UNIVERSIDAD DE SANTIAGO DE CHILE
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE INGENIERÍA INFORMÁTICA

ALGORITMO DE AGRUPAMIENTO DE CONCENSO BASADO EN GRAFOS PARA DATOS DE EXPRESIÓN GÉNICA

RICARDO FABIÁN ZÚÑIGA ANTIÑIRRE

SANTIAGO – CHILE
2015

© **Ricardo Fabián Zúñiga Antiñirre**

Se autoriza la reproducción parcial o total de esta obra, con fines académicos, por cualquier forma, medio o procedimiento, siempre y cuando se incluya la cita bibliográfica del documento.

RESUMEN

La búsqueda de información sobre grandes bases de datos, o el análisis de patrones similares de secuencias de aminoácidos en bioinformática, se han convertido en un problema de estudio, dado el enorme volúmen de información existente. Una alternativa para resolver este tipo de problema, consiste en generar herramientas para el análisis de datos y el uso de algoritmos para la interpretación de resultados con conclusiones biológicas de interés. Dentro de estos métodos, se encuentra la búsqueda de grupos o clustering de datos, que posee una fuerte presencia en la mayoría de las tareas del conocimiento ligado con esta área. Existe un nuevo enfoque a la agrupación clásica, llamado clustering de consenso. El presente proyecto estudia un método de consenso relacionado con el algoritmo MSTkNN,(Ponta, 2008), que pretende conciliar la información acerca de la agrupación de los datos establecidos procedentes de diferentes fuentes de datos o diferentes ejecuciones del algoritmo.

ÍNDICE DE CONTENIDOS

Índice de Figuras **vi**

Índice de Tablas **vii**

Índice de Algoritmos **viii**

1. Introducción **1**

1.1. Antecedentes y motivación 1

1.2. Descripción del problema 2

1.3. Solución propuesta 2

1.3.1. Características de la solución 2

1.3.2. Propósito de la solución 2

1.4. Objetivos y alcance del proyecto 3

1.4.1. Objetivo general 3

1.4.2. Objetivos específicos 3

1.4.3. Alcances 3

1.5. Metodología y herramientas utilizadas 4

1.5.1. Metodología 4

1.5.2. Herramientas de desarrollo 5

1.6. Organización del documento 5

2. Marco Teórico **7**

2.1. Agrupación de datos de expresión génica 7

2.2. Algoritmos de clustering 8

2.2.1. Algoritmos de agrupación jerárquica 10

2.2.2. Algoritmo de clustering MSTkNN 11
2.3. Validez del clustering 12
2.3.1. Análisis de correlación 12

3. Análisis estadístico **14**
3.1. Metodología de consenso utilizando algoritmo MSTkNN 14
3.1.1. Valores fuera de rango *outliers estadísticos* 14
3.1.2. Cálculo del percentiles 15
3.1.3. Identificación de valores atípicos 16
3.1.4. Determinación de máximos y mínimos 16
3.1.5. Valor atípico leve 16
3.1.6. Valor atípico extremo 16

4. Algoritmo de Consenso **18**
4.1. Modificación del algoritmo MSTkNN para establecimiento de consenso 18
4.2. Resultados Experimentales 19

5. Conclusiones **26**

Referencias **27**

Apéndices **27**

ÍNDICE DE FIGURAS

4.1. Histograma que muestra la relación promedio entre el índice promedio de *Jaccard* y el conjunto de particiones. 21
4.2. Histograma que muestra la distribución promedio ascendente entre índice de promedio de *Jaccard* y el conjunto de particiones 22
4.3. *Histograma de consenso que muetra la densidad de frecuencia y convergencia al índice promedio* 24

ÍNDICE DE TABLAS

4.1. Tabla de resultados histograma (4.2) 23
4.2. Tabla de resultados histograma (4.2) 24
4.3. Tabla de resultados componentes del cluster 1 al cluster 16 sobre la particion P_{68} . . . 25

ÍNDICE DE ALGORITMOS

2.1. algoritmo de clustering MSTkNN 12

4.1. algoritmo de clustering MSTkNN utilizando medidas consenso 18

CAPÍTULO 1. INTRODUCCIÓN

1.1 ANTECEDENTES Y MOTIVACIÓN

Bioinformática es un área de investigación interdisciplinaria, que nace en base a la necesidad de los biólogos de utilizar nuevas tecnologías para el almacenamiento e interpretación de grandes volúmenes de información que guardan relación con biología molecular (Cohen, 2004).

Sus principales enfoques, se encuentran orientados a la organización y clasificación de la información, generación de herramientas para el análisis de datos y el uso de algoritmos para la interpretación de resultados con conclusiones biológicas de interés. En base estos enfoques, las técnicas de agrupación han sido de gran aporte para comprender de mejor forma, funciones génicas y de regulación de procesos celulares. Genes con patrones similares de expresión, pueden ser agrupados en una misma categoría, lo que puede inferir a que estén involucrados en los mismos procesos celulares y existir una fuerte correlación entre ellos.(Cohen, 2004), de acuerdo a lo anterior, se hace necesario la creación de métodos fiables y eficientes para el estudio de patrones similares de expresión.

Dentro de estos métodos, se encuentra la búsqueda de grupos o clustering de datos, que posee una fuerte presencia en la mayoría de las tareas del conocimiento ligado con esta área. Los algoritmos de clustering, se caracterizan por tener un buen rendimiento y calidad en los resultados, que se ha potenciado con la inclusión de nuevas técnicas de revalidación y evaluación de estabilidad de los grupos descubiertos a través de los algoritmos de consenso.(Stefano Monti, 2003).

