Proceedings of the International Joint Conference IBERAMIA/SBIA/SBRN 2006 - 3rd Workshop on MSc dissertations and PhD thesis in Artificial Intelligence (WTDIA'2006), Ribeirão Preto, Brazil, October 23–28, 2006. CD-ROM. ISBN 85-87837-11-7

Programación Lógica, Inductiva y Progresiva de Modelos Genómicos

(Tesis en curso para optar al titulo de Dr. en Ciencias Aplicadas, ULA, Vzla.)

Autor: José H. López Prato.
Afiliación: Laboratorio de Computación de Alto Rendimiento, Universidad del Táchira, San Cristóbal, Táchira, Venezuela. jlopez@unet.edu.ve

Supervisor: Jacinto Dávila
Afiliación: CESIMO: Centro de Simulación y Modelos, Fac. Ingeniería. Universidad de Los Andes, Mérida, Venezuela, jacinto@ula.ve.

Biologo asesor: Alejandro Mujíca
Afiliación: CeCALCULA: Centro Nacional de Cálculo Científico, Parque Tecnológico. Mérida, Venezuela, alejo@ula.ve.

Resumen

El objetivo general del proyecto que aquí se propone es explorar el sentido del aprendizaje simbólico y automático en un dominio del conocimiento donde privan las excepciones a las reglas establecidas, como es el caso de la genómica. En este contexto, el uso de la programación lógica inductiva ha permitido corroborar la pertinencia del aprendizaje simbólico y automático en el desarrollo de modelos predictivos, para proponer o predecir la posible estructura de genes. En este sentido, se presentan los resultados obtenidos en un trabajo anterior, acompañados de un plan de trabajo destinado a identificar, y exponer, los fundamentos que deben ser ajustados en el formalismo de aprendizaje empleado. Esto con la finalidad de mejorar el manejo de las excepciones y los efectos sistémicos pertinentes al área de estudio. Se presentan, de manera muy breve, los fundamentos relacionados con el desarrollo de procesos de aprendizaje automático basados en programación lógica inductiva (ILP); para ello se expone un modelo de experimento que incluye el modo en que se evalúan los resultados obtenidos. De igual manera se presenta una aproximación a la estructura general de los elementos fundamentales de los genomas: Los Genes. Existen un conjunto de criterios biológicos que guían la identificación de los sitios que delimitan las secciones que constituyen un gen, secciones conocidas como exones e intrones. Tales sitios, conocidos como sitios de inicio, splicing y terminación, definen dónde empieza y dónde termina un gen, además de dónde empiezan y terminan los intrones y exones que lo constituyen. Los criterios a los que se hace mención, nos han permitido organizar experimentos de aprendizaje según una metodología ad hoc, que propone el desarrollo de cuatro etapas de aprendizaje para la construcción de un predictor de genes. Este trabajo presenta el modo en que se desarrollaron las dos primeras etapas e indica el objetivo a lograr en el desarrollo de las dos restantes. Se presenta un ejemplo, a modo ilustrativo, del tipo de análisis que es posible realizar gracias a las descripciones estructurales logradas. Finalmente, se expone la hipótesis que orienta la presente propuesta y los objetivos que serán abordados a lo largo del trabajo.

Palabras clave: Programación Lógica, Aprendizaje Automático, Bioinformática, Genes, *Splicing*, *ILP*.

Nivel: Phd. **Fecha de conclusión**: Enero 2009
Scripts ILP disponibles en: http://sourceforge.net/projects/simulants

1. Introducción

La dificultad asociada a la definición de la estructura de un gen y su posible funcionalidad está relacionada con varios aspectos. Los genes eucariotas codificantes de proteínas están divididos en Intrones y Exones (ver Fig. 1). Para identificar éste tipo de genes es esencial identificar los límites exon/intron. Existen señales que marcan sistemáticamente la ubicación de estos bloques en la secuencia. Sin embargo, las mismas señales se **repiten** con frecuencia y en otras ubicaciones en donde no existen evidencias experimentales de su funcionalidad (falsos positivos). Por otro lado algunas señales con funcionalidad comprobada se escapan de los predictores actuales (falsos negativos). Esto indica que existen falsos inicios o falsas terminaciones que dificultan la predicción de la estructura de los genes. Se han desarrollado métodos más elaborados que el simple rastreo de esas "marcas" para establecer la estructura real de un gen (Padgett & Burge, 2005; Salzberg et al, 1998). Existen predictores basados en modelos de Markov que proponen la estructura general de genes eucariotas (Majoros et al, 2004; Pertea et al, 2001; Burge, 1998). Otras aproximaciones basadas en estudios de correlación, detección de señales y alineamientos, generan modelos para zonas específicas (e.g. sitios de splicing o exones) que luego pueden ser empleados como submodelos

de predictores mas generales (Hsieh et al 2005; Wang et al 2004; Arita et al 2002). A diferencia de esos trabajos, nuestro enfoque propone la construcción de descripciones estructurales que se descubran y originen desde las secuencias de ADN, usando procesos inductivos para la generación de reglas. Nuestra meta es generalizar desde las secuencias las evidencias o criterios que en la actualidad se emplean para validar o corregir estructuras genéticas propuestas *in silico* o *in vivo*. Nuestra metodología propone una aproximación *bottom-up* para reconstruir tales evidencias y experimentar posibles generalizaciones y ajustes de las mismas.

