# UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE CIENCIAS

## UNA RED NEURONAL DE VOTACION PARA ANALISIS GENOMICO

T E S I S

QUE PARA OBTENER EL TITULO DE

A C T U A R I A

P R E S E N T A :

TATIANA TERESA MARQUEZ LAGO





DIRECTOR DE TESIS:

DR. PEDRO MIRAMONTES VIDAL



MEXICO, D. F. 2000

**MAT. MARGARITA ELVIRA CHÁVEZ CANO**
**Jefa de la División de Estudios Profesionales de la**
**Facultad de Ciencias**
**Presente**

Comunicamos a usted que hemos revisado el trabajo de Tesis:

"Una red neuronal de votación para análisis genómico"

realizado por Tatiana Teresa Márquez Lago

con número de cuenta 9650331-3 , pasante de la carrera de Actuaría

Dicho trabajo cuenta con nuestro voto aprobatorio.

Atentamente

| | |
|---|---|
| Director de Tesis<br>Propietario | Dr. Pedro Miramontes Vidal |
| Propietario | Dr. Germinal Cocho Gil |
| Propietario | M. en C. José Luis Gutiérrez Sánchez |
| Suplente | Dr. Luis Medrano González |
| Suplente | Dr. Ricardo Mansilla Corona |

**Consejo Departamental de** Matemáticas

M. en C. José Antonio Flores Díaz

**AGRADECIMIENTOS.**

A mis padres, por su incondicionalidad, objetividad y amor. A mis hermanos, Bibiana y Alejandro por su cariño y comprensión.

A mis profesores y sinodales de la Facultad de Ciencias de la U.N.A.M., Dr. Germinal Cocho, Dr. Luis Medrano, Dr. Ricardo Mansilla, M. en C. José Luis Gutiérrez y en especial al director de esta tesis, Dr. Pedro Miramontes Vidal, por su grandiosa amistad, orientación y sobre todo por sus valiosos consejos que me han ayudado a pasar varias dificultades.

A mis queridos amigos y amigas: Mireya Perryman, Karla García, Circe Henestrosa, Carlos Vizcaíno, Carlos L. Natarén, Luis Uranga, Gonzalo C. Botinelli y Wanda Becerra. También quisiera agradecer los consejos del Dr. Fernando López, M. Ana María Álvarez y Arqlga. Graciela Rodríguez de la E.N.A.H., así como al Dr. Rafael Pérez Taylor del Instituto de Investigaciones Antropológicas, U.N.A.M.

A los químicos, biólogos, matemáticos, fisicos, actuarios y computólogos con los que pasé largas horas de café hablando, entre otras cosas, de programas y cadenas de DNA.

A todos y cada uno de los que de alguna manera formaron parte del desarrollo de esta tesis, así como de los que puedan hacer uso de ella.

**Gracias.**

**Índice.**

## Introducción.

La biología molecular ofrece, tras un desarrollo considerable, técnicas sin precedentes que permiten el acceso a información genética. Esta última puede manejarse con herramientas matemáticas sofisticadas y así dar un amplio espectro de posibilidades de interpretación a distintos problemas biológicos, principalmente de tipo evolutivo; hecho reflejado en un menor crecimiento de los métodos experimentales para la localización de genes en comparación del relacionado con las herramientas matemáticas para este mismo fin.

Dentro de sus aplicaciones, identificar genes computacionalmente implica posibilidades importantes como: detección de candidatos de "genes de enfermedades", compilación de genes en proyectos de secuencias genómicas, asignación de funciones que traten de explicar un mapeo genético de los organismos, reconstrucción filogenética, identificación forense (humana y de productos biológicos diversos), medicina molecular, ingeniería genética (dentro de la cual se da la producción de genes médicos e industriales) y hasta la producción de armas biológicas. Pero veamos más detalladamente dos de las aplicaciones antes mencionadas: la reconstrucción filogenética y la ingeniería genética.

La primera de ellas trata de bosquejar trayectorias evolutivas al relacionar especies mediante la comparación de los genes que tengan en común.[1] La divergencia génica con respecto a las especies antecesoras se manifiesta en las diferentes propiedades de las proteínas que producen; dichas diferencias son una consecuencia histórica creada por mutaciones continuamente acumuladas en linajes separados. Por su parte, una proteína mutada que no funciona correctamente en el medio del individuo será eliminada; en consecuencia los individuos que contengan dicha mutación tendrán una desventaja selectiva contra los que produzcan la proteína normal: tendrán menor (o ninguna) descendencia y la mutación desaparecerá de la población a través de la selección natural. En cambio, si la alteración en la función de la proteína es mínima, *i.e.* la mutación no representa una ventaja o desventaja selectiva para los individuos que la contienen, ésta permanecerá en frecuencias pequeñas dentro de la población (Frank-Kamenetskii M., 1993). Por consiguiente, diferentes rutas evolutivas tendrán consecuencias estructurales que permitirán diferenciar

organismos a partir de su genoma. Quiero explicar esto detalladamente. Al definir tres índices (ampliamente descritos en el capítulo IV) que expresen la falta de homogeneidad en el DNA en función de las clasificaciones de los nucleótidos y al suponer un espacio de secuencias de DNA donde cada secuencia sea representada por un punto único cuyas coordenadas sean dichos índices, la distribución de los genes de distintos organismos se verá reflejada en dicho espacio –sorprendentemente- a través de conjuntos relativamente pequeños. Cada uno de ellos contendrá los diversos genes de un organismo en particular, lo cual puede llevar a hablar de la existencia de "estilos genómicos" (Miramontes P. *et al.* 1995).[2] Dichos estilos serán evidentemente dependientes de las diferencias entre los organismos. Un ejemplo de ellas es la falta de un núcleo en las células procariontes, así como la discontinuidad de porciones codificables en sus secuencias, características que se explicarán detalladamente en el capítulo I.

Pero, ¿Por qué diferenciarlos?. Esto nos lleva a la segunda aplicación citada. El desarrollo de técnicas sofisticadas de manipulación geno-fenotípica, como la ingeniería genética, hace necesario primero contar con un mecanismo eficiente de clasificación de los estilos genómicos. Como es de esperarse, los retos y riesgos que conlleva la investigación en esta materia suelen compensarse con sus implicaciones económicas. Una de ellas es la posibilidad teórica de producir prácticamente cualquier proteína en grandes cantidades y a un costo comparativamente bajo. Por ejemplo, los organismos de algunos hombres y mujeres no son capaces de producir hormonas que les permitan tener un desarrollo normal y, como se ha observado, sólo era posible obtener dichas hormonas de los cadáveres humanos. Esta "colecta" puede resultar un tanto inconveniente e infactible si el deterioro de las hormonas no les permite llevar a cabo su función completa; problema resuelto a través de su producción *in vitro*.

Lo mismo sucede con el "interferon" humano y la insulina. El primero es una proteína muy eficaz en la lucha contra varios tipos de virus y la segunda es una hormona reguladora de los niveles de azúcar en la sangre, básica en el tratamiento de la diabetes y obtenible únicamente de los seres humanos. Por otra parte, se pueden producir vacunas inofensivas,

---

[1] Los seres humanos, por ejemplo, no solamente comparten genes con los mamíferos. No obstante, compartirán más genes con sus parientes más cercanos que con los distantes.

[2] Esto no es evidente a primera vista ya que el origen de cada gen dentro de su respectivo genoma puede ser diferente.

tanto en medicina como en agricultura, pues se elimina el riesgo de que alguna partícula del virus esté aún viva, que en su caso conllevaría a crear una infección en lugar de un instrumento de cura (Frank-Kamenetskii M., 1993).

Además, la copia de ciertos organismos y la pronta detección y modificación de genes de enfermedades son temas prioritarios dentro de la ingeniería genética. Si se seleccionaran variantes con características deseadas, por ejemplo aquellos capaces de procesar substancias particulares de una manera más eficiente, se podría tratar de aislar una enzima modificada de uno de ellos y alterar aquellas que no fueran a trabajar tan efectivamente dentro de la población. Por lo tanto en ciertos casos se induce la clonación de una secuencia de DNA, lo que permite producir cantidades infinitas de una molécula original.[3] La tecnología de clonación involucra la constitución de moléculas nuevas del DNA juntando secuencias de diferentes fuentes, cuyo producto es generalmente conocido como DNA recombinante. Sus métodos han hecho posible crear genes artificiales con cualquier secuencia de nucleótidos deseada, construidos en moléculas especiales de DNA, representadas y conocidas como vectores.

Una última aplicación de la ingeniería genética, entre tantas, es la producción masiva de proteínas con un balance óptimo de componentes para alimentos y forraje, lo cual en teoría podría reflejarse a largo plazo en una mejor distribución alimenticia e inclusive erradicar problemas de hambruna (*Idem*; Lewin, R. 1997).

Hoy en día, la mayor parte de los análisis genéticos son llevados a cabo a partir del estudio de bases de datos de secuencias de DNA y de la aplicación de métodos de análisis matemático; rubros que se encuentran en continuo desarrollo. En esta tesis se hace uso de ambos y en primera instancia se trata la identificación de secuencias genómicas a través de un modelo de redes de neuronas artificiales basada, principalmente, en los estilos genómicos. Se espera que tal instrumento pueda contribuir considerablemente al desarrollo de varias de las herramientas de aplicación genética antes citadas pues, el no contar con una identificación eficiente de las secuencias haría casi imposible la puesta en práctica de aplicaciones tan especializadas. Esto resalta la importancia de contar con un mecanismo que lleve a cabo esta tarea de la manera más confiable posible y, dado que dichas aplicaciones tienen un ya evidente gran impacto económico, se hace aún más obvia la

[3] Un clon se define como un gran número de células o moléculas idénticas a un original ancestral.

necesidad de extender la línea de investigación planteada. Con dicho objetivo, se pretende acercar al lector tanto a la terminología básica de la genética como al uso general y particular (a través de un modelo) de las redes neuronales, con la siguiente organización de temas:

En el capítulo I se explica brevemente lo que es una secuencia de nucleótidos y las estructuras que compone. También se detallan algunas generalidades y conceptos de genética útiles en la comprensión y construcción de la red neuronal presentada.

En el capítulo II se describe la importancia y parámetros utilizados en el reconocimiento de patrones sentando las bases de la utilización de las redes neuronales para este propósito.

En el capítulo III se da una introducción al uso de una red neuronal elemental, así como algunas de sus limitaciones y modos de operación.

En el capítulo IV se presenta el modelo de red neuronal utilizado en este trabajo y se discuten las consecuencias de su aplicación.

Por último, se presentan las conclusiones generales, las cuales contienen algunas sugerencias y propuestas para continuar esta línea de investigación. Los detalles del programa de red utilizado, el programa que procesa la información de entrada a la red y las secuencias de nucleótidos utilizadas están a la disposición de quien lo requiera y podrán ser solicitadas en: http://themis.fciencias.unam.mx/~tatiana/

## Capítulo I. Fenomenología del DNA.

Las biomoléculas pueden dividirse en cuatro grandes categorías: proteínas, ácidos nucleicos, carbohidratos y lípidos (Babloyantz A. 1986). Los ácidos ribonucleico (RNA) y desoxirribonucleico (DNA) son las cadenas de nucleótidos que portan la información genética. Para entender esto pensemos en las pautas de representación, en nomenclatura química, de las moléculas de DNA: hay un abecedario con un número finito de símbolos, las palabras son formadas a partir de éste por una secuencia bien definida de letras y una sucesión de palabras forma un mensaje significativo. Así, un texto de DNA es una secuencia ininterrumpida de letras, las cuales incluso actúan como signos de puntuación y tienen toda la información e instrucciones necesarias para la síntesis de cada molécula en el organismo (*Idem*).

De acuerdo al modelo de Watson y Crick, la molécula de DNA está formada por dos polímeros lineales, secuencias continuas, cadenas de bases nitrogenadas o nucleótidos de cuatro tipos: A, C, G y T (*adenina, timina, guanina y citosina* respectivamente). Los monómeros que constituyen cada cadena tienen moléculas de fosfato y desoxirribosa además de una de las cuatro bases nitrogenadas (figura 1).[1]

La base nitrogenada está ligada a una posición nombrada '1' en la pentosa por medio de un enlace glucosídico y para evitar ambigüedades en los sistemas numéricos de los anillos heterocíclicos y los azúcares, se da un símbolo (') a dicha posición, como es ilustrado en la figura 2. La posición 5' de una pentosa se conecta con la posición 3' de la siguiente a través de una molécula de fosfato. Así, el último nucleótido tiene un grupo 5' unido a un fosfato mientras que en el otro la unión se encuentra en la posición 3'. Por convención las secuencias son expresadas como una cadena de nucleótidos que corre en la dirección 5'→ 3', *i.e.*. de la posición 5' a la izquierda a la 3' en la derecha (Lewin B. 1997).

[1] La difracción por medio de rayos X ha mostrado que el DNA tiene la forma de una hélice regular, dando un giro completo cada 34 Å (34 nm) con un diámetro de aproximadamente 20 Å. Como la distancia entre nucleótidos adyacentes es 3.4 Å, se estima que deben haber 10 nucleótidos por giro (Lewin B. 1997).

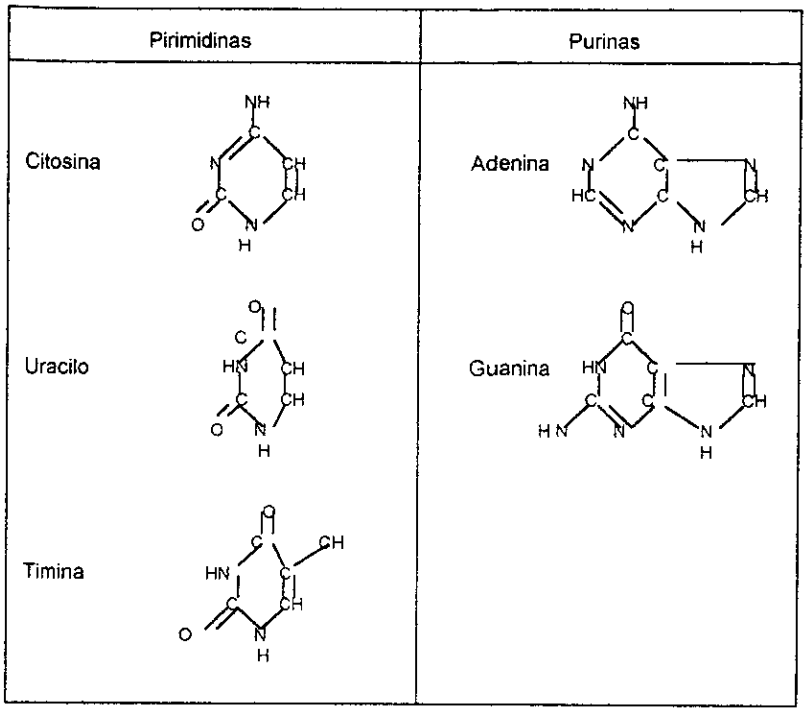


Figura 1. Los nucleótidos, divididos en purinas y pirimidinas.

