
PROYECTO NO. 1 - ANÁLISIS EPIDEMIOLÓGICO DE CONTAGIO DE ENFERMEDADES EN CÉLULAS A TRAVÉS DE UNA MATRIZ DISPERSA

202100154 – Sergio André Lima Corado

Resumen

El proyecto permite realizar un análisis epidemiológico de contagio de enfermedades, estudiando cada caso a partir de una rejilla de células contagiadas que conforme el tiempo irán contagiando a las células sanas y también sanándose ellas mismas. Este proyecto también permite realizar múltiples veces el análisis de contagio en las células e interpretar el diagnóstico del paciente. Es decir, si cuenta con un caso leve, grave o mortal.

Para realizar este proyecto se implementó el uso de listas enlazadas y de matrices dispersas, permitiendo ahorrar recursos en memoria y también efectuar el estudio de una manera eficaz y eficiente. Las matrices dispersas y las listas enlazadas permiten almacenar solamente los datos que necesitamos de forma dinámica, permitiendo durante la ejecución ir incrementando y disminuyendo los mismos; omitiendo datos intermedios que estarían vacíos si se utilizaran otros métodos.

Palabras clave

Matriz dispersa; Epidemiología; Diagnóstico; Rejilla; Periodo

Abstract

This project allows to perform an epidemiological analysis of disease contagion, studying each case from a grating of infected cells that over time will infect healthy cells and heal themselves. This project also allows to perform multiple times the analysis of contagion in the cells and to interpret the patient's diagnosis. That is, whether the patient has a mild, severe, or fatal case.

To carry out this project, the use of linked lists and sparse matrices was implemented, allowing to save memory resources and to carry out the study in an effective and efficient way. Sparse matrices and linked lists allow to store only the data we need dynamically, allowing during the execution to increase and decrease them; omitting intermediate data that would be empty if other methods were used.

Keywords

Sparse Matrix; Epidemiology; Diagnosis; Grating; Period

Introducción

La epidemiología ha cumplido con una tarea importante al permitir que el movimiento de salud pública logre una de sus tareas actuales: el control de las enfermedades infecciosas y la conquista de las enfermedades no infecciosas. Esta disciplina científica se encarga de investigar y describir las dinámicas de la salud en las poblaciones, combinando principios de las ciencias sociales y biológicas. La importancia de la epidemiología radica en que, mediante ella, es posible determinar la frecuencia y tendencia de las enfermedades, entender cuáles son las intervenciones de prevención más eficaces.

Los avances en la tecnología ayudan a hacer la vida más fácil a los enfermos, ayuda a detectar antes cualquier tipo de enfermedad y encontrar curas más rápido, pues para poder estudiar a grandes rasgos y con mayor eficacia los distintos casos y enfermedades que se presentan, el uso de programas informáticos ha sido de gran ayuda para los científicos.

Desarrollo del tema

Como fue mencionado, el uso de la tecnología dentro de las ciencias de la salud o la epidemiología juega un papel importante, por lo que para nosotros como desarrolladores implica también un campo laboral en el que podemos desempeñarnos e incluso llegar a especializarnos en él. Durante este proyecto se presenta la solución informática al análisis de células que son contagiadas inicialmente por un virus o una enfermedad y que como es de saber, progresivamente empiezan a contagiarse entre ellas.

Dichos patrones de contagio fueron previamente estudiados y definidos, por lo que forman la base teórica en cuanto a epidemiología en el proyecto. Pues en base a esas dos reglas que fueron definidas, se realizaron los algoritmos que determinaban tanto el contagio de las células sanas como la cura de las células ya contagiadas. Las dos reglas son las siguientes:

1. Toda célula contagiada, continúa contagiada si tiene exactamente 2 o 3 células contagiadas en las celdas vecinas, de lo contrario sana para el siguiente periodo.
2. Cualquier célula sana que tenga exactamente 3 células contagiadas en las celdas vecinas, se contagia para el siguiente período.

El análisis de células siguiendo estas reglas a pequeños rasgos es posible, pues puede graficarse una rejilla con las células representadas por celdas e interpretar cada una de las reglas, así pudiendo identificar los patrones de contagio en las mismas. Pero para ser más precisos en el estudio de la epidemiología se realizan estudios masivos que además deben de ser precisos. Es decir, se analizan miles de células que llevarían demasiado tiempo analizar manualmente. Y es aquí donde entra en juego la informática y su aplicación en este campo; pues por medio del uso de algoritmos podemos llegar a analizar una cantidad inmensa de datos con ayuda del procesador y realizar este análisis en segundos. Que es el principal propósito de este proyecto.