A grosso modo se puede decir que exones e intrones son transcritos a ARN, pero en el procesamiento a ARN mensajero maduro (mARN), los intrones son extraídos por "splicing" (Padgett & Burge, 2005; Wen-Hsiung, 1999; Lehninger, 1999; Rawn, 1989). Un marco de lectura contenido en los exones, en general el más extenso posible, lleva el código necesario para la síntesis de una proteína determinada. Por su naturaleza generalmente codificante, el tamaño y secuencia de exones homólogos tiende a conservarse evolutivamente, no así los intrones, cuyo tamaño y secuencia pueden variar grandemente por ser más tolerables a la acumulación de mutaciones y secuencias transposables. Sin embargo, las señales de splicing que determinan los límites exon/intron y que guían el proceso mismo de eliminación de intrones y concatenación de exones, están contenidas en la secuencia de todo gen, en particular dentro de los intrones y son comunes para todos los genes. De tal modo que Puede plantearse la siguiente Regla/Plantilla para caracterizar las triadas exon/intron/exon:

exon/GT-intron-AG/exon **(A)**

A su vez, el intrón puede detallarse del siguiente modo:

5'-AG/GTAAGT---intron----YNCTRAC------YnNAG/G-3' **(B)**

Donde "/" marca los límites exon/intron, Y es alguna pirimidina (T o C), Yn es un fragmento de ~10 pirimidinas, R es una purina (A o G). La A presente en la subcadena RAC (ver regla B) es una Adenina en algún lugar de la parte interna del intron, esencial para la reacción intermediaria de separación del intron. N es cualquier base. Toda la secuencia entre las dos barras inclinadas ("/ /") es un intron y el tamaño es muy variable. El problema tratado en este trabajo puede precisarse de la siguiente manera: El conjunto de reglas que rigen el proceso de splicing del ARN primario, se encuentra en la secuencia de ADN genómico y se aplica de manera dispersa a lo largo de cada gen. Tal conjunto de reglas está parcialmente definido y se compone de conocimiento inferido mediante métodos estadísticos validado (en algunos casos) experimentalmente.

2. Trabajos relacionados

El presente trabajo propone un modo de estudio alternativo basado en aprendizaje automático (Ian y Eibe, 2005; Mitchell, 1997) y el método empleado es la programación lógica inductiva o Inductive logic Programming ILP (Muggleton y Firth 2003; Muggleton, 1995). La programación ILP ha sido empleada en diversos problemas relacionados con la bioinformática que incluyen predicción de estructuras secundarias de proteínas, predicción de carcinogenicidad y sistemas metabólicos (Tamaddoni et al, 2006; Muggleton, 2005; Turcotte et al, 2001; Turcotte et al, 1998; Finn et al, 1998; Srinivasan et al, 1997). El análisis de regularidades en secuencias de ADN empleando reglas lógicas o ILP ha sido reportado en otros trabajos. Kolchanot et al (2002) presentan un sistema que usa reglas lógicas ponderadas probabilísticamente para describir promotores y discriminar la funcionalidad de los genes asociados. Wren et al (2005) reportan el uso de árboles de decisión para descubrir y organizar reglas acerca de características compartidas entre genes, exones, *repeats* e islas CpG, de un mismo genoma o entre genomas diversos. King et al (2004) emplean ILP y árboles de decision para predecir la funcionalidad asociable a marcos de lectura. Badea (2003) emplea ILP para discriminar la funcionalidad de genes estudiados mediante *microarrays*, expresando las reglas inherentes en términos de una ontologia. Keles et al (2004) analizan mediante regresión lógica correlaciones entre los sitios de enlace en promotores asociados a factores de transcripcion y sus efectos en la regulacion de los genes correspondientes. Estos trabajos indican que ILP ha sido y está siendo empleada en el análisis de secuencias. Sin embargo la generalización de reglas para proponer la estructura de un gen por medio de ILP parece no haber sido abordada aún. Lo mas cercano, después de una revisión detallada, es la predicción de la funcionalidad asociable a marcos de lectura o genes. Hasta ahora parece no existir un trabajo basado en ILP, organizado según los objetivos y la metodología que aqui presentamos.

3. Enfoque para la construcción de modelos genómicos

Nuestro trabajo está dirigido a la construcción de experimentos de aprendizaje automático organizados por niveles capaces de producir representaciones de conocimiento que puedan ser jerarquizadas y manipuladas por biólogos. El

tipo de reglas descubiertas deben describir desde regularidades básicas tales como posibles secuencias conservadas o regularidades aún más generales para la descripción de sitios de splicing. Tales reglas son minadas y evaluadas empleando motores de inferencia provenientes de la programación logica inductiva, cuya eficiencia y confiabilidad ha sido probada previamente. Nuestra metodología *ad hoc* procura el desarrollo de niveles de aprendizaje organizados por capas para descubrir e interrelacionar las características estructurales para la descripción de genes. Debe entenderse que cada nivel recibe conceptos aprendidos en el nivel inmediatamente anterior. Para lograr lo anterior se organizan experimentos de aprendizaje para cada capa empleando criterios biológicos bien establecidos, acerca de las características que generalmente están asociadas a cada uno los conceptos listados. Tales criterios permiten la definición de una base de conocimiento inicial descrita mediante reglas lógicas. Este documento describe los primeros resultados logrados para las dos primeras capas de aprendizaje.

3.1 Tipo de criterios a-priori, empleados para la caracterización de sitios de splicing

Nuestros ejemplos y contraejemplos se han extraído de la reanotación del cromosoma 2 del *P. falciparum* realizada por (Huestis y Fisher, 2001). A continuación algunos de los criterios empleados para el estudio (Huestis y Fisher, 2001; Wen-Hsiung, 1999; Lehninger, 1999; Rawn, 1989).

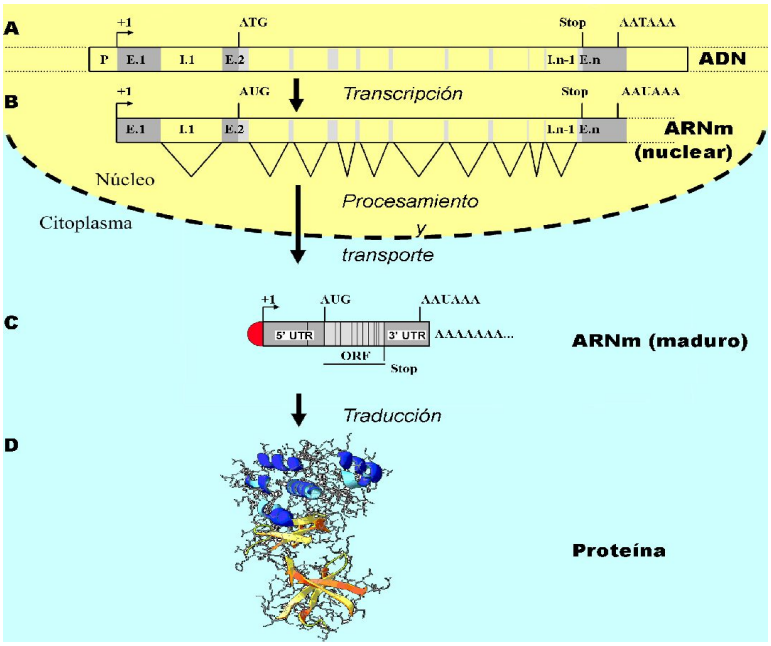

Fig 1. Estructura general y expresión de un gen eucariota codificante de Proteínas.