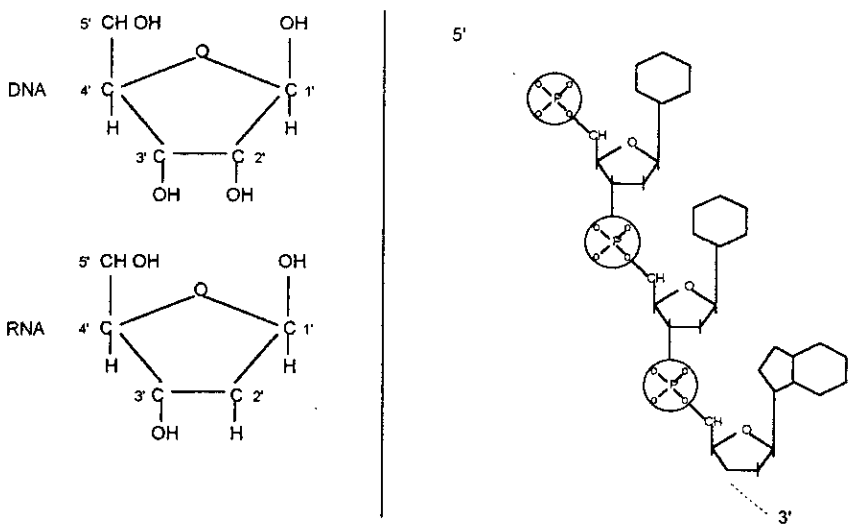


Figura 2. Las posiciones en la pentosa y la dirección 5'→3'.

## 1.1 Divisiones de los nucleótidos y su complementariedad.

Las cuatro bases nitrogenadas se han clasificado de acuerdo a tres criterios, considerando sus propiedades y el "principio de complementariedad" de las bases, el cual dice que (Hoagland M. 1985):

A siempre corresponde a T y T siempre corresponde a A
C siempre corresponde a G y G siempre corresponde a C

Así, como la secuencia de nucleótidos puede ser totalmente arbitraria pero las cadenas se encuentran fuertemente interconectadas, la información de una molécula puede ser conocida a partir de una sola rama de la doble hélice.[2]

La primera clasificación las divide en *purinas* y *pirimidinas* (figura 1). Las purinas son las bases A y G, llamadas "grandes" por ser moléculas de dos anillos, mientras que las pirimidinas son llamadas "pequeñas" por ser de un anillo. La nomenclatura de las secuencias de nucleótidos representa a las purinas con una R y a las pirimidinas con una Y.

De acuerdo al principio de complementariedad, para mantener constante el ancho de la hélice es necesario tener una purina en la misma posición de una pirimidina en la cadena complementaria, como puede observarse en la figura 3. No obstante, no todas las combinaciones son posibles ya que las bases de las cadenas se unen a través de puentes de hidrógeno diferentes y no siempre resultan ser compatibles (Miramontes P. 1992).

Como las parejas A-T están unidas por dos puentes de hidrógeno y las C-G por tres, se obtiene la segunda clasificación de las bases: por sus conexiones de puentes de hidrógeno A y T son *débiles* mientras que C y G son *fuertes*. En este caso, la nomenclatura de las secuencias de nucleótidos representa a las moléculas débiles con una W y a las fuertes con una S (Lewin B. 1997).

La tercera y última clasificación es aquélla que las separa en un grupo *aminado* (A y C), reflejado en los surcos mayores de la doble hélice, y en uno *cetónico* (G y T), el cual

[2] El modelo con el principio de complementareidad requiere que las dos cadenas polinucleótidas corran en diferentes direcciones, *i.e.* una en el sentido 5'→3' y la otra en 3'→5' (Idem).

provoca modificaciones estructurales diferentes.[3] En esta clasificación, la nomenclatura asigna el símbolo M a las bases del primer grupo y una K a las del segundo (*Idem;* Miramontes P. *et al.* 1995).

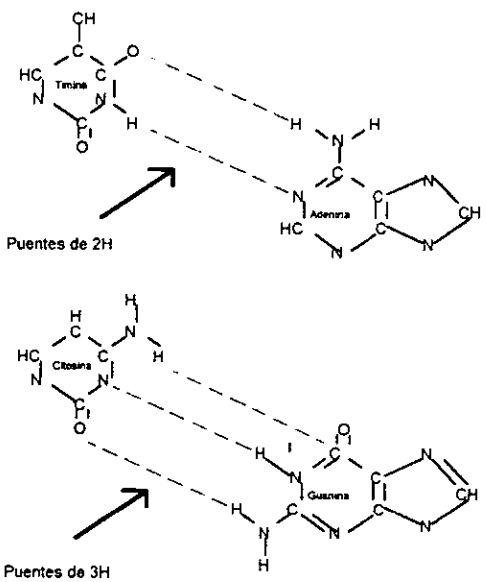


Figura 3. Los puentes de hidrógeno en las bases complementarias.

## 1.2 La relación gen-proteína.

En cuanto a su tipo de célula(s), los organismos se han dividido en dos tipos. Las bacterias, conocidas como *procariontes* (*i.e.* células en las que el material genético no está contenido dentro del núcleo), constituyen el primero de ellos. Todas las demás formas de vida son *eucariontes* y están compuestas de células con núcleo, son típicamente mayores que las procariontes y es común que contengan más material genético.[4] En este tipo de organismos,

[3] Las diferencias estructurales se deben a que el grupo cetónico tiene una carga ligeramente negativa, mientras que en el aminado la carga es positiva. En realidad esto es lo que determinaría el principio de complementareidad de las bases.

[4] Al conjunto de todo el material genético de una célula se le conoce como genoma. El genoma humano consta de aproximadamente $3*10^9$ nucleótidos, así como los animales y plantas más comúnes. Un virus tiene aproximadamente 10,000 bases, mientras que se dan casos como en algunos anfibios con 20,000 megabases (Lewin R. 1997; Miramontes P. 1992).

por lo general, no toda la molécula de DNA está constituida por genes. Por ejemplo, en los seres humanos sólo el cinco por ciento del material genético codifica en aminoácidos para la producción de proteínas o la regulación de ésta.[5]

La cadena de DNA codifica las secuencias de aminoácidos que formarán a las proteínas (polipéptidos), base estructural y funcional de todos los seres vivos. Un péptido consiste en un pequeño número de aminoácidos concatenados, una cadena de aminoácidos más larga conectada de esta manera se llama polipéptido y el término proteína se usa para describir la molécula funcional que puede consistir de una o más cadenas polipéptidas y otros elementos no protéicos. Así, un gen estructural peptídico es una secuencia de ácidos nucleicos con la información representativa de un polipéptido particular (Miramontes P. 1992).

La información codificable en proteínas de los genes de los organismos constituidos por células eucariontes está contenida en pequeños paquetes llamados *exones* mientras que a los segmentos no codificadores entre éstos se les llama *intrones*, cuya función, si tuvieren, aún no ha sido descubierta.[6] Los exones son las regiones representadas en el mRNA (RNA mensajero) maduro y los intrones son aquéllas retiradas de la secuencia recién transcrita de mRNA. El proceso con el que se remueven se llama edición del RNA y esencialmente lleva a la eliminación precisa de un intrón para posteriormente unir las puntas del RNA y así formar una molécula covalentemente intacta, como puede observarse en la figura 4.

Por definición, un gen comienza y termina con los exones, correspondiendo a las puntas 5' y 3' del RNA. Puede entonces definirse la dirección de una cadena polipéptida de acuerdo a la orientación de las ligas péptidas. El aminoácido al final de una cadena tiene un grupo $-NH_2$ libre, y define la punta N-terminal. El aminoácido en la otra punta de la cadena tiene un grupo -COOH libre y define la punta C-terminal. Por lo tanto las secuencias proteicas están escritas convencionalmente de la punta N-terminal (a la izquierda) a la C-

[5] Es importante notar que dicho porcentaje variará según las especies.

[6] Como los intrones aparentemente no codifican en proteínas, se debía buscar otra función. De aquí la formación de opiniones divergentes en dos grupos: aquéllos que piensan la estructura intrón-exón como promotora de la evolución, en el supuesto que en lugar de depender de mutaciones para crear nuevos genes, la conjunción de diferentes exones en diferentes combinaciones podría tomar esta función. El otro grupo piensa los intrones como parte del material genético ancestral y que fueron eliminados en los procariontes, constituidos por genomas relativamente más simples (Lewin R. 1997). Los intrones, de los cuales hay en promedio hay media docena por gen, son mucho más grandes (típicamente 10 veces) que los exones (*Idem*).

terminal (a la derecha). De aquí la segunda definición de gen: lo que incluye la secuencia entera representada en mRNA (Lewin B. 1997).

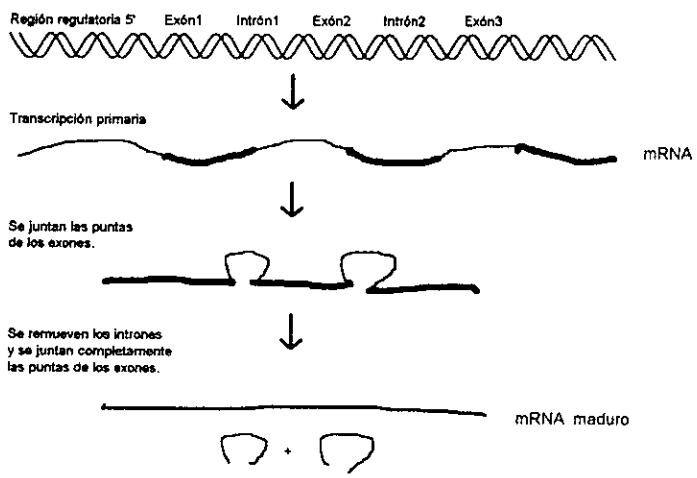


Figura 4. El proceso de edición del RNA.

En la forma usual del gen interrumpido, cada exón codifica una secuencia de aminoácidos representando una parte apropiada de la proteína mientras que los intrones no juegan ningún papel en la producción final de ésta. Así, la inexistencia de genes interrumpidos es una evidencia de una mayor longitud potencial del gen a comparación de la unidad que codifica proteína (Idem).

Como se ha dicho, la mayor parte del gen consiste de una secuencia de DNA dedicada únicamente a codificar una proteína. Sin embargo, hay algunos casos en donde una secuencia de DNA no tiene como función única el representar una proteína. Un caso curioso es el de los genes empalmados, los cuales ocurren en la situación relativamente simple en la que un gen es parte de otro. Otro caso es aquel donde una secuencia de DNA se usa en más de una manera en algunos genes, de tal forma que no pueda caracterizarse simplemente como exón e intrón. En estos genes, patrones alternativos de expresión genética crean cambios en las rutas de conexión exónica (*Idem*). Existe un sin fin de casos que puede complicar cualquier intento de análisis genómico cuya descripción no contribuiría significativamente al objetivo principal de esta tesis. No obstante, se debe

aclarar que pueden ser de suma importancia para la realización de otros tipos de investigaciones en materia genética.

### 1.3 Replicación, transcripción y traducción.

Por el principio de complementariedad, para conocer la serie de nucleótidos, en cualquier región solamente se necesita una de las hélices de DNA y basta analizar y describir las secuencias de bases y no de pares de bases. Para que el DNA sea traducido en aminoácidos, las hileras de la doble hélice deben separarse temporalmente y solamente una porción continua del genoma no tendrá su contraparte en algún momento (figura 5).

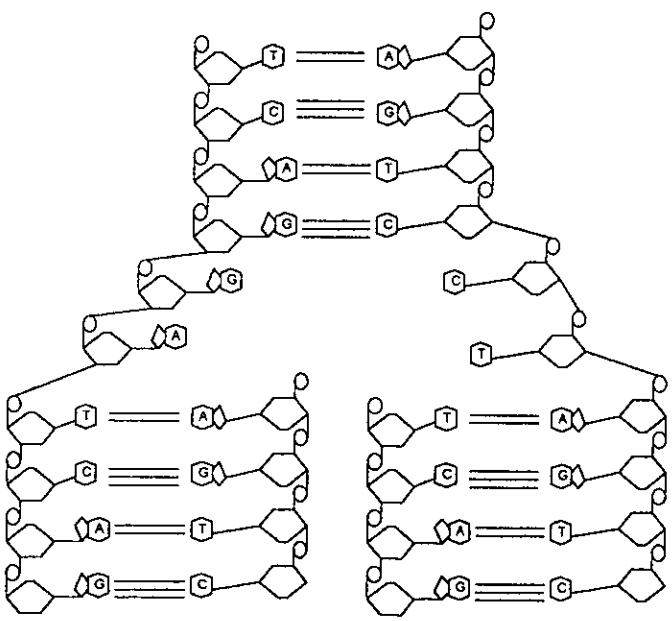


Figura 5. La replicación de la cadena de DNA.

La doble hélice se desprende en dos cadenas individuales de nucleótidos para atender dos procesos fundamentales en la célula. El primero de ellos es la *replicación*, equivalente a la duplicación del material genético: se forman dos copias idénticas del DNA sujetas a probables mutaciones debidas a sustitución, supresión, inserción o cualquier tipo de error

en el copiado, las cuales quedarán insertadas en células distintas, producto de la división celular (Babloyantz A. 1986, Miramontes P. 1992).

