ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

DISEÑO DE UN ALGORITMO PARA LA DETECCIÓN Y VALIDACIÓN DE PATRONES DE BIOMARCACIÓN EN CONJUNTOS DE DATOS DE MEDICIONES DE ESPECTROMETRÍA DE MASAS APLICADOS AL ESTUDIO DEL CÁNCER

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES

SOFÍA JAZMINA CALLE JORDÁN

jazmina.calle.jordan@gmail.com

SILVIA SOLEDAD CHASILUISA PANZA

silvia.soledad.chasiluisa@gmail.com

DIRECTOR: ING. ROBERTO HERRERA

roberto.herrera.lara@gmail.com

CODIRECTOR: ING. JORGE CARVAJAL

jorge.carvajal@epn.edu.ec

QUITO, MARZO 2016

DECLARACIÓN

Nosotros, Sofía Jazmina Calle Jordán y Silvia Soledad Chasiluisa Panza, declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he (hemos) consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos mis nuestros derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Sofía Jazmina Calle Jordán

Silvia Soledad Chasiluisa Panza

CERTIFICACION

Certificamos que el presente trabajo fue desarrollado por Sofía Jazmina Calle Jordán y Silvia Soledad Chasiluisa Panza, bajo nuestra supervisión.

Ing. Roberto Herrera-Lara
DIRECTOR

Ing. Jorge Carvajal
CODIRECTOR

AGRADECIMIENTOS

Agradezco a Dios por ser mi guía, mi creador y sobre todo por darme la dicha de vivir y haberme dado a unos seres tan maravillosos mis padres, mi tía y mi hermano, los que han sido siempre mi fortaleza, la que me alienta y no me han abandonado para hacer esta una meta en la que se plasme todo mi esfuerzo y dedicación.

A la facultad de Ingeniería Eléctrica y Electrónica, a sus profesores por impartir sus conocimientos y experiencias en el transcurso de mi carrera estudiantil.

Sofía Jazmina Calle Jordán

A Dios que me ha otorgado la sed de conocimiento, me ha guiado en esta búsqueda y ha puesto a mi familia como la más grata de la compañía en esta travesía.

A mi compañera de tesis y director por el apoyo y las enseñanzas compartidas durante la realización de esta tesis.

Silvia Soledad Chasiluisa Panza

DEDICATORIA

Dedico el presente trabajo de investigación a mis padres Dolores y Joaquín por su apoyo brindado, entrega, preocupación, sacrificio y paciencia durante toda mi vida estudiantil y en los más difíciles en los que más los necesite.

A mi segunda madre Alicia por siempre darme sus palabras de aliento para no rendirme jamás, a mi hermano Christian por compartir sus experiencias conmigo y alentarme para seguir adelante, de igual manera a una persona muy especial que siempre me ha sabido apoyar a terminar una de mis metas.

Sofía Jazmina Calle Jordán

Dedico esta tesis a Dios mi maestro, a mi madre que me enseñó a perseverar y quién ha sido mi primer educador y a mis hermanos quienes me brindaron palabras de aliento.

Silvia Soledad Chasiluisa Panza

Índice General

Índice de Figuras

Índice de Tablas

Índice de Algoritmos

Índice de Códigos de Programación

RESUMEN

Las grandes cantidades de datos generados en la actualidad por las enormes y radicales revoluciones tecnológicas han abierto nuevos campos de investigación en aplicaciones medicas orientadas al estudio, caracterización y búsqueda de posibles curas del cáncer. Existen muchos trabajos al respecto, orientados a la búsqueda de patrones de datos, que puedan dar pistas sobre como tratar este mal. Desde el punto de vista medico, es muy difícil realizar nuevos avances que no involucren estudios invasivos sobre pacientes ya en estado patológico, ademas, la eficiencia del diagnostico medico se ve muy limitada, ya que el cáncer, de por si es una enfermedad asintomática y presenta síntomas, una vez que ya ha infectado órganos, o a su vez, se encuentra ya en estado de metástasis.

Existen muchas técnicas de adquisición de datos para aplicaciones medicas, una de las mas prometedoras es la espectrometría de masas. La espectrometría de masas produce una representación geométrica de la composición química de las muestras analizadas. Este patrón geométrico esta representado por dos vectores numéricos, la intensidad relativa y la relación masa a carga. Al graficar estos dos vectores se obtiene una curva similar a una señal discreta en el tiempo. Un gran grupo de estas señales representa un conjunto de datos de donde se puede extraer información trascendental sobre la descripción química de un estado biológico, sea este normal, o patológico. Algoritmos de Data Mining y Machine Learning son aplicados a estos conjuntos de datos, para buscar información de interés y dar nuevas luces a encontrar la cura del cáncer.

El presente proyecto de titulación tiene por objetivo el desarrollo e implementación de un nuevo algoritmo de extracción de información de mediciones de espectrometría de masas haciendo uso de un filtro estadístico compuesto por las pruebas t-Student, Wilcoxon y χ^2 (Chi cuadrado) y su validación a través del algoritmo de Adaboost M.2 configurado en *crossvalidation*. El algoritmo propuesto en este proyecto tiene como objetivos disminuir la carga computacional mediante la aplicación de metodologías heurísticas en la minería de datos y aprendizaje automático en la etapa de Machine Learning. El buen rendimiento obtenido en las etapas de simulación y pruebas respalda el desempeño del algoritmo propuesto. Las pruebas realizadas están contrastadas a doble ciego, ya que no solamente se mide el rendimiento del algoritmo de clasificación en función de la curva de aprendizaje, sino que también se comprueba su rendimiento haciendo uso de pruebas de *crossvalidation*. Un análisis comparativo de las curvas de rendimiento permiten observar que los efectos del *overfitting* y *underfitting* se ven

disminuidos gracias al alto nivel de rechazo que presentan los algoritmos de *boosting learning*, especialmente *Adaboost M.2*.

Los resultados presentados obtenidos en las simulaciones realizadas son presentados en forma gráfica y como archivos de texto plano. En los gráficos se muestran de forma sobrelapada el espectro original de datos y las zonas de interés donde se encuentra la información relevante del espectro, la cual contiene la descripción química de los biomarcadores que caracterizan los diferentes estados biológicos del cáncer. Cada gráfico esta acompañado de un archivo de texto donde se incluyen los valores numéricos de las intensidades de abundancia de cada punto del espectro, donde se definan zonas de biomarcación.

En la parte final se incluyen como anexos las Publicaciones originadas de la investigación de este Proyecto de Graduación.

PRESENTACIÓN

El presente proyecto de titulación se origina siguiendo las pautas dejadas por los grandes centros de investigación en medicina aplicada en su búsqueda de una cura para el cáncer. En el Ecuador, la Escuela Politécnica Nacional se ha caracterizado por liderar los campos de investigación, orientando los rumbos en ciencia y tecnología en el Ecuador. Hoy por hoy, los grandes pasos dados en medicina, gracias al uso de las computadoras y técnicas de procesamiento de datos, data mining, machine learning da nuevas esperanzas en erradicar estos males.

La perspectiva de estudio presentada en este trabajo tiene como objetivo la masificación de herramientas informáticas orientadas a la medicina, por lo que se ha puesto como meta, el desarrollo de un algoritmo que optimice los recursos computacionales a plataformas informáticas comerciales, laptops, computadoras de escritorio, etc. Desde este punto de vista a continuación se presenta una herramienta informática en sus primeras versiones, donde se ha hecho un especial esfuerzo en pulir las etapas del desarrollo de los cálculos numéricos, controlando el rendimiento de la misma. Dicha herramienta implementa un algoritmo descrito en esta tesis, desde un punto de vista teórico - practico. El algoritmo esta implementado en Matlab usando el Bioinformatics Toolbox y el Statistics and Machine Learning Toolbox. Este primer trabajo deja abierta la puerta a nuevos desarrollos e implementaciones en lenguajes de programación dedicados, a fin de conseguir una herramienta mas rápida y efectiva. Los resultados obtenidos en este trabajo dejan ver, que los lineamientos mediante los cuales se describe el algoritmo propuesto, representan un avance prometedor en definir una herramienta informática basada en análisis estadísticos, data mining y machine learning para el estudio y búsqueda de la cura para el cáncer.

CAPÍTULO 1

Marco Teórico y Conceptos Introductorios

En el presente capítulo se presenta el objetivo general y los objetivos específicos de esta tesis, así como también se realiza una introducción y revisión de los conceptos básicos para la presentación del problema a solucionar.

OBJETIVO GENERAL

- Diseñar un Algoritmo para la Detección y Validación de Patrones de Biomarcación en conjuntos de datos de mediciones de espectrometría de masas aplicados al estudio del cáncer, usando un filtro estadístico basado en las Pruebas de *t-Student, U-Mann–Whitney(Wilcoxon rank sum test), Chi-Cuadrado* y su validación usando *Adaboost.M1* sobre Arboles de decisión.

OBJETIVOS ESPECÍFICOS

- Procesar los datos de las mediciones de masas y empaquetarlos en conjuntos de datos de dos grupos de análisis, saludable y patológico.
- Analizar el conjunto de datos de los grupos saludable y patológico usando las pruebas estadísticas *t-Student, U-Mann–Whitney, Chi-Cuadrado* y seleccionar aquellas cuya probabilidad de ocurrencia sea mínima.
- Validar los datos seleccionados mediante las pruebas estadísticas de *t-Student, U-Mann–Whitney, Chi-Cuadrado* usando como métrica el error de clasificación al aplicar el algoritmo *Adaboost.M1* sobre árboles de decisión.

1.1 INTRODUCCIÓN

En la actualidad, las revoluciones tecnológicas, tanto en la instrumentación electrónica y ciencias computacionales han dado origen al nacimiento e impulsado nuevas ciencias multidisciplinares tales como la minería de datos y el machine learning. La minería de datos y el machine learning son excelentes herramientas en la construcción y validación de modelos matemáticos, los cuales describen fenómenos naturales, procesos o a su vez el comportamiento de entidades en un sistema cerrado. Llevando estas ideas a la practica, en función de la cantidad de datos, se pueden construir modelos, mediante los cuales se puede entender la naturaleza y el entorno que nos rodea, ¿como funciona el clima?, ¿cuando caerá la bolsa de valores?, ¿cuando una un engrande en un gran proceso industrial fallara?, ¿como evoluciona un tejido?, ¿esta aquel paciente enfermo?. Las posibilidades de elaborar un modelo matemático que describa el comportamiento de determinado fenómeno, son inimaginables.

En este proyecto de titulación se hace uso de estas herramientas, para abordar desde una nueva perspectiva, un análisis de datos, en aplicaciones médicas, orientadas a la caracterización del cáncer. Los datos analizados son producidos en espectrómetros de masas, los cuales sobre cada muestra analizada, producen una serie de valores numéricos, que al ser graficados, representan en forma de un patrón geométrico, la composición química de la muestra analizada, este patrón, suele llamarse comúnmente en la literatura espectro. Una muestra analizada en un espectrómetro de masas producirá un espectro característico en donde se muestra su composición química y a su vez, su estado patológico. Si esta muestra posee mutaciones, sean estas, mínimas o muy notorias, su estado se vera reflejado en el espectro de masas, producido al analizar la muestra.

Los logros que la medicina ha conseguido en los últimos tiempos han sido revolucionarios. Muchas enfermedades han sido controladas y hasta eliminadas de las estadísticas de mortandad de los pueblos, no así el cáncer, en donde no se ha encontrado aun ninguna metodología científica ni empírica que indique la presencia de esta patología. Esta enfermedad es tan difícil de diagnosticar que empezando por su asintomatismo en estado temprano, tratarla y combatirla resulta cada vez mas difícil debido a la gran variedad de tipos que existen. Este tipo de patologías atacan a una gran variedad de órganos y sistemas vitales tales como los pulmones, cuello, garganta, piel, sangre, cerebro, estomago, ovarios, próstata, testículos, mamas, el colon. Diagnosticar usando las metodologías tradicionales bajo tanta variedad de entornos resulta un gran reto y

la mayoría de las veces, solamente es posible detectar la presencia de cáncer, cuando este mal esta ya en etapas avanzadas.

El cáncer, de manera general, representa una mutación no natural en cualquier tipo de tejidos. Esta mutación, desde un punto de vista matemático representa variaciones aleatorias en la composición química de los tejidos, la cual, puede ser estimada, estudiada y validada usando algoritmos numéricos, que la definan, caractericen y reconozcan de manera eficiente y no invasiva. La búsqueda de esta información se hará de entre dos grupos de análisis, un conjunto de mediciones de pacientes en estado saludable y un conjunto en estado patológico.

La producción de las muestras de análisis, se la realiza de distintas formas, la principal, sometiendo al espectrómetro muestras para la producción de espectros. Normalmente, estos datos suelen estar disponibles para trabajos de investigación en bibliotecas dedicadas a la búsqueda de la cura para el cáncer, las cuales actualmente, también brindan acceso a estos datos de manera online, en sus diferentes sitios web.

Las mediciones de las muestras analizadas, están constituidas por una serie de valores numéricos, en un eje bidimensional, los cuales al ser graficados representan un patrón geométrico similar a la curva de una señal discreta en el tiempo. Dicho gráfico representa la composición química de la muestra analizada. En la 1.1 se muestra una representación clásica de un espectro de masas.

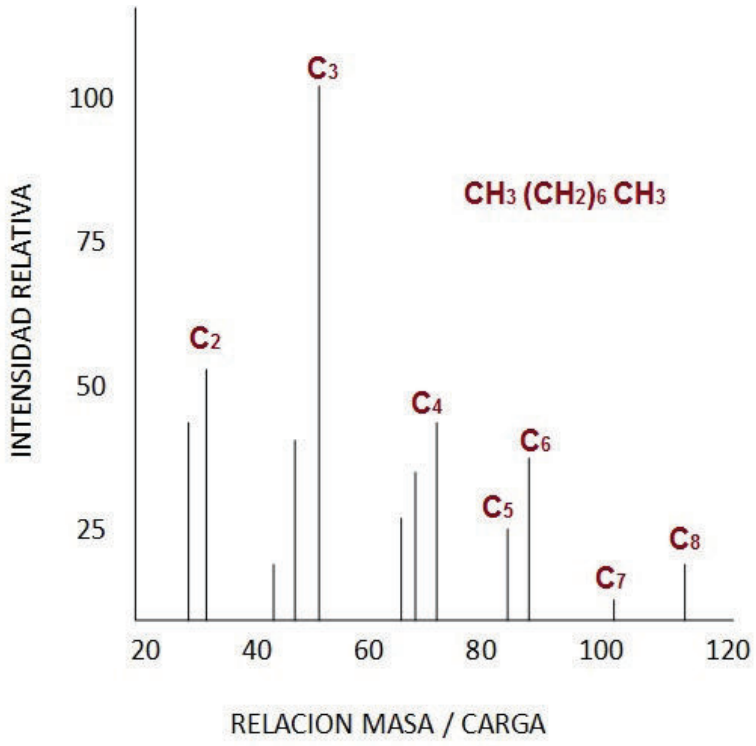


Figura 1.1.: Representación característica de una Medición o Espectro de Masas

La investigación presentada en este proyecto parte de la idea de disponer de un gran conjunto de datos $\mathcal{D}_{\mathcal{X},\mathcal{Y}}$, donde $\mathcal{X}$ es la cantidad de mediciones(o espectros de masas) y $\mathcal{Y}$ las etiquetas de estas mediciones(los nombres de las muestras analizadas), sobre los cuales se aplicaran algoritmos para definir cuales son las características que diferencian entre estados saludables y estados patológicos.

