HERRAMIENTA WEB PARA PREDECIR LA ESTRUCTURA SECUNDARIA DE PROTEÍNAS USANDO MÁQUINAS DE SOPORTE VECTORIAL

JAIME ANDRÉS COVELLI LÓPEZ
ALBERTO JAIMES VARGAS

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO MECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA
2010

HERRAMIENTA WEB PARA PREDECIR LA ESTRUCTURA SECUNDARIA DE PROTEÍNAS USANDO MÁQUINAS DE SOPORTE VECTORIAL

JAIME ANDRÉS COVELLI LÓPEZ
ALBERTO JAIMES VARGAS

Trabajo de Grado para optar por el Titulo de
INGENIERO DE SISTEMAS

Director
Bs, DEA. Alfonso Mendoza Castellanos

Codirectores
PhD. Rodrigo Torres Sáez
Ing. Darío José Delgado Quintero

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO MECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA
2010

DEDICATORIA

A Dios por acompañarme en todos los momentos de mi vida.

A mi padre por estar a mi lado y aconsejarme siempre.

A mi madre por hacer de mi mejor persona a través de sus consejos, enseñanzas y amor.

A mi hermano por estar siempre presente.

A mis compañeros y amigos por los buenos momentos.

Jaime Andrés Covelli López

DEDICATORIA

A Dios por iluminarme y guiarme en todos los momentos de mi vida.

A toda mi familia por su apoyo incondicional, su comprensión y colaboración en la consecución de esta meta.

A todas aquellas personas que de una u otra forma contribuyeron en mi vida para poder alcanzar este logro.

Alberto J.

AGRADECIMIENTOS

Los autores expresan sus agradecimientos a:

Al **Bs. DEA.** Alfonso Mendoza Castellanos, director del proyecto, por su dirección y asesoría las cuales hicieron posible la realización de este trabajo de grado.

Al **PhD.** Rodrigo Torres Sáez y al **Ing**. Darío José Delgado, codirectores del proyecto, por su constante colaboración y importantes aportes para el desarrollo del proyecto.

Al Grupo de Investigación **GIIB** y al Grupo de Investigación en Bioquímica y Microbiología de la UIS y sus integrantes por darnos la oportunidad de robustecer nuestro proceso de formación mediante el desarrollo del proyecto y por el apoyo brindado a lo largo del mismo.

RESUMEN

TITULO: HERRAMIENTA WEB PARA PREDECIR LA ESTRUCTURA SECUNDARIA DE PROTEÍNAS USANDO MÁQUINAS DE SOPORTE VECTORIAL[*].

AUTORES: JAIME ANDRÉS COVELLI LÓPEZ, ALBERTO JAIMES VARGAS[**]

PALABRAS CLAVES: Estructura Secundaria, Máquinas de Soporte Vectorial, Predicción, Proteína.

DESCRIPCIÓN

El presente proyecto tiene el propósito de llevar a la web el proceso de predicción de la estructura secundaria de una proteína, conociendo la secuencia primaria se puede codificar dicha secuencia en subsecuencias llamadas N-gramas, a su vez agregando información estadística y teniendo en cuenta las propiedades físico-químicas de los aminoácidos se puede generar información que permite establecer un conjunto de datos (Vector de Codificación) sobre el cual se hará la predicción de los motivos estructurales correspondientes a la estructura secundaria por medio del uso de máquinas de soporte vectorial, dicha predicción es un problema de multiclasificación y se resuelve a través de clasificadores binarios.

Algunos métodos computacionales han demostrado ser de gran ayuda a la hora de predecir la estructura secundaria de una proteína, para este trabajo se emplearon Maquinas de Soporte Vectorial (MSV) las cuales partir de la cadena de aminoácidos (Estructura Primaria) pueden entrenarse para que aprendan a reconocer que subsecuencias de aminoácidos pueden producir determinados motivos estructurales (Estructura Secundaria).

La herramienta se implementó en un entorno web a través de una aplicación Cliente-Servidor con la cual el usuario puede realizar la predicción de la estructura secundaria de una proteína de manera sencilla, ingresando la secuencia de aminoácidos a través de un formulario HTML y enviando la información a un Servlet que procesa la secuencia y retorna la predicción, esta herramienta web es importante debido a la disponibilidad y fácil acceso que ofrece, además es de gran ayuda a los usuarios que necesiten disponer de herramientas bioinformáticas que aporten en la obtención de resultados en sus investigaciones.

[*] Proyecto de Grado

[**] Universidad Industrial de Santander. Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingeniería de Sistemas. Director: Bs, DEA. Alfonso Mendoza Castellanos. Codirectores: PhD. Rodrigo Torres Sáez, Ing. Darío José Delgado Quintero

ABSTRACT

TITLE: WEB TOOL FOR PREDICTING THE SECONDARY STRUCTURE OF PROTEINS USING SUPPORT VECTOR MACHINES[†].

AUTHORS: JAIME ANDRÉS COVELLI LÓPEZ. ALBERTO JAIMES VARGAS**

KEYWORDS: Secondary Structure, Support Vector Machines, Prediction, Protein.

DESCRIPTION

This project aims to bring the web the process of predicting the secondary structure of a protein, knowing the primary sequence can be encoded in the sequence called N-gram subsequences, in turn adding statistical information and taking into account the physico-chemical properties of amino acids can generate information that allows a data set (Vector Coding) on which make the prediction of structural reasons for secondary structure by using support vector machines, such Prediction is a problem and solve multiclassification
through binary classifiers.

Some computational methods have proved of great help in predicting the secondary structure of a protein to this work we employ Support Vector Machines (SVMs) from which the amino acid chain (primary structure) can be trained to learn to recognize that subsequences of amino acids can produce certain structural motifs (secondary structure).

The tool was implemented in a web environment through a client-server application with which the user can predict the secondary structure of a protein simply by entering the sequence of amino acids through an HTML form and sending information to a servlet that processes the sequence and returns the prediction, this web tool is important because of the availability and easy access it offers is also helpful to users who need to provide bioinformatics tools in obtaining results research.

---

* Research Work.

** Universidad Industrial de Santander. Faculty of Engineering Physique-Mechanics.School of systems Engineering and Informatics. Director: Bs, DEA. Alfonso Mendoza Castellanos. Co-director: PhD. Rodrigo Torres Sáez, Ing. Darío José Delgado Quintero.

GLOSARIO

AMINOÁCIDO: Es una molécula orgánica que es la base estructural de las proteínas y constituida por un grupo amino (-NH2), un grupo carboxilo (-COOH; ácido) y una cadena lateral R.

FENOTIPO: Conjunto de caracteres visibles que un organismo presenta como resultado de la interacción de su genotipo y el ambiente.

GENOTIPO: Es el contenido genético (el genoma específico) de un ser vivo (individuo).

MONÓMERO: Molécula de masa pequeña que se une con otros similares para formar cadenas moleculares mayores o polímeros.

PÉPTIDO: Son un tipo de moléculas formadas por la unión de varios aminoácidos mediante enlaces peptídicos (amida)

POLIPÉPTIDO: Es el nombre utilizado para designar un péptido de tamaño suficientemente grande.

TABLA DE CONTENIDO

LISTA DE FIGURAS

LISTA DE TABLAS

LISTA DE ANEXOS

1. INTRODUCCIÓN

El uso de la computación en áreas como la bioquímica y la biología han tomado una importancia significativa durante los últimos años, la fusión de estas áreas de conocimiento ofrece sustanciales beneficios que mejoran la calidad en los campos de la investigación.

En la línea marcada por esta fusión se encuentra el problema de la predicción de la estructura secundaria de proteínas.

Brindar soluciones a este problema es importante, debido a que si se observa el objetivo de la biología moderna, es entender como el juego completo de información genética contenida en la secuencia nucleótida del genoma en una célula determina la estructura (genotipo), función, y el comportamiento de un organismo vivo (fenotipo).

En el medio de este esfuerzo científico esta la caracterización de los roles Bioquímicos y celulares de las proteínas, los engranajes en la maquinaria de la vida. En este documento se explican los conceptos teóricos y lineamientos a seguir para hacer uso de una técnica de inteligencia artificial aplicada al problema de la predicción de la estructura secundaria de proteínas.

2. PRESENTACION DEL PROYECTO

2.1 OBJETIVOS

2.1.1 Objetivo General. Implementar un algoritmo para realizar la predicción de la estructura secundaria de una proteína mediante la utilización de máquinas de soporte vectorial (MSV) el cual pueda ser accesible desde internet.

2.1.2 Objetivos Específicos

- Implementar un algoritmo de clasificación que use MSV para predecir la estructura secundaria de una proteína.

- Validar la sensibilidad y especificidad del algoritmo que se desarrolle para poder constatar la eficiencia del mismo.

- Elaborar una página web que permita interactuar con el algoritmo de predicción de la estructura secundaria para una secuencia de aminoácidos.

- Implantar el algoritmo planteado y la página en un servidor en el cual pueda tener acceso a la herramienta desarrollada a través de internet.

2.2 JUSTIFICACIÓN

La predicción de la estructura secundaria de las proteínas ha sido ampliamente estudiada en los últimos años, ya que ha generado importante información sobre las proteínas, proporcionando datos que han permitido aproximaciones importantes al entendimiento de su función y su estructura. Esta información ha permitido a los investigadores realizar grandes avances en el campo de la biomedicina, el diseño de fármacos y la proteómica.

La comunidad científica ha incrementado el uso de las ciencias de la computación y de sus tecnologías. Esta fusión, entre la biología y la tecnología computacional, ha abierto nuevas áreas de investigación y desarrollo de nuevas herramientas computaciones útiles en los campos de las ciencias biológicas y sus aplicaciones.

Teniendo en cuenta el aporte que ofrece el estudio de las proteínas, y en especial el estudio de su estructura, se ha generado un gran interés por parte de la comunidad científica y en especial de la comunidad bioinformática en cuanto a la investigación en esta área. Esto ha permitido profundizar en el desarrollo o aplicación de herramientas computacionales, de métodos teóricos y de análisis de datos, modelado matemático y técnicas de simulación para almacenar, organizar, archivar y visualizar información proteica. En este sentido, el desarrollo de herramientas enfocadas al procesamiento de información biológica, es un factor importante de desarrollo en el campo científico-biológico para un país que presenta una gran biodiversidad que debe ser estudiada y aprovechada para beneficio de la región y el país.

Con la realización de este proyecto se busca generar un aporte mediante la implementación de algoritmos de predicción orientados a ser implementados sobre una plataforma web que ofrezca a la comunidad científica nacional la oportunidad de contar con herramientas propias en materia de predicción de la estructura secundaria de proteínas. Teniendo en cuenta que en Colombia el desarrollo de este tipo de herramientas informáticas es muy reducido, y que este tipo de proyectos pueden servir como cimiento para la incursión en nuevas áreas de conocimiento y tecnologías no muy comunes en el entorno nacional.

3. METODOLOGIA Y PLAN DE TRABAJO

3.1 METODOLOGÍA

Para realizar la herramienta web que predice estructuras secundarias de proteínas se tuvo en cuenta la metodología de **Entrega por Etapas**[3], que es un modelo en el que se evita el problema del modelo en cascada de no terminar ninguna etapa del modelo hasta que esté completamente finalizado. La diferencia de este modelo con el prototipo evolutivo es que en este modelo conocemos exactamente lo que se va a construir. Este modelo funciona exactamente igual que el de cascada en las tres primeras fases y en el diseño detallado se divide por etapas. Con esta metodología se desarrollan las capacidades más importantes reduciendo el tiempo necesario para la construcción de un producto; el modelo de entrega por etapas es útil para el desarrollo de la herramienta debido a que su uso se recomienda para problemas que pueden ser tratados descomponiéndolos en problemas más pequeños.

Entre sus beneficios tenemos:

- Los resultados se van entregando por etapas.
- Detección de problemas antes y no hasta la única entrega final del proyecto.
- Los resultados pueden ser parciales o el mismo con varios refinamientos.
- Se adelantan en el tiempo los resultados.
- Se obtienen resultados sin estar al 100% terminado.

[3]Tomado de: http://www.osso.org.co/docu/tesis/2004/modelo/desarrollo_mod.pdf

- Proporciona signos tangibles de progreso.

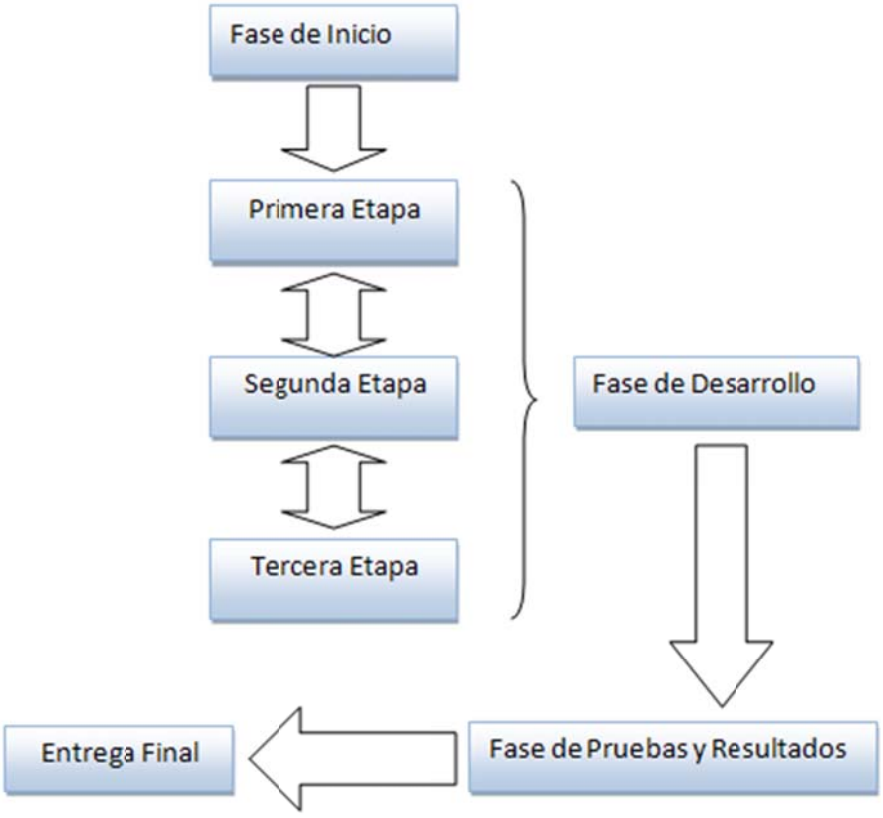
- Requiere una gestión compleja.

