Detección de Conjuntos Significativos de Genes via Silohuette

Abras G.[1], Pastore J. I.[1,2], Brun M.[1] y Ballarin V.[1]

[1] Laboratorio de Procesos y Medición de Señales, Departamento de Electrónica
Universidad Nacional de Mar del Plata, Buenos Aires, Argentina.
[2] Consejo Nacional de Investigaciones Científicas y Técnicas. CONICET.

Resumen

La selección de genes es una tarea importante en el área de la bioinformática, donde los genes significativos son agrupados utilizando algún criterio de significación. En el caso de clasificación de tejidos, como por ejemplo, tejido enfermo vs normal, etc., el criterio utilizado es la capacidad de proporcionar características apropiadas para la tarea de clasificación. En otros casos es interesante seleccionar grupos grandes de los genes con un comportamiento similar, independientemente de la clase. Esta tarea se realiza generalmente por algoritmo de agrupamiento, donde toda la familia de genes, o un subconjunto de ellos, se agrupan en grupos significativos. Estas técnicas proporcionan una visión sobre la posible co-regulación entre los genes, pero por lo general ofrecen grandes, tal vez enormes conjuntos, en función del número de clusters necesarios. En este trabajo se presenta un nuevo algoritmo que establece una lista de genes con una expresión muy similar. Esto es posible mediante el agrupamiento de árboles completos proporcionados por el algoritmo de agrupamiento jerárquico, y el índice Silueta para la clasificación de los subconjuntos..

Palabras Clave
Selección de genes, Clustering Jerárquico, Índice Silhouette

Introducción

La selección de genes es una tarea importante en el área de la bioinformática, donde los genes significativos son agrupados utilizando algún criterio de similaridad de sus expresiones en perfiles obtenidos a través de sistemas de microarreglos, o similares. Por ejemplo, en el caso de clasificación de tejidos, donde uno puede diferenciar tejido enfermo contra tejido normal, o donde se desea discriminar entre tejido de diferentes órganos, el criterio utilizado, para determinar cuales genes son significativos, es la capacidad de proporcionar características apropiadas, en su expresión genética, para la tarea de clasificación.

En otros casos, donde la tarea principal no es la de clasificación supervisada, es interesante seleccionar grupos grandes de genes con similar comportamiento, independientemente de la clase. Esta tarea se realiza generalmente por algoritmos de agrupamiento, o *clustering*, donde toda la familia de genes, o un subconjunto de ellos, se agrupan en grupos significativos, a partir de sus valores de expresión en un conjunto de experimentos. Estas técnicas proporcionan una visión sobre la posible co-regulación entre los genes, bajo la hipótesis de que co-regulación implica co-expresión, por lo que co-expresión puede indicar la existencia de co-regulación. Estas hipótesis de co-regulación deben ser finalmente corroboradas o rechazadas mediante experimentos adicionales.

Usualmente los algoritmos de clustering resultan en enormes conjuntos, en función del número de clusters solicitados o determinados por el algoritmo. El gran tamaño de estos conjuntos hace poco probable su uso como generadores de hipótesis útiles. Por otro lado, una búsqueda exhaustiva de conjuntos significativos puede ser impráctica si la cantidad de genes disponibles para análisis es muy grande, lo cual es usual en este campo.

En este trabajo presentamos un nuevo algoritmo, que combina las particularidades del algoritmo de clustering jerárquico con el índice de validación Silohuette para generar listas de conjuntos, evitando la búsqueda exhaustiva, pero proporcionando resultados de alta calidad, comparado con la búsqueda exhaustiva. En las próximas secciones presentamos a) una introducción a las herramientas de reconocimiento de patrones utilizados, b) el algoritmo propuesto, junto con detalles de su implementación y c) los resultados sobre datos genéticos experimentales. Las conclusiones finales muestran que el algoritmo puede resultar en una herramienta útil para los investigadores como técnica preliminar de análisis de datos.

Elementos del Trabajo y Metodología

Reconocimiento de Patrones

Las técnicas de Reconocimiento de Patrones han sido ampliamente usadas para identificar objetos de acuerdo a distintos tipos de características [1]. Existen dos tipos básicos de clasificación de patrones, la clasificación supervisada y la no supervisada. En la supervisada se utilizan muestras previamente clasificadas para diseñar el clasificador, el cual será utilizado cuando aparezca un nuevo patrón desconocido. En la clasificación no supervisada se utilizan muestras no clasificadas y son estas mismas las que se desean clasificar. El nivel de información que se tiene es mucho menor que en la clasificación supervisada.