La definición formal del conjunto de datos $\mathcal{D}_{\mathcal{X},\mathcal{Y}}$ se muestran en la Ecuación 1.1,

$$\mathcal{D}_{\mathcal{X},\mathcal{Y}} = \begin{cases} \mathcal{X}, \text{ donde } \mathcal{X} = \{x_1, x_2, x_3, \ldots, x_n\} \\ \mathcal{Y}, \text{ donde } \mathcal{Y} = \{y_1, y_2, y_3, \ldots, y_n\} \end{cases} \tag{1.1}$$

cada uno de los vectores, $\mathcal{X}$ y $\mathcal{Y}$ se definen a su vez en las ecuaciones 1.2 y 1.3,

$$\mathcal{X} = \{x_1, x_2, x_3, \ldots, x_n\} = x_{i=1}^{n}, \text{ donde } i \in \mathbb{Z}^+ \tag{1.2}$$

$$\mathcal{Y} = \{y_1, y_2, y_3, \ldots, y_n\} = y_{i=1}^{n}, \text{ donde } i \in \mathbb{Z}^+ \tag{1.3}$$

los elementos $x_{i=1}^{n}$ y $y_{i=1}^{n}$ se presentan en forma pareada, es decir $\{x_1, y_1\}$, $\{x_2, y_2\}$, ..., $\{x_n, y_n\}$. Cada uno de los elementos $x_{i=1}^{n}$ esta formado a su vez por una serie de puntos, descritos por dos vectores de valores, el primero de la abundancia relativa de los iones y el segundo de la relación masa a carga. Juntos describen lo que para esta investigación se asumirá es una señal discreta en el tiempo. Haciendo referencia nuevamente a la Figura 1.1, con las condiciones, aquí descritas, esta adquiere las características mostradas en la Figura 1.2.

En la Figura 1.2 se muestra sobrelapada sobre la figura original una curva formada por los puntos de las coordenadas descritas por los vectores de abundancia y relación masa a carga de cada una de las mediciones o espectros representados en los elementos $x_{i=1}^{n}$, esta curva a su vez, representa el nexo entre estos patrones geométricos y la conceptualización base del procesamiento de estas mediciones agrupadas en conjuntos de datos.

Cada uno de los elementos $x_{i=1}^{n}$ están definidos por la Ecuación 1.4,

$$x_{i=1}^{n} = \{I_{i=1}^{n}, mz_{i=1}^{n}\}, \text{ donde } I_{i=1}^{n} \in \mathbb{R}^+ \text{ y } mz_{i=1}^{n} \in \mathbb{R}^+ \tag{1.4}$$

los elementos de la Ecuación 1.4, $I_{i=1}^{n}$ y $mz_{i=1}^{n}$, son producidos en cada medición realizada. El grupo de mediciones, en cada caso de estudio, representa un grupo de

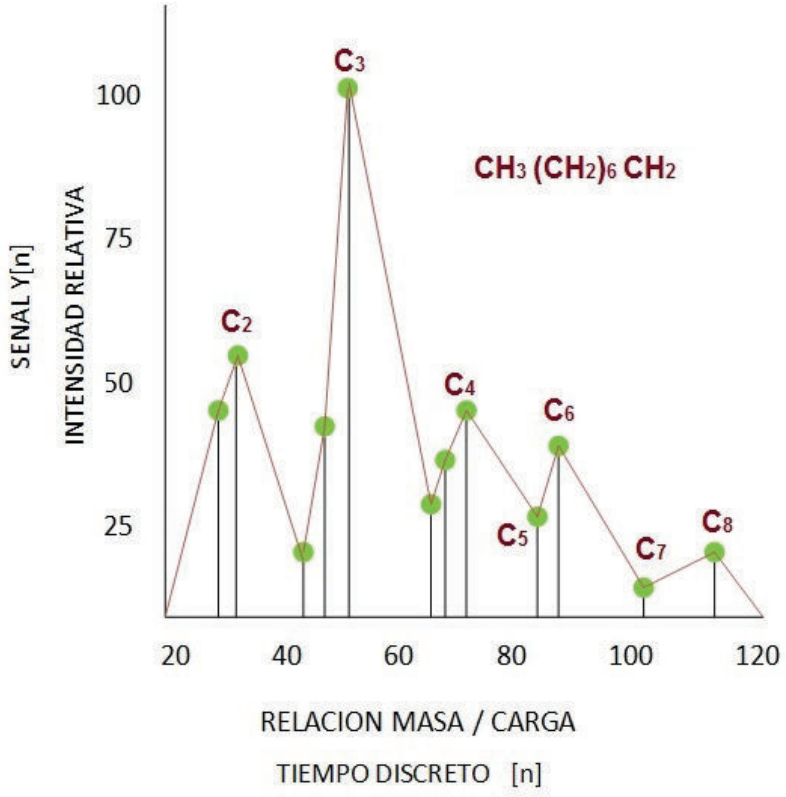


Figura 1.2.: Analogía entre el Espectro de Masas y una Señal Discreta en el tiempo

análisis de datos. Es decir, si se miden n veces un grupo de pacientes en estados no patológicos, se dispondrá de un grupo de análisis de n muestras, al cual se representara como $\mathbb{G}_n$ y estará formado por la medición y una etiqueta de identificación.

En esta investigación, se tiene como objetivo, analizar un conjunto de datos, formado por dos grupos de análisis, uno en estado saludable, abreviado como $\{\text{e.s.}\}$ y un grupo de análisis formado por mediciones en estado patológico , abreviado como $\{\text{e.p.}\}$, de tal forma que el conjunto de datos, descrito en la Ecuación 1.1, toma la definición de la Ecuación 1.5,

$$\mathcal{D}_{\mathcal{X},\mathcal{Y}} = \{\mathbb{G}_1, \mathbb{G}_2\} \tag{1.5}$$

donde $\mathbb{G}_1$ contendrá a todas las mediciones realizadas sobre muestras saludables y $\mathbb{G}_2$, las muestras de estados patológicos, en ambos casos, se incluirán además sus respectivas etiquetas de identificación. Una vez definidas de manera formal, las mediciones, los grupos de análisis y el conjunto de datos, el siguiente paso, es definir donde se encuentran las mutaciones entre los grupos de análisis $\mathbb{G}_1$ y $\mathbb{G}_2$, lo cual indicara un estado patológico, en consecuencia, pre-cancerígeno o a su vez, cancerígeno.

La etapa, donde se definirán las mutaciones entre los grupos de análisis, se define, selección de características. Desde el punto de vista de la minería de datos y el machine learning, estas características, adquieren un carácter discriminan-tes inter-grupal, es decir, la idea es definir, cuales son los puntos de las curvas de medición, que cambian entre un estado saludable y un estado patológico. Estos puntos, permiten que diferenciar el estado biológico de la muestra analizada.

Los puntos de características discriminantes seleccionados, definen, lo que en medicina se conoce como biomarcadores. Los biomarcadores son parámetros de análisis que definen el estado biológico de la muestra analizada, es decir, pueden describir estados biológicos tales como, saludable, patológico y describir el nivel de evolución frente a la administración de un fármaco.

La ventaja esencial que presenta el algoritmo descrito en este trabajo de graduación, es, la validación de resultados. Si bien, todo el proceso descrito hasta esta etapa, se basa en la aplicación de algoritmos computacionales a un conjunto de vectores numéricos, estos algoritmos, pueden fallar y dar resultados falsos, ya sea falsos positivos o falsos negativos. El control de los resultados se lo realiza en la etapa de validación, donde se realizar simulaciones de reconocimiento, de muestras en estado patológico o saludables de manera aleatoria. El error en el reconocimiento de los estados biológicos de las mediciones usadas para la etapa de validación, definirá, si las características seleccionadas por el algoritmo, como parámetros discriminantes, son o no las mas adecuadas.

Debido a las facilidades que presentan las plataformas computacionales para la implementación de escenarios de simulación, en esta investigación se crea un escenario virtual, donde se asume para la etapa de validación, conocidas todas las mutaciones intergrupales y se realizan búsquedas aleatorias sobre grupos de análisis de forma secuencial usando validación cruzada. Este escenario se describe gráficamente en la Figura 1.3.

Una vez realizada la validación de resultados, la etapa final consiste en la traducción del patrón geométrico encontrado a su equivalente químico, el cual, se denominara de forma genérica como biomarcador.

Muchos centros de investigación han desarrollado algoritmos en búsqueda de los objetivos expuestos anteriormente, pero el trabajo a abordar es tan complejo, que estos esfuerzos resultan pocos y la cantidad de información a analizar, muy exuberante.

Figura 1.3.: Proceso de Validación Cruzada de Resultados

La cantidad de datos a analizar y someter a todas las etapas descritas anteriormente suele alcanzar fácilmente los cientos de miles de puntos, donde una medición(espectro de masas), dimensionalmente luego del espectrómetro alcanza dimensiones vectoriales del orden de 10000 puntos hasta los 300000 puntos. Haciendo referencia a que por un grupo de análisis de datos, es este tipo de aplicaciones se manejan números de 100 a 200 mediciones en promedio, un conjunto de datos binario($\{\text{saludable}, \text{patológico}\}$), el número de valores numéricos a procesar alcanzaría fácilmente los 60 millones de puntos, lo cual representa una excesiva carga computacional para cualquier computadora promedio. El gran dilema ante todo esto es, como abordar el análisis de tal cantidad de datos sin incurrir en equipos costosos, ni gastos elevados en cuanto a infraestructura de hardware y software. La respuesta radica en la aplicación de algoritmos matemáticos, los cuales vayan analizando en varias etapas, cual es el grado de importancia de diferentes subconjuntos de datos, siguiendo el paradigma de divide y vencerás. Una vez evaluados estos conjuntos, los cuales, no posean ningún grado de discriminación, son descartados y el proceso se repite hasta alcanzar cierto criterio de paro de la secuencia, normalmente, un valor de error de clasificación.

En las siguientes secciones se realiza una revisión al estado del arte, teniendo en cuenta los últimos trabajos publicados sobre las temáticas de adquisición de mediciones(espectros de masas), procesamiento de mediciones, algoritmos, metodologías y técnicas aplicadas, procesos de selección de características y técnicas de valida-

ción de resultados usando aprendizaje supervisado. Todas las referencias citadas a continuación se adjuntan en la bibliografía de este trabajo.

1.2 METODOLOGÍAS DE ADQUISICIÓN DE DATOS

La espectrometría de masas ha experimentado grandes cambios desde sus inicios a tal punto que hoy en día existen una serie de variantes en cuanto a la metodología de ionizar las muestra a ser analizada. Entre las mas importantes estan: Electron Ionization(EI), Fast Atom Bombardment(FAB), Electrospray Ionization(ESI), Matrix-Assited Laser Desorption Ionization(MALDI), Electrospray Ionization Mass Spectrometry(ESI-MS), Surface Enhanced Laser Desorption Ionization Mass Spectrometry (SELDI-MS).

La técnica mas comúnmente usada es EI, la cual trabaja muy bien para moléculas pequeñas, que son fácilmente vaporizadas. Sin embargo esta técnica no trabaja muy bien con muestras térmicamente sensibles, donde la rápida vaporización de la muestra no brinda un tiempo adecuado para la adquisición del espectro. En el caso de moléculas grandes con baja estabilidad térmica y no volátiles hay que usar otros métodos de ionización y vaporización.

Los métodos suaves de ionización como FAB, ESI MALDI, mejoran las limitaciones de éste para moléculas pequeñas volátiles del EI. Los métodos ESI-MS, MALDI-MS, SELDI-MS, han ganado gran aceptación en procesos de ionización de moléculas grandes. Estos métodos pueden detectar moléculas de gran tamaño, baja volatilidad y térmicamente estables, las cuales forman parte de proteínas, aminoácidos y enzimas en muestras biológicas.

Los métodos de ionización en fase gaseosa mencionados anteriormente requieren de procesos adecuados de preparación de la muestra, previo a ser sometida al espectrómetro de masas. En el caso del ESI, la preparación de la muestra no requiere mezclas o procesos de saturación con reactivos adicionales, ya que las moléculas son separadas usando técnicas de cromatografía liquida, donde el liquido de la muestra es introducida desde el final de la columna de cromatografía directamente hacia el ionizador.

En el caso de preparar muestras para ionizar usando metodologías de MALDI, el primer paso es preparar una solución con la muestra a analizar. Esta mezcla reposa en un pozuelo mientras el solvente se seca y la muestra a analizar se evapora, dejando como residuo una matriz cristalizada. En la metodología de SELDI se una una técnica similar a la de MALDI, a diferencia que SELDI es una metodología propietaria. SELDI

se enfoca en hacer la separación selectiva de proteínas desde mezclas heterogéneas, donde el proceso de preparación de la muestra consiste en colocarla en una base de propiedades químicas similares a la proteína aplicada, los componentes similares a la mezcla se adherirán a la base, mientras que los diferentes de la mezcla, serán rechazados de la base. Una vez terminada esta reacción, la base es sometida a un proceso de secado y cristalizado, previo a ser colocada en el ionizador.

Entre las aplicaciones principales de las técnicas anteriormente mencionadas están la adquisición de datos para: la identificación de compuestos desconocidos, determinación de la composición isotónica de los elementos de una molécula y determinar la estructura de un compuesto por observación de su fragmentación, determinación del peso molecular de péptidos, proteínas y oligonucleicos, definición de secuencias de aminoácidos en muestras de polipéptidos y proteínas, identificación de compuestos químicos en fluidos biológicos como sangre, orina y saliva, determinación del peso molecular de substancias, determinación de la formula química molecular, entre otras. Adicionalmente, estas técnicas son usadas para cuantificar la cantidad de un compuesto o estudiar sus propiedades fundamentales en fase gaseosa.