- Estimación de tiempo por versión, evitando errores en la estimación del proyecto general.

- Cumplimiento a la fecha por los desarrolladores.

Para la realización de este proyecto se describe de forma generalizada las etapas que debe seguir este en la siguiente figura (Ver Figura 1.)

Figura 1. Metodología Entrega por Etapas



Si bien el ciclo de vida de entrega por etapas se ajusta bien a las condiciones y características del modelo, es posible definir aspectos adicionales que beneficiarían al modelo, tales como:

Realizar planificación de las etapas: identificar un orden que permita la realización temprana de revisión y pruebas, como mecanismos de control de calidad.

Análisis específico de requerimientos: a fin de definir con mayor detalle cuáles serán las funcionalidades y limitaciones del sistema.

Definición del sistema crítico: La identificación, diseño, codificación y prueba de este componente tendrá mayores exigencias de calidad y verificación, para asegurar confiabilidad.

Aplicación de un plan de calidad sobre las etapas: Es la confirmación de otros procesos que se encuentran ligados a la gestión del riesgo del software.

3.2 PLAN DE TRABAJO

3.2.1 Fase de Inicio. Estudio y comprensión de los conceptos básicos relacionados con las proteínas y los aminoácidos que las componen, así como los diferentes tipos de estructuras de las proteínas.

Revisión de textos, libros de modelado molecular y artículos especializados en la predicción de la estructura secundaria de una proteína y uso de las máquinas de soporte vectorial.

Selección de la información bibliográfica más relevante que sirve como soporte al proyecto.

3.2.2 Fase de Desarrollo. Se extrae la información proveniente de la secuencia de aminoácidos (estructura primaria), una vez realizado este proceso se codifica la información extraída de la estructura primaria para inferir la estructura secundaria asociada a cada aminoácido.

Se definen los parámetros para la creación y entrenamiento de las máquinas de soporte vectorial (MSV).

Se define la arquitectura para que la herramienta esté disponible en la web.

3.2.3 Fase de pruebas y resultados. En esta fase se verifica el correcto funcionamiento de la herramienta para predecir estructuras secundarias de proteínas, además de una comparación de los resultados obtenidos con otras metodologías de predicción.

3.2.4 Fase final. Se entrega la herramienta bioinformática para predecir estructuras secundarias de proteínas.

4. MARCO TEÓRICO

4.1 PROTEÍNAS Y AMINOÁCIDOS

La palabra proteína proviene del griego protos, que significa "lo primero o lo más importante".

Son macromoléculas que contienen nitrógeno. Son el componente clave de cualquier organismo vivo y forman parte de cada una de sus células y son para nuestro organismo lo que la madera es para el barco.
Las proteínas están formadas por: carbono, oxígeno, hidrógeno y nitrógeno fundamentalmente, aunque también podemos encontrar, en alguna de ellas, azufre, fósforo, hierro y cobre. Las proteínas se distinguen de los carbohidratos y de las grasas por contener además nitrógeno en su composición, aproximadamente un 16%.

La parte más pequeña en que pueden dividirse son los aminoácidos. Estos aminoácidos son como las letras del abecedario, que con un número determinado se pueden formar infinidad de palabras. Existen 20 aminoácidos y con ellos se forman todas las proteínas. De estos aminoácidos 8 son esenciales (imprescindibles), es decir los tenemos que ingerir con la dieta ya que nuestro organismo no los puede obtener de ninguna otra forma.

4.1.1 Proteínas. Las proteínas son macromoléculas de enorme importancia biológica; prácticamente todas las funciones biológicas desempeñadas en todos los organismos vivos (incluyendo a los virus incluso) son llevadas a cabo por proteínas.

Las proteínas son cadenas de aminoácidos que se pliegan adquiriendo una estructura tridimensional que les permite llevar a cabo miles de funciones. Las

proteínas están codificadas en el material genético de cada organismo, donde se especifica su secuencia de aminoácidos, y luego son sintetizadas por los ribosomas.

4.1.1.1 Función de las Proteínas. Las proteínas cumplen prácticamente todas las funciones posibles en los organismos.

Algunas proteínas catalizan (hacen más veloces) reacciones químicas dentro de los organismos, tal es el caso de las enzimas; otras forman túneles en las membranas de las células y permiten el flujo de iones y moléculas entre el interior de la célula y el exterior, a estas proteínas se les llama canales; también hay proteínas estructurales que forman el andamiaje interno de las células (tubulina y actina) y de la matriz extracelular (colágeno y quitina); otras proteínas son receptores a diversos estímulos (calor, presión, luz) y moléculas, así las rodopsinas son receptores de luz que se hallan en los ojos de la mayor parte de los vertebrados; también algunas proteínas cumplen un rol de mensajeras en los organismos, tal es el caso de algunas hormonas que son de naturaleza proteica; también hay proteínas encargadas del transporte, como la hemoglobina, que transporta oxígeno desde los pulmones a las células.

4.1.1.2 Principales Funciones de las proteínas. Las funciones de las proteínas son las siguientes:

- Las proteínas tienen una función defensiva, ya que crean los anticuerpos y regulan factores contra agentes extraños o infecciones. Toxinas bacterianas, como venenos de serpientes o la del botulismo son proteínas generadas con funciones defensivas.

Las mucinas protegen las mucosas y tienen efecto germicida. El fibrinógeno y la trombina contribuyen a la formación coágulos de sangre para evitar las

hemorragias. Las inmunoglobulinas actúan como anticuerpos ante posibles antígenos.

- Las proteínas tienen otras funciones reguladoras puesto que de ellas están formados los siguientes compuestos: Hemoglobina, proteínas plasmáticas, hormonas, jugos digestivos, enzimas y vitaminas que son causantes de las reacciones químicas que suceden en el organismo. Algunas proteínas como la ciclina sirven para regular la división celular y otras regulan la expresión de ciertos genes.

- Las proteínas cuya función es enzimática son las más especializadas y numerosas. Actúan como biocatalizadores acelerando las reacciones químicas del metabolismo.

- Las *proteínas funcionan como amortiguadores*, manteniendo en diversos medios tanto el pH interno como el equilibrio osmótico. Es la conocida como función homeostática de las proteínas.

- La contracción de los músculos través de la miosina y actina es una función de las proteínas contráctiles que facilitan el movimiento de las células constituyendo las miofibrillas que son responsables de la contracción de los músculos. En la función contráctil de las proteínas también está implicada la dineína que está relacionada con el movimiento de cilios y flagelos.

- La *función de resistencia* o función estructural de las proteínas también es de gran importancia ya que las proteínas forman tejidos de sostén y relleno que confieren elasticidad y resistencia a órganos y tejidos como el colágeno del tejido conjuntivo fibroso, reticulina y elastina del tejido conjuntivo elástico. Con este tipo de proteínas se forma la estructura del organismo. Algunas *proteínas forman estructuras celulares* como las histonas, que forman parte de los cromosomas que

regulan la expresión genética. Algunas glucoproteínas actúan como receptores formando parte de las membranas celulares o facilitan el transporte de sustancias.

- Si fuera necesario, *las proteínas cumplen también una función energética* para el organismo pudiendo aportar hasta 4 Kcal. de energía por gramo. Ejemplos de la función de reserva de las proteínas son la lactoalbúmina de la leche o a ovoalbúmina de la clara de huevo, la hordeína de la cebada y la gliadina del grano de trigo constituyendo estos últimos la reserva de aminoácidos para el desarrollo del embrión.

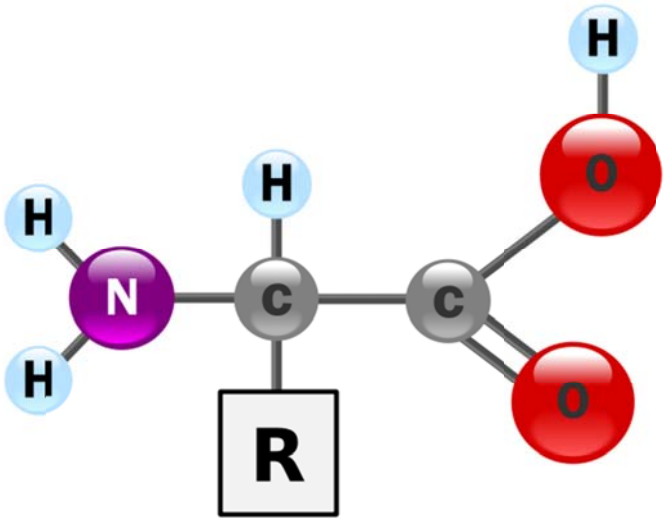
- Las proteínas realizan funciones de transporte. Ejemplos de ello son la hemoglobina y la mioglobina, proteínas transportadoras del oxígeno en la sangre en los organismos vertebrados y en los músculos respectivamente. En los invertebrados, la *función de proteínas* como la hemoglobina que transporta el oxígeno la realizas la hemocianina.

Otros ejemplos de *proteínas cuya función es el transporte* son citocromos que transportan electrones e lipoproteínas que transportan lípidos por la sangre.

4.1.2 Aminoácidos. Los aminoácidos son las unidades químicas o "bloques de construcción" del cuerpo que forman las proteínas, Es una molécula orgánica que es la base estructural de las proteínas y constituida por un grupo amino (-NH2), un grupo carboxilo (-COOH; ácido) y una cadena lateral R.(ver Figura 2.[4])

[4] Tomado de: http://hominidos.blogspot.com/2009/04/hacia-el-origen-de-la-vida.html

Figura 2. Estructura de un Aminoácido



Existen 20 aminoácidos diferentes que se combinan entre ellos de múltiples maneras para formar cada tipo de proteínas. Cada aminoácido es representado por un código de una o tres letras. (Ver Tabla 1.[5]) Los aminoácidos pueden dividirse en 2 tipos: Aminoácidos esenciales que son 9 y que se obtienen de alimentos y aminoácidos no esenciales que son 11 y se producen en nuestro cuerpo.

Tabla 1. Código de los Aminoácidos.

NO	AMINOÁCIDO	CÓDIGO 1 LETRA	CÓDIGO 3 LETRAS
1	Alanina	A	Ala
2	Arginina	R	Arg
3	Asparagina	N	Asn
4	Acido aspártico	D	Asp
5	Cisteína	C	Cys
6	Glutamina	Q	Gln
7	Acido Glutámico	E	Glu

[5]Tomado de: http://www.fao.org/docrep/004/y2775s/y2775s0i.htm

NO	AMINOÁCIDO	CÓDIGO 1 LETRA	CÓDIGO 3 LETRAS
8	Glicina	G	Gly
9	Histidina	H	His
10	Isoleucina	I	Ile
11	Leucina	L	Leu
12	Lisina	K	Lys
13	Metionina	M	Met
14	Fenilalanina	F	Phe
15	Prolina	P	Pro
16	Serina	S	Ser
17	Treonina	T	Thr
18	Triptófano	W	Trp
19	Tirosina	Y	Tyr
20	Valina	V	Val

Los aminoácidos que se obtienen de los alimentos se llaman "Aminoácidos esenciales". Los aminoácidos que puede fabricar nuestro organismo a partir de otras fuentes, se llaman "Aminoácidos no esenciales". El crecimiento, la reparación y el mantenimiento de todas las células dependen de ellos. Después del agua, las proteínas constituyen la mayor parte del peso de nuestro cuerpo.

- **Aminoácidos esenciales.** Se llaman aminoácidos esenciales[6] aquellos que no pueden ser sintetizados en el organismo y para obtenerlos es necesario tomar alimentos ricos en proteínas que los contengan. Nuestro organismo, descompone las proteínas para obtener los aminoácidos esenciales y formar así nuevas proteínas.

- **Aminoácidos no esenciales.** Los aminoácidos no esenciales son aquellos que pueden ser sintetizados en el organismo a partir de otras sustancias.

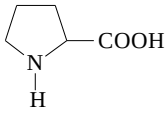
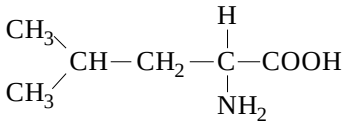
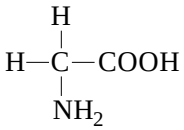
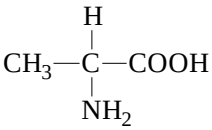
[6]Tomado de: http://proteinas.org.es/aminoacidos

Tabla 2. Aminoácidos Esenciales y no Esenciales

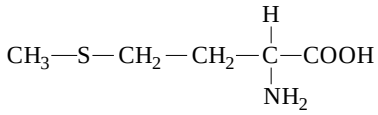
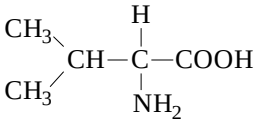
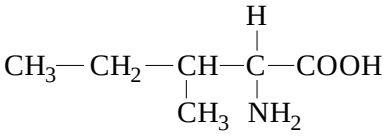
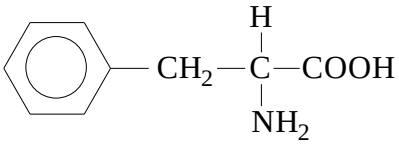
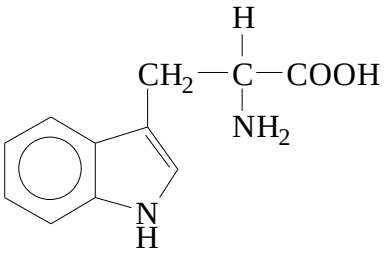
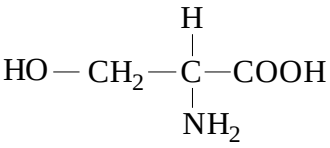
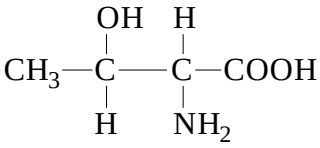
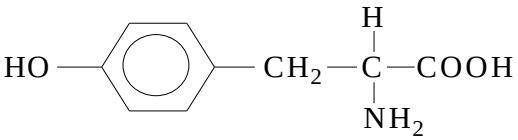
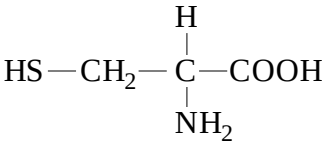
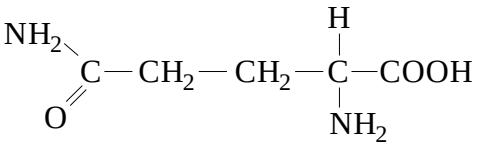
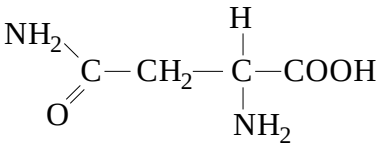
AMINOÁCIDOS ESENCIALES	AMINOÁCIDOS NO ESENCIALES
Histidina	Arginina
Isoleucina	Ácido Aspártico
Leucina	Cisteína
Lisina	Ácido Glutámico
Metionina	Glutamina
Fenilalanina	Glicina
Treonina	Ornitina
Triptófano	Prolina
Valina Alanina	Serina Taurina
	Tirosina

