Desarrollo de un método de predicción de actividad proteica a partir de su secuencia aminoacídica.

Alumno: Francisco Luis Andújar Vera
Plan de estudios: Máster Universitario de Bioinformática y Bioestadística.

Consultor: Dr. Melchor Sánchez Martínez

29.05.2016

FICHA DEL TRABAJO FINAL

Título del trabajo:	*Desarrollo de un método de predicción de actividad proteica a partir de su secuencia aminoacídica*
Nombre del autor:	*Francisco Luis Andújar Vera*
Nombre del consultor/a:	*Melchor Sánchez Martínez*
Nombre del PRA:	*Maria Jesús Marco Galindo*
Fecha de entrega (mm/aaaa):	05/2016
Titulación::	*Máster de Bioinformática y Bioestadística*
Área del Trabajo Final:	*Bioinformática farmacéutica*
Idioma del trabajo:	*Castellano*
Palabras clave	*Secuencia, Proteína, Ontología*

Resumen del Trabajo (máximo 250 palabras): *Con la finalidad, contexto de aplicación, metodología, resultados y conclusiones del trabajo.*

La revolución en cuanto a la obtención de datos biológicos procedentes de la secuenciación de alto rendimiento genera un distanciamiento importante entre los datos secuenciados y los datos analizados. Para reducir esta distancia, y poder llegar a un mayor entendimiento de la información que disponemos, se hace necesaria la aparición de herramientas que permitan digerir tanta información generada y ponerla en situación disponible para la comunidad científica para su comprensión y utilización en las diferentes áreas. Es por ello, por lo que hoy en día se trabaja concienzudamente en la realización de métodos informáticos, que permitan en cierta medida, completar el trabajo realizado por los métodos experimentales costosos y laboriosos, aunque necesarios, puesto que constituyen, en la mayoría de las ocasiones, la base sobre la que trabajar con los métodos informáticos. Con este trabajo se pretende proponer un método sencillo de predicción de función proteica basándose en la homología de secuencias de proteínas con función conocida y anotadas mediante el sistema de ontología de genes. Para ello se elaboró un *script*, mediante el lenguaje de programación Python, que pudiera comparar una secuencia de aminoácidos, con una base de datos local con información proveniente de la base de datos de proteínas de UniProt y que, a su vez, garantizara unos resultados óptimos en la predicción de la función proteica gracias a un sistema de herramientas de control de calidad.

Abstract (in English, 250 words or less):

The revolution in obtaining biological data from the high-throughput sequencing generates a significant gap between sequenced data and analyzed data. In order to reduce this problem, and to find a better understanding of the information we have, it is necessary the emergence of tools to digest the enormous quantity of information generated and make it available to the scientific community for their understanding and use in different fields. That is why nowadays, the bioinformaticians are hard working to develop computational methods that allow in this context, complete the work done by the expensive and laborious experimental methods, which are necessary, since they are in most cases, the basis on which to work using bioinformatic tools. This work aims to propose a simple method for predicting protein function based on sequence homology of proteins with known and annotated functions by gene ontology system. For this purpose, a *script* was developed, using the Python programming language that could compare an amino acid sequence with a local database containing information from UniProt protein database. This new method will guarantee results in predicting optimal protein function since it has been tested using quality control tools.

Índice

Lista de figuras

1. Introducción

1.1 Contexto y justificación del Trabajo

Durante los últimos años estamos asistiendo a una auténtica revolución en lo que obtención de datos biológicos se refiere. La cantidad de datos procedentes de experimentos, implican la necesidad de analizarlos con el objetivo de aumentar el conocimiento biológico. El principal motivo de esta acumulación de datos sin analizar, es el sorprendente avance que ha ocurrido en las tecnologías de secuenciación, que desde que se describió el método de secuenciación o método Sanger [1] y sobre todo en los últimos años con la secuenciación de alto rendimiento unido a un abaratamiento de los costes, han producido tal cantidad de datos, que incluso hoy en día, se plantean nuevos retos que antes no se tenían en cuenta, como es la capacidad de almacenamiento de estos datos. Esta capacidad de almacenamiento ha crecido de forma lineal, mientras que la generación de datos de genoma y proteoma, ha crecido de manera exponencial [2]. La demanda a diciembre de 2015 fue de 75 petabytes (Fig.1).

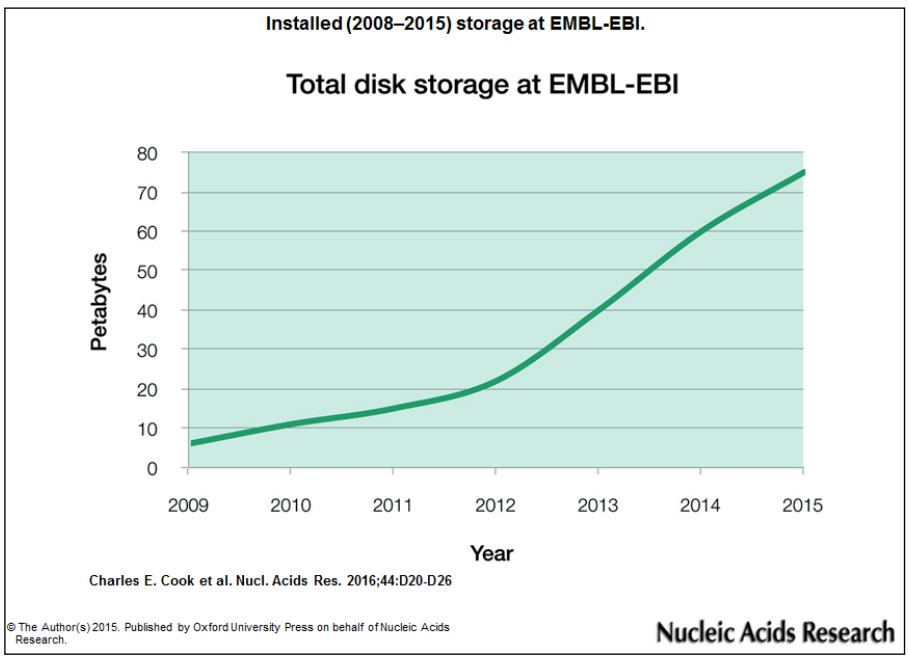


Fig.1 Demanda de almacenamiento a fecha de diciembre de 2015. Figura tomada de Cook, E. *et al.* (2016) [2].

Por todo esto, se hace necesario el desarrollo de metodologías eficientes para almacenar, analizar, interpretar y modelar las enormes cantidades de datos generadas, haciéndose necesaria su introducción en la práctica de la investigación [3], lo que ha llevado a desarrollar el campo de la bioinformática para este fin. El perfil del investigador encargado de esta labor, es el de bioinformático, el cual debe tener la cualificación necesaria tanto en el campo de lo biología como en el de la

computación, y cuyo objetivo principal es trabajar con estas secuencias de las biomoléculas, según el Centro Nacional para la Información Biotecnológica (NCBI) sobre la bioinformática: "*el fin último de este campo es facilitar el descubrimiento de nuevas ideas biológicas así como crear perspectivas globales a partir de las cuales se pueden discernir principios unificadores en Biología*".

Una de las áreas de investigación más importantes en la biología y con la que encontramos este tipo de situación de aumento de información, es el área de la proteómica. Las proteínas se encuentran en conexión directa con la secuenciación de alto rendimiento debido a su relación con los ácidos nucleicos secuenciados.

Actualmente podemos encontrar numerosas bases de datos de secuencias de proteínas que van desde simples repositorios hasta bases de datos de alto nivel, revisadas y maduradas por expertos que abarcan todas las especies y en las que incluso la secuencia original, va reforzada por la adición de información que se asocia a cada registro de secuencia [4]. Debido a las nuevas técnicas moleculares que hacen que sea posible identificar rápidamente un gran número de proteínas, mapear sus interacciones, localizar su ubicación dentro de la célula y analizar su actividad biológica, las bases de datos pasan a ser el eje fundamental para almacenar la información generada experimentalmente y permitir que dicha información esté disponible para toda la comunidad científica.