Este nuevo enfoque escala como una solución importante a la agrupación clásica. El consenso surge en la medida en que se han obtenido una serie de diferentes agrupaciones de entrada para un determinado conjunto de datos y se desea obtener un solo consenso de agrupación general sobre los agrupamientos ya existentes. Luego, esta solución pretende conciliar la información acerca de la agrupación de los datos establecidos procedentes de diferentes fuentes de datos o diferentes ejecuciones del algoritmo.

1.2 DESCRIPCIÓN DEL PROBLEMA

Dado un conjunto de datos de expresiones génicas se busca encontrar grupos de genes con patrones similares de expresión, utilizando un algoritmo de agrupamiento de consenso basado en grafos. Con el agrupamiento de consenso, se pretende aumentar el rendimiento y calidad en los resultados a través de la inclusión de nuevas técnicas de revalidación y evaluación de estabilidad de los grupos.

1.3 SOLUCIÓN PROPUESTA

1.3.1 Características de la solución

Se contempla el diseño un algoritmo de consenso basado en grafos, que utilizará las propiedades de los grafos: *Minimum Spanning Tree* (Árbol de Cobertura Mínima), y *k-Nearest Neighbors* (k vecinos más cercanos). El algoritmo recibirá como parámetro de entrada una matriz de distancias entre objetos, y retornará un grafo de componentes conexas. El algoritmo buscará el consenso a través de múltiples iteraciones y evaluaciones de estabilidad de los grupos descubiertos. El algoritmo se programará en forma secuencial utilizando recursividad en algunos ámbitos del desarrollo.

1.3.2 Propósito de la solución

En biología, es importante encontrar grupos de genes que tengan patrones similares de expresión, puesto que es necesario determinar y regular ciertos procesos celulares para curar y prevenir enfermedades. Desde esta perspectiva, las expresiones génicas cumplen un papel fundamental, puesto que es el proceso por medio del cual las células transforman la información codificada por los ácidos nucleicos en proteínas necesarias para su desarrollo y funcionamiento. Luego, el propósito de la solución se encuentra orientado bajo dos enfoques, el primero es apoyar el estudio y análisis de expresiones génicas utilizando algoritmos de clustering basado en grafos, el objetivo

es encontrar nuevas estructuras que entreguen de manera eficiente grupos de genes relacionados de acuerdo a los patrones de entrada establecidos, y el segundo enfoque está dirigido en proporcionar a los biólogos una herramienta que entregue de manera confiable estos grupos de genes de acuerdo a su expresión génica para la determinación de los procesos celulares.

1.4 OBJETIVOS Y ALCANCE DEL PROYECTO

1.4.1 Objetivo general

Crear un algoritmo de agrupamiento de consenso basado en grafos para datos de expresión génica.

1.4.2 Objetivos específicos

1. Diseñar e implementar el algoritmo de agrupamiento de consenso utilizando características de los grafos de proximidad *Minimum Spanning Tree* y *k-Nearest Neighbors*.
2. Estimar la calidad de la solución con datos de expresión génica a través de diferentes agrupaciones de entrada.
3. Evaluar la complejidad del algoritmo de forma experimental.

1.4.3 Alcances

En relación a lo establecido anteriormente, se tiene en cuenta los siguientes alcances y limitaciones:

1. La solución se encuentra sujeta a la búsqueda de grupos relacionados sobre datos de expresión génica, por lo tanto, la calidad de la solución se evaluará sobre estos datos de expresión.

2. Para estudiar la robustez y complejidad de algoritmo, se utilizarán conjuntos de datos de diferente naturaleza, con diferentes medidas de distancia.

1.5 METODOLOGÍA Y HERRAMIENTAS UTILIZADAS

1.5.1 Metodología

La metodología a utilizar será el método científico. El proyecto contempla el diseño del algoritmo a través de una sucesión de etapas y procedimientos organizados para el ciclo entero de la investigación. Al finalizar cada etapa se realizará una revisión final, con el objetivo de saber si el proyecto está listo para avanzar hacia la siguiente fase, con esto se pretende asegurar la calidad del producto final. La metodología incluirá las siguientes etapas:

1. Análisis del problema
2. Diseño del algoritmo
3. Codificación
4. Compilación y ejecución
5. Validación de resultados
6. Depuración
7. Documentación

En relación al tipo de programación, se utilizará programación estructurada puesto que las características del tipo de desarrollo cumplen esta condición. De acuerdo a lo anterior, se establece atacar el problema utilizando:

1. Recursos abstractos

   Descomponer el problema en otro más simple

2. Diseño descendente (Top-Down)

   El programa se descompone en estructuras jerárquicas

3. Estructuras de control

   Estructuras secuenciales, selectivas y repetitivas

1.5.2 Herramientas de desarrollo

Las herramientas a utilizar (hardware y software) para el desarrollo del proyecto son las siguientes:

1. Estación de trabajo

   Sistema operativo: Linux Ubuntu 10.04 (64 bits)

2. Documentación

   LaTeX

3. Desarrollo

   Lenguaje de programación C

   GNU Compiler Collection (GCC)

   Editor VIM

4. Control de versiones

   SVN

1.6 ORGANIZACIÓN DEL DOCUMENTO

El presente trabajo está dividido en ocho capítulos considerando éste como el primero. En el Capítulo 2 se formalizan los fundamentos de documento XML, modelo de árbol XML, y lenguaje de definición de expresiones de camino para definir claves XML. ...

El presente proyecto, se encuentra dividido en 5 capítulos. En el Capítulo 1 se formalizan los fundamentos de la teoría sobre diferentes métodos de agrupación, se describe el problema en cuestión y tambiĺa solución propuesta. En el Capítulo 2, se analiza con más detalle la teoría de sobre diferentes algoritmos de *clustering* y métodos de correlación existentes para validar diferentes particiones de grafos.En el Capítulo 3, se analizará la metodología estadísitca que se utilizará para la determinación del concenso entre las diferentes particiones. En el Capítulo 4 se desarrollará el algoritmo de consenso en base a la versión MSTkNN de (Ponta, 2008),y en el Capítulo 5, se analizarán conclusiones.