La espectrometría de masas es una técnica muy común en laboratorios que estudian las propiedades físicas, químicas y biológicas de una amplia variedad de compuestos. Entre las ventajas de esta técnica están la elevada sensibilidad sobre otras técnicas de adquisición de datos similares, alta precisión en la caracterización de patrones de fragmentación para identificar o confirmar la presencia de presuntos compuestos en una muestra, brinda ademas información del peso molecular y información de la abundancia de isotopos de determinado elemento en la muestra analizada, sin embargo no todo son ventajas, ya que la espectrometría de masas presenta limitaciones para la detección de muestras isómeras. En su utilización, la espectrometría de masas presenta limitaciones ya que como se expuso anteriormente, es difícil de aplicar en compuestos no volátiles, necesita compuestos lo mas puros posibles, ya que las impurezas producen en la adquisición de la señal, el denominado ruido químico y finalmente, el espectro de medición no expresa la información directamente, sino que necesita ser interpretado.

Debido a la orientación de este trabajo al procesamiento y análisis de datos, el detalle mismo de la instrumentación química no se aborda de manera extensa. El objetivo de este capitulo es conocer la procedencia de los datos, de forma general. Los procesos involucrados en la producción de los espectros y en instrumental involucrado puede encontrarse a detalle en las referencias bibliográficas de este trabajo. El punto de

partida para esta investigación es por tanto disponer de los datos listos en formato digital. En la Figura 1.4 se muestra un diagrama de flujo de la producción de datos con las técnicas mencionadas en esta sección en un espectrómetro de masas, hasta el punto donde se disponen ya de las mediciones en formato digital, donde dichos valores numéricos se visualizan en una computadora.

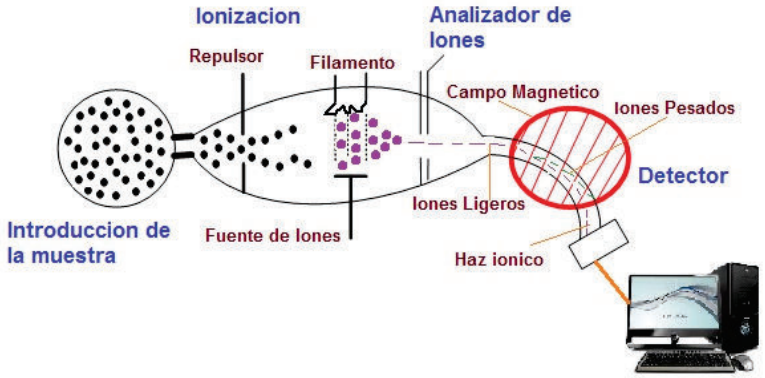


Figura 1.4.: Adquisición de Espectros hasta su visualización en una Computadora

Los datos visualizados están estructurados en como una cadena de valores numéricos, donde cada medición posee sus valores de intensidad y de relación masa a carga, como se muestra en la Figura 1.5. Estos datos serán los que, desde aquí en adelante, esta investigación tomara como punto de partida en el desarrollo del algoritmo propuesto. De manera generica, se hara referencia a estos datos como, el conjunto de datos a analizar, definido anteriormente como $\mathcal{D} = \{\mathbb{G}_1, \mathbb{G}_2\}$. Donde $\mathbb{G}_1$ y $\mathbb{G}_2$ representan los grupos de análisis de las mediciones realizadas, los cuales estan representados por los vectores numéricos mostrados en la Figura 1.5.

	$\{I_1, mz_1\}$	$\{I_2, mz_2\}$	$\{I_3, mz_3\}$	...	$\{I_{n-1}, mz_{n-1}\}$	$\{I_n, , mz_n\}$
Medición 1	...	...	...	...	...	...
Medición 2	...	...	...	...	...	...
Medición 3	...	...	...	...	...	...
...	...	...	...	...	...	...
Medición $n-1$	...	...	...	...	...	...
Medición n	...	...	...	...	...	...

Figura 1.5.: Datos adquiridos en su Representación Matricial

1.3 TÉCNICAS Y ALGORITMOS DE PROCESAMIENTO DE MEDICIONES

Una vez definidos los grupos de análisis $\mathbb{G}_1$ y $\mathbb{G}_2$ y el conjunto de datos $\mathcal{D} = \{\mathbb{G}_1, \mathbb{G}_2\}$, el siguiente paso es, el procesamiento de estos valores numéricos para mejorar la ca-

lidad de la información a estudiar. Esta fase parte con los la disposición de los datos a procesar, pero aquí cabe la pregunta, para que se hace el procesamiento de las mediciones de datos adquiridos?, la respuesta, como en todos los procesos de adquisición de datos, esta, en mitigar los efectos propios de la aplicación sobre la cual, se usa instrumentación, para adquirir los datos. En el proceso de adquisición de datos de aplicaciones relativas a la espectrometría de masas suelen producirse problemas de calibración de equipos, contaminación de la muestra a analizar, efectos del ruido, corrimientos en los ejes de intensidad o a su vez en los ejes de masa a carga y el mas difícil de lidiar, la excesiva cantidad de valores numéricos, puntos a analizar, los cuales, por ser tantos, saturan cualquier Plataforma Computacional de análisis[1].

En esta etapa de de minería de datos, entra en escena el procesamiento digital de señales, aplicado a las mediciones, las cuales, por analogía, al ser una curva de valores tomados en tiempos discretos, pueden tratarse de igual manera que las señales en tiempo discreto. Este procesamiento parte con la disminución de la enorme cantidad de puntos de los vectores de cada medición de datos[2], a una cantidad mas manejable por las plataformas computacionales comerciales. Esta disminución se la realiza por medio de etapas secuenciales de remuestreo de señales.

El remuestreo de señales asume que cada medición(espectro) es una señal discreta en el tiempo, donde, tanto la intensidad relativa como la relación masa a carga, cambian su tamaño en función de la capacidad computacional disponible. La dimensionalidad de las mediciones puede ser alterada elevando o disminuyendo la cantidad de valores numéricos(frecuencia de muestreo[3]), en función de las necesidades de las mediciones y la aplicación. Con la reducción dimensional, apenas ha empezado el procesamiento de las mediciones. Luego de esta etapa, cuando las mediciones han sido homogeneizadas y redimensionadas, la siguiente etapa es la corrección de la linea de base.

La corrección de la linea de base corrige un efecto de saturación químico expresado en la medición como un nivel de offset al inicio de la señal. Esta corrección se hace mediante la estimación de la linea que define este efecto de saturación(efecto de linea de base) usando curvas diferenciables(splines). Una vez corregido este efecto, las siguientes etapas en el procesamiento de las mediciones es la alineación de los picos en las mediciones del conjunto de datos. La alineación de picos corrige los errores

[1] La referencia a Plataforma Computacional, en esta tesis, se enfoca en las Laptops y Computadoras de Escritorio con recursos relativos a una Workstation
[2] Dimensión Vectorial
[3] Undersampling / Oversampling / Decimation

de calibración de los instrumentos en el eje de los valores masa a radio, los cuales se ven representados por la heterogeneidad de la coordenadas de los picos en cada una de las mediciones. Para la corrección de este efecto se buscan picos de referencia, los cuales los tienen a presentarse en mediciones que han sido realizadas bajo criterios similares, o a su vez, en la preparación de la mezcla, se suele marcar el compuesto a analizar con compuestos cuya composición es conocida. Las coordenadas de referencia que estos picos sirven para calcular los desplazamientos de las nuevas coordenadas a los que se moverán los valores de intensidades desplazados a alinear. Este desplazamiento se calcula mediante el escalado de los valores de masa a radio cuando la correlación cruzada entre los vectores de medición a corregir y los picos referenciales es máxima, este método preserva la forma de los picos. Finalmente, en el procesamiento de las mediciones, son necesarias las etapas de normalizacion y suavizamiento del ruido.

La normalización de las mediciones soluciona un efecto similar al corrimiento de picos en el eje de los valores de masa a radio pero en esta etapa se trabaja en el eje de las intensidades. El corrimiento y diferencias de los picos en las intensidades de una misma medición se produce por diferencias en la cantidad total de proteínas liberadas en la ionización. Para compensar este efecto se puede normalizar las intensidades de las mediciones a una intensidad relativa de un pico conocido, o a su vez, al igual que en la etapa de alineación, marcar la muestra sometida al espectrómetro y con la información conocida del reactivo de marcación, definir los picos conocidos y normalizar las intensidades de las mediciones con respecto a uno de estos picos de marcación.

El suavizamiento del ruido de las mediciones tiene como objetivo corregir los efectos propios del proceso de adquisición de señales, los ruidos eléctricos, electrónicos, las deficiencias de los sensores, añadiendole los problemas propios de la digitalización de la adquisición de los espectros. Uno de los caminos para atenuar los efectos del ruido en las mediciones de espectrometría de masas es la aplicación de filtros digitales sobre cada una de las mediciones, pero debido a la heterogeneidad de las mediciones, esta alternativa es muy poco practica, ya que se debería calcular un conjunto de coeficientes para cada medición. Para no incurrir en esta práctica, que aunque poco práctica, no es incorrecta, existe la alternativa de usar filtros estadísticos, específicamente en la literatura revisada se hace mención al Filtro de Savitzky–Golay. Este filtro se basa en el calculo de una regresión local polinomial de grado k, donde el resultado del espectro suavizado sera una curva similar al espectro original, donde se eliminan picos de baja significancia estadística, conservando la distribución de probabilidad original de la muestra analizada. Ademas, el filtro de Savitzky–Golay tiene la ventaja de conservar

el ancho de los picos, así como también máximos y mínimos relativos originales de la medición filtrada.

Cuando el conjunto de datos $\mathcal{D} = \{\mathbb{G}_1, \mathbb{G}_2\}$ ha sido procesado a través de todas las etapas mencionadas anteriormente, $\mathcal{D}$ se convierte en un grupo de matrices, formada por un gran grupo de vectores $\{I_{i=1}^{n}\}_{\mathbb{G}_{1,2}}$ y un vector de coordenadas $mz_{\mathbb{G}_{1,2}}$ para $\mathbb{G}_1$ y $\mathbb{G}_2$ correspondientemente. Esta matriz se define en la Ecuación 1.6,

$$\mathfrak{D}_{\mathbb{G}_1,\mathbb{G}_2} = \begin{bmatrix} \{I_{i=1}^{n}\}_{\mathbb{G}_1} \\ \{I_{i=1}^{n}\}_{\mathbb{G}_2} \\ mz_{\mathbb{G}_{1,2}} \end{bmatrix} \tag{1.6}$$

1.4 SELECCIÓN DE CARACTERÍSTICAS

La selección de características es la etapa central de esta investigación, donde lo que se busca es discriminar, cuales son los valores de intensidades $\{I_{i=1}^{n}\}_{\mathbb{G}_{1,2}}$ mínimos para describir a los conjuntos $\mathbb{G}_1$ y $\mathbb{G}_2$. Este proceso se puede abordar desde dos enfoques, la selección de características en base a determinadas parámetros descriptivos(estadísticos) o a su vez a la generación de características en base a criterios geométrico - estadísticos. En el primer caso, el proceso se basa en analizar el conjunto de datos y en base a ciertos criterios ir descartando las muestras del conjunto que no lo cumplan, reduciendo así, el conjunto original, al mínimo, mediante el cual se lo pueda describir. En el caso del conjunto $\mathfrak{D}_{\mathbb{G}_1,\mathbb{G}_2}$, aplicar este método, daría como resultado un nuevo subconjunto $\mathscr{D}$ tal que $\mathscr{D} \in \mathfrak{D}$ y la dimensión $dim(\mathscr{D})$ sea mucho menor que la $dim(\mathfrak{D})$.

En el caso de la generación de características en base a criterios geométrico - estadísticos, el proceso es, mediante los datos disponibles generar nuevos valores o conjuntos de datos, lo cuales suelen suelen presentar niveles de independencia medidos en función de la correlación de estos. La idea en este tipo de métodos es generar conjuntos de datos linealmente independientes, donde para el caso de $\mathfrak{D}_{\mathbb{G}_1,\mathbb{G}_2}$, se produzca un nuevo conjunto $\mathscr{D}$ cuyos valores no pertenecen directamente al conjunto $\mathfrak{D}$, sino que han sido generados mediante operaciones matemáticas tomando los valores numéricos de $\mathbb{G}_1$ y $\mathbb{G}_2$ para producirlos. Los nuevos valores de este conjunto $\mathscr{D}$ son al igual que para el primer caso de una dimensión menor a la de $\mathfrak{D}$.

El nuevo conjunto, luego de la selección de características quedaría definido por la Ecuación 1.7, donde $\mathscr{D}_{\mathscr{G}_1,\mathscr{G}_2}$ es el nuevo conjunto de datos dimensionalmente reducido

con $\mathscr{G}_1$ y $\mathscr{G}_2$ como los grupos de análisis, que contienen las mediciones de las muestras analizadas en el espectrómetro.

$$\mathscr{D}_{\{\mathscr{G}_1,\mathscr{G}_2\}} = \{\mathscr{G}_1, \mathscr{G}_2\} \tag{1.7}$$

La Ecuación 1.7 representada en su equivalente matricial se muestra en la Figura 1.6. Las columnas en color rojo representan las características que se escogerán en el proceso de selección de características, al lado izquierdo, el conjunto de datos original y al lado derecho, el conjunto de datos reducido. Las columnas seleccionadas estan formadas por las intensidades que definirán las zonas de biomarcación.

