MÉTODOS DE MACHINE LEARNING PARA EL APOYO AL DIAGNÓSTICO DE LA SUPERVIVENCIA DE PACIENTES CON SEPSIS BASADAS EN DATOS CLÍNICOS Y ZIMOGRAFIAS

SERGIO ALEXANDER GELVES MENDOZA
WANDA CATALINA RINCON CADENA

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERIAS FISICO-MECANICAS
ESCUELA DE INGENIERIA DE SISTEMAS E INFORMATICA
BUCARAMANGA
2015

MÉTODOS DE MACHINE LEARNING PARA EL APOYO AL DIAGNÓSTICO DE LA SUPERVIVENCIA DE PACIENTES CON SEPSIS BASADAS EN DATOS CLÍNICOS Y ZIMOGRAFIAS

SERGIO ALEXANDER GELVES MENDOZA
WANDA CATALINA RINCON CADENA

Trabajo de grado para optar al título de
Ingeniero de Sistemas

Director
Ph.D. RAUL RAMOS POLLAN

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERIAS FISICO-MECANICAS
ESCUELA DE INGENIERIA DE SISTEMAS E INFORMATICA
BUCARAMANGA
2015

DEDICATORIA

Dedico este trabajo primero que nada a Dios y a mi familia, a mi madre que se esfuerza día a día para que no me falte nada, brindándome su dulzura y apoyo en cada momento.

A mi hermana Ory, por su sabiduría, por devolverme los ánimos y hacerme reír cuando más lo necesito, por recordarme lo importante que es mantenerme motivada.

A mi padre por ser mi amigo y saber escuchar.

A mis amigos de la universidad que estuvieron ahí no sólo para estudiar sino para compartir tantos momentos que no olvidaré.

A mis maestros de la Universidad Industrial de Santander por dar ejemplo de integridad profesional.

A Sergio, mi compañero de proyecto y amigo, eres una persona con muchas capacidades y en este proyecto lo demostraste.

Wanda Rincon Cadena

DEDICATORIA

Este trabajo no hubiera sido posible sin mis tutores, siempre comprometidos y dispuestos a generar un cambio; sin mi familia, apoyándome a cada momento y brindándome tanto como tenían y por supuesto no hubiera sido posible sin mis amigos, también aquellos que ya no lo son, pero que siempre han sido poseedores de cualidades como la sensibilidad y el compromiso.
De todos ellos aprendí a aprender y aprendí a vivir. Gracias.

Sergio Alexander Gelves Mendoza

CONTENIDO

LISTA DE FIGURAS

LISTA DE TABLAS

GLOSARIO

Django: Framework para desarrollo web escrito en Python, sigue el paradigma Modelo-vista-controlador.

Enzima: Molécula proteica catalizadora de procesos químicos.

Machine Learning: Rama de la inteligencia artificial que tiene como objetivo desarrollar técnicas que le permiten a las máquinas aprender sin ser específicamente programadas, mediante un proceso de inducción. En un programa de aprendizaje automático se provee de un conjunto de datos de ejemplo (experiencia) a partir de los cuales se construye un modelo que calcula un resultado aproximado a los datos reales.

OpenCV: Biblioteca desarrollada por Intel para la visión por computacional, liberada bajo licencia BSD. OpenCV es multiplataforma y actualmente cuenta con interfaces C, C++, Python y Java.

Procesamiento digital de imágenes: Conjunto de técnicas que se aplican para el mejoramiento de imágenes y la extracción de información de estas.

Python: Es un lenguaje de programación de alto nivel, interpretado y de código abierto ampliamente usado. Cuenta con una gran cantidad de librerías que hacen de Python una herramienta adaptable a las necesidades de los desarrolladores. Puede ser empleado para, desarrollo de aplicaciones Web y de escritorio, computación científica, data science, machine learning y scripting.

Sepsis: Sepsis o septicemia es el síndrome de respuesta inflamatoria sistémica (SRIS) causado por una infección en el organismo.

Sustrato: Molécula sobre la que actúa una enzima.

Zimografía: Imagen obtenida a partir de técnicas de electroforesis en geles. Evidencia la actividad de enzimas en muestras de tejido.

RESUMEN

TITULO: MÉTODOS DE MACHINE LEARNING PARA EL APOYO AL DIAGNÓSTICO DE LA SUPERVIVENCIA DE PACIENTES CON SEPSIS BASADAS EN DATOS CLÍNICOS Y ZIMOGRAFIAS*

AUTORES:
GELVES MENDOZA, Sergio Alexander
RINCON CADENA, Wanda Catalina**

PALABRAS CLAVE: Machine Learning, Sepsis , Procesamiento de imágenes.

DESCRIPCION: El grupo Mediadores Inflamatorios y Enfermedad (M.I.N.E.N) de la Universidad Autónoma de Bucaramanga investiga la inmunología de la sepsis y otras enfermedades infecciosas. M.I.N.E.N recolectó datos clínicos desde enero del 2010 hasta diciembre del 2012 en el marco del proyecto G-Sepsis con el estudio del Valor Pronóstico de Marcadores Serológicos, Genéticos y Ecocardiográficos para Mortalidad en Pacientes Sépticos. Estos datos fueron organizados y almacenados por el sistema de información SIPPAM G-SEPSIS realizado por estudiantes de la Universidad Industrial de Santander en el 2012.

Los datos resultantes de la investigación se pre-procesaron y a partir de estos se entrenaron clasificadores usando técnicas de machine learning para predecir el grado de severidad de la enfermedad y la mortalidad en pacientes, estos métodos constituyen un avance para el soporte al diagnóstico de enfermedades infecciosas como la sepsis. A la vez se implementó un programa para la extracción automática de características de la imagen que indica la presencia de enzimas comunes en pacientes con la patología y se incluyó dentro de la base de datos de entrada de los clasificadores para concluir si la imagen contribuye a la mejora de la precisión de los clasificadores. Los clasificadores pueden ser usados y puestos a prueba por los profesionales de la salud a través de una web app.

* Trabajo de grado.

** Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingeniería de Sistemas e Informática. Director: Ph.D Raúl Ramos Pollan.

ABSTRACT

TITLE: MACHINE LEARNING METHODS TO SUPPORT DIAGNOSIS OF SURVIVAL IN PATIENTS WITH SEPSIS BASED ON CLINICAL DATA AND ZIMOGRAPHY [*]

AUTHORS: GELVES MENDOZA, Sergio Alexander
RINCON CADENA, Wanda Catalina[**]

KEYWORDS: Machine Learning, Sepsis, Image Processing.

ABSTRACT: The Group Mediadores Inflamatorios y Enfermedad (M.I.N.E.N) from the Universidad Autónoma de Bucaramanga investigates the immunology of the sepsis and other infectious diseases. M.I.N.E.N gathered clinical information from January 2010 to December 2012 during the project G-Sepsis con el estudio del Valor Pronóstico de Marcadores Serológicos, Genéticos y Ecocardiográficos para Mortalidad en Pacientes Sépticos. This data was organized and stored by the information system SIPPAM G-SEPSIS made by students from the Universidad Industrial de Santander in 2012.

The data resulting from research is pre-processed in order to train machine learning classiffiers to forecast the severity degree of the disease and the mortality in patients, these methods represent progress supporting the diagnosis of infectious diseases like sepsis. Simultaneously a program was implemented for automatic extraction of image features indicating the presence of common enzymes in patients with the pathology, this features were included within the database entry classifiers to conclude whether the images improves classifiers' accuracy. The classifiers can be used and tested by health professionals through a web app.

[*] Bachelor Thesis.

[**] Faculty of Physical-Mechanical Engineering. School of Engineering and Computer Science. Director:Ph.D Raúl Ramos Pollan.

INTRODUCCIÓN

Las enfermedades son casos de estudio en cualquiera parte del mundo, en muchos casos sus tratamientos o diagnósticos pueden variar según marcadores serológicos, genéticos o ecocardiográficos, entre otros. Es importante notar que involucrar genes en investigaciones puede ser equivalente a estudiar aspectos demográficos en el área en cuestión, es decir que es un factor que relaciona y permite distinguir poblaciones y sus enfermedades. Por esto y la alta tasa de mortalidad en pacientes con la enfermedad sepsis, el grupo MINEN comenzó a recolectar datos, desde enero del 2010, para el estudio Valor Pronóstico de Marcadores Serológicos, Genéticos y Ecocardiográficos en Pacientes Sépticos, para la región de Santander.

El grupo MINEN durante su investigación presentó primeramente la necesidad de involucrar el estudio con las tecnologías de información, con lo cual surgieron sistemas de información para la administración de datos. Paralelo a esto se realizaron exámenes clínicos usando las muestras sanguíneas captadas en el estudio, el resultado fue una base de datos con sesenta y tres imágenes denominadas zimografías, cada una con capacidad para demostrar la actividad enzimática de las Metaloproteinasa MMP-2 y MMP9, de hasta nueve pacientes diferentes. La medición de dicha actividad se realizó con la herramienta libre ImageJ que la cuantifica usando un algoritmo de tratamiento de imágenes basado en la intensidad de pixeles.

Una hipótesis que adoptó el grupo MINEN es que la cantidad y actividad de MMP-9 y MMP-2 en pacientes sépticos influye en la severidad de la enfermedad y por lo tanto en la mortalidad, lo cual dota de relevancia a la zimografía como principal y único examen que representa a la actividad enzimática en el estudio.

El objetivo principal de este proyecto es medir la actividad enzimática demostrada en las zimografías y relacionarla con la información captada para pacientes sépticos, en miras a un apoyo al diagnóstico médico. Para procesar la zimografía se propone usar técnicas de procesamiento de imágenes o *feature learning*, paralelamente se construirán modelos predictivos usando técnicas de aprendizaje de máquina para predecir la severidad de la sepsis o la mortalidad; al unirse las nuevas mediciones sobre las zimografías y los modelos predictivos se podría estimar la validez de las nuevas mediciones.

En el primer par de capítulos encontramos los detalles sobre el origen del proyecto y sus objetivos, seguido en el segundo par está el fundamento teórico y el estado

del arte de los algoritmos de procesamiento de enzimas o técnicas de aprendizaje de máquina aplicadas al sector salud. El capítulo cinco describe la forma como aplicamos SCRUM al proyecto, sus objetivos y entregables cuyos detalles se encuentran en el siguiente capítulo, el desarrollo del proyecto. Para terminar se encuentran las conclusiones y las recomendaciones cuando se use la información del presente libro o se quiera profundizar en el campo de estudio.

1. DEFINICIÓN DEL PROBLEMA

Se denomina sepsis al Síndrome de respuesta inflamatorio sistémica (SRIS) causado por infecciones en tejidos o fluidos en el organismo. Tiene una alta incidencia de mortalidad en las unidades de cuidados intensivos debido a dificultades en el diagnóstico y seguimiento del estado del paciente después de su ingreso. En las últimas décadas se han realizado investigaciones en busca de una mejor comprensión del síndrome, para optimizar el diagnóstico y su tratamiento. Sin embargo, los estudios centrados en muestras de países no han podido obtener resultados generalizables para la población mundial, por esta razón en el departamento de Santander investigadores del grupo Mediadores Inflamatorios y Enfermedad (MINEN) comenzó el proyecto G-Sepsis con el estudio del Valor Pronóstico de Marcadores Serológicos, Genéticos y Ecocardiográficos para Mortalidad en Pacientes Sépticos, en el que se captaron datos de 530 pacientes desde enero del 2010 hasta diciembre del 2012, estos datos se organizaron en el Sistema de Información para la Gestión de Datos y Administración del Estudio de Detección de Marcadores Pronósticos de Mortalidad por Sepsis (SIPPAM G-SEPSIS) hecho por estudiantes de ingeniería de sistemas, posteriormente fue creado un sistema para optimizar el proceso de obtención, presentación y manipulación de los datos obtenidos a partir del estudio y análisis de la enfermedad, llamado Codeware G-Sepsis.

Adicionalmente el grupo MINEN incursionó en una nueva metodología para la apreciación de actividad enzimática sobre muestras de sangre de pacientes sépticos, que da como resultado una imagen llamada zimografía que actualmente es procesada manualmente por un profesional de la salud usando el software ImageJ dando resultados cuantificados difíciles de interpretar y susceptibles a errores humanos dentro del procedimiento. Por lo cual se hizo evidente la necesidad de procesar la imagen automáticamente con otros métodos que den resultados que permitan asociar una medida fácil de interpretar, sobre la zimografía.

Como resultado de la investigación se encuentra disponible una gran cantidad de información de pacientes sépticos en Santander; en busca de métodos que puedan arrojar más conocimiento de la enfermedad se ha planteado una solución que utilice técnicas de *machine learning* para su uso en la predicción de la etapa de la enfermedad, así como en la probabilidad de mortalidad, con el objetivo de apoyar el diagnóstico médico. Las diversas técnicas de aprendizaje de máquina permiten analizar el conjunto de datos y obtener resultados de gran utilidad para la investigación de la sepsis
.

2. OBJETIVOS GENERAL Y ESPECÍFICOS

2.1 OBJETIVO GENERAL

Desarrollar y evaluar técnicas de machine learning y feature learning para la clasificación de pacientes en los diferentes estadios de la enfermedad sepsis y pronosticar la supervivencia para el apoyo al diagnóstico.

2.2 OBJETIVOS ESPECÍFICOS

1. Diseñar y evaluar métodos para la cuantificación de la actividad enzimática en las zimografías.

2. Evaluar y seleccionar métodos de pre-procesado de datos.

3. Evaluar y seleccionar métodos de aprendizaje supervisado para la predicción de la supervivencia de pacientes sépticos.

4. Evaluar y seleccionar métodos de aprendizaje supervisado para la clasificación de pacientes según la severidad del síndrome.

5. Integrar los métodos anteriores en una herramienta usable por los médicos y evaluar su utilidad en la predicción de los desenlaces de la enfermedad.

3. MARCO TEÓRICO

3.1 SEPSIS

Sepsis o septicemia es el síndrome de respuesta inflamatoria sistémica (SRIS) causado por una infección en el organismo, la severidad de la enfermedad se encontrará clasificada en tres estados: sepsis no grave, sepsis grave (severa) y choque séptico [1]. Los patógenos buscan incapacitar el sistema inmunológico del paciente. La prevención de la inmunosupresión inducida por sepsis, o su mismo tratamiento, es una prioridad para los médicos o especialistas.

El Síndrome de la reacción inflamatoria sistémica (SRIS) presenta la existencia de dos o más de las cuatro condiciones siguientes:

- Fiebre (temperatura oral > 38 ºC) o hipotermia (temperatura oral < 36 ºC).
- Taquicardia (> 90 latidos por minuto).
- Taquipnea (> 24 respiraciones por minuto), o hiperventilación (presión parcial arterial de CO2 < 32 mm Hg) o necesidad de ventilación mecánica.
- Leucocitosis (> 12.000/mm3), leucopenia (< 4.000/mm3), o > 10 % de formas juveniles en el recuento de leucocitos.

3.1.1 Sepsis no grave

Respuesta grave a una infección debida a la presencia de bacterias en cualquier tejido u órgano del cuerpo. Cuando hay presencia de bacterias en la sangre acompañado de sepsis se denomina septicemia, los primeros síntomas son: fiebre alta, frecuencia cardiaca alta, respiración acelerada, escalofríos. Los síntomas progresan rápidamente llevando a una confusión u cambio en el estado mental, manchas rojas en la piel, shock.

3.1.2 Sepsis Severa

Los órganos ven afectado su funcionamiento y se presentan síntomas como hipotensión o hipoperfusión. Además, los pacientes con sepsis severa tienen una tasa de mortalidad mayor que en el estado anterior.

3.1.3 Choque Séptico

En esta etapa el paciente ha presentado estado de hipotensión por más de dos horas y tiene anomalías en la perfusión tisular (volumen de sangre presente en los tejidos vasculares). El choque séptico puede llevar a fallo multiorgánico y la muerte.

Los pacientes con mayor riesgo de entrar a este estado son aquellos con factores como: diabetes, leucemia, linfoma, infección reciente, uso reciente de esteroides, trasplante de órgano sólido o médula ósea, entre otros.