A) Una región de ADN genómico dada, de longitud indeterminada (líneas punteadas), contiene un gen con un número N de exones (E.1,…,E.n) separados por un número N-1 de intrones (I.1,..I.n-1). B) La región promotora P determina el inicio de transcripción (+1) que resulta en el ARN mensajero nuclear (o primario) con exones e intrones. B->C) El ARN es procesado: los intrones son escindidos (splicing) representado aquí por líneas diagonales, en el extremo 5' se añade la estructura CAP (semicírculo) y una señal de poliadenilación (AAUAAA) determina un corte a unas 25 bases aguas abajo y la adición de una cola de adeninas (AAAAAA…). El ARN mensajero maduro (ARNm) es transportado al citoplasma. Flanquedo por las regiones no traducibles (UTR) y delimitado por las señales de inicio (AUG) y parada (UAA, UGA o UAG), se encuentra un marco abierto de lectura (ORF) que codifica la información para la síntesis (traducción) de una proteína (D). Las señales de inicio de lectura, de parada, de poliadenilación así como los límites exón/intrón son comunes para todos los genes, no así la información contenida en el ORF, el inicio de transcripción, el número de exones e intrones ni su tamaño.

1. Algunos nucleótidos suelen mostrar cierta distribución en el interior del intron. Las Aes tienden a estar hacia el extremo 5´ del intron, las T´s hacia el extremo 3´, y la zona poly-RY hacia el centro.
2. La repetición de dinucleótidos es típica en intrones. Todas las formas [RY]n con n>5 generalmente se asignan a intrones.
3. Cuando ocurre un sitio de splicing GTGT éste suele asociarse al primer dinucleótido. Esto se decide así puesto que es extraño que un exon finalice con GT.
4. La zona de enlace o ramificación es una zona bastante cargada de T´s que suele ser interrumpida por A, C o G y

que suele ubicarse a unas 40 b "aguas arriba" del sitio de splicing AG.
5. Una región rica en bloques AT precedida de una señal GT es considerada parte interna de un intron. Las zonas GT[AT]n generalmente ocurren bastante después del sitio de splicing GT, dado que la zona [AT]n ocurre hacia el centro del Intron y no al principio.
6. Una señal GT seguida de A o AA es un sitio de splicing bastante probable. GTC al contrario, suele ser un sitio de splicing falso. Los sitios de splicing GTAA y GTA ocurren aproximadamente dos terceras partes de las veces en que se detecta una señal GT válida.
7. Cuando hay varias señales GT agrupadas se da preferencia a aquellas precedidas por G o A, dado que son sitios de splicing más probables; sin embargo, tal decisión no debe sacrificar zonas de ADN con marcada tendencia GC.
8. Una señal AG verdadera suele estar precedida por una citosina, una base cualquiera y el inicio de una la zona rica en pirimidinas seguida de la zona de ramificación. El sitio candidato AG suele estar seguido de una guanina.
9. Un sitio de splicing AG es poco probable cuando tiene otro sitio similar cercano aguas arriba.
10. El splicing alternativo se reconoce cuando hay más de una manera de escindir un intron. Esto implica la presencia de más de una señal GT o AG positivas.
11. Las zonas codificadoras suelen ser más ricas en GC que las no codificantes.
12. Las zonas codificantes suelen tener codones particulares que suelen repetirse. En particular GAA, GAT, AAT.

3.2 Uso de un motor de inferencia inductiva. Progol.

Progol es un sistema inductor de reglas que opera sobre el espacio de modelos, o reticulado de hipótesis, asociado a una teoría lógica (Hogger, 1990). El sistema toma como entrada una teoría con el conocimiento establecido **B, y** los ejemplos $\mathbf{E^+}$**,** y contraejemplos $\mathbf{E^-}$**,** pertinentes al concepto que se desea generalizar. En general, cualquier experimento de aprendizaje basado en Progol, requiere que se definan los siguientes componentes: Teoría parcial (o conceptos); Definiciones estructurales (modos y tipos); Restricciones y Podas; Ejemplos y Contraejemplos. La herramienta emplea tres etapas para realizar sus procesos de aprendizaje 1) Construcción de la cláusula más específica, 2) Construcción y recorrido del reticulado de hipótesis y 3) Ejecución del algoritmo de cubrimiento (Muggleton, 1995). A continuación una breve descripción del modo en que Progol se emplea en el presente trabajo.

3.2.1 Construcción de la cláusula más específica

Para explicar el funcionamiento del proceso de inducción se emplea el concepto sitio de splicing GT**,** *sitio_splicing_gt(*`Bases, Señal`*),* destinado describir sitios de splicing del tipo indicado. Tal concepto debe ser capaz de recibir una secuencia de `Bases` y la `Señal` que se desea estudiar. La idea es generalizar reglas para el concepto tales que, dados los argumentos anteriores, se pueda responder si la secuencia posee o no un sitio de splicing GT. Para generalizar (aprender, descubrir) el concepto *sitio_splicing_gt*, Progol emplea los criterios para caracterizar sitios de splicing GT expuestos en la sección 3.1; por ejemplo, el uso de las transiciones de zonas no codificantes a zonas que si lo son. También existen reglas que permiten descartar posibles sitios de splicing; por ejemplo, está bien establecido que aquellos sitios de splicing GT precedidos de otros sitios cercanos suelen ser pobres candidatos (sección 3.1, ítem 5). Para manejar los criterios anteriores se programaron conceptos como: *`trans_AT/4`* y *`trans_GC/4`* que determinan si hay transiciones importantes de contenido AT o GC en un sitio determinado; *`psc_gt/3`*, que establece si un sitio determinado satisface o no la distribución de posibles secuencias conservadas aprendida en la etapa 1. En el caso de los sitios de spling AG, también se incorpora el concepto *`zona_rica_pirimidinas/4`*, aprendido en la primera etapa. Además de tales criterios, se incorporan restricciones del tipo: *`no_gt_vec_prec_ag/2`*, que valida que el sitio GT candidato no presente otro sitio GT vecino precedido por A o G; *`no_gt_prec_gt/2`*, que *d*etermina si el sitio GT en estudio no está precedido por otra señal GT y `no_ag_vecino/2` , que determina si el sitio AG en estudio no está precedido por otra señal AG vecina. La tabla III resume los resultados obtenidos durante esta etapa. En el caso del sitio de splicing GT, Progol aplica los criterios correspondientes al primer ejemplo positivo suministrado determinando el modo en el que estos están presentes (incluyendo posibles replicaciones y variaciones en su aplicación). Esa primera regla es conocida como la claúsula mas específica o MSC, Most Specific Clause (Muggleton, 1995). La MSC pasa luego a ser depurada mediante la construcción de un reticulado de posibles hipótesis que se generan a partir de ella, según se explica a continuación.