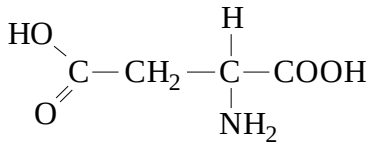
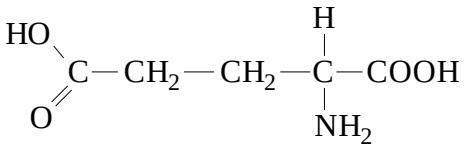
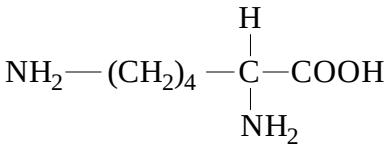
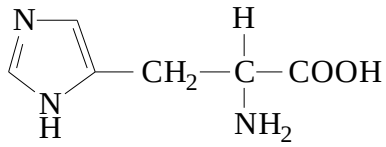
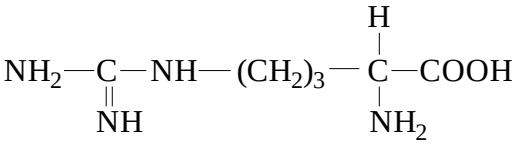
4.1.2.1 Estructura y Clasificación de los Aminoácidos. Dado que los diferentes aminoácidos difieren unos de otros por su cadena lateral[7], podemos clasificarlos según el tipo de cadena lateral que posean.

Tabla 3. Clasificación de los Aminoácidos

HIDROFOBOS: Grupo R no polar (apolares) formado por cadenas hidrocarbonadas	
AMINOÁCIDOS APOLARES ALIFATICOS	
	
PROLINA	LEUCINA
	
GLICINA	ALANINA

[7]Tomado de: http://quimica.laguia2000.com/conceptos-basicos/cadena-lateral-en-aminoacidos

	
METIONINA	VALINA
	
ISOLEUCINA	
AMINOÁCIDOS APOLARES AROMATICOS	
	
FENILALANINA	TRIPTÓFANO
HIDROFILOS: Grupo R polar pero sin carga.	
AMINOÁCIDOS POLARES SIN CARGA	
	
SERINA	TREONINA
	
TIROSINA	CISTEÍNA
	
GLUTAMINA	ASPARAGINA
ACIDOS: Con carga negativa, poseen dos grupos ácidos.	
AMINOÁCIDOS POLARES CON CARGA	

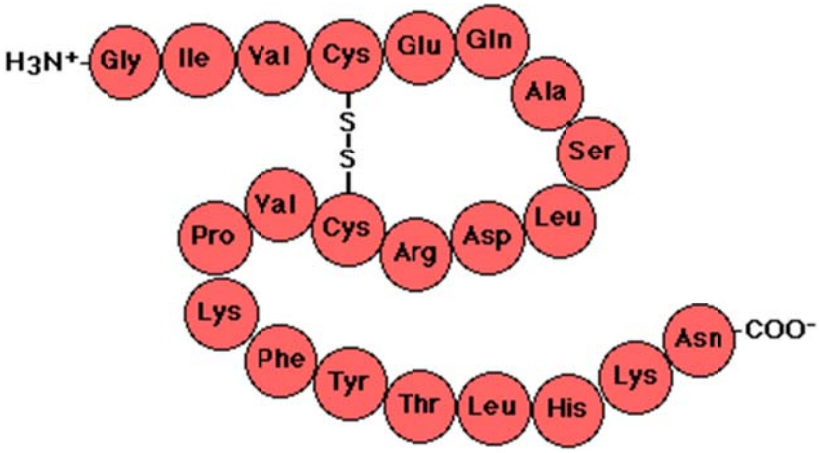
	
ACIDO ASPARTICO	ACIDO GLUTAMICO
BASICOS: Con carga positiva, poseen dos grupos aminos.	
AMINOÁCIDOS POLARES CON CARGA	
	
LISINA	HISTIDINA
	
ARGININA	

4.1.3 Estructura de las Proteínas. Las proteínas poseen una *estructura química* central que consiste en una cadena lineal de aminoácidos plegada de forma que muestra una estructura tridimensional, esto les permite a las proteínas realizar sus funciones. En las proteínas están codificadas en el *material genético* de cada organismo y en él se especifica su secuencia de aminoácidos. Estas secuencias de aminoácidos se sintetizan por los ribosomas para formar las macromoléculas que son las proteínas. La *composición de las proteínas* consta de carbono, hidrógeno, nitrógeno y oxígeno además de otros elementos como azufre, hierro, fósforo y zinc.

4.1.3.1 Niveles de Estructura Proteica. La organización de una proteína viene definida por cuatro niveles estructurales denominados: estructura primaria, estructura secundaria, estructura terciaria y estructura cuaternaria. Cada una de estas estructuras informa de la disposición de la anterior en el espacio.

- **Estructura Primaria.** La estructura primaria viene determinada por la secuencia de aminoácidos en la cadena proteica, es decir, el número de aminoácidos presentes y el orden en que están enlazados (Ver Figura 3[8].). Las posibilidades de estructuración a nivel primario son prácticamente ilimitadas. Como en casi todas las proteínas existen 20 aminoácidos diferentes, el número de estructuras posibles viene dado por las variaciones con repetición de 20 elementos tomados de n en n, siendo n el número de aminoácidos que componen la molécula proteica.

Figura 3. Estructura Primaria



Generalmente, el número de aminoácidos que forman una proteína oscila entre 80 y 300. Los enlaces que participan en la estructura primaria de una proteína son covalentes: son los enlaces peptídicos. El enlace peptídico (Ver Figura 4.[9]) es un enlace amida que se forma entre el grupo carboxilo de una AA con el grupo amino de otro, con eliminación de una molécula de agua. Independientemente de la longitud de la cadena polipeptídica, siempre hay un extremo amino terminal y un extremo carboxilo terminal que permanecen intactos. Por convención, la secuencia de una proteína se lee siempre a partir de su extremo amino (Ver Figura 3.).

[8]Imagen extraída de : www.ehu.es/biomoleculas/proteinas/jpg/primary.gif
[9]Tomado de: http://www.biologia.arizona.edu/biochemistry/problem_sets/large_molecules/01t.html

Figura 4. Enlace Peptídico

Como la estructura primaria es la que determina los niveles superiores de organización, el conocimiento de la secuencia de aminoácidos es del mayor interés para el estudio de la estructura y función de una proteína. Clásicamente, la secuenciación de una proteína se realiza mediante métodos químicos.

- **Estructura Secundaria.** La estructura secundaria es el plegamiento que la cadena polipeptídica adopta gracias a la formación de puentes de hidrógeno entre los átomos que forman el enlace peptídico.

Los puentes de hidrógeno se establecen entre los grupos -CO- y -NH- del enlace peptídico (el primero como aceptor de H, y el segundo como donador de H). De esta forma, la cadena polipeptídica es capaz de adoptar conformaciones de menor energía libre, y por tanto, más estables.

Se pueden distinguir varios tipos de conformaciones que determinan la estructura secundaria de una proteína:

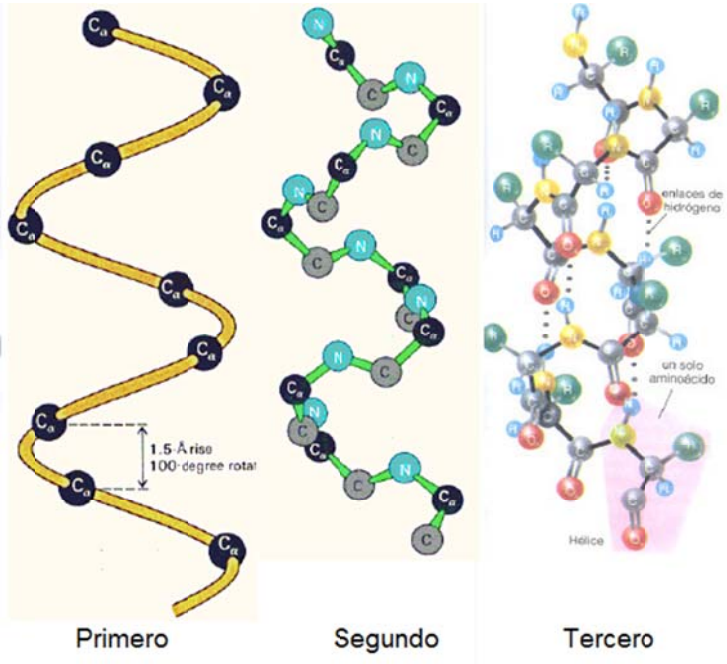
- **Hélices alfa α.** Una hélice alfa es una apretada hélice formada por una cadena polipeptídica. La cadena polipeptídica principal forma la estructura central, y las cadenas laterales se extienden por fuera de la hélice. El grupo carboxilo (CO)

de un aminoácido n se une por puente hidrógeno al grupo amino (NH) de otro aminoácido que está tres residuos más allá (n + 4). De esta manera cada grupo CO y NH de la estructura central (columna vertebral o "backbone") se encuentra unido por puente hidrógeno.

Existen tres modelos de alfa hélice. El primero muestra solo al carbono alfa de cada aminoácido. El segundo muestra todos los átomos que forman la columna vertebral del polipéptido.

El tercero y más completo modelo, muestra todos los puentes hidrógeno que mantienen el alfa-hélice. Las hélices generalmente están formadas por aminoácidos hidrófobos, en razón que son, generalmente, la máxima atracción posible entre dichos aminoácidos. Las hélices se observan, en variada extensión, prácticamente en todas las proteínas. (Ver Figura 5.[10])

Figura 5. Modelos hélices Alfa



[10]Tomado de: http://www.biologia.edu.ar/macromoleculas/structup.htm

- **Láminas β.** Otro tipo de estructura secundaria que pueden adoptar las cadenas polipeptídicas cuando se encuentran muy extendidas es el de lámina b, en este tipo de conformación, conocida también con el nombre de hoja plegada, el esqueleto peptídico se encuentra extendido en "zig-zag" en lugar de plegarse como una hélice.

Este tipo de estructura permite la asociación de dos o más cadenas dispuestas una al lado de la otra. De esta forma logran su estabilidad mediante puentes de hidrógeno entre los grupos amida y carbonilo del enlace peptídico entre cadenas adyacentes. Cada cadena constituye un grupo lineal, es decir todos los ángulos ϕ y ψ son idénticos.

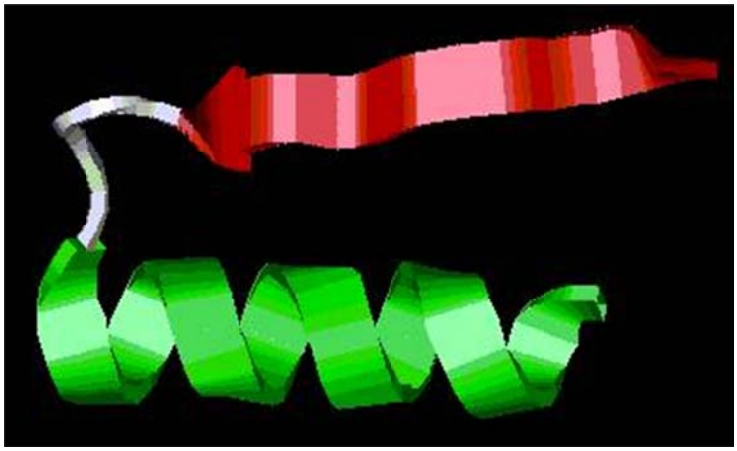
La forma habitual de representar este tipo de estructura secundaria es con una flecha que indica la dirección de la cadena, y el sentido desde el extremo amino hasta el carboxilo terminal. (Ver Figura 6.[11])

Figura 6. Láminas β

[11]Tomado de: http://upload.wikimedia.org/wikipedia/commons/thumb/0/0a/im8n_Choristoneura_fumiferana.png .png/200px-1m8n_Choristoneura_fumiferana.png

- **Giros β o Coil.** Secuencias de la cadena polipeptídica con estructura α o β a menudo están conectadas entre sí por medio de los llamados giros β (Ver Figura 7.[12]). Son secuencias cortas, con una conformación característica que impone un brusco giro de 180^{o} a la cadena principal de un polipéptido.

Figura 7. Giro β o Coil



- **Estructura Terciaria.** Se llama estructura terciaria a la disposición tridimensional de todos los átomos que componen la proteína, concepto equiparable al de conformación absoluta en otras moléculas. La estructura terciaria de una proteína es la responsable directa de sus propiedades biológicas, ya que la disposición espacial de los distintos grupos funcionales determina su interacción con los diversos ligandos.

Para las proteínas que constan de una sola cadena polipeptídica (carecen de estructura cuaternaria), la estructura terciaria es la máxima información estructural que se puede obtener.

La estructura terciaria (Ver Figura 8.[13]) es una disposición precisa y única en el espacio, y surge a medida que se sintetiza la proteína. En otras palabras, la

[12]Tomado de: www.ehu.es/biomoleculas/proteinas/prot42d.htm
[13]Tomado de: http://xquimicx.blogspot.com/2009_05_01_archive.html

estructura terciaria está determinada por la secuencia de Aminoácidos (estructura primaria).