El Clustering es una técnica no supervisada de clasificación de objetos, y se basa en un conjunto de diferentes métodos determinísticos cuyo objetivo es formar grupos (clusters) de puntos en el espacio de las mediciones [2]. Estos clusters se forman de acuerdo a una medida de similitud, la cual usualmente se define como la proximidad de los puntos de acuerdo a una función distancia determinada. Una de las funciones distancia mas utilizadas es la Euclídea:

$$d(x,z) = \|x - z\| = \sqrt{(x-z)^t (x-z)} \qquad (1)$$

donde x y z son dos vectores. Cuanto menor es la distancia, mayor es la similitud.

Cuando se desea realizar una clasificación de un conjunto de objetos, primero se debe medir una cantidad m de características observables de la muestra. Estas m mediciones conforman un vector de características en el espacio m-dimensional. El total de objetos a clasificar forman un conjunto Ω . Sea W = { W1 , W2 , ... , Wk } una partición de □ . Cada subconjunto W1 , W2 , ... , Wk de Ω se llama clase . Para asignar los objetos a alguna de las clases se define una partición H = { H1 , H2 , ... , Hk } sobre Rm y una función D, que asigna a cada vector X de Rm la clase D(X) = Hi a la que X pertenece, que refleje en la mejor forma posible la partición W. Esto quiere decir que si la observación X(w) de un nuevo objeto w se encuentra en la región Hi de Rm , se puede afirmar con cierto grado de error, que el objeto w pertenece a la clase Wi . Si la partición H de Rm se correspondiera en forma biunívoca con la partición W de Ω se podría clasificar con absoluta certeza los objetos w de W con el conocimiento de X. Esto no ocurre dado que existen errores en las observaciones, o la característica que los diferencia no ha sido

observada o por la simplificación de las funciones de decisión que delimitan las regiones Hi. En este trabajo, los objetos a clasificar son genes.

Los datos están formados por un conjunto de m muestras S1, S2,…, Sm y n genes g1, g2, …, gn que normalmente están representados por una matriz de 2 dimensiones M donde M(i,j) representa la expresión del gen gi para la muestra Sj. Cada gen gi corresponde a una fila de la matriz M, y es representado por el vector de característicos Xi (xi1, xi2, …, xim), donde cada valor xij representa la expresión del gen gi para la muestra Sj. Cada gen debe ser asignado a una de K posibles clases. El resultado de la clasificación es un conjunto W = { W1 , W2 , ... , Wk }, donde cada Wi es un grupo de genes con cierto grado de similitud de acuerdo a determinado patrón dentro de las células. El vector de características está compuesto entonces por m muestras, por lo que las regiones de la partición H están en el espacio Rm .

Clustering jerárquico

El Hierarchical Clustering es uno de algoritmos de clustering más utilizados en bioinformática [3]. En este tipo de algoritmos, no es necesario conocer previamente el número de clases en que el conjunto total de elementos puede llegar a dividirse de acuerdo a sus características

Se inicia asumiendo que hay tantas clases como muestras o vectores se tengan, es decir " n " . En cada iteración se unen los dos clusters más cercanos entre sí (basado en alguna medida de distancia) formando uno nuevo. Esto reduce el número de clusters en uno, en cada iteración. Este procedimiento se repite hasta que finalmente todos los elementos se unen para formar un solo grupo. Si el número de clases se conoce previamente, el algoritmo finaliza cuando se llega al número K de clases buscado. Si el número de clases no se conoce previamente, el algoritmo puede ser detenido en algún nivel arbitrario, considerando algún índice de similitud o de calidad de los agrupamientos formados.

La forma escalonada en que los clusters se van agrupando puede ser visualizada en un diagrama en árbol denominado dendograma, donde se muestran las sucesivas uniones entre grupos a medida que aumenta la distancia entre los mismos. Si es posible medir la similitud entre los clusters, entonces el dendograma es, por lo general, realizado con una escala que muestre el grado de similitud entre los clusters que han sido unidos.

La Fig. 1 muestra un dendograma para un problema hipotético mostrando los datos en formato heatmap, y el dendograma a la derecha.

