Método para la selección de características en la clasificación de péptidos antimicrobianos.

Jesús Armando Beltrán-Verdugo, Carlos A. Brizuela
Departamento de Ciencias de la Computación
Centro de Investigación Científica
y de Educación Superior de Ensenada
Ensenada, B.C., México 22860
Email: {abeltran, cbrizuel}@cicese.edu.mx

***Resumen*—Los péptidos antimicrobianos (AMPs) son una alternativa potencial para combatir los patógenos resistentes a los antibióticos. Actualmente, el diseño *in silico* de AMPs es un área prometedora para asistir al diseño experimental en proporcionar un conjunto de AMPs candidatos. Un aspecto importante en el diseño *in silico* es crear un predictor de AMPs con buen desempeño. QSAR (*Quantitative Structure-Activity Relationship*) es el método más usado para crear modelos de clasificación, puesto que relacionan las propiedades fisicoquímicas (descriptores moleculares) del péptido con la actividad biológica. Un problema en QSAR es la selección de los descriptores que representen las propiedades relevantes del péptidos. En el presente artículo se describe un algoritmo genético y una máquina de soporte vectorial para la selección de descriptores moleculares útiles para la predicción de AMPs.**

I. Introducción

El problema de patógenos resistentes a los antibióticos convencionales se incrementó en las últimas décadas, requiriendo de nuevos enfoques para el tratamiento de infecciones. Los péptidos antimicrobianos (AMPs) son una alternativa potencial para el diseño de nuevos fármacos debido a que muestran una actividad microbicida hacia bacterias, hongos, parásitos y virus [1], sin embargo presentan altos niveles de toxicidad. Una oportunidad de investigación es el diseño de nuevos péptidos que tengan una alta actividad antimicrobiana sin exhibir altos niveles de toxicidad (*i.e.,* deben tener un alto índice terapéutico) [12].

Las investigaciones recientes en AMPs se enfocan en métodos que utilizan una gran cantidad de secuencias con actividad biológica conocida para obtener información que ayude en la predicción de la actividad de nuevos péptidos [7]. Unos de los métodos más usados es QSAR (*Quantitative Structure-Activity Relationship*), debido a que relaciona las propiedades fisicoquímicas cuantificables en los péptidos (descriptores moleculares) con la actividad biológica (*i.e.*, clasificar los péptidos en AMPs y no AMPs) [7], [8]. Para asociar la información del péptido con la actividad biológica, se utiliza un modelo estadístico que se construye mediante algoritmos de aprendizaje de máquina (*e.g.*, redes neuronales, máquinas de soporte vectorial).

Un aspecto importante para la construcción del modelo de clasificación es la selección de los descriptores moleculares. Actualmente, existen miles de descriptores medibles en los péptidos (*e.g.,* el programa Dragon6 puede calcular 4885 descriptores [11]), por lo que elegir los descriptores moleculares que capturen las propiedades relevantes de los AMPs se torna una tarea difícil [8]. Idealmente, este problema puede ser resuelto mediante la selección automática de los descriptores a través de un método denominado selección de características (FS) [9]. De manera general, los métodos de FS tratan de encontrar el subconjunto de características (descriptores) que maximice algún criterio de evaluación (*e.g.,* exactitud de clasificación) dado un conjunto de características.

Un método de FS está compuesto principalmente de dos elementos: una estrategia de búsqueda para la generación de posibles subconjuntos de características y un criterio de evaluación para determinar el desempeño del subconjunto.

El presente trabajo se enfoca en encontrar un subconjunto de descriptores moleculares útiles para la construcción de un predictor de AMPs. El algoritmo de FS recibe como entrada los péptidos representados como descriptores moleculares y da como salida el subconjunto de descriptores que maximicen la exactitud del clasificador. En este artículo se reporta un algoritmo genético (GA), y una máquina de soporte vectorial (SVM) lineal para la selección de un subconjunto de descriptores moleculares relevantes para la identificación de AMPs.