3.2.2 Construcción y recorrido del reticulado de hipótesis

La cláusula más específica debe ser optimizada. La finalidad es determinar si existe una manera más eficiente de organizar los conceptos de la regla. Para ello se recorre un reticulado de reglas alternativas (vistas como subconjuntos de la MSC). El recorrido del reticulado permite eliminar la presencia de cualquier concepto redundante, establecer un mejor orden de los conceptos y evaluar la calidad de las alternativas propuestas en la medida en que se van definiendo éstas. Para cada hipótesis se chequean las restricciones establecidas (si acaso existen). Si alguna

hipótesis satisface alguna restricción, entonces el sistema regresa hasta el punto en el que la restricción deja de ser válida e incorpora otro componente de la MSC, creando una nueva regla alternativa para explorar.

3.2.3 Aplicación del algoritmo de cubrimiento

Este algoritmo se encarga de evaluar la totalidad de los ejemplos positivos y determina cuáles ejemplos son cubiertos mediante una hipótesis recién propuesta por el inductor. Cuando una hipótesis califica como regla, cada ejemplo que la satisface es extraído del conjunto inicial de ejemplos. En resumen: Tomado el primer ejemplo, primero se define una MSC a partir de éste, partiendo de la MSC se recorre un reticulado de hipótesis, se evalúa cada una de ellas para hallar la mejor y se determinan los ejemplos positivos y negativos que son cubiertos por la misma. Una vez determinada la mejor hipótesis, los ejemplos son extraídos del total de ejemplos positivos suministrados. Progol, procede, de allí en adelante, a tomar el primer ejemplo positivo disponible y repite todo el proceso nuevamente. Esto se repite hasta que hayan sido generadas todas las reglas necesarias para implicar el total de los ejemplos positivos, procurando siempre cubrir la menor cantidad de ejemplos negativos (en caso de que se permita tal flexibilidad). Aplicados los algoritmos anteriores, Progol aprende dos reglas para describir el concepto *sitio_splicing_gt* (reglas C y D). Se aprecia como la descripción descubierta emplea el concepto `pscgt_gt` aprendido en la primera etapa, cuya descripción queda expresada mediante las reglas E, F y G.

```
sitio_splicing_gt(Secuencia,Senal) si
                                  trans_AT(Secuencia,Posicion,Senal,'no_cod->cod').   (C)

sitio_splicing_gt(Secuencia,Senal) si pscgt_gt(Secuencia,Senal,izq).                  (D)
```

Interpretacion: Una **Secuencia** de ADN incluye un sitio de splicing GT *si*, en cierta posición de la misma se detecta una señal GT que presenta una transición AT del tipo *no codificante* a *codificante* (regla C); *o*, la secuencia posee una distribución de posibles bases conservadas "aguas arriba o hacia la izquierda" del sitio GT, según describe el concepto `pscgt_gt` (regla D).

```
pscgtup(Sequencia,Senal,Direccion) si                                                (E)
                zona_es(Direccion,Sequencia,Senal,Posicion,Zona) y
                 b(r,Zona,Rest_Zona) y b(a,Rest_Zona,Nuevo_Rest_Zona).
pscgtup: Posibles secuencias conservadas "aguas arriba o hacia la izquierda" en un sitio de splicing GT

pscgtdown(Sequencia,Senal,Direccion) si
                zona_es(Direccion,Sequencia,Senal,Posicion,Zona) y
                 b(a,Zone,Rest_Zona) y b(a,Rest_Zona,Nuevo_Rest_Zona).
                                                                                     (F)
pscgtdown(Sequencia,Senal,Direccion) si
                zona_es(Direccion,Sequencia,Senal,Posicion,Zona) y
                 b(a,Zone,Rest_Zona) y b(y,Rest_Zona,Nuevo_Rest_Zona),
                 b(a,Nuevo_Rest_Zona,Ultimo_Rest_Zona).
pscgtdown: Posibles secuencias conservadas "aguas abajo o hacia la derecha" en un sitio de splicing GT   (G)
```

Interpretacion regla E: Una Secuencia de bases incluye un sitio de splicing GT *si*, en una delimitada Zona que incluye una senal GT, una base purina r es seguida por una base adenina a "aguas arriba" respecto de la senal GT.

3.2.4 Presentación de resultados

La tabla II muestra el resultado final en el análisis de PSC en sitios GT para el cromosoma 2 del *P. falciparum*. Las reglas halladas fueron evaluadas por el mismo inductor mediante un estudio estadístico, basado en tablas de contingencia, que comparan predicciones (apoyadas en las reglas) con una clasificación aleatoria basada en una distribución probabilística uniforme. Los valores Chi-square = `24.64` y Chi-square probability = `0.0000`, estiman que el resultado está muy distante a una simple clasificación azarosa. Puede observarse en la columna A de la tabla de contingencia que se manejaron 39 ejemplos positivos mientras que la columna ~A muestra un total de 36 ejemplos negativos. Las filas P y ~P especifican cuántos del total de ejemplos son clasificados como positivos y negativos. Puede verse que las reglas logran clasificar adecuadamente a 33 de los 39 ejemplos positivos y agregan allí (erróneamente) 9 de los 36 negativos suministrados. También puede notarse que las reglas, desafortunadamente, no logran clasificar a 6 de los positivos, mientras que 27 de los negativos son correctamente clasificados. Los ejemplos evaluados corresponden aproximadamente a un 30% de los sitios GT verdaderos reportados en (Huestis y Fisher, 2001). El 70% restante se empleó en el proceso de aprendizaje de reglas. El inductor presenta estadísticos que permiten ponderar la precisión de las reglas. Estos son los valores C: 83, 98, 24, 0 `(ver tabla)` que pueden reescribirse como `C: f, p, n, h` siendo 'f' el parámetro que mide lo precisa que es la regla. 'f' es una medida del poder predictivo y de compresión de cada regla hallada. Los parámetros `p` y `n` corresponden al número de ejemplos positivos y negativos cubiertos por la regla mientras que `h` indica el número de conceptos que aún deben probarse