Figura 8. Estructura Terciaria

Se distinguen dos tipos de estructura terciaria:

- Proteínas con estructura terciaria de tipo fibroso en las que una de las dimensiones es mucho mayor que las otras dos. Son ejemplos el colágeno, la queratina del cabello o la fibroína de la seda), En este caso, los elementos de estructura secundaria (hélices a u hojas b) pueden mantener su ordenamiento sin recurrir a grandes modificaciones, tan sólo introduciendo ligeras torsiones longitudinales, como en las hebras de una cuerda.

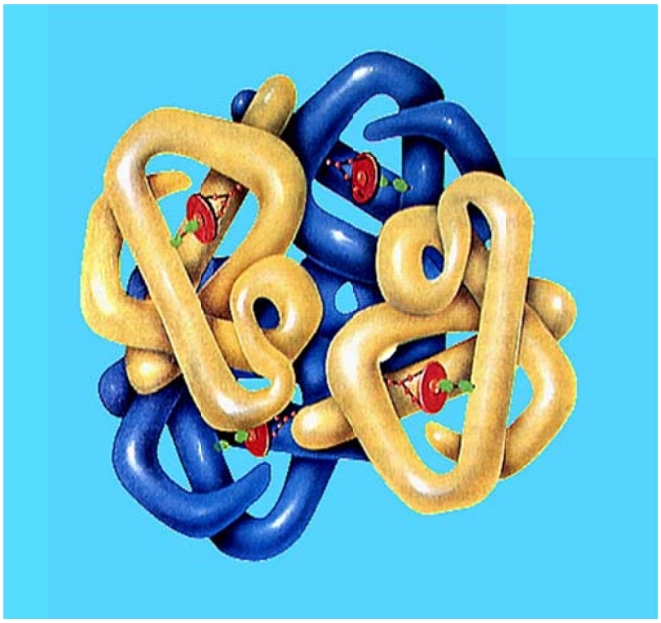
- Proteínas con estructura terciaria de tipo globular, más frecuentes, en las que no existe una dimensión que predomine sobre las demás, y su forma es aproximadamente esférica. En este tipo de estructuras se suceden regiones con estructuras al azar, hélice a hoja b, acodamientos y estructuras súpersecundarias. La figura inferior de la derecha corresponde a la mioglobina.

- **Estructura Cuaternaria.** Cuando una proteína consta de más de una cadena polipeptídica, es decir, cuando se trata de una proteína oligomérica, decimos que tiene estructura cuaternaria.

La estructura cuaternaria debe considerar: el número y la naturaleza de las distintas subunidades o monómeros que integran el oligómero y la forma en que se asocian en el espacio para dar lugar al oligómero.

La estructura cuaternaria (Ver Figura 9.[14]) modula la actividad biológica de la proteína.

Figura 9. Estructura Cuaternaria



4.2 MAQUINAS DE SOPORTE VECTORIAL (MSV)

Aunque existen muchas técnicas alternativas para enfrentar problemas de regresión y clasificación, las máquinas de soporte vectorial han sido desarrolladas

[14]Tomado de : www.ehu.es/biomoleculas/proteinas/prot44.htm

como una herramienta robusta para regresión y clasificación en dominios complejos y ruidosos. Las MSV pueden ser usadas para extraer información relevante a partir de conjunto de datos y construir algoritmos de clasificación o de regresión rápidos para datos masivos.

Desarrollado por Vladimir Vapnik y sus colaboradores, su primera presentación fue en 1992. Aunque sus propiedades ya existían y habían sido usadas en las máquinas de aprendizaje desde 1960.

- Los hiperplanos de margen grande en el espacio de entrada fueron tratados por Duda y venado, Cubra, Vapnik, y otros.

- El uso de Kernels se propuso por Aronszajn, Wahba, Poggio, y otros, pero fue Aizermann en 1964 quien introdujo la interpretación geométrica de los Kernels como los productos internos en un espacio característico.

- El uso de variables de relajación para superar el problema de ruido y la no separabilidad, se introdujo en los años sesenta por Smith y perfeccionado por Bennett y Mangasarian.

Sin embargo, no fue hasta 1992 que todas estas características se reunieron para formar el clasificador del margen fuerte, la máquina de soporte vectorial básica, y hasta 1995 la versión del margen débil se introdujo por Cortés y Vapnik: “Es sorprendente como natural y elegantemente todas las piezas se ajustan entre si y se complementan unas con otras”.

Las publicaciones de Shawe-Taylor y Barteltt dieron el primer límite estadístico riguroso en la generalización de las MSV de margen fuerte. Luego en trabajos Shawe-Taylor y Cristianini dan límites similares para los algoritmos de margen débil.

Después de la introducción de las MSV, un creciente número de investigadores han trabajo en el análisis algorítmico y teórico de estos sistemas, creando en unos pocos años una nueva línea de investigación, fusionando conceptos de disciplinas tan distantes como estadística, análisis funcional, optimización y máquinas de aprendizaje.

Por otra parte, en la última década, una considerable comunidad de teóricos e ingenieros se ha formado alrededor de estos métodos, y se han realizado numerosas aplicaciones prácticas. Aunque la investigación sobre las MSV no ha concluido, ya son muchos los métodos basados en ellas que aparecen en el estado del arte de diversas tareas de aprendizaje de máquina. Su fácil uso, su atractivo teórico, y su notable desempeño han hecho de ellas una buena elección para muchos problemas de aprendizaje computacional. Las aplicaciones exitosas varían desde la categorización de textos y reconocimiento de caracteres escritos hasta la clasificación de datos de expresiones de genes. En muchos aspectos, los últimos años han sido testigos del surgimiento de un nuevo paradigma para el aprendizaje de máquina, comparable a lo ocurrido en los años 80´s cuando la casi simultánea introducción de los algoritmos de árboles de decisión y de redes neuronales revolucionó la práctica en reconocimiento de patrones y minería de datos.

En Colombia, la inserción real en esta tendencia es todavía incipiente, pero ya está tomando un fuerte impulso, aunque todavía existe un desconocimiento general sobre esta nueva concepción de los modelos, por parte de los investigadores y consultores. Las causas posibles de esto, podrían ser falta de fomento e ilustración de los procedimientos básicos, falta de evidencia teórica y práctica de la efectividad operacional de ellas o falta de herramientas conceptuales y computacionales.

4.2.1 Metodología de las MSV. Las MSV, aplicadas al problema de clasificación, mapean los datos a un espacio de características n-dimensional, donde se puede hallar más fácilmente un hiperplano de separación. Este mapeo puede ser llevado a cabo aplicando el kernel, el cual transforma implícitamente el espacio de entrada en un espacio de características de alta dimensión. El hiperplano de separación es calculado maximizando la distancia de los patrones más cercanos, es decir la maximización del margen. Las MSV pueden ser definidas como un sistema para el entrenamiento eficiente de máquinas de aprendizaje lineal en un espacio de características inducido por un kernel, mientras respeta los principios de la teoría de la generalización y explota la teoría de la optimización [N. Cristianini, J. Shawe-Taylor,2000].

4.2.1.1 Clasificación por Vectores de Soporte

- **Caso Linealmente Separable.** Considere el problema de separar el conjunto de vectores de entrenamiento (x_1, y_1),..., (X_i, Y_i), $\in$ R^n que pertenecen a dos clases separadas (Y_i= {1,-1}), (Ver Figura 10.). En este problema la meta es separar los vectores de entrenamiento en dos clases mediante un hiperplano.

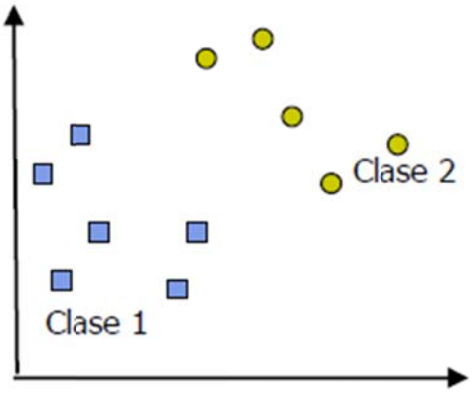
$$(w \,.\, x) + b = 0, w \in R^n y b \in \mathrm{R} \quad (1)$$

Donde *w* y *b* son parámetros que se inducen a partir de los ejemplos disponibles correspondientes a la función de decisión:

$$f(x) = sign(wx + b) \qquad (2)$$

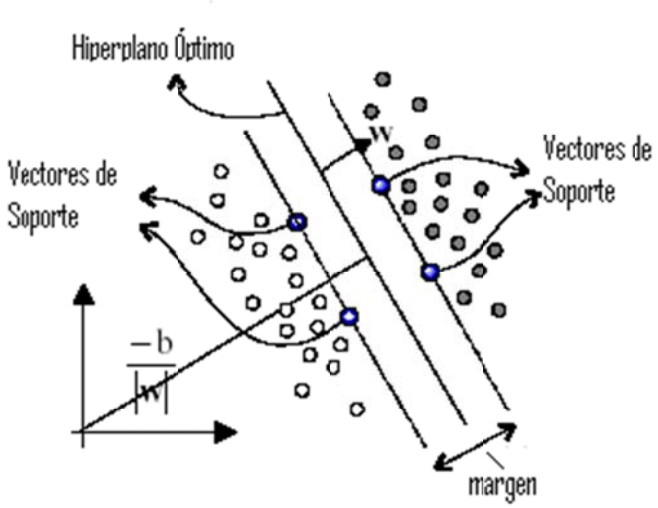
Tal que ella se desempeña bien sobre ejemplos no vistos, es decir que generaliza bien.

Figura 10. Caso linealmente separable



Para el caso del espacio de entrada bidimensional (Ver Figura 11.) hay muchos posibles clasificadores lineales que pueden separar los datos; pero hay sólo uno que maximiza el margen (es decir, maximiza la distancia entre él y el dato más cercado de cada clase). Este clasificador lineal es llamado el hiperplano de separación óptima. Se ha demostrado, además que el hiperplano óptimo, definido como el que tiene el margen máximo de separación entre las dos clases, tiene la capacidad más baja y minimiza la cota sobre el riesgo real.[C. Cortes et al.,1995][V.Ñ. Vapnik,1998].

Figura 11. Hiperplano de separación óptima para el caso bidimensional



El hiperplano $(wx) + b = 0$ satisface las condiciones:

$$(w\ .\ x_i) + b > 0\ \ si y_i = 1\ \ \ y(w\ .\ x_i) + b < 0\ \ si y_i = -1 \quad (3)$$

Combinando las dos expresiones en la ecuación (3) y escalando *w* y *b*, con un factor apropiado, una superficie de decisión equivalente se puede formular como aquella que satisfaga la restricción:

$$y_i[(w.x_i) + b] \geq 1, \quad i = 1,2, \dots, l \qquad (4)$$

Se puede demostrar que el hiperplano que separa óptimamente los datos en dos clases es aquel que minimiza el funcional:

$$\phi(w) = \frac{\|w\|^2}{2} \qquad (5)$$

Por lo tanto, el problema de optimización puede ser reformulado como un problema de optimización no restringida, usando multiplicadores de LaGrange y su solución estaría dada por la identificación de los puntos de silla del funcional de LaGrange como sigue:

$$L(w, b, \propto) = \frac{\|w\|^2}{2} - \sum_{i-1}^{l} \propto_i \{[(w.x_i) + b]y_i - 1\} \quad (6)$$

Donde α_i son los multiplicadores de LaGrange. El Lagrangiano tiene que ser minimizado con respecto a *w* y *b*, es decir:

$$\frac{\partial L}{\partial b} = 0\ \Rightarrow \sum_{i-1}^{l} \propto_i y_i = 0\, y \frac{\partial L}{\partial w} = 0 \Rightarrow w - \sum_{i-1}^{l} y_i \propto_i x_i = 0\ \Rightarrow w = \sum_{i-1}^{l} y_i \propto_i x_i$$

Poniendo la expresión para $w_0$en la ecuación (6) resultará en la siguiente forma dual de la función, que debe ser maximizada con respecto a las restricciones $\propto_i \geq 0$

$$W(\propto) = \sum_{i-1}^{l} \propto_i - \frac{1}{2}\sum_{i-1}^{l}\sum_{j-1}^{l} \propto_i \propto_j y_i y_j (x_i . x_j) \quad (7)$$

Encontrar la solución de la ecuación (7) para problemas el mundo real usualmente requerirá la aplicación de técnicas de optimización de programación cuadrática (QP) y métodos numéricos. Una vez se halla la solución en la forma de un vector $\alpha^0 = (\alpha_1^0, \alpha_2^0, ..., \alpha_l^0)$, el hiperplano de separación óptimo estará dado por:

$$w_0 = \sum_{vectoresdesoporte} y_i \propto_i^o x_i y b_0 = -\frac{1}{2} w_o . [x_r + x_s] \quad (8)$$

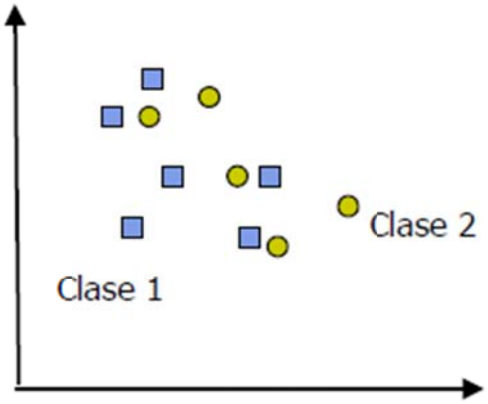
DondeX_r y X_s son cualesquiera vectores de soporte uno de cada clase. Los clasificadores pueden ser, entonces construidos como:

$$f(x) = sign(w_0 . x + b_0) = sign\left(\sum_{vectoresdesoporte} y_i \propto_i^o (x_i x) + b_0\right) \quad (9)$$

Solamente los puntos x_i, que tienen multiplicadores de LaGrange α_i^0 diferentes de cero son llamados Vectores de Soporte(VS). Si los datos son linealmente separables, todos los vectores de soporte estarán sobre el margen y por lo tanto, el número de VS puede ser muy pequeño.
La solución anterior sólo se verifica para datos separables linealmente.