El algoritmo puede presentarse sencillo para un desarrollador principiante, como lo somos la mayoría en este curso. Sin embargo, la solución a este problema utilizando los métodos y herramientas que previamente conocíamos le llevarían muchísimo tiempo y recursos al ordenador para poder realizarlos. Por lo que se nos presentó en este curso el tema de *Gestión de Memoria*, en dónde poco a poco hemos ido aprendiendo la importancia de la gestión de la memoria en el diseño de un programa o software.

Por lo que en este proyecto se implementaron listas enlazadas, que nos permiten solo ir almacenando e insertando los datos que vamos creando conforme el tiempo. Y no realizar un desperdicio de memoria con los datos que no ocupan todo el espacio que se les asigna. Además, por la cantidad inmensa de células que podemos llegar a manejar durante el proyecto, la implementación de estas listas nos permite llegar a analizar solamente las células contagiadas y simular el proceso a partir de ellas. No necesariamente analizando cada una de las células de la rejilla independientemente si están sanas o contagiadas, sino que solo tomamos la minoría que son las contagiadas y de esa manera ahorramos muchísimos recursos y tiempo en la ejecución.

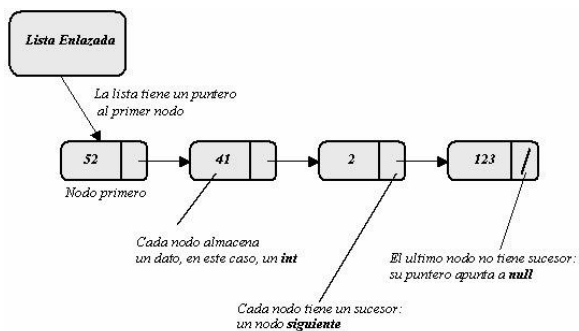


Figura 1. Representación gráfica de una lista enlazada.

Personalmente elegí realizar el proyecto por medio de una matriz dispersa, que es una aplicación más compleja de listas enlazadas. Se forma a partir de una o dos listas de encabezado que dan acceso a nodos internos que conforman la matriz. Esto le da mucha más versatilidad al proyecto además que regresa a ser muy parecido a lo que previamente se utilizaba en el curso anterior de IPC.

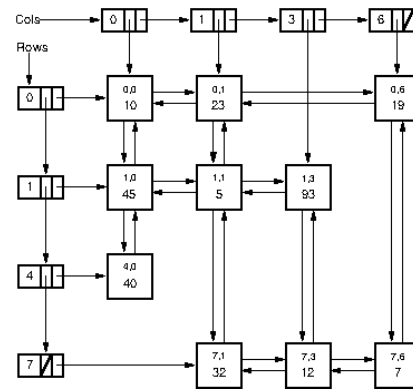


Figura 2. Representación gráfica de una matriz dispersa.

De este modo, se puede solamente analizar las células contagiadas como previamente se menciona, y solamente centrarse en ellas y sus células vecinas. Puede añadir un poco de complejidad del lado del desarrollador, pero será mucho más eficaz y eficiente para el usuario, que es el objetivo que debería de tener el programa o software.

El algoritmo que desarrollé dentro de esta matriz dispersa de acuerdo con las reglas 1 y 2 adjuntas en la página previa es el siguiente:

Tras ingresar cada una de las células contagiadas obtenidas del XML a una matriz dispersa, se obtiene la rejilla inicial. Para cada periodo se realiza lo siguiente, primero se extrae de la rejilla todas las células contagiadas almacenadas en una lista enlazada.

Para cada una de las células contagiadas se analizan las dos reglas previamente mencionadas en las 8 celdas vecinas a la misma. Que en el código fuente aparecen numeradas del 1 al 9.

Tabla I.

Esquema de celdas vecinas.

Celda 1	Celda 2	Celda 3
Celda 4	C. Contagiada	Celda 6
Celda 7	Celda 8	Celda 9

Fuente: elaboración propia.

Se omitió la celda 5 para poder conservar el orden que se presenta en los teclados numéricos y ser más intuitivo para los desarrolladores el ubicar la posición de la celda a tratar. Para cada celda vecina se realiza lo siguiente:

Se verifica si la celda vecina pertenece a la matriz dispersa, ya que esto indica que es otra célula contagiada y se suma a un contador interno de células vecinas contagiadas.

