Predicción del tratamiento para la leishmaniasis cutánea mediante datos génicos e inferencia gramatical

J. Peña Atencio[1]

[1]*Facultad de Ingeniería y Ciencias, Pontificia Universidad Javeriana Cali*

Resumen

La leishmaniasis es una enfermedad parasitaria transmitida por moscas de arena infectadas que viven en ambientes tropicales. La forma más común de leishmaniasis en Colombia es leishmaniasis cutánea, la cual provoca úlceras en la piel. Para esta forma, el tratamiento actual mediante el medicamento Glucantime tiene un porcentaje de fracaso que varía entre el 19% y el 81%. Colombia y otros países afectados tienen poco interés en esta enfermedad y no cuentan con el conocimiento médico para garantizar un tratamiento seguro.

En el presente trabajo, se hace uso de 7 conjuntos de datos provistos por el CIDEIM de Cali, los cuales recolectan información de expresión génica de tres tipos de glóbulos blancos provenientes de 14 pacientes anónimos con leishmaniasis. Se utilizan dos algoritmos de inferencia gramatical llamados OIL y RPNI para predecir el posible resultado del tratamiento y así ayudar a prevenir la falla y complicaciones del mismo o para establecer un tratamiento alternativo. Estas técnicas han sido aplicadas con éxito en campos como la biología computacional y el procesamiento del lenguaje natural.

Se realizan 54 experimentos para OIL y 54 experimentos para RPNI; en cada uno los experimentos cada algoritmo se entrena y evalúa mediante una validación cruzada de 4 iteraciones. Se usan las métricas Accuracy, Precision, Recall y F1-Score para la evaluación. Se llegó a resultados satisfactorios en el trabajo, logrando en múltiples experimentos una tasa muy competitiva del 90% de Accuracy para RPNI y 68.8% de Accuracy para OIL.

Introducción

Debido a la reducida atención que se le presta a la enfermedad en Colombia y en los demás países afectados, no se cuenta con la suficiente experiencia y conocimiento médico para garantizar un tratamiento completamente seguro y exitoso. La medicina de primera línea para la enfermedad, las sales de antimonio (conocida comercialmente como Glucantime), tiene un porcentaje de falla terapéutica que varía desde el 19% hasta el 75% [1], y en hasta el 64% de los casos, causa efectos negativos en los pacientes [2]. Los efectos adversos clínicos notificados con mayor frecuencia incluyen fuerte dolor musculoesquelético, trastornos gastrointestinales y dolor de cabeza leve a moderado [2].

La dificultad para tratar la enfermedad recae sobre las complejas interacciones que se dan entre el parásito y el sistema inmune [3], estando este a su vez estrechamente relacionado con la expresión génica al ser parte fundamental para la capacidad de un organismo de generar una inmunidad sólida frente a los patógenos [4]. Leishmania es un parásito adaptado a su hospedador que a lo largo de los años ha ido desarrollando numerosas estrategias para lograr evadir los múltiples mecanismos que tiene el sistema inmune para eliminarla [5].

Objetivo General
Predecir la efectividad del tratamiento con Glucantime para la leishmaniasis cutánea empleando técnicas de inferencia gramatical con base en los datos génicos de pacientes.

Objetivos específicos
- Definir una representación gramatical adecuada para los datos génicos.
- Desarrollar las técnicas escogidas.
- Pre-procesar los datos génicos.
- Diseñar experimentos.
- Ejecutar experimentos.
- Evaluar el desempeño de predicción de las técnicas usadas.

Fundamentación teórica

Expresión génica
La expresión génica es el proceso mediante el cual las instrucciones que se encuentran codificadas en nuestro ADN se convierten en un producto funcional, como una proteína. Este regulado proceso permite que las células puedan responder a su entorno. Podría decirse que la expresión génica actúa como un botón de encendido o apagado que dicta cuáles y cuántas proteínas son producidas frente a algún estímulo [6].

Secuenciación de ARN
La secuenciación de ARN (Abreviada comúnmente como 'RNA-Seq' en inglés) es una técnica particular de secuenciación basada en tecnología que utiliza secuenciación de próxima generación (NGS) para revelar la presencia y cantidad de ARN en una muestra biológica en un momento dado. Ésta facilita la capacidad de observar procesos como los cambios en la expresión génica en diferentes tratamientos [7]. En los datos proporcionados por el CIDEIM, la secuenciación ARN se realiza con muestras biológicas de eosinófilos, monocitos, y neutrófilos; los cuales son tipos de glóbulos blancos en la sangre. Estas son células del sistema inmunológico que participan en la protección del cuerpo contra las enfermedades infecciosas, parásitos e invasores extraños.