En el choque séptico la tasa de mortalidad es alta que depende de la edad del paciente y su salud en general, de la causa de la infección, la cantidad de órganos que presentan insuficiencia, entre otros. [1]

3.2 MACHINE LEARNING

Machine Learning o Aprendizaje de máquina es un conjunto de técnicas que le permiten a un programa de computadora aprender de un conjunto de datos. Una definición comúnmente citada es la de Tom M. Mitchell que como científico en computación, reconocido por sus aportes al área de aprendizaje de máquina, expresó: “Se dice que un programa de computadora aprende de una experiencia E respecto a unas clases de tareas T y con medida de desempeño P; si su desempeño en dichas tareas T, con una medida P, mejora con la experiencia E.” [2]

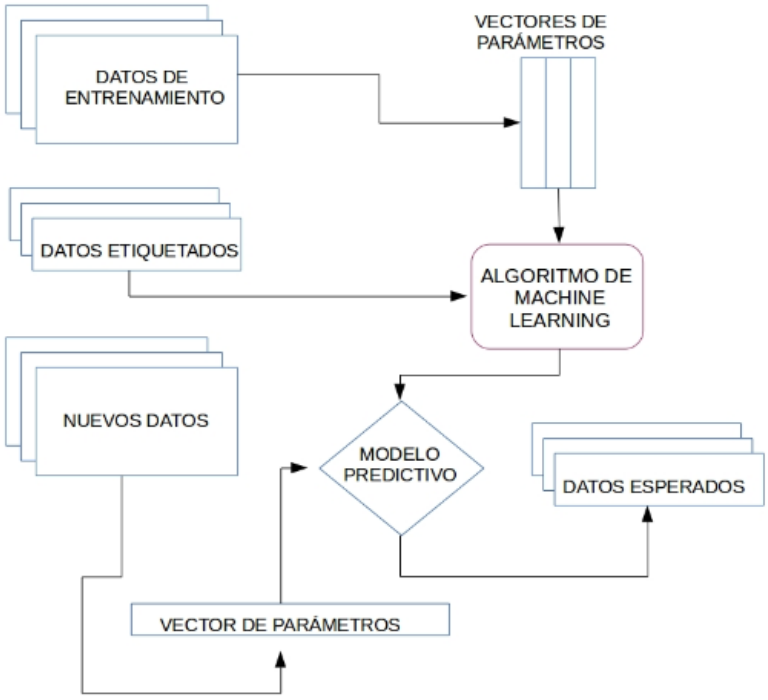
De forma más específica el aprendizaje de máquina tiene dos grandes ramas diferenciadas por el conjunto de datos de entrada, que le proveen de experiencia al programa de computadora. El aprendizaje supervisado, que es concerniente al presente libro, recibe como conjunto de entrada ejemplos o sujetos que ya se encuentran etiquetados, es decir que según sus características dichos sujetos son distinguibles unos entre otros por unas etiquetas que los clasifican. Mientras que en el aprendizaje no supervisado, no hay unas etiquetas de entradas que permitan

[1] SHOCK SÉPTICO. ENCICLOPEDIA MÉDICA. A.D.A.M. [en línea] <http://www.nlm.nih.gov/medlineplus/spanish/ency/article/000668 >[citado en 21 de octubre del 2014]

distinguir los sujetos, por lo tanto su tarea de clasificación involucra también definir cuáles son las etiquetas pertinentes al problema.

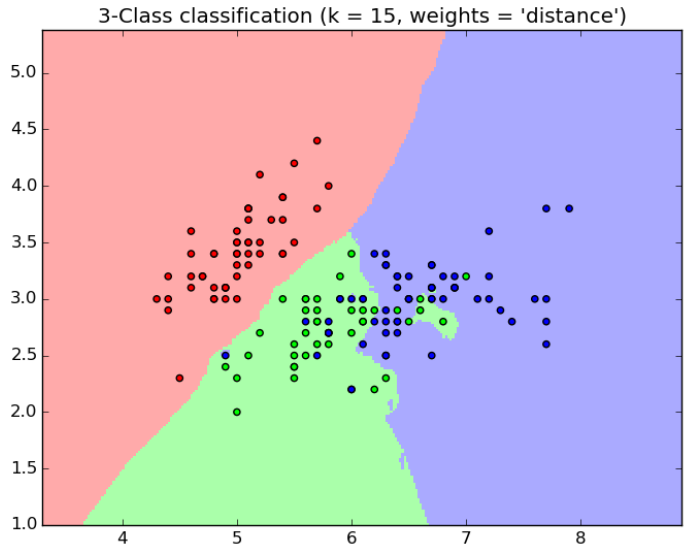
Ahora, en un programa de aprendizaje supervisado se provee de un conjunto de datos de entrada el cual se suele segmentar en subconjuntos de datos para entrenamiento y pruebas, que proveen de experiencia y medidas de desempeño respectivamente, a partir de los cuales se construye un modelo predictivo que calcula resultados aproximados para nuevos datos.

Figura 1. Esquema del flujo de datos en las técnicas de aprendizaje supervisado



El modelo define entonces fronteras de clasificación para datos de entrada, es importante notar que dichos datos se clasifican dimensionalmente según las características que poseen las observaciones, como puede ser la presión arterial y el nivel de azúcar en la sangre, que definen dos dimensiones para pacientes en un conjunto de entrada. Como se puede ver en la siguiente figura 2, una frontera de clasificación se realiza evaluando puntos del espacio de entrada en el modelo predictivo.

Figura 2. Frontera de clasificación para un clasificador K-Vecinos



Fuente: Documentación Scikit-learn.

3.2.1 Clasificación en Aprendizaje Supervisado

La clasificación es una parte del aprendizaje supervisado que se aplica a los datos que se encuentran etiquetados con números enteros, caracteres o cadenas de caracteres. Para aquellos datos etiquetados con una infinidad de posibilidades, como los números reales, se abordan con técnicas de regresión, que no serán explicadas dado que no convienen al problema en cuestión. Los problemas se abordan con un algoritmo, también llamados estimador o clasificador, que generan un modelo.

Un problema de clasificación es equivalente a un problema de optimización matemática, dado que un modelo consta de un conjunto de parámetros

$$\theta = \theta_0 + \theta_1 + \theta_2 + \cdots \theta_n$$

Que al interactuar con el conjunto de entrada X se predicen la clase, o la probabilidad de que pertenezca a una clase, de un sujeto dado x.

$$h_\theta(x) = \text{etiqueta, probabilidad}$$

Como el fin es mejorar la predicción, se define una función objetivo o de costo que se optimiza para obtener los θ_i que generan menor costo y por lo tanto mejores predicciones.

$$^{argmin}_{\theta} J(\theta)$$

Pero cuando la función de costo se optimiza a tal punto que los θ_i hallados me permiten predecir muy bien el conjunto de entrada, entonces se ajustan a las irregularidades o el ruido, por lo que se generan modelos que no generalizan y por ende su desempeño prediciendo datos nuevos no va a ser eficiente. Una solución a dicho problema es regularizar la función de costo, como es el caso de la regresión logística:

$$^{argmin}_{\theta} J(\theta) + \frac{\lambda}{m}\sum_{i=1}^{n} \theta_i$$

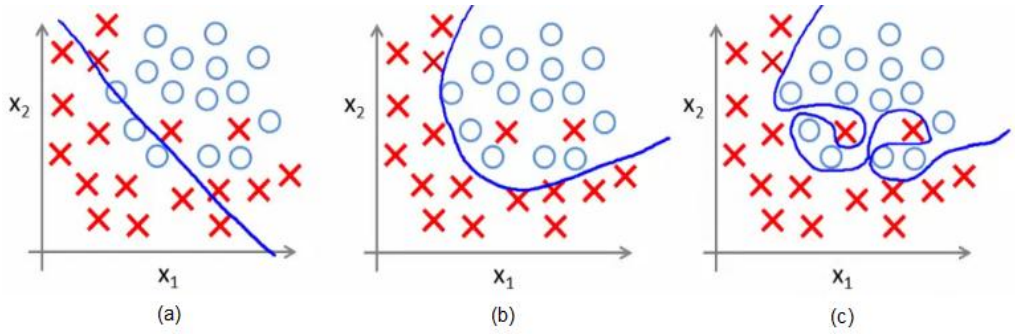
Donde λ es el coeficiente de regularización y m la cantidad de sujetos en el conjunto de entrada, ambas constantes, pero a λ se le considera hiperparámetro.

3.2.1.1 Overfitting y Underfitting

Cuando el modelo que se pretende crear no se ajusta a la complejidad de los datos, los datos son insuficientes, la función de costo no se optimiza adecuadamente o no es la apropiada, entre otras razones, se genera underfitting o bias. En la siguiente figura parte (a) se observa que una línea recta no es la adecuada para dividir a las dos clases, por el contrario en la parte (b) es evidente que la frontera de clasificación se asemeja más a una función polinómica.

Por otro lado, el overfitting se presenta en modelos definidos para una complejidad de datos menor a la supuesta o por una optimización de la función de costo sin considerar la importancia de construir un modelo que generalice. Cuando se trata de tener en cuenta cada punto en el conjunto de entrada también se presenta un sobre ajuste a los datos, dado que es más probable que las todas las pequeñas variaciones sean ruido en vez de señales verdaderas. En la figura 3 parte (c), se observa que la frontera se ajusta para todos los puntos del conjunto de entrenamiento y que es bastante ondulada, algo poco probable, se establece así una clasificación que no va conforme a un posible patrón o naturaleza de los datos. Este problema genera a su vez varianza (*variance*), que consiste en una frontera de clasificación cambiante con los datos de entrada, lo cual crea diferentes fronteras que no generalizan sobre la naturaleza de los datos.

Figura 3.Ajuste de la complejidad de un modelo (a) Underfitting (b) Buen ajuste (c) Overfitting.



3.2.1.2 Hiperparámetros

Los hiperparámetros son valores que incrementan o disminuyen la precisión de un clasificador, pero su valor óptimo no se optimiza con la función costo que tiene involucra un algoritmo: $^{argmin}_{\theta} J(\theta)$, por lo tanto la búsqueda de los hiperparámetros óptimos influye en la precisión de la predicción, como es el caso de λ muy grande: $^{argmin}_{\theta} J(\theta) + \frac{\lambda}{m}\sum_{i=1}^{n}\theta_i$, se evitaría que el modelo se ajuste a los datos, underfitting, porque se le da más importancia al valor de los θ_i que al ajuste de los datos.

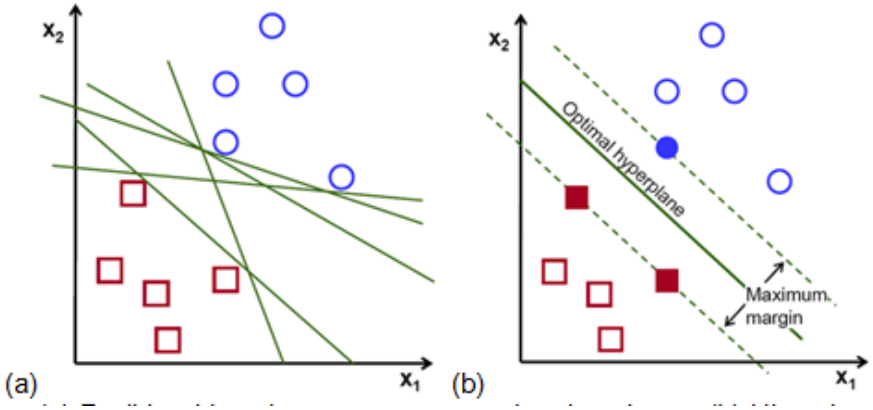
3.2.2 Algoritmos de clasificación

Los algoritmos más usados en tareas de clasificación se explicarán a continuación, mostrando su lógica, planteamiento matemático, pero no su implementación, que varía según las herramientas usadas e inclusive una misma herramienta puede proveer de diferentes implementaciones, que convienen según las dimensiones del conjunto de entrada o el tipo de datos que este contiene.

3.2.2.1 Máquinas de soporte Vectorial

Las máquinas de soporte vectorial son un conjunto de algoritmos desarrollados por Vladimir Vapnik y su equipo en los laboratorios AT&T, que buscan dentro de un espacio de alta dimensionalidad el hiperplano que separe con margen máximo las clases de un conjunto de datos. Los vectores formados por los puntos más cercanos al hiperplano se denominan vectores de soporte.

Figura 4. : Hiperplanos (a) Posibles hiperplanos que separan las dos clases. (b) Hiperplano con margen máximo a las dos clases que separa.



Fuente: Documentación OpenCV.

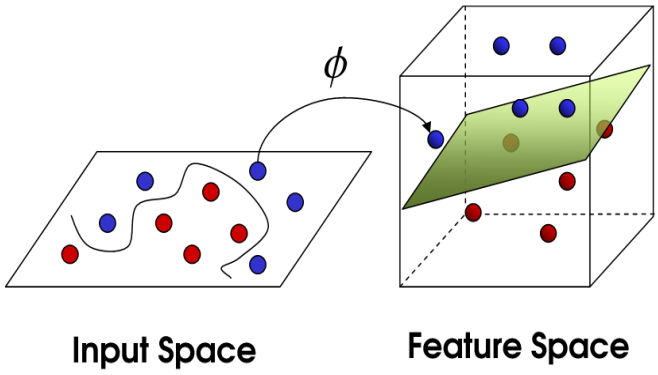
Los algoritmos de máquinas de soporte vectorial utilizan funciones kernel para operar en un espacio multidimensional sin transformar los datos explícitamente, la única operación que se realiza con los datos es el producto punto escalar, lo cual disminuye el costo computacional. Esto se logra mediante las funciones kernel,

donde cada una tiene sus propios parámetros y el resultado en el espacio original es no-lineal. Podemos expresar una función kernel de la forma:

$$K(x,x') = < \varphi(x), \varphi(x') >$$

Que representa similaridad entre los vectores x, x' (datos de entrada) son elementos y φ es una transformación que se realiza en los datos y para $\langle x, x' \rangle$ definido como el producto punto entre la matriz x y su transpuesta.

Figura 5. Cambio de representación del espacio de entrada.



Fuente: http://i.stack.imgur.com/1gvce.png

Kernel lineal:

$$K(x,x') = \gamma + c$$

El producto punto punto sumado a una constante opcional c.

Kernel polinomial:

$$K(x,x') = (\gamma * + r)^d$$

El kernel polinomial es adecuado en problemas con datos normalizados, d será el grado de las variables polinomiales.

Kernel Gaussiano:

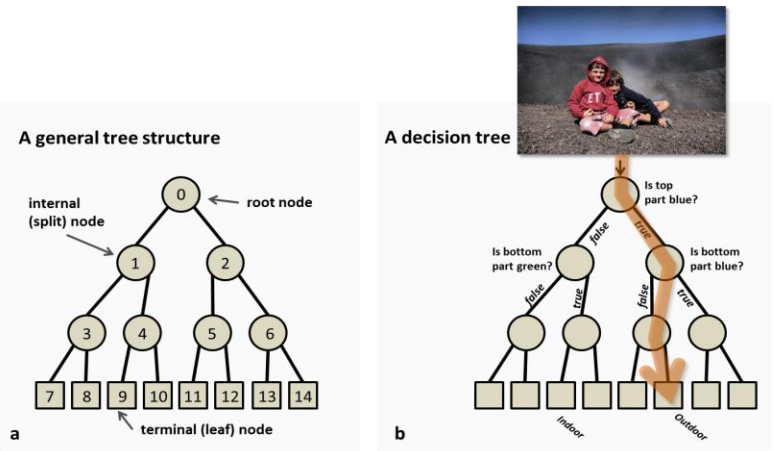
$$K(x, x') = e^{-\gamma|x-x'|^2}$$

En la ecuación se reconoce la distancia euclidiana entre los vectores de entrada. La variable γ afecta el desempeño de la función por lo cual debe ser elegido cuidadosamente. Significa que tanta influencia tiene un solo ejemplo de entrenamiento. [3]

3.2.2.2 Árboles de Decisión

Un árbol es un grafo dirigido, sin bucles, que tiene siempre: un nodo raíz, nodos hojas o finales, nodos internos (o de división, diferentes a los finales o raíz) y aristas (conectores entre los nodos). Cada nodo del árbol contiene una pregunta y posibles repuestas a dicha pregunta, por ejemplo para un árbol binario que busca identificar si una foto fue tomada en el exterior o en un interior, la primera pregunta es: ¿La parte superior es azul?, ésta solo tiene dos respuestas, dependiendo de la respuesta voy a ir a alguna de dos posibles preguntas (nodos). Para la siguiente figura se tiene que efectivamente la parte superior es azul, por lo que se pregunta si la parte inferior es azul también, porque podría ser un interior cuyo fondo es una pared azul, la respuesta es que la parte inferior no es azul, por lo que se realiza otra pregunta para concluir que es una foto tomada en el exterior.

Figura 6. Arboles de decisión. (a) Estructura general de un árbol de decisión: nodo raíz, nodo interno y nodo final (b) Árbol de decisión binario para identificar si una foto se tomó en el exterior o en un interior.



Fuente: A. Criminisi. Decision Forests for Classification,Regression, Density Estimation, Manifold Learning and Semi-Supervised Learning.

Cuando se ha evaluado todo el conjunto de entrenamiento en el árbol de decisión se tiene una correspondencia de cada sujeto con los nodos finales, es decir que se generó una partición en los datos, o particionado recursivo, y a cada nodo final se le asigna un pequeño modelo para evaluar los nuevos datos de entrada. Por esto el modelo global tiene dos partes: una que es el particionado recursivo y la otra que es el modelo asociado a cada nodo final [4].

Planteamiento matemático

Dado un conjunto de entrenamiento $x_i \in R^n$, con *i= 1,2,.., l* y un vector de etiquetas $y \in R^l$; Un árbol de decisión particionará el espacio de tal forma que los sujetos con mismas etiquetas quedarán juntos.

Los datos en un nodo m serán representados por Q. Ahora, para cada división $\theta = (j, t_m)$ que involucra una característica j y un umbral t_m., los subconjuntos de datos respectivos a los nodos de la división se identifican con:

$$Q_{left}(\theta) = (x, y)|x_j <= t_m$$
$$Q_{right}(\theta) = Q \setminus Q_{left}(\theta)$$

Donde la impureza en m se computa usando la función de impuridad $H()$, como puede ser *gini, cross-entropy* y *missclassification.*

$$G(Q, \theta) = \frac{n_{left}}{N_m} H(Q_{left}(\theta)) + \frac{n_{right}}{N_m} H(Q_{right}(\theta))$$

Como se puede observar, la función de impuridad depende del nodo m, más específicamente de los datos Q que allí se encuentran. Y depende de los parámetros $\theta = (j, t_m)$, sobre ellos se busca minimizar la impureza:

$$\theta^* = \text{argmin}_\theta \, G(Q, \theta)$$

Proceso que se realiza para cada subconjunto $Q_{left}(\theta^*)$ y $Q_{right}(\theta^*)$ hasta que se alcance la profundidad máxima permitida para el árbol, $N_m < \min_{samples}$ (cantidad de sujetos en el nodo sea menor que el mínimo número de sujetos permitidos por nodo) o que $N_m = 1$ [5].