Si dicho contador interno es exactamente 2 o 3, la célula contagiada que estamos evaluando permanecerá contagiada, por lo que la añadimos directamente a nuestra nueva rejilla o la rejilla que resulta de este periodo.

De no pertenecer a la matriz dispersa, indica que es una célula sana, se añade dicha célula a una rejilla auxiliar o temporal que ejemplifica las células sanas que están siendo contagiadas, en esta matriz dispersa temporal al insertar una célula existente se aumenta un contador interno que indica las veces que la célula sana ha sido contagiada.

Al finalizar el análisis de todas las células contagiadas, se realiza un pequeño filtro en la rejilla auxiliar o temporal que previamente se utilizó. En el que obtenemos solamente aquellas células que fueron contagiadas tres veces a partir del contador interno con el que cuentan. Estas células en específico, las añadimos directamente a nuestra nueva rejilla ya que representan las nuevas células contagiadas.

A este punto ya contamos con la nueva rejilla o la rejilla resultante del algoritmo al finalizar este periodo. Ahora ya se puede repetir todo este proceso una y otra vez, media vez la rejilla inicial de cada periodo se iguale a la rejilla resultante al final de cada periodo. Y puede limitarse su ejecución a partir de un parámetro n que indique la cantidad de periodos que se esperan evaluar.

De este modo se generó un algoritmo eficaz y eficiente al gestionar los recursos de memoria durante la ejecución del programa y también presentando un ahorro significativo de tiempo, permitiendo a los usuarios ejecutar múltiples estudios y periodos de estudio epidemiológico inclusive de múltiples pacientes en un reducido tiempo.

Por otra parte, en el análisis epidemiológico no es suficiente solamente estudiar los contagios una cantidad n de veces, sino que también se tiene que concluir algo con lo que se obtiene de dichos estudios. La base teórica con que se cuenta para interpretar el estudio epidemiológico en un paciente es el siguiente:

1. Que el patrón inicial se repita siempre después de “N” períodos, en este caso la enfermedad produce un caso grave. Si “N” es igual a 1, entonces, el paciente morirá a causa de la enfermedad, ya que ésta será incurable.

b. Que algún patrón, distinto al patrón inicial, se repita luego de “N” períodos cada “N1” períodos, en este caso la enfermedad producirá un caso grave. Si “N1” es igual a 1, entonces, el paciente morirá a causa de la enfermedad, ya que ésta será incurable, en caso de que “N1” es mayor que 1, la enfermedad será grave

Para implementar la interpretación del estudio epidemiológico en cada paciente, se creó una lista enlazada para cada paciente que almacene todas las rejillas, desde la inicial hasta la del enésimo periodo. Accediendo a cada una de las rejillas de esta lista realizamos lo siguiente:

Analizamos si la rejilla o matriz dispersa es igual a la rejilla inicial del paciente, de este modo sabemos que se encontró un patrón repetitivo, pues al volver a ser la misma rejilla se repetirán todas las rejillas anteriores a la que está siendo analizada. Luego determinamos el número de periodos que han transcurrido desde la rejilla inicial. De ser este número igual a uno, el diagnóstico es un caso mortal, de ser mayor a uno se presentaría un caso grave.

Por otra parte, al mismo tiempo que se realiza la verificación de la rejilla actual con la rejilla inicial, se realiza la verificación de la rejilla actual con cada una de las anteriores excepto la primera. Es decir, se busca si la rejilla actual es idéntica a alguna rejilla

previa, ya que al ser idénticas se presentaría un nuevo patrón debido a que a partir de la rejilla en donde se encontró la similitud, se repetirán todas las rejillas hasta la rejilla actual. Siendo “N” la rejilla idéntica a la actual y “N1” la diferencia entre la rejilla actual y su rejilla idéntica. Similar al análisis anterior, de ser “N1” igual a 1, se diagnostica un caso mortal; y siendo “N1” mayor a 1 se diagnostica un caso grave.

Conclusiones

El uso de una matriz dispersa hace que se ahorren recursos a la hora de almacenar estas matrices en memoria, además de volver más eficaz y eficiente el trabajar con ellas.

Al diseñar un software es importante contemplar la cantidad de datos a analizar y la forma en que los datos de almacenarán, para de este modo diseñarlo con un TDA adecuado que lo convierta en un software eficaz y eficiente para el usuario.