El segundo proceso es el de la *transcripción - traducción*. En la transcripción, una enzima reconoce la secuencia de nucleótidos entre los genes (llamada promotora) y moviéndose a lo largo del gen hace una copia en la forma de molécula de RNA. Esta copia es idéntica a la de DNA, excepto por el cambio de las timinas por moléculas de uracilo, representadas con una U (Frank-Kamenetskii M. 1993). Así, puede comenzar el proceso de traducción en donde una enzima, conocida como polimerasa del RNA, forma el RNA mensajero (mRNA) y lo utiliza en la síntesis de la proteína, tras su debido proceso de edición en los eucariontes (figura 6).[7]

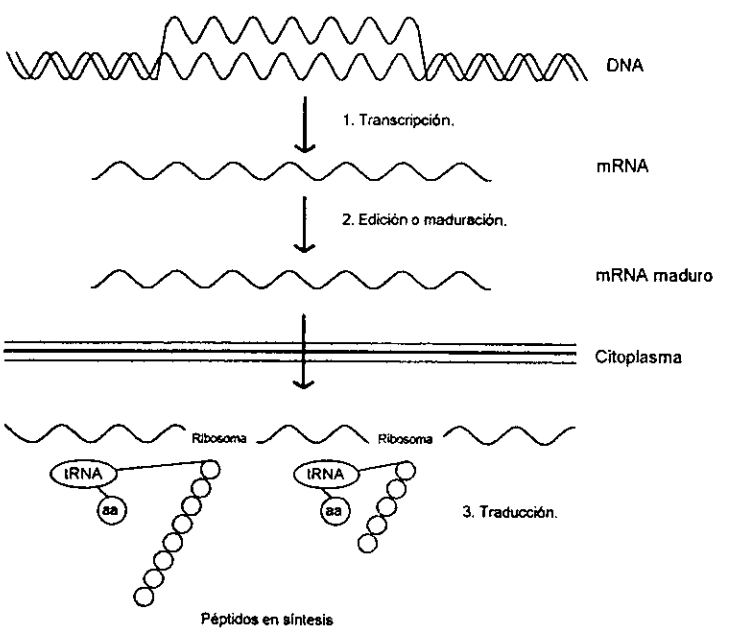


Figura 6. El procesamiento, transporte y traducción del mRNA.

[7] La síntesis de los ácidos nucleicos es catalizada por enzimas específicas, las cuales reconocen el patrón y catalizan la adición de subunidades a la cadena polinucleótida que está siendo sintetizada. El nombre de las enzimas se da de acuerdo al tipo de cadena sintetizada: DNA polimerasa y RNA polimerasa. La degradación de ácidos nucleicos también es llevada a cabo por enzimas específicas, desoxirribonucleasas y ribonucleasas. Las nucleasas caen dentro de las clases generales de exonucleasas y endonucleasas. Las endonucleasas cortan vínculos dentro de las moléculas de DNA o RNA, generando fragmentos discretos. Las exonucleasas remueven los residuos uno por uno desde el final de la molécula, generando mononucleótidos (Frank-Kamenetskii M. 1993).

Pero no solamente el mRNA es creado durante la transcripción, también lo son el tRNA (RNA de transferencia) y el rRNA (RNA ribosomal). El tRNA es la molécula encargada de llevar a cabo la adaptación de la traducción en el sentido de que existe una incongruencia estructural básica entre el lenguaje de los nucleótidos y el de los aminoácidos que codifican.[8] Sus dos propiedades principales son el representar un aminoácido único (al que se encuentra unido covalentemente, figura 7) y contener un anticodón correspondiente (Lewin B. 1997).[9]

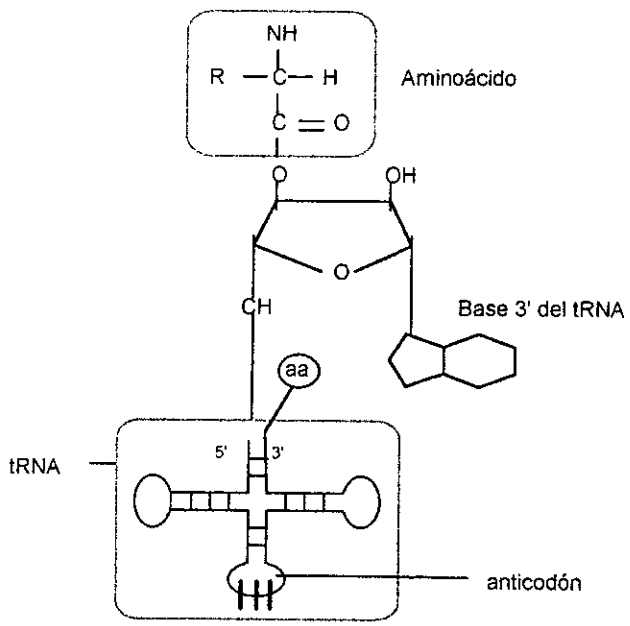


Figura 7. Aminoácido ligado a la punta 3' del tRNA.

Conviene ahora introducir al principal actor de la traducción: el ribosoma. Este es una complicada maquinaria, formada por aproximadamente 50 proteínas y una molécula de RNA ribosomal (rRNA), el cual actúa como un programa de computadora al traducir textos del lenguaje de DNA y RNA al lenguaje de aminoácidos de las proteínas.

[8] La traducción es llevada a cabo por unas enzimas especificas: aminoacil tRNA transferasas.

Por su parte, cada aminoácido se ha clasificado como una secuencia de tres nucleótidos de RNA, comúnmente llamada codón. En 1961 Crick y sus colegas mostraron que las secuencias deben leerse en tripletes no empalmados desde un punto fijo de comienzo. El que no estén empalmados implica que cada codón consiste de tres nucleótidos, por lo tanto codones sucesivos son representados por trinucleótidos sucesivos. El uso de un punto de comienzo significa que el montaje de una proteína debe empezar en una punta y trabajar hacia la otra, de tal manera que las diferentes partes de una secuencia codificadora no puedan ser leídas independientemente. Tomando en cuenta todas estas consideraciones, se le llama código genético a las relaciones de traducción codón-aminoácido (Idem).

Pero, si el código genético se lee en tripletes no empalmados, debe haber tres maneras posibles de traducir una secuencia de nucleótidos en proteínas dependiendo del punto de comienzo. Estas maneras son llamadas "marcos de lectura" y se dice que es abierto aquel que consiste exclusivamente de tripletes que representen aminoácidos, como se ilustra en la figura 8.

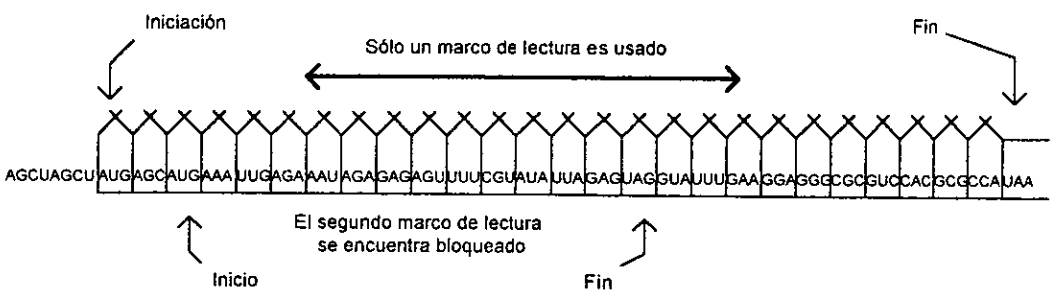


Figura 8. El marco de lectura abierto y los posibles marcos bloqueados.

Cabe añadir que, en genes celulares, una secuencia de DNA es usualmente leída en sólo uno de los tres marcos de lectura potenciales, pero en algunos genes virales y mitocondriales hay un empalme entre dos genes adyacentes, los cuales son leídos en diferentes marcos de lectura (*Idem*).

[9] El anticodón es una secuencia de 3 nucleótidos representativos de algún aminoácido. El anticodón hace al tRNA capaz de reconocer la secuencia de una cadena a través de la complementareidad de las bases (Lewin B. 1997).

## 1.4 El código genético.

Para explicar el código genético, habrá que comenzar diciendo que existen 20 aminoácidos, los cuales son (Hoagland 1985): fenilalanina (Phe), leucina (Leu), isoleucina (Ile), metionina (Met), valina (Val), serina (Ser), prolina (Pro), treonina (Thr), alanina (Ala), tirosina (Tyr), histidina (His), glutamina (Gln), asparagina (Asn), lisina (Lys), ácido aspártico (Asp), ácido glutámico (Glu), cisteína (Cys), triptófano (Trp), arginina (Arg) y glicina (Gly).[10] Entonces, si se tienen 20 diferentes tipos de aminoácidos y se sabe que los codones pueden tener 64 diferentes combinaciones, se puede ver claramente la existencia de sinónimos en el código genético, *i.e.* diferentes codones codifican en el mismo aminoácido,[11] como puede observarse en la tabla 1.

| Primera base | Segunda base: U | | C | | A | | G | |
|---|---|---|---|---|---|---|---|---|
| U | UUU | Phe | UCU | Ser | UAU | Tyr | UGU | Cys |
| | UUC | | UCC | | UAC | | UGC | |
| | UUA | Leu | UCA | | UAA | Stop | UGA | Stop |
| | UUG | | UCG | | UAG | | UGG | Trp |
| C | CUU | Leu | CCU | Pro | CAU | His | CGU | Arg |
| | CUC | | CCC | | CAC | | CGC | |
| | CUA | | CCA | | CAA | Gln | CGA | |
| | CUG | | CCG | | CAG | | CGG | |
| A | AUU | Ile | ACU | Thr | AAU | Asn | AGU | Ser |
| | AUC | | ACC | | AAC | | AGC | |
| | AUA | | ACA | | AAA | Lys | AGA | Arg |
| | AUG | Met | ACG | | AAG | | AGG | |
| G | GUU | Val | GCU | Ala | GAU | Asp | GGU | Gly |
| | GUC | | GCC | | GAC | | GGC | |
| | GUA | | GCA | | GAA | Glu | GGA | |
| | GUG | | GCG | | GAG | | GGG | |

Tabla 1. El código genético.

[10] Veamos algunas de las divisiones que se han hecho sobre los aminoácidos (Idem):
a) Según sus cargas iónicas, los aminoácidos se clasifican en cuatro grupos: básicos, ácidos, neutrales (algunos de ellos polares), y apolares (que también son hidrofóbicos).
b) Hay ocho familias de codones en las que cuatro codones comparten las mismas dos bases tienen el mismo significado, de tal forma que la tercera base no sirve en mucho para especificar el aminoácido. Hay siete pares de codones cuyo significado es el mismo con cualquier pirimidina en la tercera posición, y hay cinco pares de codones donde cualquier purina puede estar presente sin cambiar el aminoácido codificado.
Hay solamente tres casos en los que se da un significado único por la presencia de una base particular en la tercera posición: AUG (metionina), UGG (triptofano), UGA (término). Entonces C y U nunca tienen un significado único en la tercera posición y A nunca significa un aminoácido único.

[11] Aparte de los veinte aminoácidos estándar, algunos otros son ocasionalmente encontrados en las proteínas. Estos son creados modificando uno de los aminoácidos estándar después de ser incorporado a la proteína.

El código genético es lo suficientemente vasto para incluir también sus propios signos de puntuación, los cuales son de inicio y término. La señal para comenzar una cadena polipeptídica es un codón de iniciación especial que marca el comienzo del marco de lectura. En el caso de los codones de inicio, no existen aquéllos que tengan exclusivamente esta función; en condiciones específicas su función es asumida por los codones AUG y GUG, los cuales normalmente corresponden a los aminoácidos metionina y valina. Dos tipos de tRNA pueden acarrear este aminoácido; uno es usado para la iniciación y el otro en la extensión, por lo tanto el significado de dichos codones depende de su contexto. Por otra parte, se sabe que un exón termina cuando en el marco de lectura se halla el codón de fin, el cual a su vez tiene sinónimos (Frank-Kamenetskii M. 1993).

Cabe añadir que los primeros investigadores especificaban el código genético como el de un organismo: *Escherichia coli*. El primer sistema fuera de la célula se obtuvo a partir de esta bacteria y con el tiempo todo parecía indicar que el código genético de los demás organismos no era diferente al de ésta, lo cual resultó casi cierto (*Idem*).[12] Hoy se estima que 3.5 millones de especies vegetales y 1 millón de especies animales comparten el mismo código genético (Babloyantz A. 1986) de manera que se pueden hacer comparaciones directas con la reserva debida de aquellos organismos que no lo compartan. De cualquier forma, está claro que la estructura del código genético ha sido prácticamente inmune ante el proceso evolutivo, se ha conservado intacta desde la aparición de la vida en la tierra y ha resistido multitud de cambios en todas las escalas.

### 1.5 Diferencias entre procariontes y eucariontes.

Todas las secciones anteriores pueden entonces resumirse en el siguiente paradigma: los genes codifican a las proteínas que, a su vez, son responsables de la síntesis de otros tipos de estructuras. La secuencia de un gen especifica la secuencia de una proteína, la de la

[12] Niremberg M. realizó otros experimentos con sistemas fuera de la célula, tomados desde sapos hasta conejillos de indias. Estos experimentos no detectaron variación alguna con respecto al código genético de *E. coli*; entonces pareció que el código era universal. Es cierto que algunos mutantes de *E. coli* presentan variaciones: algunos de sus codones eran leídos como sensibles (*i.e.* respondían a aminoácidos determinados). A este fenómeno se le conoce como supresión (Frank-Kamenetskii M. 1993).

proteína especifica su estructura molecular y ésta, finalmente, determina su ubicación dentro de la célula.[13]

Como se hizo notar, las células eucariontes contienen toda la información genética dentro de un núcleo y sólo en ellos (a excepción de algunas arqueobacterias consideradas procariontes por la presencia de un núcleo) se da la alternancia de intrones y exones en las secuencias de nucleótidos. La divergencia evolutiva entre los tipos de organismos sugiere que las diferencias funcionales y estructurales básicas deben reflejarse directamente en la organización genómica. De ser así, dichas diferencias podrían conformar un criterio básico de discriminación entre las secuencias de los organismos.

En esta tesis, se supone un modelo de identificación de organismos a partir de las diferencias estructurales en las secuencias de bases nitrogenadas, basado tanto en las clasificaciones descritas en la sección 1.1 como en los índices de información relacionados con la distancia entre los propios nucleótidos. Al suponer que las diferencias estarán claramente representadas en sus estilos genómicos, se puede pensar en una herramienta que separe nítidamente los grupos de organismos a analizar, causa de los siguientes capítulos.

[13] Las caracteristicas estructurales individuales no son el único determinante de las funciones de una molécula, su ubicación también es importante. No obstante, el orden de los eventos puede diferir entre las proteínas; algunas proteínas obtienen su estructura antes de su ubicación mientras que otras alcanzan su ubicación antes de obtener su estructura final (*Idem*). A estos cambios se les conoce como postraduccionales, los cuales se manifiestan después de que las proteínas sean sintetizadas en el ribosoma.

