listdir(directorio)
  for item in lista:
    aux = item.split('.')
    if (aux[len(aux)-1] != 'sup'):
      subprocess.call(['rm', directorio + item])
```

Nota: La presente figura muestra parte del script que identifica las unidades de repetición de la proteína repetida en las conformaciones de la misma. Elaboración propia.

Después de identificar las unidades de repetición de cada confórmero de cada proteína repetida se elaboró un script que permite generar los archivos en formato pdb (pdb files) de los mismos. Parte de este script se puede apreciar en la Figura I.14 y se puede visualizar en la siguiente dirección: *https://drive.google.com/m3/generar_unidades_conformers.py*

Figure I.14

Script para generar las unidades de repetición en las conformaciones de las proteínas repetidas

generar_unidades_conformers.py

```
def generarConformacionesUnidades(lista_unidades):
  i = 1
  for tupla in lista_unidades:
    generarPDBFile(tupla[0], tupla[1], tupla[2], i)
    i = i + 1

def crearDirectorios(directorio):
  os.makedirs(directorio, exist_ok = True)

def generarPDBFile(conformer, lim_inf, lim_sup, idx):
  directorio = DIR_CONFORMERS + '/' + conformer
  crearDirectorios(directorio)
  name_out_file = directorio + '/' + conformer + '_' + lim_inf + '_' + lim_sup + '_' + str(idx) + '.pdb'
  out_file = open(name_out_file, 'w')
  is_final = 0
  archivo = open(DIR_PDB_FILES + '/' + conformer + '.pdb', 'r')
  for linea in archivo.readlines():
    line = linea.split()
    if (line[0] == 'ATOM'):
      if (len(line[4]) == 1):
        try:
          if (line[4] == conformer[5:6]):
            if (int(lim_inf) <= int(line[5]) and int(line[5]) <= int(lim_sup)):
              is_final = 1
              out_file.write(linea)
            elif (is_final):
              out_file.write('TER')
              break
        except:
          if (line[5] == conformer[5:6]):
            if (int(lim_inf) <= int(line[6]) and int(line[6]) <= int(lim_sup)):
              is_final = 1
              out_file.write(linea)
            elif (is_final):
              out_file.write('TER')
              break
      else:
        identifier = line[4][1:len(line[4])]
        if (line[4][0] == conformer[5:6]):
          if (int(lim_inf) <= int(identifier) and int(identifier) <= int(lim_sup)):
            is_final = 1
            out_file.write(linea)
          elif (is_final):
            out_file.write('TER')
            break
    elif (line[0] == 'TER' and is_final):
      out_file.write('TER')
      break
  archivo.close()
  out_file.close()
```

Nota: La presente figura muestra parte del script que genera los pdb files de las unidades de repetición de las proteínas repetidas en las conformaciones de la misma. Elaboración propia.

Es así que, teniendo los pdb files del nuevo conjunto de confórmeros de cada proteína repetida delimitada por sus unidades de repetición. Se procedió a formar la máxima cantidad de pares de confórmeros por cada proteína repetida para luego utilizar el software Mammoth para superposicionar las estructuras de cada par y calcular la diversidad conformacional a través de la medidad estadística RMSD. Y para realizar lo mencionado recientemente, se elaboró un

script que se puede encontrar en la siguiente ruta: *https://drive.google/m3/generar_resultados.py* y parte del código se puede apreciar en la Figura I.15.

Figure I.15

Script para generar los resultados de diversidad conformacional de cada proteína repetida

generar_resultados.py

```
def calcularRMSD(conformer_1, conformer_2):
  subprocess.call(['./mymammoth', '-p', conformer_1, '-e', conformer_2, '-o', './TextFiles/output_rmsd_metodo3.txt'])

def listarResultados(lista_conformers):
  lista = []
  i = 0
  num_conf = 0
  laxus = os.listdir(DIR_CONFORMERS_FILES)
  for tupla in lista_conformers:
    num_conf = num_conf + 1
    sublista = []
    pares = list(itertools.combinations(tupla[1], 2))
    for pair in pares:
      i = i + 1
      conf_1, conf_2 = pair
      if (conf_1 in laxus and conf_2 in laxus):
        l_conf_1 = os.listdir(DIR_CONFORMERS_FILES + '/' + conf_1)
        l_conf_2 = os.listdir(DIR_CONFORMERS_FILES + '/' + conf_2)
        for j in range(len(l_conf_1)):
          conformer_1 = DIR_CONFORMERS_FILES + '/' + conf_1 + '/' + l_conf_1[j]
          arr_1 = l_conf_1[j].split('_')
          lim_inf_1 = arr_1[2]
          lim_sup_1 = arr_1[3]
          conformer_2 = DIR_CONFORMERS_FILES + '/' + conf_2 + '/' + l_conf_2[j]
          arr_2 = l_conf_2[j].split('_')
          lim_inf_2 = arr_2[2]
          lim_sup_2 = arr_2[3]
          print(str(i) + '. ' + conf_1 + ' - ' + conf_2 + ' - ' + str(j+1))
          calcularRMSD(conformer_1, conformer_2)
          rmsd_file = open('./TextFiles/output_rmsd_metodo3.txt', 'r')
          for linea in rmsd_file.readlines():
            if ('Sstr(LG)' in linea):
              line = linea.split()
              rmsd = line[7]
            if ('Seq.sim=' in linea):
              line = linea.split()
              num_unidades = j + 1
              par = conf_1, lim_inf_1, lim_sup_1, conf_2, lim_inf_2, lim_sup_2, line[3], rmsd, num_unidades
              sublista.append(par)
              break
    cluster = tupla[0].split()
    cluster = cluster[1]
    item = cluster, sublista
    lista.append(item)
  return lista
```

Nota: La presente figura muestra parte del script que permite generar los resultados de diversidad conformacional a través del RMSD de cada proteína repetida. Elaboración propia.

Finalmente, se generó un archivo Tsv que registra los cálculos de la diversidad conformacional sobre el conjunto de proteínas repetidas empleando el método seleccionado. Asimismo, este archivo se puede encontrar en la siguiente ruta: *https://drive.google.com/m3/res_rmsd.tsv* y se puede apreciar en la Figura I.16.

Figure I.16

Resultados obtenidos de la propuesta de método N° 3: Unidades de repetición de los confórmeros

	A	B	C	D	E	F	G	H	I	J
1	cluster	conformero_1	lim_inf_1	lim_sup_1	conformero_2	lim_inf_2	lim_sup_2	sec_similitud	rmsd	unidad
2	1	1A12_A	151	192	1A12_B	151	192	4.976	0.67	1
3	1	1A12_A	193	260	1A12_B	193	260	4.3	1.62	2
4	1	1A12_A	261	314	1A12_B	261	314	5.611	0.64	3
5	1	1A12_A	315	365	1A12_B	315	365	4.98	0.12	4
6	1	1A12_A	35	87	1A12_B	35	87	5.038	0.24	5
7	1	1A12_A	366	417	1A12_B	366	417	5.269	0.12	6
8	1	1A12_A	88	150	1A12_B	88	150	5.238	0.38	7
9	1	1A12_A	151	192	1A12_C	151	192	4.976	0.8	1
10	1	1A12_A	193	260	1A12_C	193	260	4.3	1.5	2
11	1	1A12_A	261	314	1A12_C	261	314	5.611	0.32	3
12	1	1A12_A	315	365	1A12_C	315	365	4.98	0.1	4
13	1	1A12_A	35	87	1A12_C	35	87	5.038	0.16	5
14	1	1A12_A	366	417	1A12_C	366	417	5.269	0.16	6
15	1	1A12_A	88	150	1A12_C	88	150	5.238	0.51	7

Nota: La presente figura muestra el archivo tsv que contiene la información de diversidad conformacional del conjunto de datos de prueba de las proteínas repetidas empleando la propuesta de método N° 3: Unidades de repetición de los confórmeros. Elaboración propia.

I.3. Comparación entre los resultados obtenidos y los resultados calculados de los métodos genéricos

A través de un juicio crítico por parte de los expertos de proteínas repetidas, la Dra. Layla Hirsh, y de la diversidad conformacional en proteínas, el Dr. Gustavo Parisi y el Dr. Nicolás Palopoli , se optó por no considerar los resultados de los diversos valores de la medida estadística TM-score, que nos proporciona el método genérico que utiliza el software TMAlign, para esta comparación.

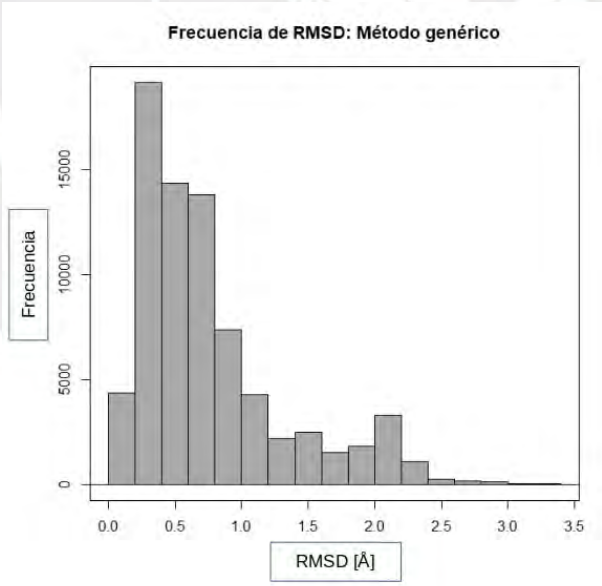
Partiendo con lo mencionado anteriormente y considerando a la medida estadística RMSD como medida principal para la comparación de los resultados, se usó el software RStudio para generar 4 histogramas correspondientes al método genérico que utiliza el software Mammoth para calcular el RMSD de los diferentes pares de confórmeros (Ver Figura I.17), a la propuesta de método N° 1: Región de repetición (Ver Figura I.18), a la propuesta de método N° 2: Unidades de repetición como confórmeros (Ver Figura I.19) y a la propuesta de método N° 3: Unidades de repetición de los confórmeros (Ver Figura I.20). Esto se hizo con la finalidad de observar la frecuencia de los diversos valores de RMSD que existe entre cada uno de los métodos, tanto el genérico como los propuestos.

Antes de detallar los histogramas mencionados previamente para luego brindar una conclusión de la comparación entre estos, hay que tener presente que mientras más cerca esté el valor de RMSD a 0 quiere decir que la distancia que existe entre las estructuras del par de confórmeros utilizados es más pequeña y que la superposición es casi perfecta; caso contrario, si el valor de RMSD se aleja de 0, la distancia que existe entre las estructuras del par de confórmeros utilizados es mayor.

En la Figura I.17 se puede apreciar que las distribuciones de frecuencias se dividen entre valores de 0 y 3.5 de RMSD. Esto significa que hay una gran variedad de diversidad conformacional en el conjunto de datos, debido a que el rango de valores de RMSD es amplio. Asimismo, se puede apreciar que las mayores distribuciones se encuentran entre los valores de 0.25 y 0.75 de RMSD, lo cual denota que en una mayor cantidad de pares de confórmeros la diferencia que existe entre sus estructuras es moderada.

Figure I.17

Histograma de las frecuencias de los diversos valores de RMSD obtenidos de CoDNaS



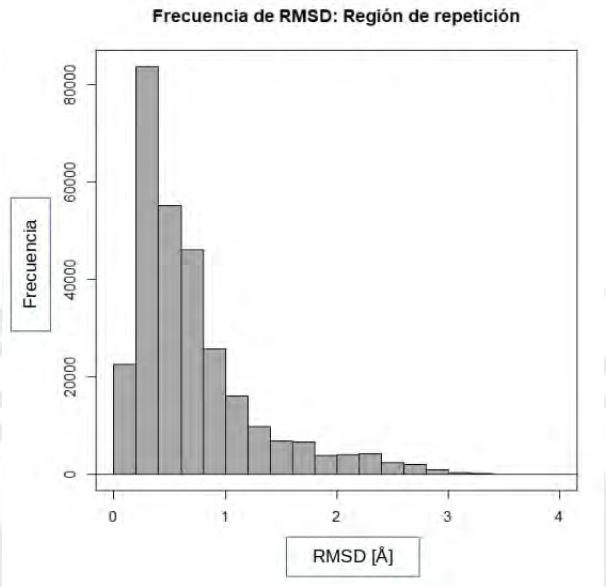
Nota: El gráfico presenta la distribución de las frecuencias de los diversos valores de RMSD de las proteínas repetidas obtenidas de la base de datos CoDNaS. Elaboración propia.

En la Figura I.18 las distribuciones de frecuencias se distribuyen entre los valores de 0 y 3.5 de RMSD. Al tener un amplio rango de valores de RMSD se puede decir que el conjunto de datos presenta una gran variedad de diversidad conformacional. Además, se puede observar que las mayores distribuciones oscilan entre los valores de 0.20 y 0.80 de RMSD, lo cual

señala que la diferencia que existe entre las estructuras de los diferentes pares de confórmeros, ubicados en este rango, es moderada.

Figure I.18

Histograma de las frecuencias de los diversos valores de RMSD obtenidos de la propuesta de método N° 1: Región de repetición



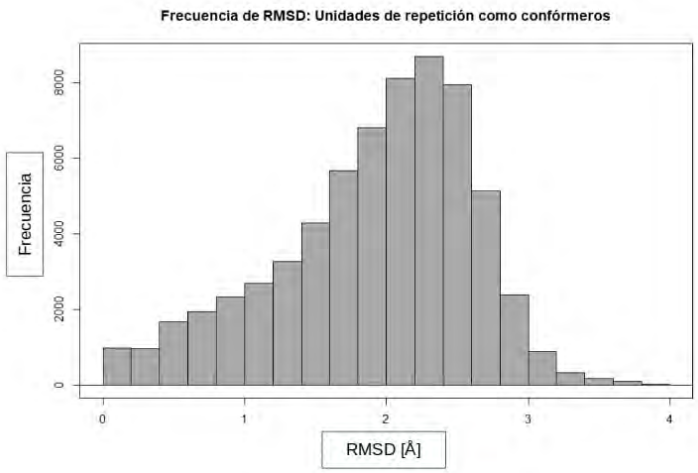
Nota: El gráfico presenta la distribución de las frecuencias de los diversos valores de RMSD de las proteínas repetidas obtenidas empleando la propuesta de método N° 1: Región de repetición. Elaboración propia.

En la Figura I.19 se observa que las distribuciones de frecuencias se dividen entre valores de 0 y 4 de RMSD. Por consiguiente, hay una gran variedad de diversidad conformacional en el conjunto de dato a causa de que el rango de valores de RMSD es amplio. Además, el presente gráfico indica que hay una mayor cantidad de pares de confórmeros que poseen un RMSD mayor 1, lo cual señala que en el conjunto de datos, en su mayor parte, la diferencia que existe a nivel estructural entre cada par de confórmeros es muy significativa.

En la Figura I.20 se puede destacar que una gran parte del conjunto de datos se ubica entre los valores de 0 y 0.25 de RMSD. Esto indica que, en el conjunto de datos, en su mayor parte, la diferencia que hay entre las estructuras de cada par de confórmeros es mínima.

Figure I.19

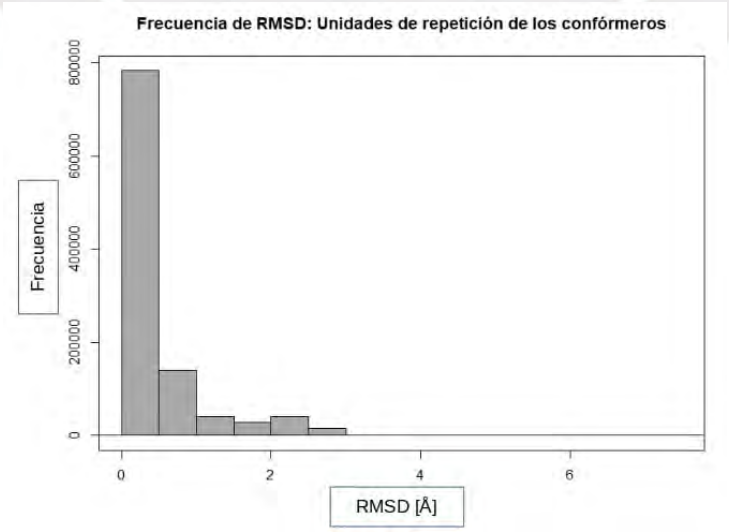
Histograma de las frecuencias de los diversos valores de RMSD obtenidos de la propuesta de método N° 2: Unidades de repetición como confórmeros



Nota: El gráfico presenta la distribución de las frecuencias de los diversos valores de RMSD de las proteínas repetidas obtenidas empleando la propuesta de método N° 2: Unidades de repetición como confórmeros. Elaboración propia.

Figure I.20

Histograma de las frecuencias de los diversos valores de RMSD obtenidos de la propuesta de método N° 3: Unidades de repetición de las conformaciones



Nota: El gráfico presenta la distribución de las frecuencias de los diversos valores de RMSD de las proteínas repetidas obtenidas empleando la propuesta de método N° 3: Unidades de repetición de los confórmeros. Elaboración propia.

En resumen, aplicando 4 métodos, el método genérico que utiliza el software Mammoth y las 3 propuestas de métodos, utilizando como base un mismo conjunto de datos se generaron valores de RMSD por cada par de confórmeros que se pueda formar. Luego, se usó el software RStudio para representar estos resultados obtenidos en 4 histogramas que muestran las frecuencias de los diversos valores de RMSD; uno por cada método. A partir de estos histogramas, se puede apreciar que los valores de RMSD obtenidos por el método genérico (Ver Figura I.17) y la propuesta de método N° 1: Región de repetición (Ver Figura I.18), representan una diferencia moderada entre las diversas estructuras de los pares de conformaciones formados, en su mayor parte. Asimismo, esta diferencia es muy significativa, en su mayor parte, cuando se emplea la propuesta de método N° 2: Unidades de repetición como confórmeros (Ver Figura I.19); y es mínima, en su mayoría, al aplicar la propuesta de método N° 3: Unidades de repetición de los confórmeros (Ver Figura I.20).

Anexo J

Reporte de resultados del método seleccionado

J.1. Introducción

El presente reporte describe la generación de los resultados del método seleccionado sobre el conjunto de datos de prueba de las proteínas repetidas. Asimismo, se detalla paso a paso para poder generar estos resultados y se muestran figuras que muestran evidencia de ello. Y finalmente, se presenta el archivo csv que representa la información recogida a través del método seleccionado.

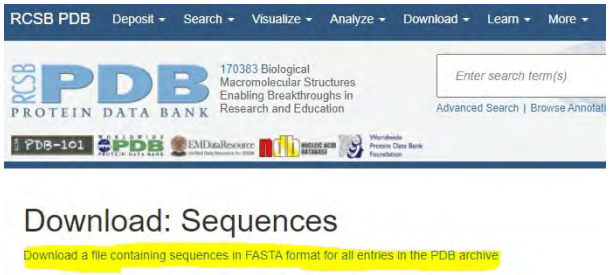
J.2. Generación de los resultados

Para generar los resultados por medio del método seleccionado, primero se tuvo que acceder a la base de datos RCSB PDB para descargar todas las secuencias de las proteínas repetidas en formato FASTA (Ver Figura J.1) a través de la siguiente dirección: *www.rcsb.org/downloads/fasta*.

Al realizar la descarga se obtuvo un archivo en formato zip con las secuencias de todas las proteínas. Este archivo fue utilizado como dato de entrada para el software CD-HIT Suite (Ver Figura J.2), el cual compara todas las secuencias de las proteínas que se le insertan. Esto se realizó con la finalidad de separar cada proteína en un conjunto, denominado clúster, cuyos elementos son similares entre sí.

Figure J.1

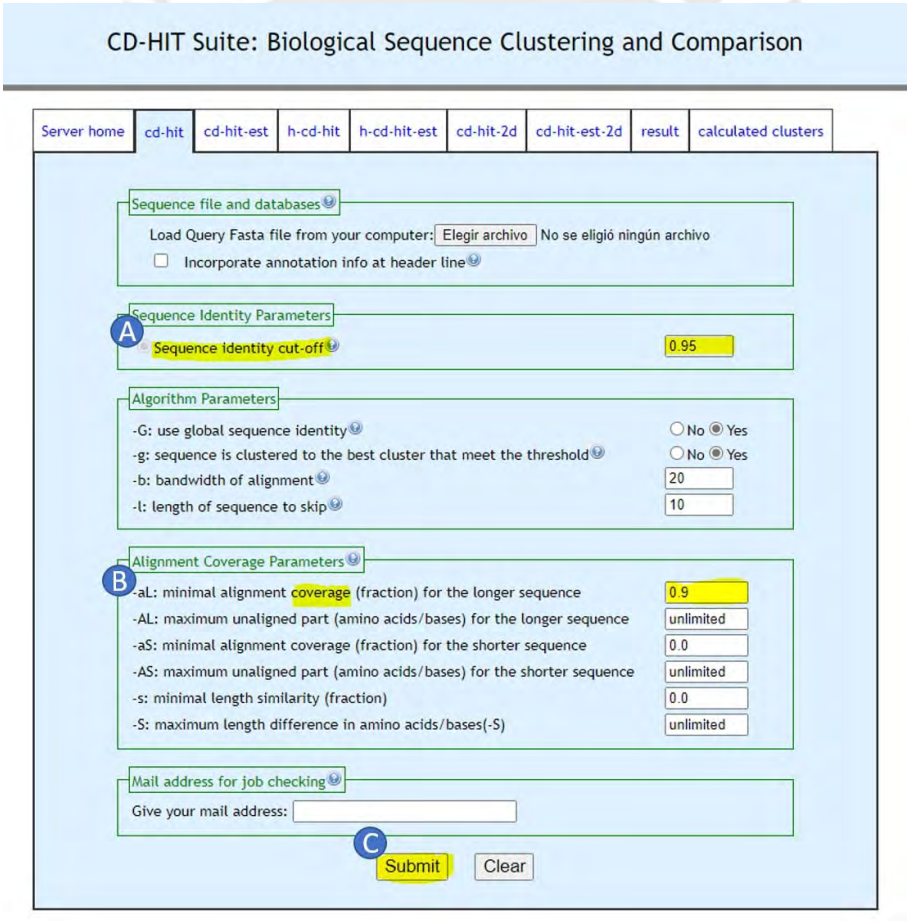
RCSB PDB: Descarga de secuencias de proteínas



Nota: El gráfico presenta la sección de la base de datos RCSB PDB para poder descargar las secuencias de las proteínas en formato FASTA. Elaboración propia.

Figure J.2

Pasos para generar el archivo de clústeres



Nota: A: Se utiliza 95% de secuencia de identidad. B: Se utilizar el 90% de coverage. C: Se presiona submit luego de establecer la secuencia de identidad y el coverage. Elaboración propia.

Luego de utilizar el software CD-HIT Suite, se obtuvo un archivo el cual contiene todas las proteínas separadas en clústeres y se puede encontrar a través de la siguiente dirección: *https://drive.google.com/file/full.out.clstr*. Después, se elaboró un script, el cual se puede ubicar en la siguiente dirección: *https://drive.google.com/file/filtrar_proteinas_repetidas.py*, para identificar los clústeres que poseen alguna proteína repetida del conjunto de datos y para filtrar las proteínas repetidas que tienen más de una conformación. Asimismo, parte de este script mencionado se puede apreciar en la Figura J.3.

Figure J.3

Script para identificar clústeres y filtrar proteinas repetidas

filtrar_proteinas_repetidas.py