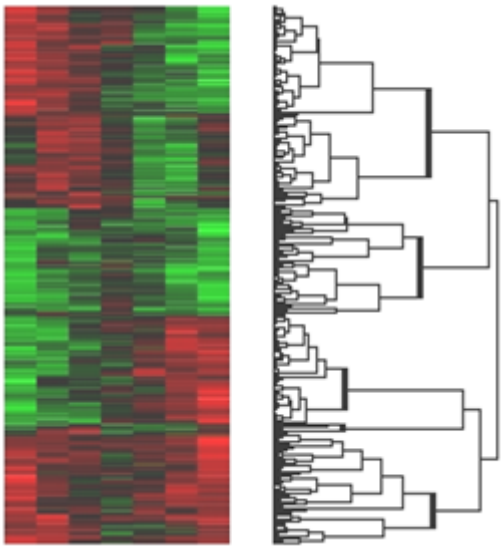


Figura 1. Dendograma

Los pasos principales del hierarchical clustering son:

1. Se inicializa el algoritmo con P=n clusters definidos por Ci = {Xi}, con i = 1,2,…,n.
2. Si P = K, terminar ya que se ha logrado el número deseado de clusters K.
3. Encontrar el par de clusters más cercanos, por ejemplo Ci y Cj.
4. Unir Ci y Cj.en un solo cluster Cs = {Ci,Cj}, disminuir P en uno.
5. Volver a 2.

Cuando los clusters son simples, es decir que están formados por un solo elemento cada uno, para determinar cuales son los clusters más próximos la distancia entre ellos se reduce a la distancia entre los elementos que los forman. Considerando Ci como el cluster i-ésimo, y X una muestra o vector perteneciente a ese cluster, se tiene:

$$d(C_i, C_j) = \min_{X \in C_i, X' \in C_j} \left\| X - X' \right\| \qquad (2)$$

Cuando los clusters están compuestos por varios elementos, para determinar cuáles son los clusters más cercanos, se pueden considerar diferentes medidas:

$$d_{\min}(C_i, C_j) = \min_{X \in C_i, X' \in C_j} \left\| X - X' \right\| \qquad (3)$$

$$d_{\max}(C_i, C_j) = \max_{X \in C_i, X' \in C_j} \left\| X - X' \right\| \qquad (4)$$

$$d_{avg}(C_i, C_j) = \frac{1}{n_i n_j} \sum_{X \in C_i} \sum_{X' \in C_j} \left\| X - X' \right\| \qquad (5)$$

$$d_{mean}(C_i, C_j) = \left\| m_i - m_j \right\| \quad \text{, siendo } m_i \text{ el vector promedio del cluster } C_i \qquad (6)$$

Si los clusters forman grupos bien compactos y están lo suficientemente separados, todas las medidas anteriores producen el mismo resultado, que es el esperado.
Si los grupos de puntos no forman aproximadamente hiperesferas, adoptando formas arbitrarias poco separadas, o alargadas, o con puntos que actúan como puentes entre ellos, los resultados dependen de la medida de similitud adoptada. Por ejemplo, la distancia mínima de la ec. (3) es apropiada para unir grupos que tienen forma alargada y paralela, pero no son apropiados cuando dos grupos tienen elementos entre ellos que actúan como puentes. La distancia máxima de la ec. (4) es exactamente lo opuesto. Las ecuaciones (5) y (6) tienen comportamientos intermedios entre las otras dos medidas de similitud.

Índice Silhouette

Un problema común de los algoritmos de clustering es la validación de los resultados. Hay básicamente dos tipos de validación. La *validación interna*, la cual se basa en cálculos realizados sobre los clusters resultantes como por ejemplo la separación entre ellos, su redondez o que tan compactos son. Este tipo de validación no requiere información adicional a la ya obtenida. La otra es la *comparación de particiones*, que puede ser la comparación de las particiones obtenidas con otras que genera el mismo algoritmo con parámetros diferentes (*validación relativa)*, o con la partición real de los datos originales (*validación externa*) [4].

Uno de los métodos de validación interna utilizados es el cálculo del *índice Silhouette*, el cual permite determinar la calidad de los resultados de un clustering [5]. Refleja qué tan separados están dos clusters y qué tan compactos son. Su valor varía desde -1 a 1, y cuanto más alto es, mejor es el resultado del clustering obtenido.

