Cálculo de la importancia de características y evaluación de la calidad de proteínas en el problema de discriminación de Decoys.

Edwin Gérman Maldonado Távara[1], Marley M.B.R Vellasco[2], Bruno A.C. Horta[3], and Fabio L. Custodio[4]

[1] PUC Rio de Janeiro, Departamento de Engenharia Elétrica, Rio de Janeiro, Brazil, edwin@ele.puc-rio.br,
[2] PUC Rio de Janeiro, Departamento de Engenharia Elétrica, Rio de Janeiro, Brazil, marley@ele.puc-rio.br,
[3] Universidade Federal de Rio de Janeiro, Instituto de Química, Rio de Janeiro, Brazil, bruno@iq.ufrj.br,
[4] Laboratorio Nacional de Computação Científica, LNCC, Petrópolis, Brazil, flc@lncc.br

Resumen. En el cuerpo humano la función de las proteínas esta determinada por la estructura tridimensional, la estructura tridimensional puede ser predicha por métodos computacionales. Los métodos computacionales usados generan una gran cantidad de modelos candidatos(Decoys), para evaluar la calidad de estos existen dos tipos de métodos: (1)basados en la similaridad y (2) Aprendizaje de máquina(AM). El primer grupo es usado cuando la estructura nativa de la proteína es conocida (RMSD, TM-Score, Z-Score). El segundo tipo de métodos usan un sub conjunto de características fisicoquímicas. Estas características son seleccionadas manualmente para ser usadas en el modelo de AM. Haciendo esto se podría estar dejando de lado características sumamente relevantes para la evaluación de la calidad de los modelos candidatos(Decoys). El presente trabajo proporciona tres características de suma importancia: (1)Considera diferentes tipos de características simultáneamente; (2)Proporciona la importancia relativa de los diferentes tipos de características que intervienen en la evaluación de los Decoys; (3)El nuevo modelo, llamado Método Envolvente para Cálculo de la Importancia de Características (MECIC) también calcula la calidad de los modelos de proteína candidatos. Como muestran los resultados, estas tres características del modelo, lo convierten en una poderosa herramienta no solo para la predicción de la calidad de los Decoys sino que también brinda un mejor entendimiento de los factores a ser considerados en su evaluación.

Palabras llave: Método Envolvente para selección de Características, Cálculo de la importancia de Características, Aprendizaje de Máquina(AM), Discriminación de Decoys, Evaluación de la Calidad de Proteínas,Redes Neurales, Algoritmos Genéticos, Método Envolvente para Cálculo de la Importancia de Características (MECIC).

1 Introducción

Las proteínas se encargan de una gran variedad de funciones en los organizmos vivos. Por exemplo, las enzimas son proteínas catalizadoras de reacciones específicas, muchas de las cuales son vitales para estos organizmos. La estructura de las proteínas esta dividida en cuatro niveles [15]: (a) estructura primaria, (b) estructura secundaria, (c) estructura terciária y (d) estructura cuaternaria. La estructura primaria corresponde a la secuencia de aminoácidos(orden lineal) [15] [13]. La estructura secundaria esta definida por los patrones formados por los enlaces de hidrógeno, los patrones mas comunes son helices-α y hojas-β [13].Estas estructuras son altamente estables y consituyen elementos claves en la estructura tridimensional de las proteínas(3D).La estructura terciaria de las proteínas es la distribución de las estructuras secundarias en el espacio 3D. La estructura tridimensional asumida por una proteína es llamada de estructura nativa. El conocimiento de la estructura nativa es vital en la biología molecular pues permite determinar la función de la proteína en las células. Finalmente, la estructura cuaternaria esta definida por proteínas con multiples domínios de estructura terciaria y esta caracterizada por el ordenamiento espacial de estos dominios.

Uno de los principales desafíos en la bioinformática estructural es la determinación de la estructura terciaria de proteínas. Determinar la estructura terciaria de proteínas experimentalmente es computacionalmente costoso [6]. Esto ha creado una brecha entre secuencias y estructuras tridimensionales conocidas. Por lo tanto, existe la necesidad de métodos computacionales para la predicción de la estructura terciaria.