```
import os

DIR_PRS = '/mnt/d/Ronaldo/CoDNaS-PRs/DataSet/All/PDBFiles'
FILE_CLUSTERS = '/mnt/d/Ronaldo/CoDNaS-PRs/Resultados/All/ClstrFiles/full.out.clstr'
DIR_TEXT_FILES = '/mnt/d/Ronaldo/CoDNaS-PRs/Resultados/All/TextFiles'

def listarClusters(archivo):
    lista = []
    with open(archivo, 'r') as cluster_file:
        cluster = ''
        lista_pdb = []
        for linea in cluster_file.readlines():
            identificador = linea.split()
            if ('>' in identificador[0]):
                if (cluster != ''):
                    tupla = cluster, lista_pdb
                    lista.append(tupla)
                    cluster = identificador[0][1:] + ' ' + identificador[1]
                    lista_pdb = []
                else:
                    cluster = identificador[0][1:] + ' ' + identificador[1]
            else:
                line = linea.split()
                pdb = line[2][1:len(line[2])-3]
                lista_pdb.append(pdb)
        tupla = cluster, lista_pdb
        lista.append(tupla)
    return lista

def listarPRs(directorio):
    return os.listdir(directorio)

def listarPRsFiltradas(lista_prs, lista_clusters):
    lista = []
    for pr in lista_prs:
        pr_minus = pr[:4].lower() + pr[4:len(pr)]
        for cluster in lista_clusters:
            if (pr_minus in cluster[1]):
                if (pr not in lista):
                    lista.append(pr)
                break
    return lista
```

Nota: La presente figura muestra parte del script que identifica clústeres con alguna proteína repetida y que filtra a las proteínas repetidas que tienen más de una conformación. Elaboración propia.

Al tener las proteínas filtradas y los clústeres identificadas, se utilizó el software Mammoth para alinear cada proteína repetida con sus respectivas conformaciones extraídas del clúster a la que pertenece. Esto se hizo con la finalidad de identificar la región repetida de la proteína repetida en cada conformación y para esto se elaboró un script, el cual se puede apreciar en la Figura J.4 y se puede encontrar en la siguiente ruta: *https://drive.google.com/file/ident_reg.py*

Figure J.4

Script para identificar las regiones repetidas de las conformaciones de la proteína repetida

identificar_regiones.py

```
def alinearEstructuras(path_conformer_1, path_conformer_2, conformer, i):
    ubicacion = './SupFiles/' + conformer + '/' + conformer + '_reg_' + str(i) + '.sup'
    subprocess.call(['./TMalign', path_conformer_1, path_conformer_2, '-a', 'T', '-o', ubicacion])

def limpiar(conformer):
    directorio = './SupFiles/' + conformer
    lista = os.listdir(directorio)
    l = []
    for item in lista:
        l.append('./SupFiles/' + conformer + '/' + item)
    if (len(l) > 15):
        subprocess.call(['rm', l[1], l[2], l[3], l[4], l[5], l[6], l[7], l[8], l[9], l[10], l[11],
            l[12], l[13], l[14], l[16], l[17], l[18], l[19], l[20], l[21], l[22], l[23], l[24], l[25], l[26], l[27], l[28], l[29]])
    else:
        subprocess.call(['rm', l[1], l[2], l[3], l[4], l[5], l[6], l[7], l[8], l[9], l[10], l[11],
            l[12], l[13], l[14]])

def identificarRegiones(lista_conformeros_filtrados):
    archivo = open('./TextFiles/conformers_sin_regiones_ubicadas.txt', 'w')
    for tupla in lista_conformeros_filtrados:
        lista = os.listdir(DIR_PROTEINAS_REPETIDAS + '/' + tupla[0])
        lista_reg = []
        for reg in lista:
            if ('unit' not in reg):
                lista_reg.append(reg)
        for conformer in tupla[2]:
            #if (conformer != tupla[0]):
            crearDirectorios('./SupFiles/' + conformer)
            i = 1
            for reg in lista_reg:
                conformer_1 = DIR_PROTEINAS_REPETIDAS + '/' + tupla[0] + '/' + reg
                conformer_2 = DIR_PDB_FILES + '/' + conformer + '.pdb'
                alinearEstructuras(conformer_1, conformer_2, conformer, i)
                i = i + 1
            try:
                limpiar(conformer)
            except:
                archivo.write(conformer)
                archivo.write('\n')
    archivo.close()
```

Nota: La presente figura muestra parte del script que identifica las regiones repetidas de la proteína repetida en las conformaciones de la misma. Elaboración propia.

Después de identificar las regiones repetidas de cada confórmero de cada proteína repetida se elaboró un script que permite generar los archivos en formato pdb (pdb files) de los mismos. Parte de este script se puede apreciar en la Figura J.5 y se puede visualizar en la siguiente dirección: *https://drive.google.com/file/generar_regiones_conformers.py*

Figure J.5

Script para generar las regiones repetidas en las conformaciones de las proteínas repetidas

generar_regiones_conformers.py