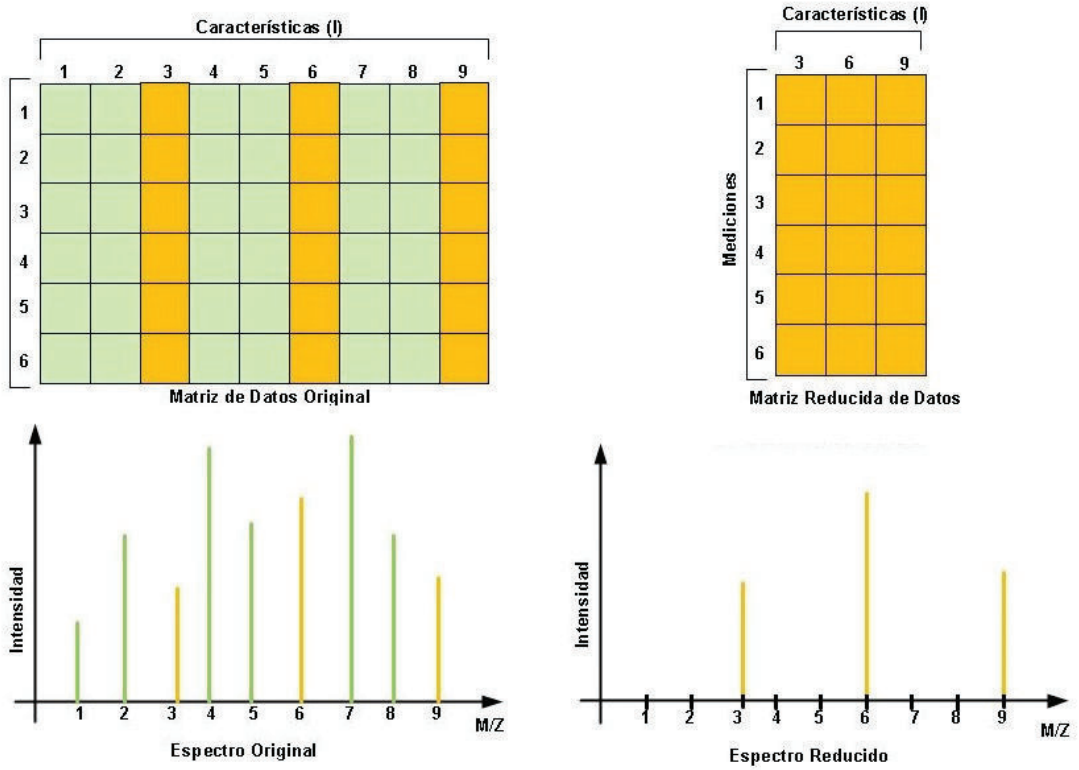


Figura 1.6.: Selección de Características en su Representación Matricial

La selección de características es una necesidad en la exploración de datos de espectrometría de masas, ya sea por motivos de reducción dimensional, o a su vez, por la identificacion de patrones para la discriminación intergrupal. Existen tres grupos de metodologías de selección de características: los filtros, los métodos *wrapper* y los métodos embebidos. Los filtros se dividen en univariantes y multivariantes. Los univariantes no analizan la dependencia integrupal e ignoran la interacción con los clasificadores, en cambio los multivariantes son mas complejos para ser escalados a clasificadores multiclase, son mas lentos que los univariantes y ignoran las interacciones con

algoritmos de clasificación. Entre las principales técnicas aplicadas como filtros para la selección de características están la prueba de $t - Student$, $wilcoxonRankSum$, $F - test$, χ^2, distancia euclideana, entre otras.

Los métodos *wrapper* se dividen en determinísticos y randómicos. Los métodos determinísticos son simples de implementar, interactuan con el clasificador y son menos exigentes computacionalmente hablando que los filtros debido a su naturaleza iterativa, lo cual los vuelve mas sensibles de caer en *overfitting*, soluciones discontinuas de gradiente y *underfitting*. El rendimiento de estos métodos depende del clasificador utilizado. Los ejemplos mas representativos de este tipo de métodos son: la búsqueda secuencial hacia adelante, la eliminación secuencial hacia atrás, *plus q take-away r*, búsqueda *beam*, etc. Finalmente los métodos embebidos son los mas complejos de esta lista, siendo similares a los métodos *wrapper*, los métodos embebidos interactuan con el clasificador, son mas eficientes computacionalmente que sus dos antecesores y su rendimiento depende del clasificador usado. Ejemplos de este tipo de métodos son: los arboles de decisión, clasificadores de *naive bayes* y *support vector machines*.

En esta tesis se hará un especial enfoque al uso de los filtros estadísticos, debido a la simplicidad de entenderlos e implementarlos desde el punto de vista de la codificación en programas. Como se había mencionado anteriormente, el objetivo de esta tesis, es desarrollar una herramienta informática que no necesite de requisitos computacionales excesivamente costosos. La forma de disminuir los tiempos de procesamiento y conseguir estos objetivos es dividiendo las etapas de la minería de datos en fases independientes, en donde la etapa actual entregue los datos a la siguiente etapa hasta conseguir los resultados finales.

1.5 TÉCNICAS DE VALIDACIÓN DE RESULTADOS

Una vez realizada la etapa de selección de características con el conjunto $\mathscr{D}_{\{\mathscr{G}_1,\mathscr{G}_2\}}$, definido en la Ecuación 1.7, la última etapa en la minería de datos para conjuntos de mediciones de espectrometría de masas es la validación de cuan correctas son estas características. Esta etapa, si bien es complementaria al proceso, es la mas crucial de todas. Hay que hacer hincapié que hasta ahora se ha visualizado el proceso solamente como una aplicación de los algoritmos a los valores numéricos de los conjuntos, donde los resultados, serán, nuevamente, conjuntos de valores numéricos. Pero, en aplicaciones, donde la vida de seres humanos esta en juego, es crucial, saber cuan correctos son esos resultados. Entran en escena entonces, dos nuevos conceptos, los falsos positivos y los falsos negativos. Los falsos positivos conceptualmente son

aquellos resultados que siendo incorrectos, al ser validados, se presentan como correctos elevando el porcentaje de certeza del sistema y los falsos negativos, en caso contrario, aquellos que siendo incorrectos, disminuyen el rendimiento del sistema.

En el diagnóstico médico los falsos positivos son tan graves como los falsos negativos. Haciendo referencia al diagnostico del cáncer, para un falso positivo, el decirle a una persona que tiene cáncer y que debe realizarse una cirugía, ya sea exploratoria, o extractiva, sin que esta sea necesaria, resulta un problema muy grave. En el caso contrario, un falso negativo, se refleja en la situación donde el diagnóstico se de como la ausencia de cáncer, existiendo este en el paciente, y que además, se le diga que no debe realizarse ningún tipo de operación, ni tratamiento.

La etapa de validación del conjunto $\mathscr{D}_{\{\mathscr{G}_1,\mathscr{G}_2\}}$ se basa en la modelación de un clasificador de patrones $h\{\mathscr{E},\mathscr{P}\}$, el cual se construye a partir de subconjuntos del conjunto $\mathscr{D}_{\{\mathscr{G}_1,\mathscr{G}_2\}}$ para las etapas de entrenamiento($\mathscr{E}_{\{\mathscr{G}_1,\mathscr{G}_2\}}$) y pruebas($\mathscr{P}_{\{\mathscr{G}_1,\mathscr{G}_2\}}$). Si el clasificador $h\{\mathscr{E},\mathscr{P}\}$ posee un rendimiento adecuado, a las pruebas de clasificación, rechazando los efectos de *overfitting* y *underfitting*, la conclusión final es, que las selección de características es correcta. La certeza de estas características se define probabilisticamente, es decir, el definirlas como correctas, implica un porcentaje de certeza. Si el clasificador alcanza un rendimiento del $96\,\%$, la interpretación sera que existe un 96 de probabilidades de certeza, de que, las zonas definidas por las características seleccionadas son las correctas, además, de un $4\,\%$ de probabilidades de que las zonas de información(regiones de biomarcación) no sean las correctas.

1.6 APLICACIONES Y DESARROLLOS ACTUALES

El diagnóstico y tratamiento de enfermedades como el cáncer en etapa temprana es uno de los retos mas grandes que enfrenta la medicina actualmente. En las ultimas décadas el uso de plataformas computacionales ha hecho posible la búsqueda de una solución a este problema desde nuevas lineas de investigación. La incursión de nuevas tecnologías ha permitido el disponer hoy en día de información sobre la cual se aplican algoritmos en búsqueda de aspectos que definan y caractericen fenómenos médicos, estados biológicos, patológicos y ayuden a mejorar la salud de las personas. En este tipo de aplicaciones, el análisis de datos de mediciones de espectrometría de masas usando algoritmos estadísticos como herramientas de minería de datos en conjunto con *Ensemble Learning* para validar estos resultados representan nuevas ideas en conseguir estos objetivos.

Existen desarrollos previos donde se usan este tipo de herramientas algoritmos, sin embargo, el costo computacional y la complejidad de las implementaciones de estos trabajos opacan su popularidad. Una de las primeras limitaciones de estos desarrollos es trabajar en forma autónoma con la dimensionalidad de los datos sin incurrir en modificaciones que las alteren o que creen nuevos valores en función de las mediciones disponibles(PCA, ICA, ANOVA, Correlación). Estos métodos realizan operaciones sobre los valores numéricos de la intensidades y de las relaciones masa a radio de cada una de las mediciones de los conjuntos de datos, con los cuales se busca evitar los problemas de dimensionalidad(demasiados valores) generando nuevos conjuntos de datos linealmente independientes. El problema de este método es el hecho que la información original se ve trastocada, lo cual dificulta poseer resultados precisos en la determinación de patrones de biomarcación. Otra alternativa a este método es usar metodologías heuristicas básicas tomando todo el conjunto de datos evaluando una a una las mediciones en cada una de sus intensidades, mediante un método que pueda determinar cual es la contribución a mejorar el reconocimiento intergrupal, descartando las que degraden este parámetro, a estos métodos se los conoce comúnmente como los algoritmos genéticos. Los algoritmos genéticos al igual que los métodos de reducción dimensional basado en operaciones sobre los datos de las mediciones poseen limitaciones en cuanto a la autosintonización de parámetros de salida, es necesario advertir previamente al algoritmo de cuantos biomarcadores, cuantas características o intensidades son necesarias como resultados a la salida del proceso. La reducción dimensional tiene como objetivo eliminar la información redundante, la cual una vez eliminada, para las aplicaciones abordadas en este proyecto, permite vislumbrar donde están los cambios o mutaciones genéticas, primeros indicios de problemas patológicos relacionados con el cáncer.

Finalmente, la aplicaciones de técnicas de validación han incurrido en el uso de clasificadores como metodologías para la validación de los patrones seleccionados donde la idea es simular escenarios de laboratorio, en los cuales se ha instruido previamente a los laboratoristas(clasificadores), las cualidades de los biomarcadores, antígenos y proteínas que alertan al sistema biológico de los seres humanos sobre la presencia de cáncer. Una vez instruidos estos laboratoristas virtuales, deben seguir un método para encontrar correctamente el grupo de las mediciones de análisis no etiquetadas. La idea es que estos laboratoristas(clasificadores) busquen el estado patológico de las muestras analizadas tomando como métricas las intensidades obtenidas en el proceso de reducción dimensional, o selección de características, para optimizar los procesos de calculo, tiempo de resultados y evitar casos de falsos positivos o falsos

negativos. La formas tradicionales de validar estos procesos es a través de la medición del rendimiento de clasificación usando mediciones previamente etiquetas sometidas a un sistema clasificador, para luego comparar las etiquetas que este obtuvo como resultados con las etiquetas originales, metodologías adicionales a esta son, la *cross-validation* y la evaluación de la *Receiver Operating Characteristic Curve* o curva ROC. Existen trabajos en la bibliografia consultada los cuales hacen uso de clasificadores simples basados en *Support Vector Machines*, *k-nearest neighbors*, *Artificial Neural Networks*, *Decision tree learning*, entre otros. Este tipo de tecnicas tiene la particularidad de posseer baja capacidad de generalización en aplicaciones complejas. Dicho de otra forma, los datos a clasificar deben cumplir con ciertas características para poder aplicarlos en este tipo de algoritmos. Esta limitación entra en el mismo paradigma de la reducción dimensional, donde es necesario tratar los datos para poder adaptarlos a los algoritmos. El caso de las *Artificial Neural Networks* es bastante especial, en comparación a los otros algoritmos citados, ya que esta técnica posee una resistencia superior a los fenómenos de *Overfitting* y *Underfitting*, las técnicas restantes, deben ser por tanto evaluadas muy cuidadosamente antes de poder afirmar la certeza o no de los resultados. *Support Vector Machines* y *k-nearest neighbors* son computacionalmente costosos y de convergencia lenta(encontrar soluciones), *Decision tree learning* por tanto es de rápida convergencia, lamentablemente esta rápida convergencia lo vuelve impreciso para conjuntos masivos de datos. Una vez aplicada la validación de datos, los resultados finales(conjuntos de puntos de mz vs I) de todo este proceso servirán para la definición de biomarcadores por parte de profesiones en biología o medicina, los cuales finalmente podrán ser aplicados en la producción de nuevos fármacos, caracterización de tratamientos y diagnósticos tempranos del cáncer.

CAPÍTULO 2

Descripción del Algoritmo Implementado

En el presente capítulo se realiza una descripción de la implementación del algoritmo propuesto en esta tesis detallando el flujo de datos, paso a paso, etapa a etapa a fin de visualizar y entender como se realiza la extracción de biomarcadores de las mediciones de espectrometría de masas en conjuntos de datos de grupos de análisis cáncer vs control.

El algoritmo propuesto en este proyecto de titulación esta basado en el principio de la búsqueda *metaheurística* de soluciones a la caracterización de mutaciones en espectros de masas. Específicamente la idea radica en darle a conocer al algoritmo ciertas pistas para encontrar el camino a la solución en base a aprendizaje supervisado. Dicho aprendizaje a su vez se ve reforzado por las técnicas de *Boosting Learning*, las cuales rechazan de forma considerable los efectos de *Underfitting* y *Overfitting*.

En su etapa inicial, el algoritmo dispone del conjunto de entrada $\mathcal{D}_{\mathcal{X},\mathcal{Y}} = \{\mathbb{G}_1, \mathbb{G}_2\}$, el cual esta formado por una serie de archivos en texto plano[4], donde se guardan los valores numéricos de las mediciones de espectrometría de masas(intensidades y valores de las relaciones masa a radio). Estos archivos son cargados luego en una plataforma computacional, donde se van aplicando secuencialmente las etapas de procesamiento, selección de características y validación de resultados.

[4]texto sin formato

La definición en pseudocódigo de las etapas del proceso propuesto por este trabajo de titulación se presenta a continuación en el Algoritmo 1.