3.2.2.3 Naive Bayes

Es un método de clasificación basado en el Teorema de Bayes con una suposición "ingenua" (*naive*) sobre la independencia de los datos, donde cada par de características es independiente entre sí. Se le considera un método generativo dado que para realizar su predicción tiene que realizar un paso intermedio, aprender de la probabilidad conjunta (*joint probability*) $P(x, y)$ con x una observación del conjunto de entrenamiento, y su etiqueta, para calcular la clase más probable $P(y|x)$. [6]

Planteamiento Matemático: Gaussian Naive Bayes

La formulación matemática varía según la forma de calcular $P(y|x)$ para predecir la clase más probable. Para ello se tienen en cuenta el conjunto de entrada de dimensiones $[m,n]$, m sujetos con n características, donde con cada sujeto $x = [x_0, x_1, x_2, \ldots, x_n]$ y cada clase C_i se debe calcular:

$$P(x|C_i) = P(C_i) \prod_{j=0}^{n} P(x_j|C_i)$$

Se asume que cada columna tiene una distribución gaussiana diferente:

$$P(x_j|C_i) = \frac{1}{\sigma_{ij}\sqrt{2\pi}} * e^{\frac{-(x-\mu_{ij})^2}{2\sigma_{ij}^2}}$$

Donde μ_{ij} y σ_{ij} son la media y la desviación estándar para cada columna i y clase j. La predicción se realiza maximizando la probabilidad:

$$\underset{C_i}{argmax} P(x \vee C_i)$$

3.2.3 Ensembles

Los *ensembles* son conjuntos de estimadores base que en vez de ofrecer una predicción realizan varias, provenientes muchas veces de modelos de diferente tipo, y se selecciona la mejor predicción mediante voto. Según la forma como el voto afecta a la predicción, los clasificadores que se usan y el particionado de los datos de entrada, se divide los *ensembles* en varios tipos.

3.2.3.1 Voting

Los modelos se construyen de forma independiente y su predicción se calcula a partir de un promedio ponderado, mínimo, máximo, la mediana de las predicciones de sus clasificadores base, entre otros criterios. El conjunto de datos de entrada puede ser diferente, como el sonido acústico de la voz o un video que capta el movimiento de los labios para predecir qué palabras se dicen en un discurso, o

transformaciones diferentes de datos para cada estimador base. Lo que se busca es que los clasificadores sean diversos y así mejorar el aporte que hacen a la predicción. Un *ensemble* creado mediante votación (*voting*) puede ser el *averaging*, donde la predicción final es el promedio de las predicciones individuales.

3.2.3.2 Bagging

En esta técnica se usan estimadores base débiles (*weak learners*), el objetivo es muestrear el conjunto de entrada respecto a sus sujetos y características, lo que propicia una distribución de la clasificación donde cada estimador base débil tendrá mejor desempeño en predecir solamente ciertas clases. Un ejemplo de bagging es el *Random Forest Classifier.*

3.2.3.3 Boosting

Para *boosting* se especializa cada clasificador en ciertas clases, los *weak learners* se enfocan en aquellos sujetos que no fueron bien clasificados, cuya predicción erró, es decir que si un primer clasificador base tiene una precisión del 80%, el siguiente clasificador enfatizará en predecir mejor el 20% en el que hubieron fallas. Un algoritmo muy usado para implementar esta técnica es *Adaboost.*

3.2.3.3 Precisión contra Diversidad

Cuando se construye un modelo basándose solo en un algoritmo; sea K-Neighbors, SVM, Naive Bayes, entre otros; se optimizan sus θ_i e hiperparámetros para obtener la mejor precisión posible. En el caso de los ensambladores, dicha búsqueda de la precisión ideal es innecesaria, dado que si tenemos una precisión del 70% sobre el conjunto de entrada, debemos enfocarnos en mejorar el desempeño sobre el 30% restante, lo cual se puede lograr incluyendo diferentes estimadores que tengan éxito prediciendo sobre los datos en los que otros fallaron. Además, los algoritmos de clasificación realizan asunciones sobre los datos de entrada, lo cual introduce un sesgo que impide predecir correctamente todo los datos para grandes conjuntos de entrada. La forma de disminuir el sesgo introducido es implementar diferentes algoritmos [7].

3.2.4 Validación de Modelos

Después de construir un modelo podemos evaluar su precisión mediante métricas como son el TPR, TNR, *accuracy*, *precision*, f1, AUC, entre otras. Pero, dichas medidas no son relevantes si las aplicamos solamente al conjunto de entrenamiento del modelo, por eso hay que usar las métricas sobre datos que no fueron incluidos en la fase de entrenamiento. Como los datos que permiten construir el modelo contienen variaciones y ruido, existen técnicas para disminuir el error midiendo la precisión sobre observaciones que no habían sido vistas.

3.2.4.1 Métricas o Scores

Las métricas son estadísticos que revelan características en una predicción, algunas son útiles para medir la precisión sobre una clase dada, otras calculan el desempeño general de la predicción. Se usan como el objetivo de un modelo o para medir el desempeño del mismo.

Aunque cualquiera puede definir sus propias métricas, ya hay unas aceptadas de forma general y con objetivos claros, definidas sobre la falsedad o veracidad de la predicción. A continuación se presentan métricas para problemas de clasificación dicotómicos.

Tabla 1. Tabla de verdad binaria para Falso y Verdadero vs Negativo y Positivo.

	Negativo	**Positivo**
Falso	FN= Falso Negativo	FP= Falso Positivo
Verdadero	VN = Verdadero Negativo	VP = Verdadero Positivo

3.2.4.1.1 Recall, True Positive Rate (TPR) o Sensibilidad

La sensibilidad es una métrica para evaluar la predicción de las instancias positivas, generalmente la clase uno (1) para un problema de clasificación binario. Hay que notar que las instancias negativas verdaderas, clase cero (0), no se toman en cuenta en esta métrica, por lo tanto los clasificadores medidos solamente con *recall* pueden tender a clasificar los datos como positivos, lo que

genera falsos positivos (clasifican datos en clase uno cuando son realmente clase cero).

$$TPR = \frac{VP}{P} = \frac{VP}{VP + FN}$$

3.2.4.1.2 True Negative Rate (TNR) o Especificidad

La especificidad mide la predicción respecto a las instancias negativas, su valor crece cuando las instancias negativas que fueron clasificadas positivas, falsos positivos, disminuyen. Usar solo la especificidad como métrica puede generar que un modelo tienda a clasificar los sujetos como instancias negativas, lo que causa falsos negativos y disminuye el desempeño del modelo para clasificar sujetos positivos.

$$TNR = \frac{VN}{N} = \frac{VN}{VN + FP}$$

3.2.4.1.3 False Positive Rate (FPR) o Fall-out

Es el complemento de la especificidad y mide la proporción de falsos positivos respecto a los negativos. Cuando se minimizar esta métrica disminuyen las fallas que cometa un modelo al clasificar los datos negativos, a su vez que evita una tendencia a clasificar como los datos como positivos.

$$FPR = 1 - TNR = \frac{FP}{N} = \frac{FP}{FP + TN}$$

3.2.4.1.4 Precision o Positive Predictive Value (PPV)

El valor predictivo positivo se define como el objetivo de aumentar los sujetos clasificados correctamente como positivos, pero busca disminuir los sujetos negativos clasificados como positivos (falsos positivos).

$$PPV = \frac{VP}{VP + FP}$$

3.2.4.1.5 Accuracy, Precisión o Exactitud

Accuracy se traduce a español como precisión o exactitud, por lo que se debe distinguir bien entre la precisión que tiene un modelo en la tarea de clasificación y la medida de la métrica precisión (*accuracy*) para un modelo. El objetivo de la *accuracy* es mejorar el desempeño general, clasificar sujetos positivos y negativos de forma correcta. Pero por su definición matemática, generalmente no es la métrica correcta para conjuntos de entrada desbalanceados.

$$acc = \frac{TP + TN}{P + N} = \frac{TP + TN}{(TP + FN) + (TN + FP)}$$

Por ejemplo, si tengo un conjunto de entrada A, donde los sujetos negativos son en total n, mientras que el conjunto de los sujetos positivos tiene un tamaño de $10n$, se supone que se clasificaron todos los datos como negativas, obteniendo así un *accuracy* de 0.91 (91%), lo cual es una medida errónea porque la clasificación no es correcta.

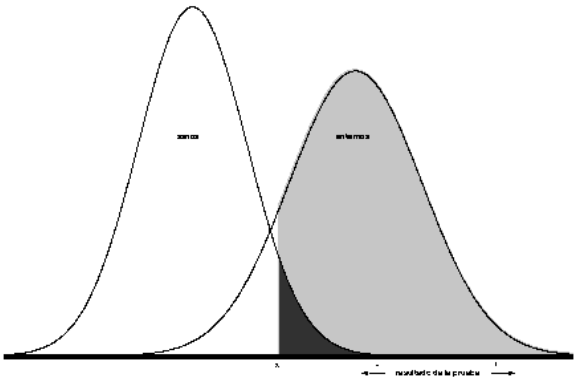
$$acc = \frac{TP + TN}{P + N} = \frac{0 + 10n}{n + 10n} = 0.91$$

3.2.4.1.6 Area Under the ROC Curve (AUC)

La curva ROC se define al graficas sobre los ejes coordenados TPR vs FPR las densidades de probabilidad para algunos puntos de corte, sacadas de las distribuciones de probabilidad para los sujetos negativos y positivos, las cuales se suponen distribuciones normales de media y desviación típica diferentes.

Los puntos de corte son sujetos seleccionados arbitrariamente, para los cuales se definen los valores de FPR y TPR como el área a la derecha del punto de corte para las distribuciones de probabilidad de sujetos negativos (izquierda) y positivos (derecha) respectivamente, como se muestra en la figura 7.

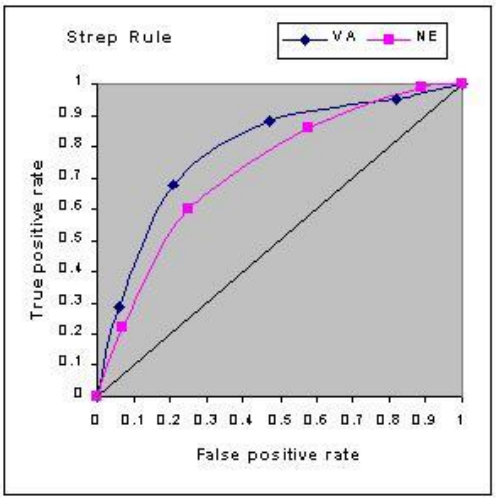
Figura 7. Distribuciones normales.



Fuente: http://www.fisterra.com/mbe/investiga/curvas_roc/curvas_roc.htm

En la Figura 7 se ven distribuciones con media y desviación estándar diferente, izquierda a derecha: distribuciones para sujetos negativos y positivos, con un punto de corte x. FPR es el área a la derecha, la más oscura, respecto al punto de corte. TPR es el área a la derecha, la más clara, respecto al punto de corte.

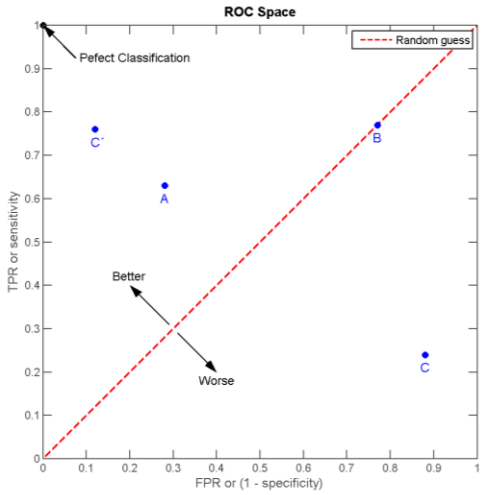
Figura 8. Curvas ROC construidas mediante un ajuste de una curva suave para cuatro puntos de corte.



Fuente: http://gim.unmc.edu/dxtests/roc3.htm Interpreting Diagnostic Tests, Thomas G. Tape, MD.

La métrica que proviene de la curva ROC es el Área bajo la curva ROC (AUC por sus siglas en inglés). Y al ser mayor su área la curva se acerca al punto (0,1), debido a que es allí donde se encuentra el máximo de TPR y el mínimo de FPR. En la figura 9, por encima de la línea punteada los modelos mejoran, por debajo empeoran y el valor óptimo se encuentra en (0,1).

Figura 9. Diferentes puntos de corte provenientes de conjuntos de entrada diversos



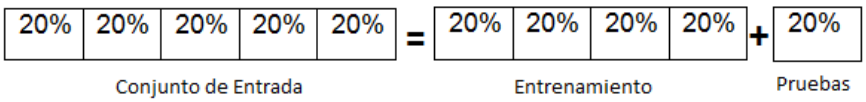
Fuente: http://commons.wikimedia.org/wiki/File:ROC_space-2.png

3.2.4.2 Cross Validation

La *cross validation* es una técnica que divide los datos en k *folds* (partes iguales) de los cuales usa $k-1$ como el conjunto de entrenamiento y deja el *fold* restante

para calcular la veracidad de las predicciones en el conjunto de entrenamiento (entrada) y el de pruebas. Este proceso se repite k veces, hasta que cada *fold* haya sido escogido como conjunto de pruebas, lo que arroja como resultado los promedios para las mediciones sobre los conjuntos de entrenamiento y pruebas, además de sus desviaciones estándar.

Figura 10. K folds. $K-1$ para entranamiento y 1 fold para pruebas.

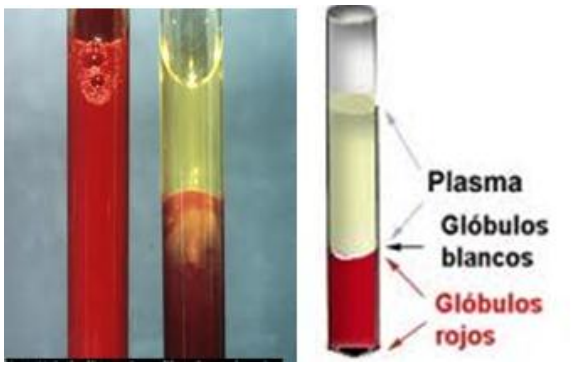


3.3 ZIMOGRAFIA

Una zimografía es la imagen resultante de aplicar la técnica de electroforesis en gel a muestras de células o tejido para visualizar la actividad de enzimas. La degradación del sustrato por una enzima se evidencia en formas de regiones de color más claras que el fondo. Las técnicas de electroforesis son comúnmente empleadas en estudios forenses, análisis clínico en perfiles de proteínas, análisis de paternidad e identificación de enfermedades genéticas como la anemia falciforme.

La técnica de electroforesis consiste en centrifugar muestras de sangre para obtener el plasma o suero; a continuación se prepara una mezcla de gelatina y plasma humano, se vierte en una matriz porosa a la cual se le aplica una fuerza electromotriz causando el desplazamiento de la muestra separando así sus componentes; posteriormente la matriz se incuba en un buffer apropiado que favorece la actividad de las enzimas, finalmente se aplica un tinte compatible con el gel que revela los resultados creando bandas verticales por cada paciente y espacios claros que según su tamaño y tonalidad representan la actividad de cada enzima en una muestra sanguínea.

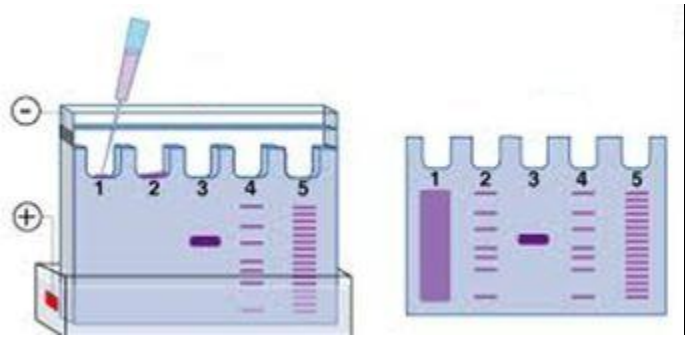
Figura 11. Plasma sanguíneo obtenido por centrifugación.



Fuente: Grupo M.I.N.E.N.

La preparación, procesamiento y conservación de las muestras de cada paciente son procesos largos y la variación de factores como el voltaje aplicado y el tiempo de incubación influyen directamente en la calidad y los patrones de la imagen final. Cada imagen resultante contiene hasta diez muestras de pacientes.

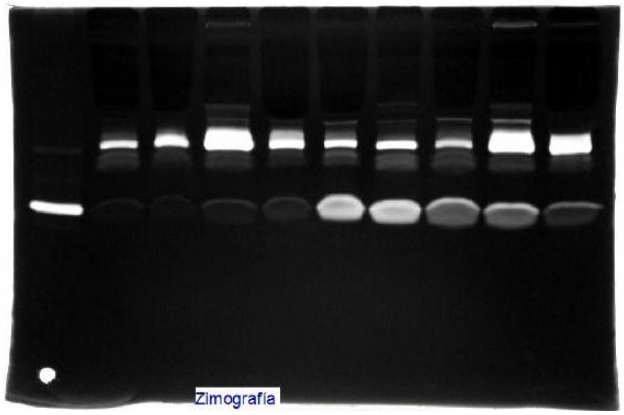
Figura 12. Voltaje aplicado en matriz de gel y zimografía obtenida.



Fuente: http://www.ehu.eus/biomoleculas/isotopos/blot.htm

En el estudio y detección de Sepsis la zimografía se usa para medir la actividad de algunas metaloproteinasas en la sangre siendo MMP2 y MMP9 las más frecuentes.