II. Materiales y métodos

II-A. Conjunto de datos

La base de datos CAMP (*Collection of Antimicrobial Péptide*) [13] se utilizó para construir el conjunto de datos positivos, seleccionando sólo los péptidos con longitud de 10 a 100 aminoácidos, que estuvieran experimentalmente validados. Después, se descartaron las secuencias con aminoácidos no estándares. Por último, con el objetivo de tener un conjunto de prueba no redundante se creo un conglomerado de péptidos con BlastClust [4] utilizando un 50 % de identidad y se seleccionó un representante por conglomerado. La base de datos Unitprot [2] fue usada para construir el conjunto de datos negativos, seleccionando proteínas de longitud de 10 a 100 aminoácidos y después se aplicó un filtro basado en la metodología propuesta en [6]. En total, el conjunto resultante de péptidos con y sin actividad antimicrobiana fueron de 870 en ambos casos.

II-B. Cálculo de características

A cada péptido recolectado se le calculó sus descriptores moleculares, lo que involucra transformar la secuencia primaria del péptido en un conjunto de números que capturen

las propiedades fiscoquímicas. En nuestro caso, se calcularon descriptores moleculares que dependen de la estructura molecular de los péptidos; los tipos de descriptores son los siguientes: descriptores de dimensión cero (0D) que contienen información derivada de la frecuencia de los residuos (*e.g.*, carga neta, peso molecular); descriptores de dimensión 1D, contienen información acerca de fragmentos del péptido (*e.g.,* distancia entre dos residuos de triptófano); descriptores 2D, se les conoce como grafos invariantes y contienen información derivada de un grafo molecular [11].

Para calcular los descriptores moleculares se utilizó el programa PaDel-Descriptor [14]. Como resultado de este procedimiento obtuvimos el conjunto de péptidos representados por 770 características, por convención llamaremos a este conjunto como AMP_B.

II-C. Algoritmo genético para la selección de características

Se propone un algoritmo genético para la selección de características (GAFS), donde cada individuo en la población representa un subconjunto de características. El objetivo es encontrar el subconjunto que satisfaga la siguiente expresión:

$$G_{opt} = \arg\max_{G \in \mathcal{G}} Fitness(G) \tag{1}$$

donde G es la representación de un subconjunto de características. El Algoritmo 1 muestra el pseudocódigo de la propuesta de solución.

Algoritmo 1 Algoritmo genético para la selección de caracaterísticas (GAFS)

Entrada: datos de entrenamiento $\mathcal{D}$ con características X,$|X| = n$,
J medida de evaluación a maximizar,
n_g número máximo de generaciones,
n_{gwi} número de generaciones sin mejora,
n_i número de individuos en la población I
n_P número de padres,
p_c probabilidad de cruzamiento,
p_m probabilidad de mutación

Salida: subconjunto de características X' y el valor del criterio de evaluación $J(X')$

1: Generar una población I inicial aleatoria de tamaño n_i
2: Calcular la aptitud para cada invididuo
3: **repetir**
4: Seleccionar a los padres P de la población I
5: Aplicar operador de cruzamiento a P con una probabilidad p_c para generar los hijos O
6: Aplicar operador de mutación a O con una probabilidad p_m
7: Calcular la aptitud para cada invididuo en O
8: Seleccionar a los sobrevivientes de $I + O$ para la siguiente generación
9: **hasta que** el número de generaciones sea igual n_g

1) Representación: Dado un conjunto de características $X = \{X_1, ..., X_n\}$, un individuo es un subconjunto $X_G \subseteq X$ representado por el vector G, entonces,

$G = (g_1, g_2, g_3, ..., g_m)$ donde $X_G = \{X_{g_1}, X_{g_2}, ..., X_{g_m}\}$

tal que, $m \leq n$,

$g_i \neq g_j$, $i \neq j$ $\forall i \in \{1, 2, \ldots, m\}$

$g_i = k$, para $1 \leq k \leq n$, si X_k es parte de la solución

$g_1 < g_2 < \ldots < g_m$

Esta representación nos permite que cada característica esté representado por un entero. Un ejemplo de una representación factible se muestra en la Fig. 1.