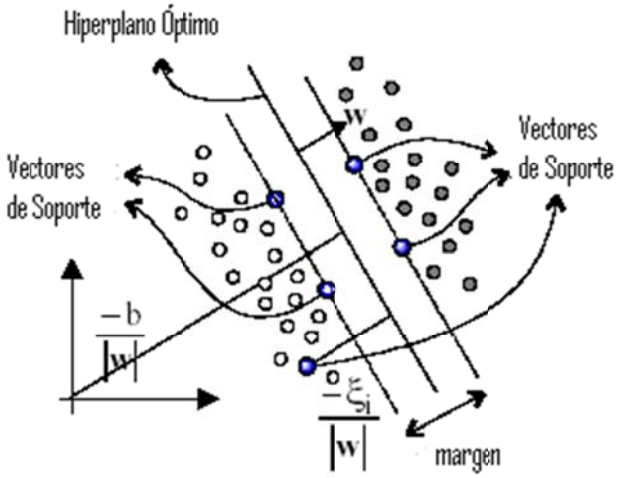
- **Caso No Linealmente Separable.** Si el conjunto de vectores de entrenamiento no es linealmente separable (Ver Figura 12.), violaciones a la clasificación deben ser permitidas en la formulación de la MSV.

Figura 12. Caso no linealmente separable



Para tratar con datos que no son linealmente separables, el análisis previo debe ser ligeramente modificado introduciendo un nuevo conjunto de variables ε_i que mide la cantidad en la cual las restricciones son violadas (Ver Figura 13.).

Figura 13. Hiperplano de separación



Luego el margen es maximizado, asumiendo una penalización proporcional a la cantidad de la violación de la restricción. Formalmente se resuelve el siguiente problema:

$$Minimice\phi(w) = \frac{\|w\|^2}{2} + C\left(\sum \varepsilon_i\right)$$

$$sujetoay_i[(w.x_i + b)] \geq 1 - \varepsilon_i y \varepsilon_i \geq 0 \quad i = 1, \ldots, l \quad (10)$$

Donde C es un parámetro elegido a priori y que define el costo de la violación de la restricción. El primer término en la ecuación (10) proporciona una minimización de la dimensión VC (La dimensión VC es una medida de la complejidad del clasificador y ella es a menudo proporcional al número de parámetros libre en el clasificador$f_\propto$. especialmente cuando $\frac{l}{h}$ es pequeño,) de la máquina de aprendizaje, minimizando por lo tanto, el segundo término en la cota de la siguiente ecuación:

$$R(\propto) \leq R_{emp}(\propto) + \sqrt{\frac{h\left(log\frac{2l}{h}+1\right)-log\left(\frac{n}{4}\right)}{l}} \quad (11)$$

De otra parte, la minimización en el segundo término de la ecuación (10) controla el riesgo empírico, el cual es el primer término en la ecuación (11). Esta aproximación, por lo tanto, constituye una implementación práctica de la minimización del Riesgo Estructural sobre el conjunto de funciones dado. Con el fin de resolver este problema, el Lagrangiano se construye como sigue:

$$L(w, b, \propto) = \frac{\|w\|^2}{2} + C\left(\sum_{i-1}^{l} \varepsilon_i\right) - \sum_{i-1}^{l} \alpha_i\{[(w.x) + b]y_i - 1 + \varepsilon_i\} - \sum_{i-1}^{l} y_i \varepsilon_i \ (12)$$

Donde α_i y γ_i están asociados con las restricciones en la ecuación (10) y los valores de α_i tienen que ser acotados como $0 \leq \alpha_i \leq C$. De nuevo, la solución de este problema se determina por los puntos de silla de este Lagrangiano de forma similar para el caso de datos separables.

En el caso donde una frontera lineal sea definitivamente inapropiada(o cuando la superficie de decisión es no lineal), la MSV puede mapear el vector de entrada x, en un espacio de características n-dimensional z, eligiendo un mapeo no lineal a priori. Entonces la MSV construye el hiperplano de separación óptimo en este espacio n-dimensional más alto.

En este caso, los problemas de optimización quedan de la siguiente manera:

$$W(\propto) = \sum_{i-1}^{l} \propto_i - \frac{1}{2} \sum_{i-1}^{l} \sum_{j-1}^{l} \propto_i \propto_j y_i y_j K(x_i . x_j) \quad (13)$$

Donde K(x,y) es la función kernel que realiza el mapeo no lineal en el espacio de características, y las restricciones permanecen sin cambio.

La solución de la ecuación anterior determina los multiplicadores de LaGrange, y un clasificador que implementa en hiperplano de separación óptima en el espacio de característica está dado por,

$$f(x) = sign\left(\sum_{vectoresdesoporte} y_i \propto_i^o K(x_i x) + b_0 \right) \quad (14)$$

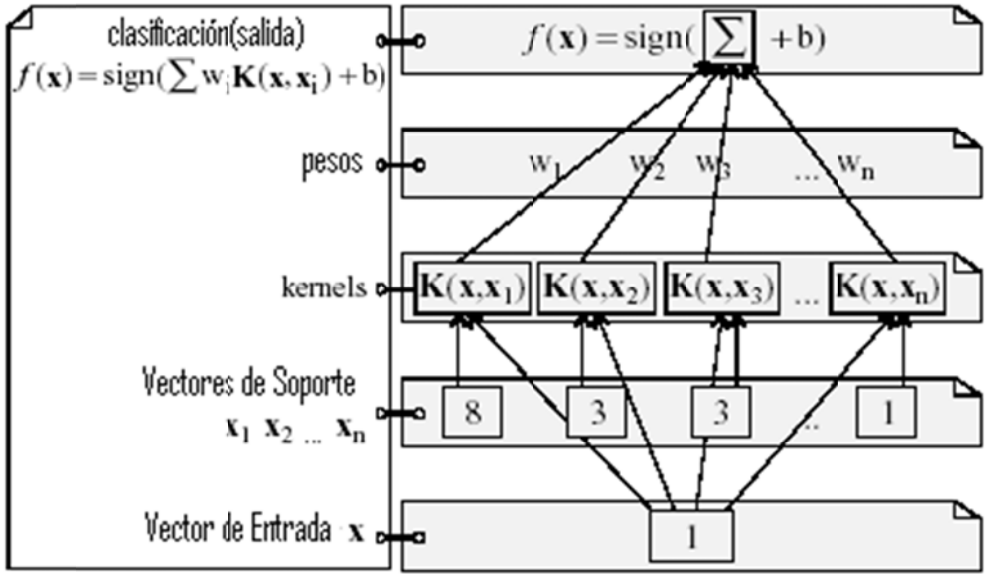
Consecuentemente, todo lo que se ha derivado para el caso lineal es también aplicable para un caso no lineal usando un kernel conveniente K en vez del producto punto. Además, usando diferentes funciones kernel, el algoritmo de VS puede construir una variedad de máquinas de aprendizaje (Ver Figura 14.), algunas de las cuales parecen ser similares a arquitecturas clásicas. Funciones de base radial, funciones polinomiales y ciertas funciones sigmoideas son entre otras que proporcionan Kernels aceptables y los correspondientes mapeos son descritos como sigue:

- El kernel simple polinomial: $K(x, x_i) = \left((x \cdot x_i) + 1\right)^d$, donde el grado del polinomio d, es definido por el usuario.

- Kernel de Funciones de Base Radial: $K(x, x_i) = e^{-\gamma |x - x_i|^2}$, donde γ es definido por el usuario.

- Kernel de redes Neuronales: $K(x, x_i) = \tanh\left(b(x \cdot x_i) - c\right)$, donde b y c son definidos por el usuario.

Figura 14. Arquitectura de máquinas de soporte vectorial



El otro caso surge cuando los datos están en múltiples clases. Con el fin de obtener una clasificación de k-clases, se construye un conjunto de clasificadores binarios $f_1, f_2, \ldots, f_k$, cada uno entrenado para separar una clase del resto, y estos son combinados para llevar a cabo la multiclasificación (en un esquema de votación) de acuerdo con la salida máxima, antes de aplicar la función signo.(Scholkopf, 1997)

4.2.1.2 Arquitecturas de descomposición estándares. En el esquema de descomposición estándar se construyen m máquinas biclasificadoras en paralelo, que son entrenadas sobre modificaciones del conjunto de aprendizaje, creándose una matriz de descomposición. Los elementos de unas clases son asignados a salidas positivas, los de otras a salidas negativas, y si es el caso, los restantes no son tenidos en consideración en aquel clasificador en particular.

- **Uno contra el resto.** Conocido como 1-v-r (one-versus-rest), este esquema se basa en la idea de que si existe un grupo de n datos de entrenamiento donde existen l clases (l >2), se pueden tener un grupo de m clasificadores binarios (donde m =l), cada uno entrenado para separar una clase del resto de clases existentes (l− 1). La descomposición se realiza de la siguiente manera: existe un grupo de datos (n_j) que pertenecen a la j-ésima clase (j ∈{1,…, l}), a los cuales se les dará una etiqueta positiva ($t_j = +1$) y al resto de datos$(n_r = n - n_j)$ se les dará una etiqueta negativa$(t_r = -1)$para el entrenamiento de la i-ésima SVM. Así se crea una matriz de descomposición (D1−v−r) de m filas y l columnas:

$$D_{i,j} = \begin{cases} +1 & \text{si } n_h \in n_j \\ -1 & \text{si } n_h \in n_r \end{cases}$$

Por ejemplo, una máquina de clasificación multiclase (1-v-r) con l = 5, se obtiene m = 5. Entonces la correspondiente matriz de descomposición es la siguiente:

$$D_{1-v-r} = \begin{pmatrix} +1 & -1 & -1 & -1 & -1 \\ -1 & +1 & -1 & -1 & -1 \\ -1 & -1 & +1 & -1 & -1 \\ -1 & -1 & -1 & +1 & -1 \\ -1 & -1 & -1 & -1 & +1 \end{pmatrix}$$

En esta arquitectura propuesta por [V. Vapnik, C. Cortes ,1995] el tiempo de entrenamiento es proporcional al número de clases, y debido a que el entrenamiento de cada biclasificador es con el conjunto de datos de entrenamiento completo su costo computacional es alto.

- **Uno contra uno.** Conocido como 1-v-1 (one-versus-one), se realiza implementando $m = \frac{l(l-1)}{2}$ clasificadores binarios. Luego, el entrenamiento de la i-ésima SVM se realiza con solo 2 de las l (l >2) clases existentes en el grupo de n datos de entrenamiento, otorgándole etiqueta positiva ($t_j = +1$) a los datos (n_j) que pertenecen al subgrupo de datos de la clase j(j ∈ {1, . . . , l}), y etiqueta negativa ($t_p = -1$)a los datos (n_p) que pertenecen al subgrupo de datos de la clase p (p ∈ {1, . . . , l} y p ≠j). Los demás datos ($n_r = n - n_j - n_p$) no se utilizan en el entrenamiento de la i-ésima SVM por lo tanto son etiquetados con cero ($t_r = 0$), creándose la matriz de descomposición (D1−v−1).

$$D_{i,j} = \begin{cases} +1 & \text{si } n_h \in n_j \\ -1 & \text{si } n_h \in n_p \\ 0 & \text{si } n_h \in n_r \end{cases}$$

Por ejemplo, para una máquina de clasificación multiclase (1-v-1) con l = 5, se obtiene m = 10. Entonces la correspondiente matriz de descomposición es la siguiente:

$$D_{1-v-1} = \begin{pmatrix} +1 & -1 & 0 & 0 & 0 \\ +1 & 0 & -1 & 0 & 0 \\ +1 & 0 & 0 & -1 & 0 \\ +1 & 0 & 0 & 0 & -1 \\ 0 & +1 & -1 & 0 & 0 \\ 0 & +1 & 0 & -1 & 0 \\ 0 & +1 & 0 & 0 & -1 \\ 0 & 0 & +1 & -1 & 0 \\ 0 & 0 & +1 & 0 & -1 \\ 0 & 0 & 0 & +1 & -1 \end{pmatrix}$$

- **Arquitectura de descomposición ECOC.** La técnica ECOC (Error Correcting Output Codes) [T. G. Dietterich ,G. Bakiri, 1995],utiliza la codificación estándar para obtener robustez contra fallos en las máquinas biclasificadoras. Se denomina codificación estándar a cada una de las posibles particiones de todo el conjunto de clases $y_i \in \{1,\ldots, l\}$, en problemas de biclasificación, que asignan etiquetas positivas $t_p = +1$a los patrones de entrenamiento $n_j$de un cierto subconjunto de clases y_i, y etiquetas negativas $t_p = -1$a los patrones de entrenamiento $n_r$representantes del resto de clasesy_r. La de descomposición generada por:

$$D_{i,j} = \begin{cases} +1 & \text{si } n_h \in n_j \\ -1 & \text{si } n_h \in n_r \end{cases}$$

Debe ser tan diferente como sea posible en términos de la distancia Hamming para añadir redundancia, en este caso $m = 2^{l-1} - 1$. Por ejemplo, para una máquina de clasificación multiclase (ECOC) con l = 4, se obtiene m = 7. Entonces la correspondiente matriz de descomposición es la siguiente:

$$D_{ECOC} = \begin{pmatrix} +1 & -1 & -1 & -1 \\ +1 & -1 & -1 & +1 \\ +1 & -1 & +1 & -1 \\ +1 & -1 & +1 & +1 \\ +1 & +1 & -1 & -1 \\ +1 & +1 & -1 & +1 \\ +1 & +1 & +1 & -1 \end{pmatrix}$$

4.2.1.3 Métodos de reconstrucción. Cada máquina biclasificadora entrenada emite una respuesta en forma numérica $z^i = g_i(\vec{x})$a una entrada $(\vec{x})$. La información más importante en esta respuesta, en principio, se encuentra en el signo $s_i = f_i(\vec{x}) = sign\ (\ g_i(\vec{x}))$que adopta la función de decisión. En la determinación de la respuesta final facilitada por el método de reconstrucción de la máquina de aprendizaje multiclase han de ser tomados en consideración los siguientes elementos:

- Las predicciones numéricas parciales de los nodos de dicotomía,$z^i = g_i(\vec{x})$.

- El signo de las predicciones numéricas, $s_i = f_i(\vec{x}) = sign\ (\ g_i(\vec{x}))$

- Un elemento interprete de las predicciones numéricas y binarias,$\theta(z^i, s^i)$con el fin de asignar o no, una o varias clases como posible respuesta de clasificación a una entrada $(\vec{x})$.