Sea $X \in C_K$ un dato perteneciente al cluster k-ésimo, n_k la cantidad de elementos de C_k, entonces:

$$a(X) = \frac{1}{n_k - 1} \sum_{Y \in C_k, Y \neq X} d(X,Y) \tag{7}$$

es la distancia promedio de X a puntos de C_k

$$b(X) = \min_{h=1,\ldots,k} \left[\frac{1}{n_h} \sum_{Y \in C_h} d(X,Y) \right] \tag{8}$$

es la distancia promedio al cluster más cercano

Se define el índice Silhouette de X como:

$$S(X) = \frac{b(X) - a(X)}{\max(a(X), b(X))} \tag{9}$$

Se puede obtener un índice Silhouette para un cluster:

$$S(C_h) = \frac{1}{n_k} \sum_{X \in C_k} S(X) \tag{10}$$

También se puede obtener un índice Silhouette promedio para todo el clustering:

$$S = \frac{1}{k}\sum_{h=1}^{k}\left[\frac{1}{n_h}\sum_{X\in C_h} S(X)\right] \qquad (11)$$

Se puede observar que si b(X)>>a(X) , S(X) para cada X en C_k es muy cercano a 1. Esto significa que C_k es compacto y está bien alejado de los otros clusters. Si b(X) ≈ a(X), S(X) para cada X en C_k es muy cercano a 0. Esto significa que C_k es poco compacto y está poco separado de los otros clusters.

Algoritmo Propuesto

El objetivo de este estudio es el de seleccionar subconjuntos de genes *altamente correlacionados*. En el espacio de perfiles de expresión de los genes (expresión a través de todas las muestras), provisto de la distancia Euclídea, donde cada perfil genético se representa con un punto de R^n, esto se corresponde con conjuntos compactos y separados de otros puntos.

Una manera posible de encontrar estos grupos compactos y separados sería calcular un índice de *compacidad* en todos los subconjuntos posibles de los N perfiles. Una medida adecuada para medir la compacidad puede ser el índice Silohuette, utilizado con éxito por los autores para otros objetivos [6,7]. En la figura 1 podemos ver 3 ejemplos de agrupamientos de diferente calidad. Los valores de Silhouette para esos agrupamientos, de izquierda a derecha, son 0.29, 0.43 y 0.69 respectivamente.

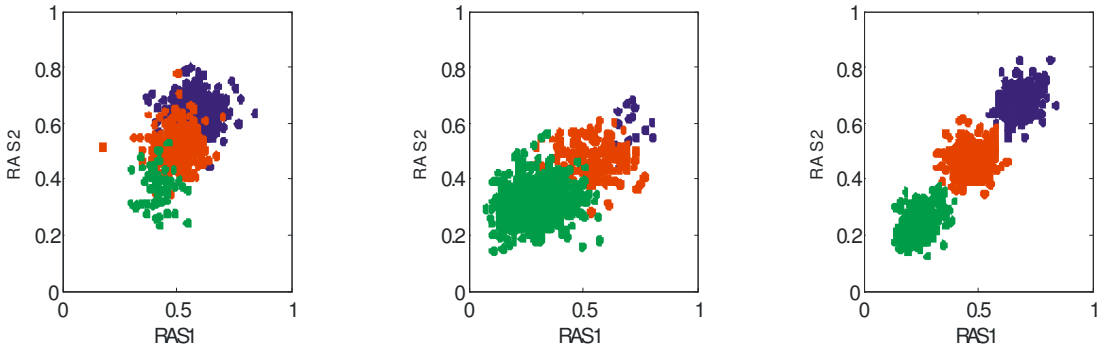


Figura 1. Ejemplo de diferentes agrupamientos.

Uno de los problemas de este enfoque es que la cantidad de subconjuntos crece en forma combinatoria. Por ejemplo para un conjunto de 2000 genes, la cantidad de subconjuntos a evaluar es de 2^{2000}. Por esta razón se requiere de alguna técnica que permita encontrar los mejores subconjuntos evitando la búsqueda exhaustiva. Para solucionar este problema, en este trabajo proponemos limitar la familia de subconjuntos donde se realiza la búsqueda, utilizando clustering jerárquico para generar una familia de subconjuntos, y luego evaluar el índice Silhouette solamente en esos subconjuntos. El algoritmo de clustering genera un total de 2*N subconjuntos a procesar.