CAPÍTULO 2. MARCO TEÓRICO

El presente capítulo tiene como propósito intoducir formalmente los conceptos que hacen referencia a agrupación de datos de expresión génica, algoritmos de clustering y análisis de correlaciones externas.

2.1 AGRUPACIÓN DE DATOS DE EXPRESIÓN GÉNICA

Los algoritmos de clustering, pueden ser clasificados de acuerdo a la manera en que se agrupan los datos: de partición o jerárquicos. El primero produce una única partición (o muchas en caso de un método de clustering difuso), mientras que el último produce una jerarquía de particiones. Dentro de los métodos de partición, los algoritmos de clustering basados en grafos han presentado un buen rendimiento en comparación con otros (Ponta, 2008).

Entre los algoritmos de clustering más comunes para agrupar genes, se tiene Self Organizing Maps y K-means, los cuales consideran atributos tales como la estructura, función y localización sub-celular de cada producto genético.

Existen además, para el enfoque de agrupamiento, algoritmos de clustering basados en grafos como MST-kNN, el cual utiliza dos grafos de proximidad (Minimum Spanning Tree y k Nearest Neighbors), este recibe como parámetro de entrada una matriz de distancias entre objetos, y retorna un grafo de componentes conexas. (Ponta, 2008)

Para considerar, un árbol de cobertura mínima (Minimum Spanning Tree) de un grafo conexo y no dirigido, es un subgrafo que en sí a su vez es un árbol que contiene todos los vértices del grafo inicial. Cada arista tiene asignado un peso proporcional entre ellos, que es un número representativo asociado a la distancia, que se usa para asignar un peso total al árbol de cobertura mínima computando la suma de todos los pesos de las aristas del árbol en cuestión.

Un árbol de cobertura mínima es un árbol recubridor que pesa menos o igual que otros árboles recubridores.

Luego, el algoritmo k-Nearest Neighbors, parte de un grafo representado mediante una matriz, y obtiene para cada nodo, el k vecino más cercano para éste, finalmente retorna el grafo originado del grafo padre.

Además de las técnicas de agrupación tradicionales, existe una nueva metodología basada agrupamiento de consensos (consensus clustering), cuyo método se basa en un enfoque de análisis más exacto para guiar y ayudar a la amplia gama de algoritmos de clustering existentes. Está basado en una técnica de revalidación, donde se proporciona un método para representar un consenso a través de múltiples ejecuciones de un algoritmo de clustering, y para la evaluación de la estabilidad de los grupos descubiertos. El método también puede ser usado para representar el consenso entre diferentes series de un algoritmo de agrupamiento con reinicio aleatorio (tales como K-means, la agrupación basada en modelos Bayesianos, entre otros), así como para tener en cuenta su sensibilidad en las condiciones iniciales de la ejecución, (Stefano Monti, 2003),(Laderas T.., 2007).

2.2 ALGORITMOS DE CLUSTERING

Dado un conjunto de objetos, el problema agrupación consiste en encontrar grupos de objetos similares, tal que se encuentren en el mismo grupo o bien objetos disímiles en diferentes grupos.

Sea $N = \{e_1, e_2, e_3, ...e_n\}$, un conjunto de objetos, donde cada objeto e_i, es descrito como un vector de m valores diferentes, luego, un algoritmo de agrupamiento producirá una partición $C(N) = \{c_1, c_2, c_3..., c_c\}$, de N en c grupos diferentes.

Los algoritmos de agrupamiento, se pueden clasificar según diferentes tipos parámetros, estas pueden ser de acuerdo a técnicas de agrupamiento o de partición. El primero produce un conjunto anidado de particiones, mientras que el segundo produce una sola partición. (Jain & Dubes, 1998)

Los objetos se describen en base a conjunto de características, que puede ser cualitativas o cuantitativas. La represensentación y el uso de estas características son de ayuda en la definición de

la medida de similitud en el algoritmo a utilizar. Una contribución importante dentro de este proceso, es la taxonomía de la agrupación. La taxonomía puede ser utilizada como base para comprender los diferentes tipos de algoritmos de agrupación, como también para clasificar otros.

Se presenta el proceso de agrupación como una actividad de cinco pasos,(Jain & Dubes, 1998):

1. Representación de patrones Definición de medida de similitud
2. Agrupación
3. Abstracción de datos
4. Evaluación de la salida

A partir de lo anterior, se identifican dos características:

1. La representación de los objetos
2. La distancia o similitud utiliza

Luego, de forma complementaria, en cada una de las categorías de la taxonomía, también de define la aplicación de los algoritmos de clustering en cuatro áreas,(**?**):

1. La segmentación de imágenes
2. Reconocimiento de objetos y carácteres
3. Recuperación de documentos
4. Minería de datos

Un estudio más específico de algoritmos de agrupamiento en bioinformática fue presentado por (Jiang & Zhang, 2014)

Se identifican tres tipos diferentes de problemas para un conjunto de datos de expresión génica:

1. La agrupación de genes,
2. Las muestras de agrupación
3. Agrupación en el sub-espacio.

2.2.1 Algoritmos de agrupación jerárquica

El algoritmo de agrupación jerárquica (*Hierarchical clustering algorithms*), produce un conjunto anidado de particiones basado en la medida de distancia entre los objetos. Esta clase de algoritmos se clasifican como tipo aglomerativo, ya que comienzan con cada objeto en un solo grupo, y el algoritmo será el encargado de fusionar los diferentes grupos de hasta una condición se cumpla, o todos los objetos estén en el mismo grupo relacionado. Dependiendo de la definición de la distancia entre cada uno de los grupos, se les llamam,$single - linkage, complete - linkage, Average - linkage$. En el algoritmo $Single - linkage$, la distancia entre dos grupos está dada por la distancia mínima entre todos pares de objetos. En el algoritmo $Complete - linkage$, la distancia es máxima distancia entre todos los pares de objetos de los clusters. Finalmente, en el algoritmo $Average - linkage$,la distancia media entre todos los miembros de los grupos es la que se considera,(Jain & Dubes, 1998). Formalmente, las distancias entre clusters en algoritmos jerárquicos se definen como sigue:

1. Single-linkage
$$d(c_1, c_2) = min(d(e_i, e_j)), \forall e_i \in c1 \wedge e_j \in c2 \tag{2.1}$$
2. Complete-linkage
$$d(c_1, c_2) = max(d(e_i, e_j)), \forall e_i \in c1 \wedge e_j \in c2 \tag{2.2}$$
3. Average-linkage
$$d(c_1, c_2) = \frac{\sum_{\forall e_i \in c_1} \sum_{\forall e_j \in c_2} d(e_i, e_j)}{|c1|\,|c2|} \tag{2.3}$$

2.2.2 Algoritmo de clustering MSTkNN

El algoritmo de clustering MSTkNN corresponde a un algoritmo de partición basado en grafos, que se encuentra basado en la utilización de los grafos de proximidad

- *Minimum Spanning Tree* (MST, Árbol de Cobertura Mínimo)
- *k-Nearest Neighbors* (kNN, k vecinos más cercanos).

El algoritmo parte de una matriz D con la distancia entre n objetos relacionados. Luego, se crea un grafo completo no dirigido *G(V, E, W)*, donde el peso $w(e) \in W$ de cada arista $e \in E$ corresponde a la distancia entre los objetos. Posteriormente, se calcula el MST de G (llamado $G_{MST}(V, E_{MST})$, con $E_{MST} \subset E$) y el $kNNdeG$ (llamado $G_{kNN}(V, E_{kNN})$, con $E_{kNN} \subseteq E$) utilizando $k = ln(n)$. Posteriormente se calcula el grafo particionado (con los clusters identificados, G cluster (V, E cluster)), se la intersección de ambos conjuntos de aristas $E_{cluster} = E_{MST} \cap E_{kNN}$. Esto producirá una partición del grafo en una o más componentes conexas, que corresponden a los clusters naturales del conjunto de datos.(Ponta, 2008)

Una modificación a algoritmo, fue presentada por (Ponta, 2008), en la cual:

- El valor de k es redefinido como:

$$k = \{\lfloor ln(n) \rfloor \, ; min \; i / G_{iNN} \; es \; conexo\} \tag{2.4}$$

- El algoritmo se aplica recursivamente en cada una de las componentes conexas, hasta que el número de componentes generadas sea uno.

Después de calcular el grafo *Gcluster*, el algoritmo se aplica de de forma recursiva, de forma independiente sobre cada una de las componentes conexas encontradas. La recursión se detiene cuando la cantidad de componentes conexas encontradas es $(c = 1)$. Luego, el algoritmo retorna un grafo $G_{cluster}(V, E_{cluster})$, compuesto de $c \geq 1$ componentes conexas $(G_{cluster} = G^1_{cluster} \cup ... \cup G^c_{cluster})$.

Luego, el algoritmo se muestra en la Figura 2.1:

Algoritmo 2.1: algoritmo de clustering MSTkNN

Entrada: Matriz de Distancias D
Salida: Cluster naturales conjunto de datros
1: calcular G
2: calcular G_{MST}
3: calcular G_{kNN}
4: $G_{cluster} = V_{cluster} = V, E_{cluster} = E_{MST} \cap E_{kNN}$
5: $c = connectedComponents(G_{cluster})$
6: **if** $c > 1$ **then**
7: $G_{cluster}$=$\bigcup_{i=1}^{c} MSTKNN(submatrix(D, G^{i}_{cluster}))$
8: **end if**
9: **return** $G_{cluster}$

2.3 VALIDEZ DEL CLUSTERING

El análisis de clusters es un proceso no supervisado. Esto implica que no existe una forma única de validar los resultados. Luego se hacen necesarios métodos de valición extera o análisis de correlación que indiquen que efectivamente se converge a una solución determinada. A continuación se presenta diferentes análisis de correlación que pueden aplicarse en la determinación del conjunto final.

2.3.1 Análisis de correlación

Un índice de validez externa o índice de correlación, compara dos particiones distintas de los datos y trata de medir cuánto se parecen o en qué medida concuerdan dichas particiones. Ya que se trata de validar el resultado de un método de clustering, las particiones a comparar son la que entregue el método y la partición ideal, que se determina a través de las etiquetas asignadas por un experto. Se denomina validez "externa" ya que se necesita esta información adicional a los datos para realizar la comparación.

Si $P = \{p_1, p_2, p_3...p_s\}$ es la partición ideal y $C = \{c_1, c_2, c_3...c_m\}$ es la partición generada por el método, entonces se pueden usar los siguientes términos para referirse a un par de patrones $\{x_i, x_j\}$ de conjunto de datos.

1. SS: si ambos patrones pertenecen al mismo cluster en C y al mismo grupo en P.

2. SD: si los patrones pertenecen al mismo cluster en C, pero a diferentes grupos en P.
3. DS: si los patrones están en distintos clusters en C, pero en el mismo grupo en P.
4. DD: si los patrones están en distintos clusters en C y en distintos grupos en P.