```
def generarConformacionesRegiones(lista_regiones):
    for tupla in lista_regiones:
        generarPDBFile(tupla[0], tupla[1], tupla[2])

def crearDirectorios(directorio):
    os.makedirs(directorio, exist_ok = True)

def generarPDBFile(conformer, lim_inf, lim_sup):
    directorio = DIR_CONFORMERS + '/' + conformer
    crearDirectorios(directorio)
    name_out_file = directorio + '/' + conformer + '_' + lim_inf + '_' + lim_sup + '.pdb'
    out_file = open(name_out_file, 'w')
    is_final = 0
    archivo = open(DIR_PDB_FILES + '/' + conformer + '.pdb', 'r')
    for linea in archivo.readlines():
        line = linea.split()
        if (line[0] == 'ATOM'):
            if (len(line[4]) == 1):
                try:
                    if (line[4] == conformer[5:6]):
                        if (int(lim_inf) <= int(line[5]) and int(line[5]) <= int(lim_sup)):
                            is_final = 1
                            out_file.write(linea)
                        elif (is_final):
                            out_file.write('TER')
                            break
                except:
                    if (line[5] == conformer[5:6]):
                        if (int(lim_inf) <= int(line[6]) and int(line[6]) <= int(lim_sup)):
                            is_final = 1
                            out_file.write(linea)
                        elif (is_final):
                            out_file.write('TER')
                            break
            else:
                identifier = line[4][1:len(line[4])]
                if (line[4][0] == conformer[5:6]):
                    if (int(lim_inf) <= int(identifier) and int(identifier) <= int(lim_sup)):
                        is_final = 1
                        out_file.write(linea)
                    elif (is_final):
                        out_file.write('TER')
                        break
        elif (line[0] == 'TER' and is_final):
            out_file.write('TER')
            break
    archivo.close()
    out_file.close()
```

Nota: La presente figura muestra parte del script que genera los pdb files de las regiones repetidas de las proteínas repetidas en las conformaciones de la misma. Elaboración propia.

Es así que, teniendo los pdb files del nuevo conjunto de confórmeros de cada proteína repetida delimitada por sus regiones de repetición. Se procedió a formar la máxima cantidad de pares de confórmeros por cada proteína repetida para luego utilizar el software Mammoth para superposicionar las estructuras de cada par y calcular la diversidad conformacional a través

de la medidad estadística RMSD. Y para realizar lo mencionado recientemente, se elaboró un script que se puede encontrar en la siguiente ruta: *https://drive.google.com/file/gen_res.py* y parte del código se puede apreciar en la Figura J.6.

Figure J.6

Script para generar los resultados de diversidad conformacional de cada proteína repetida

resultados_rmsd.py

```
def listarResultados(lista_conformers):
  lista = []
  i = 0
  num_conf = 0
  laxus = os.listdir(DIR_CONFORMERS_FILES)
  for tupla in lista_conformers:
    num_conf = num_conf + 1
    sublista = []
    pares = list(itertools.combinations(tupla[1], 2))
    for pair in pares:
      i = i + 1
      conf_1, conf_2 = pair
      if (conf_1 in laxus and conf_2 in laxus):
        l_conf_1 = os.listdir(DIR_CONFORMERS_FILES + '/' + conf_1)
        l_conf_2 = os.listdir(DIR_CONFORMERS_FILES + '/' + conf_2)
        for j in range(len(l_conf_1)):
          conformer_1 = DIR_CONFORMERS_FILES + '/' + conf_1 + '/' + l_conf_1[j]
          arr_1 = l_conf_1[j].split('_')
          lim_inf_1 = arr_1[2]
          arr_1 = arr_1[3].split('.')
          lim_sup_1 = arr_1[0]
          conformer_2 = DIR_CONFORMERS_FILES + '/' + conf_2 + '/' + l_conf_2[j]
          arr_2 = l_conf_2[j].split('_')
          lim_inf_2 = arr_2[2]
          arr_2 = arr_2[3].split('.')
          lim_sup_2 = arr_2[0]
          print(str(i) + ', ' + conf_1 + ' - ' + conf_2)
          calcularRMSD(conformer_1, conformer_2)
          rmsd_file = open('./TextFiles/output_rmsd_vf_14.txt', 'r')
          for linea in rmsd_file.readlines():
            if ('Sstr(LG)' in linea):
              line = linea.split()
              rmsd = line[7]
            if ('Seq.sim=' in linea):
              line = linea.split()
              num_region = j + 1
              par = conf_1, lim_inf_1, lim_sup_1, conf_2, lim_inf_2, lim_sup_2, line[3], rmsd, num_region
              sublista.append(par)
              break
    cluster = tupla[0].split()
    cluster = cluster[1]
    item = cluster, sublista
    lista.append(item)
  return lista

def calcularRMSD(conformer_1, conformer_2):
  subprocess.call(['./mymammoth', '-p', conformer_1, '-e', conformer_2, '-o', './TextFiles/output_rmsd_vf.txt'])

def generarCSVResultados(lista_resultados):
  with open('./CsvFiles/resultados_rmsd_vf.csv', 'w') as out_file:
    csv_writer = csv.writer(out_file, delimiter = ',', lineterminator = '\n')
    csv_writer.writerow(['cluster', 'conformero_1', 'lim_inf_1', 'lim_sup_1', 'conformero_2', 'lim_inf_2',
      'lim_sup_2', 'sec_similitud', 'rmsd', 'region'])
    for row in lista_resultados:
      cluster, sublista = row
      for tupla in sublista:
        csv_writer.writerow([cluster, tupla[0], tupla[1], tupla[2], tupla[3], tupla[4], tupla[5],
          tupla[6], tupla[7], tupla[8]])
```

Nota: La presente figura muestra parte del script que permite generar los resultados de diversidad conformacional a través del RMSD de cada proteína repetida. Elaboración propia.

Finalmente, se generó un archivo Csv que registra los cálculos de la diversidad conformacional sobre el conjunto de proteínas repetidas empleando el método seleccionado. Asimismo, este archivo se puede encontrar en la siguiente ruta: *https://drive.google.com/file/res_rmsd.csv* y se puede apreciar en la Figura J.7.

Figure J.7

Resultados obtenidos empleado el método seleccionado

	A	B	C	D	E	F	G	H	I	J
1	cluster	conformero_1	lim_inf_1	lim_sup_1	conformero_2	lim_inf_2	lim_sup_2	sec_similitud	rmsd	region
2	1	1A0R_B	55	340	1B9X_A	55	340	5.257	0.39	1
3	1	1A0R_B	55	340	1B9Y_A	55	340	5.257	0.44	1
4	1	1A0R_B	55	340	1GG2_B	55	340	5.392	1.12	1
5	1	1A0R_B	55	340	1GOT_B	55	340	5.302	1.08	1
6	1	1A0R_B	55	340	1GP2_B	55	340	5.392	1.09	1
7	1	1A0R_B	55	340	1OMW_B	55	340	5.392	1.14	1
8	1	1A0R_B	55	340	1TBG_A	55	340	5.392	1.09	1
9	1	1A0R_B	55	340	1TBG_B	55	340	5.392	1.09	1
10	1	1A0R_B	55	340	1TBG_C	55	340	5.392	1.06	1
11	1	1A0R_B	55	340	1TBG_D	55	340	5.392	1.05	1
12	1	1A0R_B	55	340	1XHM_A	55	340	5.392	1.11	1
13	1	1A0R_B	55	340	2BCJ_B	55	340	5.392	1.15	1
14	1	1A0R_B	55	340	2TRC_B	55	340	5.257	0.39	1
15	1	1A0R_B	55	340	3AH8_B	55	340	5.392	1.13	1

Nota: La presente figura muestra el archivo csv que contiene información de diversidad conformacional luego de emplear el método seleccionado. Elaboración propia.

Anexo K

Documento del modelamiento de la estructura de base de datos

K.1. Introducción

El presente documento describe la definición de las tablas que se utilizarán en la elaboración del modelo relacional de la base de datos. Posteriormente, se detalla el modelamiento de la estructura de la base de datos a través del modelo relacional. Y finalmente, se presenta el modelo relacional de la base de datos, la cual almacenará la información obtenida a través del análisis de diversidad conformacional de las proteínas repetidas.

K.2. Definición de tablas a utilizar en el modelo relacional

Para definir las tablas que se utilizarán para el modelo relacional de la base de datos se tomó en cuenta los datos que se van a obtener a través del análisis de diversidad conformacional. Estos datos vienen a ser la información general, la información estructural y las diversas conformaciones de las proteínas repetidas.

Teniendo en cuenta lo descrito en el párrafo anterior, se definieron tres tablas y estas se pueden apreciar en la Figura K.1.

Figure K.1

Definición de las tablas a utilizar en el modelo relacional

A

info_estructural	
Columna	**Tipo de dato**
cluster	int
region	int
num_conformaciones	int
rmsd_min	double
rmsd_max	double
rmsd_avg	double

B

info_general	
Columna	**Tipo de dato**
pdb_id	varchar(10)
cluster	int
nombre_proteina	varchar(300)
titulo	varchar(500)
organismo	varchar(300)
long_secuencia	int
clasificacion	varchar(300)
num_regiones	int

C

conformación	
Columna	**Tipo de dato**
conformero_1	varchar(10)
conformero_2	varchar(10)
lim_inf_1	int
lim_sup_1	int
lim_inf_2	int
lim_sup_2	int
sec_similitud	double
rmsd	double
region	int

Nota: A: La tabla info_estructural contiene los datos de la información estructural de la proteína repetida. B: La tabla info_general contiene los datos de la información general de la proteína repetida. C: La tabla conformación contiene los datos de las diversas conformaciones de la proteína repetida. Elaboración propia.

K.3. Elaboración del modelo relacional de la estructura de base de datos

Para elaborar el modelo relacional de la estructura de base de datos, primero se tuvo que acceder a la herramienta de visualización de base de datos MySQL Workbench para crear un nuevo "schema"[1] (Ver Figura K.2) con el nombre de codnas-prs. Luego de crear el schema, se realizó la "ingeniería inversa" con la finalidad de poder elaborar el modelo relacional de la base de datos creada.

Al ejecutar la ingeniería inversa, se accedió a una ventana la cual brinda una variedad de opciones para la vista del modelo, el schema físico, entre otros (Ver Figura K.3). Sin embargo, para la elaboración del modelo relacional de la base de datos se usaron las opciones de "Añadir Diagrama" y "Añadir Tabla" (Ver Figura K.4).

La primera opción que se utilizó fue la de "Añadir Tabla" para agregar las tablas que se definieron en la sección K.2 y, posteriormente, se usó la opción "Añadir Diagrama" para utilizar

[1]La base de datos es un conjunto de tablas relacionadas y para MySQL Workbench, esto es conocido como schema.

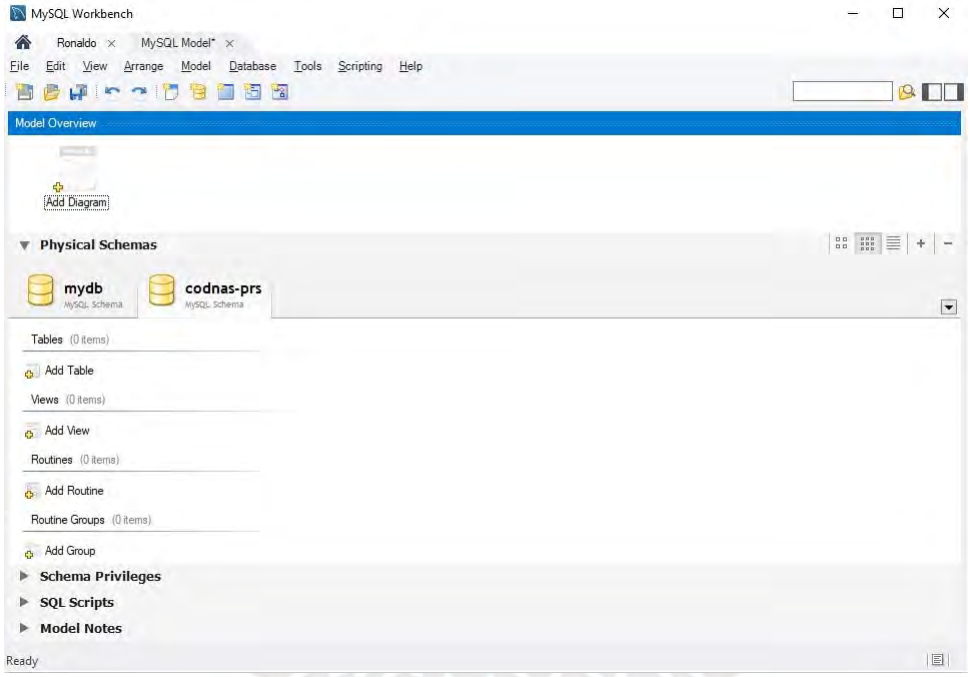
Figure K.2

Pasos para crear un nuevo schema

Nota: El gráfico muestra los pasos para crear un nuevo schema. Elaboración propia.

Figure K.3

Ventana de la Ingeniería Inversa



Nota: El gráfico muestra la sección de la Ingeniería Inversa. Elaboración propia.

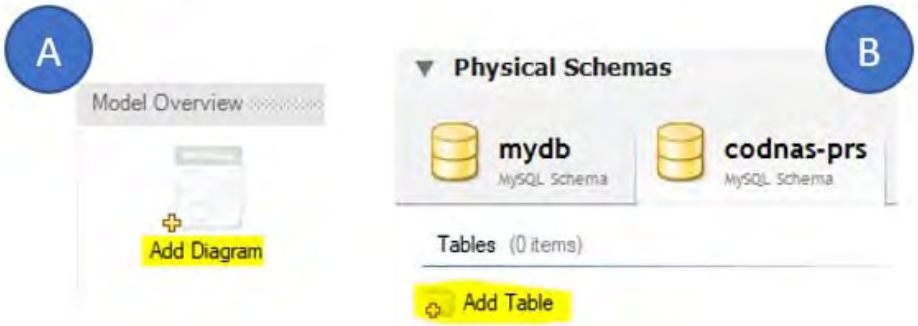
las tablas agregadas y relacionarlas. Es así que llevando a cabo estos pasos descritos se realizó el modelamiento de la estructura de la base de datos y esto se puede apreciar en la Figura K.5.

K.4. Modelo relacional de la base de datos

A continuación, se presenta el modelo relacional de la base de datos en la Figura K.6.

Figure K.4

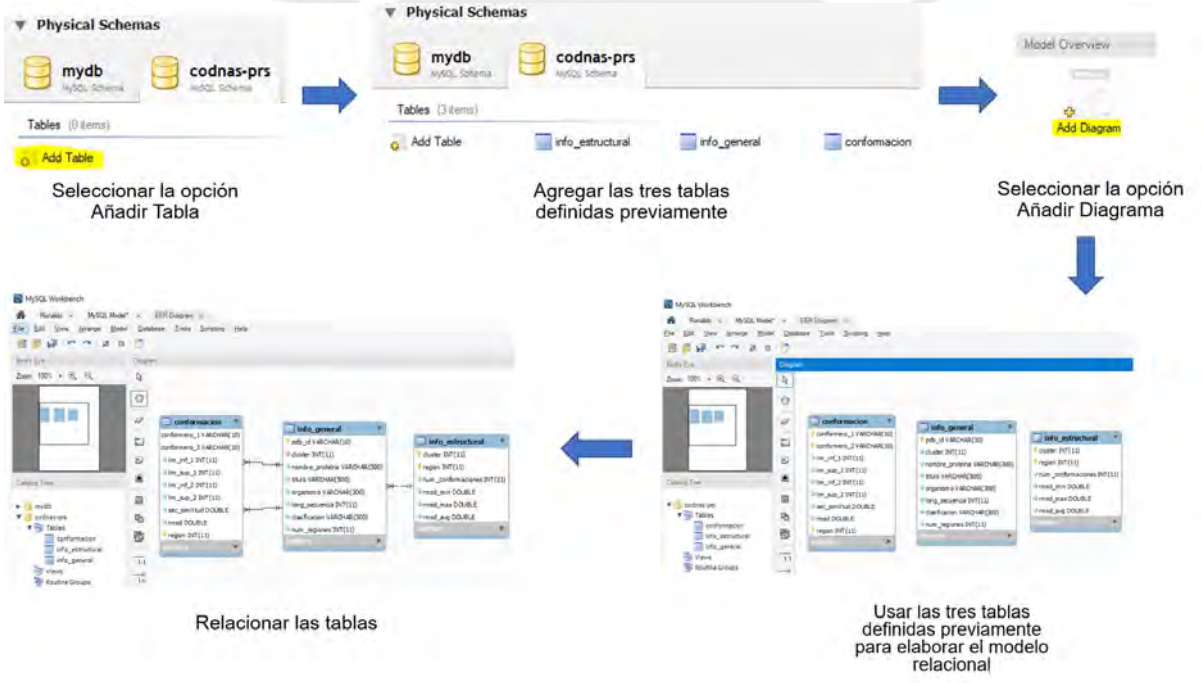
Las opciones Añadir Diagrama y Añadir Tabla



Nota: A: La opción Añadir Diagrama. B: La opción Añadir Tabla. Elaboración propia.

Figure K.5

Pasos para elaborar el modelo relacional de la base de datos



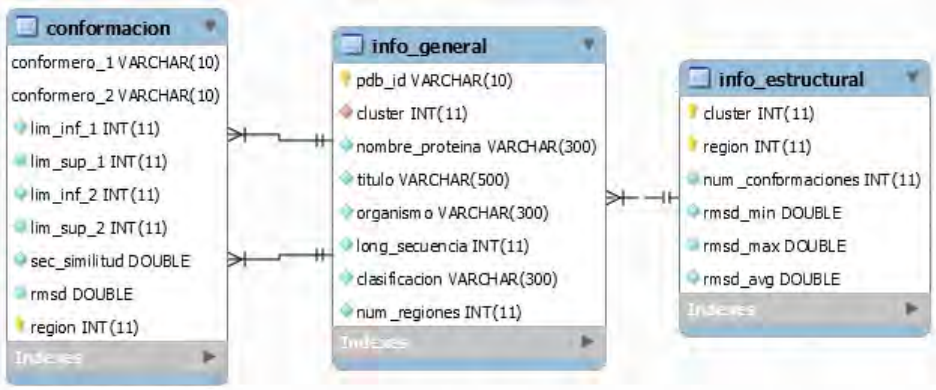
Nota: El gráfico muestra los pasos para elaborar el modelo relacional de la base de datos. Elaboración propia.

K.5. Script de creación de la base de datos

En la presente sección se muestra el script completo de la creación de la base de datos.

Figure K.6

Modelo relacional de la base de datos



Nota: El gráfico muestra el modelo relacional de la base de datos. Elaboración propia.

Figure K.7

Script de creación de la base de datos