## Capítulo II. Reconocimiento de patrones.

La clasificación es una parte básica del reconocimiento de patrones y es preciso tratar este tema para sentar las bases que orientarán la identificación de estilos genómicos. Al realizar una clasificación primero es necesario identificar los rasgos que definen al objeto, *i.e.* las variables del patrón con el que se clasificará. Posteriormente, el clasificador será provisto con la lista de rasgos medidos y su tarea consistirá en mapearlos en sus respectivos estados diferenciales, *i.e.* asignarlos a la categoría que mejor los defina. Para lograr esto, es común basar los clasificadores en métricas de distancia y en la teoría de probabilidades, así como requerir varias medidas o rasgos del patrón de entrada para distinguir adecuadamente cuando los objetos pertenecen a diferentes categorías o, como llamaremos en adelante, clases.

### 2.1 Vectores, espacio de rasgos y funciones discriminantes.

Tómese como base la medición de $n$ rasgos, así la información se presentará en la forma de un vector de dimensión $n$. Entonces, el problema de clasificar un conjunto arbitrario $C$ de elementos de $R^n$ consiste en hallar hipersuperficies –llamadas "fronteras de decisión"- de dimensión $k<n$ que separen con nitidez a $C$ en varios subconjuntos (figura 9).

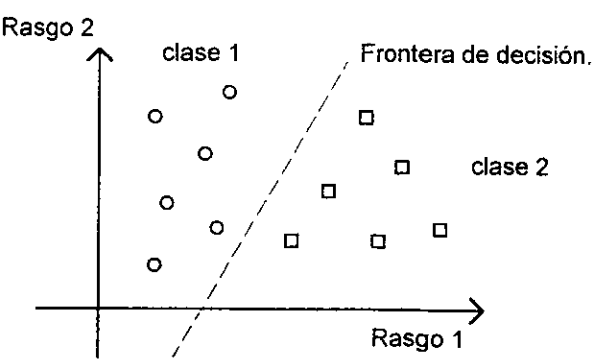


Figura 9. La frontera de decisión divide las diferentes clases.

Si no es posible caracterizar tales fronteras mediante una expresión analítica, se procede a construirlas mediante funciones discriminantes. Habrá de notarse que hay un número infinito de alternativas para estas formas matemáticas, por lo que en general se tratará de probar primero las más simples.

La superficie más simple, la de menor grado, es la lineal (*i.e.* un hiperplano). En aquellos casos en los que una superficie de este tipo marque una frontera de decisión bien definida se adoptará como clasificador dada su relativa simplicidad en expresión y representación. No obstante, si los grupos no son separables linealmente, es posible adoptar formas polinomiales de mayor grado. Puede hacerse el intento de recurrir a funciones más complicadas para definir las superficies de separación, siempre y cuando haya suficientes razones para suponer que los datos de entrada se distribuyen en conformidad con esto (Beale R., Jackson T. 1990). Debemos notar, sin embargo, que la separación de los puntos es óptima sólo en relación con el criterio adoptado.

Como se había dicho, en el caso de no contar con una expresión analítica para la frontera de decisión, ésta puede definirse en términos de funciones discriminantes continuas $\delta_k(\vec{x})$, $k = 1,2,....,m$, valuadas en el vector de rasgos $\vec{x}$.

Así, la frontera de decisión entre las clases $S_i$ y $S_j$, $\delta_{ij}$, se define como: $\delta_{ij} = \{x \in R^n | \delta_i(\vec{x}) - \delta_j(\vec{x}) = 0\}$, de manera que $\vec{x} \in S_i$ si y sólo si $\delta_i(\vec{x}) > \delta_j(\vec{x})$ para cualquier $j \neq i$ o, equivalentemente, si y sólo si $\delta_i(\vec{x}) = max_k \{\delta_k(\vec{x})\}$.

Esto define una frontera de decisión con base en una "función discriminante" (Kohonen T. 1989). En la práctica es necesario formar un discriminante sencillo dentro de una infinidad de posibles; sin embargo, en muchos casos se complicará dicha búsqueda al ser más necesario buscar una frontera que optimice las funciones del clasificador.

### 2.2 Métricas de distancias comúnmente utilizadas.

Para elegir $\delta_k(\vec{x})$, es necesario especificar una métrica en cuyos términos se defina la distancia de una clase a otra y medir así la similitud de dos muestras en el espacio geométrico de los mismos. Algunas de las más comunes son (*Idem*; Beale R., Jackson T. 1990):

a) Distancia de Hamming. Para dos vectores $\vec{x} = (x_1, x_2, \ldots, x_n)$, $\vec{y} = (y_1, y_2, \ldots, y_n)$ se evalúa la diferencia entre cada componente de $\vec{x}$ con su componente correspondiente en $\vec{y}$ y se suma el número de diferencias para obtener un valor absoluto de la variación entre los vectores: $d_{ham} = \sum i$, $i = \begin{cases} 0, x_j = y_j \\ 1, E.O.C. \end{cases}$, $j = 1, \ldots, n.$

b) Distancia euclideana. Dados, por ejemplo, dos vectores $x, y$ en un sistema coordenado rectangular, la distancia más corta será: $d_{euc} = (\vec{x}, \vec{y}) = \|\vec{x} - \vec{y}\| = \sqrt{\left(\sum_{i=1}^{n} (x_i - y_i)^2\right)}$, donde $n$ es la dimensión del vector de entrada, *i.e.* su número de rasgos. Un caso especial es el de los vectores binarios, donde esta métrica es equivalente a la raíz cuadrada de la "distancia de Hamming".

c) Distancia "Manhattan". Es una versión simplificada de la distancia euclidiana ya que no calcula las funciones de raíz cuadrada. Entonces: $d_{man}(\vec{x}, \vec{y}) = \sum_{i=1}^{n} |x_i - y_i|$. Aparte de ser mucho más rápida de computar, los puntos equidistantes del vector caen en una frontera cuadrada alrededor del mismo mientras que en la distancia euclidiana forman una frontera circular, como puede observarse en la figura 10. Esto introduce un error en la medida, el cual es aceptado a cambio de su velocidad de cálculo.

d) Distancia cuadrada. Simplificando aún más la distancia euclidiana y consecuentemente añadiendo más error, esta distancia se define como el máximo de las diferencias de cada componente de los vectores, *i.e.* $d_{cua}(\vec{x}, \vec{y}) = max_i |x_i - y_i|$

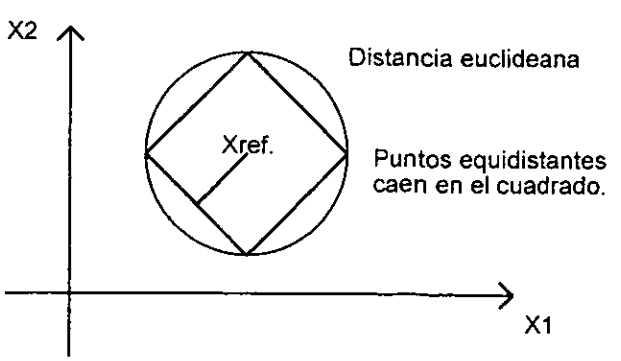


Figura 10. La diferencia entre la distancia euclideana y la Manhattan.

## 2.3 Clasificadores lineales.

Pensemos en un problema de dos clases (A y B) en un espacio bidimensional, donde el discriminante es una recta (figura 11). Sea $\vec{w} = (w_1, w_2)$ un vector al que llamaremos "vector de pesos". Entonces, la frontera de decisión (única puesto que sólo hay dos clases) es la recta $\delta = \left\{\vec{x} = (x_1, x_2) \middle| f(\vec{x}) = \sum_{i=1}^{2} w_i x_i - \theta = 0\right\}$ y $f(\vec{x})$ es un número real cuyo signo depende tanto de las coordenadas del vector de pesos como de las de $\vec{x}$. De hecho, las dos regiones en que $\delta$ divide al plano se caracterizan por que, en una, $f(\vec{x}) > 0$ y, en la otra, $f(\vec{x}) < 0$ (Beale R., Jackson T. 1990).

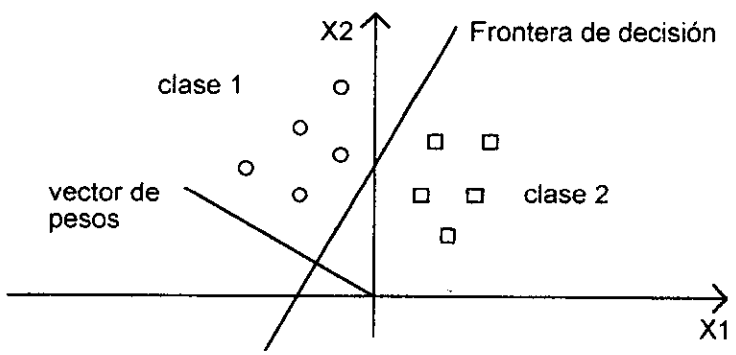


Figura 11. El vector de pesos como determinante de la frontera de decisión.

Al usar álgebra matricial, en el discriminante puede observarse la dependencia de la salida con el valor del vector de pesos: $f(\vec{x}) = \sum_{i=1}^{n} x_i w_i - \theta$, $f(\vec{x}) = \left(|\vec{w}| \bullet |\vec{x}| \cos\phi\right) - \theta$, donde $\phi$ es el ángulo que forman los vectores $\vec{x}$ y $\vec{w}$, y el $\cos\phi$ fluctúa entre -1 y 1, lo que influirá en el signo de la salida. En este caso hay dos parámetros, la pendiente y la ordenada al origen de la recta, que controlan la posición de la frontera de decisión y, como se puede ver, la primera está determinada por las coordenadas del vector de pesos: $\sum_{i=1}^{n} w_i x_i - \theta = 0$ i.e. $x_1 * w_1 + x_2 * w_2 - \theta = 0$, entonces $x_2 = -\frac{w_1}{w_2} * x_1 + \frac{\theta}{w_2}$. Esta es la ecuación cartesiana de la recta y la pendiente es igual al cociente $-w_1 / w_2$, mientras que la

ordenada al origen es $\theta/w_2$. Por consiguiente, de tenerse los valores correctos de las coordenadas del vector de pesos, es posible obtener la frontera de decisión adecuada. Pero ¿Cómo encontrar dichos valores? Esta tarea es la más crítica y, en el caso de esta tesis, se llevará a cabo mediante el uso de redes neuronales artificiales, como se explicará detalladamente en el próximo capítulo.

Una aclaración importante es que si en lugar de dos clases se tuvieran cuatro (A,B,C,D), como en la figura 12, las fronteras de decisión podrían seleccionarse para hacer pruebas entre A o el conjunto BCD; si el resultado no resultara ser de clase A entonces se probaría entre B o el conjunto CD y consecuentemente si no fuera de clase B se seleccionaría entre C y D.

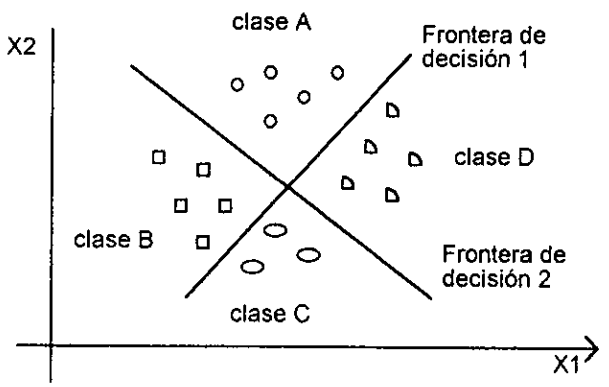


Figura 12. Separación de cuatro clases con dos fronteras de decisión.

Aún así, el clasificador lineal descrito solamente puede definir fronteras para problemas de separabilidad lineal (*i.e.* clases separables por una recta en el ejemplo de A y B, o por un hiperplano en $R^n$ para problemas con más rasgos). En problemas de separabilidad no lineal es posible introducir la no linealidad requerida en la superficie de decisión aplicando una transformación a los datos de entrada antes de ser clasificados. Para lograr esto es necesario encontrar la transformación que mapee los patrones en un código que sea capaz de ser clasificado por medio de un clasificador lineal. Gráficamente, es fácil explicar esto en $R^2$, razón por la cual se basó el ejemplo en un espacio bidimensional. No obstante, es importante notar que lo mismo ocurre en un espacio de dimensión $n$ (*Idem*).

## Capítulo III. Redes neuronales.

El problema de identificación que se trata de resolver requiere del procesamiento de muchos tipos de información que interactúan para dar una solución. Así, se usarán redes neuronales, cuya estrategia es capturar y tratar de simular la solución que daría el cerebro humano a este tipo de problemas y aplicarlos en sistemas computacionales (Viana L. 1998).

En el diseño del cerebro humano, lo más importante es la posibilidad de procesar simultáneamente, o en paralelo, y no la velocidad de operación; por ello el cerebro tiene la capacidad de representar y guardar información de manera accesible así como de procesarla al mismo tiempo que recibe muchos otros estímulos. Importa destacar la característica del paralelismo por ser causa de la capacidad de aprender y entrenarse a sí mismo (Freeman J., Skapura D. 1992).

El cerebro realiza tareas muy complejas con un aparente mínimo de esfuerzo gracias a su estructura, en la cual muchos elementos simples comparten el trabajo de dilucidar lo que ocurre en lugar de esperar la acción de un nodo resolviendo rápidamente todo el trabajo. Esta división del trabajo tiene también otras ventajas ya que muchas neuronas están involucradas al mismo tiempo y las contribuciones hechas por una sola no son tan importantes (Beale R., Jackson T. 1990). Esto significa que si una de ellas llegase a responder de manera inadecuada, es muy poco probable que afectase considerablemente a las demás neuronas. Repartir el trabajo de esa manera es conocido como "proceso distribuido" y brinda la ventaja de ser tolerante ante errores, así como de seguir funcionando ante la pérdida de alguna de sus unidades de procesamiento, razón por la que se les considera "robustas", lo cual las conducirá finalmente a tener una mayor certeza de no producir fácilmente una salida sin sentido.