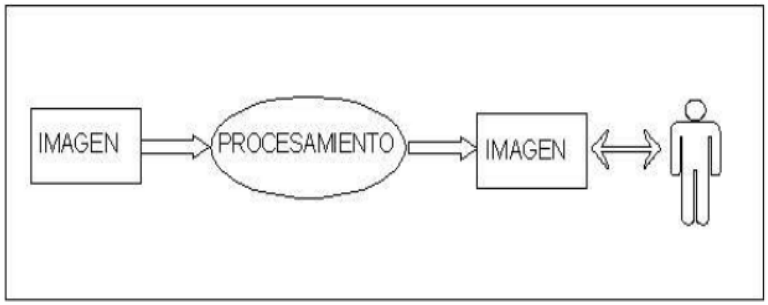
Figura 13. Imagen resultado de zimografía.



3.4 TRATAMIENTO DE IMAGENES DIGITALES

Una imagen puede ser definida como una matriz de tamaño m filas por n columnas, donde $f(x, y)$ es el valor de intensidad de cada píxel en la coordenada (x, y). En una imagen en escala de grises el valor de cada píxel estará en el rango [0,255] siendo 0 negro y 255 blanco. Tratamiento de imágenes es el conjunto de técnicas usadas para resaltar, mejorar, ignorar o describir las características de una imagen de interés.

Figura 14. Tratamiento de imágenes: Imagen de entrada a la que se le aplica un proceso para obtener otra que sirve para dar una mejor interpretación.



Fuente: SUCAR, GOMEZ. Visión Computacional.

En el tratamiento de imágenes siempre tenemos una imagen de entrada que es preprocesada para obtener detalles físicos de la imagen como discontinuidades, bordes, color, textura, gradiente y profundidad.

3.4.1 Filtrado de imágenes

La operación de filtrado consiste en aplicar una transformación T a una imagen de entrada f para obtener una imagen g con el fin de resaltar ciertas características de interés como los contornos o de lo contrario suavizarla para eliminar ruido.

$$g(x,y) = T[f(x,y)]$$

El filtrado de imágenes se puede realizar en el dominio de la frecuencia o el espacio. En el dominio del espacio mediante la convolución, teniendo en cuenta que una imagen es una matriz, se aplica un kernel o filtro sobre la vecindad de cada píxel alterando directamente el valor de la imagen. Los valores y el tamaño del kernel determinarán el resultado final sobre la imagen, por ejemplo:

$$K = \begin{matrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{matrix}$$

K es un kernel que aplica la media de la vecindad de los píxeles produciendo una imagen con bordes más suaves.

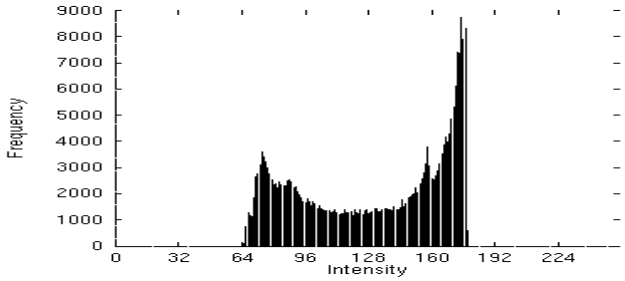
3.4.2 Histograma

El histograma de una imagen es la representación gráfica de la distribución de los valores de intensidad en los píxeles, dando la probabilidad de cada nivel de gris presente en la imagen de entrada. Si tenemos un nivel $0 \leq r_k \leq 255$ entonces:

$$P(r_k) = n_k/n$$

Donde n_k es el número de pixeles con el valor r_k y n es el número total de pixeles en la imagen. Los histogramas son útiles en el análisis inicial de la imagen ya que nos presenta de manera visual el contraste de una imagen, además de facilitar la umbralización.

Figura 15. Histograma de una imagen. Muestra la distribución de los niveles de gris en una imagen.



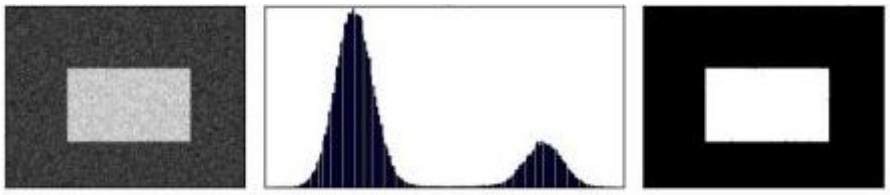
3.4.3 Umbralización

La umbralización consiste en transformar una imagen de escala de grises en una binaria, por lo tanto sólo hay dos posibles valores: blanco o negro. Esto se consigue comparando cada píxel con un umbral. Si es mayor que el umbral se le asigna un valor, de lo contrario se le asigna otro. Esta técnica es útil para construir máscaras para la separación de un objeto y su fondo.

Sea T un valor entre [0,255] f la imagen original y g la imagen binaria resultante podemos definir la umbralización de la siguiente manera:

$$Sif(x,y) > T \rightarrow g(x,y) = valor1$$
$$Sino \rightarrow g(x,y) = valor2$$

Figura 16. Ejemplo de Umbralización de Otsu. De izquierda a derecha una imagen con ruido, su histograma bimodal y la imagen después de umbralizar.



Fuente: Documentación de OpenCV - Python.

Existen diferentes métodos para determinar que umbral es óptimo a la hora de separar un objeto de su fondo basándose en el histograma, para imágenes cuyo histograma es bimodal suele utilizarse el método de Otsu, el cual encuentra el nivel entre los dos picos del histograma con varianza mínima en las dos clases.

Cuando un umbral parece no segmentar bien una sección de la imagen global se puede aplicar métodos de umbralización adaptativa, estos métodos subdividen la imagen en varias secciones y calculan el umbral óptimo para cada una. La umbralización adaptativa es computacionalmente más costosa que la global. Es muy útil a la hora de segmentar objetos en fondos no homogéneos.

3.4.4 Operaciones Morfológicas

Las operaciones morfológicas son operaciones sencillas que alteran la forma de una imagen binarizada. Las más usadas se explican a continuación.

3.4.4.1 Erosión

Se realiza una convolución de la imagen con un kernel que convierte en 1 el valor de un pixel si los pixeles en su vecindad no son 1, de lo contrario será 0. El efecto que produce es reducir bordes, eliminar objetos pequeños y desconectar regiones.

Figura 17. Erosión de una imagen.

3.4.4.2 Dilatación

Es la operación opuesta a la erosión, expande los píxeles blancos en una imagen causando el engrosamiento del contorno del objeto.

Figura 18. Dilatación.

3.4.4.3 Apertura

Consiste en aplicar la operación de erosión seguida de dilatación. Dando como resultado la remoción de ruido.

Figura 19. Operación de apertura. Imagen original e imagen resultado.

3.4.4.4 Cerradura

La operación opuesta a la apertura, se aplica primero dilatación seguido de erosión. Sirve para remover pequeños espacios dentro de un objeto.

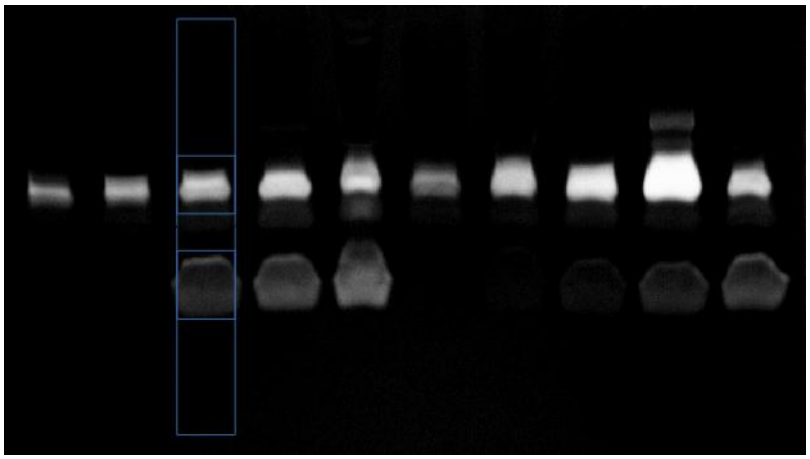
Figura 20. Operación cerradura. Imagen original e imagen resultado.

4. ESTADO DEL ARTE

4.1 PROCESAMIENTO DE ZIMOGRAFÍAS

Una zimografía de calidad tendrá un patrón de bandas verticales, cada una representando un paciente que puede presentar a lo largo de la imagen la presencia de una o más enzimas en la muestra del tejido. La identificación de la enzima se realiza por la altura en la que aparece ya que esta indica su peso molecular.

Figura 21. Zimografía. Ejemplo de cómo se realiza manualmente el procesamiento deslizando una banda a la vez para la segmentación.



Si bien la detección de bandas y enzimas es una tarea realizada manualmente señalando cada una de las bandas y a continuación la posición de las enzimas con ayuda de algún software, es una tarea que consume mucho tiempo, además de ser propensa a errores humanos, por lo cual en los últimos años se han desarrollado diversos métodos para la detección, segmentación y descripción de este tipo de imágenes.

Algunos de los primeros métodos de procesamiento semiautomáticos para la detección y segmentación de las bandas verticales tomaban como datos de entrada el centro de la primera y última banda seleccionado manualmente por el usuario junto con el número de bandas entre la primera y última banda, asumiendo que las bandas eran de un ancho constante.

Es común ver métodos automatizados como en [8] dónde utilizan la transformada de watershed para representar la imagen como una superficie topográfica de textura y forma variable permitiendo así la segmentación de la imagen en diferentes regiones. Kabouch [9] en cambio propone un algoritmo que elige un umbral de manera automática y ecualiza los valores de gris en la imagen original para eliminar el fondo de la imagen y así detectar los objetos (enzimas) presentes.

Ismail, Gh. S. Eltaweel y H. Nassar [10] implementaron un algoritmo con una tasa 99.5% de precisión en imágenes de electroforesis en gel de muestras de ADN de huellas digitales, normaliza el valor de los píxeles de la imagen, después aplica un filtro de desenfoque para aumentar los bordes sin incrementar el ruido de la imagen, las imágenes usadas para los experimentos de esta publicación presentaban un histograma bimodal, por medio del método de otsu se encuentra un umbral para la aplicar una máscara binarizada y así extraer el fondo. Una vez separados los objetos se detectan las bandas verticales basándose en varios supuestos entre estos que las actividades de enzimas en una misma banda tienen forma cóncava y son similares.

Sin embargo en la práctica las zimografías muestran variaciones en la forma y tamaño de la actividad de una enzima haciendo los métodos anteriores susceptibles a fallos, ya que las bandas entre cada paciente no son necesariamente del mismo ancho.

4.2. MACHINE LEARNING APLICADO A ANÁLISIS DE DATOS CLÍNICOS

Desde que los computadores se empezaron a usar en hospitales y centros de investigación de salud han traído como consecuencia cambios en la manera de capturar, almacenar e interpretar datos clínicos. Los datos obtenidos suelen ser de gran tamaño, diferentes formatos y su complejidad parece incrementarse con el paso del tiempo gracias al desarrollo de la ingeniería del software y las plataformas web, ejemplos de estos datos son genomas, resultados de exámenes médicos, notas de consultas, mediciones hechas con instrumentos, etc.

Machine learning, al igual que las redes neuronales artificiales, es una rama de la inteligencia artificial que nació del deseo de hacer máquinas que aprendieran a partir de datos suministrados, su objetivo es identificar patrones en datos de alta complejidad. Con la cantidad de datos recolectados en ambientes médicos se ha estimulado el uso de *machine learning* para extraer información que pueda ahorrar costos y salvar vidas. Uno de los usos más comunes de técnicas de aprendizaje

de máquina es el soporte computacional en el diagnóstico de enfermedades como el cáncer de seno [11] permitiendo tener más seguridad al dar un segundo criterio a los expertos.

La popularidad de *machine learning* crece no sólo en la informática de la salud sino en todas las industrias y esto se ve reflejado en el aumento de conjuntos de datos disponibles para el público en portales como Kaggle, con una comunidad abierta que compite y comparte nuevo conocimiento.

5. METODOLOGÍA

Este proyecto se desarrolló guiado por los principios de las metodologías ágiles de desarrollo iterativo e incremental, ya que son adecuadas para equipos interdisciplinarios que colaboran entre sí y en nuestro caso trabajar de la mano de los profesionales de la salud usuarios de esta herramienta fue importante para comprender sus necesidades.

El desarrollo ágil permite tener una visión clara del avance de la solución, se realiza una lista de items a realizar y se dividen en un conjunto de paquetes o etapas, cada paquete cuenta con determinadas actividades a realizar por el equipo que conllevan a las entregas de una parte funcional del producto, estas son evaluadas y de ser necesario, corregidas en la siguiente iteración, siguiendo el cronograma en el plan.

PT: Paquete de trabajo. **A**: Actividad. **E:** Entregable

PT1: Análisis de Zimografías

A1.2: Procesos de extracción automática del nivel enzimático: cuantificar los niveles de enzimas mediante aplicación de técnicas de procesamiento de imágenes para la automatización de la medición de sustancias en la zimografía.

E1.1: Script en Python para la extracción del nivel enzimático desde la zimografía
E1.2: Descripción y justificación del desempeño del proceso de extracción

PT2: Predicción de estado de SEPSIS

A2.1: Preprocesado de datos (incluyendo la integración de los niveles enzimáticos predichos desde las zimografías), Identificar datos útiles, descartar datos incompletos o erróneos.
A2.2: *Machine learning* para la predicción de la probabilidad de supervivencia
A2.3: *Machine learning* para la clasificación de pacientes según la severidad de la enfermedad

E2.1: Módulo software para el preprocesado de datos
E2.2: Módulo software para la predicción y clasificación de pacientes
E2.3: Framework para el entrenamiento y análisis con nuevos datos

PT3: Integración en herramientas médicas
A3.1: Análisis y definición de la implementación en herramientas existentes
A3.2: Implementación de un módulo web para la ejecución sobre pacientes nuevos de E1.1, E2.1 y E2.2

E3.1: Documentación de diseño
E3.2: Web App.

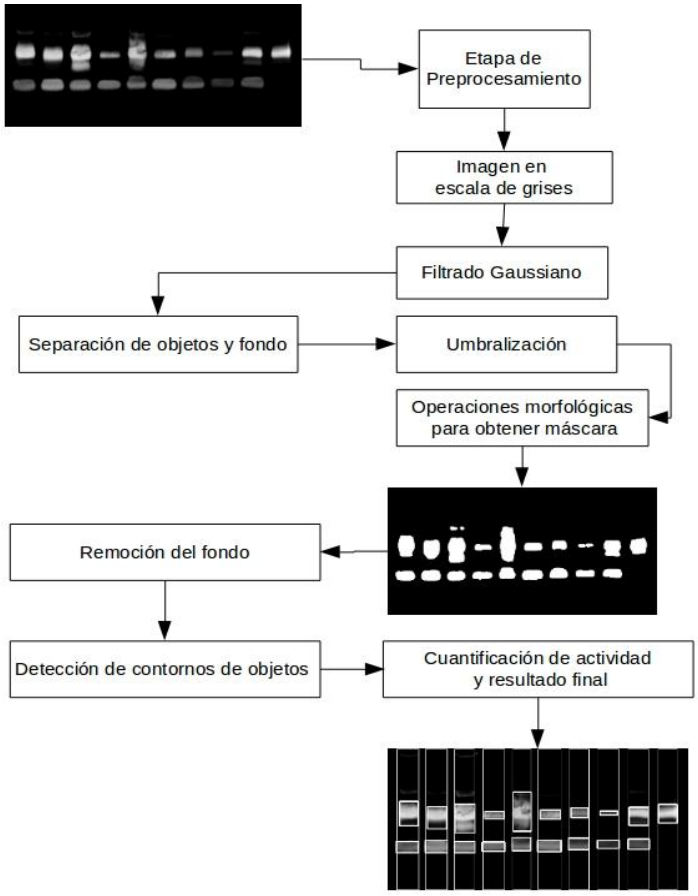
6. DESARROLLO DEL PROYECTO

En este capítulo los autores presentan una descripción de cómo se diseñaron e implementaron los módulos de predicciones y la herramienta de procesamiento de imágenes de zimografías utilizando una metodología de desarrollo ágil.

6.1 ETAPA 1: ALGORITMO PARA LA SEGMENTACIÓN Y MEDICIÓN DE LAS METALOPROTEINASAS EN ZIMOGRAFÍAS

Después de hacer una revisión a fondo del estado del arte del procesamiento de imágenes de geles obtenidas por técnicas de electroforesis, se diseñó una serie de pasos para segmentar y cuantificar la actividad de enzimas en una zimografía. El lenguaje empleado para implementar el script final fue en Python utilizando la librería OpenCV.