Diversos métodos computacionales han sido propuestos para predecir la estructura terciaria de las proteínas [11] [26]. Estos métodos generan gran cantidad de modelos candidatos conocidos como Decoys [16]. Cuando la estructura nativa es conocida, la calidad de un Decoy puede ser evaluada a través de diferentes medidas de similitud tales como: Root-Mean-Square Deviation (RMSD) [2], Global Distance Test Total Score (GDT_TS) [7], Template-Modeling Score (TM-Score) [27] y Maximal Substructure (MaxSub) [23]. Por otro lado, en ausencia de la estructura nativa han sido propuestas funciones de puntuación las cuales permiten discriminar entre modelos de alta y baja calidad. Estas funciones de puntuación pueden ser clasificadas en cuatro categorías: (1) Funciones potenciales basadas en la física; (2)Funciones estadísticas de potencial; (3) Funciones basadas en consenso; y (4) Algoritmos de aprendizaje de máquina. Funciones potenciales basadas en la física calculan la energía potencial asociada a un Decoy(modelo de proteína candidato), en este cálculo se incorpora las interacciones de este con el solvente. Por otro lado, las funciones basadas en la estadística evalúan la calidad de los Decoys haciendo un análisis estadístico de las propiedades estructurales extraídas de una base de datos de proteínas conocidas [22]. Es importante resaltar que estos dos métodos solo consideran propiedades médias de las estructuras de proteínas conocidas. Esto limita su capacidad para discriminar y calificar modelos estructurales. Funciones basadas en consenso proporcionan un buen desempeño cuando la mayoría de los modelos son similares a su estructura

nativa. Sin embargo, si la base de datos solo esta conformada por modelos de baja calidad, estos modelos tienden a mostrar un desempeño mucho menor que enfoques basados en el conocimiento [26]. Finalmente, algoritmos de Aprendizaje de Máquina, tales como Support Vector Machine(SVM), Neural Networks (NN) y Random Forest(RF), evalúan la calidad de los modelos candidatos usando ciertas características [10] [26]. Estas características son extraídas de la secuencia y de la estructura 3D del modelo para luego ser usadas como entradas de los modelos de AM y así poder evaluar su calidad. La gran ventaja de estos métodos es que consideran varias características al mismo tiempo y de esta forma pueden descubrir relaciones ocultas entre estas, esto es bastante difícil de conseguir con los métodos 1-3 descritos anteriormente. Por otra parte, enfoques de selección de características han sido usados en el contexto de Relación Estructura-Actividad [9] [18].En este tipo de problemas, un gran número de características son generados y utilizadas para mostrar la relación entre las estructuras químicas de las moléculas y su función. Estos enfoques pueden ser usados para extraer las características que serán usadas en técnicas de AM para predecir la calidad de un modelo Decoy. Las características pueden ser extraídas de la estructura secundaria, por ejemplo el número medio de aminoácidos que forman la estructura hélice-α. Otros tipos de características ampliamente utilizadas están basadas en la distancia Euclidiana(por ejemplo, distancia entre los Carbonos Alpha(CA) de dos aminoácidos contiguos). Adicionalmente, las caracteristícas pueden ser extraídas de las propiedades fisicoquímico de los modelos de proteína, tales como el número de Aminoácidos Hidrofóbico-alifáticos y la accesibilidad al solvente en diferentes niveles [19] [4]. Finalmente, la proteína puede ser representada como red, haciendo esto es posible extraer otros tipos de características [5]. Todos estos tipos de características evalúan diferentes aspectos de los modelos Decoy. Sin embargo, aun cuando existen una gran gama de estas caracteristicas, muchos estudios seleccionan un subconjunto el cual es usado como entrada de los modelos de aprendizaje de máquina. Esta selección es realizada típicamente arbitrariamente lo cual implica la obtención de resultados tendenciosos, limitando la habilidad de predicción del modelo usado. Por lo tanto, es altamente recomendable obtener un conjunto imparcial de las características mas importantes de forma automática para predecir de la calidad de Decoys. Este trabajo propone un nuevo método llamado Método Envolvente para el Cálculo de la Importancia de Características (MECIC) para seleccionar un subconjunto óptimo de caracteristícas de la base de datos. Adicionalmente, este modelo proporciona la importancia relativa de las características relacionada a la evaluación de la calidad del modelo Decoy. Esta propiedad del modelo es muy importante para calificar las características que miden diferentes aspectos la calidad de los Decoys. En la próxima sección se proporciona una descripción detallada de este modelo.

2 Modelo Propuesto

Como fue visto en la sección previa, existen diferentes tipos de posibles características y cada método usa un subconjunto de estas [10]. Generalmente estas carac-

terísticas son seleccionadas manualmente y enfocadas a un solo tipo(estructurales, fisicoquímicas, etc). Al usar un reducido grupo de características, se esta dejando de lado propiedades que podrían tener impacto en la evaluación de la calidad de un modelo Decoy. El presente trabajo, adicionalmente a la capacidad de predecir la calidad de un Decoy, tiene las siguientes ventajas adicionales:

- Evaluación en paralelo de diferentes tipos de caraterísticas,
- Selecciona las características que tengan mayor impacto en la determinación de la calidad de modelo candidato(Decoy),
- Finalmente, califica las características usando su importancia relativa.