Algoritmo 1: Algoritmo de Búsqueda de Zonas de Biomarcación

Input: Conjunto $\mathcal{D}_{\mathcal{X},\mathcal{Y}} = \{\mathbb{G}_1, \mathbb{G}_2\}$

Output: Regiones de Biomarcación, indices $idx_{\{mz_i, I_j\}}$

1. Cargar Datos
2. **for** $\{i, j\} < \{x, y\}$ **do**
3. `MatlabMatrix(i,j)` $\leftarrow \{\mathbb{G}_1, \mathbb{G}_2\}\{i, j\}$
4. **return** *MatlabMatrix*
5. 2. Procesamiento de Mediciones
6. **for** $\{i, j\} < \{x, y\}$ **do**
7. `MedicionesRemuestreadas(i,j)` $\leftarrow$ `msresample(MatlabMatrix(i,j),'freq');`
8. `MedicionesLinBase(i,j)` $\leftarrow$ `msbackadj(MatlabMatrix(i,j),'method');`
9. `MedicionesAlineadas(i,j)` $\leftarrow$ `msalign(MatlabMatrix(i,j),'PeakRef';)`
10. `MedicionesNorm(i,j)` $\leftarrow$ `msnorm(MatlabMatrix(i,j),'fact');`
11. `MedicionesGolay(i,j)` $\leftarrow$ `mssgolay(MatlabMatrix(i,j),'degree');`
12. `MedicionesProcesadas(i,j)` $\leftarrow$ `MedicionesGolay(i,j);`
13. **return** *MedicionesProcesadas*
14. 3. Seleccion de Caracteristicas discriminantes
15. Crear Subconjuntos entrenamiento, pruebas, validación;
16. **for** $\{i, j\} < \{x, y\}$ **do**
17. `[pt,t-Student(j)]` $\leftarrow$ `ttest(transpose(MedicionesProcesadas(j:,)),'equal');`
18. `[pu,U-test(j)]` $\leftarrow$ `utest(transpose(MedicionesProcesadas(j:,)),'method');`
19. `[pch,`χ^2`(j)]` $\leftarrow$ `chi2(transpose(MedicionesProcesadas(j:,)),'godfit');`
20. Estimar $FEP(pt, pu, pch)$; Funcion Empirica de Probabilidad;
21. Estimar Subconjunto de Mutaciones; $FEP(pt, pu, pch)$; Probabilidades $\ll 0$
22. Ordenar Probabilidades y extraer indices; `indx`$_{\{x,y\}}$`;` Evaluar Overfitting;
23. **for** $\{:, j\} < \{:, y\}$ **do**
24. `Classificador` $\leftarrow$ `modelar(MedicionesProcesadas(:,train),labels(,train))`
25. `error(j)` $\leftarrow$ `error(Classificador)`
26. Parar si `error(j+1)<error(j)`
27. Seleccionar las $j - esimas$ características
28. Eliminar Características redundantes;
29. Filtrar por Distancia Modificada de Mahalanobis;
30. Seleccionar $idx_{\{mz_i, I_j\}}$ de caracteristicas reducidas;
31.
32. **return** $idx_{\{mz_i, I_j\}}$
33. 4. Validación de Resultados

Algoritmo 2: ALGORITMO DE BÚSQUEDA DE ZONAS DE BIOMARCACIÓN(CONT...)

Input: Conjunto $\mathcal{D}_{\mathcal{X},\mathcal{Y}} = \{\mathbb{G}_1, \mathbb{G}_2\}$

Output: Regiones de Biomarcación, indices $idx_{\{mz_i, I_j\}}$

4. Validación de Resultados
4.1 Modelar Grupos en Crossvalidation(10);
4.2 Modelar Classificador en Tree Classification con Adaboost.M1;
4.3 Evaluar el Error de Clasificacion;
4.4 Estimar la Curva ROC;
4.5 Clasificar muestras del Conjunto Validación;
4.6 Estimar Error de Clasificación de Conjunto Validación;
5. El error es menor que ϵ?
5.1 Si? Guardar Resultados `fichero.txt`
5.2 No? Reevaluar Selección de Características;
5.2.1 Reevaluar Procesamiento de Mediciones;
6. Fin del Procesamiento;

La metodología presentada anteriormente muestra como **contribución** el tener una naturaleza semisupervisada. En la literatura consultada muchos de los trabajos hacen referencia a una función objetivo(target function) sobre la cual se trabajara y mediante técnicas de optimización se buscará cumplir con un número mínimo de biomarcadores(pares $\{mz_i, I_j\}$), sin embargo esto no debería ser controlable de manera externa, ya que el objetivo como tal es encontrar los pares $\{mz_i, I_j\}$ óptimos que produzcan un alto rendimiento a la hora de hacer discriminación intergrupal(clasificación de muestras). El algoritmo que se presenta en esta tesis al contrario de esto, no asume un número mínimo de biomarcadores, sino que define zonas cuyos pares son estadisticamente significativos en 3 etapas, la primera mediante la aplicación de las pruebas estadísticas, la segunda al buscar el punto de inflexión de donde exactamente se producen cambios en la curva de aprendizaje(de naturaleza exponencial negativa), descartando las características consecuentes asumiendo presencia de ruido que degrade la discriminación intergrupal. La tercera parte y parte del núcleo de la contribución de este trabajo es el filtro geométrico basado en la distancia de Mahalanobis, el cual tiene la habilidad de reducir en total las características en factores de 200 a 1 (en promedio), sin que se intervenga directamente en este proceso, de esta forma se tiene mayor certeza de que se obtuvo un resultado mas limpio de influencias externas y mas completo, sin limitaciones ni restricciones impuestas.

Los pasos descritos en el pseudocódigo anterior se muestran en la Figura 2.1 en forma de diagrama de flujo. En el procesamiento se cargan los archivos que contienen las mediciones de las muestras analizadas al banco de memoria de la plataforma computacional utilizada, en este caso, hacia el workspace de Matlab. Una vez aquí, cada una

de las mediciones e procesan secuencialmente en las subetapas de remuestreo, corrección de linea de base, alineación, normalización y suavizado del ruido, los detalle de estos procesos se describen en el Capitulo 3. Una vez todas las mediciones del conjunto de datos han sido procesadas, el siguiente paso es la selección de características. En esta etapa, entran en escena dos grupos de datos, que se han originado en la partición del conjunto original de datos en 3 subconjuntos, uno de entrenamiento(train), otro de pruebas(test) y un tercero, el que no participara en ningún proceso de minería de datos, denominado pruebas externas. La selección de características entregara como resultados pares numéricos de relaciones masa a radio con su correspondiente valor de intensidad sin alterar los datos. En esta etapa se realizan análisis de datos, no se generan nuevos datos. Este proceso se aborda bajo la hipótesis de que las muestras analizadas deben poseer características similares, analizando las mediciones como vectores, la medición de la muestra no patológica es linealmente dependiente de la muestra en estado patológico. El trabajo del filtro estadístico propuesto tiene como objetivo eliminar la información redundante dejando libre los valores de las intensidades que han cambiado del un grupo de análisis al otro, estos cambios a su vez representas las mutaciones, los primeros indicios de la presencia del cáncer.

La validación de los resultados obtenidos en la selección de características representa el ultimo paso en todo este proceso. Afirmar que un conjunto de resultados es correcto o a su vez representa información estadísticamente significativa para poder realizar un diagnóstico no representa tarea fácil teniendo en cuenta las aplicaciones en las que estos resultados serán utilizados. El proceso de prueba de la certeza de los resultados obtenidos se realiza mediante 2 pruebas cruzadas a doble ciego, la *crossvalidation* y las pruebas con muestras externas. Para la *crossvalidation* se realizo una evaluación exhaustiva en 10 casos(este número puede variar dependiendo de los lineamientos planteados de la investigacion) aleatorios de división de conjuntos de entrenamiento y pruebas, cuyos errores de clasificación se promedian y en función de estos se evaluá una curva de aprendizaje. Esta curva de aprendizaje es el primer escalón a subir por el método propuesto ya que de presentar características inestables(normalmente debe ser exponencial decreciente), esto representara que las características seleccionadas no son las correctas, de no haber evidencia de inestabilidad(overfitting o underfitting), el siguiente paso es someter el conjunto de validación(sin etiquetas) a un clasificador genérico de árbol(tree Desicion) en Adaboost y evaluar el número de aciertos como segunda métrica de evaluación.

El error promedio del sistema viene dado por $E = \frac{1}{K}\Sigma_{i=1}^{K} E_i$, donde k representa el número de iteraciones o casos de análisis que se crossvalidarán y E_i el error de cada

$i-esimo$ caso.Si este error supera el $10\,\%$ en promedio de las dos pruebas externas, se deben recalcular el subconjunto de características y volver a evaluar su rendimiento. Hay que tener en cuenta adicionalmente a estos aspectos la influencia directa de la etapa de procesamiento, la cual de no se realizada adecuadamente con los parámetros correctos de sintonización, existirán errores como alta presencia de ruido, picos exógenos, discontinuidates, etc. Las pruebas realizadas en las simulaciones produjeron rendimientos altos, en tres conjuntos de datos independientes, cuya dificultan de discriminación ademas es de un grado alto. En dos de los tres ademas se realizo la implementación del procesamiento desde cero, con sintonización de parámetros controlada por el rendimiento a la salida de los algoritmos evitando tener zonas muertas o discontinuidades en los valores de las intensidades de las muestras.

Los resultados finales se exportan en un formato $.txt$ en texto plano con codificación ascii facilitando así el compartir las zonas de biomarcación encontradas independientemente de los sistemas operativos en que se analizen. Todos estos resultados y código pueden descargarse del sitio web del proyecto, cuyo dirección web y pantallas de inicio se muestran en el Anexo C. Este sitio web esta alojado en git (https://git-scm.com/), un sitio web dedicado al desarrollo de proyectos de software en colaboración. La idea de alojar ahıel código desarrollado y los resultados obtenidos tiene como objetivo seguir evaluando el proyecto desarrollado a fin de poder pulir los detalles y a futuro exponerlo como un método formal de diagnóstico.

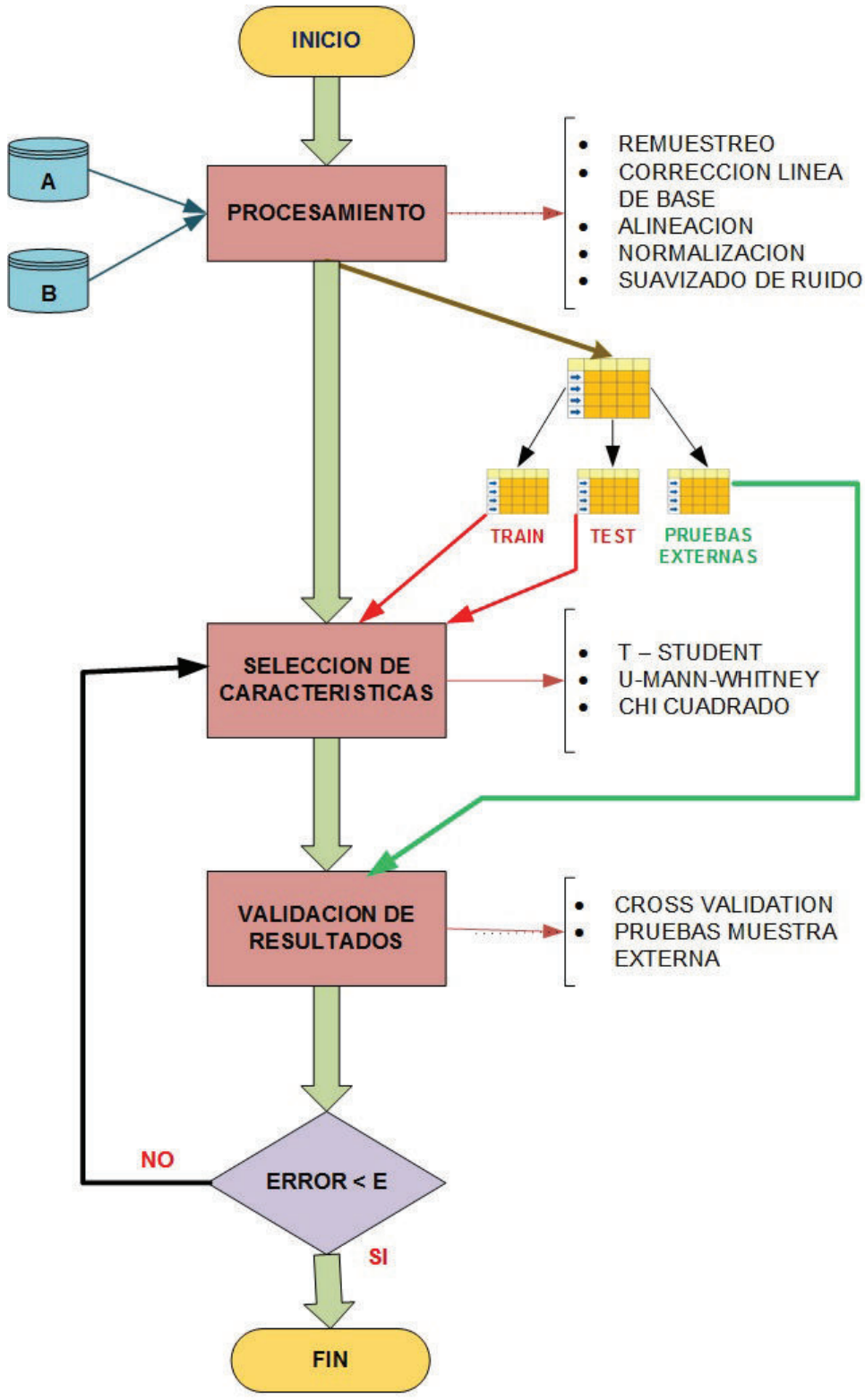


Figura 2.1.: Diagrama de Flujo del Algoritmo Propuesto

2.1 PLATAFORMA COMPUTACIONAL

La plataforma computacional usada para la implementación de este algoritmo se baso en el uso de scripts personalizados en *Matlab 2008* y sus *Bioinformatics Toolbox* y *Statistics and Machine Learning Toolbox*. Debido a la alta exigencia a nivel de cálculos numéricos, no fue viable la creación de interfaces gráficas o incluir detalles que hagan mas amigable la experimentación con el algoritmo propuesto en esta tesis.

Los recursos computacionales con los que se trabajo para la simulación del algoritmo propuesto son:

- Procesamiento: *Intel Core i7 (Nehalem)* de cuarta generación, con una memoria Cache de $4\ [Mb]$
- Almacenamiento: $1[TB]$
- Memoria RAM: $16[GB]$
- Frecuencia de Bus Frontal: $1600[Mhz]$

La cantidad de Memoria RAM disponible resulto ser el factor decisivo a la hora de echar a andar las simulaciones, ya que en los primeros intentos, $4[GB]$ en RAM resultaron insuficientes, con $8[GB]$, el ordenador se sentía ralentizado, ya con $16[GB]$, las operaciones en la laptop se realizaban de manera normal.

Los tiempos de Procesamiento en la ejecución de las operaciones fueron optimizados cargando los valores numéricos en espacios de memoria previamente reservados. Esta técnica de optimización es muy común en Matl

ab donde la preubicación de memoria disminuye los tiempos de procesamiento considerablemente. La reserva de espacios de memoria se la realiza con la definición de vectores de ceros de dimensión igual al de datos a cargar, es decir, sea `data`, el vector que contiene la información de la medición $i-esima$ formada por los vectores $\{I_{i=1}^{n}, mz_{i=1}^{n}\}$, el proceso de reserva de memoria se realiza creando el vector `data=zeros(1,n)`. La primera versión del vector `data` sera por tanto un conjunto de zeros, con ubicaciones definidas en el espacio de memoria, en las cuales se guardaran los valores numéricos de las intensidades y relaciones masa a carga de cada una de las mediciones de los grupos de análisis del conjunto de datos. Adicionalmente el proceso de minería se dividió en tres etapas: el procesamiento, la selección de características y la validación de resultados.