Figura 22. Pasos en el procesamiento de zimografía



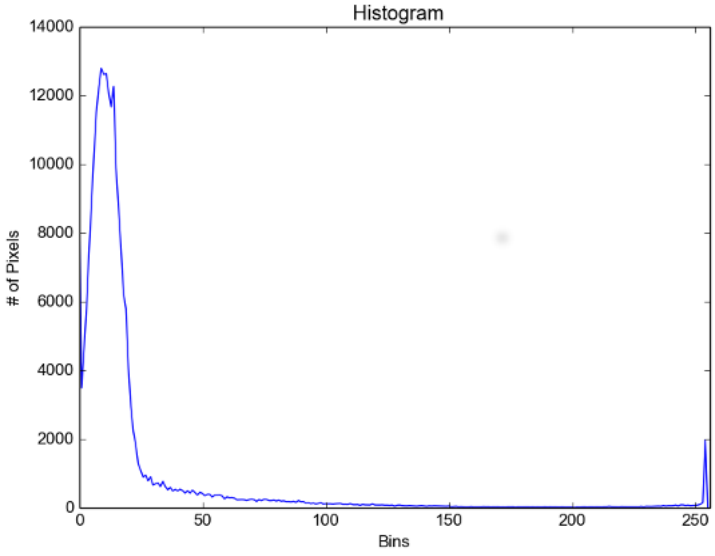
6.1.1 Preprocesamiento

El primer paso es cambiar la imagen original de color a escala de grises y a continuación pasar un filtro de desenfoque gaussiano para eliminar ruido presente. Aunque a simple vista no se note una diferencia drástica por el bajo contraste de la imagen, es un paso obligatorio de lo contrario se detectarán objetos que no corresponden a actividades enzimáticas en la etapa de segmentación. El valor de cada píxel es el resultado de promediar con distintos pesos la intensidad de los píxeles vecinos. Este tipo de filtro reduce especialmente el ruido producido por diferencias de ganancias del sensor, ruido en la digitalización, etc.

6.1.2 Segmentación

Para una umbralización óptima se estudió el histograma de las zimografías hechas en el grupo de investigación M.I.N.E.N y se encontró que la mayoría de los píxeles son opacos lo cual resulta en una distribución unimodal, contrario a la mayoría de las publicaciones de este tipo de imágenes que suelen mostrar histogramas bimodales y encuentran un umbral óptimo por medio del método de otsu. La detección del umbral óptimo que separa las actividades de las enzimas del fondo se encontró calculando el nivel de intensidad donde se concentra un cambio drástico de la distribución de pixeles.

Figura 23. Histograma de zimografías hechas en el grupo M.I.N.E.N



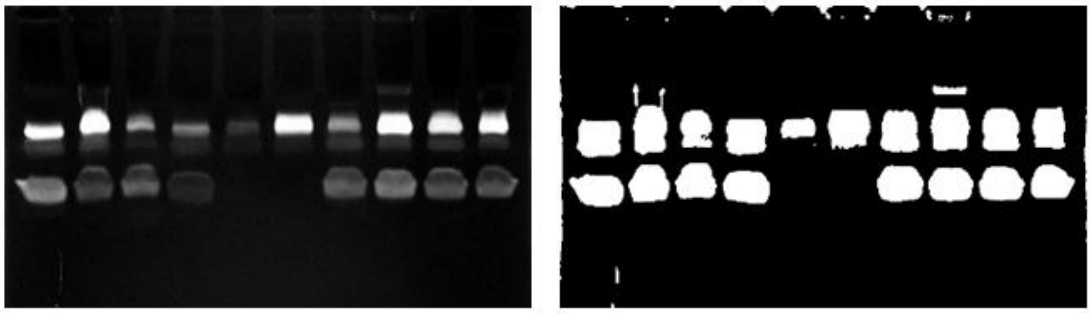
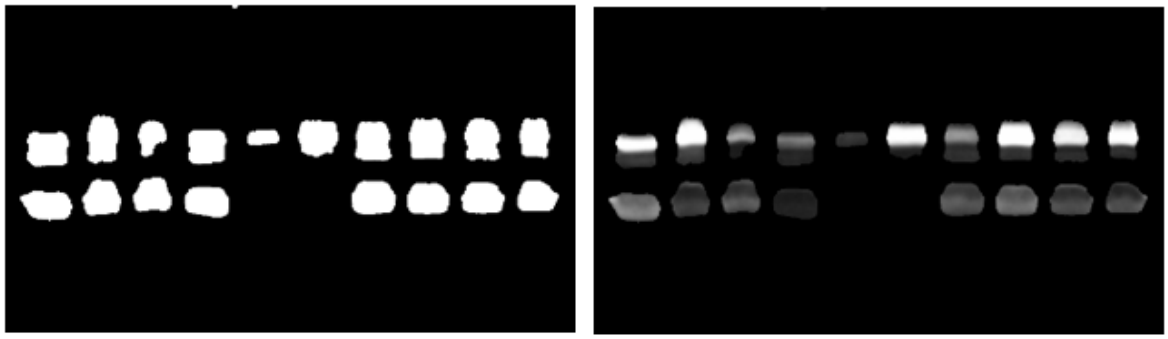


Figura 24. Imagen original y después de la Umbralización.

Después de la umbralización pueden quedar objetos pequeños que no nos interesan, además de columnas juntas por lo que es necesario realizar operaciones morfológicas para mejorar la segmentación de la imagen. Al aplicar una dilatación a la imagen binaria seguido de la operación de apertura tendremos la máscara deseada.

Figura 25. Máscara final y extracción del fondo.



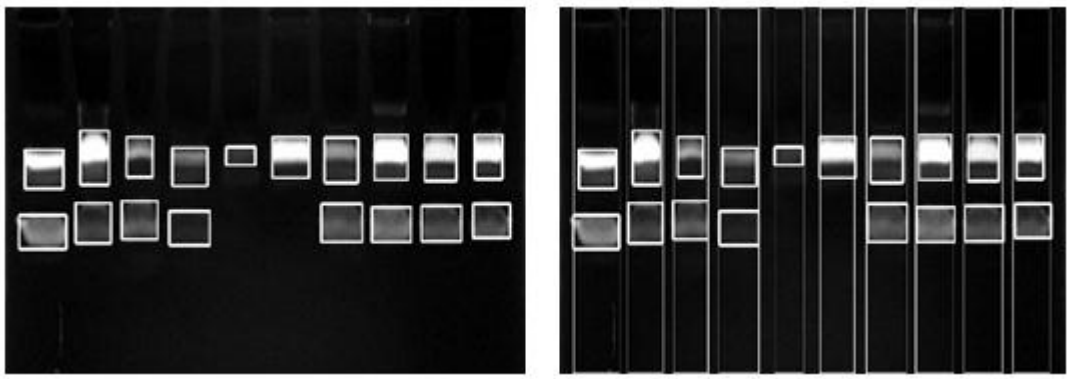
6.1.3 Cuantificación de la Actividad Enzimática

Finalmente tenemos los objetos deseados separados del fondo de la zimografía. Para cuantificar la actividad rodeamos cada objeto con el rectángulo de menor área posible, para la cuantificación se tuvo en cuenta que en las zimografías cada marca de actividad es muy variable tanto en tamaño como en forma por lo cual se decidió utilizar un método sencillo que tenga en cuenta el tamaño y el color del área para cuantificar su actividad.

$$Actividad = \acute{x} * (base + altura)$$

Donde $\acute{x}$ es el resultado de promediar el valor de los pixeles dentro del área del rectángulo. De esta manera se obtiene una medida representativa.

Figura 26. Resultado final, detección de enzimas y segmentación de bandas verticales.



Para asociar cada medida a un paciente se usa la coordenada en el eje x del centro de los rectángulos y se agrupan los de coordenada similar, dibujando así el rectángulo que encierra las enzimas pertenecientes a un mismo paciente.

Los resultados de la cuantificación son validados a través de la interfaz descrita posteriormente en la sección 6.4.

6.2. ETAPA 2: MODELO PREDICTIVO PARA EL ESTADO DE LA ENFERMEDAD SEPSIS Y SU MORTALIDAD

El modelo predictivo se construyó por etapas en las cuales se tuvieron en cuenta los pacientes contenidos en las hojas de cálculo: “Columnas-Base de datos G-Sepsis y MMACDYS UNAB” primeramente y luego “Base de datos MATRIX Definitiva 31 Julio 2014v4” (Depurada), entregadas por el grupo MINEN, cada una con un diccionario que explicaba brevemente la información que se obtuvo de cada paciente y la escala de medición para cada variable.

6.2.1 El Conjunto de Entrada

La selección del conjunto de entrada para la primera fase se hizo siguiendo tres criterios: eliminar las columnas (características) y los pacientes (filas) con muchas celdas vacías con el objetivo de conservar la mayor cantidad de filas y columnas posibles sin espacios en blanco, además se eliminaron las columnas que representaban una relación entre otras características del paciente, como las escalas SOFA, Glasgow, entre otras presentes en la hoja de cálculo.

Si una columna tenía en promedio 140 espacios en blanco o más, se consideraba no tenerla en cuenta para usarla dentro del conjunto de entrada, igualmente si habían pacientes (filas) con gran cantidad de espacios vacíos se eliminaban. Al final la base de datos "Columnas-Base de datos G-Sepsis y MMACDYS UNAB" se redujo a un total de 100 características por cada uno de los 529 pacientes restantes. Este conjunto X con $dim(X) = (529,100)$ junto con la clasificación de la severidad de la sepsis Y con $dim(Y) = (529,1)$, que especialistas le asignaron a los pacientes sépticos, conforman el conjunto de entrada para la primera etapa de análisis de datos (6.2.2). Si por el contrario el conjunto Y correspondía al estado vital de los pacientes al egreso de la institución, entonces el conjunto de entrada era diferente, pero solo en sus etiquetas Y, sus dimensiones se conservaban.

Para los clasificadores Binarios (6.2.4) y etapas subsecuentes se seleccionaron columnas y pacientes usando los mismos tres criterios anteriores, pero en vez de eliminar los espacios en blanco se buscó reducirlos de la hoja de cálculo: "Base de datos MATRIX Definitiva 31 Julio 2014v4", de tal forma que se pudieran conservar la mayor cantidad de pacientes (filas) y características posibles (columnas). El resultado fue con conjunto de datos X con dimensiones $dim(X) = (563,107)$ y Y con $dim(Y) = (563,1)$, donde las etiquetas Y definían si el conjunto de entrada involucraba el estado vital al egreso de la institución o la severidad de la sepsis en los pacientes.

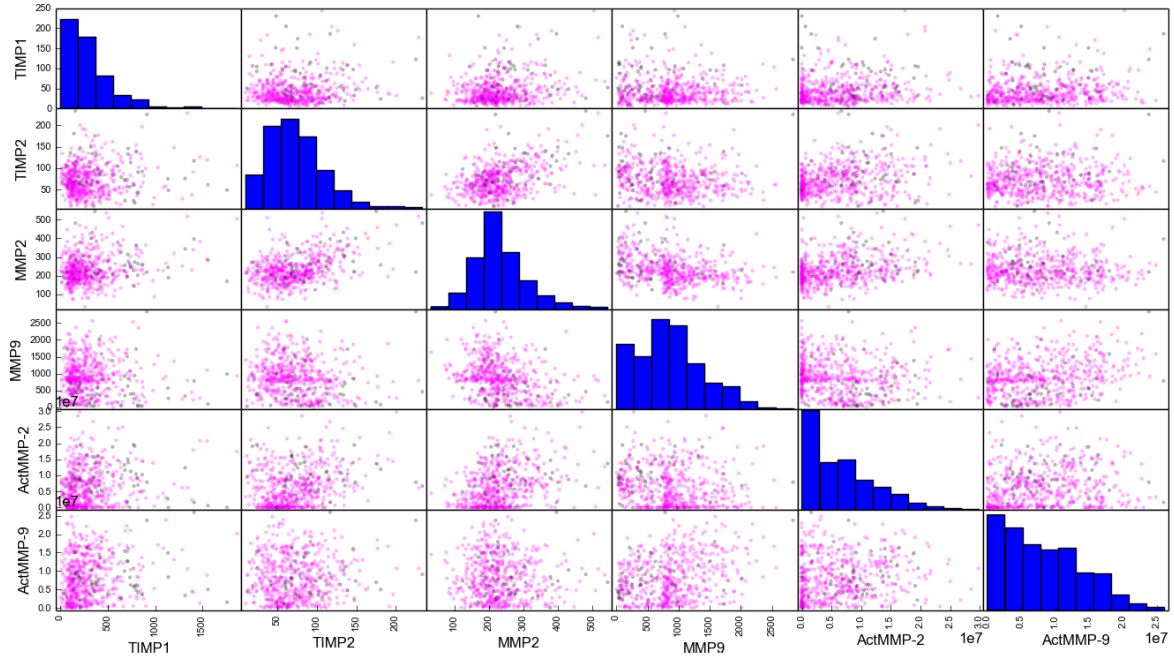
Es importante resaltar que los modelos se crearon dependiendo del objetivo, si era éste para predecir el estadio de la enfermedad o su mortalidad.

6.2.2 Análisis de Datos

La tupla (X, Y) es el caso de estudio en esta fase, el objetivo fue identificar las propiedades de las características, sus rangos de valores, su relación unas con

otras y las etiquetas. En la figura a continuación se seleccionan las columnas TIMP1, TIMP2, MMP-2, MMP-9, ActMMP-2 y ActMMP-9, las primeras cuatro son relativas a la cantidad de metaloproteinasa encontrada en una muestra tomada del paciente, TIMP-1 y TIMP-2 son los inhibidor tisular de las metaloproteasas MMP-2 y MMP-9, las dos últimas hacen parte de la medición de la actividad enzimática sobre las zimografías usando ImageJ. Los puntos color magenta son pacientes que registraron un estado vital vivo al egreso de la institución, mientras que los puntos negros registraron el estado contrario.

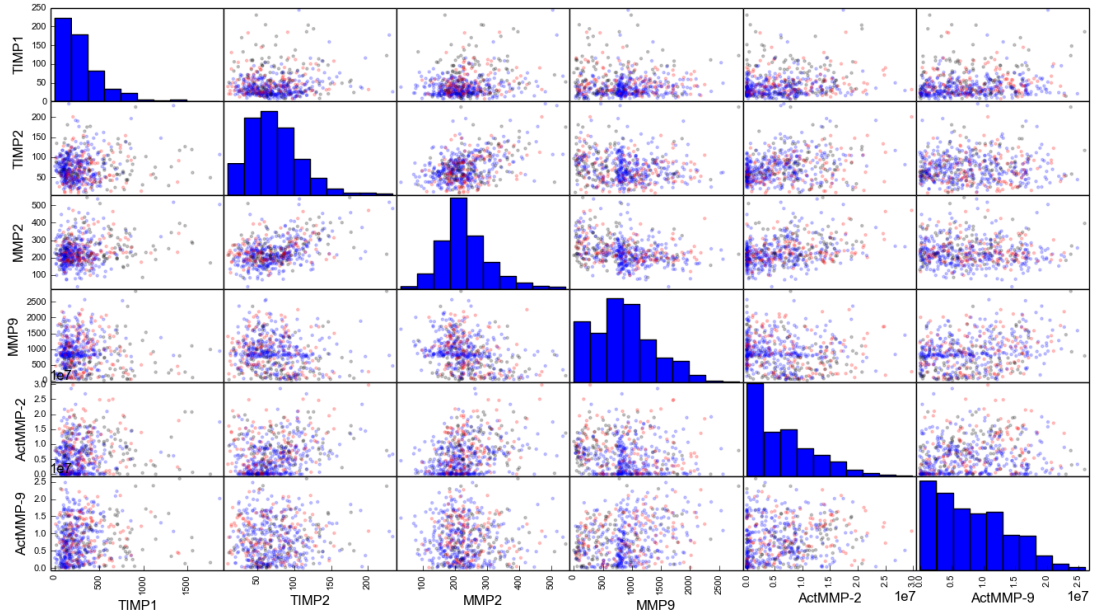
Figura 27. Matriz de distribución para el estado vital del paciente al egreso de la institución.



De la figura 27 se pueden sacar tres conclusiones, la primera es que el rango de las variables difiere en porciones de hasta 1/1200 para el caso de la ActMMP-9 respecto a MMP-9, es decir actividad medida respecto a la cantidad de la misma enzima. La segunda es que los datos no contiene una relación aparente entre ellos, además se puede observas que para algunas variables, la mayoría de pacientes registran un valor encontrado al comienzo del histograma de frecuencias, como TIMP1 y ActMMP-2, una razón para esto puede ser la tercera conclusión, la proporción de pacientes que murieron es mucho menor que la de los pacientes que vivían al salir de la institución.

La figura 28 es homóloga a la 27 en el sentido que representa también una matriz de distribución de los mismos datos, pero respecto a la severidad de la sepsis. Los puntos azules son para los pacientes con sepsis no grave, los rojos para paciente con sepsis severa y los negros para choque séptico. Una vez más es notorio que hay menos pacientes de algún tipo (clase) en específico, como es el caso de los pacientes que sufrieron de choque séptico.

Figura 28. Matriz de distribución para la severidad de la sepsis



En resumen, el análisis de datos indicó características relevantes de los conjuntos de entrada, como el desbalanceo entre los diferentes casos registrados de sepsis o la mortalidad en los pacientes, la alta variabilidad entre los rangos de las variables y la evidencia de que se necesitarían más características para construir los modelos.