```
-- MySQL Workbench Forward Engineering

SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';

-- -----------------------------------------------------
-- Schema mydb
-- -----------------------------------------------------
-- -----------------------------------------------------
-- Schema codnas-prs
-- -----------------------------------------------------

-- -----------------------------------------------------
-- Schema codnas-prs
-- -----------------------------------------------------
CREATE SCHEMA IF NOT EXISTS `codnas-prs` DEFAULT CHARACTER SET utf8 ;
USE `codnas-prs` ;

-- -----------------------------------------------------
-- Table `codnas-prs`.`info_estructural`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `codnas-prs`.`info_estructural` (
  `cluster` INT(11) NOT NULL,
  `region` INT(11) NOT NULL,
  `num_conformaciones` INT(11) NOT NULL,
  `rmsd_min` DOUBLE NOT NULL,
  `rmsd_max` DOUBLE NOT NULL,
  `rmsd_avg` DOUBLE NOT NULL,
  PRIMARY KEY (`cluster`, `region`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8;


-- -----------------------------------------------------
-- Table `codnas-prs`.`info_general`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `codnas-prs`.`info_general` (
  `pdb_id` VARCHAR(10) CHARACTER SET `utf8` COLLATE `utf8_bin` NOT NULL,
  `cluster` INT(11) NOT NULL,
  `nombre_proteina` VARCHAR(300) NOT NULL,
  `titulo` VARCHAR(500) NOT NULL,
  `organismo` VARCHAR(300) NOT NULL,
  `long_secuencia` INT(11) NOT NULL,
  `clasificacion` VARCHAR(300) NOT NULL,
  `num_regiones` INT(11) NOT NULL,
  PRIMARY KEY (`pdb_id`),
  INDEX `fk_cluster_gen_est_idx` (`cluster` ASC) VISIBLE,
  CONSTRAINT `fk_cluster_gen_est`
    FOREIGN KEY (`cluster`)
    REFERENCES `codnas-prs`.`info_estructural` (`cluster`)
    ON DELETE CASCADE
    ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8;

-- -----------------------------------------------------
-- Table `codnas-prs`.`conformacion`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `codnas-prs`.`conformacion` (
  `conformero_1` VARCHAR(10) CHARACTER SET `utf8` COLLATE `utf8_bin` NOT NULL,
  `conformero_2` VARCHAR(10) CHARACTER SET `utf8` COLLATE `utf8_bin` NOT NULL,
  `lim_inf_1` INT(11) NOT NULL,
  `lim_sup_1` INT(11) NOT NULL,
  `lim_inf_2` INT(11) NOT NULL,
  `lim_sup_2` INT(11) NOT NULL,
  `sec_similitud` DOUBLE NOT NULL,
  `rmsd` DOUBLE NOT NULL,
  `region` INT(11) NOT NULL,
  PRIMARY KEY (`conformero_1`, `conformero_2`, `region`),
  INDEX `fk_conformero_2_conf_gen_idx` (`conformero_2` ASC) VISIBLE,
  CONSTRAINT `fk_conformero_1_conf_gen`
    FOREIGN KEY (`conformero_1`)
    REFERENCES `codnas-prs`.`info_general` (`pdb_id`)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
  CONSTRAINT `fk_conformero_2_conf_gen`
    FOREIGN KEY (`conformero_2`)
    REFERENCES `codnas-prs`.`info_general` (`pdb_id`)
    ON DELETE CASCADE
    ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8;


SET SQL_MODE=@OLD_SQL_MODE;
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```

Nota: El gráfico muestra el script de creación de la base de datos. Elaboración propia.

Anexo L

Documento de arquitectura del servicio web

L.1. Introducción

En el presente documento se describe la manera en cómo se elaboró la arquitectura del servicio web en base al diagrama de componente y al diagrama de despliegue.

L.2. Elaboración de la arquitectura

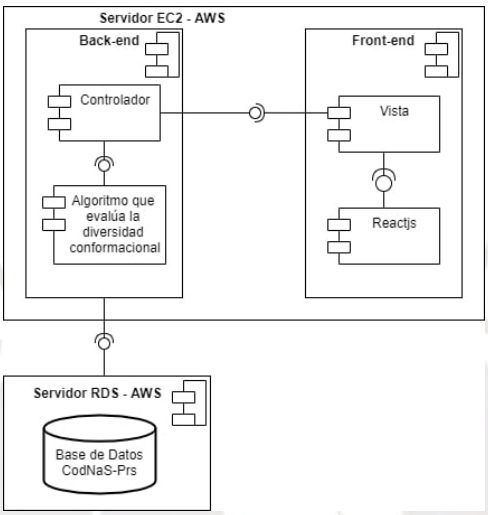
La arquitectura del servicio web está representado por el diagrama de despliegue y el diagrama de componentes. Para elaborar estos diagramas se utilizó la herrmamienta draw.io, la cual se puede acceder a través de la siguiente dirección: *https://app.diagrams.net/*.

Teniendo en cuenta lo anterior, para elaborar el diagrama de componentes (Ver Figura L.1), se utilizó el patrón de arquitectura modelo-vista-controlador (MVC), ya que separar las funciones usando este patrón permite que el flujo de la información sea más rápida, asimismo, usando el patrón MVC, permite desarrollar tanto el servicio web como la interfaz de usuario de manera rápida, modular y mantenible. Asimismo, para implementar este módulo se está contando con un back-end que utilizará Flask para la comunicación con la base de datos y para el front-end se utilizará la librería Reactjs. Además, el algoritmo que permitirá evaluar la diversidad conformacional de las proteínas repetidas se realizará en Python, ya que este lenguaje de programación cuenta con las librerías que van a facilitar la elaboración del mismo.

Por otro lado, para elaborar el diagrama de despliegue (Ver Figura L.2) se determinó en utilizar servidores en la nube de AWS con base de datos en MySQL, versión 5.7.28, con la finalidad que los científicos puedan acceder a esta información de manera rápida y porque el presente cuenta con experiencia utilizando los servicios que brinda AWS.

Figure L.1

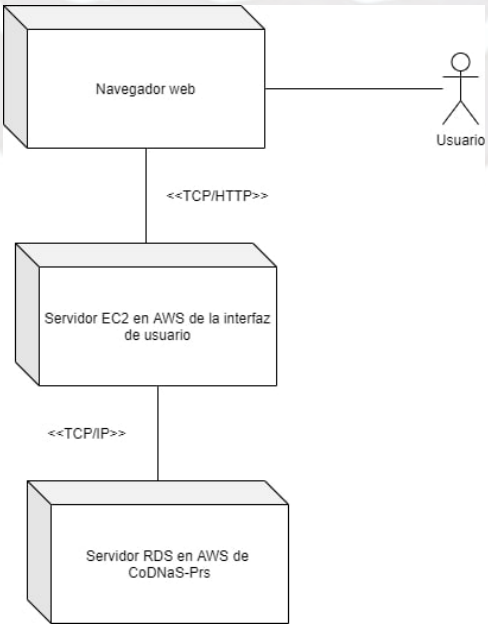
Diagrama de componentes



Nota: El gráfico muestra el diagrama de componentes. Elaboración propia.

Figure L.2

Diagrama de despliegue



Nota: El gráfico muestra el diagrama de despliegue. Elaboración propia.

Anexo M

Informe de pruebas funcionales del servicio web

M.1. Introducción

El presente informe describe y muestra las pruebas funcionales del servicio web, el cual hará uso la interfaz de usuario para obtener o estimar la diversidad conformacional de las proteínas repetidas. Y finalmente, se detalla los resultados obtenidos de estas pruebas funcionales.

M.2. Elaboración de las dos pruebas funcionales

Para crear las pruebas funcionales del servicio web se utilizó la herramienta Postman, el cual nos brinda una sección particular para crear una colección (Ver Figura M.1) con la finalidad de añadir todas las solicitudes y verificar que estas funcionen correctamente.

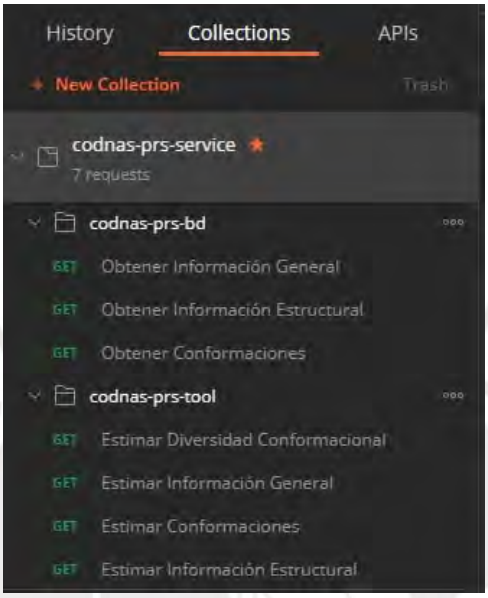
Se elaboraron 7 servicios o funcionalidades y se separaron en dos grupos: servicios para acceder y obtener la información de diversidad conformacional de las proteínas repetidas de la base de datos; y servicios para estimar y obtener la información de diversidad conformacional de las proteínas repetidas.

Para el primer grupo de servicios, llamado codnas-prs-bd, el cual contiene a los servicios de Obtener Información General, Obtener Información Estructural y Obtener Conformaciones, se realizaron las siguientes pruebas que se pueden apreciar en la Figura M.2, la Figura M.3 y la

Figura M.4, respectivamente. Estas pruebas permitirán demostrar que los servicios elaborados no contienen algún error.

Figure M.1

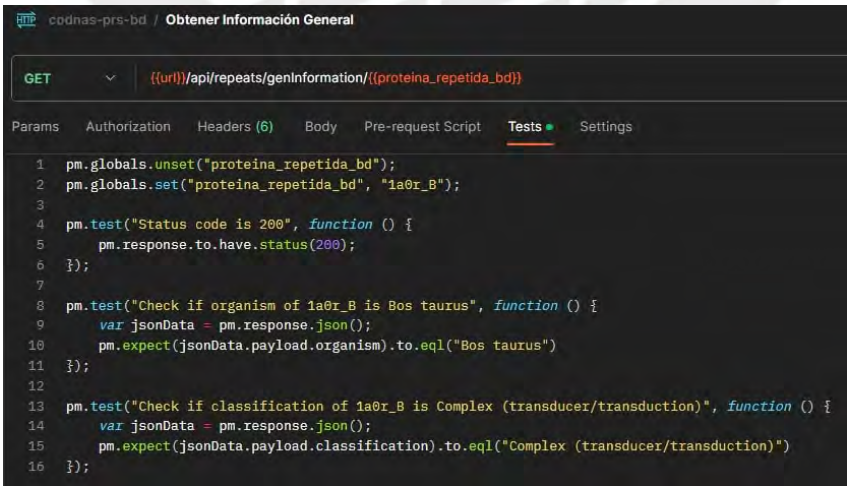
Colección codnas-prs-service del servicio web



Nota: El gráfico muestra la colección codnas-prs-service, el cual está dividido en dos grupos: codnas-prs-bd y codnas-prs-tool. Elaboración propia.