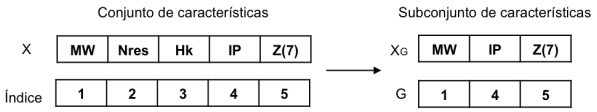


Figura 1: Representación de una solución factible en el algoritmo genético para la selección de características. En el lado izquierdo se muestra todas las características con sus respectivos índices y en el lado derecho los índices seleccionados.

2) Función objetivo: Para evaluar la calidad de los individuos (*i.e.,* subconjuntos) se propone la siguiente función objetivo:

$$Fitness(G) = J(X_G, D')$$

donde X_G corresponde al subconjunto de características codificadas en el genotipo G, y $D' \subseteq D$ es el conjunto de entrenamiento removiendo las variables que no estén en X_G, es decir, $\mathcal{D}' = \cup_{i=1}^{p}\{(x_i, y_i)|x_i \in R^{|X_G|},\ y_i \in \{0, 1\}\}$. $x_i =< x_{i1}, ..., x_{i|X_G|} >$ es un vector de números reales que toma el subconjunto de características $X_G = \{X_{g1}, ..., X_{gm}\}$ tal que, $X_{g1} = x_{i1}, ..., X_{gm} = x_{i|X_G|}$. Un ejemplo del conjunto de entrenamiento se muestra en la Tabla I.

Para definir la función de evaluación J es necesario introducir primero algunas definiciones básicas. Los conjuntos de prueba están formados por un grupo de casos positivos y un grupo de casos negativos. Cuando el predictor acierta en la etiqueta de un elemento que pertenece a los casos positivos se le conoce como verdadero positivo (TP), sin embargo, cuando no lo reconoce se tiene un falso negativo (FN). De otra manera, cuando el predictor se equivoca en la clasificación de un elemento que pertenece a los casos negativos se le conoce como falso positivo (FP) y cuando no se equivoca se tiene un verdadero negativo (TN). A partir de las comparaciones entre el valor esperado y el arrojado por el predictor se definen las siguientes medidas de calidad:

$$ACC(\mathcal{I}(\mathcal{D}')) = \frac{TP + FN}{TP + FN + TN + FP} 100 \tag{2}$$

$$Sens(\mathcal{I}(\mathcal{D}')) = \frac{TP}{TP + FN} \tag{3}$$

$$Espec(\mathcal{I}(\mathcal{D}')) = \frac{TN}{TN + FP} \tag{4}$$

donde $\mathcal{I}$, es una máquina de soporte vectorial (SVM) lineal, ACC es la exactitud del clasificador $\mathcal{I}$, $Espec$ es la medida de especificidad y $Sens$, es la sensibilidad del clasificador.

Con base en las medidas de calidad definidas previamente se propone la siguiente función de evaluación:

$$J(X_G, \mathcal{D}') = ACC(\mathcal{I}(\mathcal{D}')) + 1 - \mid Sens - Espec \mid + 1 - \frac{\mid X_G \mid}{\mid X \mid} \quad (5)$$

El intervalo de valores que puede tomar la función J es [0,102), donde la exactitud (ACC) en J tiene mayor importancia y el resto de los términos de la función sirven como criterios de desempate. La expresión $1 - \mid Sens - Espec \mid$ da un mayor peso a individuos que tengan la especificidad y sensibilidad similares. Por otra parte, el término $1 - (|X_G|/|X|)$ sirve para dar un mayor peso a individuos que tengan un menor número de características.