- Un elemento $\varphi(\theta(z^1, s^1), \dots, \theta(z^m, s^m))$de combinación de las predicciones, que tenga o pueda tener en consideración las predicciones numéricas, sus signos y/o la clase o clases asignadas.

- **Esquemas de votación.** Son la forma de reconstrucción más habitual. Se tiene en consideración solo el signo de las predicciones de todas las máquinas biclasificadoras. Estos signos son interpretados en función de las clases implicadas en las máquinas biclasificadoras utilizado en el esquema de descomposición.

- i-ésimo 1-v-r máquina biclasificadora

$$\Theta\left(s^i\right) = \begin{cases} y_i & \text{si} \quad s^i = +1 \\ \emptyset & \text{si} \quad s^i = -1 \end{cases}$$

- i-ésimo 1-v-1 máquina biclasificadora

$$\Theta\left(s^i\right) = \begin{cases} y_j & \text{si} \quad s^i = +1 \\ y_p & \text{si} \quad s^i = -1 \end{cases}$$

- i-ésimo ECOC máquina biclasificadora

$$\Theta\left(s^i\right) = \begin{cases} Y_j & \text{si} \quad s^i = +1 \\ Y_r & \text{si} \quad s^i = -1 \end{cases}$$

Tras la interpretación de las predicciones, el elemento de combinación φrealiza un recuento del número de clases votadas, acción de la que toma el nombre de esquema de reconstrucción, que posee diferentes variantes. Se define a continuación algunas de estas posibilidades para las arquitecturas de descomposición:

- Votación por unanimidad: se determina como respuesta aquella única clase que haya obtenido todos los votos posibles en las predicciones.

- Votación por mayoría absoluta: se determina como respuesta final aquella única clase que haya obtenido más de a mitad de los votos posibles.

- Votación por mayoría simple: se determina como respuesta final aquella única clase que haya obtenido más votos que el resto de clases.

4.3 ARQUITECTURA CLIENTE-SERVIDOR

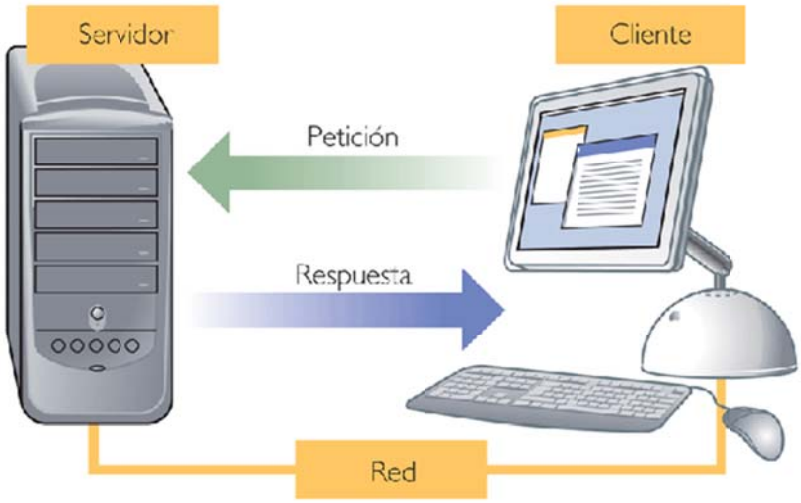
Toda la estructura matemática descrita en los numerales anteriores se puede implementar a partir de una arquitectura cliente-servidor la cual permita crear una herramienta bioinformática (En este caso predecir estructuras secundarias de proteínas) que esté disponible en un entorno web que sea útil a los usuarios.

En el mundo de TCP/IP las comunicaciones entre computadoras se rigen básicamente por lo que se llama modelo Cliente-Servidor, éste es un modelo que intenta proveer usabilidad, flexibilidad, interoperabilidad y escalabilidad en las comunicaciones. El término Cliente/Servidor fue usado por primera vez en 1980 para referirse a PC's en red. Este modelo Cliente/Servidor empezó a ser aceptado a finales de los 80's.

Su funcionamiento es sencillo: se tiene una máquina cliente, que requiere un servicio de una máquina servidor, y éste realiza la función para la que está programado (no tienen que tratarse de máquinas diferentes; es decir, una computadora por sí sola puede ser Cliente y servidor dependiendo del software de configuración).

4.3.1 Modelo Cliente-Servidor. Desde el punto de vista funcional, se puede definir la computación Cliente/Servidor como una arquitectura distribuida que permite a los usuarios finales obtener acceso a la información en forma transparente aún en entornos multiplataforma. En el modelo cliente servidor, el cliente envía un mensaje solicitando un determinado servicio a un servidor (hace una petición), y este envía uno o varios mensajes con la respuesta (provee el servicio) (Ver Figura 15.[15]). En un sistema distribuido cada máquina puede cumplir el rol de servidor para algunas tareas y el rol de cliente para otras.

Figura 15. Esquema Cliente-Servidor



La idea es tratar a una computadora como un instrumento, que por sí sola pueda realizar muchas tareas, pero con la consideración de que realice aquellas que son más adecuadas a sus características.

Si esto se aplica tanto a clientes como servidores se entiende que la forma más estándar de aplicación y uso de sistemas Cliente/Servidor es mediante la explotación de las PC's a través de interfaces gráficas de usuario; mientras que la administración de datos y su seguridad e integridad se deja a cargo de

[15]Tomado de: http://cervantes1bachdyg.wikispaces.com/Arquitectura+cliente-servidor

computadoras centrales tipo mainframe. Usualmente la mayoría del trabajo pesado se hace en el proceso llamado servidor y el o los procesos cliente sólo se ocupan de la interacción con el usuario (aunque esto puede variar). En otras palabras la arquitectura Cliente/Servidor es una extensión de programación modular en la que la base fundamental es separar una gran pieza de software en módulos con el fin de hacer más fácil el desarrollo y mejorar su mantenimiento Esta arquitectura permite distribuir físicamente los procesos y los datos en forma más eficiente lo que en computación distribuida afecta directamente el tráfico de la red, reduciéndolo grandemente.

4.3.2 Cliente. El cliente es el proceso que permite al usuario formular los requerimientos y pasarlos al servidor, se le conoce con el término *front-end*. El Cliente normalmente maneja todas las funciones relacionadas con la manipulación y despliegue de datos, por lo que están desarrollados sobre plataformas que permiten construir interfaces gráficas de usuario (GUI), además de acceder a los servicios distribuidos en cualquier parte de una red.

Las funciones que lleva a cabo el proceso cliente se resumen en los siguientes puntos:

- Administrar la interfaz de usuario.
- Interactuar con el usuario.
- Procesar la lógica de la aplicación y hacer validaciones locales.
- Generar requerimientos de bases de datos.
- Recibir resultados del servidor.
-

- Formatear resultados.

4.3.3 Servidor. Es el proceso encargado de atender a múltiples clientes que hacen peticiones de algún recurso administrado por él. Al proceso servidor se le conoce con el término back-end.

El servidor normalmente maneja todas las funciones relacionadas con la mayoría de las reglas del negocio y los recursos de datos. Las funciones que lleva a cabo el proceso servidor se resumen en los siguientes puntos:

- Aceptar los requerimientos de bases de datos que hacen los clientes.

- Procesar requerimientos de bases de datos.

- Formatear datos para trasmitirlos a los clientes.

- Procesar la lógica de la aplicación y realizar validaciones a nivel de bases de datos.

4.3.4 Características de la arquitectura Cliente-Servidor. Las características básicas de una arquitectura Cliente/Servidor son:

- Combinación de un cliente que interactúa con el usuario, y un servidor que interactúa con los recursos compartidos. El proceso del cliente proporciona la interfaz entre el usuario y el resto del sistema. El proceso del servidor actúa como un motor de software que maneja recursos compartidos tales como bases de datos, impresoras, módems, etc.

- Las tareas del cliente y del servidor tienen diferentes requerimientos en cuanto a recursos de cómputo como velocidad del procesador, memoria, velocidad y capacidades del disco.

- Se establece una relación entre procesos distintos, los cuales pueden ser ejecutados en la misma máquina o en máquinas diferentes distribuidas a lo largo de la red.

- Existe una clara distinción de funciones basada en el concepto de "servicio", que se establece entre clientes y servidores.

- La relación establecida puede ser de muchos a uno, en la que un servidor puede dar servicio a muchos clientes, regulando su acceso a recursos compartidos.

- Los clientes corresponden a procesos activos en cuanto a que son éstos los que hacen peticiones de servicios a los servidores. Estos últimos tienen un carácter pasivo ya que esperan las peticiones de los clientes.

- No existe otra relación entre clientes y servidores que no sea la que se establece a través del intercambio de mensajes entre ambos. El mensaje es el mecanismo para la petición y entrega de solicitudes de servicio.

- El ambiente es heterogéneo. La plataforma de hardware y el sistema operativo del cliente y del servidor no son siempre la misma. Precisamente una de las principales ventajas de esta arquitectura es la posibilidad de conectar clientes y servidores independientemente de sus plataformas.

- El concepto de escalabilidad tanto horizontal como vertical es aplicable a cualquier sistema Cliente/Servidor. La escalabilidad horizontal permite agregar

más estaciones de trabajo activas sin afectar significativamente el rendimiento. La escalabilidad vertical permite mejorar las características del servidor o agregar múltiples servidores.

4.3.5 Ventajas

- Centralización del control: los accesos, recursos y la integridad de los datos son controlados por el servidor de forma que un programa cliente defectuoso o no autorizado no pueda dañar el sistema. Esta centralización también facilita la tarea de poner al día datos u otros recursos.

- Escalabilidad: se puede aumentar la capacidad de clientes y servidores por separado. Cualquier elemento puede ser aumentado (o mejorado) en cualquier momento, o se pueden añadir nuevos nodos a la red (clientes y/o servidores).

- Fácil mantenimiento: al estar distribuidas las funciones y responsabilidades entre varios ordenadores independientes, es posible reemplazar, reparar, actualizar, o incluso trasladar un servidor, mientras que sus clientes no se verán afectados por ese cambio (o se afectarán mínimamente). Esta independencia de los cambios también se conoce como encapsulación.

- Existen tecnologías, suficientemente desarrolladas, diseñadas para el paradigma de Cliente-Servidor que aseguran la seguridad en las transacciones, la amigabilidad del interfaz, y la facilidad de empleo.

5. FASE DE DESARROLLO

5.1 PRIMERA ETAPA

Para extraer la información necesaria de la secuencia de aminoácidos (Estructura Primaria) y su posterior codificación se siguieron los pasos que se exponen a continuación.

5.1.1 Adquisición de datos. Para predecir las estructuras secundarias, en primer lugar se debe obtener los datos de las proteínas con los cuales se entrenaran las máquinas de soporte para la posterior predicción de la estructura secundaria.

Para la realización de este trabajo se utilizaron dos conjuntos de secuencias de proteínas, denominados CB513 y RS126.

Se utilizó la base de datos CB513 [Cuff and Barton] que consta de 513 secuencias de proteínas que presenta la estructura primara con la correspondiente estructura secundaria real, de las cuales se tomaron solamente las secuencias de proteínas que estaban conformadas por la combinación de los 20 aminoácidos mencionados anteriormente, dando como resultado un total de 478 secuencias para realizar el entrenamiento.

Siendo este entrenamiento la manera como se le enseña a la máquina de soporte las características presentes a lo largo de los segmentos de las proteínas para formar los diferentes motivos estructurales.

Para verificar el entrenamiento de las MSV elaboradas se utilizó la base de datos RS126 [Rost and Sander] la cual consta de 126 secuencias de proteínas.

5.1.2 Codificación de la secuencia primaria. A partir de la estructura primaria, convertimos la cadena de aminoácidos en información numérica, para predecir la estructura secundaria, existen diversas maneras de generar información numérica a partir de la secuencia de aminoácidos, para este trabajo se utilizó el enfoque planteado por [Delgado et al., 2010].

5.1.2.1 Extracción de N-Gramas. La posición de cada aminoácido dentro de la secuencia juega un papel importante para la predicción de la estructura secundaria, para esto se empleó la metodología denominada N-grama.

El N-grama consiste en un segmento de aminoácidos donde el aminoácido en el centro es el que se desea codificar, para este trabajo se determinó la longitud *n* de cada N-grama, *n = 19*.

Para generar datos a partir de la secuencia de aminoácidos, se define a: P = $\{A_1, A_2, \dots, A_n\}$ como la estructura primaria, la cual está compuesta por diferentes combinaciones de los 20 aminoácidos ya mencionados y a *m* como el número total de aminoácidos que componen la secuencia (longitud de la secuencia).

La forma para generar los N-gramas a partir de la secuencia se puede ver en el algoritmo 1.

Algoritmo 1 Generar Los N-gramas a partir de la secuencia.

Entrada: *P.*

- Sea $i = 1,2, \dots, m$
- Sea *n* = 19 Longitud definida para cada N-grama.
- Sea $ini = i - \left[\frac{n}{2}\right]$ El punto de inicio en P del $N_i grama$.
- Sea $fin = \left(i + n - \left[\frac{n}{2}\right]\right) - 1$ El punto final en P del $N_i grama$.

El término i hace referencia a la posición de cada aminoácido dentro de la secuencia, los N-gramas se generan de izquierda a derecha a lo largo de la secuencia.

Las posiciones fuera del rango para los N-gramas que se encuentran al inicio y al final de la secuencia, se reemplazan por un símbolo que permite su posterior codificación. (Ver figura 16.).

Figura 16. Representación de la información contenida en la estructura primaria y los N-gramas generados a partir de ella.

$$P = RTDCYGNVNRIDTTGASCKTAKPEG$$
$$N_1 grama = * * * * * * * * * RTDCYGNVNR$$
$$N_{15} grama = GNVNRIDTTGASCKTAKPE$$
$$N_{22} grama = DTTGASCKTAKPEG * * * * *$$

5.1.2.2 Vector de Clasificación. Cada N-grama generado debe ser convertido en un vector de características para poder plantear un algoritmo de clasificación.

Para este trabajo se utilizaron las metodologías planteadas por [Ruant et al., 2005] y por [Ganapathiraju et al., 2004] [Yang and Wang, 2003].Los cuales plantean la codificación de los N-gramas con base en el vector composición de momento (VCM) y las propiedades del grupo al cual pertenece cada aminoácido.