Figure M.2

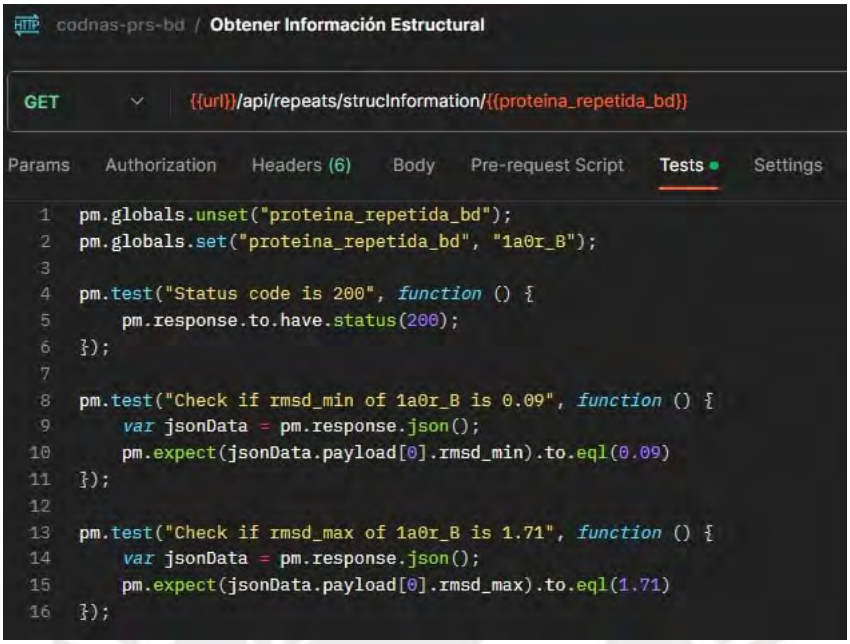
Script para comprobar funcionamiento del servicio Obtener Información General



Nota: El gráfico muestra el script que se utilizó para comprobar el funcionamiento del servicio Obtener Información General. Elaboración propia.

Figure M.3

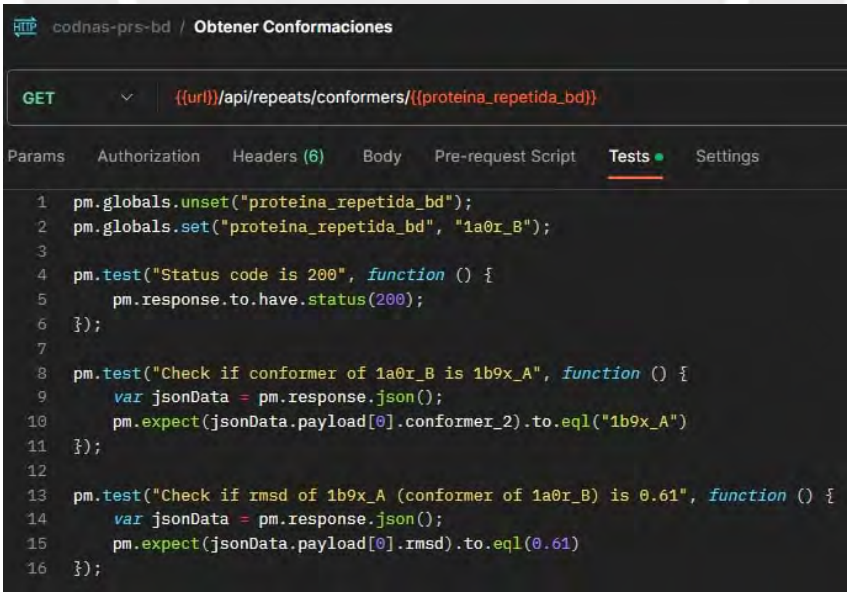
Script para comprobar funcionamiento del servicio Obtener Información Estructural



Nota: El gráfico muestra el script que se utilizó para comprobar el funcionamiento del servicio Obtener Información Estructural. Elaboración propia.

Figure M.4

Script para comprobar funcionamiento del servicio Obtener Conformaciones

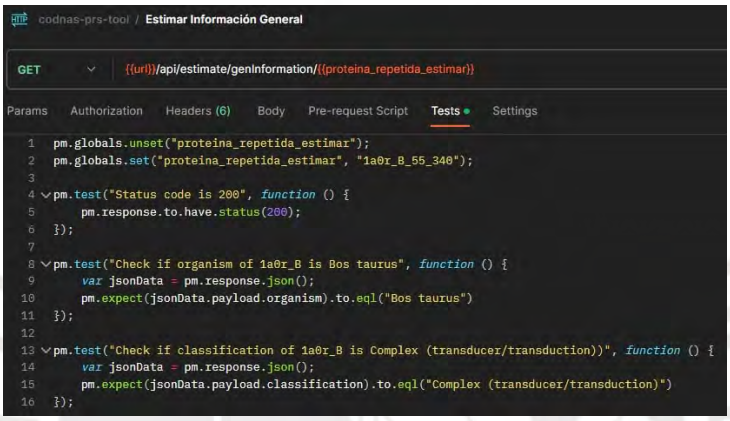


Nota: El gráfico muestra el script que se utilizó para comprobar el funcionamiento del servicio Obtener Conformaciones. Elaboración propia.

Para el segundo grupo de servicios, llamado codnas-prs-tool, el cual contiene a los servicios de Estimar Información General, Estimar Información Estructural, Estimar Conformaciones y Estimar Diversidad Conformacional, se realizaron las siguientes pruebas que se pueden apreciar en la Figura M.5, la Figura M.6, la Figura M.7 y la Figura M.8, respectivamente. Estas pruebas permitirán demostrar que los servicios elaborados cumplen con la metodología seleccionada pra estimar la diversidad conformacional de las proteínas repetidas.

Figure M.5

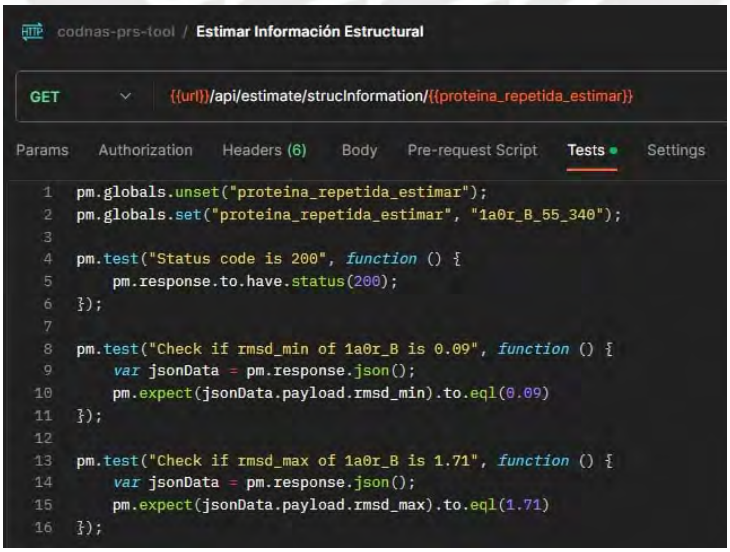
Script para comprobar funcionamiento del servicio Estimar Información General



Nota: El gráfico muestra el script que se utilizó para comprobar el funcionamiento del servicio Estimar Información General. Elaboración propia.

Figure M.6

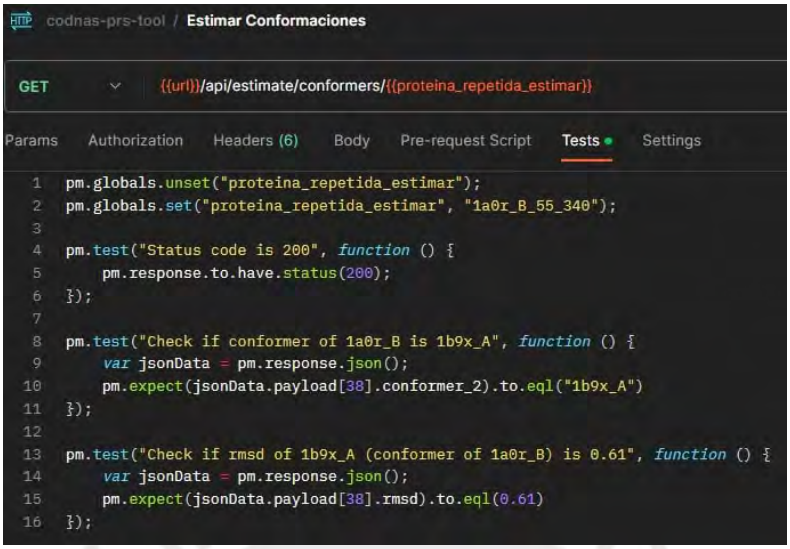
Script para comprobar funcionamiento del servicio Estimar Información Estructural



Nota: El gráfico muestra el script que se utilizó para comprobar el funcionamiento del servicio Estimar Información Estructural. Elaboración propia.

Figure M.7

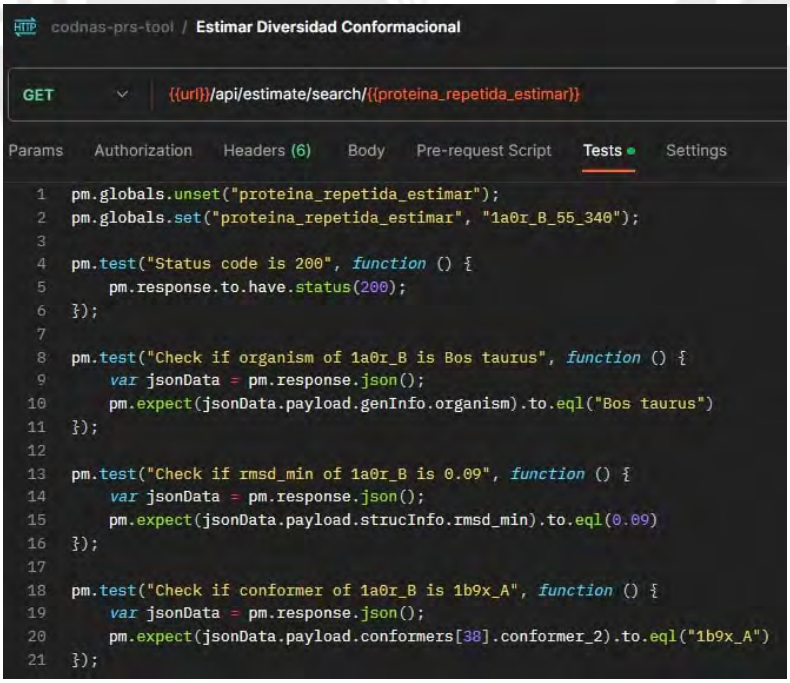
Script para comprobar funcionamiento del servicio Estimar Conformaciones



Nota: El gráfico muestra el script que se utilizó para comprobar el funcionamiento del servicio Estimar Conformaciones. Elaboración propia.

Figure M.8

Script para comprobar funcionamiento del servicio Estimar Diversidad Conformacional



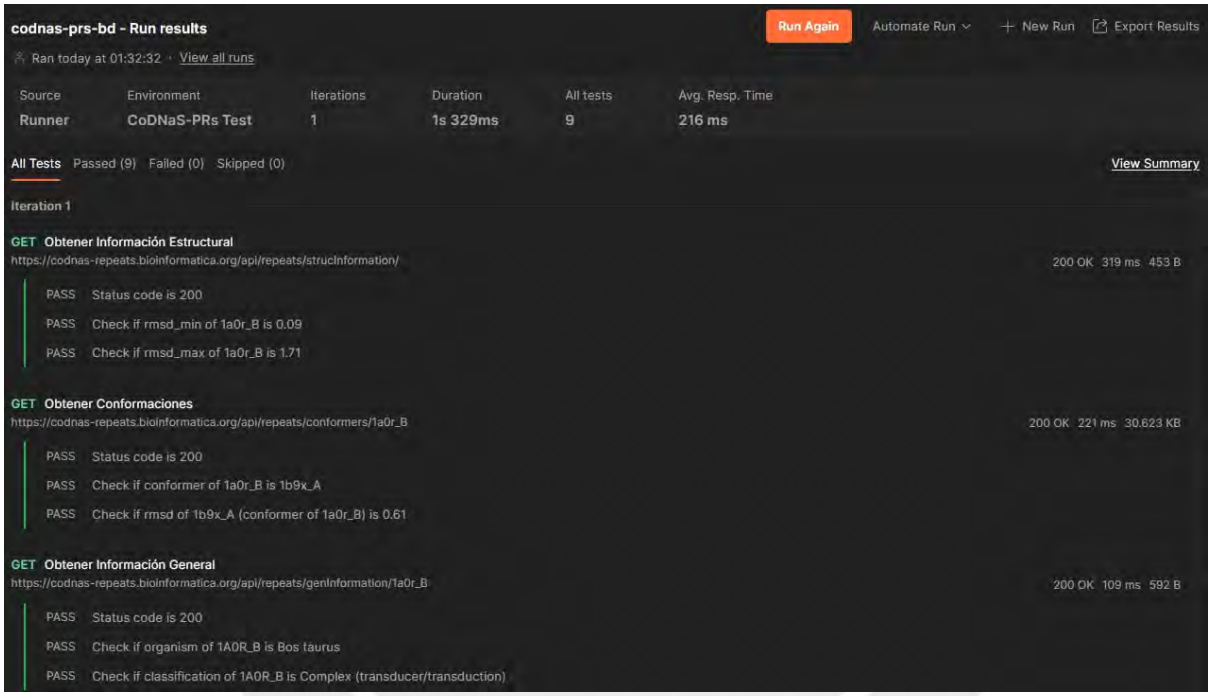
Nota: El gráfico muestra el script que se utilizó para comprobar el funcionamiento del servicio Estimar Diversidad Conformacional. Elaboración propia.

M.3. Resultados de las pruebas funcionales

A continuación, se presentan los resultados de las pruebas funcionales realizadas a los grupos de servicios. En la Figura M.9 se aprecian los resultados satisfactorios de las pruebas realizadas a los servicios del grupo codnas-prs-bd (Ver Figura M.1). Para este caso, se realizaron 3 pruebas para cada servicio del grupo.