para considerar completa una regla. La Chi cuadrado mide cuán aleatorio es el conjunto de resultados predichos por las reglas que se someten a evaluación (estadístico Chi-square) y qué tan probable es que esos resultados equivalgan a selecciones al azar (estadístico Chi-square probability).

3.3 Experimentos

Nuestro trabajo esta organizado por etapas de aprendizaje de las cuales se han desarrollado dos. Para cada experimento se organizaron archivos de entrenamiento y archivos de evaluación. Los primeros se obtuvieron de las anotaciones del cromosoma 2 del *P. falciparum*, suministradas por (Huestis y Fisher, 2001). Para el caso de ejemplos negativos se optó por dos aproximaciones: La primera, generar ejemplos tomando señales falsas detectadas en zonas intergénicas. Para la segunda, ubicada una señal ATG verdadera, por ejemplo, se programaron *scripts* que desplazan la ubicación del sitio de inicio ya sea hacia la derecha o hacia la izquierda. Los ejemplos y contraejemplos se dividieron en dos conjuntos de datos destinados a entrenamiento y evaluación, tomando 70% y 30% de los casos disponibles para cada fin.

3.3.1 Primera etapa de aprendizaje

La primera etapa exploró el aprendizaje de reglas para posibles secuencias conservadas y las zonas de ramificación y la zona rica en pirimidinas. En el primer caso se proponen distribuciones de bases para los sitios de splicing. Para las zonas se proponen bandas de posicionamiento (ver tablas I y II).

3.3.2 Segunda etapa de aprendizaje

En esta etapa se incorporan varios de los criterios para caracterizar los sitios de splicing (GT/AG), expuestos en la sección 3.1. La organización de los experimentos relativos a esta etapa se explica en la sección 3.2.1. La tabla III resume los resultados obtenidos.

```
Aguas arriba:
      C:83,98,24,0  pscgtup(Sequencia,Senal,Direccion) si
                            zona_es(Direccion,Sequencia,Senal,Posicion,Zona) y
                            b(r,Zona,Rest_Zona) y b(a,Rest_Zona,Nuevo_Rest_Zona).
Aguas abajo:
Regla 1:
      C:89,72,12,0  pscgtdown(Sequencia,Senal,Direccion) si
                            zona_es(Direccion,Sequencia,Senal,Posicion,Zona) y
                            b(a,Zone,Rest_Zona) y b(a,Rest_Zona,Nuevo_Rest_Zona).
Regla 2:
     C:16,8,1,0   pscgtdown(Sequencia,Senal,Direccion) si
                            zona_es(Direccion,Sequencia,Senal,Posicion,Zona) y
                            b(a,Zone,Rest_Zona) y b(y,Rest_Zona,Nuevo_Rest_Zona),
                            b(a,Nuevo_Rest_Zona,Ultimo_Rest_Zona).
```

Contingency table (pcsgtup/3)=

	A	~A	
P	33 (21.8)	9 (20.2)	42
~P	6 (17.2)	27 (15.8)	33
	39	36	75

Overall accuracy	Chi-square
80.00% +/- 4.62%	24.64
79.44% +/- 3.91%	33.33

Contingency table (pcsgtdown /3)=

	A	~A	
P	32 (17.2)	14 (28.8)	46
~P	8 (22.8)	53 (38.2)	61
	40	67	107

Without Yates correction	Chi-square probability
27.00	0.0000
35.70	0.0000

Tabla I._ Posibles secuencias conservadas en sitios de spling GT

3.4 Resultados

3.4.1._ Primera etapa

La primera etapa de aprendizaje se ha centrado en el estudio de PSC, zonas ricas en pirimidinas, y de ramificación. Los dos primeros definen criterios de decisión importantes mientras que el último resulta muy pobre en la identificación de los sitios de interés. Sin embargo el motor de inferencia propone posibles bandas de posicionamiento para ambas zonas que se corresponden con lo reportado. El análisis de PSC valida varias de las regularidades conocidas pero acompañadas de otras no reportadas. Esto debe cotejarse con cuidado con lo reportado en otros trabajos, en particular los referentes a eucariotas superiores. En otros casos, se validan los consensos reportados pero existen simultáneamente otras reglas que describen otros patrones de regularidad más eficientes según se lee en el estadístico f asociado a cada regla.

Aguas arriba:
C:54,57,12,0 pscagup(Sequencia,Senal,Direccion) *si*
zona_es(Direccion,Sequencia,Senal,Posicion,Zona) *y*
b(**y**,Zone,Rest_Zona) *y* b(**n**,Rest_Zona,Nuevo_Rest_Zona),
b(**y**,Nuevo_Rest_Zona,Ultimo_Rest_Zona).

Aguas abajo:

Regla 1:
C:66,18,3,0 pscagdown(Sequencia,Senal,Direccion) *si*
zona_es(Direccion,Sequencia,Senal,Posicion,Zona) *y*
b(**g**,**Zone**,Rest_Zona) *y* b(**t**,Rest_Zona,Nuevo_Rest_Zona).

Regla 2:
C:1,6,1,0 pscagdown(Sequencia,Senal,Direccion) *si*
zona_es(Direccion,Sequencia,Senal,Posicion,Zona) *y*
b(**r**,Zone,Rest_Zona) *y* b(**g**,Rest_Zona,Nuevo_Rest_Zona).