6.2.2.1 Reducción de Dimensionalidad

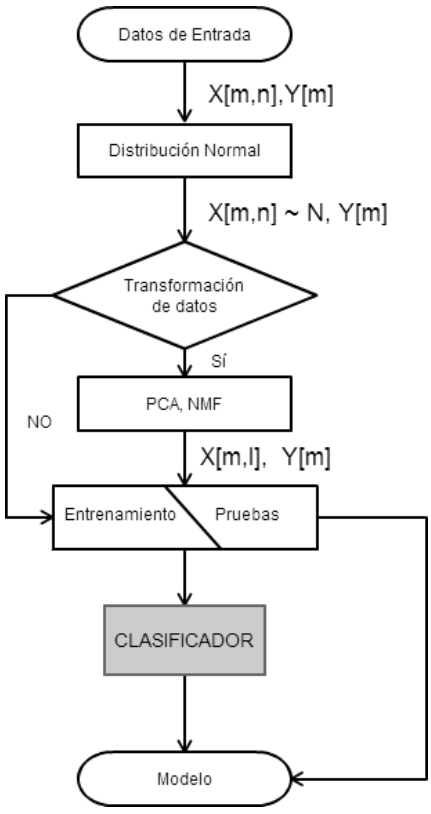
Si bien incluir más características es importante, se disponen también de herramientas para disminuir la dimensionalidad de los datos conservando cierto grado de información que contenían inicialmente. Algunos algoritmos que permiten hacer esto son *Singular Value Decomposition (SVD), Non-Negative Matrix Factorization (NMF)* e *Independent Component Analisys (ICA).*

6.2.3 Los Clasificadores Binarios y Multiclase

Un clasificador multiclase es aquel que me permite crear modelos que predicen sobre etiquetas no binarias, como el caso de la severidad de la sepsis donde tenemos tres posibilidades: sepsis no grave, severa y choque séptico. Los modelos para predecir la mortalidad en la sepsis serán binarios, pero se construirán a partir de los mismos clasificadores multiclase que nos provee el paquete de herramientas para la minería y análisis de datos scikit-learn, en python.

Los clasificadores se entrenarán a partir de datos preprocesados que seguirán el esquema de flujo de datos de la figura 29. Generando un modelo que tiene como entrada datos tratados y salida etiquetas binarias o multiclase.

Figura 29. Flujo de datos de entrada para entrenar los clasificadores del scikit-learn



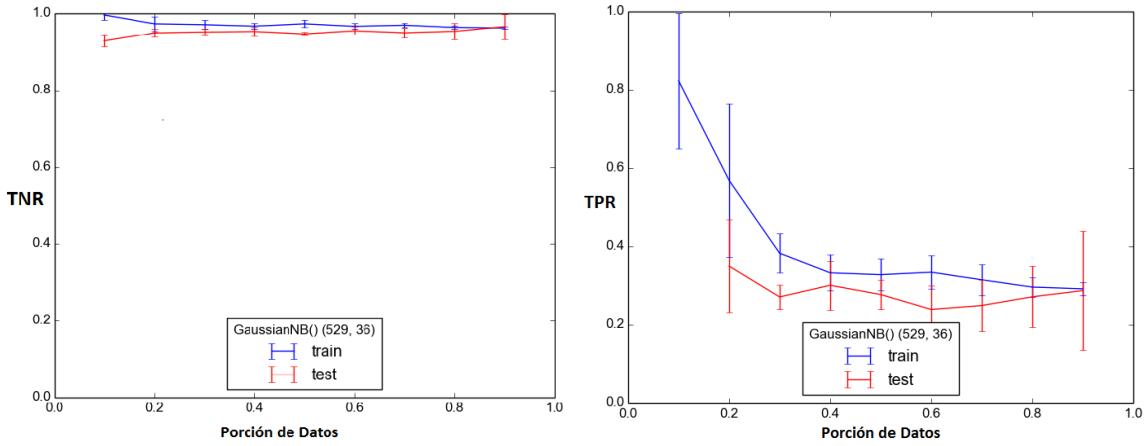
Los clasificadores usados en esta etapa fueron: K Nearest Neighbours, Decission Tree Classifier, Gaussian Naive Bayes y Support Vector Classifier. El desempeño se midió mediante las curvas de aprendizaje o *learning curves* que realizan un muestreo aleatorio sin repetición del conjunto de datos de entrada, de forma ascendente, como el 20%, 40%, 80% y 100% de los pacientes, con cada muestra se entrena el clasificador, es decir se construye el modelo y a partir de éste se evalúa su desempeño bajo las métricas mencionadas en el marco teórico. Las medidas serán tomadas varias veces bajo muestras diferentes, pero del mismo tamaño sobre el conjunto de entrenamiento y sobre el complemento del conjunto de entrenamiento, llamado conjunto de pruebas o *test*. Como se ve en la figura 30, la abscisa es la porción de datos de entrenamiento y la ordenada el valor de su métrica para el clasificador *Gaussian Naive Bayes* (GaussianNB), en la curva hay pequeñas líneas que la intersectan, éstas representan la desviación estándar de las diferentes mediciones de TNR, entonces los puntos que conforman la curva son la media y no una medición en particular.

6.2.3.1 Modelos Binarios para Mortalidad

Las clases o etiquetas para mortalidad corresponden a 0 y 1 para los pacientes vivos y que fallecieron respectivamente. Con el fin de encontrar la mejor forma de medir el desempeño de los clasificadores, se usan las métricas de proporción de casos negativos clasificadas correctamente TNR, su homóloga para casos positivos TPR y la métrica de exactitud o *accuracy*, para casos de ambas clases clasificados correctamente.

GaussianNB entrenado con datos cuyas características fueron escaladas a una distribución normal y posteriormente se les realizó una reducción de dimensionalidad por descomposición de sus valores singular (SVD por sus siglas en inglés), presenta las curvas de aprendizaje de la siguiente figura.

Figura 30. Curvas de aprendizajes TNR y TPR del GaussianNB binario para mortalidad

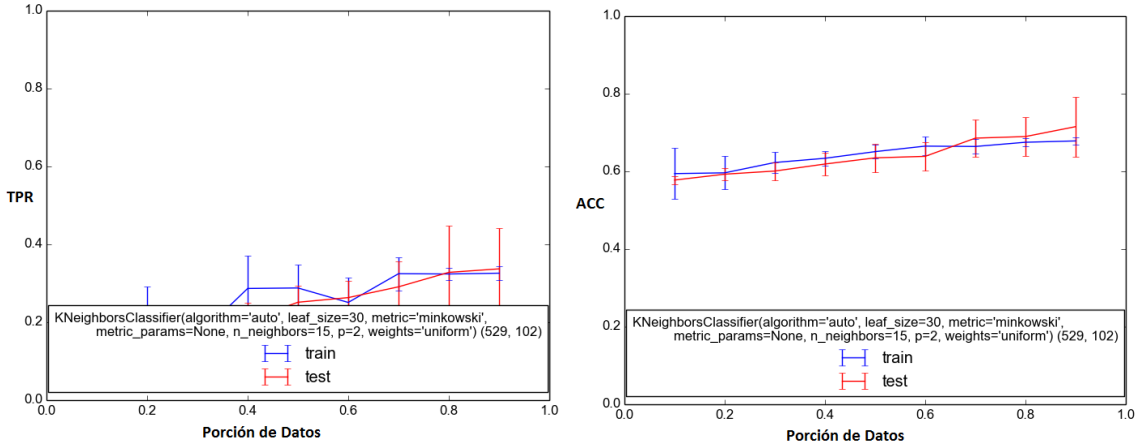


La curva para TNR evidencia que el clasificador está en capacidad de etiquetar correctamente los pacientes que vivían al egreso de la institución. Por el contrario TPR nos indica que su clasificación no es precisa, porque conforme se incluyen más pacientes en la creación del modelo, se clasifican de forma incorrecta más pacientes que fallecieron al egreso de la institución. En conclusión GaussianNB y otros clasificadores con transformaciones de datos similares, no eran representativos para la clasificación, dado que su precisión se basa en clasificar bien la clase con mayor proporción, pacientes vivos, pero el desempeño en la clase con menos observaciones es pobre.

6.2.3.2 Modelos Multiclase para Severidad de Sepsis

Similar a los clasificadores binarios, los clasificadores multiclase en la severidad presentaron un desempeño en su métrica de precisión que representaba algo más que una predicción aleatoria, pero en su TPR para los pacientes con sepsis grave el desempeño tiende a ser pobre, como es el caso del clasificador K vecinos, entre otros, ilustrado en la figura 31.

Figura 31. Curvas de aprendizaje ACC y TPR de K-Neighbours binario para severidad en sepsis.



6.2.4 Los Clasificadores Binarios para Severidad de la Sepsis

Una estrategia para mejorar la predicción de modelos multiclase es el método *One vs. The rest* debido a que se formulan tres nuevos problemas de clasificación, uno por cada clase existente:

- El problema cero equivale a clasificar la clase cero (nueva clase uno) contra la clase uno y dos (nueva clase cero).
- El problema uno equivale a clasificar la clase uno contra la clase cero y dos (nueva clase cero)
- Y el problema dos busca clasificar la clase dos (nueva clase uno) contra la clase cero y uno (nueva clase cero)

Además, los clasificadores binarios se construyen primeramente usando el modelo de flujo de datos especificado en la figura 29 con las diferencias: se rellenan los vacíos con la media de las características que sí registran medición, las características se ajustan independientemente a una distribución normal estándar y las transformaciones pueden variar entre PCA, NMF e ICA. El primer flujo de datos proveía de facilidad en la implementación de los clasificadores, pero inducía un *bias* al sacar parámetros del conjunto de entrada, porque incluía el conjunto de pruebas, sesgando así la medida de desempeño sobre este. Por esto, se planteó un nuevo flujo de datos, figura 32, para aquellos clasificadores que tuvieran buen desempeño con el *bias* inducido.

El Árbol de decisión mostró un buen desempeño siguiendo el primer flujo de datos con *bias*, por eso se construyeron *ensembles* con el flujo de datos sin *bias* como alternativas para los clasificadores binarios, en la figura 33 se muestra el desempeño del árbol de decisión con TPR en (a), (b) y (c) para los problemas 0, 1 y 2 respectivamente, de forma equivalente en la figura 34 se muestra el desempeño de TNR. Mientras que clasificadores como los k vecinos más cercanos arrojaron resultados que evidencian una predicción aleatoria, además no todas las transformaciones espaciales convenían, como ICA que presentó problemas para converger con el modelo de la figura 32.

Los hiperparámetros adecuados para los clasificadores sin *bias* fueron hallados haciendo una búsqueda en grilla sobre los clasificadores con bias.

Figura 32. Flujo de datos sin bias

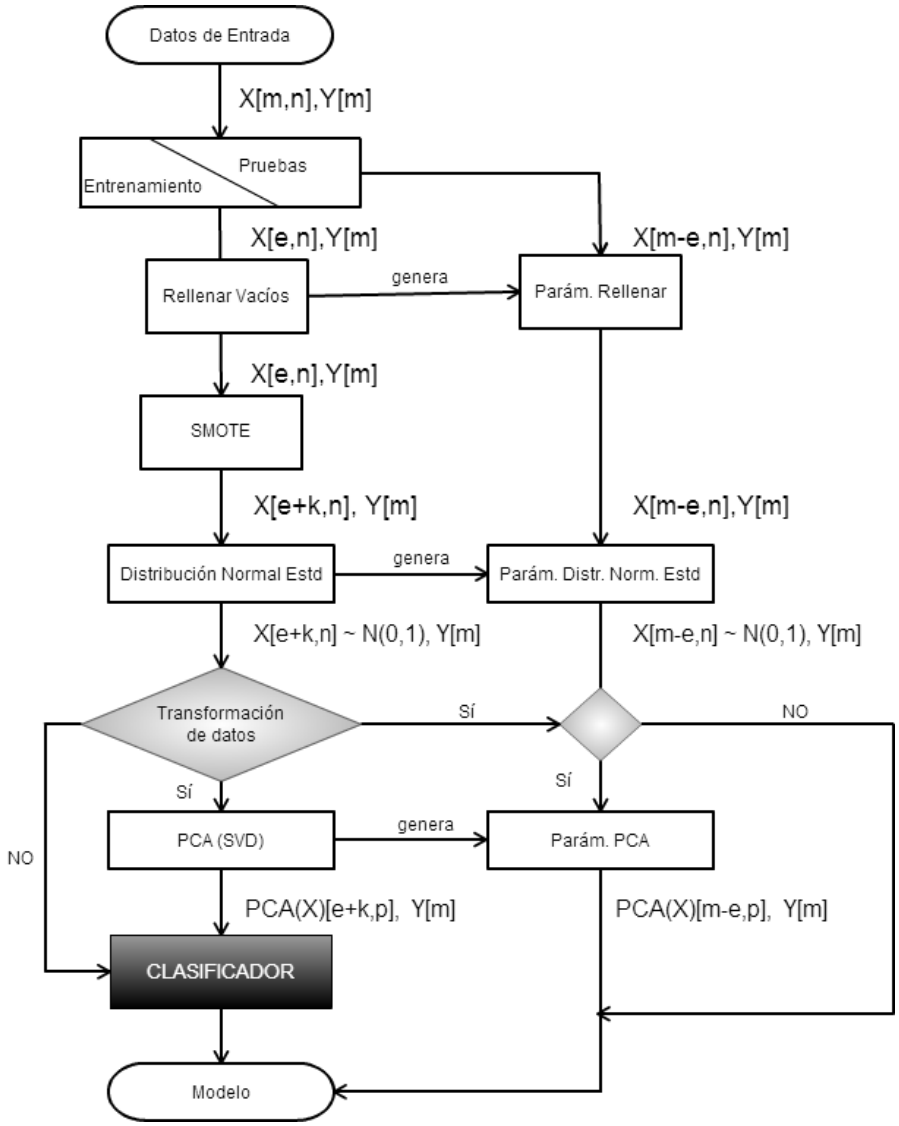
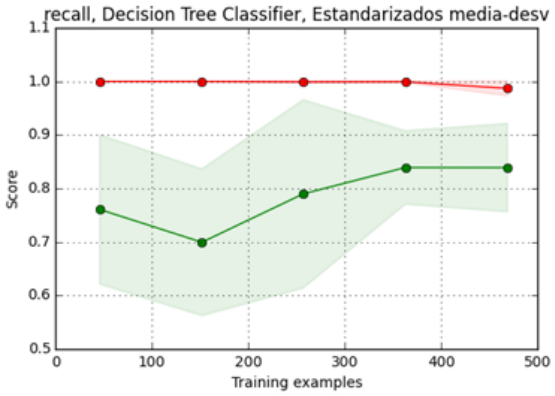
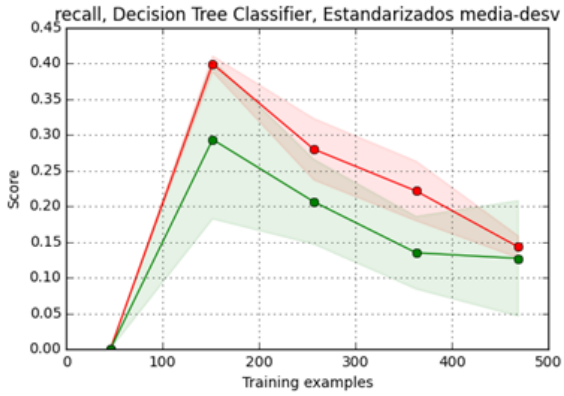


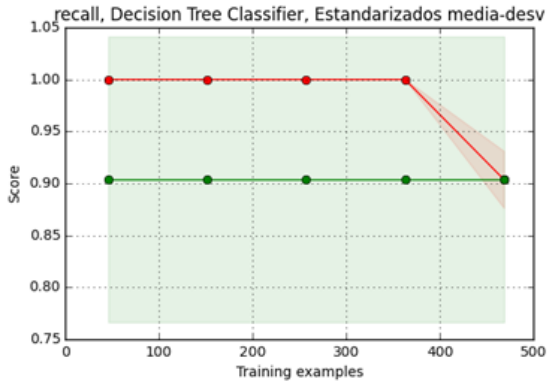
Figura 33. TPR problema 0, 1 y 2 del Árbol de Decisión para severidad Sepsis.


recall, Decision Tree Classifier, Estandarizados media-desv
Score
Training examples
1.1
1.0
0.9
0.8
0.7
0.6
0.5
0
100
200
300
400
500

(a)

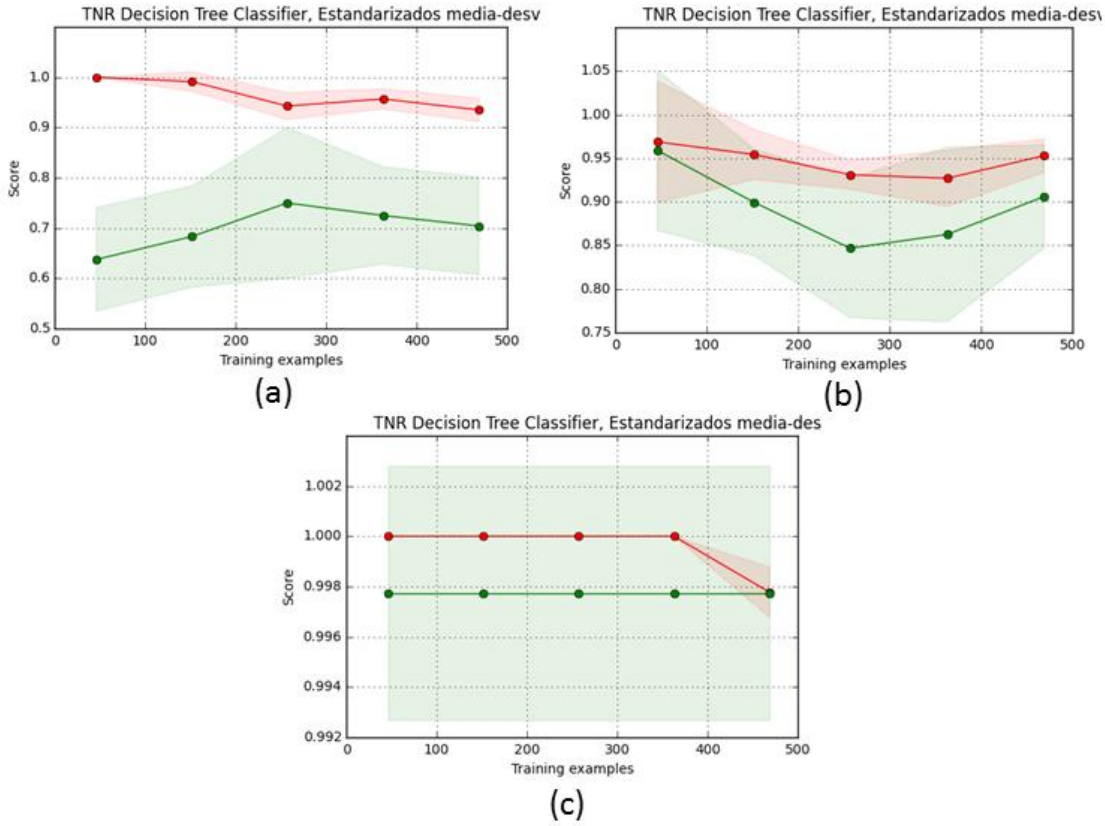

recall, Decision Tree Classifier, Estandarizados media-desv
Score
Training examples
0.45
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
0
100
200
300
400
500

(b)


recall, Decision Tree Classifier, Estandarizados media-desv
Score
Training examples
1.05
1.00
0.95
0.90
0.85
0.80
0.75
0
100
200
300
400
500

(c)

Figura 34. TNR problema 0, 1 y 2 del Árbol de Decisión para severidad Sepsis.