Lo necesario en este caso es un modelo que pueda capturar las características importantes de los sistemas neuronales reales y así producir un comportamiento similar. No obstante, el modelo debe jerarquizar las variables a utilizar e ignorar deliberadamente particularidades que no modifiquen el resultado significativamente. La extracción de aspectos considerados importantes y el rechazo del resto es una característica general de la modelación; el objetivo de un modelo es producir una versión simplificada de un sistema, al sostener el mismo

comportamiento general, de tal manera que pueda ser entendido más fácilmente y no trate de reproducir con total exactitud el original. De no ser así se trataría de una manera, complicada y en la mayoría de los casos imposible, de obtener una copia idéntica del sistema.

### 3.1 La neurona elemental.

Veamos un poco la estructura general del sistema nervioso dentro del cerebro humano, el cual puede observarse como sistema original simplificado en la figura 13 y servirá como base del modelo de neurona elemental. El soma es el cuerpo de una neurona, del cual emergen las dendritas, que son las conexiones a través de las cuales llegan a la neurona todos los estímulos. El axón es el canal de salida y produce una respuesta eléctrica; este canal termina en una zona de contacto llamada sinapsis que lo une con las dendritas de otra célula. Por su parte, en la sinapsis se emite una sustancia química llamada neurotransmisor cuando su potencial es suficientemente grande, creado por la respuesta eléctrica.[1] Los neurotransmisores activan químicamente las puertas de las dendritas que, cuando están abiertas, permiten el flujo de iones con carga, los cuales alterarán sucesivamente su potencial y darán una nueva respuesta o valor eléctrico a la dendrita, que será conducida hasta la siguiente neurona (*Idem*; Freeman J., Skapura D. 1992).

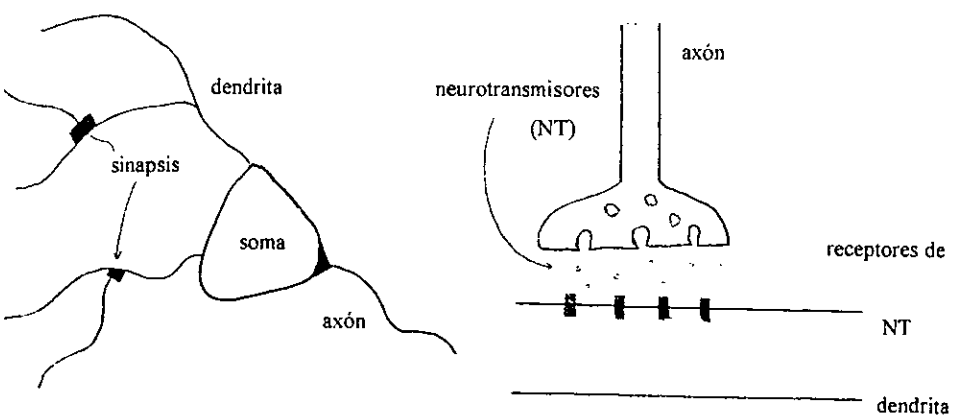


Figura 13. Las características principales de una neurona real y el esquema de la sinapsis.

[1] Esta respuesta eléctrica también es conocida como "disparo". El disparo es un fenómeno "todo o nada" resultante de la suma de los efectos en la dendrita y soma.

Teniendo una idea de cómo funcionan las neuronas podríamos simular su trabajo de la siguiente manera: la neurona es la unidad básica del cerebro, constituye una unidad de procesamiento lógico análogo y tiene varias entradas cuyos impulsos son sumados de cierta manera; si dichos impulsos son suficientes, la neurona se activa y "dispara"; de lo contrario, permanece en un estado inactivo, como se ilustra en la figura 14.

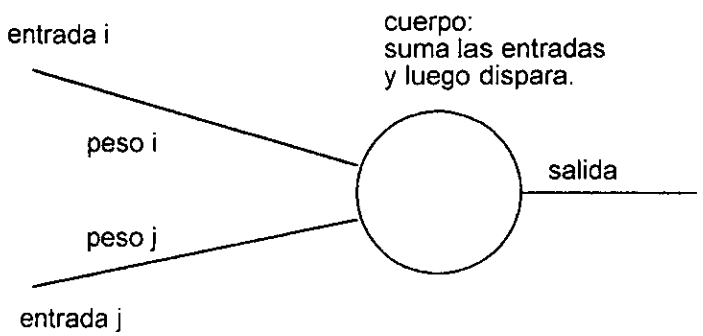


Figura 14. Esquema general de la neurona elemental.

Si tenemos $n$ entradas entonces tendremos $n$ pesos asociados. El modelo de neurona elemental calcula la suma ponderada de las entradas, lo que puede expresarse como: $\sum_{i=1}^{n} w_i x_i$ donde $w_i$ es el peso que corresponde a esta entrada o rasgo y $x_i$ es la componente i-ésima del vector de entrada. Esta suma tiene que ser comparada sucesivamente a un valor intrínseco de la neurona, el umbral de disparo ( $\theta$). El proceso de disparo se logra por la siguiente comparación, en donde los valores 0 y 1 representan las salidas de una neurona no activada y activada respectivamente (Beale R., Jackson T. 1990):

$\sum_{i=1}^{n} w_i x_i > \theta \quad \rightarrow \quad 1$ dispara

$\sum_{i=1}^{n} w_i x_i < \theta \quad \rightarrow \quad 0$ no dispara

De la misma forma, el umbral de disparo puede sacarse de la suma ponderada y comparar el valor resultante con respecto a 0; si el resultado es positivo mandará un 1 como salida; de otra manera, un 0. Otra forma de lograr este efecto es sacar el disparo del cuerpo del modelo de neurona conectándolo a una entrada extra fijada para que esté activada todo el

tiempo y entonces llamamos $-\theta$ al sesgo neuronal (*Idem*). Ambas formas del modelo se muestran en la figura 15.

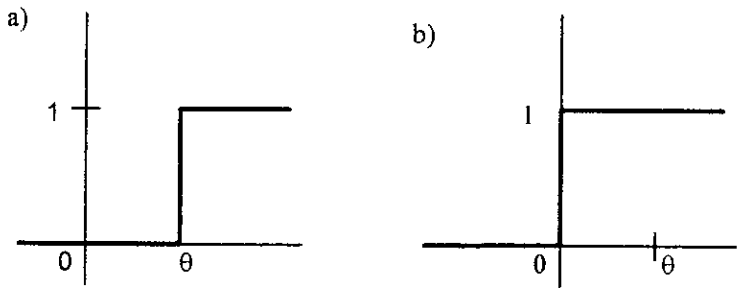


Figura 15. Función de disparo: a) en θ, b) en 0. También es conocida como función de "paso" o función "escalón".

Denominando a la salida "y" se tiene $y = f_h\left(\sum_{i=1}^{n} w_i x_i - \theta\right)$, donde $f_h$ es una función escalón y $f_h(x) = 1$ si $x>0$, $f_h(x) = 0$ si $x \leq 0$. Si consideramos $-\theta$ como otra entrada, esto equivaldría a: $y = f_h\left(\sum_{i=0}^{n} w_i x_i\right)$ con $w_0 = -\theta$ y $x_0 = 1$, lo que se puede esquematizar como en la figura 16.[2]

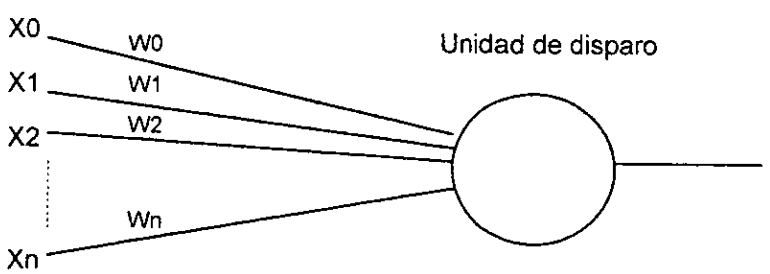


Figura 16. Modelo de McCulloch y Pitts.

[2] Se trata del modelo de McCulloch y Pitts, propuesto en 1943. A su vez Frank Rosenblatt llamó al modelo neuronal conectado de manera simple "perceptrón" (Beale R., Jackson T. 1990).

El aprendizaje en la neurona elemental es una variante de la propuesta por Donald Hebb en 1949, conocida como la "regla de premio-castigo". De esta manera, como el aprendizaje es guiado por el conocimiento de lo que se quiere obtener, se le conoce como aprendizaje supervisado.

### 3.1.1 Algoritmo de aprendizaje de la neurona elemental.

En el capítulo anterior se había pospuesto la adaptación de los pesos para obtener la inclinación del vector que formaría una frontera de decisión óptima. Veamos entonces el algoritmo de aprendizaje para el modelo más elemental que se pudiera adoptar: [3]

1. Asignar los pesos y el umbral de disparo ($\theta$) iniciales. Definir $w_i(t)$ con $0 \le i \le n$, $w_0 = -\theta$ y $x_o = 1$, $w_i(0)$ valores al azar y preferentemente pequeños.
2. Presentar la entrada conocida $(x_0, x_1, ...., x_n)$ con su respectiva salida esperada *d(t)*.
3. Calcular la salida $y(t) = f_h\left(\sum_{i=0}^{n} w_i(t) x_i(t)\right)$.
4. Adaptar los pesos. Si resultó correcto $w_i(t+1) = w_i(t)$, de lo contrario:

   1) salida = 0, *d(t)* = 1 $\qquad w_i(t+1) = w_i(t) + x_i(t)$

   2) salida = 1, *d(t)* = 0 $\qquad w_i(t+1) = w_i(t) - x_i(t)$

La adaptación de los pesos se puede hacer más gradual, para evitar grandes saltos innecesarios, con el siguiente cambio: $w_i(t+1) = w_i(t) + \eta\, x_i(t)$, donde $0 \le \eta \le 1$.

La limitación de este modelo estriba en los problemas de separabilidad no lineal, los cuales se resuelven con el perceptrón multicapa, propuesto en 1986 por Rumelhart y McClelland (*Idem*), discutido en la siguiente sección.

[3] Hay muchos tipos de algoritmos de aprendizaje. Otro es el propuesto por Widrow y Hoff, el cual calcula la diferencia entre la suma ponderada y la salida esperada y lo llama error; el ajuste de pesos del paso 4 del algoritmo original será hecho entonces en proporción a este error (*Idem*), *i.e.*: $\Delta = d(t) - y(t)$, con $w_i(t+1) = w_i(t) + \eta\, \Delta x_i(t)$, donde *d(t)*= 1 si es del primer tipo y *d(t)*= 0 si es del segundo.

### 3.2 El perceptrón multicapa.

Un primer intento de resolver el problema de la separabilidad no lineal sería usar más de un perceptrón, cada uno de ellos orientado a identificar pequeñas secciones linealmente separables de las entradas, para después combinar sus salidas con las entradas de otro perceptrón, lo que produciría una indicación final de la clase a la que pertenece la entrada. Tratemos de observar esto en la solución al problema XOR[4]: como se puede ver en la figura 17, el primer perceptrón detecta la presencia del patrón correspondiente a (0,1) mientras que el segundo detecta la de (1,0).

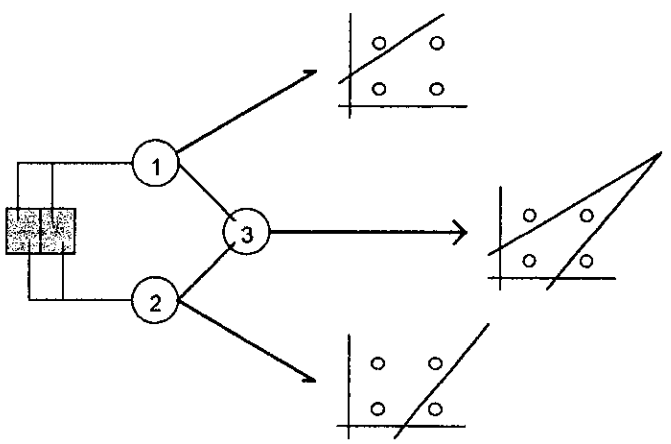


Figura 17. La combinación de perceptrones resolviendo el problema XOR.

Combinar ambos le permite al tercer perceptrón clasificar la entrada correctamente. Sin embargo, un arreglo de perceptrones de esta manera no tendrá la capacidad de aprender ya que cada neurona procesa de manera individual y capas posteriores procesan las respuestas de las anteriores sin saber cuáles de las entradas reales estaban activadas y cuáles no (*Idem*). El aprendizaje se refiere al refuerzo de las conexiones entre entradas y unidades activadas, por lo que es imposible realizar este reforzamiento cuando las unidades de salida desconocen a las de entrada por tener procesadores intermedios. Por otra parte, la función

[4] La función lógica XOR tiene dos entradas a partir de las cuales se procesará o no una salida. El primer caso sucede (comúnmente expresado como '1' ) si una de las dos entradas es 1. Por su parte, no producirá nada (expresado con '0') si ambas entradas son 1 o ninguna de ellas lo es (*Idem*).

de disparo remueve la información que se necesita para aprender, la red queda incapacitada para determinar cuáles pesos deberían incrementarse y cuáles no, de aquí que se puedan hacer cambios para producir respuestas más adecuadas (*Idem*).

En el caso de usar la función escalón, el problema se resuelve usando una no-linealidad diferente. Si la función escalón ahora posee en medio una región inclinada que dé algo de información de las entradas, se podrán determinar los pesos a ajustar y así la red podrá aprender. La entrada no es ahora simplemente activa o inactiva sino que cae dentro de un rango limitado; así, la salida de la neurona puede relacionarse a sus entradas en una forma más útil e informativa (*Idem*; Hassoun M. 1995). Tanto la función escalón modificada como la sigmoide son representadas en la figura 18.

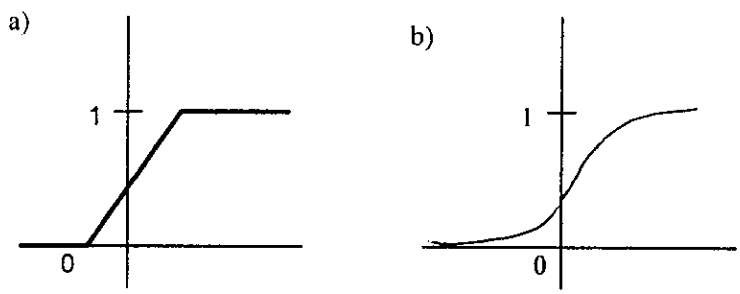


Figura 18. Función de disparo: a) disparo lineal entre límites, en otro caso vale 0 o 1, b) disparo sigmoidal.