Método Envolvente para el Cálculo de la Importancia de Características (MECIC)

El modelo MECIC esta conformado por dos módulos: (1) Módulo envolvente y (2) Módulo de importancia de características, como se muestra en la Figura 1. El primero tiene dos submódulos: (a)El submódulo evolucionario y el (b) El submódulo predictivo. E submódulo evolucionario esta conformado por un algoritmo genético que busca el mejor subconjunto de características para optimizar la respuesta en el submódulo predictivo. El submódulo predictivo esta representado por una modelo de Aprendizaje de Máquina. Específicamente el submódulo de Aprendizaje de Máquina puede ser una Red Neuronal, Support Vector Machines, etc. El valor predicho es retornado al módulo evolucionario y este se convierte en la evaluación de cada individuo. El módulo de importancia de características, consiste de dos submódulos: (1) El submódulo registrador de datos, el cual ordena la información evolucionária hasta que el algoritmo genético converja; (2) El submódulo de cálculo de puntuación usa la información salvada por (1) y calcula la puntuación (importancia) para cada característica.

Módulo envolvente Métodos de selección de características han sido ampliamente estudiados y son clasificados en dos grupos : (1) Métodos de filtro, los cuales son aplicados independientemente de la variable objetivo; y los (2)Métodos envolventes, que usan el mismo modelo de Aprendizaje de Maquina para evaluar el rendimiento de un subconjunto de características en la predicción de la variable objetivo [14]. Los métodos envolventes usualmente tienen mejores resultados sin embargo los métodos de filtro son computacionalmente menos caros. En este trabajo el módulo envolvente esta basado en un Algoritmo Genético(AG) y una Red Neuronal como submódulos. Estos submódulos también pueden ser reemplazados por otras técnicas tales como Support Vector Machine(SVM) o Particle Swarm Optimization(PSO). Los detalles de este módulo son descritos a continuación:

(a) SubMódulo Evolucionário: Este submódulo esta basado en un Algoritmo Genético Steady State con individuos de representación binaria . El tamaño de la población es N y cada individuo tiene una longitud L. Es importante remarcar que L es igual al número de características (variables) de la base

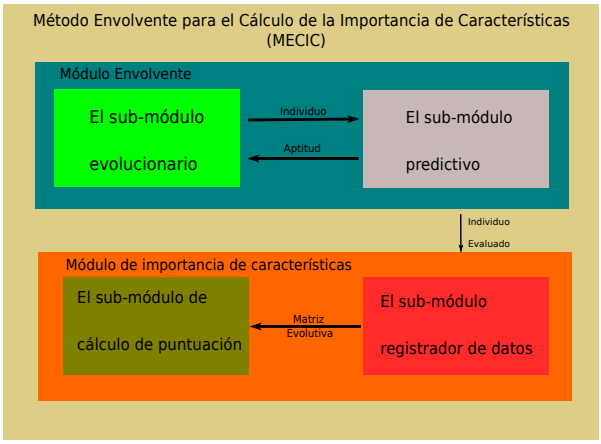

Fig. 1: Método Envolvente para el Cálculo de la Importancia de Características (MECIC)

de datos. Si el Gen de un individuo tiene valor '1' indica que la característica en esa posición ha sido seleccionada, mientras tanto '0' indica lo contrario. Cuando un método envolvente para selección de características es usado, la inicialización de la población debe lidiar con dos problemas : (1) Cuando los individuos tienen varias características seleccionados entonces el tiempo usado para el cálculo de su aptitud es bastante elevado. Por lo tanto, un menor número de individuos serán evaluados en el mismo tiempo. (2) La presencia de características "malas" dentro de un conjunto de "buenas" características pasa desapercibido. Un individuo con esta cualidad puede tener una alta aptitud y no se notará la presencia de la característica no significativa. De este modo, una inicialización aleatoria de la población daría a los genes la misma probabilidad de ser '0' o '1, lo cual no es recomendado. Para evadir este problema la probabilidad de un gen con valor '1' debe ser menor que la probabilidad de ser '0' cuando los individuos son creados. De acuerdo con esto, la Ecuación 1 muestra la probabilidad de un gen tener el valor '1':

$$3/(L) \tag{1}$$

Individuos con menos de tres características seleccionadas son considerados inválidos. Los individuos inválidos son descartados y en su lugar se generan nuevos hasta que se complete la población.