Contingency table (cagi/3)=

	A	~A	
P	36 (21.5)	7 (21.5)	43
~P	4 (18.5)	33 (18.5)	37
	40	40	80

Contingency table (cagd/3) =

	A	~A	
P	8 (4.0)	0 (4.0)	8
~P	32 (36.0)	40 (36.0)	72
	40	40	80

Overall accuracy	Chi-square	Without Yates correction	Chi-square probability
86.25% +/- 3.85%	39.42	42.29	0.0000
60.00% +/- 5.48%	6.81	8.89	0.0091

Tabla II._ Posibles secuencias conservadas en sitios de spling AG

3.4.2 Segunda etapa

En este caso se señalan la transición de contenidos AT y la distribución de PSC a la izquierda de los sitios GT como las reglas discriminatorias más sólidas. En la primera regla se esperaría una transición del tipo codificante a no codificante, sin embargo la transición descubierta es del tipo `'no_cod->cod'`. Lo que la primera regla establece en este caso es que al final del exon el contenido AT es mas bajo que el correspondiente al inicio del intron. Se aclara que el estudio de las proporciones se hace considerando solo varias decenas de bases en cada dirección. El valor de Chi cuadrado es bastante alto. Sugiere que el par de conceptos presentados en las reglas tienen un peso muy importante en la identificación de sitios de splicing GT. Pueden observarse tres reglas para identificar el sitio de splicing AG. Primero, se establece una zona rica en pirimidinas y la ausencia de otra señal AG vecina aguas arriba. En segundo lugar, se observa que es usual detectar transiciones GC `'no_cod->cod'`. Por último, se establece la presencia de transiciones AT alto a bajo (`'cod->no_cod'`), en conjunción con la distribución de conservados a la derecha del sitio AG aprendida en la primera etapa. El valor Chi cuadrado y la probabilidad correspondiente definen un nivel alto de confianza en el resultado.

3.5 Conclusión etapas 1 y 2

Hemos creado una plataforma computacional, basada en el inductor Progol, para identificación de estructuras genéticas, a partir de datos publicados del *P. falciparum*. Dada una descripción formal de ejemplos positivos y negativos y una base de conocimiento inicial, la plataforma produce reglas lógicas que describen regularidades estructurales. A continuación la descripción de los aspectos que justifican nuestra propuesta de tesis doctoral, partiendo de la estrategia y resultados hasta ahora descritos.

4. Motivación para el desarrollo de la propuesta.

Los experimentos que se resumen la sección 3 provenientes de (López, 2004), demuestran que una metodología ad hoc, fue capaz de conducir resultados significativos estadísticamente, aun cuando el formalismo de aprendizaje es meramente simbólico (no estocástico) y esa metodología no fue fundada sobre la tradición de aprendizaje en biología, caracterizada por la excepcionalidad, sino mas bien sobre la tradición lógica constructiva, reduccionista y categórica. *El problema al que nos enfrentaremos en este trabajo es el revisar esas fundaciones y tratar de reconciliarlas con esa tradición biológica caracterizada por la expresión "Toda regla tiene una excepción"*. Esto sin renunciar a las virtudes técnicas que ya ha mostrado el formalismo de la programación lógica inductiva. Se requiere extender las dos etapas iniciales y desarrollar dos más, objeto de desarrollo de la presente propuesta. Las cuatro etapas persiguen desarrollar modelos genómicos y sus objetos de estudio se resumen en la tabla IV.

sitio_splicing_gt(Sequencia,Senal, Posicion) *si* trans_AT(Sequencia,Posicion,Senal,'no_cod->cod').

sitio_splicing_gt(Sequencia,Senal, Posicion) *si* pcsgt_gt(Sequencia,Posicion,Senal,izq).

sitio_splicing_ag(Sequencia,Senal, Posicion) *si*

zona_rica_pirimidinas(Sequencia,Posicion,Senal,izq) y no_ag_vecino(Sequencia,Posicion,Senal).

sitio_splicing_ag(Sequencia,Senal, Posicion) *si* trans_GC(Sequencia,Posicion,Senal,'no_cod->cod').

sitio_splicing_ag(Sequencia,Senal, Posicion) *si*

trans_AT(Sequencia,Posicion,Senal,'cod->no_cod'),conservados_ag(Sequencia,Posicion,Senal,der).

Contingency table (sitio_splicing_gt/2)=

	A	~A	
P	35 (18.1)	0 (16.9)	35
~P	10 (26.9)	42 (25.1)	52
	45	42	87

Contingency table (sitio_splicing_ag/2)=

	A	~A	
P	31 (18.3)	3 (15.7)	34
~P	4 (16.7)	27 (14.3)	31
	35	30	65

Overall accuracy	Chi-square	Without Yates correction	Chi-square probability
88.51% +/- 3.42%	51.47	54.65	0.0000
89.23% +/- 3.84%	36.89	39.98	0.0000

Tabla III._ Resultados generalización conceptos sitios de splicing GT y AG.

Nivel de aprendizaje	***Conceptos a generalizar e interrelacionar***
I	Posibles secuencias conservadas, zona de ramificación, zona Rica en pirimidinas, recurrencia de *repeats*, zona de poliadenilación, transición de contenido AT y GC, caracterización de promotores y habilitadores (enhancers), Otros.
II	Sitios de inicio, sitios de *splicing*, sitios de parada, sitios de splicing alternativo, sitios de inicio alternativos.
III	Intrones, Exones
IV	Genes

Tabla IV._ Niveles de aprendizaje automatico para descubrir modelos genómicos.

4.1. Tercera etapa de aprendizaje

La tercera etapa de aprendizaje propone la generalización de reglas para la definición de predictores de intrones y exones (conceptos *intron* y *exon*). En el caso de la generalización del concepto *intron* se deben suministrar tripletas o bloques del tipo exon-intron-exon, solicitando la generalización de las reglas. Como es de esperar, las reglas halladas para los cuatro sitios de interés en las etapas uno y dos deben incorporarse. En el caso de los exones, se deben emplear bloques del tipo promotor/UTR-exon-intron e intron-exon-intron, procediendo de manera similar. La tercera etapa del proceso de aprendizaje debe dar paso a reglas del tipo:

> Una secuencia dada contiene un intron entre una posición inicial y otra posición final, si la secuencia contiene una "verdadera señal GT" que define la posición inicial del intron y contiene una "verdadera señal AG" cuya posición delimita su posición final.