Inferencia gramatical
La inferencia gramatical (IG) es transversal a una serie de campos que incluyen el aprendizaje automático, la teoría de lenguajes formales, la biología computacional, etc [8]. El problema a resolver en la IG es el aprendizaje del autómata que describe un lenguaje a partir de las muestras de observación del mismo. Los algoritmos OIL y RPNI logran la tarea de aprendizaje del lenguaje a través de autómatas finitos de fusión de estados (o state merging) deterministas y no-deterministas, respectivamente. Estos buscan fusionar los estados de un autómata finito inicial para dar como resultado un autómata más general, el cual posibilita realizar predicciones a partir de muestras nuevas que nunca ha observado antes.

Resultados

Preprocesamiento
El objetivo del preprocesamiento consiste en obtener los genes que pueden tener mayor incidencia en el resultado del tratamiento, y para esto se seleccionan los que presentan mayores niveles de inhibición o sobre-expresión génica entre todos los pacientes. Para los conjuntos de datos que contienen la información general de todos los pacientes, se eliminan los registros que tengan un valor P menor a 0.05 (valores significativos estadísticamente

hablando) y se realiza un ordenamiento y limpieza general de los datos para facilitar el resto de actividades y análisis subsecuentes.

Creación de alfabeto

En el contexto de autómatas de estados finitos, un alfabeto es un conjunto finito y no vacío de símbolos. Una cadena sobre un alfabeto es una secuencia finita de símbolos extraídos del mismo, y un lenguaje se define como un conjunto finito o infinito de cadenas de símbolos que son aceptadas por un autómata particular. A lo que se quiere llegar con la creación de un alfabeto es poder tener los bloques de construcción para representar el estado de expresión génica de un paciente en una cita particular, de tal forma que esto pueda ser entendido por las técnicas de inferencia gramatical. Se pensaría que para la creación de un alfabeto se necesitarían utilizar caracteres como 'a' o 'b'. Pero, la cantidad de símbolos del alfabeto latino no es suficiente para las cantidades de genes tratadas. Es por esto que se usan números enteros como '1', '20' o '60'.

Codificación de genes

El siguiente paso es codificar la cantidad de ARN mensajero que presenta cada paciente en sus diferentes citas. Se toma, por cada gen seleccionado, la media y desviación estándar de la cantidad de ARN de ese gen en cada cita de cada paciente. Estas dos métricas se utilizan para clasificar a cada gen como infra-expresado, sobre-expresado, o estable.

Codificación Binaria

Cuando el valor de ARN registrado en un paciente es inferior a la media menos una desviación estándar, se considera que este gen está inhibido o infra-expresado. A este gen se le asigna un número i entre 1 y n, siendo n la cantidad de genes en la tabla de datos de pacientes determinada. Este i es el número de la fila en la que se encontraba el gen. Lo anterior quiere decir que los genes que tienen un símbolo entre [1, n] son infra-expresados, mientras que los que tengan un símbolo entre [n+1, 2n] son sobre-expresados. Se muestra un ejemplo de una palabra creada en codificación binaria.

Ejemplo creación de palabra para paciente Z en codificación binaria

Fila (i)	**$\overline{x}$-std**	**$\overline{x}$+std**	**Paciente Z**	**Clasificación**	**Símbolo**
1	0.111	1.095	0.702	No clasifica	No tiene
2	5.780	11.808	10.753	No clasifica	No tiene
3	2.532	**5.914**	6.569	**Sobreexpresado**	$i + n = 13$
4	**154.445**	277.777	124.205	**Infraexpresado**	$i = 4$
5	-0.006	**0.160**	0.386	**Sobreexpresado**	$i + n = 15$
6	0.227	0.793	0.773	No clasifica	No tiene
7	18.094	34.744	25.505	No clasifica	No tiene
8	0.916	**4.134**	5.024	**Sobreexpresado**	$i + n = 18$
9	10.589	24.549	22.413	No clasifica	No tiene
10	-0.048	**0.156**	0.386	**Sobreexpresado**	$i + n = 20$

Codificación ternaria

Se incluye la noción de genes 'estables' con el fin de obtener un poco más de información del perfil de expresión génica de cada paciente. Se usa 1/4 de la desviación estándar para clasificar un gen como estable: si la cantidad de ARN del gen se encuentra entre 1/4 de la desviación estándar sobre o bajo la media, se dice que el gen se ha mantenido estable, y se le asigna un símbolo entre [2n+1, 3n]; Es decir tomando la fila en la que se encontraba, i, y sumando 2n. En esta codificación, de la misma forma que con la codificación binaria, los rangos [1, n] y [n+1, 2n] corresponden a genes infra y sobre expresados, respectivamente.