Este trabajo usa un mecanismo de selección por torneo de tamaño T. Para adicionar los nuevos individuos en la población se usa un método de sustitución parental en el cual los nuevos individuos reemplazan los peores la población actual.

(b) SubMódulo predictivo: Este módulo calcula la aptitud de los individuos generados por el submódulo evolucionario. Primero, el individuo es usado

para seleccionar las características de la base de datos que serán usadas como entrada en el módulo predictivo, la salida de este módulo es la aptitud del individuo. En este trabajo, el submódulo predictivo es una Red Neuronal multicamada perceptron [8] [25] en la cual se usa el algoritmo Backpropagation [20] para su entrenamiento. El valor del MSE proporcionado por la red neuronal es la aptitud de cada individuo, este valor que será minimizado en el proceso evolutivo del AG. Finalmente, es empleado el mecanismo de parada anticipada para prevenir el sobre entrenamiento.

Módulo de importancia de características En la etapa de entrenamiento, este módulo es responsable por ordenar la información generada por el módulo envolvente y calcular la importancia relativa. Para hacer esto, este módulo consta de dos submódulos con diferentes tareas: (a)El submódulo registrador de datos (b)El submódulo de cálculo de puntuación. El primero crea todas las estructuras necesarias para guardar la informaci'on generada por el módulo envolvente. Posteriormente, esta información es ordenada usando el algoritmo Quick Sort [12]. El segundo submódulo toma esta información y computa la puntuación para obtener la importancia de los atributos. Estos submódulos serán descritos a continuación:

(a) SubMódulo registrador de datos: Este submódulo sirve como soporte para el submódulo de cálculo de puntuación proporcionando los datos ordenados y las estructurados para el cálculo directo de la importancia de las características. Posteriormente, los mejores individuos por experimento y por generación son salvados para uso posterior. Luego, el mejor individuo de la última generación por experimento es almacenado en la matriz B. Por lo tanto, la matriz B tiene E (máximo número de experimentos) filas y F columnas (máximo número de características). Finalmente, esta matriz es usada como entrada del submódulo de cálculo de puntuación.

(b) El submódulo de cálculo de puntuación: Métodos envolventes tradicionales para selección de características solamente extraen un subconjunto de estos de toda la base datos. En este trabajo se propone un nuevo método envolvente que asigna una importancia (puntuación) para cada característica en relación a la variable objetivo y entre ellas mismas. El método propuesto se describe a continuación:

- Frecuencia media de selección por experimento (FMSE): Esta puntuación trata de capturar la frecuencia en que una característica es seleccionada por el algoritmo genético en un determinado experimento. Por lo tanto, si la característica es frecuentemente seleccionada, esto significa que probablemente tenga una gran importancia para la predicción de la variable objetivo. El cálculo de esta puntucación por característica esta representada por el vector S (2) y calculada por la Ecuación 3, donde el atributo actual es representado por f y F es el número máximo de características en la base de datos; e es el experimento actual y E es número máximo de experimentos.

$$S = [S_1, S_2, S_3...S_F] \tag{2}$$

$$S_f = \sum_{e=1}^{E} B_{ef} \tag{3}$$

3 Configuración del modelo y caso de estudio

La base de datos

Samudrala y Levit presentaron una base de datos de Decoys [21], esta base de datos es conocida como Decoys 'R' y se encuentra disponible en (`http://ram.org/compbio/dd/`). La base de datos en cuestión esta dividida en tres grupos (1)Loop; (2)Multiple y (3)Single. Cada grupo tiene diferentes tipos de Decoys. La base de datos Loop contiene decoys para proteínas de secuencia pequeña. La base datos Single contiene estructuras correctas e incorrectas cuya relación es de $1:1$. Finalmente, la base de datos Multiple que contiene decoys con diferentes valores de RMSD con respecto a su estructura nativa. Este grupo esta conformado a su vez por diez subconjuntos, por ser este un estudio preliminar en este trabajo solo serán usados dos (4Stated_reduced) [17] y Fisa_casp3 [24] con siete y seis proteínas respectivamente).