La base de datos de conocimiento universal de proteínas, UniProt (*Universal Protein Knowledgebase*) es la base de datos de secuencias de proteínas que tiene más anotaciones en el mundo, habiendo archivado y anotado más de un millón de proteínas gracias a una combinación entre técnicas manuales y electrónicas [5], proporcionando una alta calidad y utilizando el vocabulario estandarizado de ontología de genes (GO), al cual se hará referencia más adelante. El Consorcio UniProt consiste en grupos del Instituto Europeo de Bioinformática (EBI), el instituto suizo de Bioinformática (SIB) y los Recursos de Información Proteica (PIR) [6]. Su misión principal es apoyar la investigación biológica mediante un mantenimiento especializado, bien clasificado, rico en sus anotaciones y preciso en el conocimiento de las secuencias de proteínas, aportando referencias cruzadas y consultas con acceso libre a la comunidad científica (www.uniprot.org). Como fuente de las secuencias de UniProt podemos encontrar las bases de datos DDBJ, ENA, GeneBank, ProteinData, RefSeq, Esembl, entre otras.

Los tres grandes pilares sobre los que se asienta UniProt son en primer lugar, la continuación del trabajo de las bases de datos de proteínas, Swiss-Prot, TrEMBL y PIR-PDS, ofreciendo una base de datos madura y revisada; en segundo lugar, la existencia del archivo UniParc, en el que las secuencias nuevas y las actualizadas se cargan diariamente en la base de datos; y en tercer lugar, el modo UniProt NREF, que ofrece una visualización no redundante de las entradas [4].

La integración de datos juega un papel cada vez más importante para tratar de reunir la información disponible procedente de diversos recursos y poder presentarla a la comunidad científica. El éxito de esta integración en UniProt, depende de que cada base de datos utilice el mismo lenguaje para caracterizar las proteínas y poder distribuir los datos en formatos analizables [5]. En este sentido, la ontología más importante y frecuentemente utilizada dentro de la comunidad bioinformática es la ontología de genes GO.

Una ontología tiene dos propósitos, primero, facilitar la comunicación entre personas y organizaciones; y segundo, que pueda ser manejada desde diferentes sistemas [7]. El objetivo del proyecto de ontología de genes GO (http: // www. Geneontology.org/), con más de 38.000 frases definidas llamadas términos GO [8], es proporcionar un conjunto de vocabularios estructurados para los dominios biológicos específicos y que se puedan utilizar para describir los productos de genes de cualquier organismo [9–11]. El proyecto de ontología de genes, en su esfuerzo de integración de información, consigue proporcionar descriptores consistentes para los productos génicos y estandarizar la clasificación y características de las secuencias. El proyecto se inició en 1998 como una colaboración entre tres bases de datos, FlyBase, SGD y MGI; desde entonces, el Consorcio ha crecido hasta incluir numerosas bases de datos de los principales repositorios del mundo de plantas, animales y genomas microbianos [10,12].
Muchas bases de datos de organismos modelo, utilizan los términos GO y contribuyen con sus anotaciones al recurso GO. Los miembros del consorcio trabajan constantemente, junto con la participación de otros expertos, para ampliar y actualizar los vocabularios GO. El recurso web de GO ofrece acceso a una amplia documentación sobre el proyecto GO, así como enlaces a las aplicaciones que utilizan los datos para hacer análisis funcionales. El éxito de GO se basa en gran parte, a su enfoque de código abierto y participación, a lo largo de su desarrollo, de las diversas comunidades científicas [5].

Hay tres categorías de GO, (i) Procesos biológicos, que se refiere a lo que contribuye el producto génico o gen. Los procesos a menudo implican una transformación física o química, en el sentido en el sustrato es diferente cuando entra a cómo sale tras el proceso; (ii) Función molecular, que se define como la actividad bioquímica de un producto génico, o la capacidad potencial que pueda tener dicho producto de gen; y (iii) Componente celular, que se refiere al lugar de la célula donde el producto del gen es activo [13]. Dentro de estas tres categorías se pueden englobar todos los atributos de los genes, de manera que un producto de genes o gen, puede englobarse en una o varias categorías, reflejando la realidad biológica en la que una proteína podría participar en varios procesos.

Los términos GO están conectados como nudos de una red, por lo tanto se pueden conocer las conexiones entre padres e hijos, formando lo que

se conoce como gráficos acíclicos dirigidos [13–15] en los que cualquier término, puede tener más de un padre, así como ninguno, uno o más hijos. Esto hace que los intentos para describir la biología, sean mucho más ricos de lo que sería un gráfico jerárquico.

Las normas principales para la adquisición del término GO, son que todos los caminos deben ser verdaderos, deben de cubrir como mínimo el nivel jerárquico de clase, estar acompañados de citas apropiadas e incorporar declaraciones que evidencien la vinculación con el término [9].

La adquisición de la nomenclatura estandarizada GO por Uniprot fue en 2001 [14]. En ella se asocia un identificador de producto génico y un término de ontología de genes [12]. Las anotaciones GO en Uniprot, pueden ser incorporadas de manera experimental o automática. Las predicciones experimentales o manuales son más laboriosas, lentas y costosas, aunque son base importante para el desarrollo de las automáticas. Las predicciones de manera automática son muy valiosas ya que proporcionan un gran número de anotaciones funcionales de alta calidad para una amplia gama de taxonomías; de hecho, para muchas especies, puede ser la única forma de anotación posible. Las anotaciones automáticas, se crean utilizando algoritmos basados en similitud de secuencias, ortologías o dominios de información preexistente [12,16]. En general, cuando no es posible determinar experimentalmente la función de una proteína, es cuando entran en juego las técnicas de predicción automáticas.

Conocer la función proteica, se ha convertido en una pieza clave para el desarrollo de la investigación. Esto es debido a que las proteínas realizan las tareas más importantes en los organismos, tales como catálisis de reacciones bioquímicas, transporte de nutrientes, reconocimiento y transmisión de señales, etc. [17]. Las anotaciones con alta precisión tienen una gran implicación biomédica y farmacéutica, pero es necesaria una predicción con gran fiabilidad. Además de la función proteica, las anotaciones han servido para determinar otros aspectos, como determinación de redes de interacción proteína-proteína, relaciones evolutivas, etc. [18].

La explosión de información ha ampliado considerablemente la brecha entre el número de secuencias de proteínas obtenidas y el número de proteínas caracterizadas experimentalmente. A esto hay que unirle que la automatización en la producción de proteínas y determinación de estructuras, trabajando con numerosas proteínas en paralelo, ha generado más de una tercera parte de proteínas con función desconocida, anotadas simplemente como “proteínas hipotéticas” [19]. Se calcula que para entre un 10 y 40% de todas las secuencias, se puede deducir su estructura por homología, y que para entre un 40 y 60% de todas las secuencias de actuales proyectos de genoma, la homología de su secuencia podría sugerir aspectos de su función proteica.

Es por todos estos motivos, por los que se hace necesaria una predicción de función proteica con alta eficiencia, y que vaya cerrando la brecha generada entre lo anotado y lo secuenciado, que además tendrá gran utilidad para diferentes ramas científicas como biomedicina o farmacología, entre otras.

En el presente trabajo se propone un método por el que a partir de una secuencia aminoacídica desconocida, se pueda por homología, predecir la posible función que tendría ésta, comparándola con secuencias extraídas de la base de datos UniProt. Para determinar la función de la secuencia en cuestión, nos apoyaremos en los términos GO, que ya han sido descritos en esta introducción.