El detalle es determinar cuándo se está en presencia de una "verdadera señal". Para ello las reglas deben contar con definiciones que permitan determinar si ese es el caso o no, aprovechando, desde luego, los conceptos aprendidos en etapas previas de aprendizaje, según la estrategia esbozada en la sección 3. (López, 2004, para mayores detalles).

En una secuencia de ADN, que contiene un gen y que posee más de un sitio de splicing puede existir más de una manera válida de describir la estructura del gen hallado, debido a que existen maneras alternas de cortar sus intrones y exones (*splicing alterno).* Este es un concepto muy elusivo en biología molecular, sobre el cual se han propuesto varias caracterizaciones (Majoros et al, 2004; Pertea et al, 2001; Burge, 1998), que muy probablemente sean elaboradas y re-elaboradas en los próximos años, debido a que es un problema aún no resuelto. Así que es un buen concepto de estudio para efectos de validar el formalismo con las nuevas elaboraciones. Por otro lado, está demostrado que pueden existir sitios de inicio y parada alternos. Es decir, otros conceptos pueden requerir el mismo tratamiento flexible al momento de ser representados. Como se ve, la complejidad (número de conceptos, reglas por conceptos, posibles excepciones) se incrementa en cada etapa de aprendizaje. En resumen, el objetivo la tercera

etapa es construir un predictor de intrones y exones; además de un predictor de sitios con potencialidad de *splicing* alterno.

4.2 Cuarta etapa de aprendizaje

La cuarta etapa del trabajo propone la construcción de un predictor de genes. En esta etapa los conceptos relacionados con los sitios de inicio, splicing, splicing alterno, parada, zona rica en pirimidinas, patrones conservados, entre otros, se integrarán a nuevos conceptos y restricciones que al final deben definir un predictor de estructuras. Lo expuesto indica que son diversos los procesos de aprendizaje que deben ser realizados. Los procesos de aprendizaje son graduales, escalonados y jerárquicos. Cada etapa del trabajo construye predictores y cada uno puede ser evaluado en cuanto a su capacidad predictiva, empleando tablas de contingencia y pruebas de Chi-Cuadrado (Grant, 1999). Estas pueden ser comparadas con otras obtenidas mediante otros métodos de minería (e.g. HMMs, redes neuronales) o de aprendizaje automático (e.g. árboles de decisión). Compete el uso de herramientas especializadas en la predicción de sitios de splicing (Majoros et al, 2005; Arita et al 2002; Pertea et al, 2001), procurando simultáneamente mejoras en los procedimientos de evaluación (e.g. *x-validation*)(Ian y Eibe, 2005). Deben explorarse mecanismos para el aprendizaje que incorporen representaciones de la incertidumbre en las reglas. Esto es posible mediante lo que se conoce como programas lógicos estocásticos (Watanabe & Muggleton, 2005) (Muggleton, 2000)(Page, 2000). En nuestro caso esto podría permitir representar reglas y razonar con la incertidumbre inherente o asociable a los conceptos aprendidos en cada etapa. Por otro lado, es necesario determinar si los resultados son extensibles a otros cromosomas del mismo organismo realizando predicciones basadas en las reglas halladas. Compete además la determinación de modelos para otros organismos. Para desarrollar esto último se pueden tomar estructuras de genes validadas manual o semi-automáticamente disponibles en la comunidad científica (i.e. VEGA, Ensembl) (Searle SM et al 2004; Ashurst JL et al 2005; Ashurst JL and Collins JE 2003; Birney E. et al 2006). No obstante, las herramientas de procesamiento simbólico, como ILP, ofrecen otro criterio cualitativo: la capacidad para explicar lo aprendido. Las reglas generadas pueden ser usadas, por un humano, para explicarse el concepto, siempre y cuando, el formalismo y el lenguaje en que se escriben esas reglas sea suficientemente expresivo como para capturar la intuición biológica. Por esta razón, nos planteamos la siguiente hipótesis de trabajo:

5. Hipótesis de trabajo

"La lógica formal es suficientemente expresiva para permitir descripciones tolerantes a la elaboración del conocimiento genómico, aún considerando efectos sistémicos y excepcionales que pueden surgir en las teorías biológicas y en los datos genéticos".

Sobre esa base hipotética, planteamos el objetivo central de este proyecto doctoral:

6. Objetivo general

Crear mecanismos para producir modelos, descripciones y definiciones generales de estructuras genómicas, construidos mediante procesos progresivos de inducción que permitan formas de razonamiento por omisión abiertas a las excepciones.

7. Metodología

En general, la estrategia consiste en generar modelos de genes y validarlos contra biodata experimental. Si los modelos no corresponden, serán revisados incorporando excepciones a las reglas (el uso de programas lógicos estocásticos será cuidadosamente tratado en esta tarea). Iteraremos en este ciclo, con la ayuda del tutor en biología, hasta que los modelos alcancen un grado de elaboración biológicamente aceptable. La metodología seguida en la construcción de nuestros predictores estará determinada por dos aspectos fundamentales: 1) El aprendizaje por etapas, que se acumulan como capas en una estructura jerárquica de conocimiento y 2) El formalismo ILP, que requiere la programación lógica de conocimiento establecido y la indicación de aquello que debe ser generalizado. Tal como se expone en la sección 3 y en (López, 2004), existe un soporte de diseño, experimentación y de programación, que suministran el "por dónde y cómo continuar" la construcción de los predictores genómicos, que serán objeto de análisis desde el punto de vista del formalismo lógico. Esto último a fin abordar la hipótesis planteada en esta propuesta.

Referencias

Arita M, Tsuda K and Asai K. (2002); Modeling splicing sites with pairwise correlations, BIOINFORMATICS Vol. 18 Suppl. 2.

Ashurst JL et al (2005). The Vertebrate Genome Annotation (Vega) database. Nucleic Acids Res. Jan 1;33

Ashurst JL and Collins JE (2003). Gene annotation: prediction and testing. Annu Rev Genomics Hum Genet. 2003;4:69-88.

Badea L. (2003); FUNCTIONAL DISCRIMINATION OF GENE EXPRESSION PATTERNS IN TERMS OF THE GENE ONTOLOGY; Pacific Symposium on Biocomputing 8:565-576(2003).