Se tendrá que usar una función de disparo no lineal ya que capas de perceptrones con funciones lineales no son más poderosas que una sola capa bien escogida (Beale R., Jackson T. 1990). Entonces, se tomará como modelo nuevo uno con tres capas: una de entrada, una de salida y una intermedia, llamada escondida, la cual no está conectada directamente a ninguna de las dos anteriores, como en la figura 19. Las unidades en la capa de entrada servirán para distribuir los valores recibidos en la siguiente capa y éstas no desempeñarán sumas ponderadas o disparos. Por su parte, cada unidad en la capa escondida y la de salida es como un perceptrón, a excepción de que se usan la función sigmoide en lugar de la escalón.

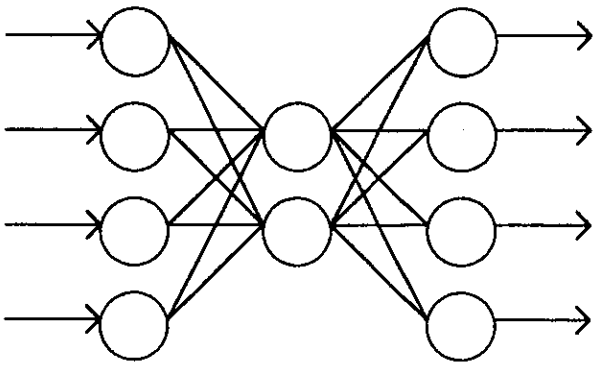

Figura 19. El perceptrón multicapa.

Obviamente una red de esta naturaleza necesitará diferentes algoritmos y reglas de aprendizaje, causa de la siguiente sección.

### 3.2.1 Regla y algoritmo de aprendizaje del perceptrón multicapa.

En 1986, Rumelhart, McClelland y Williams llamaron a la necesaria regla de aprendizaje "delta generalizada" o de "retropropagación". La operación de dicha red es similar a la del perceptrón simple ya que, después de mostrarle un patrón y calcular su respuesta, se comparará con la respuesta esperada y así se podrán alterar los pesos y obtener una respuesta más adecuada (*Idem*; Freeman J., Skapura D. 1992; Hassoun M. 1995). Si: $E_p$ es la función de error para el patrón $p$, $t_{pj}$ es el blanco de salida para el patrón $p$ en el nodo $j$, $o_{pj}$ es la salida actual del patrón $p$ en el nodo $j$ y $w_{ij}$ es el peso del nodo $i$ al nodo $j$, se define: $E_p = \left(\frac{1}{2}\right)\sum_j \left(t_{pj} - o_{pj}\right)^2$ y la activación de cada unidad j para el patrón p como $net_{pj} = \sum_i w_{ij} o_{pi}$. La salida de cada unidad $j$ es la función de disparo $f_j$ sobre la suma ponderada, *i.e.* $o_{pj} = f_j\left(net_{pj}\right)$.

Por la Regla de la Cadena se puede ver que $\frac{\partial E_p}{\partial w_{ij}} = \frac{\partial E_p}{\partial net_{pj}} \frac{\partial net_{pj}}{\partial w_{ij}}$, por lo tanto:

$\frac{\partial net_{pj}}{\partial w_{ij}} = \frac{\partial}{\partial w_{ij}} \sum_k w_{kj} o_{pk} = \sum_k \frac{\partial w_{jk}}{w_{ij}} o_{pk} = o_{pi}$ ya que $\frac{\partial w_{kj}}{\partial w_{ij}} = 0$ , excepto cuando $k=i$ , caso en el que la derivada parcial es igual a 1.

Se puede definir el cambio en el error como una función del mismo en las entradas de la red, unidad por unidad, como: $-\frac{\partial E_p}{\partial net_{pj}} = \delta_{pj} \Rightarrow -\frac{\partial E_p}{\partial w_{ij}} = \delta_{pj} o_{pi}$. Entonces, disminuir el valor de $E_p$ significa hacer cambios en los pesos proporcionales a $\delta_{pj} o_{pi}$, *i.e.* $\Delta_p w_{ij} = \eta \delta_{pj} o_{pi}$. Ahora bien, para disminuir $E$ se necesita conocer cada unidad: $\delta_{pj} = \frac{\partial E_p}{\partial net_{pj}} = -\frac{\partial E_p}{\partial o_{pj}} \frac{\partial o_{pj}}{\partial net_{pj}}$, pero $\frac{\partial o_{pj}}{\partial net_{pj}} = f_j'(net_{pj})$.

Ahora podemos hacer $\frac{\partial E_p}{\partial o_{pj}} = -(t_{pj} - o_{pj})$, entonces $\delta_{pj} = f_j'(net_{pj})(t_{pj} - o_{pj})$.

Esto es útil para las unidades de salida ya que se conocen tanto las $t_{pj}$ como las $o_{pj}$. No obstante, si $j$ no es una unidad de salida:

$$\frac{\partial E_p}{\partial o_{pj}} = \sum_k \frac{\partial E_p}{\partial net_{pk}} \frac{\partial net_{pk}}{\partial o_{pj}} = \sum_k \frac{\partial E_p}{\partial net_{pk}} \frac{\partial}{\partial o_{pj}} \sum_i w_{ik} o_{pi} = -\sum_k \delta_{pk} w_{jk}.$$

Finalmente: $\delta_{pj} = f_j'(net_{pj}) \sum_k \delta_{pk} w_{jk}$.

Esta ecuación representa el cambio en la función error con respecto a los pesos de la red, lo que nos da la oportunidad de tener un método que reduzca el error sin falla. El error se calcula primero en las capas de salida y se va pasando a capas anteriores para permitir que cambien los valores de los pesos de sus conexiones, de ahí su denominación como "red de retropropagación".[5]

Como se había mencionado, ahora se usará una función sigmoide, la cual se define como: $f(net) = \frac{1}{1 + e^{-knet}}$ con $0 < f(net) < 1$ , donde $k$ es una constante positiva que controla el despliegue de la función, *i.e.* su nivel de empinamiento. Para valores cercanos a 0 la pendiente será muy pronunciada y la función cambiará rápidamente mientras que se dará el caso contrario en aquellos cuyo valor absoluto sea relativamente grande.

[5] La notación utilizada, por simplicidad, fue extraída de (Beale R., Jackson T. 1990). Pueden encontrarse algunas diferentes en (Freeman J., Skapura D. 1992; Hassoun M. 1995).

$$\text{si } O_{pj} = f(net) = \frac{1}{1+e^{-knet}} \text{ entonces } f'(net) = \frac{ke^{-knet}}{(1+e^{-knet})^2} = kf(net)(1-f(net)) = k\,O_{pj}(1-O_{pj})$$

Ahora se puede dar forma al nuevo algoritmo de aprendizaje:

1. Asignar los pesos y umbrales de disparo iniciales con valores al azar y de preferencia pequeños.
2. Presentar la entrada $X_p = (x_0, x_1, ...., x_{n-1})$ y su salida esperada $T_p = (t_0, t_1, ...., t_{m-1})$, donde $n$ es el número de nodos de entrada y $m$ el de nodos de salida. Para la clasificación, $T_p$ está compuesto de ceros a excepción de un 1, que corresponde a la clase a la que $X_p$ pertenece. Fijar $w_0 = -\theta$, $x_0 = 1$.
3. Calcular para cada capa $y_{pj} = f\left(\sum_{i=0}^{n-1} w_i x_i\right)$ y pasarlo como entrada a la siguiente capa. La capa final dará $o_{pj}$.
4. Adaptar pesos; empezando por la capa de salida y cubriendo las anteriores sucesivamente $w_{ij}(t+1) = w_{ij}(t) + \eta\delta_{pj}o_{pj}$, en donde:

   a) Para unidades de salida $\delta_{pj} = ko_{pj}(1-o_{pj})(t_{pj}-o_{pj})$.

   b) Para unidades escondidas $\delta_{pj} = ko_{pj}(1-o_{pj})\sum_k \delta_{pk}w_{jk}$

y la suma se realiza en todos los nodos $k$ en la capa que está encima del nodo $j$.

Como se vio, la red computa un error (o función de energía) que representa la diferencia existente entre la salida de la red y su salida esperada. Diferencias grandes corresponden a energías grandes mientras que diferencias pequeñas al caso contrario; entonces se toma la energía como función de los pesos y entradas de la red. Por lo general, se pueden ajustar los pesos de la red y puede haber muchos de ellos, dando una función multidimensional energética, la cual no puede ser graficada.

Es más fácil entender el caso multidimensional mediante analogías de la situación en tercera dimensión. La superficie de energía es un paisaje con colinas, picos y valles; los puntos de mínima energía corresponden a los valles y los de máxima energía a los picos. Entonces, la regla delta generalizada pretende minimizar la función de error E ajustando los pesos para que correspondan a aquellos en los que la superficie de energía es mínima. Esto se lleva a cabo por el método del gradiente, donde la función de energía es calculada y los

cambios se hacen en la dirección del descenso más empinado. Por esta razón, cada posible solución se representa como un hoyo o valle en el paisaje, *i.e.* los valles de atracción representan las soluciones a los valores de los pesos que producen la salida correcta para una entrada dada (Beale R., Jackson T. 1990).

### 3.3 El perceptrón multicapa como clasificador.

Ahora veamos la manera en la que se utilizará un perceptrón multicapa para el problema de clasificación planteado. Consideremos una red con tres perceptrones como en la figura 20:

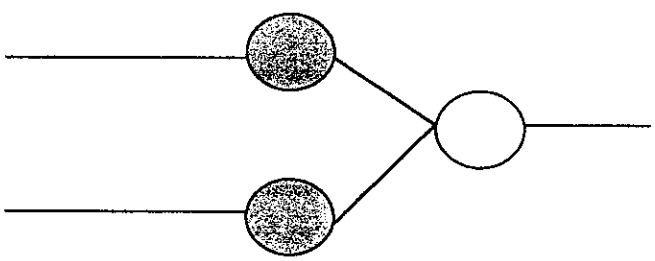

Figura 20. Dos perceptrones producen la entrada a un tercero.

Si la unidad de la segunda capa tiene su umbral de disparo determinado para activarse únicamente cuando las dos unidades de la primera capa lo permitan, *i.e.* estén activas, se dice que está desempeñando funciones lógicas y operacionales. En pocas palabras: como cada unidad de la primera capa define una frontera de decisión en el espacio de patrones, la segunda capa produce una clasificación basada en la combinación de las fronteras de la primera capa (*Idem*), como puede verse en la figura 21.

No obstante, pueden usarse más de dos unidades en la primera capa haciendo así que todas las regiones producidas sean convexas.[6] El añadir más unidades en la primera capa permitirá definir más y más bordes, por lo tanto el número total de lados que se pueden tener en las regiones será a lo más igual al número de unidades en la primera capa, manteniendo las regiones definidas como convexas. Sin embargo, añadir otra capa de

[6] Una región convexa es aquélla en la que cualquier punto puede ser conectado con cualquier otro por una línea recta que no cruza los bordes de la región.

perceptrones se traduce en que las unidades de esta capa recibirán como entrada cortezas convexas y no líneas y las combinaciones de éstas no necesariamente son convexas, por lo tanto tres capas de perceptrones pueden formar figuras arbitrariamente complicadas y son capaces de separar cualesquiera tipos de clases (*Idem*).

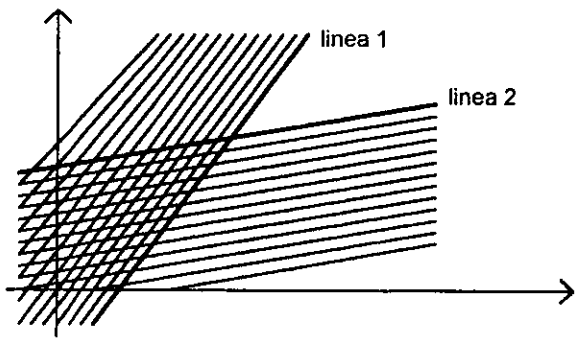


Figura 21. Región de soluciones factibles producida por la combinación de dos perceptrones.

El Teorema de Representación de Kolmogorov demuestra que cualquier clasificación hecha con cuatro capas puede ser hecha con tres, razón por la que nunca necesitaremos más de las ya citadas. Este teorema limita el número de capas que son necesarias para representar una función arbitraria pero desafortunadamente no nos indica cuántas unidades necesita la red, cómo deben ir conectadas o cómo deben establecerse los pesos entre ellas (*Idem*; Hassoun M. 1995).[7]

Para terminar, podría incluirse que las redes neuronales son eficientes en casos de interpolación y los son muy poco si el caso es una extrapolación. En la presentación de un patrón jamás procesado pero representable como una mezcla de dos patrones previamente procesados, la red lo clasificará como un ejemplo del patrón predominante. No obstante, si el patrón no corresponde a alguno similar procesado por la red anteriormente, la clasificación será mucho más pobre (Beale R., Jackson T. 1990).

[7] La literatura en este punto es inconsistente. Lo que en esta tesis se llama una red de tres capas puede ser visto de otra manera: una red recibe cierto número de entradas que son distribuidas por una capa de nodos de entrada, la cual no pondera ni activa nada. Estas entradas pasan entonces a la primera capa (aquélla que produce líneas clasificatorias), cuya salida pasa a la segunda capa (cortezas convexas) y sucesivamente a la tercera (formas arbitrarias). Al contar el número de capas activas, ésta es una red de tres capas pero, si se incluye la capa inactiva de entradas, se le puede considerar como de cuatro (Beale R., Jackson T. 1990).

## Capítulo IV. Modelo propuesto.

El trabajo para probar a utilidad del modelo de red neuronal en la identificación de estilos genómicos, objeto de esta tesis, puede dividirse en tres fases. La primera consistió en escoger y allegarse la información de entrada, *i.e.* las secuencias de nucleótidos. La segunda, elaborar un programa en lenguaje C para procesar las secuencias de entrada y presentar a la información en un vector de dimensión *n*, que alimentara a la red (Barrett M., Wagner C. 1996; Oualline S. 1997). La tercera, consistió en construir la red neuronal que identificara las secuencias de los organismos. A continuación se describe detalladamente cada etapa.

### 4.1 Características de la información de entrada, su procesamiento y la red neuronal.

La información de entrada está compuesta por secuencias de nucleótidos tomadas de GenBank (http://www.ncbi.nlm.nih.gov/Entrez). Por simplicidad, se escogieron secuencias de dos organismos: *Escherichia coli* y *Mus musculus*. El primero es una bacteria comúnmente utilizada en investigación genética; el segundo, el típico ratón casero. Esta selección se consideró suficientemente ilustrativa pues un organismo es procarionte y el otro eucarionte; entonces, se podría esperar que contengan patrones significativamente distintos debido a sus estilos genómicos. De ser así, los organismos ocuparían regiones muy distintas y bien diferenciadas en el espacio de secuencias de DNA (Miramontes P. *et al.* 1995).