1.2 Objetivos del Trabajo

-Creación de una base de datos propia a partir de una existente y pública.
-Desarrollo de un método computacional de predicción de función proteica.
-Entrenamiento y evaluación de los métodos de predicción.

1.3 Enfoque y método seguido

La estrategia para llevar a cabo el trabajo está muy vinculada a los objetivos, ya que estos se han descrito de manera asociada, de tal forma que para avanzar hay que previamente, realizar el anterior. De esta manera el trabajo fue desde lo general hasta lo particular, lo que fortalece el sistema de trabajo, ya que permite un análisis profundo de cada etapa, de manera que hay que solucionar los posibles problemas o debilidades que van surgiendo para poder ir avanzando.

En primer lugar, se trabajó con las bases de datos existentes, estudiándolas, analizando sus puntos fuertes y eligiendo aquella que se podía adaptar más a las necesidades del trabajo. Una vez elegida la base de datos, que en este caso fue la base de datos de UniProt, utilizando la información que esta base de datos ofrecía, se generó un archivo propio con extensión .sql, a modo de base de datos, sobre la que se trabajó. A partir de este punto, el trabajo se centró en el desarrollo de un método eficiente y que por comparación de secuencias, nos predecía la función proteica, con un método sencillo y eficaz para poder ingresar los datos. Por último se validó el método y se hicieron las pruebas de calidad necesarias para aprobar dicho método.

1.4 Planificación del Trabajo

La planificación del trabajo se ha realizado siguiendo una serie de hitos principales, como puntos críticos del plan de trabajo. Estos hitos han sido:

(i) Búsqueda bibliográfica sobre el tema que se trata en la memoria de trabajo: debido a que el tema que se trata en este trabajo, es un tema bastante actual, la búsqueda bibliográfica y en concreto, aquella que ha sido publicada en los últimos años, se hace indispensable para conocer el punto en el que se encuentra la situación actual. En este contexto, y analizando la rapidez en los avances informáticos llevado a cabo en los últimos años, es de especial relevancia contar con bibliografía lo más reciente posible, para asegurar un desarrollo de un método eficaz y válido en la actualidad. Para ello se utilizaron buscadores especializados en la materia, los cuales, aparte de estar altamente actualizados, ofrecían una gran cantidad de información sobre los temas a tratar. También se revisaron, con especial atención, propuestas de trabajos realizados anteriormente por parte de la comunidad científica y que eran similares a la de este trabajo.

(ii) Elección de la base de datos: el gran número de bases de datos disponibles y su variabilidad, hicieron necesario profundizar en ellas, para conocer su funcionamiento y el tipo de información que ofrecían; se revisaron el conjunto de las bases de datos disponibles y con acceso libre más importantes sobre el tema que se trata en este trabajo y se eligió aquella que se adaptaba más a las necesidades del tema elegido para el citado trabajo. Las bases de datos que se revisaron fueron Swiss-Prot, PDB, SCOP, PFAM, ProSite, Uniprot y MMDB, en las que además de revisar los aspecto generales ya comentados, se investigó sobre su estabilidad de cara al futuro, prestando atención a aspectos sobre su capacidad de almacenamiento, mantenimiento o costes, entre otros aspectos.

(iii) Creación de la base de datos local: a partir de una base de datos que ofrecía todos los requisitos necesarios para cubrir la demanda de este trabajo, se creó una base de datos local, filtrada respecto a la original, sobre la cual se estuvo trabajando. Esta base de datos local, fue un archivo con extensión .sql, sobre el que se trabajó utilizando un gestor de este tipo de base de datos, para mejorar su visualización, posibles modificaciones posteriores y su sencillez a la hora de trabajar con ella.

(iv) Generación de un *Script* para la consulta en la base de datos local: se desarrolló un *Script*, que permitía la consulta de la base de datos local, basándonos en la introducción de una secuencia de aminoácidos y reportando, según homología de secuencia, la posible función que podría ejercer. Este punto se tratará más

adelante aportando un diagrama de flujo que lo divide en diferentes puntos críticos internos.

(v) Comprobación del método: una vez creado el método, se realizó un banco de pruebas en que se consultaron secuencias con función conocida y que estaban incluidas dentro de la base de datos. También se probaron secuencias que se modificaron manualmente sobre una conocida y por último, se probaron secuencias donde no se encontraba un grado de homología necesario para ofrecer una función; estas últimas se consideraron como controles negativos.

(vi) Redacción de la memoria del trabajo fin de máster: por último, una vez llegado a este punto, se redactó la memoria haciendo un análisis exhaustivo de los diferentes aspectos que se trataron en el trabajo.

Para proporcionar una mayor claridad en el punto de la generación del *Script*, se realizó un diagrama de flujo (Fig.2), para seguir los pasos intermedios dentro de este hito, pasos, que al igual que en la planificación general, corresponden a puntos asociados, de tal manera, que para continuar con un punto hay que previamente, desarrollar completamente el punto anterior:

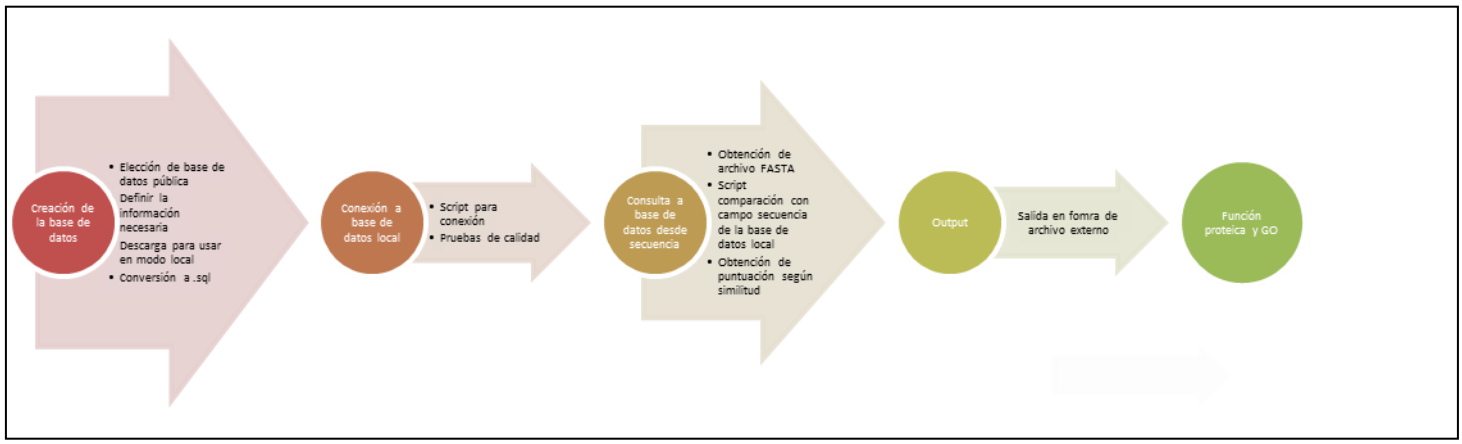
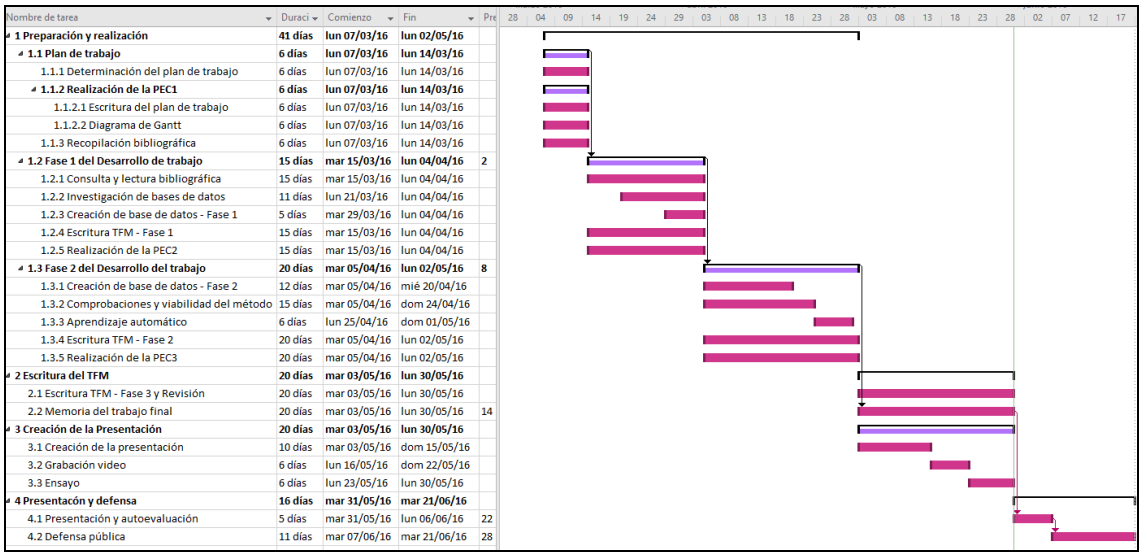