Figure M.9

Resultados de los test para cada servicio del grupo de servicios codnas-prs-bd

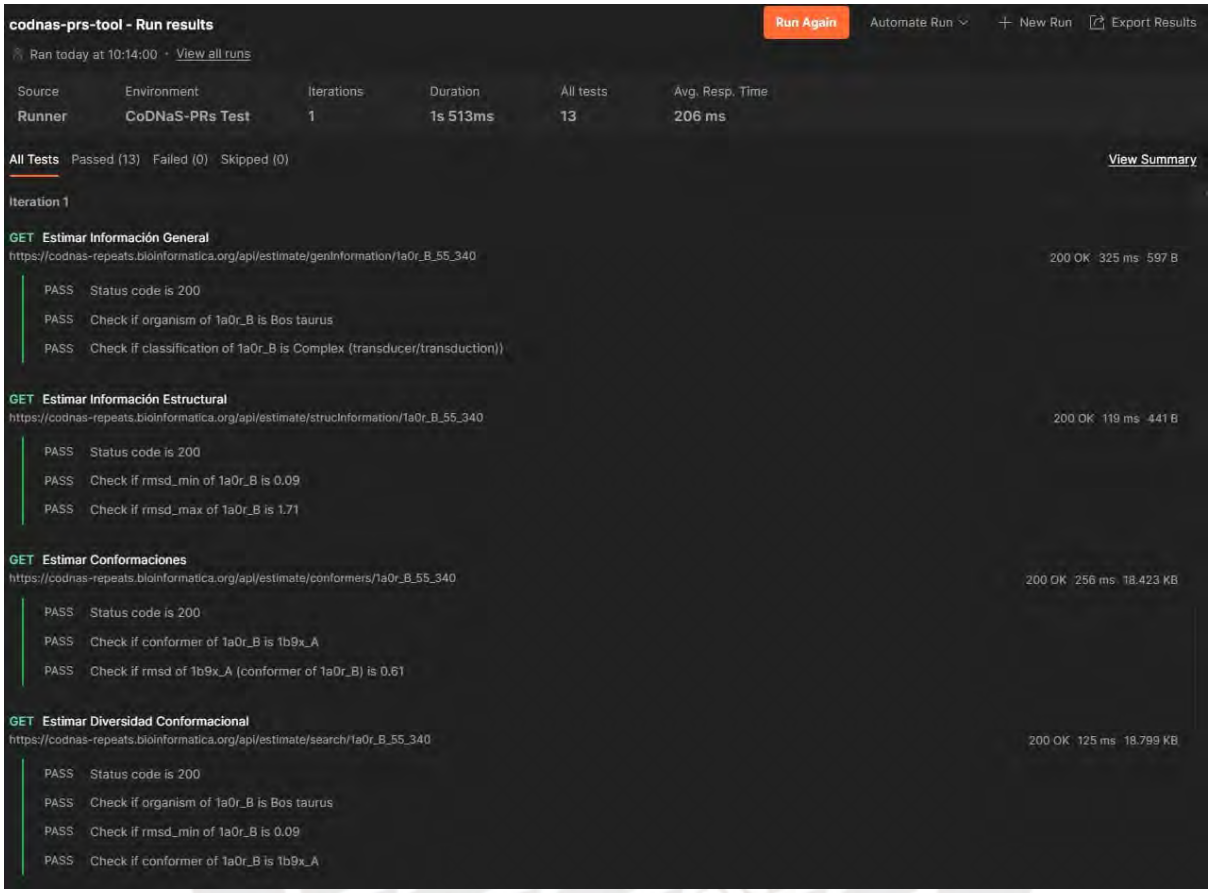


Nota: El gráfico muestra los resultados satisfactorios de las pruebas realizadas para cada servicio del grupo codnas-prs-bd. Elaboración propia.

Además, en la Figura M.10 se aprecian los resultados satisfactorios de las pruebas realizadas a los servicios del grupo codnas-prs-tool (Ver Figura M.1). Y, de la misma manera, para este caso, se realizaron 3 pruebas para cada servicio del grupo con excepción del servicio Estimar Diversidad Conformacional, el cual se realizaron 4 pruebas. Asimismo, estas pruebas permitieron demostrar que los servicios cumplen con la metodología seleccionada pra estimar la diversidad conformacional de las proteínas repetidas, ya que estas pruebas fueron basadas en la proteína repetida 1A0R_B y los resultados de cada test son iguales a los que se obtienen empleando los servicios que acceden a la base de datos (Ver Figura M.9).

Figure M.10

Resultados de los test para cada servicio del grupo de servicios codnas-prs-tool



Nota: El gráfico muestra los resultados satisfactorios de las pruebas realizadas para cada servicio del grupo codnas-prs-tool. Elaboración propia.

Anexo N

Informe del prototipo de la interfaz de usuario

N.1. Introducción

El presente informe describe la creación del prototipo de la interfaz de usuario, la cual será el medio para que los científicos puedan evaluar la diversidad conformacional de las proteínas repetidas. Posteriormente, se describen las ventanas que se elaboraron, las cuales son: Home, Detail, Estimate y Tutorial. Y finalmente, se detalla el tipo de letra y los colores que se han utilizado para la elaboración del prototipo.

N.2. Elaboración del prototipo

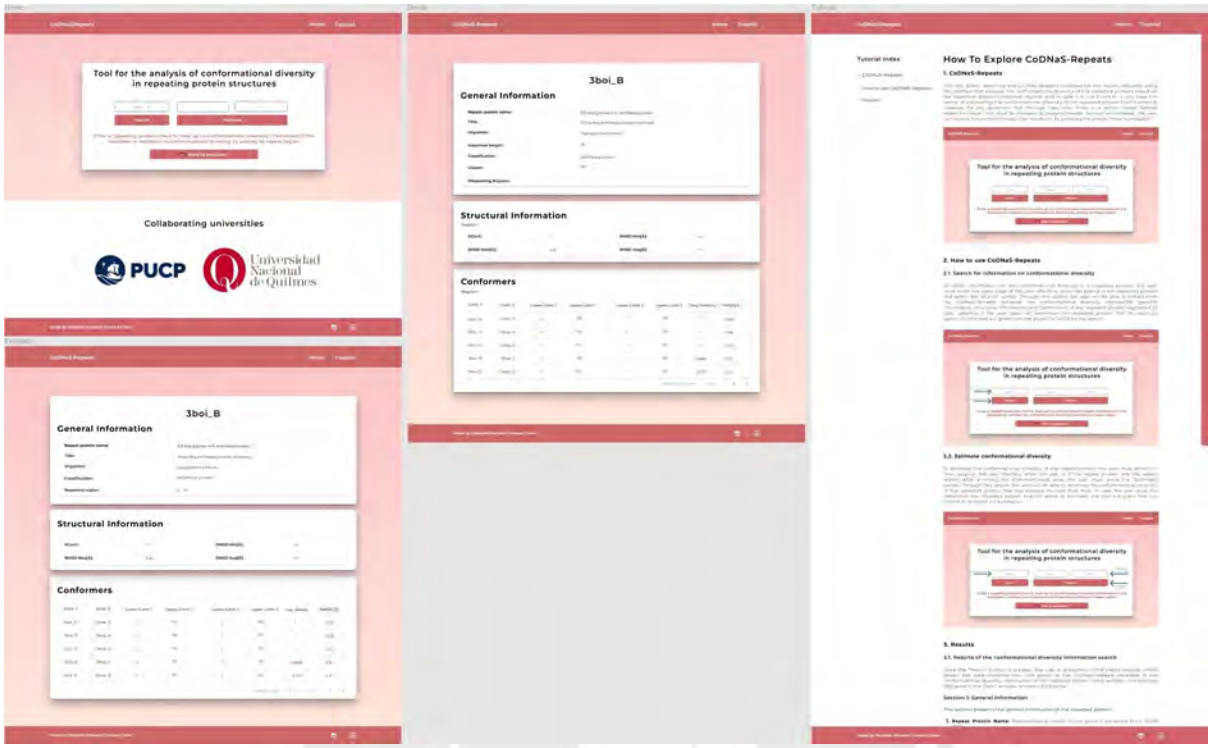
Para crear el prototipo de la interfaz de usuario se utilizó el software Figma y se elaboraron cuatro ventanas, las cuales son: Home, Detail, Estimate y Tutorial. Estas ventanas se pueden apreciar en la Figura N.1. Además, cada ventana tiene un header y un footer; y desde el header se puede ir a la ventana Home y Tutorial.

Por otro lado, para la elaboración de este prototipo se utilizó "Montserrat" como tipo de letra (font) y los colores blanco, negro y variación del rojo. Esta información se puede apreciar en la Figura N.2.

Por otra parte, la interacción entre las diferentes ventanas del prototipo se puede ver a través de la siguiente dirección: *https://www.figma.com/prototype/CoDNaS-Repeats*.

Figure N.1

Ventanas del prototipo de la interfaz de usuario



Nota: El gráfico muestra la ventanas del prototipo de la interfaz de usuario hecho en Figma. Elaboración propia.

Figure N.2

Tipo de Letra y Colores

Nota: El gráfico muestra el tipo de letra (font) y colores que se utilizaron para la creación del prototipo de la interfaz de usuario. Elaboración propia.

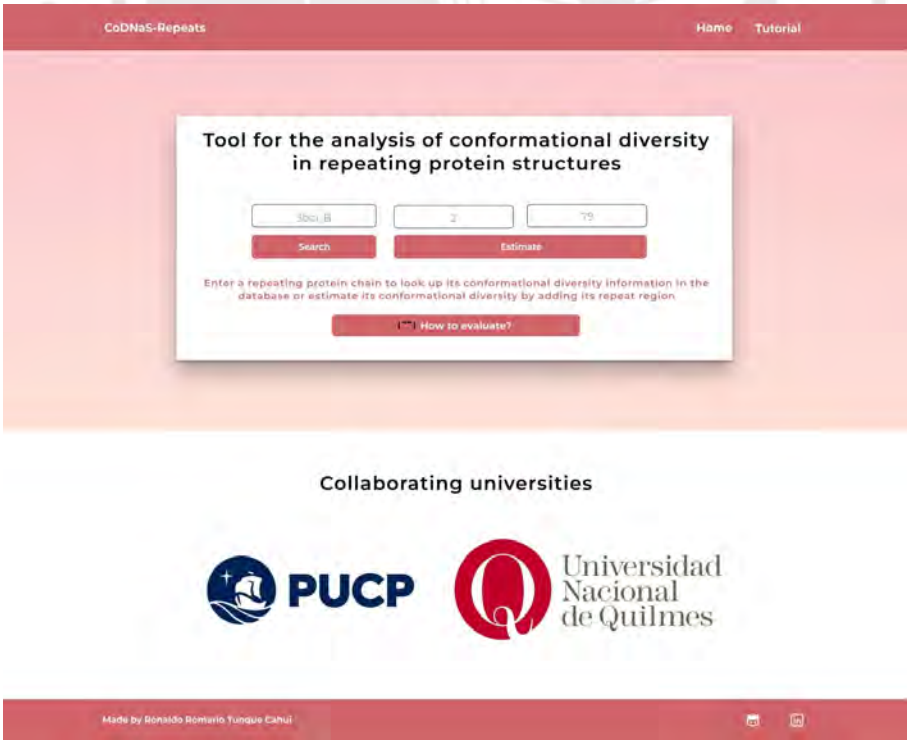
N.3. Descripción de las ventanas

En la presente sección, se procede a describir las ventanas mencionadas en la sección anterior (Inicio, Análisis de diversidad conformacional y Tutorial), las cuales forman parte del prototipo de la interfaz de usuario.

Primero, la ventana Home (Ver Figura N.3) es la primera pantalla que un usuario observará cuando ingrese al sitio web. En esta ventana, el usuario puede escribir cualquier cadena de proteína repetida en el campo de texto (text field) y presionar la opción "Search" si desea extraer la información de la base de datos o presionar la opción "Estimate" para poder evaluar la diversidad conformacional de la misma, siempre y cuando halla colocado el rango de la región de repetición. Asimismo, la presente ventana tiene la opción "How to evaluate?", el cual redirige al usuario a la ventana Tutorial donde se describen los pasos a seguir para evaluar la diversidad conformacional de una cadena de proteína repetida o buscar la información de diversidad conformacional de la misma en la base de datos.

Figure N.3

Ventana Home



Nota: El gráfico muestra la ventana Home del prototipo de la interfaz de usuario hecho en Figma. Elaboración propia.

Segundo, la ventana Detail (Ver Figura N.4) es la pantalla en la que se encontrará la información de diversidad conformacional de la cadena de la proteína repetida que se quiso buscar. Esta información está dividida en tres secciones: General Information, Structural Information y Conformers. En primer lugar, en la sección General Information se tiene el nombre, el título, el organismo, la longitud de secuencia, la clasificación, el cluster a la que pertenece y la cantidad de regiones de repetición de la proteína repetida. En segundo lugar, en la sección Structural Information se puede encontrar el número de conformaciones que tiene la proteína repetida y el grado de diversidad conformacional de la misma representado por el RMSD. Por último, en la sección Conformers se encuentran las diversas estructuras por cada región de repetición de la proteína repetida describiendo su rango de región de repetición, la secuencia de similitud y el rmsd.