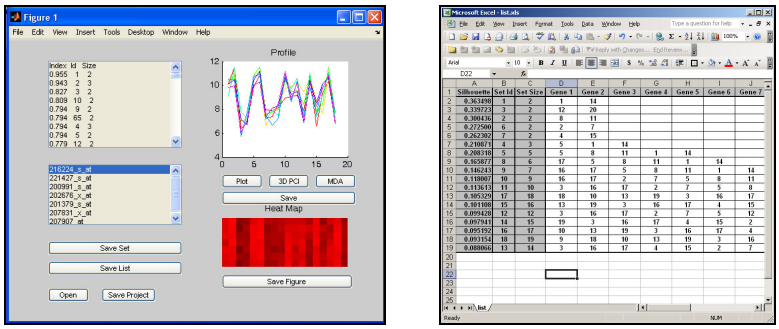
Los pasos principales del algoritmo propuesto son los siguientes:

- *Clustering jerárquico de los datos originales*. El algoritmo de clustering jerárquico es aplicado a los datos originales, el cual finaliza al obtener un único cluster que contiene todos los elementos de la muestra. De esta manera queda definido el dendograma que define las agrupaciones intermedias de los datos, las cuales son los subconjuntos a analizar.

- *Ordenamiento por índice Silhouette*. Se calcula el índice Silhouette a cada agrupación (cluster) obtenido a través del clustering. Se ordenan los conjuntos en función del índice, de mayor a menor.

Software

El algoritmo propuesto fue implementado en Matlab, utilizando una interfaz gráfica de usuario que, después de seleccionar y cargar gen expresión de la matriz, ejecuta el algoritmo de agrupamiento jerárquico, calcula el índice Silhouette de todos los subconjuntos, y proporciona una lista de los grupos resultantes, ordenados por índice. Los mejores grupos (mayor índice) se mostrarán en la parte superior. Al seleccionarse un grupo el programa muestra información detallada del grupo, incluidos los perfiles de expresión de los genes en el grupo, tanto como gráfica lineal y como imagen *heatmap*. Botones adicionales permiten guardar los resultados, ya sea la totalidad de las listas de juegos, o el contenido del grupo seleccionado. La figura 1 muestra la interfase del programa



(a) *(b)*

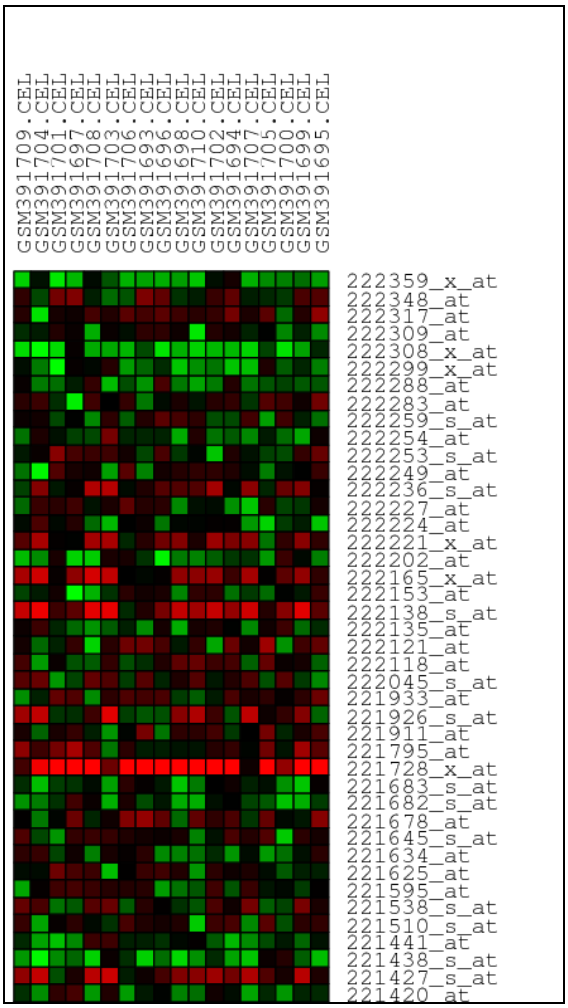
Figure 1. a) Program's interface for gene selection, b) Example of list of sets and their silhouette index