La etapa del procesamiento inicia con la carga de datos $\mathcal{D}_{\mathcal{X},\mathcal{Y}} = \{\mathbb{G}_1, \mathbb{G}_2\}$ en memoria, luego estos datos son homogeneizados, mediante la etapa de remuestreo, alineados, normalizados y filtrados, todos estos resultados son guardados en una matriz de valores, la cual sera el punto de partida de la etapa de selección de características, donde los resultados de la selección de características se guardaran nuevamente en una matriz de resultados, con los cuales se realizara la etapa de validación de resultados.

2.2 DATOS DE ENTRADA

El gran interés que despierta la investigación computacional aplicada a la medicina ha logrado que haya gran apertura en la disponibilidad de datos en los diferentes websites de los centros de investigación y universidades a nivel mundial. Los datos para las simulaciones del algoritmos implementado en esta tesis fueron tomados de UCI Machine Learning Repository [5] y de la Clinical Proteomics Program Databank [6]. De estas dos bases de datos se obtuvieron los siguientes conjuntos:

1. Ovarian cancer case vs. high-risk control
2. Premalignant Pancreatic Cancer
3. Arcene

Cada uno de los conjuntos anteriormente mencionados disponen de las mediciones de las muestras en archivos independientes, es decir, por cada una de las muestras, los valores numéricos de estas están guardadas en forma de texto. Debido a que los conjuntos de datos usados están en formatos diferentes(`.csv` y `.txt`), se crearon dos secuencias de programación para cada uno de estos casos, dichas secuencias se muestran en el Codigo 2.1.

Código: 2.1: Programación para cargar los Datos en Memoria

```
% CARGAR ARCHIVOS .txt
% PARAMETROS:
%   datos = estructura donde guardaran m/z e intensidad
%   n = numero de archivos .txt a cargar
d = dir(['*.txt']);
n = length(d);
```

[5] https://archive.ics.uci.edu/ml/datasets/Arcene
[6] http://home.ccr.cancer.gov/ncifdaproteomics/ppatterns.asp

```
datos{1} = [0 0];
namesC{1} = '';
for i=1:n
    datos{i} = (load([d(i).name]));
    name = d(i).name;
    a = regexp(name, '.txt');
    namesC{i} = name(1:a-1);
    if i == n
        fprintf([blanks(1) '%d\n' blanks(1) '\n'],i)
    else
        if (( 12 * round(double(i)/12) == i ))
            fprintf([blanks(1) '%d' blanks(1) '\n'],i)
        else
            fprintf([blanks(1) '%d' blanks(1)],i)
        end
    end
end
% CARGAR ARCHIVOS .csv
% PARAMETROS:
%   datos = estructura donde guardaran m/z e intensidad
%   n = numero de archivos .csv a cargar
d = dir(['*.csv']);
n = length(d);
datos{1} = [0 0];
    for i=1:n
    datos{i} = importdata([d(i).name]);
    fprintf('%s\n',d(i).name)
    end
end
```

2.3 COMPUTACIÓN PARALELA

En la carga de los datos en memoria entra en escena una de las primeras limitaciones practicas del procesamiento masivo de datos, la velocidad de computación. En este caso, al trabajar en un microprocesador de 4 núcleos físicos, con capacidad para manejar 8 hilos en paralelo, dicha limitación se ve atenuada radicalmente. La compu-

tación paralela aplicada al algoritmo propuesto significa una enorme ventaja a la hora de manejar eficiencia de tiempos en la obtención de resultados.

Una de las primeras subetapas del procesamiento de mediciones es el remuestreo, la cual se realiza de forma secuencial, donde entra en escena una situación típica que puede ser optimizada con procesamiento paralelo. En Matlab existen dos funciones dedicadas al procesamiento en paralelo `parfor` y `matlabpool`. La primera hace lazos `for` tipicos pero dedicados en cada uno de los núcleos de la plataforma computacional, la segunda enciende todos los núcleos disponibles haciendo un balanceo de carga. Tipicamente los procesados de las plataformas comerciales cuentan con la opción de *Turboboosting* en sus microprocesadores lo que permite hacer un *overclocking* sobre la frecuencia natural de trabajo de los microprocesadores lo que eleva su rendimiento, siendo esta característica una herramienta natural de las plataformas computacionales comerciales para mejorar los tiempos de obtención de resultados.

Los datos usados para las simulaciones resultan ser exigentes a la hora del procesamiento debido al formato usado por estos y a la forma como son leídos en el Matlab. Desde el punto de vista de la presición numérica traducida a bits, el punto flotante con el que se trabajo representa un alto grado a la hora de tratar elementos numéricos con parte decimal, sin embargo, esta precisión puede ser controlada a una métrica necesaria lo que resulta en consecuencia como una reducción en la carga computacional, lo cual en conjunto a la computación paralela optimiza el tiempo total de procesamiento.

Un factor adicional que entra en escena en la implementación del algoritmo propuesto en este proyecto de titulación es la producción de números aleatorios de manera controlada. Estos números si bien se conocen como números aleatorios, su producción suele darse por medio de algoritmos que necesitan de una semilla para producirlos. La semilla como tal, representa un valor numérico de entrada, el cual a su vez produce una salida controlada cambiando el valor de entrada por un valor no conocido. Siempre y cuando la entrada sea la misma, el valor aleatorio producido sera el mismo, lo cual permite realizar experimentos controlados al momento de la selección de las mediciones que participaran en la división de subconjuntos para el entrenamiento, pruebas y validación usando muestras externas. El generador de números aleatorios usado esta basado en la serie de Fibonacci siguiendo la Ecuación 2.1:

$$x_n = (x_{n-j} + x_{n-k}) \ \times mod(m) \ \forall \ 0 < j < k; \ m = 2^M; \ M = 32, 64 \qquad (2.1)$$

donde:

- x_n es el numero aleatorio producido por el algoritmo.
- m es la semilla de entrada del algoritmo.
- M es el numero de combinaciones posibles, el cual depende de la palabra del microprocesador usado(numero de bits), x86(32 bits), x64(64 bits).

En el proyecto de tesis se uso la instrucción `rng(semilla,'multFibonacci');` debido a las instrucciones que están corriendo en varios núcleos como herramientas aleatorias de selección de mediciones usadas. El prefijo `'multFibonacci'` activa la generación de números aleatorios en cada uno de los núcleos para que la selección se realice de manera simultanea en varios grupos de mediciones a la vez.

2.4 EL LENGUAJE DE PROGRAMACIÓN

El presente proyecto de titulación esta implementado totalmente en scripts de Matlab. El lenguaje de programación Matlab esta categorizado en un nivel muy alto en comparación a los otros lenguajes como C o C++(Lenguajes de Nivel Medio), lo cual si bien es una ventaja, a la hora de implementar herramientas informáticas definitivas compiladas representa una desventaja. Matlab es una herramienta de Cálculo Numérico orientado a las ciencias e ingeniería, dichas características deben por tanto facilitar el cálculo, mas no el rendimiento. El hecho de que Matlab sea un lenguaje interpretado sobre una capa de Java disminuye aun mas su rendimiento. En la actualidad, el desarrollo de algoritmos en Matlab se ha visto mejorado mediante la aplicación de computación paralela, así como también del uso de procesadores gráficos (GPU), sin embargo, a pesar de que usar estas tecnologías disminuye el tiempo de procesamiento, los rendimientos alcanzados no igualan tan siquiera a la implementación de algoritmos en lenguajes dedicados como C o C++. Para obtener un rendimiento mas real del algoritmo propuesto se debería diseñar una herramienta informática en base a lineamientos de software, los cuales deben describir el método a usar y definir el escenario en el cual funcionara la metodología del algoritmo propuesto, estas actividades quedan abiertas para próximos proyectos, ya que no entran en el alcance de esta tesis.

2.5 HERRAMIENTAS INFORMÁTICAS UTILIZADAS

Las herramientas usadas para la implementación de los scripts de las simulaciones del algoritmo propuesto en esta tesis son: Bioinformatics Toolbox, Pattern Recognition

Toolbox y el Statistics and Machine Learning Toolbox. Estos *toolboxes* son herramientas adicionales a la plataforma de Matlab, las cuales interaccionan entre dependiendo de las aplicaciones. En las simulaciones realizadas se hacen uso secuencialmente del Bioinformatics Toolbox para el procesamiento, y del Pattern Recognition Toolbox y el Statistics and Machine Learning Toolbox para la selección de características y validación de resultados.

Las funciones utilizadas en los scripts son:

- Procesamiento de Mediciones
  - `importdata`
    Carga datos desde un archivo
    Sintaxis: `A = importdata(filename)`
  - `msresample`
    Remuestrea mediciónes de espectrometria de masas
    Sintaxis: `[Xout, Intensitiesout] = msresample(X, Intensities, N)`
  - `msbackadj`
    Remueve la linea de base de mediciónes de espectrometria de masas
    Sintaxis: `Yout = msbackadj(X, Intensities)`
  - `msalign`
    Alinea mediciones en funcion de picos de referencia
    Sintaxis: `IntensitiesOut = msalign(X, Intensities, RefX)`
  - `msnorm`
    Normaliza mediciones de espectrometria de masas
    Sintaxis: `Yout = msnorm(X, Intensities)`
  - `mssgolay`
    Suavizamiento de Ruido usando el Filtro de Savitzky–Golay
    Sintaxis: `Yout = mssgolay(X, Intensities)`
- Selección de Características Discriminantes
  - `ttest2`
    Prueba t-Student para dos muestras de datos
    Sintaxis: `[h,p,ci,stats] = ttest2(x,y,Name,Value)`

- `ranksum`

  Prueba de Mann-Whitney U-test

  Sintaxis: `[p,h,stats] = ranksum(x,y)`

- `chi2gof`

  Prueba χ^2 ajuste de curvas

  Sintaxis: `[h,p,stats] = chi2gof(x,Name,Value)`

- `fitensemble`

  Modela Clasificadores Sepervisados usando Boosting Learning

  Sintaxis:

```
Ensemble = fitensemble(TBL, ResponseVarName, ...
    Method, NLearn, Learners)
```

- `predict`

  Calcula las etiquetas de Salida usando Clasificadores Supervisado

  Sintaxis: `[label,score,cost] = predict(obj,X)`

– Validación de Resultados

- `kfoldLoss`

  Evaluá un clasificador con muestras que no han sido usadas en el entrenamiento

  Sintaxis: `L = kfoldLoss(obj,Name,Value)`

- `perfcurve`

  Estima la curva ROC en función de las Probabilidades a posteriori del clasificador

  Sintaxis:

```
[X,Y,T,AUC,OPTROCPT,SUBY,SUBYNAMES] = ...
    perfcurve(labels,scores,posclass)
```

CAPÍTULO 3

Procesamiento de Mediciones

En este capítulo se describen los detalles teóricos de la etapa del procesamiento de las mediciones de los conjuntos de datos. En una primera instancia se exponen los detalles del proceso de adquisición de datos por medio de los espectrómetros de masas. Luego, una vez adquiridas las mediciones en formato digital, estos datos pasan a plataformas computacionales, en donde se realizan los procesos de homogeneización de muestras y tratamiento digital para la atenuación de los efectos del ruido. Las etapas de procesamiento presentadas son aplicadas en función de las necesidades de cada aplicación, conjunto de mediciones intergrupales y datos adquiridos. En este trabajo de titulación, como proceso genérico de tratamiento de mediciones, se recomienda aplicar de manera secuencial las siguientes etapas: Remuestreo, Corrección de la Linea de Base, Alineación de picos, Normalización de la intensidad, Filtro de Suavizamiento de Savitzky y Golay, cada una de estas etapas es revisada a detalle en las siguientes subsecciones, donde además se anexan tipos y recomendaciones para la sintonización de los algoritmos para la obtención de resultados óptimos.

3.1 ADQUISICIÓN DE MEDICIONES

La adquisición de mediciones se realiza a través del espectrómetro de masas, el cual se encarga de descomponer la muestra analizará en su correspondiente representación de valores en los ejes masa a radio versus intensidad. Las muestras analizadas pueden ser de naturaleza líquida, sólida o gaseosa y su funcionamiento se detalla en cuatro etapas principales: introducción de la muestra, ionización, analizador de masas y detector. A continuación en la Figura 3.1 se diagrama de flujo que esquematiza la interacción de estas etapas hasta la obtención de los valores numéricos en formato digital de la medición.

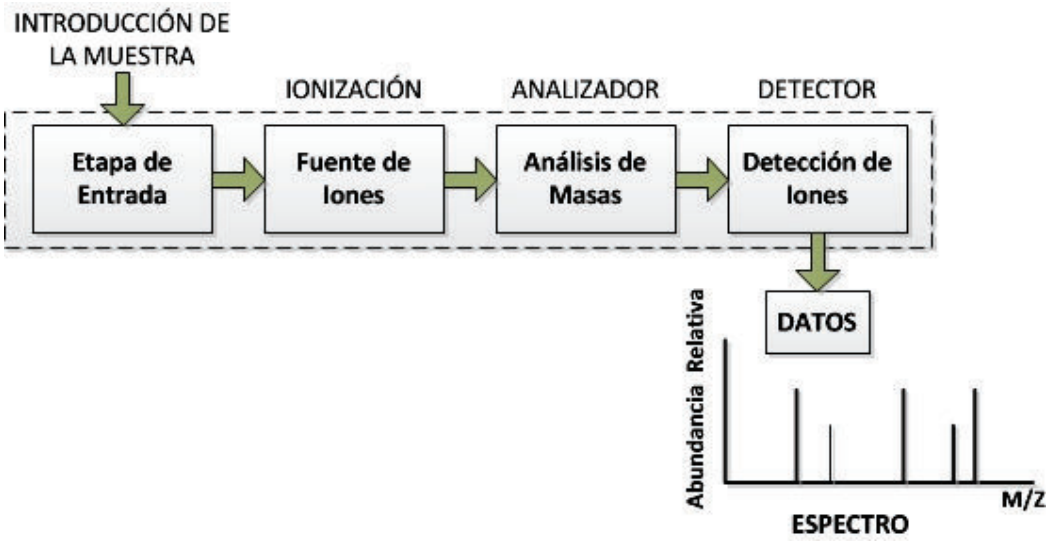


Figura 3.1.: Etapas de un Espectrómetro de Masas

3.1.1 ETAPA DE INTRODUCCIÓN DE MUESTRAS

El sistema de entrada introduce pequeñas muestras en el instrumento a una cámara de volatilización en el vacio con una fuente de calor que cambie el estado de la muestra sea líquido o sólido a estado gaseoso.

El gran limitante de la etapa de introducción de muestras es la vaporización, ya que existen moléculas sensibles al calor que se evaporan fácilmente y externamente produciendo fragmentación y pérdida de información. En la Figura 3.2 se muestra un esquema de como la muestra es calentada antes de ser sometida a la fuente de iones, en la siguiente etapa de extracción de los datos.