Se puede apreciar que no necesariamente dos particiones presenten los mismos cluster como resultado, el número de grupos en P puede ser distinto al número de clusters en C. Supóngase que a, b, c y d corresponden al número de pares tipo SS, SD, DS y DD, respectivamente, con lo que se tiene que $a + b + c + d = M$ (que corresponde al total de pares que se pueden formar con los datos, es decir $M = \frac{n(n-1)}{2}$). Se definen los siguiente índices de validez, que indican la similitud entre C y P:

1. Rand
$$R = \frac{a+d}{M} \tag{2.5}$$
2. Jaccard Index
$$J = \frac{a}{a+b+c} \tag{2.6}$$
3. Fowlkes y Mallow
$$FM = \frac{a}{\sqrt{m_1 m_2}} = \sqrt{\frac{a}{a+b} \cdot \frac{a}{a+c}} \tag{2.7}$$

donde $m_1 = a + b$ es el número de pares que pertenecen al mismo clusters en C, y $m_2 = a + c$ es el número de pares que pertenecen al mismo grupo en P. Para los tres índices anteriores, un valor más alto indica una mayor correspondencia entre C y P.

Los criterios de validación externa son útiles cuando se dispone de una partición de referencia. Debido a esto, en la práctica su uso se limita a evaluar el desempeño de un algoritmo de clustering en conjuntos de datos tipo benchmark, es decir, aquellos para los que se dispone de etiquetas previamente asignadas por un experto. Este análisis es de utilidad para caracterizar el comportamiento de un método en conjuntos de datos con diferentes estructuras de clusters. Aplicando este análisis a varios métodos de clustering se puede concluir cuáles son los más adecuados para cada tipo de estructura. Sin embargo, en una aplicación real de clustering, no se dispone de etiquetas previas, y en este caso los índices de validez externa no son de utilidad.

CAPÍTULO 3. ANÁLISIS ESTADÍSTICO

En el presente capítulo, se analizará la metodología estadísitca que se utilizará para la determinación del concenso entre las diferentes particiones del algoritmo MSTkNN.

3.1 METODOLOGÍA DE CONSENSO UTILIZANDO ALGORITMO MSTKNN

Una vez determinado el algoritmo de partición por grafos, como también la forma de perturbar los datos de entrada a través de múltiples iteraciones, se procede a construir una matriz de correlación de Jaccard, que posteriormente se analizará de forma estadística, con el objetivo de obtener el consenso entre las diferentes particiones de grupos iterados.

Luego, el análisis estadístico se encuentra basado principalmente en:

- Identificación de valores atípicos, *.ºutliers"*
- Cálculo de percentiles
- Determinación de máximos y mínimos
- Análisis de correlación entre particiones basado en índice Jaccard
- Análisis de distribuciones de frecuencias empíricas

A continuación, se presenta la descripción de cada uno de estos procesos.

3.1.1 Valores fuera de rango *outliers estadísticos*

Son observaciones cuyos valores son muy diferentes a las otras observaciones del mismo grupo de datos. Los datos atípicos son ocasionados por:

1. Errores de procedimiento.
2. Acontecimientos extraordinarios.
3. Valores extremos.
4. Causas no conocidas.

Los datos atípicos distorsionan los resultados de los análisis, y por esta razón hay que identificarlas y tratarlos de manera adecuada, generalmente excluyéndolos del análisis. En estadística, un valor atípico es un dato que es considerablemente diferente a los otros datos de la muestra. Con frecuencia, los valores atípicos en un conjunto de datos pueden alertar a los estadísticos sobre las anormalidades experimentales o los errores en las mediciones tomadas, y debido a esto puede que los descarten del conjunto de datos. Si los valores atípicos del conjunto se ignoran, puede haber cambios importantes en las conclusiones obtenidas del estudio.

3.1.2 Cálculo del percentiles

Un percentil es el valor de una variable bajo el cual un cierto porcentaje de las observaciones caen. De este modo el percentil 20 es el valor bajo el cual el 20 % de las observaciones pueden ser encontradas.

Algunos tipos de percentiles importantes son:

1. Los cuartiles: percentil 25, 50 (mediana) y 75
2. Los quintiles: percentil 20, 40, 60 y 80
3. Los deciles: percentiles 10, 20,..., 90

No existe una definición estándar de percentil, sin embargo todas las soluciones dan resultados similares cuando el número es grande:

1. Definición 1: El percentil p de N valores, se obtiene primero calculando el siguiente rango: rango n-(N/100)*p en el caso de número impares y en el caso de números pares el rango es: rango n-(N/100)*p + (p/100), redondeándolo al entero más próximo y tomando el número que corresponde a dicho rango.

2. Definición 2: Usando interpolación linean entre los dos rangos, más cercanos en vez de redondear.

En el presente trabajo se utilizará la primera definición.

3.1.3 Identificación de valores atípicos

Existen diversos criterios para detectar a los valores "outliers" en un conjunto determinado de datos. Uno de los métodos más utilizados es el que utiliza el concepto de cuartil de un conjunto de datos. Si se tiene un conjunto de datos y se ordena de mayor a menor, el Cuartil 1, se denomina Q1, es el valor tal que desde ese valor hacia su izquierda se encuentran la primera cuarta parte de los valores de este conjunto de datos.

El Cuartil 2, se denomina Q2, es el valor tal que desde ese valor hacia su izquierda se encuentran la primera mitad de los valores de este conjunto de datos. Y así sucesivamente.

3.1.4 Determinación de máximos y mínimos

3.1.5 Valor atípico leve

Siendo Q1 y Q3 el primer y tercer cuartil e IQR, el rango intercuartil (Q3-Q1), un valor atípico leve será:

$$< Q1 - 1,5 * IQRo > Q3 + 1,5 * IQR \tag{3.1}$$

Q1 y Q3 son límites interiores, a partir de los cuales la observación se determina "atípico leve"

El factor de IQR puede variar dependiendo de qué tan leve o extremo se define el valor atípico.