Creación de palabras

Se busca reconocer en qué instancias el tratamiento de un paciente va a fallar, lo cual es una ocurrencia minoritaria entre todos los casos del mundo real de la enfermedad. Se dice entonces que nos interesa generar autómatas que acepten las palabras de pacientes cuyo tratamiento va a fallar en el futuro. En la inferencia gramatical, a las palabras que pertenecen al lenguaje objetivo también se les llama muestras positivas. Por otra parte, las palabras que no nos interesan reconocer, son las de pacientes cuyo tratamiento va a ser exitoso, que son la mayoría de casos en esta enfermedad. Se dice entonces que los autómatas que nos interesa generar deben rechazar estas palabras que no pertenecen al lenguaje objetivo. Las palabras que no pertenecen al lenguaje también se les denomina muestras negativas.

Algoritmo RPNI

El algoritmo de RPNI consiste en generar un autómata de prefijos inicial a partir de las palabras positivas y negativas, y realizar la mezcla de la mayor cantidad de estados posibles hasta llegar al autómata mínimo que generalice al lenguaje que se está tratando de identificar. El entrenamiento consiste en tomar los datos preprocesados en forma de muestras de palabras y escribirlos en un archivo de texto que el algoritmo lee y produce un resultado. Este resultado es el autómata final luego de realizar todas las mezclas intermedias. Se utiliza una validación cruzada de 4 bloques para cada tipo de célula (neutrófilos, eosinófilos, y monocitos) para evaluar el desempeño de la técnica. Se muestra 1 de las 6 tablas de resultados obtenidos con el algoritmo.

Métricas de Desempeño RPNI - Neutrófilos - Codificación Binaria

	Accuracy		Precision		Recall		F1-Score	
Tamaño	**mean**	**std**	**mean**	**std**	**mean**	**std**	**mean**	**std**
12	0.854	0.172	0.875	0.250	0.917	0.167	0.867	0.163
16	0.875	0.144	0.854	0.172	1.000	0.000	0.914	0.102
20	0.750	0.300	0.708	0.344	0.833	0.333	0.725	0.320
24	0.750	0.300	0.708	0.344	0.833	0.333	0.725	0.320
28	0.650	0.252	0.708	0.344	0.708	0.344	0.642	0.263
32	0.650	0.252	0.708	0.344	0.708	0.344	0.642	0.263
40	0.650	0.252	0.708	0.344	0.708	0.344	0.642	0.263
60	0.550	0.100	0.583	0.500	0.458	0.417	0.417	0.289
80	0.600	0.283	0.500	0.577	0.250	0.319	0.325	0.395

Algoritmo OIL

A diferencia de RPNI, este algoritmo utiliza autómatas no-deterministas en lugar de deterministas. En vez de tener dos conjuntos de palabras positivas y negativas, OIL hace uso de particiones de los conjuntos de palabras para realizar la mezcla de estados. Cuando se intenta realizar una mezcla y esta causa inconsistencias (mezclar un estado de aceptación con uno de rechazo), el algoritmo genera un autómata extra con un estado de inicio adicional desconectado del resto del autómata general, aprovechando la cualidad de no-determinismo y acomodando el patrón de observación que quiere ser mezclado. El procedimiento de entrenamiento y evaluación es igual al usado en el algoritmo RPNI. Se muestra 1 de las 6 tablas de resultados obtenidos con el algoritmo.

Métricas de Desempeño OIL - Monocitos - Codificación Binaria

	Accuracy		Precision		Recall		F1-Score	
Tamaño	**mean**	**std**	**mean**	**std**	**mean**	**std**	**mean**	**std**
12	0.667	0.312	0.500	0.577	0.417	0.500	0.450	0.526
16	0.500	0.430	0.000	0.000	0.000	0.000	0.000	0.000
20	0.500	0.430	0.000	0.000	0.000	0.000	0.000	0.000
24	0.479	0.443	0.000	0.000	0.000	0.000	0.000	0.000
28	0.479	0.267	0.000	0.000	0.000	0.000	0.000	0.000
32	0.479	0.267	0.000	0.000	0.000	0.000	0.000	0.000
40	0.479	0.267	0.000	0.000	0.000	0.000	0.000	0.000
60	0.475	0.250	0.000	0.000	0.000	0.000	0.000	0.000
80	0.413	0.272	0.000	0.000	0.000	0.000	0.000	0.000

Discusión y conclusiones

Se observa que RPNI demuestra mejores resultados para la predicción de la falla de tratamiento que el algoritmo OIL. Como OIL es un algoritmo más complejo que RPNI, y la cantidad de datos con la que estamos trabajando es pequeña, es normal obtener resultados poco precisos. En casi todos los casos, el autómata resultante producido por el algoritmo no tiene la suficiente información para generalizar el lenguaje. Si se tuvieran conjuntos de datos bastante más grandes, se podría apreciar el poder de predicción de esta técnica.