Fig.2 Desarrollo del proceso de elaboración del *script*

A modo de visión general del plan de trabajo, se construyó un diagrama de Gantt (Fig.3), para programar la planificación de manera temporal. En este diagrama se marcan los hitos parciales asociados a cada una de las PEC entregadas.



Nombre de tarea	Duraci	Comienzo	Fin	Pre
1 Preparación y realización	**41 días**	**lun 07/03/16**	**lun 02/05/16**	
1.1 Plan de trabajo	**6 días**	**lun 07/03/16**	**lun 14/03/16**	
1.1.1 Determinación del plan de trabajo	6 días	lun 07/03/16	lun 14/03/16	
1.1.2 Realización de la PEC1	**6 días**	**lun 07/03/16**	**lun 14/03/16**	
1.1.2.1 Escritura del plan de trabajo	6 días	lun 07/03/16	lun 14/03/16	
1.1.2.2 Diagrama de Gantt	6 días	lun 07/03/16	lun 14/03/16	
1.1.3 Recopilación bibliográfica	6 días	lun 07/03/16	lun 14/03/16	
1.2 Fase 1 del Desarrollo de trabajo	**15 días**	**mar 15/03/16**	**lun 04/04/16**	2
1.2.1 Consulta y lectura bibliográfica	15 días	mar 15/03/16	lun 04/04/16	
1.2.2 Investigación de bases de datos	11 días	lun 21/03/16	lun 04/04/16	
1.2.3 Creación de base de datos - Fase 1	5 días	mar 29/03/16	lun 04/04/16	
1.2.4 Escritura TFM - Fase 1	15 días	mar 15/03/16	lun 04/04/16	
1.2.5 Realización de la PEC2	15 días	mar 15/03/16	lun 04/04/16	
1.3 Fase 2 del Desarrollo del trabajo	**20 días**	**mar 05/04/16**	**lun 02/05/16**	8
1.3.1 Creación de base de datos - Fase 2	12 días	mar 05/04/16	mié 20/04/16	
1.3.2 Comprobaciones y viabilidad del método	15 días	mar 05/04/16	dom 24/04/16	
1.3.3 Aprendizaje automático	6 días	lun 25/04/16	dom 01/05/16	
1.3.4 Escritura TFM - Fase 2	20 días	mar 05/04/16	lun 02/05/16	
1.3.5 Realización de la PEC3	20 días	mar 05/04/16	lun 02/05/16	
2 Escritura del TFM	**20 días**	**mar 03/05/16**	**lun 30/05/16**	
2.1 Escritura TFM - Fase 3 y Revisión	20 días	mar 03/05/16	lun 30/05/16	
2.2 Memoria del trabajo final	20 días	mar 03/05/16	lun 30/05/16	14
3 Creación de la Presentación	**20 días**	**mar 03/05/16**	**lun 30/05/16**	
3.1 Creación de la presentación	10 días	mar 03/05/16	dom 15/05/16	
3.2 Grabación video	6 días	lun 16/05/16	dom 22/05/16	
3.3 Ensayo	6 días	lun 23/05/16	lun 30/05/16	
4 Presentacón y defensa	**16 días**	**mar 31/05/16**	**mar 21/06/16**	
4.1 Presentación y autoevaluación	5 días	mar 31/05/16	lun 06/06/16	22
4.2 Defensa pública	11 días	mar 07/06/16	mar 21/06/16	28

Fig.3 Diagrama de Gantt sobre la planificación para el desarrollo del trabajo

1.5 Breve sumario de productos obtenidos

En realidad según se desarrolló este trabajo fin de máster, no se obtienen productos al final del proceso, el principal núcleo sobre el que se trabajó, fue el desarrollo del *Script* que permitía la obtención de una función, desde una entrada que consistía en la secuencia de aminoácidos de una proteína dada. Esta posible función que se obtiene por homologías de secuencias de proteínas, enfrentando una proteína consulta o desconocida, frente a la base de datos local generada, ofreciendo un archivo de texto con la predicción de la función de la proteína con función desconocida.

1.6 Breve descripción de los otros capítulos de la memoria

Aparte de los capítulos indicados hasta este momento, se han descrito otros capítulos a los cuales hago mención a continuación:

- Capítulo 2: Material y métodos.

En este capítulo se trataron los materiales y métodos utilizados para la realización de este trabajo fin de máster, es decir, cómo se ha realizado el proyecto, qué herramientas y análisis se utilizaron para el desarrollo de cada uno de los puntos establecidos en la realización del trabajo. Este capítulo guarda especial interés ya que se muestra en general la manera de trabajar y cómo llegar a las conclusiones finales descritas en el trabajo.

- Capítulo 3: Resultados

Aunque como se ha comentado antes, el resultado de este trabajo no es un producto final, puesto que depende de la consulta que se haga, es necesario mencionar en este apartado, a modo de ejemplo, algún resultado obtenido tras una de las consultas realizadas en las pruebas del *Script*, de esta manera el lector puede hacerse una idea de cómo y qué tipo de salida nos ofrece el propio *Script*.

2. Material y métodos

Siguiendo la dinámica que se ha estado utilizando hasta ahora, y basándonos en el plan de trabajo establecido para la realización de este trabajo fin de máster, se muestran a continuación, a modo de material, las herramientas utilizadas o herramientas TIC, que han sido necesarias para completar el desarrollo del mismo, organizado según los hitos principales que lo componen:

2.1 Búsqueda bibliográfica

Aunque es un apartado un tanto obvio, se hace necesario, para no romper el hilo propuesto, mencionar y describir algunos de los buscadores especializados que se consultaron. La importancia de tener una buena fuente bibliográfica para los temas que se tratan en este trabajo es esencial debido a la velocidad en la que aparecen nuevas tecnologías, métodos y funciones. En general la información que se ha utilizado para el desarrollo y consulta, se trató que fueran lo más reciente posibles. En concreto tomaron especial relevancia PubMed (http://www.ncbi.nlm.nih.gov/pubmed) de NCBI (*National Center for Biotechnology Information*) que cuenta con más de 25 millones de citas en la literatura biomedica y Science Direct de Elsevier, que contine más de 14 millones de publicaciones, 3.800 revistas y más de 35.000 libros.

2.2 Bases de datos

Se consultaron varias de las bases de datos más completas y actuales que se conocen, entre ellas cabe destacar Swiss-Prot, PDB, SCOP, PFAM, ProSite, Uniprot y MMDB. Todas estas bases de datos fueron analizadas, pero se decidió utilizar la base de datos Uniprot debido principalmente a su importancia en el campo de la proteómica (http://www.uniprot.org/) y a que se cubría con su utilización otras de las grandes bases de datos analizadas como es Swiss-Prot.