Figure N.4

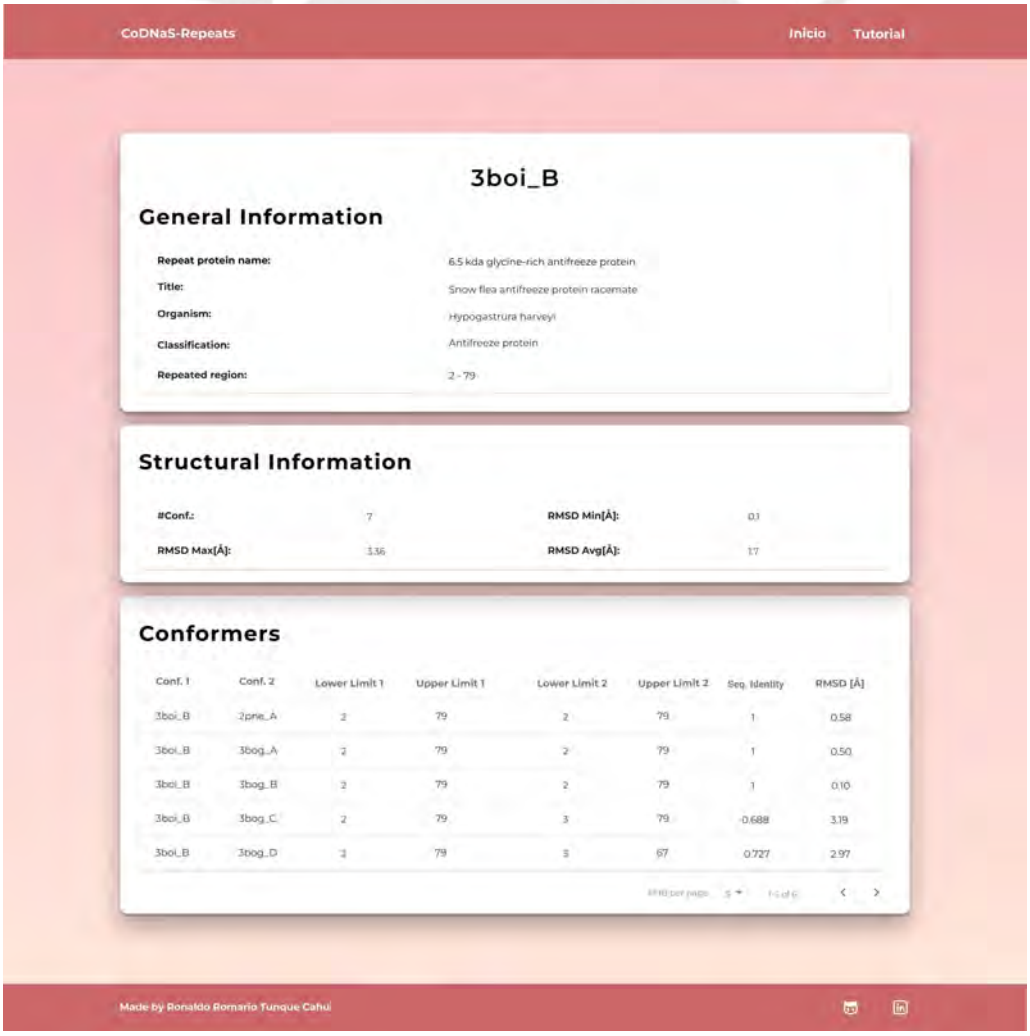
Ventana Detail

Nota: El gráfico muestra la ventana Detail del prototipo de la interfaz de usuario hecho en Figma. Elaboración propia.

Tercero, la ventana Estimate (Ver Figura N.5) es la pantalla en la que se encontrará la información de diversidad conformacional de la cadena de la proteína repetida que se quiso estimar. Esta información está dividida en tres secciones: General Information, Structural Information y Conformers. En primer lugar, en la sección General Information se tiene el nombre, el título, el organismo, la región de repetición y la clasificación de la proteína repetida. En segundo lugar, en la sección Structural Information se puede encontrar el número de conformaciones que tiene la proteína repetida y el grado de diversidad conformacional de la misma representado por el RMSD. Por último, en la sección Conformers se encuentran las diversas estructuras basados en la región de repetición de la proteína repetida a estimar describiendo su rango de región de repetición, la secuencia de similitud y el rmsd.

Figure N.5

Ventana Estimate

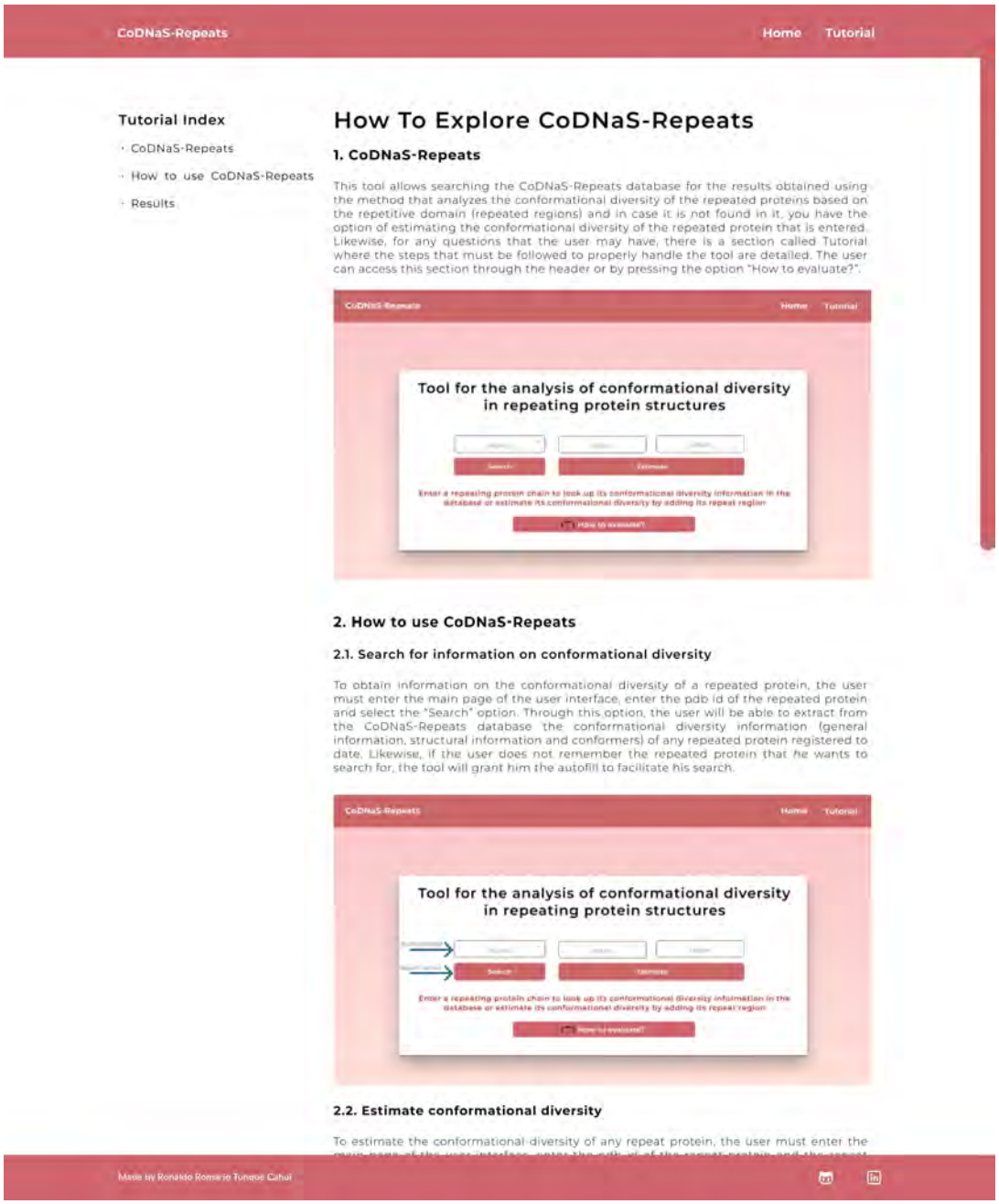


Nota: El gráfico muestra la ventana Estimate del prototipo de la interfaz de usuario hecho en Figma. Elaboración propia.

Finalmente, la ventana Tutorial (Ver Figura N.6) es la pantalla donde se detalla los pasos a seguir para evaluar la diversidad conformacional o buscar la información de diversidad conformacional de alguna proteína repetida en la interfaz de usuario.

Figure N.6

Ventana Tutorial



Nota: El gráfico muestra la ventana Tutorial del prototipo de la interfaz de usuario hecho en Figma. Elaboración propia.

Anexo Ñ

Manual de uso

Ñ.1. Introducción

El presente manual de uso describe la herramienta que va a permitir el análisis de diversidad conformacional en estructuras de proteínas repetidas. Posteriormente, detalla la manera en cómo el usuario puede buscar la información de diversidad conformacional de cualquier proteína repetida que esté registrada hasta la fecha. De la misma manera, se detalla la manera en cómo el usuario puede estimar la diversidad conformacional de cualquier proteína repetida ingresando el pdb_id de la misma y la región de repetición que quiera analizar. Y finalmente, se describen los resultados que se obtienen luego de buscar o estimar la diversidad conformacional de la proteína repetida que se haya ingresado como dato de entrada.

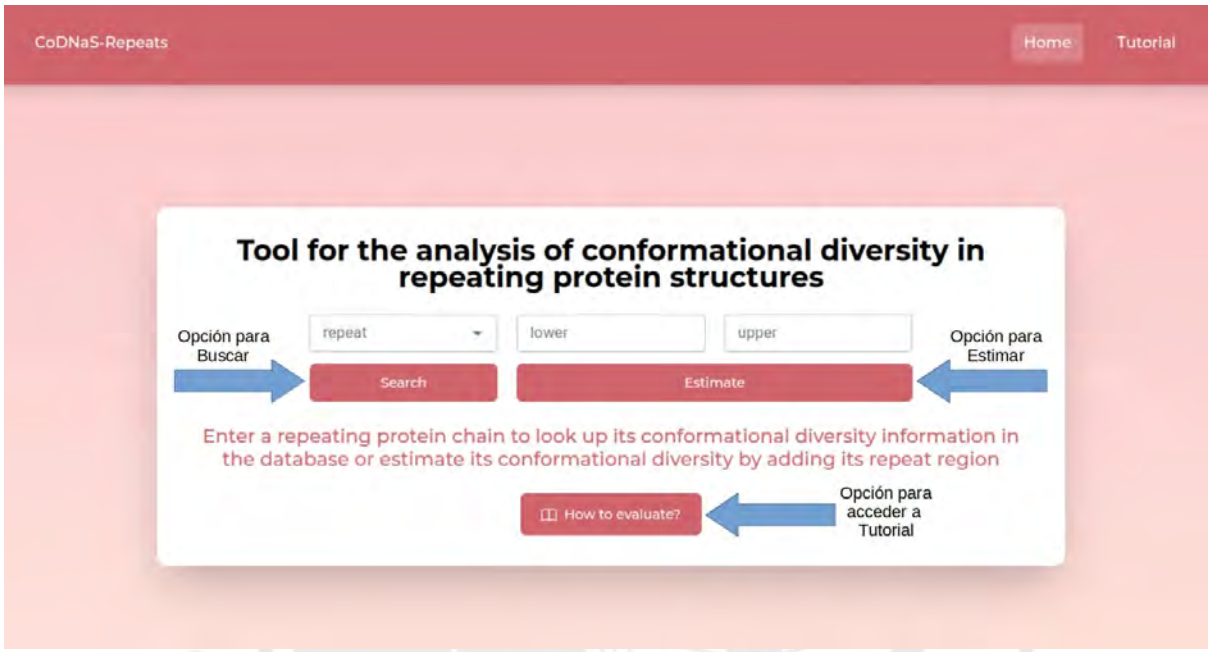
Ñ.2. Herramienta para el análisis de diversidad conformacional en estructuras de proteínas repetidas

La presente herramienta (Ver Figura Ñ.1) permite buscar en la base de datos CoDNaS-PRs los resultados obtenidos empleando el método que analiza la diversidad conformacional de las proteínas repetidas en base al dominio repetitivo (regiones repetidas) y en caso no se encuentre en la misma, se tiene la opción de estimar la diversidad conformacional de la proteína repetida que se ingrese. Asimismo, para cualquier duda que tenga el usuario, existe una sección llamada Tutorial donde se detallan los pasos que se deben seguir para manejar de manera adecuada la herramienta. El usuario puede acceder a esta sección a través del encabezado o presionando la

opción “How to evaluate?”.

Figure Ñ.1

Herramienta para el análisis de diversidad conformacional en estructuras de proteínas repetidas



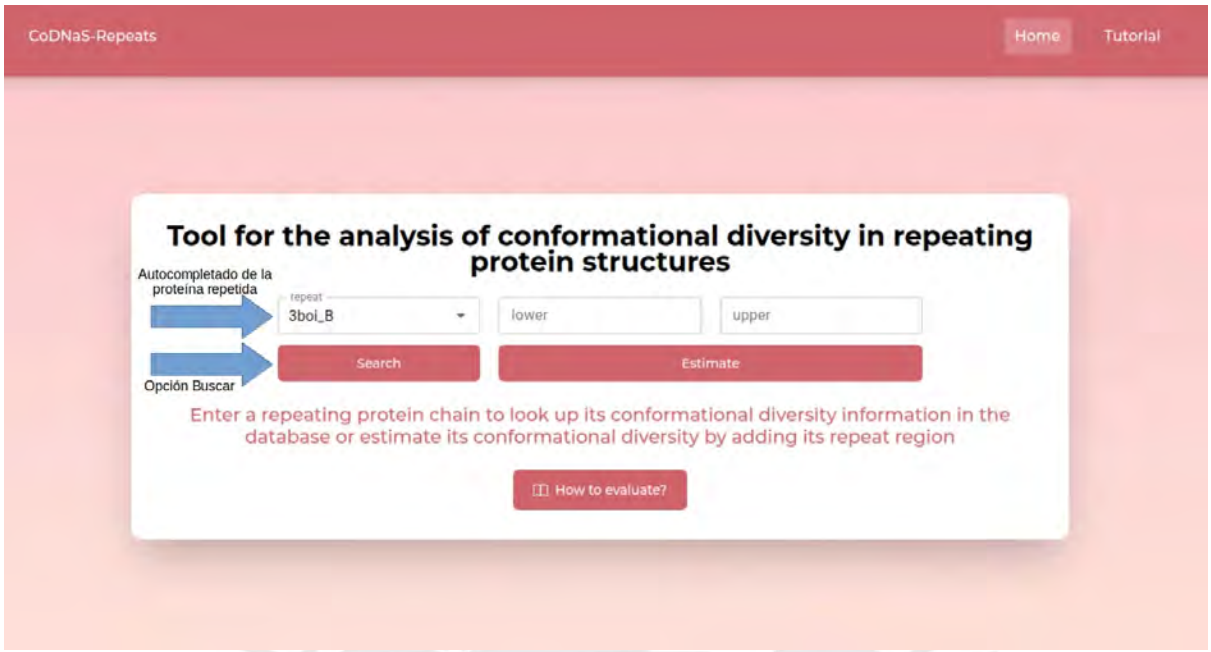
Nota: El gráfico muestra la interfaz de usuario de la herramienta para el análisis de diversidad conformacional en estructuras de proteínas repetidas con sus diferentes opciones. Elaboración propia.

Ñ.3. Buscar información de diversidad conformacional

Para obtener información de la diversidad conformacional de alguna proteína repetida (Ver Figura Ñ.2), el usuario debe ingresar a la página principal de la interfaz de usuario, ingresar el pdb_id de la proteína repetida y seleccionar la opción “Search”. A través de esta opción, el usuario va a poder extraer de la base de datos CoDNaS-PRs la información de diversidad conformacional (información general, información estructural y conformaciones) de cualquier proteína repetida registrada hasta la fecha. Asimismo, en caso el usuario no recuerde la proteína repetida que quiera buscar, la herramienta le otorgará el autcompleado para facilitar su búsqueda.