El procedimiento llevado a cabo para la codificación de los N-gramas se basó en el proceso planteado por [Delgado, Fuentes y Torres, 2010].

Vector Composición de Momento Modificado (VCMM)

Algoritmo 2

Entrada: N-grama $N_g = \{ng_1, ng_2, \dots, ng_n\}$

- Sea $A = \{A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y, *\}$
  Los diferentes símbolos que pueden estar en un N-grama.
- Sea n la longitud de cada N-grama.
- Sea A_i el $i - esimo$ AA, cuando los AA se ordenan como se muestra en A.
- Para un $w > 0$ donde $w \in Z$, se define $(x_1^w, x_2^w, \dots, x_{20}^w)$ como el VCM de orden w

  $x_i^w = \frac{\sum_{j=1}^{w_i} n_{i,j}^w}{\prod_{d=0}^{w_i}(n-d)}$ Para $i = 1,2, \dots, 20$
- Donde $n_{i,j}$ es la $j - esima$ posición del $i - esimo$ AA en N_g
- w_i es el número total de veces que aparece el $i - esimo$ AA en N_g

- **R-Grupo.** A partir de las propiedades químicas de los aminoácidos podemos extraer información de interés para la codificación de cada N-grama, tomando como referencia el grupo biológico al cual pertenece cada aminoácido se realizará la codificación llamada *R-grupo,* (Ver Tabla 4.)

Tabla 4. Clasificación de los aminoácidos de acuerdo al grupo biológico.

R-grupos	Codificación	Aminoácidos
No polares, Alifáticos (c1)	[1,0,0,0,0]	A,V L I,M
Aromáticos (c2)	[0,1,0,0,0]	F,Y,W
Polares, No cargados (c3)	[0,0,1,0,0]	G,S,P,T,C,N,Q
Cargados Positivos (c4)	[0,0,0,1,0]	K,H,R
Cargados Negativos (c5)	[0,0,0,0,1]	D,E

- **Probabilidades.** Según plantean [Nelson and Cox., 2000] existe una probabilidad de que cada aminoácido pueda adoptar algún nivel estructural (H, E y -) dependiendo del grupo biológico al cual pertenezca.

Para este trabajo se calcularon las probabilidades de ocurrencia a partir de los datos proporcionados por la base de datos CB513 y teniendo en cuenta el grupo biológico al cual pertenece cada aminoácido, estas probabilidades se muestran en la siguiente tabla. (Ver Tabla 5.)

Para la predicción se calculan las probabilidades de ocurrencia correspondientes a cada motivo estructural para el aminoácido del centro del N-Grama y se escoge la mayor de ellas para crear el vector de clasificación.

Tabla 5. Probabilidades calculadas a partir de la base de datos CB513.

	Hélices α (H)	Lamina β (E)	Coil (C)
R-grupo c1	0,35428252	0,35516186	0,22595343
R-grupo c2	0,09623431	0,10818325	0,08765248
R-grupo c3	0,28052834	0,33835385	0,43272986
R-grupo c4	0,13574125	0,11998938	0,11917217
R-grupo c5	0,13321358	0,07831165	0,13449207

- **Vector de Clasificación.** Usando el producto de Kronecker [D.Zwillinger and K.H. Rosen., 1996] podemos obtener la codificación de cada N-grama, ver ecuación 15.

$$Vcs = P_{aai}^{\tau}, c_j \times c_j \otimes V_{cmm} \qquad (15)$$

En donde P_{aai}^{τ}, c_j es la probabilidad que el aminoácido del centro del N-grama adopte determinado nivel estructural a partir del c_j al cual pertenece, c_j es el R-grupo (Codificación de los aminoácidos en base a su información biológica) al cual pertenece el aminoácido y V_{cmm} es el vector composición de momento modificado.

El símbolo $\times$ indica un producto vectorial y el símbolo $\otimes$ indica el producto de Kronecker.

Una vez codificados todos los N-gramas de la secuencia primaria (Estructura Primaria) se obtiene el vector de clasificación (Vcs), el cual se usara para realizar la predicción de la estructura secundaria.

5.2 SEGUNDA ETAPA

En esta etapa se crean y entrenan las máquinas de soporte vectorial para su posterior uso en la predicción.

5.2.1 Multiclasificación. Un problema de multi-clasificación se puede tratar a partir de varios problemas de clasificación binaria, en donde cada uno de estos se puede resolver de manera aislada.

Sea Vcs el vector de clasificación y $\Gamma = \{H, E, C\}$ el conjunto finito de *k* clases en las que se puede clasificar el Vcs, para este trabajo *k = 3,* los problemas binarios (*k = 2*) fueron etiquetaron como {1, -1}, lo que se busca es que el Vcs pase a través de los clasificadores binarios (MSV) y así poder encontrar la clase $k \in \Gamma$ para la secuencia primaria sobre la cual se realizo la respectiva codificación.

Los diferentes clasificadores binarios se pueden encontrar a partir del enfoque de emparejamiento total [Allwein et al., 2001] de las k clases, para este trabajo se tienen $\binom{k}{2}$ clasificadores: $(f_1 = H|E, f_2 = H|C, f_3 = E|C)$.

5.2.2 Predicción

5.2.2.1 Entrenamiento. Antes de poder realizar la predicción se deben entrenar los diferentes clasificadores binarios, para este trabajo se entrenaron los clasificadores basados en el proceso de predicción expuesto(Ver figura 18.) y siguiendo la metodología planteada por [Delgado, Fuentes y Torres, 2010] para cada clasificador (MSV).

Para el entrenamiento se cuenta con información estadística exacta extraída de la base de datos CB513 ya que se tiene la secuencia primaria y su correspondiente secuencia secundaria.

Cada uno de los 3 clasificadores (MSV) recibe como entrada un conjunto de entrenamiento: $(Vcs_1, y_1), (Vcs_2, y_2,), \dots, (Vcs_m, y_m)$, donde $y \epsilon \Gamma$ es la etiqueta correspondiente a cada nivel estructural (Estructura secundaria), (Ver Tabla 6.).

Tabla 6. Matriz de codificación M

	f_1	f_2	f_3
H	1	1	0
E	-1	0	1
C	0	-1	-1

La matriz de codificación M relaciona los diferentes clasificadores binarios f_s que se pueden conformar mediante combinaciones de las clases Γ en las cuales se desea clasificar, en esta matriz se muestran las respuestas que se esperan de

cada clasificador binario cuando los datos provienen de una clase particular. El cero corresponde a los datos que no se contemplan para ese clasificador.

5.2.2.2 Decisión. Teniendo los clasificadores entrenados, y la secuencia primaria codificada se busca saber a qué clase $k \in \Gamma$ pertenece cada uno de los aminoácidos que la componen, para esto el Vcs pasa por los diferentes clasificadores f_s, para un dato x se tiene, Ver ecuación 16.

$$f(x) = f_1(x), f_2(x), f_3(x) \qquad (16)$$

La forma de encontrar la clase k es hallando la distancia mínima d entre las filas de la matriz M y el clasificador f_s.

La función de perdida L, Ver ecuación 18, permite encontrar el margen de perdida cuando el dato x es evaluado en un clasificador respecto a cada una de las filas de la matriz M, (Ver ecuación 17.).

$$L\big(M(y_i, s), f_{(x_i)}\big) = \sum_{j=1}^{l} \big(x_i - M_{ij}\big)^2 \qquad (17)$$

Para realizar la elección de la clase k en la cual se va a clasificar el dato x, nos basamos en el enfoque denominado decodificación basada en perdida, (Ver ecuación 18.), donde la ecuación permite encontrar la fila de la matriz M que presente la menor distancia respecto del vector $f_{(x)}$.

$$d_L\big(M(k), f(x)\big) = \sum_{s=1}^{l} L\big(M(k, s) f_s(x)\big) \qquad (18)$$

La clase k predicha $\hat{y} \in \{1,2, \ldots, k\}$ es:

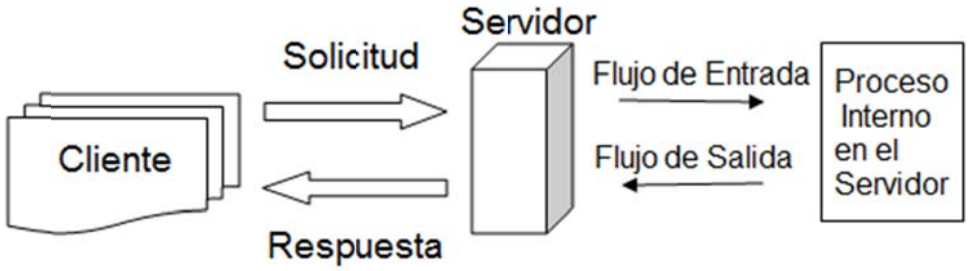
$$\hat{y} = arg_k \min d_L\big(M(k), f_{(x)}\big) \quad (19)$$

5.3 TERCER ETAPA

En esta etapa se realizó la implementación de la herramienta web.

5.3.1 Arquitectura Cliente-Servidor

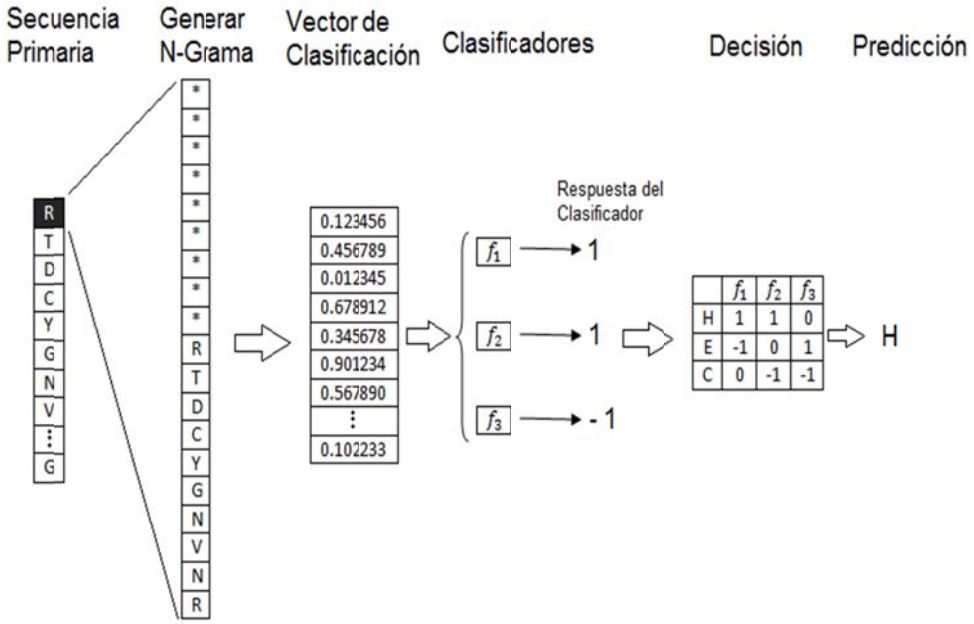
Figura 17: Arquitectura Cliente-Servidor.



Para este trabajo se eligió una arquitectura Cliente-Servidor [J. García, J.I. Rodríguez, A. Imaz, 2005], (Ver figura 17.), en la cual se lleva a cabo la implementación de la metodología expuesta, en un entorno web para realizar la predicción de la estructura secundaria de proteínas.

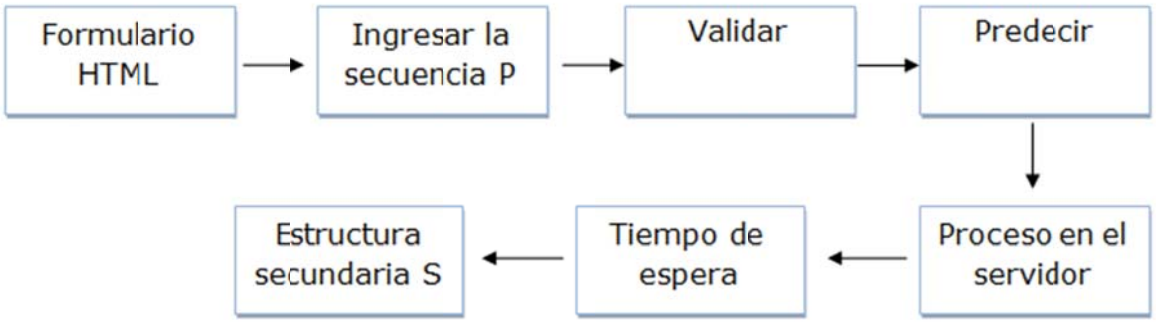
La estructura de la aplicación web se describe de la siguiente forma: En el navegador (Cliente) se ingresa la secuencia primaria en un formulario y se hace la solicitud al servidor para que un Servlet [J. García et al., 2005] realice el proceso de predicción de la estructura secundaria, (Ver figura 18.), y que genera una respuesta al cliente (Predicción).

Figura 18. Esquema del proceso de predicción en el Servlet.



El usuario ingresa una secuencia primaria P dentro de un formulario HTML, esta secuencia es validada antes que sea enviada al servidor y luego el usuario envía la solicitud de predicción accionando el botón de predicción el cual ejecuta el Servlet en el servidor generando un tiempo de espera para que realice la predicción que es devuelta al usuario mediante una página HTML que mostrara la estructura secundaria$S$. (Ver Figura 19.).

Figura 19. Esquema General



El implantar la predicción de estructuras secundarias de proteínas en la web es de gran importancia debido a la disponibilidad y fácil acceso que se ofrece, además es de gran ayuda para usuarios que necesitan disponer de herramientas bioinformáticas que aporten en la obtención de resultados en sus investigaciones.

6. PRUEBAS Y RESULTADOS

La herramienta se probó con la base de datos RS126 donde todas las proteínas fueron evaluadas por los mismos clasificadores y con los mismos parámetros para las máquinas de soporte.

El promedio de acierto en las predicciones se denomino Q, que indica el porcentaje de acierto de los tres niveles estructurales correspondientes a la estructura secundaria (H, E, -) para determinada estructura primaria, (Ver ecuación 20.).

$$Q = \frac{N_H + N_E + N_C}{N_T} \qquad (20)$$

Donde N_T es el número total de ejemplos de prueba y N_H, N_E, N_C corresponde al número correcto de aciertos para cada nivel estructural.