Uniprot es la base de datos con más anotaciones del mundo, el reporte que la propia página ofrece sobre la estadística con fecha de 11 de mayo de 2016, arroja unas cifras de 551.193 entradas, de las cuales 242 son nuevas entradas, 120.967 son entradas actualizadas y el resto son entradas sin cambios. Esto da una idea no sólo del volumen de datos que maneja esta poderosa base de datos, sino también del nivel de actualización de las entradas que contiene. Otros datos importantes hacen referencia al número de especies con el que se trabaja, en total son 10.451 especies, de las que según este último reporte, 51 son nuevas y 5.136 fueron actualizadas. En cuanto al tamaño de las secuencias con las que trabaja, la más corta es de dos aminoácidos y la secuencia más larga es de 35.213 residuos, manteniendo el total del

grueso del tamaño entre los 100 y 400 aminoácidos. Uno de los aspectos que nos interesaba para la realización de este trabajo, era la función de las proteínas. En el reporte que estamos haciendo mención, podemos encontrar el dato sobre la función en cuanto a la anotación se refiere, de tal forma que se encuentran 446.550 anotaciones sobre la función en 428.071 entradas, demostrando de esta manera la importancia de este aspecto en las anotaciones de las proteínas. La función es una de las características más anotadas en esta base de datos de proteínas, junto a las que tratan sobre localización subcelular y similitudes de secuencia.

Un último apartado interesante en el reporte estadístico que nos ofrece, también hace referencia a la vinculación con otras bases de datos, tratándolos como referencias cruzadas entre las bases de datos. En este apartado podemos encontrar que UniProt tiene casi un millón de entidades vinculadas a EMBL, algo más de 700.000 entidades vinculadas a PFAM o cerca de 150.000 respecto a PDB entre otras muchas.

Otro apartado referente a la utilización de esta base de datos, es la sencillez a la hora de buscar información en ella, lo que ha sido fundamental para la realización del siguiente hito en el plan de trabajo, que es la creación de la base de datos propia. Para filtrar la inmensa información que nos ofrece Uniprot, entramos en su apartado UniProtKB (http://www.uniprot.org/uniprot/) en donde por defecto nos muestra los casi 63 millones de entradas disponibles, incluidas en todo el conjunto que maneja UniProt con sus diferentes bases de datos. Dispone de una barra de búsqueda donde escribiendo de manera sencilla o marcando las opciones que nos ofrece como más comunes, se puede filtrar hacia el tipo de información que deseamos; en nuestro caso, los datos queríamos que estuvieran revisados (reviewed:yes) y que el organismo sobre el que trabajara fuera el humano (AND organism:"Homo sapiens(Human)"). Una vez que se estableció este filtro, las entradas se habían reducido a poco más de 20.000.

Un aspecto interesante a la hora de manejar el visionado de datos, es que UniProt, tiene la posibilidad de elegir el tipo de información que se muestra en pantalla y el orden en que aparece ésta. Para ello establece una alta variedad de columnas, cada una definida para un tipo de información concreta. Esta información elegible a modo de columnas, va agrupada para una cómoda localización del tipo de información que deseamos. Podemos encontrar un grupo referente a nombres y taxonomía, donde encontraríamos columnas que cubren la información sobre el nombre de la proteína, el nombre de genes, el organismo, proteomas…Otro grupo que hace referencia a las secuencias en general, donde podemos encontrar información sobre la secuencia, fragmento, tamaño, secuencia alternativa…En definitiva, podemos encontrar grupos que hacen referencia a la información que queremos encontrar y en cualquier caso también dispone de una barra buscadora para localizar rápidamente el tipo de información que necesitamos.

Para el caso que nos ocupaba durante el desarrollo del trabajo de fin de máster, se eligieron las columnas que hacen referencia a la información disponible sobre: entrada, que es la columna principal y que evita la redundancia dentro de la base de datos; nombre de entrada, en el que se le da un nombre particular a la entrada; nombre de la proteína, es la nomenclatura por la cual se conoce a la proteína; secuencia, la secuencia de aminoácidos que componen la proteína; y gene ontology (función molecular), que es la descripción de la función o funciones que se han anotado para la proteína en cuestión, en formato de ontología de genes.

Una vez elegida toda la información que queremos incluir en nuestra consulta, UniProt permite de manera sencilla poder descargarlo a nuestro ordenador, simplemente ejecutando el botón de descarga y eligiendo entre uno de los formatos que nos permite; en nuestro caso, se eligió formato “separado por tabulación”, aunque se puede elegir entre otros como FASTA, XLM, Excel, texto, etc.

2.3 Creación de la base de datos local

Para la creación de nuestra propia base de datos, se partió de la información descargada en el paso anterior, obteniéndose un archivo sobre el cual se empezó a trabajar para adaptarlo a nuestras necesidades. El sistema de gestión de base de datos fue MySQL dada su velocidad de lectura y flexibilidad, algo que se hace totalmente necesario para este tipo de trabajos. Aparte, su poder relacional con bases de datos, hace que sea, por otra parte, un candidato para desarrollos más futuros sobre la base de datos creada.

Para la conversión del archivo descargado de Uniprot a un archivo tipo .sql, se utilizó phpMyAdmin (https://www.phpmyadmin.net/) que es una herramienta de software libre codificada en PHP facilitando su utilización, y que permite la administración de MySQL entre otros. La misma web provee al usuario de toda la documentación y tutoriales necesarios para su correcto funcionamiento. Gracias a su interfaz intuitivo y la posibilidad de importar datos desde numerosos tipos de archivos como el descargado de la base de datos Uniprot, rápidamente se puede empezar a utilizar la mayoría de las características de MySQL una vez importado el archivo.

Una vez instalado el programa en el ordenador y ejecutado, lo primero será abrir una cuenta e ingresar como usuario. Esto es de gran ayuda para guardar el desarrollo del trabajo para futuras sesiones. La página central del programa está compuesta por pestañas claras que nos permiten movernos sin problema para sacar el potencial del programa. La primera pestaña, “Base de datos” es donde se creó nuestra base de datos, que a su vez nos permitió asociarle un nombre y empezar a construir la tabla que contendrá toda la información, empezando por el

número de columnas y definiendo el tipo de datos que contendrá cada una (nombre de la columna, tipo, longitud, cotejamiento…). Una vez creada la estructura de la tabla, que siempre podrá ser modificada, se procedió al volcado o importado de los datos descargados; en este caso, una de las pestañas indica este paso con el nombre de “importar”. En esta pestaña, principalmente lo que hay que indicar es el archivo que se desea importar y su ubicación en el ordenador. Una vez hecho esto, nos volcará todos los datos es nuestra tabla, la cual podremos examinar fácilmente para asegurarnos que todo concuerde. Además tendremos acceso al número total de filas volcadas, y a la posibilidad de gestionar la tabla, bien sea manualmente con las opciones que nos permite el programa o bien, a través de una opción en la pestaña “SQL” de sintaxis tipo MySQL. En cualquier caso, es un excelente programa para la gestión de este tipo de bases de datos, permitiendo dinamismo e intuición, sin olvidar el potencial que se puede desarrollar cuando se trabaja MySQL en consola

La manera de guardar el archivo en nuestro ordenador puede ser tan sencillo como exportando la tabla que se creó a uno de los tipos de archivo que el programa permite a la hora de exportar, en nuestro caso fue .sql.

2.4 Generación del *Script*

Una vez que se modificó la base de datos al tipo de archivo que se deseaba se procedió, mediante un lenguaje de programación, a la realización de un *script* que pudiera cumplir con las expectativas y objetivos que se habían definido para este trabajo de fin de máster. El *script* se dividió en tres partes para hacer más cómodo su entendimiento, por un lado se trabajó la conexión con la base de datos en formato sql, a continuación se trabajó para realizar la consulta a esta base de datos mediante homología de secuencia y por último, se cuidó que la salida de esta consulta se tradujera en un archivo de texto para su posterior utilización. Para todo esto el lenguaje utilizado fue Phyton.