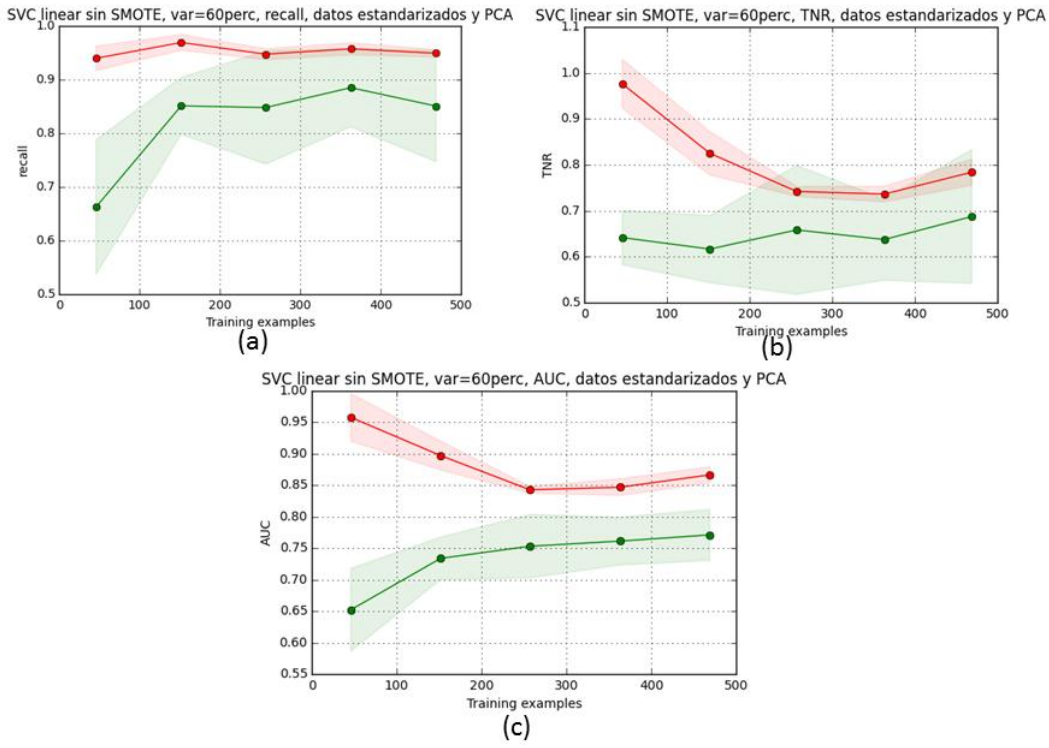
Para cada problema se seleccionaron los clasificadores binarios de *Adaboost* con Árboles de decisión como estimadores base, *Random Forest Classifier* y *Support Vector Classifier linear* con PCA, algunos de ellos incluyeron un sobre muestreo de aquellas clases con menos observaciones en comparación a las otras, se hizo mediante la técnica de Synthetic Minority Over-Sampling Technique (SMOTE)[12], porque un conjunto de entrada desbalanceado tiende a crear modelos que favorecen a la clase con más observaciones como se demuestra en [13], además del requisito de balancear el conjunto de datos para poder usar el *Random Forest Classifier* por las razones explicadas en [14].

El resultado de esta etapa fueron clasificadores binarios personalizados que permitieron ajustar los modelos a los datos de forma eficiente, debido a que el rellenado de vacíos, el escalado y la transformación de datos se realizan dentro de la función de entrenamiento del clasificador.

6.2.4.1 Problema 0

En la figura 35 se presentan los resultados para predecir los pacientes con sepsis no grave(clase 1) de los pacientes con sepsis severa y choque séptico (clase 0) usando el clasificador de soporte vectorial con kernel lineal, C = 1.7 y con características ajustadas independientemente a una distribución normal estándar conservando el 60% de la variabilidad de los datos dentro de los componentes PCA .

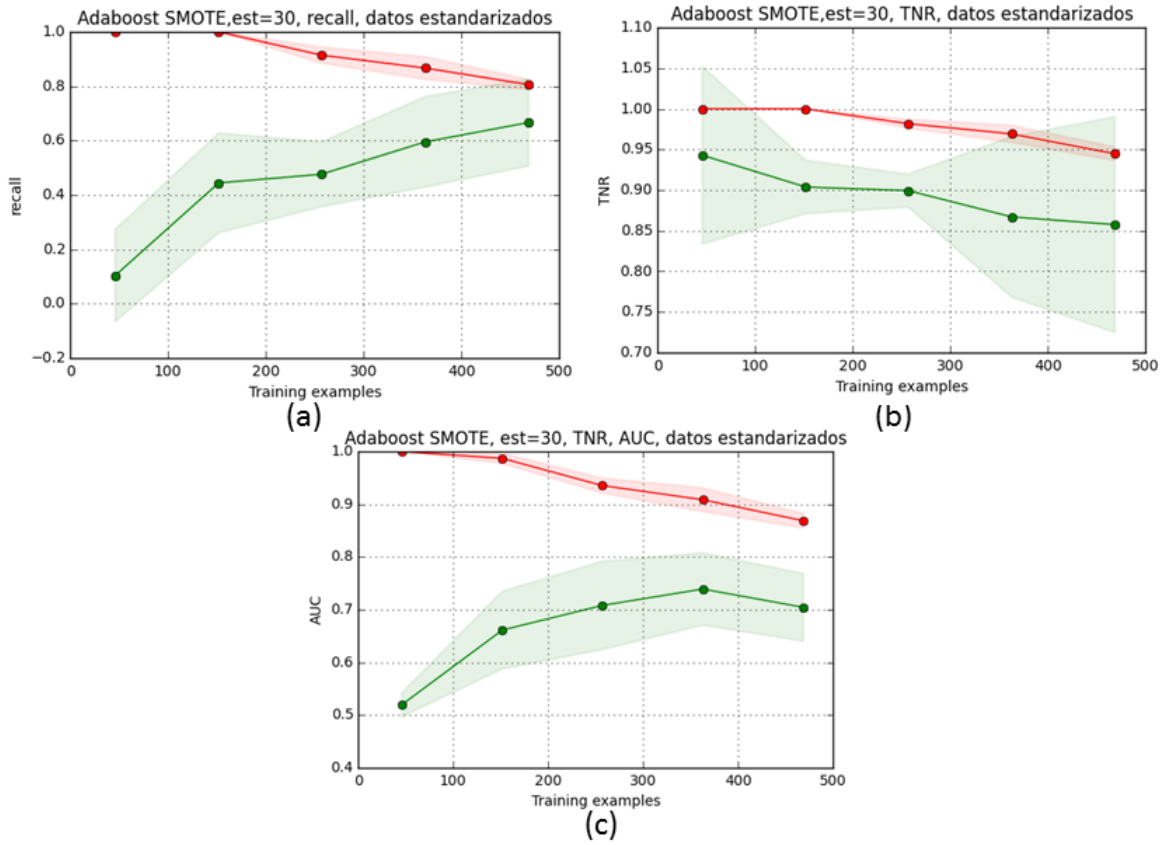
Figura 35. TPR, TNR y AUC del clasificador personalizado SVC linear



6.2.4.2 Problema 1

Los resultados del clasificador personalizado *Adaboost* se exponen en la figura 36, sus parámetros fueron 30 estimadores base de árboles de decisión, con un sobre muestreo SMOTE del 200% para la parte desbalanceada, en este caso la clase 1 (sepsis severa), es decir con 300% más datos de la clase 1.

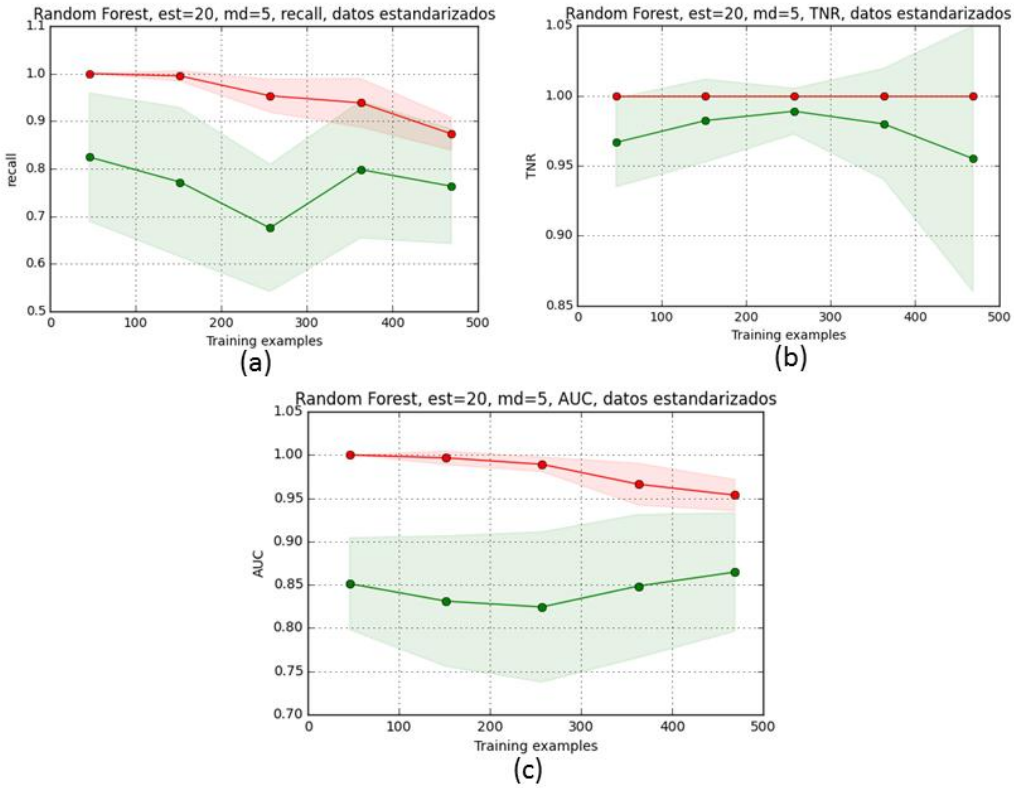
Figura 36. TPR, TNR y AUC del clasificador personalizado Adaboost (SMOTE = 200% con k=2).



6.2.4.3 Problema 2

Los resultados para el problema 2 del clasificador personalizado *Random Forest* se exponen en la figura 37, sus parámetro corresponden a 20 weak learners, una profundidad máxima de 5 y un sobre muestreo del 400% teniendo en cuenta 4 vecinos.

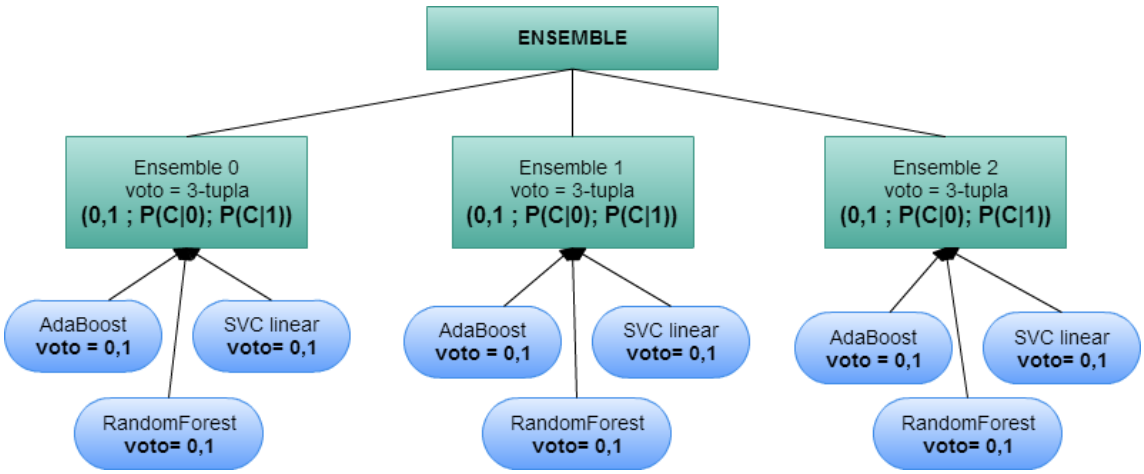
Figura 37. TPR, TNR y AUC del clasificador personalizado Random Forest (SMOTE=400% k=4).



6.2.5 El Modelo Predictivo: Ensemble, Severidad de la Sepsis

El *Ensemble* se construyó usando la técnica de *voting* y su sistema de votos se automatizó usando árboles de decisión, el esquema jerárquico de votos se muestra en la figura 38; se construyeron tres ensambladores, uno para cada problema y un cuarto que juntara sus predicciones, su desempeño se encuentra en la tabla 2 de la siguiente subsección.

Figura 38. Jerarquía de Votos del Ensemble



6.2.6 Modelo Predictivo para la Mortalidad

Para mortalidad no se reportan resultados relevantes, como se vio en la sección 6.3.2.1 la predicción con el modelo de flujo de datos con *bias*, figura 29, es aleatoria para la clase 1 (pacientes que fallecieron al ingreso de la institución). Por esto, crear un ensamblador no generará resultados favorables dado que los clasificadores predicen de forma incorrecta la clase 1. Se resalta que de los 563 pacientes seleccionados de la base: “Base de datos MATRIX Definitiva 31 Julio 2014v4”, solo 52 habían fallecido.

6.3 ETAPA 3: INTEGRACION RESULTADOS ZIMOGRAFÍA CON MODELOS PREDICTIVOS (ETAPA 1 y 2)

Con el modelo predictivo para la clasificación de grado de sepsis se probó si los resultados obtenidos de la medición automatizada de actividad de enzimas metaloproteinasas contribuyen a la mejora de su desempeño. Para esto se usó dos conjuntos de datos de entrada:

- La base de datos original con el campo de medidas de enzimas hecho por el experto.
- La base de datos original con ambas medidas de medición de enzimas, una hecha por el experto y otra usando el algoritmo.

Los resultados se pueden observar en las siguientes tablas:

Tabla 2. Métricas Accuracy y F1 score el dataset original.

METRICA/ CONJUNTO	DATOS TRAIN	DESVIACIÓN	DATOS TEST	DESVIACIÓN
ACCURACY	0.997	0.002	0.792	0.077
F1 SCORE	0.997	0.002	0.786	0.079

Tabla 3. Métricas True positive rate en los tres grupos de sepsis. Usando el dataset original

TRUE POSITIVE RATE	DATOS TRAIN	DESVIACIÓN	DATOS TEST	DESVIACIÓN
NO GRAVE	1.0	0.0	0.901	0.052
GRAVE	0.988	0.010	0.501	0.098
SHOCK SÉPTICO	0.998	0.004	0.845	0.160

Tabla 4. Métricas Accuracy y F1 score. Usando ambas medidas de zimografías.

METRICA/ CONJUNTO	DATOS TRAIN	DESVIACIÓN	DATOS TEST	DESVIACIÓN
ACCURACY	0.997	0.001	0.822	0.064
F1 SCORE	0.997	0.001	0.813	0.071

Tabla 5. Métricas True positive rate en los tres grupos de sepsis. Usando ambas medidas de zimografías.

TRUE POSITIVE RATE	DATOS TRAIN	DESVIACIÓN	DATOS TEST	DESVIACIÓN
NO GRAVE	1.0	0.0	0.892	0.099
GRAVE	0.987	0.005	0.570	0.180
SHOCK SÉPTICO	1.0	0.0	0.882	0.111

6.4. INTERFAZ WEB

Para la fácil interacción del usuario final con los algoritmos de predicción y la herramienta de procesamiento de zimografías, se creó una interfaz web llamada ZyAn usando el framework Django, que permite realizar desarrollo rápido de aplicaciones web.

6.4.1. Diagramas de Caso de Uso de Web App

Previo al desarrollo de la aplicación se diseñaron diagramas con el fin de definir las posibles situaciones e interacciones entre el usuario final y la aplicación para obtener un comportamiento deseado. El usuario debe completar el proceso de autenticación para poder acceder a los servicios de predicciones y tratamiento de imágenes.

Figura 39. Proceso para ingresar a la aplicación.