3.1.6 Valor atípico extremo

Los valores atípicos extremos son observaciones más allá de los límites externos:

$$< Q1–3 * IQRo > Q3 + 3 * IQR \tag{3.2}$$

CAPÍTULO 4. ALGORITMO DE CONSENSO

En el presente capítulo, se se desarrollará el algoritmo de consenso en base a la versión MSTkNN de (Ponta, 2008).

4.1 MODIFICACIÓN DEL ALGORITMO MSTKNN PARA ESTABLECIMIENTO DE CONSENSO

La modificación al algoritmo MSTkNN, se basa en la inclusión de estructuras que almacenan los difentes coefiencientes de correlación basados en el cálculo del índice de Jaccard (2.6), acotar el conjunto solución dejando fuera los valores *.ºutliers"* y también establecer medidas de consenso en base a distribuciones de frecuencias empíricas.

El algoritmo de consenso basado en MSTkNN, se pueden apreciar en la Figura 4.1

Algoritmo 4.1: algoritmo de clustering MSTkNN utilizando medidas consenso

Entrada: Matriz de Distancias D
Salida: Cluster naturales conjunto de datros

1: calcular G
2: calcular G_{MST}
3: calcular G_{kNN}
4: $G_{cluster} = V_{cluster} = V, E_{cluster} = E_{MST} \cap E_{kNN}$
5: $c = connectedComponents(G_{cluster})$
6: **if** $c > 1$ **then**
7: $G_{cluster}$=$\bigcup_{i=1}^{c} MSTKNN(submatrix(D, G^{i}_{cluster}))$
8: P_k=$G_{cluster}$
9: **end if**
10: $JI^h = P^h(i,j)$
11: $M^h(i,j)$=JI^h
12: calcular $M(i,j)$
13: outliers($M(i,j)$)
14: consensusClustering($M(i,j)$)
15: **return** $G_{cluster}$

El algoritmo al iterar, genera *N-1* particiones, cada una de ellas se encuentra integradas por grupos que a su vez están integrados por objetos que se relacionan unos con otros a través de una cierta medidad e distancia.

Las particiones entre sí, poseen un índice de correlación, que se calcula de acuerdo a lo establecido en (2.6).

Una vez establecido el índice de correlación entre particiones, se procede a almacenar el nuevo índice promedio, de acuerdo a lo expresado en la ecuación: (4.2):

$$M(i,j) = \frac{\sum_{i=1}^{k} M^h(i,j)}{\sum_{i=1}^{k} P^h(i,j)} \tag{4.1}$$

El índice promedio entre particiones $M^h(i,j)$, sirve para determinar la distribución de frecuencia de la curva, que luego se utiliza para determinar el conjunto solución o cluster final, a través de los métidos estadísticos descritos en los *items* anteriores.

La distribución de frecuencias de la curva, es definida como la frecuencia de concidencias entre los diferentes índices promedios de Jaccard. La distribución se define como:

$$Df(P_h) = \frac{M(i,j)}{(N-1)} \tag{4.2}$$

Donde P_h es definida como una particion dentro del conjunto de interaciones que agrupa a su vez un conjunto de elementos, $M(i,j)$ el índice promedio de Jaccard y *(N-1)* la cantidad de iteraciones.

Luego, el conjunto solución o cluster final, se determina a partir de la mayor densidad de la curva de distribución, puesto que a través de las diferentes iteraciones, se determina esta convergencia.

4.2 RESULTADOS EXPERIMENTALES

El presente experimento utiliza un conjunto de datos que corresponde la expresión génica de 79 muestras relacionadas con el ciclo celular de la levadura saccharomyces cerevisiae.

El algoritmo de acuerdo su lógica, itera *N-1* veces, esto corresponde a la alteración de cada una de las aristas relacionadas al árbol cobertor mínimo MST_0. Esto tiene como objetivo,

obtener las diferentes particiones y por tanto, el índice promedio de Jaccard correspondiente a la relación entre cada una de las particiones.

Luego, al finalizar el cliclo de iteraciones, se calcula el índice promedio de *Jaccard,M(i,j)*, lo que entrega como resultado el gráfico de la figura (4.1):