Pre-procesamiento de los datos

Los archivos de la base de datos de Decoys presentada anteriormente están en formato PDB. Este formato proporciona una representación estándar para estructuras macro-moleculares derivadas de métodos experimentales. La documentación acerca de este formato puede ser encontrada [1] o `https://www.rcsb.org/`. Por lo antes hablado, es necesario un pre-procesamiento para extraer las características empleadas por el modelo. La etapa de pre-procesamiento consta de seis etapas:

(a) La biblioteca Biopython [1] disponible en (`http://biopython.org/wiki/Main_Page`) es empleada para calcular las características 3-5,15-19 mostrados en la Tabla 1.
(b) El programa POPS disponible en `http://mathbio.nimr.mrc.ac.uk/wiki/POPS` es usado para calcular el área de la superfície que está en contacto con el solvente de los aminoácidos hidrofóbico e hidrofílicos, detalles de estas características se muestran en la Tabla 1.
(c) Las características 6-13 de la Tabla 1 están relacionadas con la estructura secundaria y son calculadas con el programa DSSP disponible en `http://swift.cmbi.ru.nl/gv/dssp/`.
(d) La característica 14 de la Tabla 1 es calculada usando un script en python (Radio de Giro M1).
(e) El programa TM-Score disponible en `http://zhanglab.ccmb.med.umich.edu/TM-score/` es empleado para calcular el RMSD(después de una alinación óptima de estructuras) de los modelos candidatos. Tener en cuenta que el RMSD es la variable objetivo. Los pasos de a-e son repetidos hasta que toda la base de datos inicial sea procesada.
(f) Finalmente, las características son normalizadas. El tipo de normalización de cada una es mostrada en la Tabla 1

Table 1: Descripción de la Base de Datos

Id	Descripción	Nombre	Normalización
1	Superficie Aminoácidos Hidrofílicos	pops_hidrofilic	$(x - min)/(max - min)$
2	Superficie Aminoácidos Hidrofólicos	pops_hydrofobic	$(x - min)/(max - min)$
3	Carbono alfa HSE	hse_ca	$(x - min)/(max - min)$
4	Carbono beta HSE	hse_cb	$(x - min)/(max - min)$
5	Número de Contactos HSE	hse_cn	$(x - min)/(max - min)$
6	Número de Alfa hélices	countH	$x/len(protein)$
7	Número de Hojas Beta	countB	$x/len(protein)$
8	Número de Hilos extendidos	countE	$x/len(protein)$
9	Número de 3-helix	countG	$x/len(protein)$
10	Número de 5-helix	countI	$x/len(protein)$
11	Número de puentes de hidrógeno	countT	$x/len(protein)$
12	Número de curvas	countS	$x/len(protein)$
13	Otras estructuras secundarias	countO	$x/len(protein)$
14	Radio de Giro M1	ragy1	$(x - min)/(max - min)$
15	Radio de Giro M2	ragy2	$(x - min)/(max - min)$
16	Distancia média carbono alfa	distCa	$(x - min)/(max - min)$
17	Distancia média carbono beta	distCb	$(x - min)/(max - min)$
18	Ángulo PHY médio	avgPhi	$(x - min)/(max - min)$
19	Ángulo PSI médio	avgPsi	$(x - min)/(max - min)$

Configuración del submódulo evolucionario

En este modelo, el tamaño de la población es $N = 100$. Como fue visto, la principal característica de la población inicial es que la probabilidad que tiene un gen de ser '1' en un individuo es $3/L$, donde $L = 19$ es el número de características. El número de nuevos individuos creados en cada generación corresponde al $k = 20\%$ de la población,posteriormente estos reemplazaran los peores individuos de la población actual. Para asegurar una presión selectiva alta en este modelo el tamaño del torneo es $T = 2$. Finalmente, este modelo considera $G = 100$ generaciones y $E = 100$ experimentos. La Tabla 2 (a) presenta un resumen de los parámetros antes presentados.

Configuración del submódulo predictivo

Como fue especificado en la sección anterior, este trabajo emplea un red neuronal multicamada perceptron. El algoritmo usado para entrenar este modelo es Back-propagation. El número de capas ocultas y el número de neuronas en estas capas es determinado utilizando una estrategia de prueba y error. En este modelo, la red neuronal es entrenada usando validación cruzada [3]. Por lo tanto, después de la selección características, la base de datos es dividida en tres subconjuntos para entrenamiento (60%), validación (30%) y teste (10%). La Tabla 2 (b) muestra los parámetros.