Tabla I: Ejemplo para el conjunto de datos de entrenamiento

Conjunto de entrenamiento $\mathcal{D}$

X

He	Hk	Z(pH5)	Z(pH7)	Z(pH9)	IP	Clase
-0.31	-1.15	-4.95	-5.9	-6.19	3.67	0
-0.18	-0.44	6.79	4.18	3.63	10.43	1
-0.21	-0.67	9.2	5.19	2.81	9.89	1
-0.01	0.41	0.21	-0.26	-5.11	5.97	0
-0.14	-0.25	0.21	-0.25	-5.11	5.97	0

Conjunto de entrenamiento $\mathcal{D}'$

$G =< 1, 2, 6 >$ $X_G = \{He, Hk, IP\}$

He	Hk	IP	Clase
-0.31	-1.15	3.67	0
-0.18	-0.44	10.43	1
-0.21	-0.67	9.89	1
-0.01	0.41	5.97	0
-0.14	-0.25	5.97	0

3) Pasos principales en GAFS: El algoritmo 1 inicia creando una población de n_i subconjuntos de características. Cada subconjunto de características se selecciona de manera aleatoria de las características disponibles en el conjunto X. Para cada individuo (subconjunto) se evalua la función de aptitud, lo que involucra crear un clasificador (SVM-lineal) por cada uno. Una vez que se tiene la evaluación de la población, se seleccionan los mejores individuo mediante la estrategia de torneo binario, esta estrategia consiste en seleccionar al azar $n_i/2$ parejas de individuos, donde n_i es el tamaño de la población.

El operador de cruzamiento se aplicó a los padres seleccionados con una probabilidad p_c. Para este proceso, se utilizó el cruzamiento SSOF (*Subset size-Oriented Common Feature*) [5], este operador permite mantener bloques informativos comunes, los padres p_i y p_{i-1} heredan a los hijos las características que ambos tiene en común. Por otra parte, las características no compartidas son seleccionadas para heradarse a los hijos con una probabilidad $Prob(h_{p_i}) = (n_{p_i} - n_c)/n$, donde n_{pi} es el número de características del padre p_i, n_c son las características comunes y n es el números total de características. Un ejemplo de este cruzamiento se muestra en la Fig. 2.

La mutación es aplicada con una probabilidad p_m a los hijos. Si el individuo es seleccionado para la mutación entonces se eligen k números para agregar o eliminar en su cromosoma, esto dependiendo si los números están presentes o ausentes en el cromosoma. Con el objetivo de que el cromosoma sufra pequeñas variaciones, k toma desde 1 hasta el 10 % de las n características totales.

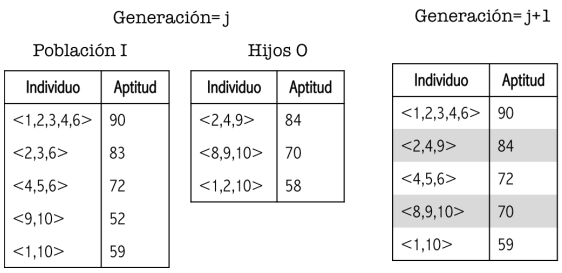


Figura 3: Ejemplo de selección de los sobrevivientes.

Para la selección de los individuos que sobreviviran en la siguiente generación de la población, se utilizó el método de reemplazar al peor. Este método consiste en ordenar de manera descendente (de acuerdo a la función de aptitud) a los individuos de la población I y los hijos O de acuerdo a su aptitud, cada hijo recorre la población I en orden descendente, si existe un individuo j que tenga una aptitud menor a la del hijo, entonces el hijo reemplaza a j en la población I (ver Fig. 3).

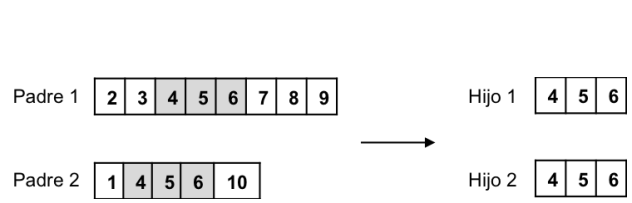
III. Pruebas y resultados

El algoritmo genético para la selección de características (GAFS) fue aplicado al conjunto de datos descrito en la Sección II-A. El conjunto de datos está compuesto por 1740 péptidos (870 AMPs y 870 no AMPs), de los cuales se seleccionaron de manera aleatoria el 90 % de los péptidos para entrenamiento y 10 % para pruebas.