A modo de conclusión, se logró predecir la efectividad del tratamiento de la leishmaniasis cutánea haciendo uso de técnicas de inferencia gramatical y datos génicos de pacientes, el cual es el objetivo general del proyecto. A su vez, se diseñó la representación gramatical de los datos génico que fueron preprocesados, y se diseñaron y ejecutaron diversos experimentos para entrenar y evaluar el desempeño de las 2 técnicas implementadas y usadas; cumpliendo de esta forma los objetivos específicos de la investigación. Para trabajos futuros, se necesita hacer un análisis más minucioso con respecto a las decisiones iniciales como en la del valor P a considerar como significativo.

En cuanto al impacto frente a las investigaciones sobre la enfermedad, es muy valioso proveer los genes particulares de cada tipo de célula que parecen estar correlacionados con buenos desempeños de predicción de tratamiento. Finalmente, se concluye que la inferencia gramatical es una técnica que debe ser tenida en cuenta en este tipo de problemas. En este proyecto la técnica demostró que se le puede sacar provecho a la estructura de los datos que se tenga para trabajar, llegando a resultados de desempeño respetables, incluso en condiciones en las que los conjuntos de datos son tan pequeños.

Referencias

1. D. A. Vargas, M. D. Prieto, A. J. Martínez-Valencia, A. Cossio, K. E. V. Burgess, R. J. Burchmore y M. A. Gómez, “Pharmacometabolomics of Meglumine Antimoniate in Patients With Cutaneous Leishmaniasis,” Frontiers in Pharmacology, vol. 10, jun. de 2019, issn: 1663-9812. doi: 10.3389/fphar.2019.00657. dirección: https://www.frontiersin.org/article/10.3389/fphar.2019.00657/full.

2. L. F. Oliveira, A. O. Schubach, M. M. Martins, S. L. Passos, R. V. Oliveira, M. C. Marzochi y C. A. Andrade, “Systematic review of the adverse effects of cutaneous leishmaniasis treatment in the New World,” Acta Tropica, vol. 118, n.o 2, págs. 87-96, 2011, issn: 0001706X. doi: https://doi.org/10.1016/j.actatropica.2011.02.007. dirección: http://www.sciencedirect.com/science/article/pii/S0001706X11000325.

3. S. Duran, C. mendez, L. Mejia, C. Ovalle, C. Arenas, Y. Camargo, L. Rubiano y A. Cossio, “Lineamientos para la Atención Clínica Integral de Leishmaniasis en Colombia,” Ministerio de Salud y Protección Social, abr. de 2018. dirección: https://www.minsalud.gov.co/sites/rid/Lists/BibliotecaDigital/RIDE/VS/PP/PAI/Lineamientos-leishmaniasis.pdf.

4. Y. G. Chen, A. T. Satpathy y H. Y. Chang, “Gene regulation in the immune system by long noncoding RNAs,” Nature Immunology, vol. 18, n.o 9, págs. 962-972, sep. de 2017, issn: 1529-2916. doi: 10.1038/ni.3771. dirección: https://doi.org/10.1038/ni.3771.

5. G.-T. Cabetas, “Inmunoparasitología de Leishmania,” Facultad de Farmacia Universidad Complutense de Madrid, 2017, Accedido: 2020-10-30. dirección: http://147.96.70.122/Web/TFG/TFG/Memoria/MACARENA%20GARCIA-TREVIJANO%20CABETAS.pdf.

6. (Ene. de 2015). “What is gene expression? | Facts | yourgenome.org.” Accedido: 2020-11-01, dirección: https://www.yourgenome.org/facts/what-is-gene-expression.

7. C. A. Maher, C. Kumar-Sinha, X. Cao, S. Kalyana-Sundaram, B. Han, X. Jing, L. Sam, T. Barrette, N. Palanisamy y A. M. Chinnaiyan, “Transcriptome sequencing to detect gene fusions in cancer,” Nature, vol. 458, 7234 mar. de 2009, issn: 0028-0836. doi: 10.1038/nature07638.

8. R. Aguilar, L. Alonso, V. López y M. N. Moreno, “Incremental discovery of sequential patterns for grammatical inference,” 2005, págs. 59-67. dirección: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84877599894&partnerID=40&md5=4 2840983be1dbd081c619fe3fd259ae.