Figure Ñ.2

Búsqueda de información de diversidad conformacional



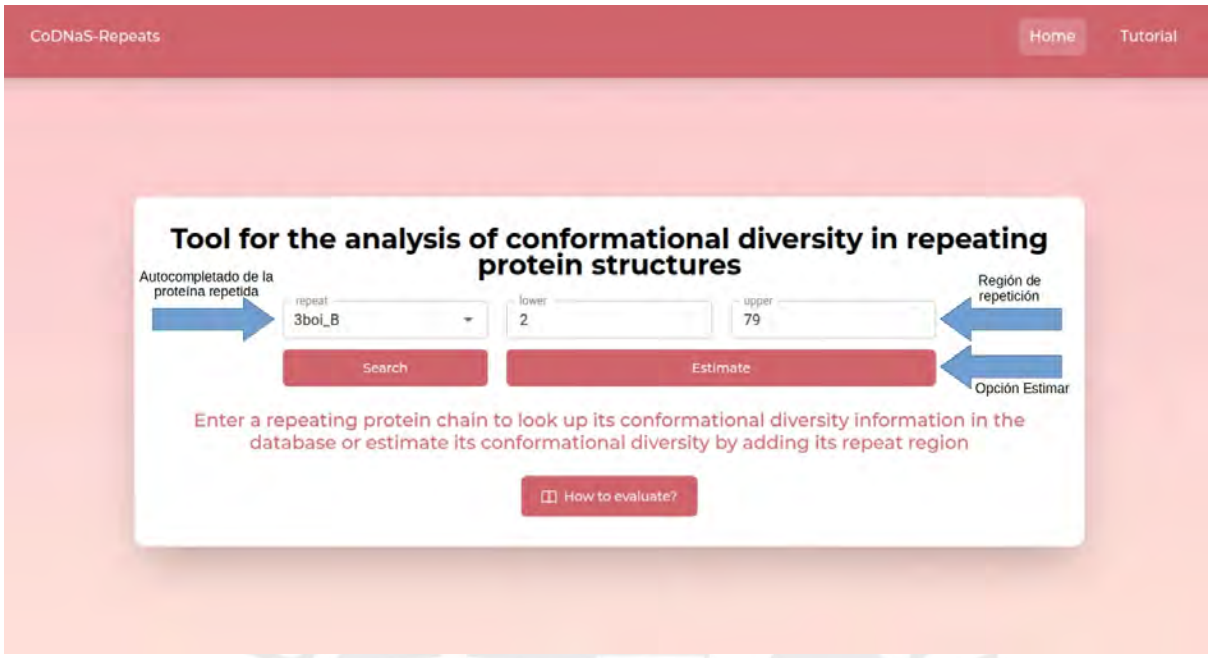
Nota: El gráfico muestra el dato que se debe ingresar y la opción que se debe presionar para buscar la información de diversidad conformacional de alguna proteína repetida. Elaboración propia.

Ñ.4. Estimar la diversidad conformacional

Para estimar la diversidad conformacional de alguna proteína repetida (Ver Figura Ñ.3), el usuario debe ingresar a la página principal de la interfaz de usuario, ingresar el pdb_id de la proteína repetida y la región de repetición. Luego de ingresar los datos mencionados, el usuario debe presionar la opción “Estimate”. A través de esta opción, el usuario va a poder estimar la diversidad conformacional de la proteína repetida que haya ingresado en el campo de texto (textfield). Asimismo, en caso el usuario no recuerde la proteína repetida que quiera estimar, la herramienta le otorgará el autcompleado para facilitar su estimación.

Figure Ñ.3

Estimación de la diversidad conformacional



Nota: El gráfico muestra los datos que se deben ingresar y la opción que se debe presionar para estimar la diversidad conformacional de alguna proteína repetida. Elaboración propia.

Ñ.5. Resultados de la búsqueda de información de diversidad conformacional

Una vez presionado el botón "Search", se redirecciona al usuario a la ventana Detail, la cual detalla los datos recopilados de la búsqueda en la base de datos CoDNaS-PRs de la información de diversidad conformacional de la cadena de proteína repetida ingresada. A continuación se describen las secciones que se muestran en la ventana Detail.

Ñ.5.1. Sección 1: General Information

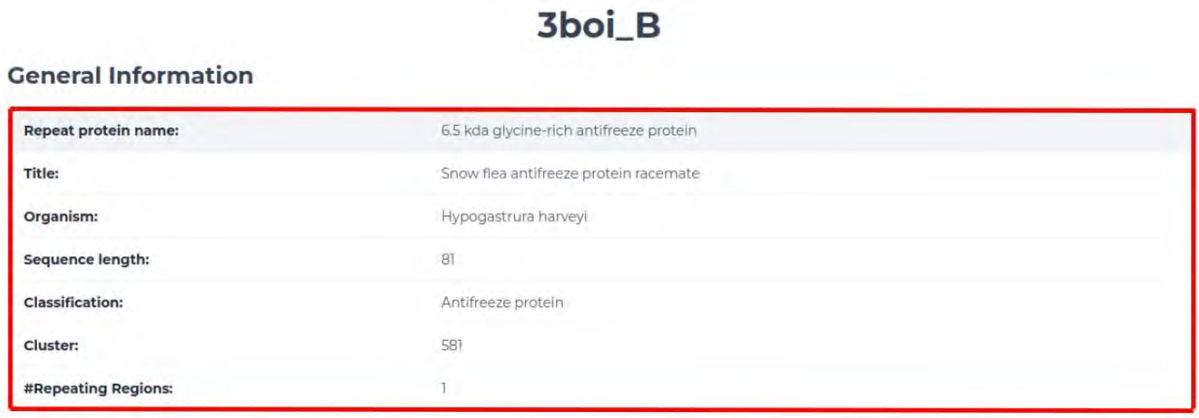
La presente sección (Ver Figura Ñ.4) presenta la información general de la proteína repetida.

1. **Repeated protein name:** Nombre representativo de la proteína extraída de RCSB PDB.
2. **Title:** Título que representa a la proteína.

3. **Organism:** Organismo a la que pertenece la proteína repetida.
4. **Sequence length:** Cantidad de aminoácidos de la cadena de proteína repetida.
5. **Classification:** Tipo de proteína clasificado por RCSB PDB.
6. **Cluster:** Clúster a la que pertenece la proteína repetida.
7. **#Repeating Regions:** Cantidad de regiones de repetición que presenta la proteína repetida.

Figure Ñ.4

Sección General Information



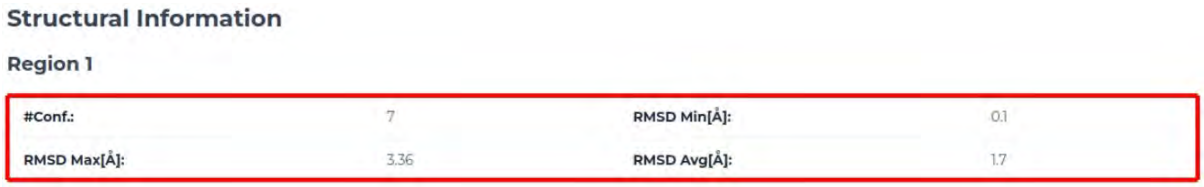
Nota: El gráfico muestra la sección General Information luego de buscar la diversidad conformacional de la proteína repetida ingresada. Elaboración propia.

Ñ.5.2. Sección 2: Structural Information

La sección Structural Information (Ver Figura Ñ.5) proporciona datos estructurales comparativos entre todas las conformaciones de la proteína repetida incluyendo la misma. Entre estos datos se tiene al número de conformaciones, el cual muestra la evidencia disponible sobre la diversidad conformacional de la proteína repetida. Además, se muestra el RMSD mínimo, máximo y promedio determinado por el software Mammoth. Estos valores proporcionan las mediciones centrales de la diversidad conformacional.

Figure Ñ.5

Sección Structural Information



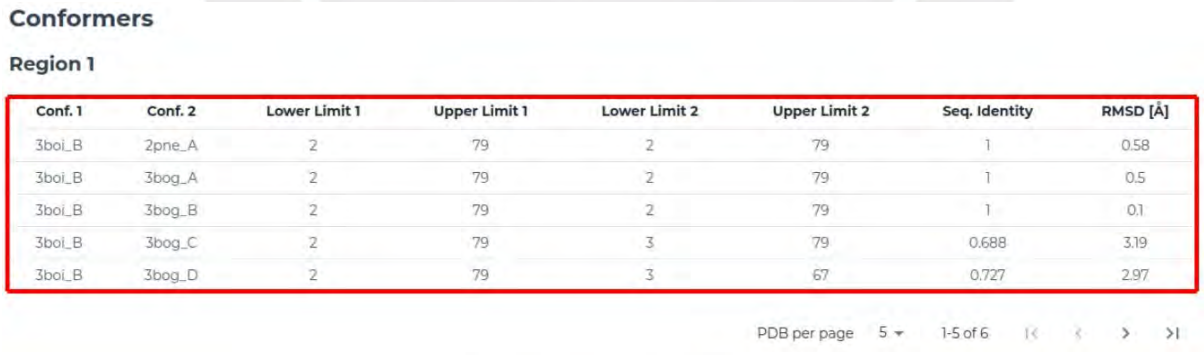
Nota: El gráfico muestra la sección Structural Information luego de buscar la diversidad conformacional de la proteína repetida ingresada. Elaboración propia.

Ñ.5.3. Sección 3: Conformers

La presente sección (Ver Figura Ñ.6) muestra las diversas conformaciones que la proteína repetida posee basados en las regiones de repetición de la misma. Asimismo, se presenta para cada conformación la región repetida a través del límite inferior y límite superior que representan el rango de la misma, la secuencia de similitud expresado en un valor numérico y la diferencia estructural entre la proteína repetida y la conformación respectiva a través de la medida estadística RMSD.

Figure Ñ.6

Sección Conformers



Nota: El gráfico muestra la sección Conformers luego de buscar la diversidad conformacional de la proteína repetida ingresada. Elaboración propia.

Ñ.6. Resultados de la estimación de diversidad conformacional

Una vez presionado el botón “Estimate”, se redirecciona al usuario a la ventana Estimación, la cual detalla los datos recopilados de la estimación de diversidad conformacional de la cadena de proteína repetida ingresada. A continuación se describen las secciones que se muestran en la ventana Estimación.

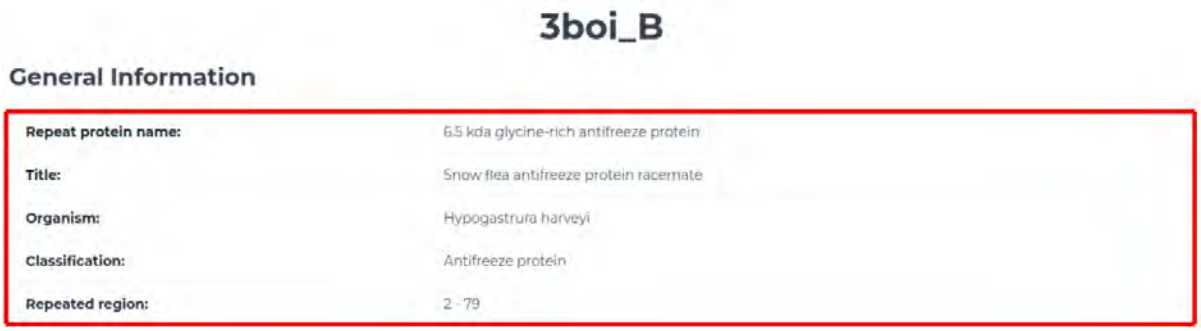
Ñ.6.1. Sección 1: General Information

La presente sección (Ver Figura Ñ.7) presenta la información general de la proteína repetida.

1. **Repeat protein name:** Nombre representativo de la proteína extraída de RCSB PDB.
2. **Title:** Título que representa a la proteína.
3. **Organism:** Organismo a la que pertenece la proteína repetida.
4. **Classification:** Tipo de proteína clasificado por RCSB PDB.
5. **Repeated region:** Región repetida de la proteína repetida a estimar.

Figure Ñ.7

Sección General Information



Nota: El gráfico muestra la sección General Information luego de estimar la diversidad conformacional de la proteína repetida ingresada. Elaboración propia.

Ñ.6.2. Sección 2: Structural Information

La sección Structural Information (Ver Figura Ñ.8) proporciona datos estructurales comparativos entre todas las conformaciones de la proteína repetida incluyendo la misma. Entre estos datos se tiene al número de conformaciones, el cual muestra la evidencia disponible sobre la diversidad conformacional de la proteína repetida. Además, se muestra el RMSD mínimo, máximo y promedio determinado por el software Mammoth. Estos valores proporcionan las mediciones centrales de la diversidad conformacional.

Figure Ñ.8

Sección Structural Information

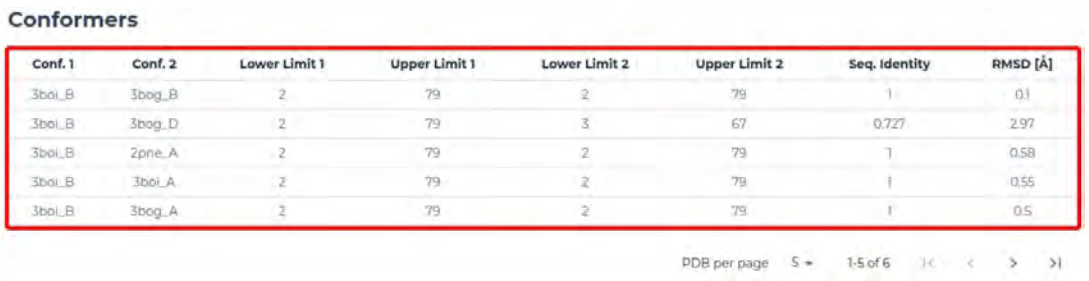
Nota: El gráfico muestra la sección Structural Information luego de estimar la diversidad conformacional de la proteína repetida ingresada. Elaboración propia.

Ñ.6.3. Sección 3: Conformers

La presente sección (Ver Figura Ñ.9) muestra las diversas conformaciones que la proteína repetida posee basado en la región de repetición de la misma ingresada como dato de entrada. Asimismo, se presenta para cada conformación la región repetida a través del límite inferior y límite superior que representan el rango de la misma, la secuencia de similitud expresado en un valor numérico y la diferencia estructural entre la proteína repetida y la conformación respectiva a través de la medida estadística RMSD.

Figure Ñ.9

Sección Conformers



Nota: El gráfico muestra la sección Conformers luego de estimar la diversidad conformacional de la proteína repetida ingresada. Elaboración propia.