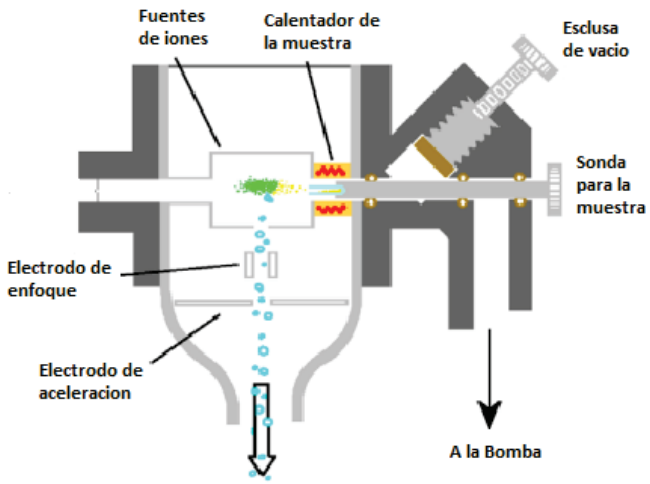


Figura 3.2.: Sistema de Introducción de Muestras

3.1.2 ETAPA DE IONIZACIÓN

En esta etapa la muestra es bombardeada con electrones esto viene determinada por la naturaleza de la muestra y la clase de información que se desea obtener. Existen dos tipos de métodos como Ionización en fase gaseosa en esta clase de ionización primero se volatiliza la muestra para luego ionizarla, se aplica básicamente en compuestos térmicamente estables con pesos moleculares menores a 10^3 Dalton $[Da]$. Por otra parte la Ionización por desorción la muestra se transforma directamente en iones gaseosos, se aplica generalmente a muestras no volátiles y térmicamente inestables con pesos moleculares mayores a 10^5 [Da] y sirven para la aplicación de Biomarcadores.

3.1.3 ANALIZADOR DE MASAS

El analizador de masas tiene dos funciones básicas: separar y enfocar hacia el detector los iones en función de su m/z y dirigir cada ión separado a un punto para cuantificar su abundancia.

A la muestra introducida se aplica un campo eléctrico, el cual obliga a cambiar la velocidad de cada ión de acuerdo a su masa. Posteriormente se aplica un campo magnético perpendicular al movimiento de los iones obligándoles a cambiar su trayectoria, cada ión describirá una única trayectoria y así podrá ser detectado, clasificado y dirigido hacia el detector como se muestra en la siguiente Figura 3.3.

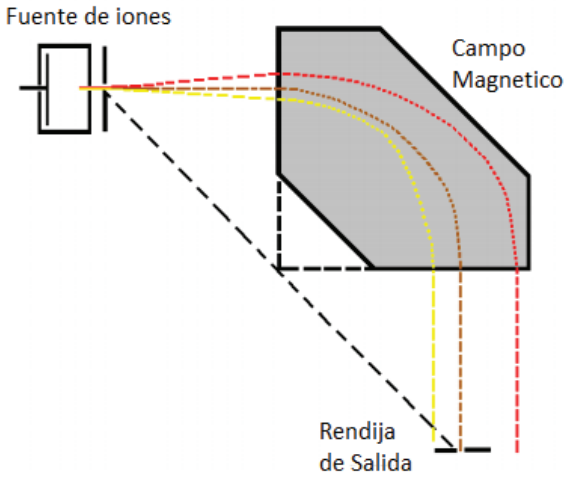


Figura 3.3.: Analizador de Masas

3.1.4 DETECTOR

Finalmente el detector se encarga de convertir el haz de iones en una señal eléctrica que es procesada y almacenada en la memoria de un ordenador. En la figura 3.4 se detalla el proceso, al ingresar un ion choca con el primer dinodo(electrodo de un fotomultiplicador cargado $+100$ [V] más que su predecesor) produciendo cierto número de electrones, que chocan con el segundo dinodo y así sucesivamente, obteniéndose al final la señal amplificada.

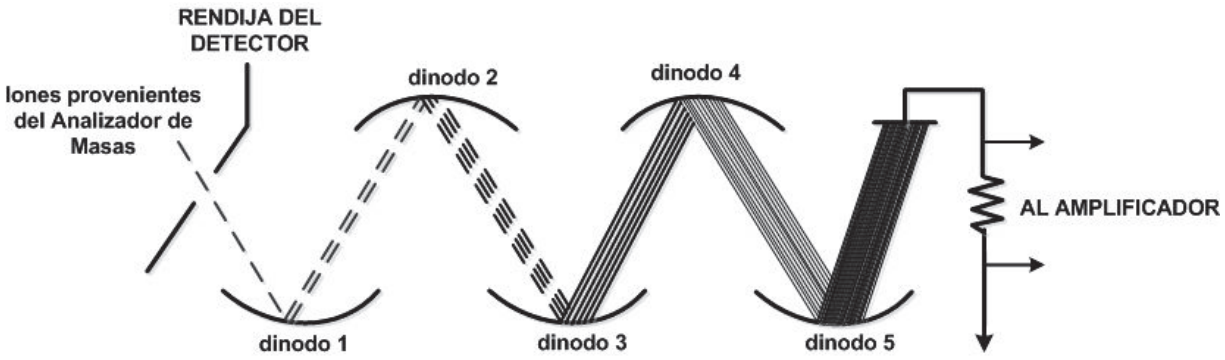


Figura 3.4.: Multiplicador de Electrones de Dínodos Discreto

3.2 PROCESAMIENTO DE MEDICIONES

Una vez adquiridas las mediciones por medio del espectrómetro y una vez que estos datos hayan sido guardados en formato digital, el cual pueda ser cargado en una plataforma computacional, es posible empezar con el procesamiento de las mediciones. El procesamiento esta formado por una serie de algoritmos y etapas que deben aplicarse en forma secuencial desde el remuestreo de las señales, para la homogeneización dimensional de las mediciones, hasta la eliminación del ruido de las mediciones mediante la aplicación de filtros estadísticos. A continuación en las siguientes subsecciones se describen cada una de estas etapas.

3.2.1 REMUESTREO DE MEDICIONES

El procesamiento de las mediciones, como se dijo en el Capitulo 1, empieza asumiendo que la curva resultante de graficar el vector de intensidades versus el vector de las relaciones masa a radio de cada medición toma la forma de una señal discreta en el tiempo. Con este criterio se puede aplicar entonces sobre las mediciones, algoritmos de procesamiento de señales. El primer paso, dentro de todas las etapas del procesamiento de señales, es el remuestreo de mediciones.

El remuestreo de mediciones en aplicaciones de minería de datos de espectrometría de datos tiene como objetivo, la homogeneización y una primera etapa de reducción dimensional de los vectores de mediciones de los grupos de análisis del conjunto de datos. La deshomogeneización se origina en los problemas de calibración de los equipos y la forma de como los datos fueron adquiridos, preparación de la muestra, niveles de contaminación del compuesto, ambiente, etc. Dichos factores intervienen directamente en la longitud vectorial de las mediciones. Los datos de las mediciones de los conjuntos usados para las simulaciones poseen una gran variabilidad en su longitud vectorial, lo cual no permite buscar patrones en ellos. Luego de esto, el criterio de reducción dimensional, nace de la característica propia de las mediciones, las cuales suelen poseer resoluciones de varios miles de puntos, en casos de hasta cientos de miles. Esta enorme cantidad de puntos colapsa cualquier plataforma computacional con características comerciales(no workstation), por lo que la cantidad de puntos(frecuencia de muestreo), debe ser disminuida a un valor manejable. Este concepto de valor manejable es subjetivo a los recursos disponibles, teniendo en cuenta que, al reducir la cantidad de puntos mediante el remuestreo, se pierde información, la búsqueda puede realizarse de manera secuencial, es decir en una primera etapa ubicar las regiones de interés, ubicadas las regiones de interés, mejorar la resolución y ganar precisión en los resultados.

Sea $y[n]$ una medición de espectrometría de masas, el remuestreo de la medición toma como puntos de partida el manejo de dos factores enteros L para elevar la frecuencia de muestreo de la medición y M para disminuir la frecuencia de muestreo(*upsampling & downsampling*). La nueva longitud de los vectores remuestreados(nueva frecuencia de muestreo) estará en función de estos dos factores tal que $F_1 = F_0 \frac{L}{M}$. En las referencias revisadas, al *upsampling* se le conoce también como interpolación y al *downsampling* como decimación. Para conseguir el remuestreo de señales, la interpolación y la decimación con combinadas, consiguiendo así, reducir o aumentar el numero de muestras(frecuencia de muestreo) en factores de $\frac{L}{M}$. En Matlab existe una función dedicada al remuestreo de mediciones de espectrometría de masas, la `msresample`, su forma de utilización se muestra a continuación en el Código 3.1 y la Figura 3.5.

Código: 3.1: Programación para el Remuestreo de Mediciones

```
1 % Remuestreo de Mediciones
2 % PARAMETROS:
3 %   Medicion = Vector de intensidades
4 %   MZ = vector de valores masa a carga
```

```
5 %   F_o = frecuencia de muestreo
6 %   [lim_inf   lim_sup] = limites de segmentacion
7 [MzRemuestreada MedicionRemuestreada] = ...
     msresample(MZ,Medicion,F_o,'RANGE',[2000 11000],'SHOWPLOT',true);
```

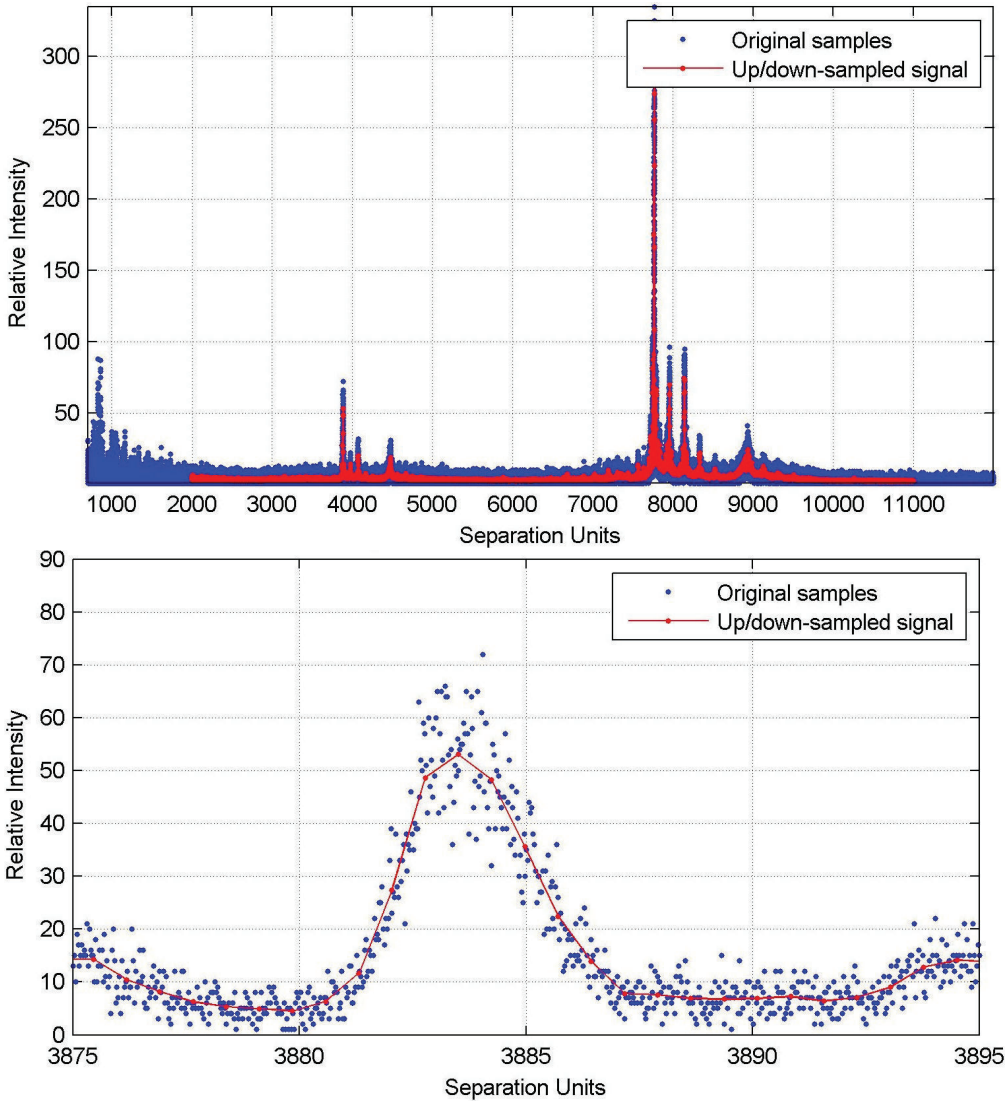


Figura 3.5.: Remuestreo de Mediciones

La curva en color rojo mostrada en la Figura 3.5 representa la nueva medición que ha sido construído en función de las muestras de la medición remuestreada. La construcción de dicha curva esta formada por dos nuevos vectores de intensidades y relación masa a carga, los cuales poseen una dimensión nueva escalada en un factor de $\frac{L}{M}$ en comparación a la medición original.

3.2.2 CORRECCIÓN DE LÍNEA DE BASE

La siguiente etapa luego del remuestreo es la Corrección de la Línea de Base. En esta etapa se busca corregir un fenómeno de offset presente en los primeros segmentos de la medición producido por la presencia de contaminantes, lo cual desemboca en una saturación de iones elevando los picos sobre una *línea de base*, la cual debe ser removida. El efecto de esta linea de base se muestra en la Figura 3.6.