Resultados

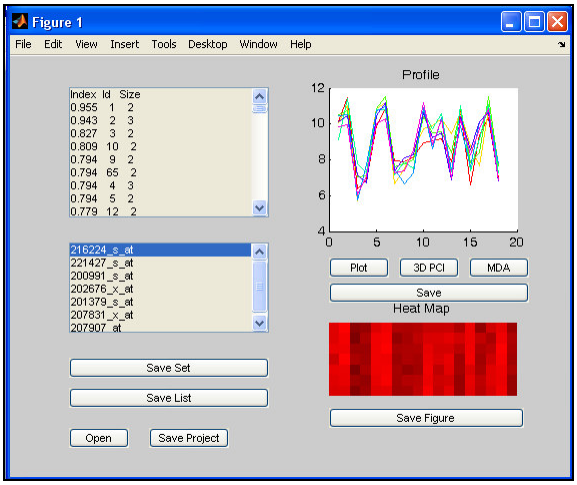
Diabetes

Como ejemplo de aplicación aplicamos el algoritmo a datos de microarray obtenidos de un estudio de diabetes, estudiando los perfiles de expresión (microarrays) para sujetos obesos y delgados [8]. Este estudio presenta perfiles de expresión genética para 18 sujetos, 13 con obesidad y 5 delgados, utilizando el chip U133A de Affymetrix. Figura 1a) muestra el

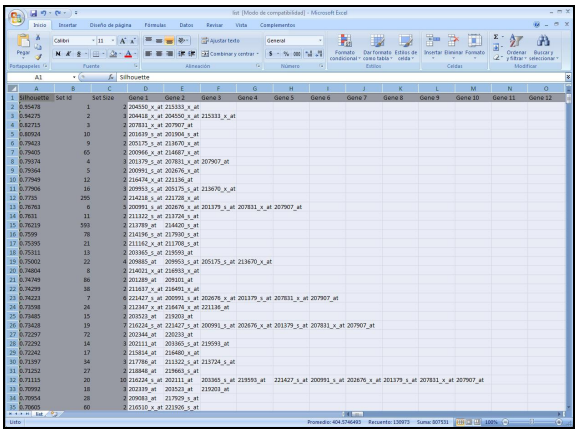
perfil de una parte de los genes, junto con una parte del clustering jerárquico luego de filtrar los genes para analizar solamente los 1000 genes de mayor variación en los datos.



(a)



(b)



(c)

Figura 1: (a) muestra del perfil genómico, (b) Resultado del algoritmo, (c) Planilla con los conjuntos ordenados de mayor a menor ranking (Silhouette).

En este análisis los conjuntos de mayor ranking fueron seleccionados para análisis, verificando si los genes encontrados en conjuntos compactos y separados tienen asignados funciones biológicas similares. La tabla 1 muestra algunos de los conjuntos compactos detectados.

Tabla 1: Conjuntos detectados con mayor índice Silohuette

Posición	Silohuette	Probes (Identificador Affy)
1	0.95478	204550_x_at, 215333_x_at
2	0.94275	204418_x_at, 204550_x_at, 215333_x_at
3	0.82715	207831_x_at, 207907_at