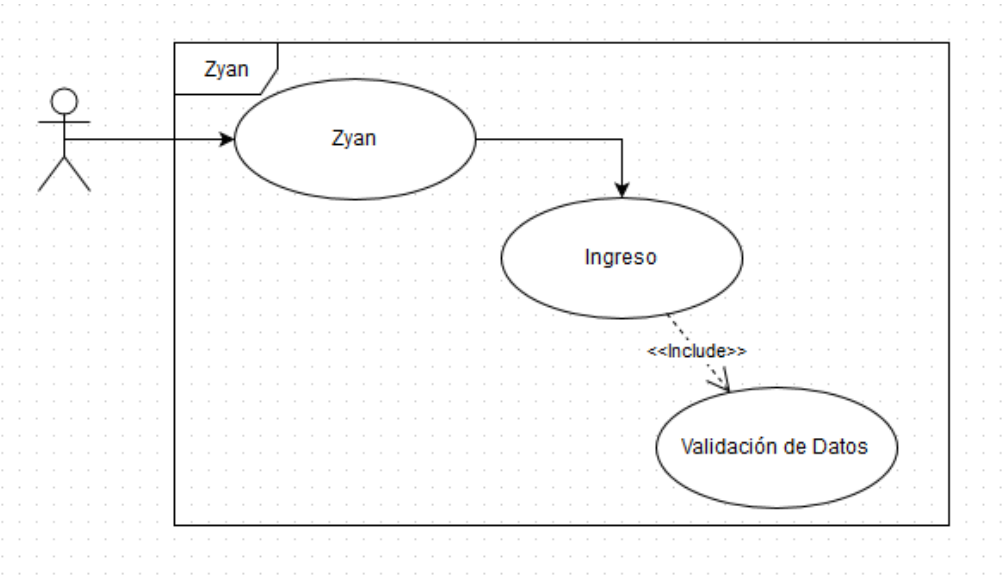
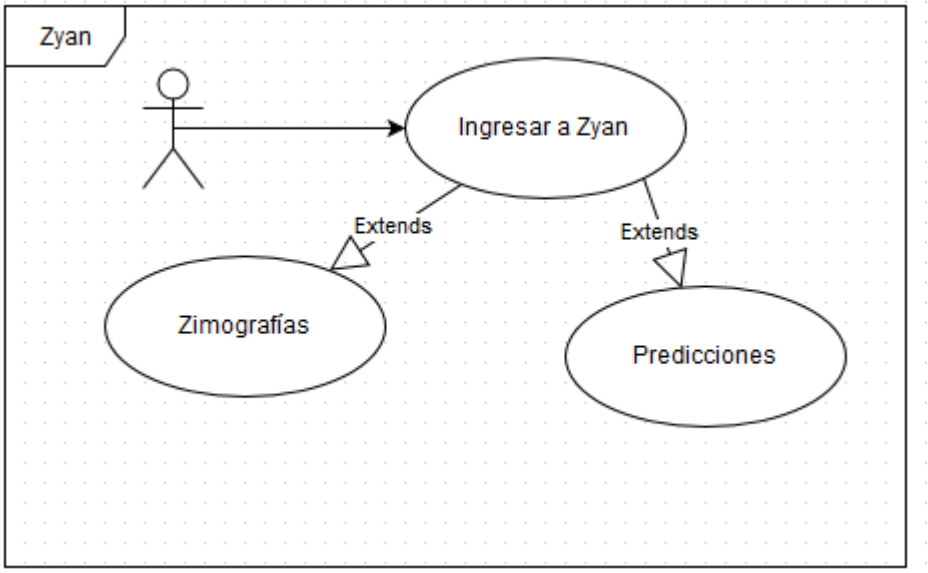


Figura 40. Caso de uso de la página de inicio de la aplicación



6.4.2 ARQUITECTURA DE SITIO WEB

Django es un framework que maneja una arquitectura escalable tipo Modelo-Plantilla-Vista (M.P.V), que permite mantener la lógica y el diseño separadas, permitiendo fácil mantenimiento y escalabilidad. M.P.V es análogo al Modelo Vista Controlador, a continuación se explica en detalle cada componente:

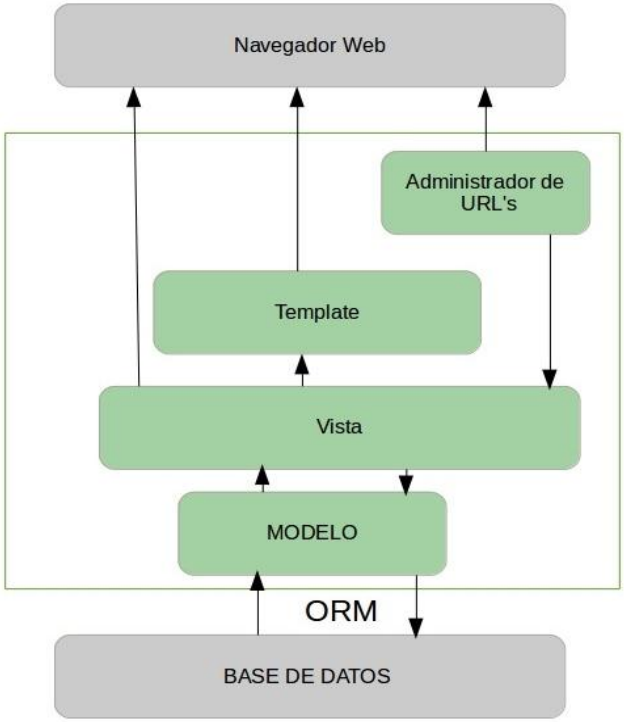
MODELO: Un modelo es la representación de los datos de una aplicación en forma de clase. Contiene los campos básicos y el comportamiento de los datos que serán almacenados. Cada modelo se convierte en una tabla de la base de datos, según el motor de base de datos seleccionado.

PLANTILLA: Una plantilla recibe los datos de la vista para su visualización en el navegador, por lo que está escrita en lenguaje HTML y Python.

VISTA: El propósito de la vista es determinar qué datos serán visualizados, para este fin utiliza funciones, también se encarga de otras tareas como el envío de correo electrónico, la autenticación con servicios externos y la validación de datos a través de formularios. El modelo orientado a objetos de Django permite escribir código Python en lugar de SQL para hacer las consultas que necesita la vista.

El sitio web consta de dos pequeñas aplicaciones: zimografías y predicciones. Dentro de la vista de cada aplicación se llaman los algoritmos creados.

Figura 41. Arquitectura de aplicación en Django.



6.4.3 INTERFAZ DE LA APLICACIÓN

Figura 42. Página inicio de Zyan.

Figura 43. Formulario de ingreso Zyan.

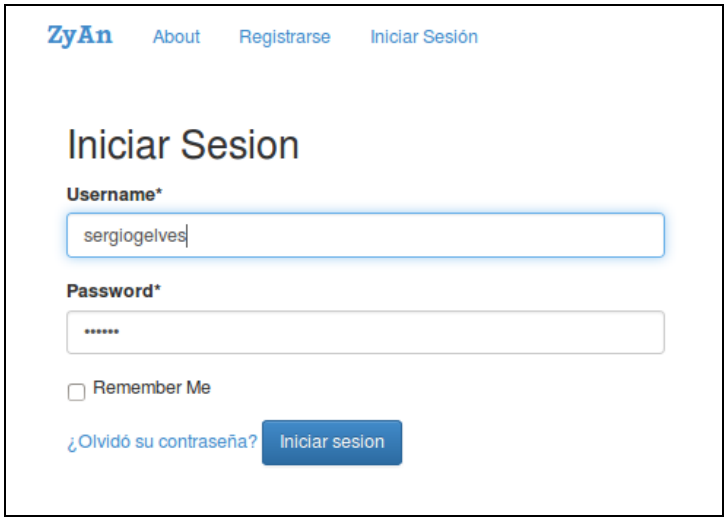


Figura 44. Interfaz módulo zimografías.

Figura 45. Visualización de resultados de procesar imagen.

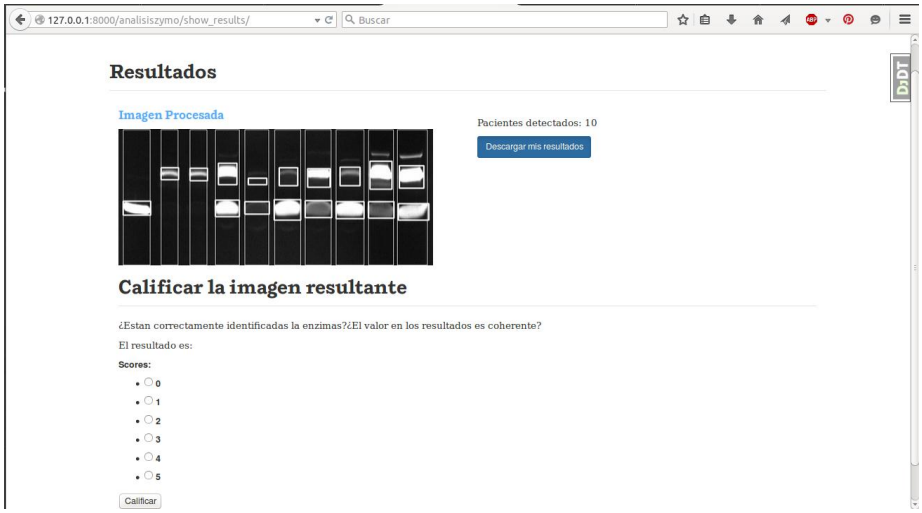


Figura 46. Interfaz módulo de Predicciones

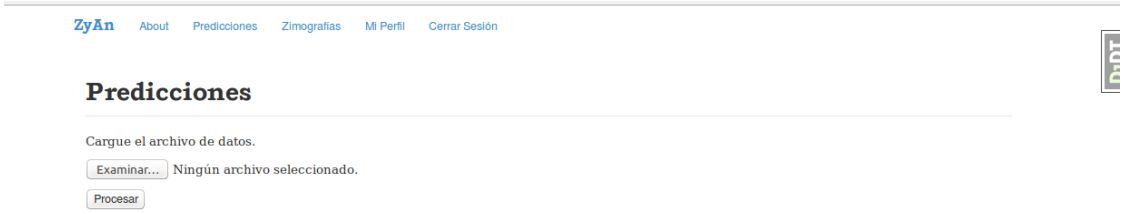


Figura 47. Interfaz de descarga de resultados de predicciones.

7. CONCLUSIONES

- El llevar un procedimiento estándar para el proceso de obtención de imágenes de zimografías es crucial para su posterior procesamiento, ya que la segmentación dependerá de factores como el tiempo en el que se escanea la imagen, el voltaje aplicado en la técnica de electroforesis y la cercanía entre las muestras de pacientes.

- El algoritmo desarrollado para la segmentación y cuantificación de la actividad de zimografías tiene un desempeño satisfactorio en imágenes de calidad. Ya que la segmentación no se basa en suposiciones de la forma en que se revela la actividad permitiendo segmentar bandas de diferente ancho y forma.

- La aplicación desarrollada en Django ayuda a que el usuario final interactúe de manera efectiva, amigable, y cuenta con una arquitectura escalable.

- Para el problema de clasificación de mortalidad por sepsis no fue posible predecir acertadamente por la falta de observaciones de pacientes que fallecieron durante el estudio.

- En el modelo predictivo para la severidad de sepsis se lograron predecir el 90.17%, 50.15% y el 84,6% (TPR medido con *cross-validation*) de los pacientes no graves, graves y con choque séptico del conjunto de pruebas. Estas medidas indican que el modelo no predice de forma aleatoria, por lo que podría servir de apoyo al diagnóstico del estadio de la sepsis. Para más detalles de los resultados revisar tablas 2 y 3.

- Al agregar al conjunto de entrada del *ensemble* las medidas de los descriptores de la actividad enzimática, las puntuaciones TPR de *cross-validation* aumentan a 89.23%, 57.05% y 88.21% para pacientes con sepsis no grave, grave y con choque séptico del conjunto de pruebas. Lo cual es un indicio de que las nuevas mediciones pueden ser representativas de la actividad enzimática. Los detalles de las desviaciones para las medidas de *crossvalidatios* se encuentran en las tablas 3 y 5.

8. RECOMENDACIONES

- En futuras investigaciones será necesario medir el desempeño del algoritmo de procesamiento de zimografías con técnicas de *feature matching*, comparando con las segmentaciones hechas manualmente por el experto en la técnica, para una valoración más precisa que no sea basada únicamente en la apreciación del usuario.

- Para hacer uso del modelo predictivo es necesario considerar factores demográficos, ya que diferentes poblaciones pueden reaccionar de formas diversas a las enfermedades y las variables necesarias para medir su estadio de sepsis podrían cambiar.

- Para futuras investigaciones sobre la aplicación de técnicas de aprendizaje de máquina a la predicción del estadio de la sepsis. Si se replica el modelo de predicción descrito en este libro, se recomienda usar un sistema de votación más complejo para el *ensemble*, que permita aprovechar aún más los aciertos en los *ensambles* binarios, de los problemas cero y dos, para la predicción de pacientes con sepsis grave (clase y problema uno).

REFERENCIAS BIBLIOGRÁFICAS

[1] BONE, R.C, et al. Definitions for Sepsis and Organ Failure and Guideline for the Use of Innovative Therapies in Sepsis. En: Chest Journal [en línea]. No. 6 (1992). <http://journal.publications.chestnet.org > [citado en 10 de febrero del 2015]

[2] STANFORD MACHINE LEARNING COURSERA [en línea]. <https://www.coursera.org/course/ml>[citado en 10 de febrero del 2015]

[3] SCIKIT LEARN. Suppor Vector Machines [en línea]. <http://scikit-learn.org/stable/modules/svm.html> > [citado en 8 de marzo del 2015]

[4] SHALIZI, Cosma. Data Mining Lectures: Classification and Regression Trees. p 4-23 . [en línea] <http://www.stat.cmu.edu/~cshalizi/350/lectures/>[citado en 4 de abril del 2015]

[5] SCIKIT LEARN. Tree algorithms [en línea]. <http://scikit-learn.org/stable/modules/tree.html#tree-algorithms > [citado en 8 de marzo del 2015]

[6] NG, Andrew; JORDAN, Michael. On Discriminative vs. Generative Classifiers: A comparison of logistic regression and naive Bayes. En: Neural Information Processing Systems (NIPS), 2001 [en línea] < http://papers.nips.cc/paper/2020-on-discriminative-vs-generative-classifiers-a-comparison-of-logistic-regression-and-naive-bayes> [citado en 23 de marzo del 2015]

[7] ALPAYDIN, Ethem. Introduction to Machine Learning. Combining Multiple Learners. Segunda edición. Ediciones The MIT Press, 2009.

[8] KIM, Youngho, et al. Segmentation of Protein Spots in 2D Gel Electrophoresis Images with Watersheds Using Hierarchical Threshold. En: Computer and Information Sciences – ISCIS 2003, Vol. 2869.

[9] KAABOUCH, N. An Analysis System for DNA Gel Electrophoresis Images based on Automatic Thresholding and Enhancement. En: IEEE EIT, 2007.

[10] ISMAIL, I; ELTAWEEL, Gh y NASSAR, H. Bands detection and Lanes segmentation in DNA Fingerprint. En: Journal of Information and Computing Science, 2014, Vol. 9 no. 4, p 243-251.

[11] RAMOS POLLÁN, Raul et al. Discovering Mammography-based Machine Learning Classifiers for Breast Cancer Diagnosis. En: Journal of Medical Systems,2012, Vol. 36, no.4, p 2259-2269.

[12] CHAWLA, Nitesh, et al. SMOTE: Synthetic Minority Over-sampling Technique. En: Journal of Artificial Intelligence Research, 2002, Vol.16, p 321-357.

[13] CHEN, Chao. Using Random Forest to Learn Imbalanced Data. [en línea] < http://statistics.berkeley.edu/> [citado en 23 de marzo del 2015]

[14] CRIMINISI; SHOTTON Y KONUKOGLU. Decision Forests for Classification, Regression, Density Estimation, Manifold Learning and Semi-Sup ervised Learning (2011)[en línea] <http://research.microsoft.com/pubs/155552/decisionForests_MSR_TR_2011_114.pdf >[citado en 12 de febrero del 2015]

BIBLIOGRAFÍA

ARDILA GOMEZ , William; MEJIA RIVERA, Sergio. Sistema de Información para la Gestión de Datos y Administración del Estudio de Detección de Marcadores Pronósticos de Mortalidad de la Enfermedad Sepsis (Sippam G-Sepsis). Bucaramanga, 2012. Trabajo de Grado (Ingeniero de Sistemas). Universidad Industrial de Santander. Facultad de Ingenierías Físico mecánicas. Escuela de Ingeniería de Sistemas.

DOCUMENTACION OPENCV [en línea] <http://opencv.org/documentation.html>

DJANGO DOCUMENTATION [en línea] < https://docs.djangoproject.com/en/1.8/>

GREENFELD, Daniel; ROY, Audrey. Two Scoops of Django. Best Practices for django 1.5. Primera edición, 2013.

MURPHY, Kevin. Machine Learning a probabilistic perspective. The MIT Press. 2012.

REYES TARAZONA, Jhon; PINILLA SANCHEZ, Samuel. Optimización, Sistematización y Visualización de los Resultados Obtenidos del Estudio de la Enfermedad Sepsis. Bucaramanga, 2014. Trabajo de Grado (Ingeniero de Sistemas). Universidad Industrial de Santander. Facultad de Ingenierías Físico mecánicas. Escuela de Ingeniería de Sistemas.

SCIKIT-LEARN DOCUMENTATION. [en línea]<http://scikit-learn.org/stable/documentation.html>

SUCAR, L. Enrique y GOMEZ, Giovani. Visión Computacional.

SZELISKI, Richard. Computer Vision: Algorithms and Applications, Electronic Draft. Disponible en: < http://szeliski.org/Book/>