Con dichas muestras se crearon dos tipos de conjuntos: los de entrenamiento de la red y los de prueba.[1] Para ambos se seleccionaron aleatoriamente genes completos o parciales de longitud mayor o igual a 100 bases y menor o igual a 1100 bases. Dichos genes debían ser porciones completamente codificables y se eliminaron de las muestras los artificiales o clonados, oncogenes, fagos y plásmidos. La utilización de genes artificiales o clonados

[1] Las redes neuronales por lo general necesitan de un conjunto de validación, el cual equivale a mezclar el de entrenamiento y tratar de encontrar una combinación que sea más representativa. En esta tesis no se usó un conjunto de validación dado que en el primer entrenamiento se obtuvo una red lo suficientemente robusta y eficiente al identificar secuencias de diferentes conjuntos de prueba.

haría que la muestra fuera potencialmente redundante y la utilización de cualquiera de los otros grupos incluiría secuencias no representativas de los patrones que exhiben los organismos.

El conjunto de entrenamiento constó de 80 secuencias de cada organismo y el de prueba constó de 20. Como se quisieron contrastar los resultados obtenidos con esta primera prueba, se creó un segundo conjunto con 60 secuencias de cada organismo, del cual se presentan resultados individuales y uno global (juntando ambas muestras). Cabe destacar que las muestras fueron elegidas aleatoriamente y posteriormente examinadas para comprobar la inexistencia de elementos iguales.

El programa en C es un procesador de secuencias. Lo primero que realiza es la apertura y lectura del archivo que contenga las secuencias de DNA previamente obtenidas. Después calcula las frecuencias de ocurrencia de los 16 dinucleótidos posibles, los coeficientes $d_{YR}, d_{WS}, d_{MK}$, el índice de información de Shanon (S) y la función de información mutua para distancias entre nucleótidos 1, 2 y 3 (M(1), M(2) y M(3)), todos los cuales serán introducidos, uno a uno, en las entradas de un vector de dimensión 23 que será representativo de cada una de las secuencias de la muestra.

Pero veamos detalladamente lo que significan y las razones por las cuales se eligió cada uno de ellos. Las unidades mínimas capaces de medir la distribución de los nucleótidos son las frecuencias de los dinucleótidos, ya que definen características importantes del DNA local como los ángulos formados entre los nucleótidos y la acumulación energética (Miramontes P. *et al.* 1995). No obstante, la falta de homogeneidad que surge a partir de la distribución de la última característica depende significativamente (como mera aproximación) de los pares de bases débiles y fuertes (W,S). De la misma manera, la distribución de purinas y pirimidinas (R,Y) influye en la distancia de fosfatos contiguos. Por su parte, como se había explicado en el capítulo I, las bases de tipo M contienen un grupo aminado en el surco mayor de la doble hélice mientras que los del grupo K, por su cualidad cetónica, promueven otro tipo de cambios estructurales. La naturaleza binaria de dichas causantes de las variaciones estructurales dio pie a crear los siguientes índices, con la finalidad de expresar la falta de homogeneidad en el DNA:

$$d_{YR} = \frac{N_{RR}N_{YY} - N_{RY}N_{YR}}{N_R N_Y} \qquad d_{WS} = \frac{N_{WW}N_{SS} - N_{WS}N_{SW}}{N_W N_S} \qquad d_{MK} = \frac{N_{MM}N_{KK} - N_{MK}N_{KM}}{N_M N_K}$$

donde $N_{\alpha\beta}$ es el número de dinucleótidos de la forma αβ y $N_\gamma$ es el número de nucleótidos γ. α, β, γ = Y, R, W, S, M, K (*Idem*).

Entre más sesgada sea la secuencia, más informativa será. En este sentido, los índices tendrán valores más altos mientras su sesgo tienda más hacia la segregación. Se ha visto que, por lo general, los genes de los organismos procarióticos tienen valores positivos en dichos índices mientras se da lo contrario en los eucarióticos. Así, se identificó cada secuencia con el vector $(d_{YR}, d_{WS}, d_{MK})$ con un espacio de imágenes en $\Omega = [-1,1]^3$. Sin embargo, dado que el tercer índice puede conocerse a partir de los otros dos, sus creadores restringieron el análisis completo a los dos primeros (*Idem*).[2] Se prefirió utilizar los tres índices explicados anteriormente, aún cuando la medida de uso de los codones ha sido una de las herramientas más utilizadas para análisis secuencial en DNA (Fickett J., Tung C. 1991; Fickett J. 1996), pues brindan una medida especifica para cada gen en un organismo y así conforman una herramienta ideal para evaluar los efectos del uso de los codones.

Por su parte, el índice de información de Shanon (*S*) y la función de información mutua (*M(d)*) se definen de la siguiente manera:

$$S = -\sum_\alpha P_\alpha Ln(P_\alpha) \quad \text{y} \quad M(d) = \sum_\alpha \sum_\beta P_{\alpha\beta}(d) Ln \frac{P_{\alpha\beta}(d)}{P_\alpha P_\beta} \quad \text{con} \quad \alpha, \beta = a, c, g, t.$$

La información mutua es la medida de dependencia entre dos variables; un indicador de la predictibilidad de una variable a partir del conocimiento de la información acumulada de la otra. Una ventaja de la función de información mutua sobre la de correlación (comúnmente usada) es que la primera mide la dependencia general mientras la segunda únicamente señala la lineal. Sin embargo, la diferencia más importante para el caso en cuestión es su utilización en secuencias simbólicas y no únicamente numéricas. (Li, W.; 1990)

Con todos los resultados explicados, finalmente el programa presenta todos los coeficientes para cada una de las secuencias en vectores de dimensión 23 (cuyas componentes representan cada uno de los índices previamente explicados) y los guarda en el archivo que se le indique. Cada componente del vector toma valores en el intervalo [-1, 1], a excepción del índice de Shanon, el cual se encuentra en los reales positivos (y particularmente para las secuencias utilizadas, en el intervalo [1.33, 1.4]). Una variante del programa, descrita más

adelante, calcula únicamente las componentes de un vector de dimensión 6, siendo éstas $d_{YR}, d_{WS}, d_{MK}$, M(1), M(2) y M(3); en el supuesto de que las tres primeras contienen la información de las frecuencias de dinucleótidos y las tres siguientes son el caso general del resultado que pudiera obtenerse del índice de Shanon. En pocas palabras se redujo la información para evitar posibles redundancias.[3]

Las secuencias, introducidas y procesadas, crean otro archivo que contiene los vectores de patrones y una vez añadidas las leyendas "[INPUTS] 23 [OUTPUTS] 1" se crean las entradas que alimentarán directamente a la red neuronal.

Ahora bien, la red neuronal que clasificará las secuencias de acuerdo a los organismos escogidos es una de retropropagación, lo cual implica que debe tener más de una capa de neuronas. La red con la que se trabajó es la Quiknet v2.23 (http://www.kagi.com/cjensen/), la cual contiene varios tipos de algoritmos de entrenamiento. Para esta tesis únicamente se experimentó con cuatro de ellos, descritos muy generalmente como:

a) Retropropagación en línea. Es aquel que actualiza los pesos después de la presentación de cada uno de los patrones de entrada.
b) Retropropagación aleatoria en línea. En esta modalidad, el orden de los patrones de entrada es aleatorio justo antes de cada "época",[4] así puede considerarse como un aprendizaje estocástico.
c) Retropropagación "en tanda". Realiza la actualización de los pesos después de cada época.
d) Quickprop. La función de error con respecto a los pesos ($E(w)$) es simulada con una parábola que abre hacia arriba, su cambio en la pendiente para cada peso no es afectado por el cambio de los demás y la actualización se da con la siguiente regla:

$$\Delta W(t) = \frac{S(t)}{S(t-1) - S(t)} \Delta W(t-1) - \eta S(t)$$

[2] Algo sorprendente de dicho análisis fue la obtención de grupos nítidos de organismos en pequeñas porciones conectadas de $\Omega$ definidas a partir de las distribuciones de los genes de diferentes organismos.

[3] A diferencia del análisis llevado a cabo por sus creadores, el índice $d_{MK}$ fue tomado en cuenta en este trabajo ya que representa cualidades importantes de los dinucleótidos. Sobre todo, estas cualidades serán significativas en el uso de vectores de dimensión 6, ya que su uso elimina el de dichas frecuencias.

[4] En esta tesis se usa el término "época" como sinónimo del número de pasos de entrenamiento que tuvo que llevar a cabo la red, a diferencia de la terminología comúnmente utilizada en redes neuronales en la cual "época" se refiere al número de validaciones realizadas en una red para que ésta pueda considerarse eficiente.

donde $\eta$ es la tasa de aprendizaje, el numerador representa la derivada del error con respecto al peso y el denominador es una aproximación de la diferencia finita de la segunda derivada. Juntos resultan muy aproximados al método de Newton para minimizar una función unidimensional pero, para evitar retroceder infinitamente, se introduce una $\mu$ (el factor de máximo crecimiento) la cual responde a la relación en la que ningún cambio en los pesos será mayor a $\mu$ veces el cambio anterior (ver http://www.kagi.com/cjensen/).

En lo que respecta a la configuración de la red se escogió una tasa de aprendizaje (para todos los algoritmos) de 0.03, un margen de error de 0.01, pesos en el intervalo [-100, 100], una perturbación máxima en los pesos del 20% y una capa escondida de neuronas. Como el número de neuronas permitido por el programa en dicha capa se encuentra en el intervalo [1, 10] se decidió experimentar con 5 y 10 neuronas para todos los casos.

Ahora bien, las funciones de activación con las que se podía trabajar con este programa eran la logística, la tangente hiperbólica, la lineal y la gaussiana. Únicamente se trabajó con las tres primeras debido a que la gaussiana, en la configuración de red establecida, no permitió el aprendizaje de los patrones. También se fijó un límite de saturación de la red de 80%, el cual refleja el porcentaje mínimo de patrones en cada época que deben llenar la red antes de considerarla saturada.[5] Por último, se fijó en 50,000 el límite de "épocas" tanto para el entrenamiento como para la prueba a fin de evitar procesos infinitos.

### 4.2 Comportamientos y resultados obtenidos.

Primero se experimentó con el conjunto de 40 muestras procesadas (20 de cada organismo), usando los 23 índices calculados y 5 neuronas en la capa escondida. Los resultados se encuentran en la tabla 2.

Como se puede observar, todas las redes dan el mismo resultado a excepción de la "Quickprop" con función de activación logística. Al examinar el conjunto de prueba, la red creó un archivo de resultados los cuales, típicamente, fueron muy aproximados a la salida esperada en caso de acierto.

[5] Una neurona se considera saturada cuando un porcentaje mayor al fijado produce una salida que rebase el 99% de su valor máximo. En este programa, se reducen automáticamente en un 90% todos los pesos introducidos a una red saturada.

Quiero explicar esto detalladamente. En el conjunto de entrenamiento se incluye una dimensión extra (en este caso la número 24) la cual corresponde al grupo de clasificación. Se etiquetó con el símbolo "1" el hecho de pertenecer al grupo de *Escherichia coli* y con "2" al de pertenecer al de *Mus musculus*. Así, por ejemplo, si la red daba una respuesta de 1.0001 se consideró que se trataba de una aproximación del "1" referente a la bacteria mientras que un 1.98 se consideró como una aproximación al "2" de los ratones. Como medida completamente arbitraria, se fijó la frontera de división de los grupos en 1.5.

| Algoritmo<br>Función | R. en línea. | R. aleatoria en línea. | R. "en tanda". | Quickprop. |
|---|---|---|---|---|
| Logística | Error máximo: 0.0134104<br>Probabilidad de acierto: 0.9 | Error máximo: 0.0143121<br>Probabilidad de acierto: 0.9 | Error máximo: 0.0141476<br>Probabilidad de acierto: 0.9 | Probabilidad de acierto: 0.9 |
| Tanh | Probabilidad de acierto: 0.9 | Probabilidad de acierto: 0.9 | Probabilidad de acierto: 0.9 | Probabilidad de acierto: 0.9 |
| Lineal | Probabilidad de acierto: 0.9 | Probabilidad de acierto: 0.9 | Probabilidad de acierto: 0.9 | Probabilidad de acierto: 0.9 |

Tabla 2. Resultados para una muestra de tamaño 40 usando 23 índices.

Como se puede ver, el desempeño de las redes es muy parecido. La excepción es la de las redes con función lineal y los tres tipos de "Quickprop", los cuales obtuvieron resultados que, en general, no necesitaron de aproximación alguna. Por otra parte, al usar la función logística se llegó a los mismos resultados que con la tangente hiperbólica (a excepción de "Quickprop"), con la diferencia de haber usado un mayor número de épocas para la fase de

entrenamiento bajo esta modalidad. Más explícitamente, para llegar a un error máximo de 0.01 se necesitaron aproximadamente 22,000 épocas para la función lineal y de tangente hiperbólica mientras que la logística nunca llegó a un error máximo tan reducido, aún rebasando en todos los casos entrenamientos de 50,000 épocas.

Una segunda modalidad, con el mismo conjunto de entrenamiento y prueba, es la utilización de 10 neuronas en la capa escondida en lugar de las 5 predeterminadas por el programa. En el caso de las 10 neuronas (y en sentido contrario a lo que se hubiera esperado antes de realizar el experimento) se obtuvieron las mismas probabilidades de aciertos para la mayoría de los casos a excepción de los "Quickprop", los cuales aumentaron su probabilidad de acierto a 0.925. La única diferencia general es que los resultados requirieron de una menor aproximación y previamente, en el entrenamiento, se dio la convergencia de error máximo en un número de épocas relativamente menor.

Quiero ahora presentar la segunda etapa del análisis, aquélla en la que se experimenta únicamente con 6 de los 23 índices calculados, como se explicó anteriormente. Las frecuencias de ocurrencia de los dinucleótidos fueron eliminadas de las muestras dado que gran parte de la información que brindan se encuentra contenida en las 3 primeras nuevas entradas ($d_{YR}, d_{WS}, d_{MK}$); lo mismo ocurrió con el índice de Shanon, el cual no permitió identificar diferencias entre los organismos en el conjunto de entrenamiento y los de prueba.