Table 2: Configuración: (a) Submódulo Evolucionario; (b) Submódulo Predictivo

(a)

Parámetro	Valor
Número de Experimentos	100
Número de Generaciones	100
Tamaño de la Población	100
Porcentaje Steady State	20%
Prob. de Cruzamiento	0.98
Prob. de Bitwise en la Mutación	0.05
Prob. de la Mutación	0.1
Prob. de un Gen ser '1'	0.157
Tamaño del Torneo	2

(b)

Parámetro	Valor
Número Entradas	19
Neuronas Camada escondida	100
Neuronas Camada de salida	1
Tasa de Aprendizaje	0.15%
Tasa de Momento	0.05
Función de Activación	Sigmoid

4 Resultados

Evaluación del desempeño del Algoritmo Genético

El desempeño de un algoritmo genénetico puede ser evaluado usando dos criterios: (1)La evolución de la aptitud; y (2) La evolución del desviación estándar. En el primer caso, la média de la aptitud del mejor individuo por generación y por experimento es usada para verificar si el algoritmo genénetico esta mejorando la aptitud através de la evolución. En este trabajo, en todos los casos, los resultados de la evolución en relación a la aptitud muestran un comportamiento normal, lo que significa que en etapas iniciales de la evolución de la aptitud tiene valores grandes y a medida que la evolución avanza la aptitud decrece hasta que se estabiliza (problema de minimización). Por otro lado, la desviación estándar presenta un comportamiento atípico para la mayoría de la proteínas. Este comportamiento podría estar siendo causado por la inicialización aleatoria de los pesos de la red neuronal. Consecuentemente, es posible encontrar individuos idénticos en la población con una pequeña diferencia en el valor de aptitud.

Resultados de la importancia de las características

La puntuación es calculada para cada subconjunto de proteínas. La Tabla 3 presenta estos resultados y la media de la puntuación por característica. Usando la media de la puntuación, las primeras siete características son seleccionadas, es fácil notar que algunos de estas son seleccionadas como importantes en ambas base de datos, como se muestra en la Figura 2. Esta información es relevante para determinar las características que tienen mayor influencia al evaluar la calidad de un Decoy.

Para cada base de datos, la importancia de las características es calculada usando la puntuación mencionada anteriormente. La importancia calculada de estas características es mostrada en la Tabla 3 para las base de datos (a)

4Stated_reduced y (b) Fisa_casp3. Haciendo uso de las siete primeras características, es posible observar un conjunto de características comunes con valores altos de importancia. Esta información es relevante pues através de ella se han identificado características que intervienen directamente en la evaluación de la calidad de modelos candidatos de proteína. Se nota fácilmente que la característica *countH*, relacionada con la estructura alfa-hélices de la estructura secundaria es identificada como importante en ambas base de datos. Adicionalmente, la característica rayGy_M2, relacionada con la compactación de la proteína es identificada como importante. Este resultado es interesante pues las proteínas en su estado nativo tiende a ser mas compacta; entonces, si el Decoy (modelo de proteína candidato) es muy similar a su estructura nativa este será compacto. La Figura 2 muestra las características que son identificadas como importantes en ambas base de datos.

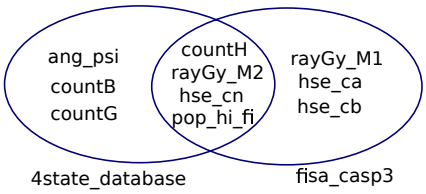

Fig. 2: Primeras siete características identificadas como importantes en ambas base de datos(4Stated_reduced, Fisa_casp3).

5 Conclusión

Esta investigación trata de responder la pregunta: que características son los más importantes para seleccionar los mejores modelos candidatos en la predicción de estructuras de proteínas. Los métodos tradicionales de Aprendizaje de Máquina solamente seleccionan un subconjunto de características del mismo tipo. Por otro lado, estos métodos no proporcionan ninguna puntuación de las características. En este contexto, esta investigación propone un nuevo método llamado Método Envolvente para Cálculo de la Importancia de Características (MECIC). MECIC proporciona la importancia relativa de cada característica, no solamente en relacion a la variable objetivo sino también entre ellas mismas. En este trabajo, el modelo MECIC usó un algoritmo genético con una red neuronal para el submódulo evolucionario y para el submódulo predictivo respectivamente. Los testes fueron hechos en dos base de datos: (1) 4Stated_database y (2)Fisa_casp3, y los resultados proporcionados indican que el modelo propuesto es una herramienta muy prometedora como método envolvente que proporciona también las importancias de las características. Como trabajo futuro, se pueden desarrollar nuevas métricas para la puntuación de las características para luego hacer una comparación de estas y descubrir cual de estas es la mejor.