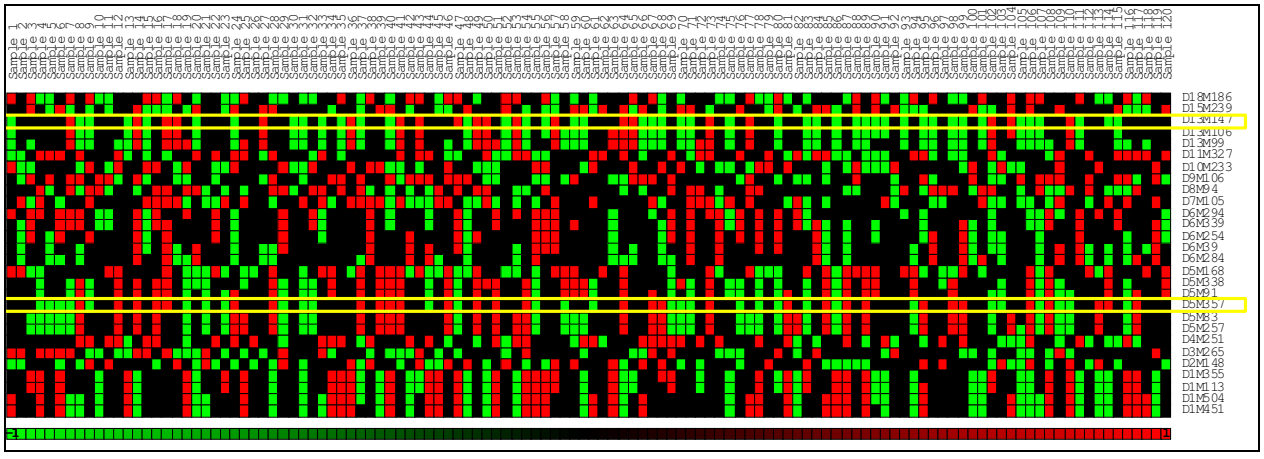
Los tres *probesets* encontrados en los primeros 2 conjuntos fueron analizados en las páginas de Affimetrix y NCBI. La tabla 2 muestra la información obtenida sobre esos 3 genes. Como podemos ver, dos probesets son detectores del mismo gen (GSTM1) y el tercero es detector de un gen de funciones similares (GSTM2). Análisis adicional de otros conjuntos muestra resultados similares en algunos casos también.

Tabla 2: Genes analizados

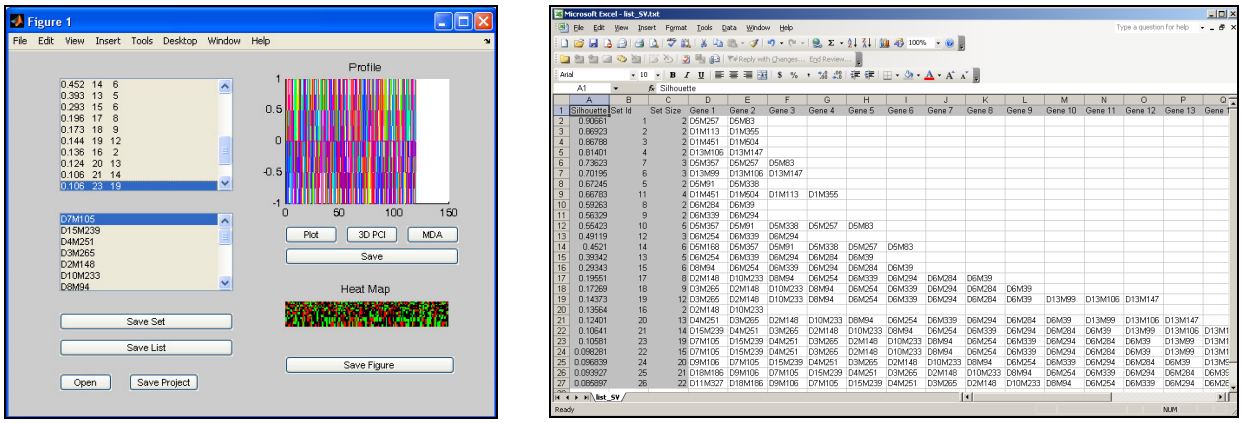
Probe Set ID	Gene Title	Gene Symbol	biological process term	molecular function term	cellular component term
204550_x_at	glutathione S-transferase mu 1	GSTM1	metabolic process	glutathione transferase activity transferase activity	cytoplasm
204418_x_at	glutathione S-transferase mu 2 (muscle)	GSTM2	metabolic process	glutathione transferase activity transferase activity	cytoplasm
215333_x_at	glutathione S-transferase mu 1	GSTM1	metabolic process	glutathione transferase activity transferase activity	cytoplasm

Listeria

Para otro ejemplo de aplicación utilizamos datos de susceptibilidad de ratones a listeria monocytogenes [9]. Este estudio consiste en el análisis de la relación entre QTLs (Qua*titative trait loci*, partes de un gen que afecta un carácter cuantitativo) y la supervivencia (mas de 240 horas) de 120 ratones después de infectarlos con listeria, consistiendo en 35 supervivientes y 85 no supervivientes, analizando un total de 133 QTLs para cada ratón. Después del filtrado de QTLs con datos incompletos, se obtuvieron 28 QTLs totalmente expresados en los 120 ratones (Figura 3a). Figura 3b) y 3b) muestran la interfase del programa y los resultados en una planilla de cálculo mostrando los resultados.