Tanto en el caso en el que se utilizaron 5 neuronas en la capa escondida como en el que se usaron 10, el entrenamiento de la red tardó mucho más que en el caso anterior, razón por la que se fijó su límite de épocas en 100,000. Curiosamente el conjunto de entrenamiento nunca pudo llegar a minimizar el error máximo hasta 0.01, a pesar de haber usado en la mayoría de los casos más del doble de épocas. Para ser más precisa, este error se fijó en todos y cada uno de los entrenamientos en 1. Por su parte, el error promedio no alcanzó a diminuir más allá del 0.025 mientras en el análisis anterior se habían obtenido errores promedios menores a 0.001. A diferencia del experimento con vectores de dimensión 23, el usar 10 neuronas en la capa intermedia en lugar de las 5 predeterminadas se tradujo solamente en una disminución del error relativamente más rápida para el conjunto de

entrenamiento y un menor uso de la aproximación en los resultados. Los resultados de dicho análisis se encuentran en la tabla 3.

| Algoritmo / Función | R. en línea. | R. aleatoria en línea. | R. "en tanda". | Quickprop. |
|---|---|---|---|---|
| Logística | 0.95 | 0.95 | 0.95 | 0.95 |
| Tanh | 0.95 | 0.95 | 0.95 | 0.95 |
| Lineal | 0.95 | 0.95 | 0.95 | 0.95 |

Tabla 3. Resultados para una muestra de tamaño 40 usando 6 índices.

Como se puede observar, aún cuando el error máximo en el entrenamiento es de 1 (*i.e.* 100%), la probabilidad de acierto incrementó considerablemente para todos los casos. Otra diferencia con el primer análisis es que todas las probabilidades son iguales, *i.e.* los casos de "Quickprop" con función logística y lineal clasifican tan eficientemente como las demás, tanto en el caso de trabajar con 5 neuronas en la capa escondida como en el de trabajar con 10.

La tercera parte del análisis consiste en usar una muestra diferente, incluso mayor, con la finalidad de observar si la probabilidad de acierto cambia en función del tamaño muestral. Para este fin se utilizó el segundo conjunto de prueba, aquel que contiene 60 muestras de cada organismo. Este conjunto consta de vectores compuestos por 6 parámetros, los cuales son los mismos índices con los que se había experimentado en el caso anterior. Aún cuando en los entrenamientos el error máximo había sido de 1 (100%) pero el porcentaje de asignación correcta en los mismos fue del 100% (para todos los casos a excepción de "Quickprop" con 99.38% y dos de las funciones lineales con 90%), la red clasificó el nuevo conjunto de prueba asignando la misma probabilidad de acierto para todos los casos, como se muestra en la tabla 4.

| Algoritmo / Función | R. en línea. | R. aleatoria en línea. | R. "en tanda". | Quickprop. |
|---|---|---|---|---|
| Logística | 0.9583333333 | 0.9583333333 | 0.9583333333 | 0.9583333333 |
| Tanh | 0.9583333333 | 0.9583333333 | 0.9583333333 | 0.9583333333 |
| Lineal | 0.9583333333 | 0.9583333333 | 0.9583333333 | 0.9583333333 |

Tabla 4. Resultados para una muestra de tamaño 120 usando 6 índices.

Al usar una muestra mayor se pudo observar que la probabilidad de acierto aumentó ligeramente. Esto se realizó con la finalidad de eliminar sospechas que pudieran hacerse por el hecho de utilizar una sola muestra; una de ellas sería que varios de los genes de la primera muestra aleatoria no fueran representativos y por ende la probabilidad de aciertos debiera su inexactitud a esta causa. Al igual que con la muestra de 40, el uso de 5 y 10 neuronas en la capa escondida se refleja en un uso menor de la aproximación en los resultados y no en un cambio de las probabilidades de acierto; hasta cierto punto esto es bastante sorprendente ya que al usar 10 neuronas en la capa escondida, dicho número es mayor al de los índices de información de entrada a la red, por lo que se esperaría mostraran una probabilidad menor a la de los casos en los que se usan 5 neuronas.

El cuarto y último análisis surgió a partir de la pregunta ¿El uso de un vector de dimensión 6 será más eficiente para la prueba de muestras mayores?. Podría pensarse que, al cumplirse en muestras pequeñas, dicha propiedad podría prevalecer en muestras mayores, pero la experimentación en este caso contradijo la ingenua inducción. A diferencia del conjunto de prueba con 40 muestras, el nuevo conjunto (con 120 muestras y 23 índices) no presentó diferencia alguna en las probabilidades de acierto variando el número de neuronas en la capa escondida, por lo tanto "Quickprop" no pudo considerarse en general como un mejor algoritmo de entrenamiento para la identificación. Sin embargo, lo más impresionante fue el hecho de que las probabilidades de acierto, como se puede observar en la tabla 5, son

mayores que las obtenidas al clasificar la muestra de vectores de dimensión 6. Esto, en pocas palabras, indica que no necesariamente el resto de los patrones son redundantes, *i.e.* repiten la información, y por esta causa hagan menos eficiente el trabajo de la red.

| Algoritmo / Función | R. en línea. | R. aleatoria en línea. | R. "en tanda". | Quickprop. |
|---|---|---|---|---|
| Logística | 0.975 | 0.975 | 0.975 | 0.975 |
| Tanh | 0.975 | 0.975 | 0.975 | 0.975 |
| Lineal | 0.975 | 0.975 | 0.975 | 0.975 |

Tabla 5. Resultados para una muestra de tamaño 120 usando 23 índices..

### 4.3 Resultado general.

La tabla 6 muestra los resultados de una muestra conjunta, con 80 secuencias de nucleótidos por cada organismo. Las probabilidades coincidieron para las muestras provenientes de vectores tanto de dimensión 6 como 23 a excepción de la columna de "Quickprop", donde se presentaron 2 resultados, el primero de ellos simbolizando el comportamiento de la red con 5 neuronas en la capa escondida para vectores de ambas dimensiones, así como una red proveniente de conjuntos de vectores de dimensión 6 con 10 neuronas. Obviamente el segundo resultado corresponde a una red con 10 neuronas en dicha capa, proveniente de conjuntos de vectores de dimensión 23.

De aquí puede observarse que el desempeño real de las redes examinadas, para el caso de la identificación de muestras de secuencias de nucleótidos de *Escherichia coli* y *Mus musculus*, se encuentra en el intervalo de probabilidad de acierto [0.95625, 0.9625], aún cuando se hayan encontrado resultados que sobrepasan las cotas de dicho intervalo.

| Algoritmo<br>Función | R. en línea. | R. aleatoria en línea. | R. "en tanda". | Quickprop. |
|---|---|---|---|---|
| Logística | 0.95625 | 0.95625 | 0.95625 | 0.95625<br>0.9625 |
| Tanh | 0.95625 | 0.95625 | 0.95625 | 0.95625<br>0.9625 |
| Lineal | 0.95625 | 0.95625 | 0.95625 | 0.95625<br>0.9625 |

Tabla 6. Resultados generales.

Por último, quisiera aclarar la naturaleza de algunos de los genes que no fueron clasificados correctamente por las redes, cuya definición según GenBank es:

a) Mus musculus nicotinamide nucleotide transhydrogenase (Nnt) gene, exon 18.

b) Mus musculus 5-hydroxytryptamine 3 receptor B subunit precursor gene, exons 7 and 8.

c) Mus musculus nicotinamide nucleotide transhydrogenase (Nnt) gene, exon 2.

Se presentan estos tres genes ya que son aquéllos en los que hubo coincidencia al ser identificados erróneamente en varias clasificaciones. Se podría pensar que el gen de la transhidrogenasa no fuera representativo del genoma del *Mus musculus*, sin embargo forma parte de éste por lo cual ésta deducción no sería válida. Un hecho que lo confirma es que varios genes de este tipo, pero provenientes de los exones 1, 2, 15 y 17, se encuentran en la muestra y fueron clasificados satisfactoriamente. La identificación fallida de los genes de la *Escherichia coli*, por su parte, nunca se refirió a un mismo gen más de una vez. Esto fue incluido, únicamente, para ejemplificar que las redes no son un mecanismo perfecto de clasificación, pero son capaces de trabajar muy eficientemente con secuencias demasiado complicadas al analizarse con otros tipos de mecanismos. Un ejemplo claro de ello, como pudimos ver, son las secuencias de DNA.

## Conclusiones.

Las secuencias de DNA actualmente representan un campo de investigación importante y, en particular, los estudios de ellas a través de redes neuronales han crecido en número y profundidad de enfoque. A través de este trabajo de tesis se pudo observar de cerca una de las muchas aplicaciones de las redes neuronales para el problema de clasificación y, como era de esperarse, la aplicación seleccionada significó un reto considerable.

Un ejemplo de creadas anteriormente, en análisis genómico, es el modelo de Richard Mural y Edward Uberbacher que, a diferencia del modelo presentado, clasifica regiones de DNA en grupos codificables y no codificables (Mural R., Uberbacher E. 1991). Una de las características destacables dentro de su investigación, particularmente un patrón que influyó la forma de modelación realizada, es el uso de sensores (*i.e.* información procesada) en lugar de la secuencia directa. El usar información previamente sintetizada brinda ventajas considerables, como por ejemplo una mayor robustez a consecuencia de la relativa independencia de los algoritmos que la procesan (*Idem*).

Uno de los logros de esta tesis fue comprobar el alto porcentaje de clasificación correcta de las redes, debido a su procesamiento en paralelo. Esto se refleja directamente en el hecho de que, si se usara una secuencia de nucleótidos no tan representativa de los organismos dentro del conjunto de entrenamiento de la red, el modelo seguiría funcionando y así, como se había mencionado, se obtendría un indicador de su fortaleza ante el ruido o los errores en la secuencia. Aún así, quisiera señalar algunas ideas que sería recomendable se llevaran a cabo en una futura investigación.

La primera de ellas es análisis más profundo de sensibilidad del error como función del número de neuronas en la capa escondida, el tamaño de la muestra, el tamaño del conjunto de entrenamiento o la tasa de aprendizaje, así como sus posibles combinaciones.

La segunda sugerencia es analizar, de una manera más formal, el fenómeno de cambio en la probabilidad de aciertos como dependiente de la dimensión de los vectores en los conjuntos de entrenamiento-prueba, así como del número de neuronas de la capa escondida.

La tercera se deriva de la última y se refiere al análisis de tolerancia de clasificación ante errores en el conjunto de entrenamiento o prueba.

La cuarta idea se relaciona con el programa realizado en lenguaje C y se trata de realizar un cambio en las operaciones del mismo. Dado que en algunas secuencias se pueden encontrar nucleótidos representados por las letras Y, R, W, S, M, K (como se explicó en el capítulo I), sería muy conveniente que el programa leyera estos caracteres y los asignara a casos creados a partir de probabilidades condicionales, para realizar un análisis relativamente distinto.

Otra podría ser la utilización de un conjunto de validación que contenga genes clasificados erróneamente, como los de la transhidrogenasa, y así obtener rendimientos comparativos. Finalmente, las obvias extensiones de este trabajo serían la clasificación de un mayor número de tipos de organismos (en lugar de 2) y el análisis de secuencias mayores a los 1100 nucleótidos (como los proyectos de genomas completos). En un principio, la razón por la que no se incluyeron secuencias menores a las 100 bases fue que, por lo general, dichas secuencias no presentan un número suficientemente grande de patrones representativos para diferenciarlas, por lo tanto todas podrían parecer muy similares. No obstante, la creación de índices y coeficientes hipersensibles sería un reto interesante y de gran valor para el tratamiento de este tipo de casos, como también se indica en (Fickett J. 1996).

**Bibliografía**

Babloyantz, A. (1986) *Molecules, dynamics and life. An introduction to self-organization of matter*, John Wiley and Sons, U.S.A.

Barrett, Martin L.; Wagner, Clifford H. (1996) *C and UNIX. Tools for software design*, John Wiley and Sons, U.S.A.

Beale, R.; Jackson, T. (1990) *Neural computing: An introduction*, Adam Hilger, U.K.

Freeman, James A.; Skapura, David M. (1992) *Neural networks. Algorithms, applications and programming techniques*, Addison-Wesley Publishing Company, U.S.A., pp. 1-41, 89-124.

Fickett, James W. (1996) *Finding genes by computer: The state of the art*, TIG, Vol. 12, No. 8: 316-320.

Fickett, James W.; Tung, Chang-Shung. (1991) *Assesment of protein coding measures*, Nucleic Acids Research, Vol. 20, No. 24: 6441-6450.

Frank-Kamenetskii, Maxim D. (1993) *Unraveling DNA*, VCH Publishers, U.S.A.

Hassoun, Mohamad H. (1995) *Fundamentals of artificial neural networks*, The MIT Press, U.S.A., pp. 46-50, 197-233.

Hoagland, Mahlon B. (1985) *Las raíces de la vida. Genes, células y evolución*, Biblioteca Científica Salvat, Barcelona.

Kohonen, Teuvo. (1989) *Self-organization ans associative memory*, Springer-Verlag, U.S.A., pp. 185-209.

Lewin, Benjamin. (1997) *Genes VI*, Oxford University Press, U.S.A.

Lewin, Roger. (1997) *Patterns in evolution. The new molecular way*, Scientific American Library, W. H. and Freeman Co., New York.

Li, Wentian. (1990) *Mutual information functions versus correlation fuctions*, Journal of Statistical Physics, Vol. 60: 823-837.

Miramontes Vidal, Pedro Eduardo. (1992) *Un esquema de autómata celular como modelo matemático de la evolución de los ácidos nucleicos*, Tesis doctoral, U.N.A.M., México.

Miramontes P.; Medrano L.; Cerpa C.; Cedergren R.; Ferbeyre G.; Cocho G. (1994) *Structural and thermodynamic properties of DNA uncover different evolutionary histories*, Journal of Molecular Evolution, Vol. 40: 698-704.

Mural, Richard J.; Uberbacher, Edward C. (1991) *Locating protein coding regions in human DNA sequences by a multiple sensor-neural network approach*, Proceedings of the National Academy of Sciences, Vol. 88: 11261-11265.

Oualline, Steve. (1997) *Practical C programming*, 3ª edición, O'Reilly, U.S.A.

Viana Castrillón, Laura. (1998) *Memoria natural y artificial*, Colección "La ciencia para todos", Vol. 88, 3ª edición, Fondo de Cultura Económica, México.