Phyton (https://www.python.org/) es un lenguaje de programación libre y sencillo, pero con un alto potencial en el campo de la biología, es un lenguaje interpretado, multiplataforma y usa tipado dinámico. Su principal objetivo es utilizar una sintaxis que favorezca la legibilidad del propio código. En general es un lenguaje de programación paradigma, soportando la orientación a objetos, programación funcional y programación operativa, permitiendo en definitiva, es uso de varios estilos de programación. Su poder también reside en la obtención de resultados excelentes en pocas líneas de código.

Uno de los principales potenciales de Phyton no sólo es que disponga de una librería estándar excepcional, sino que además, brinda la posibilidad de utilizar librerías externas de terceros, como Numpy o Biopython, entre otras muchas, que se distribuyen como paquetes instalables mediante el

gestor de paquetes de su distribución o incluso por un gestor específico de Python. Para la realización de este trabajo se utilizó principalmente la librería Biophyton que se encuentra disponible de manera gratuita en su página web (http://biopython.org/). Biopython es el programa ideal para personas que desean trabajar con datos bioinformáticos. Unas de las muchas características que se pueden extraer de trabajar con esta librería, es la de tener el código preparado para trabajar con las bases de datos más conocidas y con los servicios de bioinformática tipo BLAST o búsquedas en Swiss-Prot, pudiendo realizar además, operaciones básicas con las secuencias, como traducción, o tener soporte para los archivos más comúnmente utilizados en este campo, como fasta, Medline, Pubmed, entre otros muchos.
Cobra especial importancia para el entendimiento del trabajo realizado, el paquete *pairwise2* (http://biopython.org/DIST/docs/api/Bio.pairwise2-module.html), que contiene la librería Biopython. Este paquete realiza una alineación de secuencias por pares, utilizando para ello un algoritmo de programación dinámica, de tal manera que al hacer la alineación genera una puntuación total, en la cual puede incluir las penalizaciones por hueco, de tal manera que cuanto mayor es la puntuación que nos indica, mejor compatibilidad entre las secuencias existe.

Para obtener una predicción con altas garantías, se estableció un umbral sobre la puntuación con la que trabaja el paquete *pairwise2*, de tal manera que se generó un archivo de texto externo con la función que se predice tras la alineación de las secuencias.

La utilización de Python se decidió llevarla a cabo mediante el *IPython Notebook* (https://ipython.org/ipython-doc/3/notebook/), que se trata de un intérprete de Python, multiplataforma y de software libre. Funciona con un interfaz simple pero completo, y trabaja generando diferentes celdas sobre las que poder desarrollar el código. Las celdas están asociadas, de tal manera que se puede ir arrastrando el código ejecutado, pero a la vez, se trabaja individualmente, pudiendo ejecutar cada una de las celdas por separado, algo muy interesante a la hora de revisar o modificar, diferentes partes del código.
La elección de este intérprete fue debida a varias razones, entre ellas, la opción de autocompletado, que permite desarrollar una velocidad importante a la hora de escribir código, además de hacerlo correctamente y por otro lado, las opciones de guardar, editar, cargar y el historial de los trabajos realizados anteriormente lo que permite su acceso de una manera simple y rápida.

El *script* completo que se utilizó, se adjunta en el apartado “Anexos” de esta misma memoria, donde se indica cada uno de los subapartados que componen la generación del *script*, como son la conexión a la base de datos local, la consulta a la base de datos local y la salida que se propone.

Para finalizar este apartado y cerrar el círculo sobre la generación del *script*, éste se sometió a un banco de pruebas para comprobar, no sólo su correcto funcionamiento, sino la eficiencia de predicción de función proteica. Para ello, se eligieron aleatoriamente cincuenta secuencias de proteínas conocidas y que estaban incluidas en la base de datos, treinta proteínas que estaban incluidas en la base de datos, y a las que se le cambiaron treinta aminoácidos aleatoriamente y cinco proteínas, que se sabía que no estaban incluidas en la base de datos local y en las que se comprobó que no existía buena compatibilidad con las secuencias que estaban incluidas en la base de datos local, incluyéndolas a modo de control negativo.

3. Resultados

Aunque ya se ha mencionado brevemente la situación de los resultados para este trabajo, es importante describir a modo de resultado, cómo, gracias al *script* se puede conseguir predecir la función de una proteína con cierta probabilidad.

Para desarrollar un resultado concreto, hay que ingresar la secuencia aminoacídica de la proteína que queremos consultar, guardándose dicha secuencia en un archivo en concreto al cual se tiene acceso desde el *script*. A partir de aquí, un paquete del *script* de la librería de biopython, *pairwise2*, será el encargado de realizar la alineación de la secuencia con las secuencias de las proteínas de la base de datos local. La puntuación sobre la que se trabajó y se filtraron los resultados fue de 0.9, con objeto de lograr una gran fiabilidad. El resultado se obtendrá como un archivo de texto (Fig.4) externo donde se indicaría la función proteica según la nomenclatura de ontología de genes.

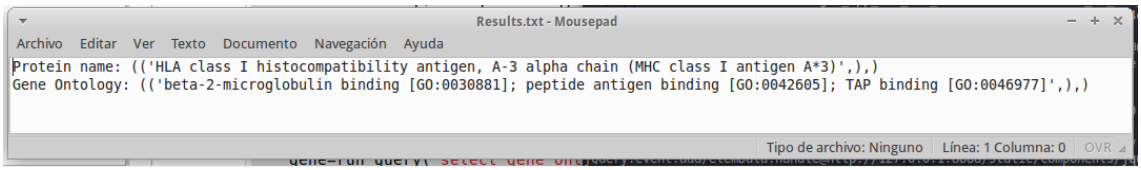


Fig. 4 Ejemplo de salida de resultados del *script* desarrollado.

Las pruebas de control del método, fueron fieles cuando se compararon secuencias conocidas de la propia base de datos: de las cincuenta secuencias de proteínas aleatorias que eran idénticas a una secuencia que se encontraba en la base de datos, cuarenta y nueve (98%), acertaron en su predicción. De las treinta secuencias que se modificaron en treinta aminoácidos, sobre otras que estaban incluidas en la base de datos local, se obtuvieron veintiséis aciertos (86%). En el caso de los controles negativos, las cinco secuencias que se probaron no ofrecieron ningún resultado de predicción de proteína, validando esto, la fiabilidad del método propuesto.

4. Conclusiones y Discusión

La predicción de la función proteica en términos de ontología de genes es un mundo que está en constante evolución. Para la realización de este trabajo de fin de máster, el desarrollo de un *script* que permita, con una alta fiabilidad, responder a esta tipo de cuestiones, ha supuesto un verdadero reto personal y profesional. Elegir y entender las herramientas adecuadas para generar este tipo de trabajo son una pieza clave para el éxito del trabajo.

Para la realización de este trabajo fin de máster se ha desarrollado un *script* que permite, de manera sencilla y eficaz, poder predecir la función de una proteína basándose en la homología de secuencias de una base de datos local con información procedente de la base de datos de proteínas UniProt.