(a)



(b) (c)

Figura 3. a) Genotipos y Fenotipos de los 28 QTLs seleccionados sobre las 133 muestras. Los recuadros en amarillo muestran las dos regiones seleccionadas para análisis.

La tabla 3 muestra algunos de los conjuntos compactos detectados. Para verificar el funcionamiento del algoritmo, considerando el hecho de que no busca todas los posibles subconjuntos, realizamos una búsqueda exhaustiva, calculando el índice Silhouette sobre todos los subconjuntos de tamaño 2, 3, 4 y 5.

Tabla 3: Conjuntos detectados con mayor índice Silohuette

Posición	Silohuette	Probes (Identificador Affy)
1	0.90661	D5M257, D5M83
2	0.86923	D1M113, D1M355
3	0.86788	D1M451, D1M504
5	0.73623	D5M357, D5M257, D5M83
8	0.66783	D1M451, D1M504, D1M113, D1M355
9	0.55423	D5M357, D5M91, D5M338, D5M257, D5M83

Del análisis completo adicional, encontramos que para conjuntos de 2 probesets, los 5 primeros conjuntos (de mayor ranking) fueron identificados por el algoritmo. Para 3 probesets, los 2 mejores conjuntos fueron detectados por el algoritmo, mientras que para 4 y 5 probesets, el mejor conjunto fue correctamente identificado en cada caso.

Discusión

Se presentó un algoritmo para determinar grupos compactos y adecuadamente separados de acuerdo a las características usadas en la aplicación del hierarchical clustering algorithm, mediante el cálculo del índice Silohuette.

Conclusión

La herramienta propuesta puede ser una herramienta poderosa para los biólogos o investigadores de biología computacional interesados en la generación de nuevas hipótesis sobre los genes co-expresados, que no se proporcionan en más de herramientas de análisis estándar.

Agradecimientos

Marcel Brun recibió fondos de la Agencia Nacional de Promoción Científica y Tecnológica (PICT-2006-02313) para el desarrollo de este trabajo.

Referencias

[1] Duda R. O., Hart P. E., Stork David G., "Pattern Classification and Scene Analysis", 2002 , John Wiley and Sons

[2] Brun Marcel, Johnson Charles D., and Ramos Kenneth S., "Clustering: revealing intrinsic dependencies in microarray data," Genomic Signal Processing and Statistics, eds. E. R. Dougherty, I. Shmulevich, J. Chen, and Z. J. Wang, EURASIP Book Series on Signal Processing and Communication, Hindawi Publishing Corporation, 2005.

[3] Chiang D.Y. , Brown Patrick O. , Eisen Michael B., "Visualizing associations between genome sequences and gene expression data using genome-mean expression profile", 17 (1) , 2001

[4] Lori Dalton, Virginia Ballarin and Marcel Brun, "Clustering Algorithms: On Learning, Validation, Performance, and Applications to Genomics", Current Genomics, Volume 10, Number 6, September 2009 , pp. 430-445(16)

[5] Rousseeuw, Peter J., "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis", Journal of Computational and Applied Mathematics , 20 (1) , pp.53-65 , 1987.

[6] Pearson John V. et.al., "Identification of the Genetic Basis for Complex Disorders by Use of Pooling-Based Genomewide Single-NucleotideÿPolymorphism Association Studies", The American Journal of Human Genetics , 80 , pp.126-139 , 2007.

[7] Jianping Hua, David W. Craig, Marcel Brun, Jennifer Webster, Victoria Zismann, Waibhav Tembe, Keta Joshipura, Matthew J. Huentelman, Edward R. Dougherty, Dietrich A. Stephan: SNiPer-HD: improved genotype calling accuracy by an expectation- maximization algorithm for high-density SNP arrays. Bioinformatics 23(1): 57-63 (2007)

[8] Pihlajamäki J, Boes T, Kim EY, Dearie F et al. Thyroid hormone-related regulation of gene expression in human fatty liver. J Clin Endocrinol Metab 2009 Sep;94(9):3521-9. PMID:

[9] Victor L. Boyartchuk, Karl W. Broman, Rebecca E. Mosher, Sarah E.F. D'Orazio, Michael N., "Multigenic control of Listeria monocytogenes susceptibility in mice", Nature Genetics, Vol 27, pp. 259-260, 2001