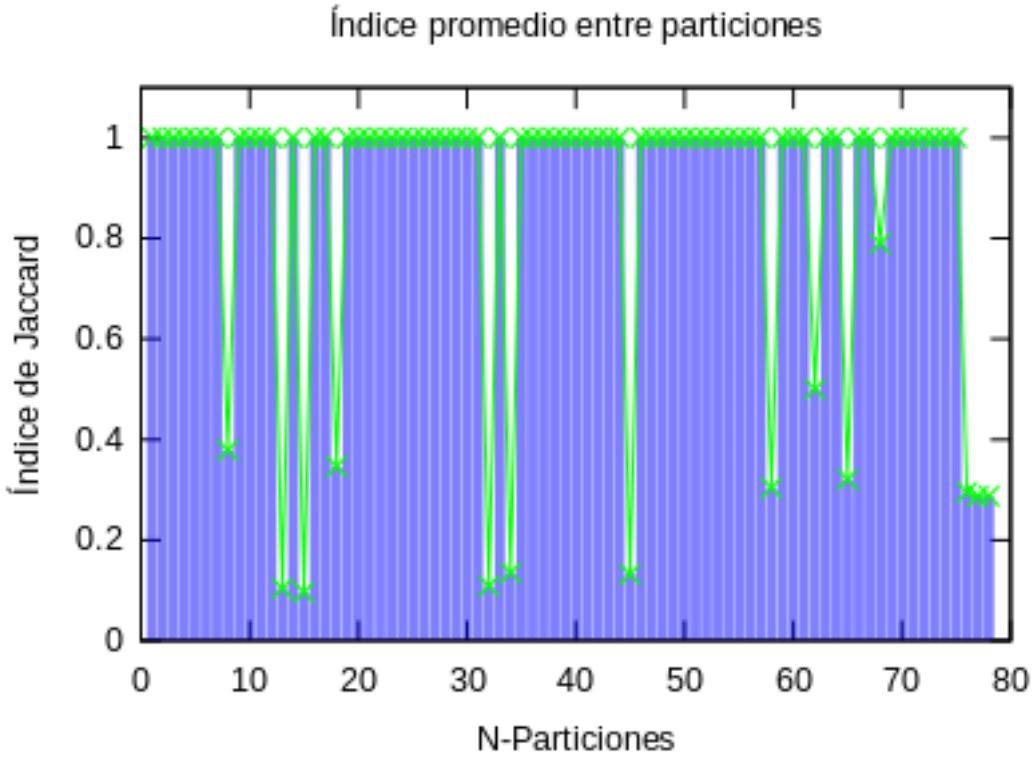

FIGURA 4.1: Histograma que muestra la relación promedio entre el índice promedio de *Jaccard* y el conjunto de particiones.

El gráfico de la figura (4.2), muestra la distribución ascendente de los'indices promedios de *M(i,j)*.

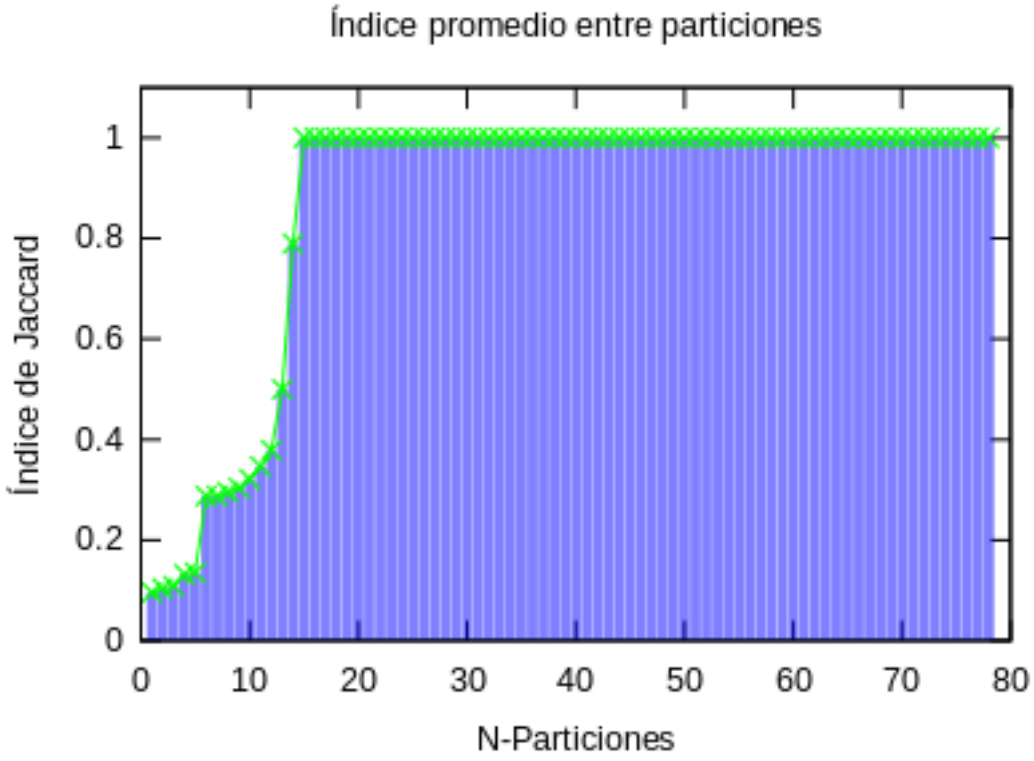

FIGURA 4.2: Histograma que muestra la distribución promedio ascendente entre índice de promedio de *Jaccard* y el conjunto de particiones

Como se puede visualizar en el gráfico (4.2), la distribución total del conjunto de particiones se presenta homogénea, en gran parte del conjunto. Luego, los umbrales, $(umax_P, umin_P)$ y percentiles (P_{25}, P_{75}), se definen en la tabla 4.1:

Resultados	
PERCENTIL 25	0.872856
PERCENTIL 75	0.872856
UMBRAL MÁXIMO	0.872856[+IQR]
UMBRAL MÍNIMO	0.872856[-IQR]

TABLA 4.1: Tabla de resultados histograma (4.2)

Luego, en base al análisis de la curva de distribución, que relaciona el índice promedio entre particiones, además de establecer como política de selección que el conjunto posea la media aritmética mas alta y el coeficiente de variación mas bajo, el conjunto solución se restringe aplicando (4.3):

$$\overline{X_{P_I}} <= umax \wedge \overline{X_{P_I}} >= umin \Rightarrow C_s = P_h \tag{4.3}$$

Posteriormente, se comienza el análisis de consenso, en base a la partición P_h seleccionada.

Sobre el conjunto de índices de la partición P_h, se aplica la ecuación de distribución de frecuencias (4.2), con el objetivo de obtener la densidad y la convergencia hacia el índice promedio. Esto da como resultado el gráfico de la figura 4.3