Los resultados obtenidos tras el análisis de control de calidad propuesto, indican que en las condiciones expuestas, el método es fiable, teniendo en cuenta que en la actualidad, a partir de un 50 – 60% de éxito se puede considerar aceptable. Evidentemente, tras la finalización de las pruebas hay ciertos puntos mejorables respecto al banco de pruebas o control de calidad establecido. Respecto las secuencias de proteínas probadas y que eran idénticas a las ya incluidas en la base de datos local, vemos que el resultado, a pesar de no ser un 100%, se ajusta de manera adecuada a los resultados esperados. La falta del éxito rotundo se asocia a problemas instrumentales, tratándose en este caso de problemas informáticos. En el caso de las secuencias de proteínas que se utilizaron como controles negativos, se garantizó su no existencia gracias a la base de datos generada, al tener un umbral de homología elevado, no se muestran posibles coincidencias respecto a las secuencias conocidas. En relación a las secuencias de proteínas que se modificaron sobre secuencias base conocidas de la propia base de datos local, se observó que a pesar de que el porcentaje de éxito fue elevado, podría entenderse esto, como un resultado incompleto a la hora de probar el *script* en su control de calidad. Conforme se estuvieron analizando los resultados comprobamos que la modificación que se realizó sobre las secuencias, podría llegar a ser incompleta, es decir, teniendo en cuenta los tamaños de las secuencias de las proteínas incluidas en la base de datos que iban desde 4 aminoácidos hasta 34.350 aminoácidos, el hecho de modificar por número de aminoácidos, puede hacer que el sistema no sea correcto. A partir de esta conclusión, se podría plantear una mejora para el futuro de este aspecto, haciendo modificaciones en base al tamaño de la secuencia a probar, o lo que es lo mismo, modificando un porcentaje de la secuencia elegida. Siguiendo en este hilo, quizás hacer un gradiente de porcentaje de modificaciones hubiera podido ser interesante a la hora de analizar la fiabilidad del método, y en definitiva, conocer cuánto se puede modificar de manera aleatoria una secuencia de proteínas, para seguir estando asociada, con un alto nivel de probabilidad, a su función original.

Evidentemente el número de secuencias de proteínas a comparar en el banco de pruebas, es otro punto, que podría mejorarse, teniendo en cuenta el tamaño de la base de datos local. La limitación principal en este caso, fue la cantidad de tiempo requerido por la computadora para dar los resultados, obteniéndose además en algunos casos resultados erróneos, debido al bloqueo del programa al ejecutar todas las operaciones de comparación de secuencias.

Por otro lado, el umbral con el que se estuvo trabajando para la predicción de función de proteínas, y cuya puntuación nos ofrecía el paquete *pairwise2* de Biopython, se estableció en 0.9, lo que indicaría que para ofrecer el resultado sobre la función proteica, las secuencias deberían de tener un muy elevado grado de homología, algo que como ya hemos comentado, podría sesgar determinados resultados aceptables y que darían una puntuación entorno al 0.6. Sería interesante añadir para el futuro, una posibilidad de que el propio *script* ofreciera los resultados a modo de gradiente de puntuación desde la puntuación 1 hasta la puntuación 0.6; de esta manera podríamos ver qué grado de homología tendrían las secuencias y por tanto adaptarlo al trabajo que le sucedería.

El método de entrada que se estableció para predecir la función de las secuencias fue mediante el método fasta, uno de los métodos más aceptados en la actualidad, aunque existen otros; adaptar el *script* para que pudiera trabajar con otros métodos de entrada, sería una muy buena manera de mejorarlo, además la propia librería utilizada, Biopython, acepta varios formatos de entrada y en última instancia se podría desarrollar manualmente un *script* para su funcionamiento, bien sea porque el propio *script* acepte otro formato de entrada, o por simple conversión a tipo fasta.

Aunque hoy en día la integración de datos en las bases de datos, es un pilar fundamental en éstas, una opción que hubiera enriquecido en gran medida el trabajo, hubiera sido poder trabajar con otra base de datos alternativa que ofreciera la posibilidad de obtener una predicción basándose en dos fuentes distintas. Lo normal es que si la predicción de la función es correcta, ambas darían un resultado semejante, pero en el caso de que dieran diferentes resultados podríamos detectar posibles incongruencias en la secuencia o en las propias funciones anotadas para dichas secuencias.

Se ha trabajado sobre el vocabulario de ontología de genes GO, algo que se eligió teniendo en cuenta su aceptación universal y su fácil comprensión, pero se conocen otras formas de catalogar las funciones, que en definitiva, deberían de ser semejantes al sistema usado por la ontología de genes; para posibles adaptaciones a métodos de trabajo en las distintas instituciones, podría ser necesario utilizarlas.

En general, durante el desarrollo del trabajo, se han adquirido conocimientos importantes en los campos de bases de datos generales,

no sólo de proteínas. El funcionamiento y las posibilidades que ofrecen cada una de ellas son casi inagotables y funcionan como una herramienta indispensable para trabajar en este tipo de materias. Además se ha adquirido un nivel importante a la hora de poder desarrollar mediante el lenguaje de programación Pyhton, un *script* que cumpla con las necesidades que el investigador precisa; y por último, se ha adquirido un gran conocimiento en las diferentes alternativas existentes en lo que a predicción de función de proteínas se refiere, no solo a través de su secuencias, sino también a través del estudio de sus estructuras tridimensionales, que a pesar de estar estrechamente vinculadas a la secuencia, hay diferencias evidentes en la predicción de la función proteica cuando se analiza a partir de la secuencia o a partir de la estructura. Esta temática supone un campo aún por descubrir a través de todos los recursos que se encuentran disponibles en la red.

La idea inicial del trabajo, consistente en la predicción de la función de una proteína dada su secuencia de aminoácidos, ha sido contestada según se ha desarrollado el trabajo. Obviamente, hay aspectos que deben de mejorarse o incluso cambiar, como hemos comentado anteriormente, pero en líneas generales, y a modo de reto personal, el objetivo general del trabajo se ha cumplido. Aun así, conforme se fue desarrollando el trabajo, y al trabajar con volúmenes de datos elevados, la necesidad de una buena máquina computadora se hizo evidente y limitó en determinados casos algunas de las pruebas que se pretendía realizar, como era el caso de comprobar numerosas secuencias de proteínas en paralelo.

La planificación propuesta para la realización de este trabajo, se estableció sin tener en cuenta los conocimientos necesarios para el desarrollo de las diferentes partes del trabajo, pero a pesar de esto, los puntos se fueron acatando fielmente, reforzando aquellos en los que era necesaria una mayor dedicación, como es trabajar con herramientas desconocidas como phpMyAdmin, MySQL o Pyhton, algo que a pesar de ser bastante laborioso, pudo servir para enriquecer y formar parte de los conocimientos que no se tenían en un principio. Por otro lado, asumir la importancia de la búsqueda bibliográfica especializada y reciente, algo que como ya se ha comentado, es esencial para el desarrollo de este trabajo, fue completamente necesario y muy enriquecedor a la hora de poder desarrollar, innovar y dar forma al trabajo. Durante el tiempo dedicado a la realización de esta actividad, se ha intentado seguir fielmente el plan de trabajo para garantizar el éxito y a pesar, de tener que reforzar determinados aspectos, las modificaciones que se realizaron son mínimas respecto al plan original.

Continuando con el hilo anterior y con vista a líneas de trabajo futuro, la adquisición de todos los conocimientos asumidos durante la realización del presente trabajo, y en el caso de realizar trabajos semejantes, el esfuerzo iría dirigido principalmente al refinamiento y mejora del *script*, pudiendo añadir todas las opciones comentadas y por otro lado, al establecimiento de un banco de pruebas, que asegure un control de

calidad excepcional. Trabajar con las bases de datos disponibles es un lujo al alcance de todos; ya hemos comentado que el futuro sobre el problema de almacenamiento de datos y mantenimiento del sistema es incierto, pero en cualquier caso la necesidad de continuar con el trabajo de análisis de toda la información presente es completamente evidente ya que puede abrir caminos que hoy en día no se conocen. Trabajar con la posibilidad de predecir estructuras, y en la misma medida, trabajar en la predicción de la función de las proteínas, puede generar un extenso desarrollo en diferentes campos biomédicos y farmacológicos, algo que se presenta inevitable y la única cuestión será, cuánto tardaremos en hacerlo.