Table 3: Importancia por caraterística: (a)4State; (b)Fisa_casp3

(a)

Carac	1ctf	1r69	1sn3	2cro	3cib	4pti	4rnx	Prom
ang_psi	0.97	1.02	0.95	1.00	0.99	0.99	1.04	0.9937
countH	0.95	1.01	0.66	0.71	0.71	0.56	0.61	0.7449
rayGy_M2	0.95	1.00	0.87	0.93	0.93	0.23	0.28	0.7398
countB	0.26	0.31	0.70	0.75	0.75	0.62	0.67	0.5808
hse_cn	0.29	0.34	0.46	0.51	0.51	0.78	0.83	0.5302
pop_hi_fi	0.28	0.33	0.33	0.38	0.39	0.81	0.86	0.4830
countG	0.41	0.46	0.53	0.58	0.58	0.24	0.29	0.4425
countE	0.27	0.32	0.46	0.52	0.51	0.48	0.52	0.4408
countl	0.37	0.42	0.38	0.43	0.43	0.40	0.45	0.4128
hse_cb	0.47	0.52	0.22	0.27	0.27	0.48	0.53	0.3963
countT	0.12	0.17	0.21	0.26	0.26	0.81	0.86	0.3842
hse_cb	0.28	0.33	0.35	0.40	0.39	0.30	0.35	0.3419
countO	0.25	0.30	0.17	0.22	0.22	0.34	0.39	0.2678
pop_hi_fo	0.17	0.22	0.28	0.33	0.33	0.23	0.29	0.2639
dist_ca	0.22	0.27	0.22	0.27	0.26	0.27	0.32	0.2599
countS	0.18	0.23	0.17	0.22	0.22	0.29	0.34	0.2364
dist_cb	0.15	0.20	0.17	0.22	0.22	0.17	0.22	0.1943
ang_phi	0.18	0.23	0.12	0.17	0.17	0.19	0.24	0.1854
rayGy_M1	0.19	0.24	0.15	0.20	0.20	0.13	0.18	0.1853

(b)

Carac	1bg8-A	1eh2	1jwe	130	1bl0	smd3	Prom
rayGy_M2	0.49	0.57	0.59	0.43	0.57	0.67	0.553
rayGy_M1	0.56	0.53	0.54	0.51	0.49	0.42	0.509
hse_cn	0.37	0.18	0.25	0.12	0.62	0.19	0.288
hse_ca	0.33	0.23	0.43	0.19	0.29	0.20	0.279
pop_hi_fi	0.15	0.13	0.23	0.10	0.21	0.67	0.248
hse_cb	0.26	0.17	0.37	0.15	0.26	0.25	0.241
countH	0.09	0.66	0.17	0.12	0.15	0.17	0.226
pop_hi_fo	0.12	0.10	0.25	0.42	0.18	0.19	0.210
countS	0.13	0.14	0.17	0.13	0.16	0.19	0.152
ang_phi	0.15	0.21	0.15	0.10	0.11	0.19	0.152
countT	0.15	0.19	0.13	0.14	0.16	0.11	0.144
ang_psi	0.20	0.18	0.13	0.12	0.15	0.09	0.143
countG	0.10	0.13	0.10	0.12	0.10	0.24	0.130
dist_cb	0.16	0.11	0.11	0.13	0.12	0.14	0.127
countl	0.16	0.11	0.11	0.12	0.15	0.11	0.125
countE	0.08	0.13	0.12	0.14	0.13	0.14	0.123
countB	0.08	0.15	0.10	0.14	0.12	0.14	0.123
dist_ca	0.17	0.10	0.10	0.14	0.11	0.11	0.119
countO	0.11	0.07	0.04	0.15	0.15	0.13	0.109

References

1. Berman, H., Henrick, K., Nakamura, H.: Announcing the worldwide Protein Data Bank. Nature Structural Biology 10(12), 980–980 (dec 2003), http://www.nature.com/doifinder/10.1038/nsb1203-980
2. Brüschweiler, R.: Efficient RMSD measures for the comparison of two molecular ensembles. Root-mean-square deviation. Proteins 50(1), 26–34 (jan 2003), http://www.ncbi.nlm.nih.gov/pubmed/12471596
3. Burman, P., Chow, E., Nolan, D.: A cross-validatory method for dependent data (1994)
4. Faraggi, E., Kloczkowski, A.: A global machine learning based scoring function for protein structure prediction. Proteins 82(5), 752–9 (2014), http://www.ncbi.nlm.nih.gov/pubmed/24264942
5. Ghosh, S., Vishveshwara, S.: Ranking the quality of protein structure models using sidechain based network properties. F1000Research 17(MAY), 1–9 (2014), http://f1000research.com/articles/3-17/v1
6. Güntert, P.: Automated NMR structure calculation with CYANA. Methods in molecular biology (Clifton, N.J.) 278, 353–378 (2004), http://www.ncbi.nlm.nih.gov/pubmed/15318003
7. Handl, J., Knowles, J., Lovell, S.C.: Artefacts and biases affecting the evaluation of scoring functions on decoy sets for protein structure prediction. Bioinformatics (Oxford, England) 25(10), 1271–9 (may 2009), http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2677743{&}tool=pmcentrez{&}rendertype=abstract
8. Haykin, S.S.: Neural Networks: A Comprehensive Foundation. Macmillan (1994), https://books.google.com.br/books?id=PSAPAQAAMAAJ
9. Inglis, G., Thomas, M., Thomas, D., Kalmokoff, M., Brooks, S., Selinger, L.: Molecular Methods to Measure Intestinal Bacteria: A Review. Journal of AOAC ... 95(1), 5–24 (2012), http://www.ingentaconnect.com/content/aoac/jaoac/2012/00000095/00000001/art00003
10. Jing, X., Wang, K., Lu, R., Dong, Q.: Sorting protein decoys by machine- learning-to-rank. Nature Publishing Group (April), 1–11 (2016)