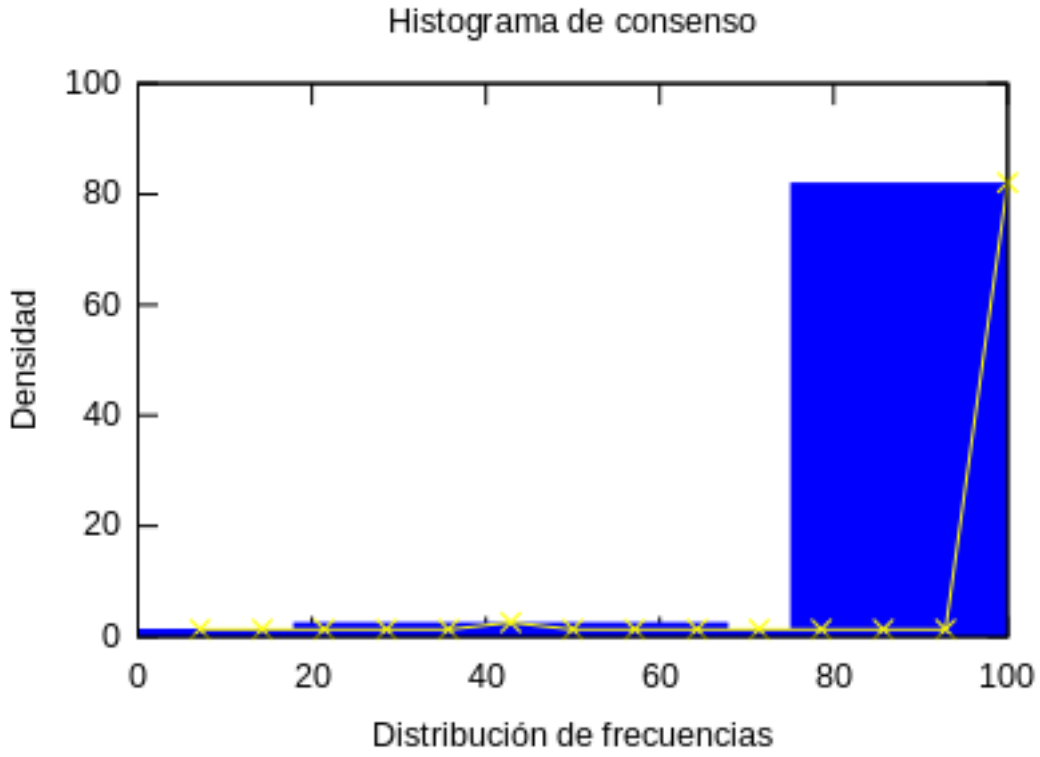

FIGURA 4.3: *Histograma de consenso que muetra la densidad de frecuencia y convergencia al índice promedio*

El gráfico anterior indica que existe una alta densidad de concentración, que agrupa al 82 % de las particiones. Esto indica que las particiones agrupadas en este segmento presentan una alta tasa de correlación. La tabla (4.2) muestra en detalle la densidad relativa de la particiónes P_h:

Resultados histograma de consenso	
1	1.2820512821
1	1.2820512821
1	1.2820512821
1	1.2820512821
1	1.2820512821
2	2.5641025641
1	1.2820512821
1	1.2820512821
1	1.2820512821
1	1.2820512821
1	1.2820512821
1	1.2820512821
1	1.2820512821
64	82.0512820513

TABLA 4.2: Tabla de resultados histograma (4.2)

Finalmente, de acuerdo al análisis anteior, se considera como solución final a los componentes de la particion P_{68}, que se encuentra dentro del conjunto que presenta mayor densidad, tabla ().

Resultado cluster final partición P_{68}	
Id	Componentes
C1	alpha14 alpha21
C2	alpha28 alpha35
C3	alpha42 alpha49 alpha56
C4	alpha63 alpha70 alpha77 alpha84 alpha91 alpha105 alpha119
C5	alpha98 alpha112
C6	spomid heat160 Elu60 Elu90 Elu120 Elu150 Elu180 Elu210 Elu240 Elu270 Elu330 Elu360 heat20 dtt60 dtt120 diauf diaug
C7	Elu300 Elu390
C8	cdc1510 cdc1530
C9	cdc1550 cdc1570
C10	cdc1590 cdc15110
C11	cdc15130 cdc1515
C12	spo511 spoearly heat0 heat10 heat40 heat80
C13	cdc15210 cdc15230 spo0 diaua diaub diauc diaud diaue
C14	spo2 spo5
C15	spo7 spo9
C16	spo11 spo52 spo57 dtt15 dtt30

TABLA 4.3: Tabla de resultados componentes del cluster 1 al cluster 16 sobre la particion P_{68}

CAPÍTULO 5. CONCLUSIONES

El objetivo del proyecto fue diseñar e implementar un algorito de agrupamiento de concenso utilizando las características de los grafos de proximidad *Minimum Spanning Tree* y *k-Nearest Neighbors*, utilizando como base de desarollo, el algoritmo *MSTkNN* (Ponta, 2008). Luego, se estima la calidad de la solición alterando con diferentes medidas de distancia el conjunto de entrada, manteniendo con ello su robustez. En relación al orden del algoritmo, las particiones generadas se comparan $\frac{n(n-1)}{2}$ veces, con el objetivo de obtener el índice de correlación de *Jaccard*, que luego se distribuye a través de una curva para obtener la distribución de concurrencia enpírica que define la partición resultado.

REFERENCIAS

Cohen, J. (2004). Bioinformatics—An Introduction for Computer Scientists. (pp. 122–155).

Jain, A. K., & Dubes, R. C. (1998). *Algorithms for Clustering Data*. New Jersey, MA, USA: Prentice Hall Englewood Cliffs.

Jiang, T. C., D., & Zhang (2014). Cluster analysis for gene expression data: A survey. IEEE Transactions on Knowledge and Data Engineering. (pp. 1370–1386).

Laderas T.., M. S. (2007). Consensus Framework for Exploring Microarray Data Using Multiple Clustering Methods. OMICS: A Journal of Integrative Biology. (pp. 219–237).

Ponta, M. I. (2008). *An Integrated and Scalable Approach Based on Combinatorial Optimization Techniques for the Analysis of Microarray Data*. Ph.D. thesis, Universidad de Newcastle, Newcastle, Australia.

Stefano Monti, J. M. T. G., Pablo Tamayo (2003). Consensus Clustering: A Resampling-Based Method for Class Discovery and Visualization of Gene Expression Microarray Data.