5. Glosario

DDBJ:	DNA Data Bank of Japan
EBI:	European Bioinformatics Institute
EMBL:	European Bioinformatics Institute
ENA:	European Nucleotide Archive
GO:	Gene Ontology
MGI:	Mouse Genome Informatics
MMDB:	Molecular Modeling Database
NCBI:	National Center for Biotechnology Information
PDB:	Protein Data Bank
PFAM:	Protein Domain Database
PIR:	Protein Information Resource
RefSeq:	NCBI Reference Sequence Database
SCOP:	Structural Classification of Proteins
SGD:	Saccharomyces Genome Database
SIB:	Swiss Institute of Bioinformatics
TIC:	Tecnología de la información y comunicación
UniParc:	UniProt Archive
UniProt:	Universal Protein Knowledgebase

6. Bibliografía

[1] Sanger F, Thompson EOP. The amino-acid sequence in the glycyl chain of insulin. 1. The identification of lower peptides from partial hydrolysates. Biochem. J. 1953;53:353–366.

[2] Cook CE, Bergman MT, Finn RD, et al. The European Bioinformatics Institute in 2016: Data growth and integration. Nucleic Acids Res. 2016;44:D20-26.

[3] Diamant E. Advances in Bioinformatics and Computational Biology: Don't take them too seriously anyway. ArXiv150504785 Cs [Internet]. 2015 [cited 2016 May 30]; Available from: http://arxiv.org/abs/1505.04785.

[4] Apweiler R, Bairoch A, Wu CH. Protein sequence databases. Curr. Opin. Chem. Biol. 2004;8:76–80.

[5] Camon E, Magrane M, Barrell D, et al. The Gene Ontology Annotation (GOA) Database: sharing knowledge in Uniprot with Gene Ontology. Nucleic Acids Res. 2004;32:D262–D266.

[6] Magrane M, Consortium U. UniProt Knowledgebase: a hub of integrated protein data. Database. 2011;2011:bar009.

[7] Gruber TR. A translation approach to portable ontology specifications. Knowl. Acquis. 1993;5:199–220.

[8] Balakrishnan R, Harris MA, Huntley R, et al. A guide to best practices for Gene Ontology (GO) manual annotation. Database J. Biol. Databases Curation [Internet]. 2013 [cited 2016 May 29];2013. Available from: http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3706743/.

[9] Consortium TGO. Creating the Gene Ontology Resource: Design and Implementation. Genome Res. 2001;11:1425–1433.

[10] Consortium GO. The Gene Ontology (GO) database and informatics resource. Nucleic Acids Res. 2004;32:D258–D261.

[11] Consortium GO. The Gene Ontology (GO) project in 2006. Nucleic Acids Res. 2006;34:D322–D326.

[12] Dimmer EC, Huntley RP, Alam-Faruque Y, et al. The UniProt-GO Annotation database in 2011. Nucleic Acids Res. 2012;40:D565–D570.

[13] Ashburner M, Ball CA, Blake JA, et al. Gene Ontology: tool for the unification of biology. Nat. Genet. 2000;25:25–29.

[14] Camon E, Barrell D, Brooksbank C, et al. The Gene Ontology Annotation (GOA) Project—Application of GO in SWISS-PROT, TrEMBL and InterPro. Int. J. Genomics. 2003;4:71–74.

[15] Camon E, Magrane M, Barrell D, et al. The Gene Ontology Annotation (GOA) Project: Implementation of GO in SWISS-PROT, TrEMBL, and InterPro. Genome Res. 2003;13:662–672.

[16] Huntley RP, Sawford T, Mutowo-Meullenet P, et al. The GOA database: Gene Ontology annotation updates for 2015. Nucleic Acids Res. 2015;43:D1057–D1063.

[17] Rost B, Liu J, Nair R, et al. Automatic prediction of protein function. Cell. Mol. Life Sci. CMLS. 2003;60:2637–2650.

[18] Radivojac P, Clark WT, Ronnen Oron T, et al. A large-scale evaluation of computational protein function prediction. Nat. Methods. 2013;10:221–227.

[19] Watson JD, Laskowski RA, Thornton JM. Predicting protein function from sequence and structural data. Curr. Opin. Struct. Biol. 2005;15:275–284.

- Recursos web mencionados (según orden de aparición en el texto):

www.uniprot.org (Mayo, 2016).

http: // www. Geneontology.org/ (Mayo, 2016).

http://www.ncbi.nlm.nih.gov/pubmed (Mayo, 2016).

http://www.uniprot.org/uniprot/ (Mayo, 2016).

https://www.phpmyadmin.net/ (Mayo, 2016).

https://www.python.org/ (Mayo, 2016).

http://biopython.org/ (Mayo, 2016).

http://biopython.org/DIST/docs/api/Bio.pairwise2-module.html (Mayo, 2016).

https://ipython.org/ipython-doc/3/notebook/ (Mayo, 2016).

7. Anexos

A continuación se muestra el *script* utilizado para el desarrollo del presente trabajo fin de máster, dividida en dos partes diferenciadas:

- Conexión a la base de datos creada y query

```
import MySQLdb
import Bio.SeqIO

DB_HOST = 'localhost'
DB_USER = 'root'
DB_PASS = 'root'
DB_NAME = 'proteins'

def run_query(query=''):
    datos = [DB_HOST, DB_USER, DB_PASS, DB_NAME]

    conn = MySQLdb.connect(*datos) # Conectar a la base de datos
    cursor = conn.cursor()         # Crear un cursor
    cursor.execute(query)          # Ejecutar una consulta

    if query.upper().startswith('SELECT'):
        data = cursor.fetchall()   # Traer los resultados de un select
    else:
        conn.commit()              # Hacer efectiva la escritura de datos
        data = None

    cursor.close()                 # Cerrar el cursor
    conn.close()                   # Cerrar la conexión
    return data


run_query(query='SELECT entry,sequence FROM funcion')

def test():
    from Bio import SeqIO
    handle = open("prueba.fasta", "rU") #abrir un fasta creado anterior o uno descargado
    result=""
    resultList = list()
    for record in SeqIO.parse(handle, "fasta"): #parsear el archivo para que detecte las 2
secuencias que hay
        result=result +record.seq
        resultList.append(record.seq) #creo la lista sólo con la secuecia
    return resultList
```

- Consulta y salida: (puntuación dada por *pairwise2*)

```
from Bio import pairwise2
from Bio.SubsMat import MatrixInfo as matlist
import numpy as np
matrix = matlist.blosum62
fullDb = run_query(query='Select entry,sequence From funcion')

scoreList = []
alignments = []
primaryKeys = {} #uso diccionario porque la lista no me comprueba que existe
result = []
```

```
text= []

fastaResult = test()
print(fastaResult)

i=0
for sequence in fullDb:
    for sequence2 in fastaResult:
        for a in pairwise2.align.globalxx(sequence[1],sequence2):
            #print(format_alignment(*a))
            #print(a[-3]/len(sequence2))
            if(a[-3]/len(sequence2)>0.9): #aqui establece el nivel al que dar resultados
                scoreList.append(a[-3]/len(sequence2))
                alignments.append(str(a))
                primaryKeys[i]=sequence[0]
                ++i

for key in primaryKeys:
    #print("select proteinnames from funcion where entry='"+primaryKeys[key]+"'")
    proteinnames=run_query("select proteinnames from funcion where
entry='"+primaryKeys[key]+"'")
    #print(proteinnames)
    gene=run_query("select gene_ontology from funcion where entry='"+primaryKeys[key]+"'")
    #print(proteinnames)
    text.append("Protein name: " + str(proteinnames) + "\n" + "Gene Ontology: " + str(gene) +
"\n")

print(text)
np.savetxt("Results.txt",np.asarray(text),fmt="%s") #formato string
```