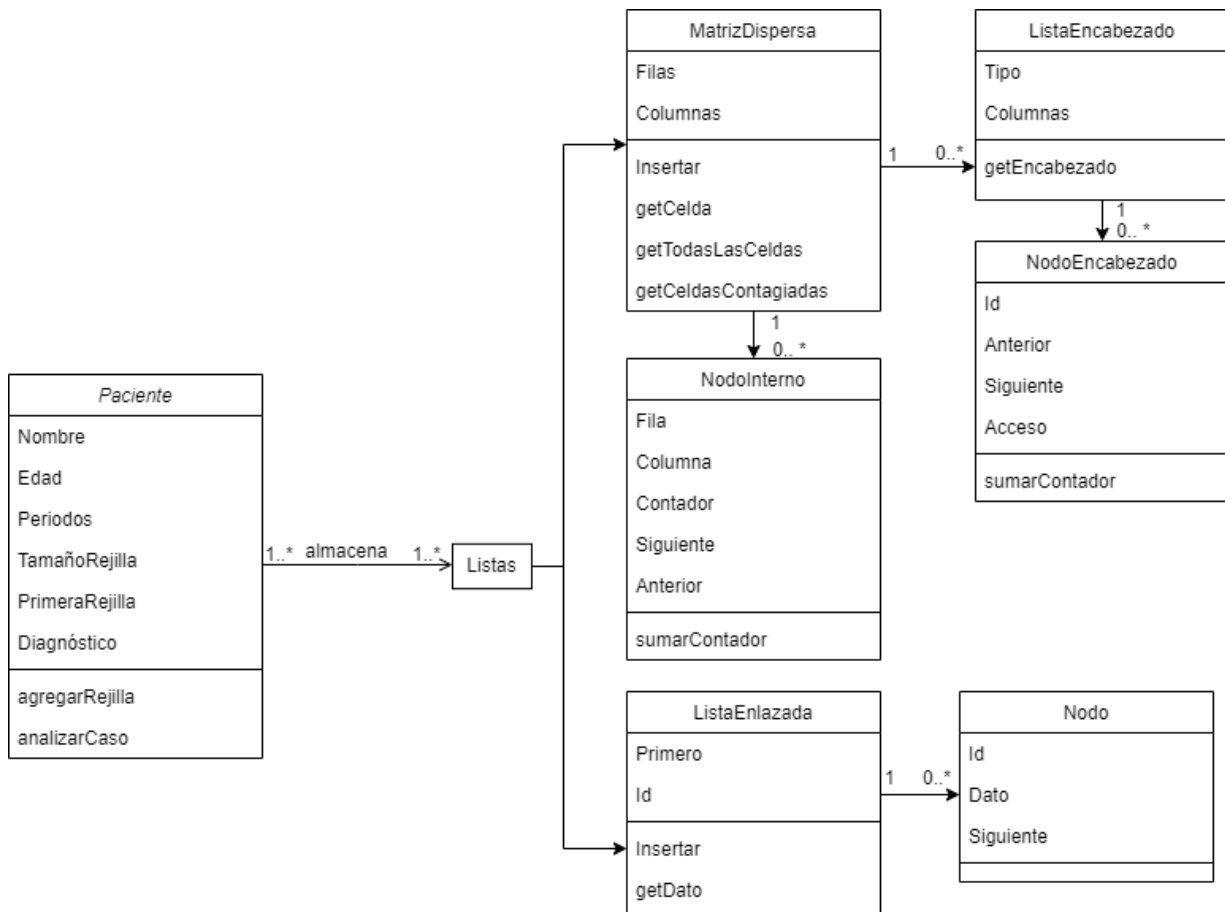
Referencias bibliográficas

- Bunch, J. R., & Rose, D. J. (Eds.). (2014). *Sparse matrix computations*. Academic Press.
- Calzadilla, J. C. F., Herrera, A. N., & Tchissingui, E. D. (2019). La enseñanza de los arrays estáticos, dinámicos y listas enlazadas; cuál usar? Análisis de códigos. *Revista Referencia Pedagógica*, 7(1), 75-93.
- Comas, I., Cancino-Muñoz, I., Mariner-Llicer, C., Goig, G. A., F. (2020). Uso de las tecnologías de secuenciación masiva para el diagnóstico y epidemiología de enfermedades infecciosas. *Enfermedades Infecciosas y Microbiología Clínica*, 38, 32-38.

Apéndices

Apéndice A.

Diagrama de clases del proyecto.



Fuente: elaboración propia