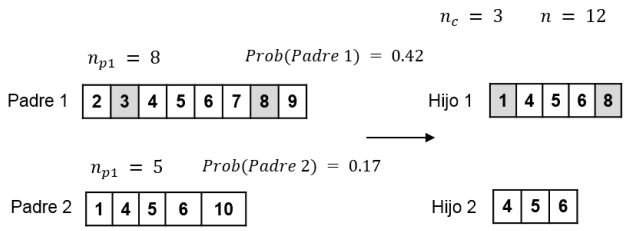
GAFS se ejecutó 30 veces usando un tamaño de población de 250 individuos y un número de generaciones $n_g = 150$ como criterio de terminación del algoritmo. La probabilidad de cruzamiento y mutación fueron de $p_c = 0.8$ y $p_m = 0.4$, respectivamente. GAFS fue codificado en Java y la SVM lineal se implementó utilizando las librerías para Java de Weka 3.6.10 [10], [3]. Los experimentos se realizaron bajo el sistema operativo Windows 7, versión Home Premium 64-bits, en una computadora con procesador Intel (TM) Core (R) i7 de 3.6 GHz de velocidad y memoria RAM de 8 GB.

En la Tabla II, se muestra la calidad promedio de las mejores soluciones encontradas por GAFS para el conjunto de prueba AMP_B. En los resultados se puede observar que en términos de características el algoritmo GAFS disminuyó alrededor de 41.05 % del total de características y en lo que respecta a la aptitud, el algoritmo aumentó un 2.42 % con respecto a la aptitud obtenida al utilizar todas las características. Por otra parte, el mejor subconjunto de características que se encontró fue de 63 características con una aptitud de 94.35. El tiempo de ejecución promedio de GAFS es de 18.87 horas, en donde el cuello de botella del algoritmo es en la evaluación de cada individuo debido a que implica construir y evaluar la máquina de soporte vectorial por cada uno.

Con el mejor subconjunto de características se aplicó un clasificador utilizando una SVM-lineal (ver Tabla III). El clasificador en general obtuvo una exactitud del 92.47 %.



(a) Los padres heredan a los hijos las características que ambos tiene en común



(b) Los padres heredan a los hijos las características que ambos no tiene en común con una probabilidad. En este ejemplo, el $hijo\ 2$ no hereda más característica debido a que la probabilidad del $padre\ 2$ es muy pequeña.

Figura 2: Ejemplo del operador de cruzamiento SSOF entre dos individuos.

IV. Conclusión

Se describió el uso de un algoritmo genético para la selección del subconjunto óptimo de características. Con el algoritmo genético propuesto, se puede explorar de manera eficiente el espacio de características posibles, seleccionando subconjuntos de longitud variable. Por otra parte, los resultados muestran que el clasificador (que se construyó con el subconjunto óptimo de caracteristicas), permite identificar péptidos activos con una considerable eficiencia (92.47 % para pruebas), por lo que se puede utilizar el clasificador para analizar una gran cantidad de secuencias de péptidos con actividad desconocida de manera rápida y confiable. En lo que respecta al tiempo de ejecución de GAFS este es costoso al ser un algoritmo clasificado dentro de los métodos de envolutra (*wrapper*). Como trabajo futuro, se pueden búscar técnicas que logren disminuir el tiempo de ejecución de GAFS. En la misma dirección, generar un conjunto de secuencias aleatorias con longitudes similares a las secuencias usadas como control positivo para evaluar la especificidad del algoritmo.

Tabla II: Resultado promedio de las mejores soluciones en términos de la función de aptitud del algoritmo GAFS durante 30 repeticiones.