11. Kelly, L., Mezulis, S., Yates, C., Wass, M., Sternberg, M.: The Phyre2 web portal for protein modelling, prediction, and analysis. Nature Protocols 10(6), 845–858 (2015), http://dx.doi.org/10.1038/nprot.2015-053
12. Kocher, G., Agrawal, N.: Analysis and Review of Sorting Algorithms. Analysis 2(3), 81–84 (2014), http://www.ijser.in/archives/v2i3/SjIwMTMxODE=.pdf
13. Lehninger, A., Nelson, D., Cox, M.: Principles of Biochemistry, vol. 17. New York, 4th edn. (2005)
14. Liu, H., Motoda, H.: Feature Extraction, Construction and Selection: A Data Mining Perspective. Kluwer Academic Publisher, Norwell MA, USA (1998)
15. Lodish, H., Berk, A., Matsudaira, P., Kaiser, C., Krieger, M., Scott, M.: Molecular Cell Biology, vol. 60. New York, 5th edn. (1990)
16. Martin, A.J.M., Vullo, A., Pollastri, G.: Neural Network Pairwise Interaction Fields for Protein Model Quality Assessment. pp. 235–248 (2009), http://link.springer.com/10.1007/978-3-642-11169-3{_}17
17. Park, B.H., Levitt, M.: Energy functions that discriminate x-ray and near-native fold from well-constructed decoys. Journal of Molecular Biology 258, 367–392 (1996)
18. Rana, P.S., Sharma, H., Bhattacharya, M., Shukla, A.: Quality assessment of modeled protein structure using physicochemical properties. Journal of Bioinformatics and Computational Biology 13(02), 1550005 (apr 2015), http://www.worldscientific.com/doi/10.1142/S0219720015500055
19. Rana, P.S., Sharma, H., Bhattacharya, M., Shukla, A.: Quality assessment of modeled protein structure using physicochemical properties. Journal of Bioinformatics and Computational Biology 13(02), 1550005 (2015), http://www.worldscientific.com/doi/abs/10.1142/S0219720015500055
20. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Neurocomputing: Foundations of Research. chap. Learning R, pp. 696–699. MIT Press, Cambridge, MA, USA (1988), http://dl.acm.org/citation.cfm?id=65669.104451
21. Samudrala, R., Levitt, M.: Decoys 'R' Us: a database of incorrect conformations to improve protein structure prediction. Protein science : a publication of the Protein Society 9(7), 1399–401 (jul 2000), http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2144680{&}tool=pmcentrez{&}rendertype=abstract
22. Sarti, E.: Assessing the structure of proteins and protein complexes through. Ph.D. thesis (2015)
23. Siew, N., Elofsson, a., Rychlewski, L., Fischer, D.: MaxSub: an automated measure for the assessment of protein structure prediction quality. Bioinformatics (Oxford, England) 16(9), 776–785 (2000)
24. Simons, K.T., Kooperberg, C., Huang, E., Baker, D.: Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and Bayesian scoring functions. Journal of molecular biology 268(1), 209–225 (1997)
25. Wasserman, P.D.: Neural Computing: Theory and Practice. Van Nostrand Reinhold Co., New York, NY, USA (1989)
26. Yang, J., Yan, R., Roy, A., Xu, D., Poisson, J., Zhang, Y.: The I-TASSER Suite: protein structure and function prediction. Nature Methods 12(1), 7–8 (2015), http://dx.doi.org/10.1038/nmeth.3213{%}5Cnpapers3://publication/doi/10.1038/nmeth.3213
27. Zhang, Y., Skolnick, J.: Scoring function for automated assessment of protein structure template quality. Proteins 57(4), 702–10 (dec 2004), http://www.ncbi.nlm.nih.gov/pubmed/15476259