Birney E. et al. Ensembl (2006). Nucleic Acids Res. 2006 Jan 1;34

Burge, C. B. (1998) Modeling dependencies in pre-mRNA splicing signals, Computational Methods in Molecular Biology, Vol.32, p. 127-163, Elsevier.

Finn P., S. Muggleton, D. Page, and A. Srinivasan (1998). Pharmacophore discovery using the Inductive Logic Programming system Progol. Machine Learning, 30:241-271.

Grant Gregory R ., Ewens Warren J (1999): Statistical Methods in Bioinformatics: An Introduction. Springer.

Hogger, Christopher Jhon (1990); Essentials of Logic Programming Clarendon Press. Oxford, 1990

Hsieha S., Chung Y., Lin C., Tang C. (2005); EXONSCAN: EXON Prediction with Signal Detection and Coding Region AligNment in Homologous Sequences; 2005 ACM Symposium on Applied Computing.

Huestis Robert, Fisher Katia (2001); Prediction of many exons and introns in Plasmodium Falciparum chromosome 2, Molecular & Biochemical Parasitololgy, Elsevier, p. 187-199.

Ian H. Witten, Eibe Frank (2005); Data Mining: Practical Machine Learning Tools and Techniques, Elsevier, 2005.

Keles S., J. van der Laan M., Vulpe C.; Regulatory motif finding by logic regression; BIOINFORMATICS, Vol. 20 no. 16 2004, pages 2799–2811.

King R., Wise P., and Clare A.; Confirmation of data mining based predictions of protein function; BIOINFORMATICS, Vol. 20 no. 7 2004, pages 1110–1118.

Kolchanov N., Pozdnyakov M., Orlov Y., Vishnevsky O., Podkolodny N. (2002); Computer System "Gene Discovery" for Promoter Structure Analysis; In Silico Biology, Issue: Volume 2, Number 3 / 2002, Pages: 257 - 262

Lehninger, Albert L (1999), Principles of biochemistry, Worth Publishers.

López José (2004), Programación Lógica Inductiva en la exploración de regularidades en secuencias de ADN, Centro de Investigación y Proyectos en Simulación y Modelos. Universidad de Los Andes. Trabajo presentado para optar al titulo de Magíster en Computación, Postgrado de Computación, ULA, Venezuela.

Majoros W. H., Pertea M. and Salzberg S. L. (2004); TigrScan and GlimmerHMM: two open source ab initio eukaryotic gene-finders, Bioinformatics Volume 20, Number 16 Pp. 2878-2879

Muggleton S.H. (2005); Machine learning for systems biology. In Proceedings of the 15th International Conference on Inductive Logic Programming, LNAI 3625, pages 416-423. Springer-Verlag, 2005.

Muggleton Stephen, Firth Jhon (2003); Cprogol4.4: A tutorial introduction, Computational Bioinformatics Laboratory, Department of Computing, Imperial College, London, United Kingdon. (http://www.doc.ic.ac.uk/~shm/Software/progol4.4/)

Muggleton, S.H. (1995). Inverse entailment and Progol. New Generation Computing, 13:245-286.

Muggleton, Stephen (2000). Learning stochastic logic programs. In Proceedings of the AAAI2000 Workshop on Learning Statistical Model from Relational Data. AAAI, 2000.

Page, David. ILP (2000): Just Do It. J. Lloyd et al. (Eds.): CL 2000, LNAI 1861, p. 25-40.

Padgett, R. A. and Burge, C. B. (2003); Splice Sites. In Encyclopedia of the Human Genome, Nature Press.

Pertea M, Lin X. and Salzberg S. L. (2001); GeneSplicer: a new computational method for splice site prediction,
Nucleic Acids Res. Mar 1;29(5):1185-90

Rawn, J. David (1989), Bioquimica, McGraw-Hill Interamericana, p. 781-820.

Salzberg S.L., Searls D.B., Kasif S., (1998) Grand Challenges in Computacional Biology, Computational Methods in Molecular Biology, Vol.32, p. 3-10, Elsevier.

Searle SM et al (2004). The otter annotation system. Genome Res. 2004 May;14(5):963-70

Siu-wai Leung, Chris Mellish, and Dave Robertson (2001); Basic Gene Grammars and AND-ChartParser for language processing of Escherichia coli promoter AND sequences. Bioinformatics 17: 226-236.

Srinivasan, R.D. King S.H. Muggleton, and M. Sternberg (1997). Carcinogenesis predictions using ILP. In N. Lavrac and S. Dzeroski, editors, Proceedings of the Seventh International Workshop on Inductive Logic Programming, p. 273-287. Springer-Verlag, Berlin LNAI 1297.

Tamaddoni-Nezhad A. , Chaleil R., Kakas A., and Muggleton S.H. (2006). Application of abductive ILP to learning metabolic network inhibition from temporal data. Machine Learning, 2006. DOI: 10.1007/s10994-006-8988-x.

Tom, Mitchell (1997); Machine Learning, ISBN 0-07-042807-7, McGraw-Hill.

Turcotte M, S.H. Muggleton, and M.J.E. Sternberg (1998). Protein fold recognition. In C.D. Page, editor, Proc. Of the 8th International Workshop on Inductive Logic Programming (ILP-98), LNAI 1446, p. 53-64, Berlin. Springer-Verlag.

Turcotte, S.H. Muggleton, and M.J.E. Sternberg (2001). Automated discovery of structural signatures of protein fold and function. Journal.of Molecular Biology, 306:591-605.

Wang Z., Rolish M., Yeo G., Tung V., Mawson M. and Burge, C. (2004); Systematic Identification and Analysis of
Exonic Splicing Silencers; Cell, Vol. 119, 831–845, December 17.

Watanabe H. and Muggleton S.H. (2005). Learning Stochastic Logical Automaton. In Proceedings of the 19th Annual Conferences of JSAI, LNCS 4012, pages 201-211. Springer-Verlag, 2005.

Wen-Hsiung, Li (1997), Molecular Evolution. ISBN 0-87893-463-4 p. 7-34 Sinauer Associates.

Wren J., Johnson D., and Gruenwald L. (2005); Automating Genomic Data Mining via a Sequence-based Matrix Format and Associative Rule Set; BMC Bioinformatics, 6(Suppl 2):S2