Nro. total de características	770
Nro. promedio de características	287.36
Aptitud promedio	91.58
Desv. Std. aptitud promedio	0.84

Tabla III: Los resultados muestran qué tan bien el predictor SVM separa los AMPs de los no AMPs para los conjuntos de prueba y validación.

Método	**Conjunto**	**Sens**	**Espec**	**ACC**
SVM AMP_B	Prueba	0.905	0.943	92.47
Todas las características	Prueba	0.875	0.925	90.05

Agradecimientos

Este trabajo fue apoyado por el Consejo Nacional de Ciencia y Tecnología bajo el proyecto SEP-CONACYT-CB-2010-154737.

Referencias

[1] W. Aoki y M. Ueda, "Characterization of antimicrobial peptides toward the development of novel antibiotics," *Pharmaceuticals*, vol. 6, no. 8, pp. 1055–1081, 2013. [Online]. Available: http://www.mdpi.com/1424-8247/6/8/1055

[2] R. Apweiler, A. Bairoch, C. H. Wu, W. C. Barker, B. Boeckmann, S. Ferro, E. Gasteiger, H. Huang, R. Lopez, M. Magrane, M. J. Martin, D. A. Natale, C. O'Donovan, N. Redaschi, y L.-S. L. Yeh, "Uniprot: the universal protein knowledgebase." *Nucleic Acids Research*, vol. 32, no. Database-Issue, pp. 115–119, 2004.

[3] C.-C. Chang y C.-J. Lin, "Libsvm: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 27:1–27:27, May 2011.

[4] I. Dondoshansky, *Blastclust (NCBI software development toolkit)*, 6th ed., 2002.

[5] C. Emmanouilidis, A. Hunter, y J. MacIntyre, "A multiobjective evolutionary setting for feature selection and a commonality-based crossover operator," in *Evolutionary Computation, 2000. Proceedings of the 2000 Congress on*, vol. 1. IEEE, 2000, pp. 309–316.

[6] F. C. Fernandes, D. J. Ridgen, y O. L. Franco, "Prediction of antimicrobial peptides based on the adaptive neuro-fuzzy inference system application," *Biopolymers*, vol. 98, pp. 280–287, 2012.

[7] C. D. Fjell, J. A. Hiss, R. E. W. Hancock, y G. Schneider, "Designing antimicrobial peptides: form follows function," *Nat Rev Drug Discov*, vol. 11, pp. 37–51, 01 2012.

[8] M. Goodarzi, B. Dejaegher, y Y. V. Heyden, "Feature selection methods in qsar studies," *Journal of AOAC International*, vol. 95, no. 3, pp. 636–651, 2012.

[9] I. Guyon y A. Elisseeff, "An introduction to variable and feature selection," *The Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.

[10] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, y I. H. Witten, "The weka data mining software: An update," *SIGKDD Explor. Newsl.*, vol. 11, no. 1, pp. 10–18, Nov. 2009.

[11] A. M. Helguera, R. D. Combes, M. P. González, y M. N. D. S. Cordeiro, "Applications of 2d descriptors in drug design: a dragon tale." *Curr Top Med Chem*, vol. 8, no. 18, pp. 1628–55, 2008. [Online]. Available: http://www.biomedsearch.com/nih/Applications-2D-descriptors-in-drug/19075771.html

[12] H. Jenssen, P. Hamill, y R. E. W. Hancock, "Peptide antimicrobial agents," *Clinical Microbiology Reviews*, vol. 19, no. 3, pp. 491–511, 2006.

[13] F. H. Waghu, L. Gopi, R. S. Barai, P. Ramteke, B. Nizami, y S. Idicula-Thomas, "Camp: Collection of sequences and structures of antimicrobial peptides," *Nucleic Acids Research*, vol. 42, no. D1, pp. D1154–D1158, 2014.

[14] C. W. Yap, "Padel-descriptor: An open source software to calculate molecular descriptors and fingerprints," *Journal of computational chemistry*, vol. 32, no. 7, pp. 1466–1474, 2011.