De manera independiente se calcularon los promedios de acierto para cada nivel estructural, los cuales se denominaron Q_H, Q_E, Q_C. (Ver Tabla 7.=

Tabla 7. Resultados

Q =	66,73%
QH =	75,70%
QE =	68,66%
QC =	58,97%

De manera individual se evaluó para cada clasificador f_s, la sensibilidad ($Sens$), la especificidad ($Espc$) y el coeficiente de correlación de Mathews (CCM). (Ver ecuaciones 21, 22 y 23.).

Considerando la predicción como un problema binario, en donde los resultados son etiquetados como positivo (p) o negativo (n) para cada una de las clases; hay cuatros posibles resultados para el clasificador binario f_s.

Si el resultado de una predicción es p y el valor real es p, entonces es llamado verdadero positivo (VP), si el valor real es n, entonces se tiene un falso positivo (FP); por el contrario, un verdadero negativo (VN) se produce cuando el resultado de la predicción y el valor real son n y un falso negativo (FN) es cuando el resultado de la predicción es n, mientras que el resultado real es p.

La sensibilidad mide la taza de aciertos positivos y la especificidad la taza de aciertos negativos, es decir aminoácidos correctamente clasificados en cada una de las clases.

El coeficiente de de correlación de Mathews proporciona una medida de la calidad del clasificador binario, un CCM de 1 o cercano a 1 indica que se construyo un clasificador eficiente, un CCM de 0 indica que el clasificador es aleatorio y un CCM de -1 o cercano a -1 indica que el clasificador es deficiente.

$$Sens = \frac{VP}{(VP + FP)} \qquad (21)$$

$$Espc = \frac{VN}{(VN + FN)} \qquad (22)$$

$$CCM = \frac{(VP * VN) - (FP * FN)}{\sqrt{(VP + FP) * (VP + FN) * (VN + FP) * (VN + FN)}} \qquad (23)$$

6.1 MEDIDAS DE RENDIMIENTO DE LOS CLASIFICADORES

- **Clasificador f_1.**

Corresponde al clasificador binario para las clases (HE)

$Sens = 78\%$
$Espc = 63\%$
$CCM = 0.69$ se considera un clasificador eficiente por acercarse a 1.

- **Clasificador f_2.**

Corresponde al clasificador binario para las clases (HC)

$Sens = 84\%$
$Espc = 55\%$
$CCM = 0.47$ se considera un clasificador aleatorio.

- **Clasificador f_3.**

Corresponde al clasificador binario para las clases (EC)

$Sens = 72\%$
$Espc = 96\%$
$CCM = 0.72$ se considera un clasificador eficiente por acercase a 1.

6.2 COMPARACIÓN CON OTROS MÉTODOS

Diferentes métodos como lo son Chou-Fasman, Garnier, Osguthorpe and Robson, Rost & Sander y Conformational Classification han aplicado con éxito en la

predicción de la estructura secundaria de proteínas, la metodología aplicada por nuestro método tiene una precisión del 66.73% (Ver Tabla 8[16].).

Tabla 8. Comparación con otros métodos

MÉTODO	PROMEDIO DE ACIERTO Q
Chou-Fasman	57%
Garnier, Osguthorpe and Robson	66%
Nuestro Método	66.73%
Rost & Sander	68 - 72%
Conformatioinal Classification	74.86%

[16] Tomado de: Guang Zheng Zhang, De-Shuang Huang, Hong Qiang Wang.Protein Secondary Estructure Prediction Based on the amino acids comformational classification and neural network technique, [2004].

7. CONCLUSIONES

Los resultados obtenidos con las máquinas de soporte vectorial para la predicción de la estructura secundaria de una proteína ratifican el poder de esta herramienta para llevar a cabo minería de datos sin importar su dimensionalidad, en este caso el éxito de las predicciones estuvo cercano al 67% lo cual se considera aceptable para este tipo de problemas.

Ya que las máquinas de soporte vectorial (MSV) son el producto de la integración del análisis multidimensional y la optimización, de la calidad de sus entradas depende el éxito de su implementación, es por esto que se debe realizar una selección apropiada de sus parámetros, el tipo de kernel y el método de optimización dependiendo del tipo de aplicación.

Se destaca la importancia de que las máquinas de soporte vectorial pueden ser utilizadas en un entorno web, para realizar la predicción de estructuras secundarias de proteínas.

El implantar la herramienta para la predicción de estructuras secundarias de proteínas en la web es de gran importancia debido a la disponibilidad y fácil acceso que ofrece, además es de gran ayuda para usuarios que necesitan disponer de herramientas bioinformáticas que aporten en la obtención de resultados en sus investigaciones.

8. RECOMENDACIONES

Es necesario desarrollar trabajos de investigación enfocados en la adecuación del conjunto de entrenamiento para reducir los tiempos de computo empleados por las MSV, pues el elevado volumen de datos hace que el tiempo de entrenamiento de estas sea muy costoso computacionalmente hablando y se haga imposible realizar una selección de parámetros del modelo para mejorar la precisión .

Se recomienda abordar el problema de seleccionar los parámetros adecuados de las MSV, escoger la función Kernel y sus parámetros para mejorar el rendimiento de las mismas.

Se recomienda abordar este tipo de problemáticas desde el punto de vista del procesamiento de alto rendimiento, pues el alto número de cálculos a realizar hacen de este problema una labor difícil de abordar desde el uso de herramientas de cómputo convencionales.

Se invita a continuar con la investigación en el problema de predicción de la estructura secundaria de proteínas ya que es una problemática aún no resuelta y la computación es uno de caminos más viables para encontrar soluciones económicas en tiempo y dinero para esta problemática.

BIBLIOGRAFIA

[Allwein et al., 2001] Allwein, E. L., Schapire, R. E., and Singer, Y. (2001).Reducing multiclass to binary: a nifying approach for margin classifiers.J. Mach. Learn. Res., 1:113–141.

[Biao Xu and Zhou, 2001] biaoXu, Y.-D. C. X.-J. L. X. and Zhou, G.-P. (2001). Support vector machines for predicting protein structural class.Guo-Ping Zhou, pages 1471–2105–2–3.

[Chen et al., 2007] Chen, C., Tian, Y., Zou, X., Cai, P., and Mo, J. Prediction of protein secondary structure content using support vector machine. Talanta, 71(5):2069–2073.

[Chin-Wei Hsu and Lin, 2008] Chin-Wei Hsu, C.-C.C. and Lin, C.-J. A practical guide ti support vector classification.

[C. Cortes and V. Vapnik, 1995]. Support vector networks. Machine Learning, 20:273-297.

[Cuff JA, 1999] Cuff JA, B. G. (1999). Evaluation and improvement of multiple sequence methods for protein secondary structure prediction proteins, page 9; 34:508 519.

[D. Zwillinger and K.H. Rosen, 1996] D. Zwillinger, S. G. K. and K.H. Rosen.Standard mathematical tables and formulae (30th edition).CRC Press.

[Delgado, Fuentes y Torres, 2010]. Predicción de la estructura secundaria de proteínas usando máquinas de soporte vectorial (inédito).

[Dietterich and Bakiri, 1994] Dietterich, T. G. and Bakiri, G. Solving multiclass learning problems via error-correcting output codes.J. Artif. Int. Res., 2(1):263–286.

[Ganapathiraju et al., 2004] Ganapathiraju, M., Klein-Seetharaman, J., Balakrishnan, N., and Reddy, R. Characterization of protein secondary structure. Signal Processing Magazine, IEEE, 21(3):78–87.

[Hua and Sun, 2001] Hua, S. and Sun. A novel method of protein secondary structure prediction with high segment overlap measure: support vector machine approach. Journal of Molecular Biology, 308(2):397–407.

[J. García et al., 2005] J. García, J.I. Rodríguez, A. Imaz (2005). Aprenda Servlets de Java como si estuviera en segundo.

[W. Kabsch and C. Sander, 1983] Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. Biopolymers, 22(12):2577–2637.

[Lukasz A. Kurgan, Leila Homaeian, 2006] Prediction of structural classes for protein sequences and domains-impact of prediction algorithms, sequence representation and homology, and test procedures on accuracy. Pattern Recogn, 39(12):2323–2343.

[N. Cristianini and J. Shawe-Taylor, 2000] Support vector machines and other kernel-based learning methods.Cambridge University Press, Cambridge MA, ISBN 0-521-78019-5.

[Nelson D. and Cox M., 2000]. Lehninger principles of biochemistry amino.Worth Publishers.

[Robert Harrison Phang, C. Tai Jieyue He, Wei Zhong, Yi Pan, 2006] Clustering support vector machines and its application to local protein tertiary structure prediction. Springer-Verlag Berlin Heidelberg, pages 710 - 717.

[Rost and Sander, 1993] Rost, B. and Sander, C. Improved prediction of protein secondary structure by use of sequence profiles and neural networks.Biophysics, Vol. 90.Pages 7558–7562.

[Ruan et al., 2005] Ruan, J., Wang, K., Yang, J., Kurgan, L. A., and Cios, K. J. Highly accurate and consistent method for prediction of helix and strand content from primary protein sequences. Artificial Intelligence in Medicine, 35(1-2):19–35.

[B. Scholkopf and A. Smola, 2002] Learning with kernels support vector machines, Regularization, Optimization and Beyond". The MIT Press, Cambridge.

[Shoyaib et al., 2007] Shoyaib, M., Baker, S., Jabid, T., Anwar, F., and Khan, H. (2007). Protein secondary structure prediction with high accuracy using support vector machine.Computer and information technology, 2007.Iccit 2007.10th international conference on, pages 1–4.

[Steven E. T. Hubbard C. Chothia A. Murzin, S. Brenner, 1995] Scop: a structural classification of protein database for the investigation of sequence and structures. Molecular Biology, 247:536–540.

[Steven M. Muskal and Sung-Hou Kim, 1992] Predicting protein secondary structure content: A tandem neural network approach. Journal of Molecular Biology, 225(3):713–727.

[SusanCostantini, Angelo M. Facchiano, 2009]. Prediction of the protein structural class by specific peptide frequencies.Biochimie, 91(2):226– 229.

[Trevor and Tibshirani, 1997] Trevor, H. and Tibshirani.Classification by pairwise coupling.The Annals of Statistics, 26:451–471.

[V. Vapnik, 1995] The nature of statistical learning theory. Springer Verlag, New York.

[Vladimir VapnikCorinna Cortes, 1995] Support-vector networks. Machine Learning, 20:273–297.

[Yang and Wang, 2003] Yang, X. and Wang, B. (2003). Weave amino acid sequences for protein secondary structure prediction. Pages 80–87.

ANEXOS

ANEXO 1. FUNCIONAMIENTO DE LA HERRAMIENTA

La herramienta software para predecir estructuras secundarias de proteínas a la cual se le dio el nombre de **SSPBIO** versión 1.0 se desarrollo en el lenguaje java bajo el entorno de programación NetBeans IDE 6.7.1.

Para la creación y entrenamiento de las Máquinas de Soporte Vectorial se utilizó la librería libsvm[17] para el entorno java.

La herramienta se encuentra disponible en el servidor de ciencias de la Universidad Industrial de Santander. (Ver Figura 20.)

Figura 20. Pagina inicial

[17] http://www.csie.ntu.edu.tw/~cjlin/libsvm/

El usuario ingresa la proteína (Secuencia Primaria) en la caja de texto como se muestra a continuación. (Ver figura 21.)

Figura 21. Ingreso de la secuencia

Se genera un mensaje de error si se ingresa una secuencia no valida al momento de pulsar el botón predecir. (Ver Figura 22.)

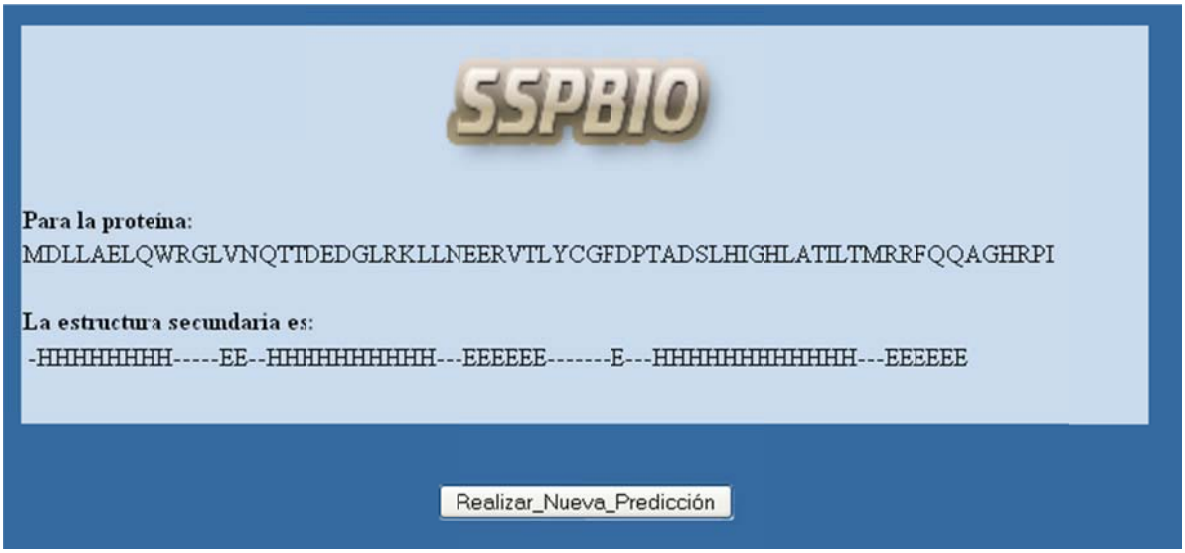
Figura 22. Validación de la secuencia.

Si la secuencia ingresada es correcta, se produce un tiempo de espera mientras se procesa la información antes de generar la predicción. (Ver Figura 23.)

Figura 23. Tiempo de espera

Una vez terminado el tiempo de espera la herramienta genera la respuesta con la predicción de la estructura secundaria para la proteína ingresada. (Ver Figura 24.)

Figura 24. Predicción



Se muestra en pantalla la predicción correspondiente a la proteína ingresada y se da la opción de realizar una nueva predicción.