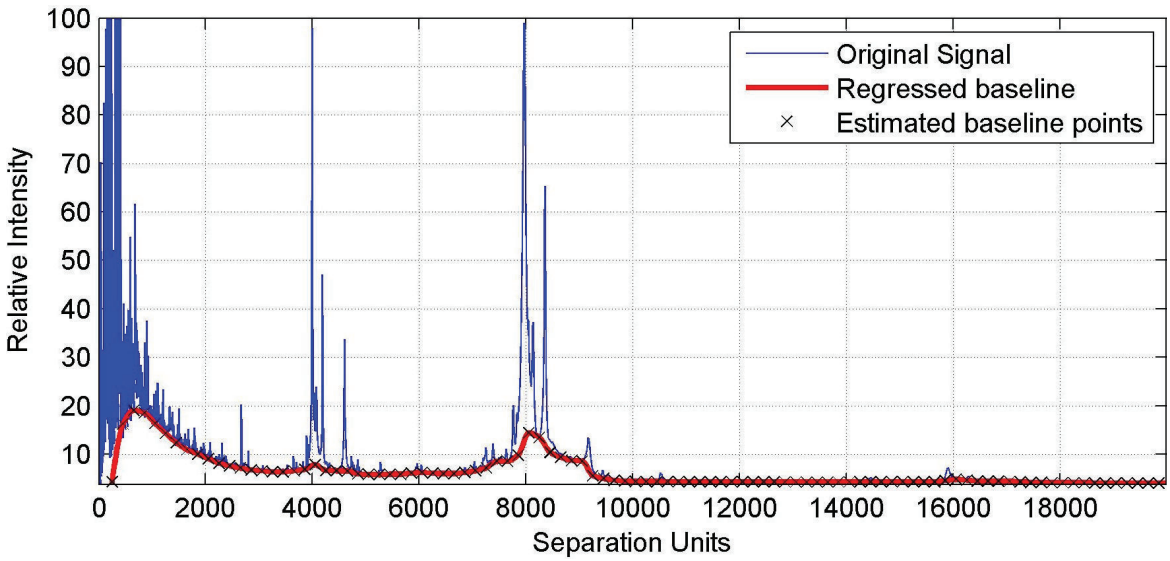


Figura 3.6.: Efecto de Linea de Base

En Matlab existe una función específica para la eliminación de la línea de base, `msbackadj`. Existen dos algoritmos implementados en esta función para la estimación de la curva en color rojo mostrada en la Figura 3.6. El primero usa como datos de entrada un cuantil en ventanas desplazadas para realizar una regresión y estimar la curva de la línea de base, el segundo usa modelos probabilísticos donde se asume que el cuantil de la muestra enventanada puede contener ruido o picos de una distribución gaussiana, lo cual se resume en un problema de Algoritmo esperanza-maximización.

El efecto de saturación que produce la linea de base se corrije dando como datos de entrada la medición a corregir y el vector de valores de masas a radios. Su forma de utilización se muestra en el Código 3.2.

Código: 3.2: Programación para la Corrección de Línea de Base

```
1 % Correccion de Linea de Base
2 % PARAMETROS:
3 %   Medicion = Vector de intensidades
4 %   MZ = vector de valores masa a carga
```

```
% Showplot muestra una grafica con las curvas estimadas
MedicionCorregida = msbackadj(MZ,Medicion,'SHOWPLOT',3);
```

3.2.3 ALINEACIÓN DE MEDICIONES

En la adquisición de mediciones de espectrometría de masas, la desalineación y corrimiento de los datos en el eje masa a radio y en el eje de las intensidades es bastante común, debido a errores de calibración de los instrumentos o a su vez de los entornos en donde se realizo la medición. La alineación de mediciones corrige el primero de estos problemas, tomando picos de referencia y corriendo hasta estos puntos todos sus equivalentes en las mediciones restantes a través de procesos de estiramiento en el eje masa a radio. En Matlab existe un proceso directo para la alineación de mediciones a través de la función `msalign`, su forma de utilización y efecto sobre las mediciones se muestra en Código 3.3 y en la Figura 3.8.

Código: 3.3: Programación para la Alineación de Mediciones

```
% Alineacion de Mediciones
% PARAMETROS:
%   Medicion = Vector de intensidades
%   MZ = vector de valores masa a carga
%   P = picos referenciales
P = [3991.4 4598 7964 9160];
title('Before Alignment')
MedicionAlineada = msalign(MZ,Medicion,P);
title('After Alignment')
```

3.2.4 NORMALIZACIÓN DE MEDICIONES

La normalización de mediciones es una etapa complementaria a la alineación de mediciones, ya que también existe un corrimiento en el eje de las intensidades. Este corrimiento o desalineación se produce debido a las diferencias en la cantidad de materia liberada de la muestra y de los iones producidos en el análisis de la muestra en el espectrómetro. Existen dos formas de compenzar estas diferencias, la primera normalizando las mediciones con respecto al área promedio bajo las curvas o a su vez normalizando las mediciones con respecto al área o altura de los picos de marcación de las sustancias con las que se preparo la muestra para su análisis. En la literatura

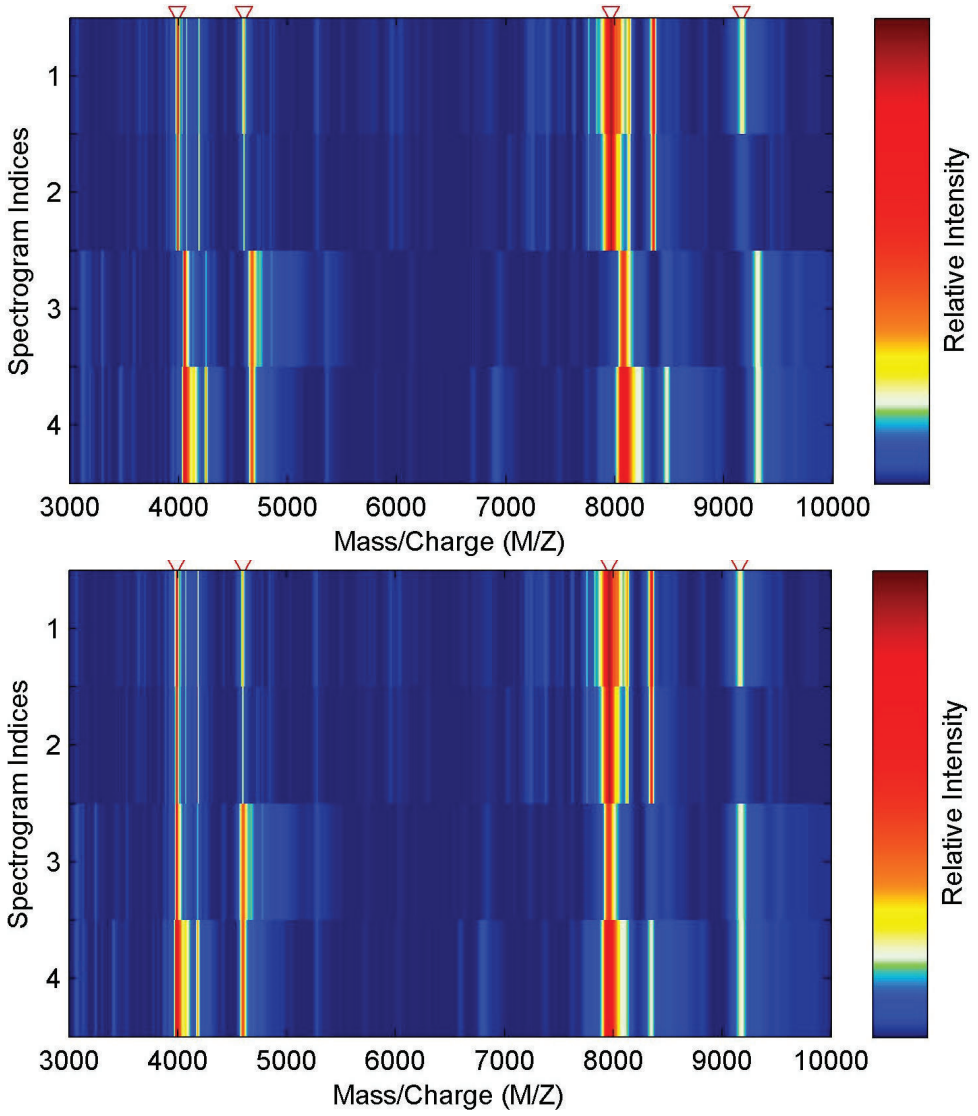


Figura 3.7.: Alineación de Mediciones

consultada se recomienda normalizar luego de corregir la línea de base y alinear las muestras. En el Código 3.4 se muestra la forma de normalizar en Matlab.

Código: 3.4: Normalización de Mediciones

```
% Normalizacion de Mediciones
% PARAMETROS:
%   Medicion = Vector de intensidades
%   MZ = vector de valores masa a carga
MedicionNormalizada = msnorm(MZ,Medicion,'QUANTILE',1,'LIMITS',[1000 ...
    inf],'MAX',100);
```

3.2.5 SUAVIZAMIENTO DE RUIDO EN MEDICIONES

La última etapa del procesamiento de mediciones de espectrometría de masas es el suavizamiento del ruido presente en las mediciones. El suavizamiento del ruido, en este tipo de aplicaciones, se aborda al igual que en cualquier aplicación de procesamiento de señales, con el objetivo de reducir al mínimo la cantidad del ruido, sin producir distorsiones o cambios radicales en la señal analizada, en este caso, una medición de espectrometría de masas. El termino suavizamiento hace referencia que al contrario de la terminología usada en el procesamiento de señales, si bien, la idea es filtrar el ruido, en este tipo de aplicaciones, no se procede a hacer un filtrado directo, sino mas bien ha hacer un suavizado de los puntos de intensidades que no sigan la distribución de probabilidad de la muestra.

En Matlab esta disponible la función `mssgolay`, la cual tiene implementado un filtro de Savitzky–Golay, con el cual se realiza el suavizamiento del ruido en las muestras de manera iterativa. El filtro de Savitzky–Golay es un filtro estadístico, el cual construye una nueva curva de medición por medio de una regresión polinomial local de grado k. La ventaja de aplicar este filtro y no los tradicionales filtros digitales, es que la nueva curva tiende a preservar las características de la distribución de probabilidad original de la muestra con los datos de entrada, se conservan ademas los máximos y mínimos relativos y el ancho de los picos. El proceso de introducción de datos y codificación en Matlab se muestra en el Código 3.5 y su efecto en la Figura.

Código: 3.5: Suavizamiento de Ruido en Mediciones

```
% Suavizamiento del Ruido
% PARAMETROS:
%   Medicion = Vector de intensidades
%   MZ = vector de valores masa a carga
MedicionSuavizada = mssgolay(MZ,Medicion,'SPAN',35,'SHOWPLOT',3);
```

3.3 PREPARACIÓN DE MUESTRAS PARA LA SELECCIÓN DE CARACTERÍSTICAS

Una vez que se ha terminado el procesamiento de las mediciones, es necesario preparar los datos de forma que la matriz que reciba la etapa de selección de características disponga de un solo vector de referencias de masa a radio y de las mediciones de las

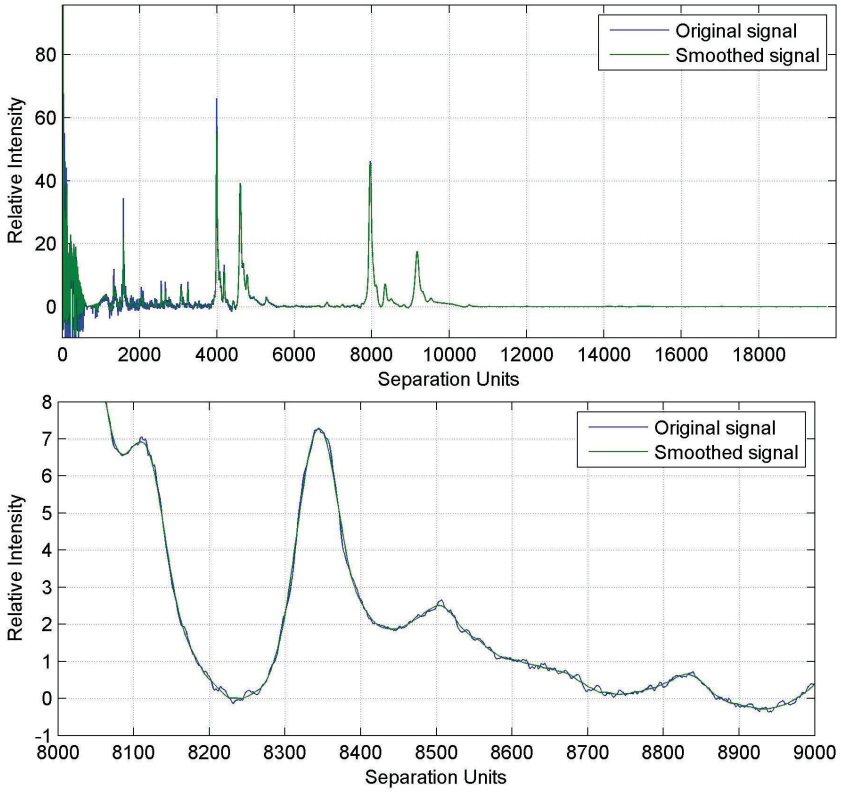


Figura 3.8.: Suavizamiento de Ruido en Mediciones

intensidades en forma apilada sobre la matriz de referencia en el eje m/z, como se muestra en la Ecuación 3.1.

$$Datos = \left[\begin{array}{ccccc} I_1^1 & I_2^1 & I_3^1 & \dots & I_n^1 \\ I_1^2 & I_2^2 & I_3^2 & \dots & I_n^2 \\ I_1^3 & I_2^3 & I_3^3 & \dots & I_n^3 \\ \dots & \dots & \dots & \dots & \dots \\ \hline mz_1 & mz_2 & mz_3 & \dots & mz_n \end{array}\right] \tag{3.1}$$

De este punto en adelante, las acciones que se realicen en los datos estarán centradas en el análisis de los valores de las intensidades y en la ubicación de las posiciones de los biomarcadores. Los algoritmos a aplicar en el siguiente capítulo determinaran los índices vectoriales de las posiciones de los pares $\{mz_i, I_j\}$ mediante las cuales se puedan definir los biomarcadores.

CAPÍTULO 4

Selección de Características del Subconjunto de Mutaciones

En este capitulo se presentan los detalles teóricos de la etapa de selección características en los conjuntos de mediciones. La selección de características es un paradigma de reconocimiento de patrones donde el objetivo es de un gran conjunto de posibles valores(características), tomar las mínimas necesarias para definir el conjuntos de valores completamente. Este concepto se aplica a todas las aplicaciones de reconocimiento de patrones, en este caso, ese conjunto mínimo de características a buscar representa el conjunto de mutaciones, los cambios en donde empieza a manifestarse el cáncer como estado patológico. El alcance de esta tesis se limita a exponer el patrón geométrico(regiones de biomarcación) con el cual se definen estas moléculas. En este trabajo, este patrón geométrico toma el nombre de región de biomarcación, la cual sera validada mediante dos pruebas independientes a doble ciego, cuyos detalles se exponen en el siguiente capítulo.

4.1 SELECCIÓN DE CARACTERÍSTICAS DISCRIMINANTES

En la etapa de Selección de características discriminantes se reciben como datos de entrada la matriz de intensidades de mediciones y un vector de valores de relaciones masa a radio. Esta Matriz es analizada en esta etapa en el eje de las intensidades, ya que es aquí donde se producen las mutaciones intergrupales(alteraciones). Los valores de las intensidades en cada de cada una de las mediciones son tomados en forma secuencial como datos de entrada para las pruebas estadísticas: t-Student, Wilcoxon y χ^2. Una vez calculados los estadísticos, se definen los indices de las posiciones de los valores de las intensidades que pasaron la prueba y los que no pasaron la prueba en cada uno de los tres casos(t-Student, Wilcoxon y χ^2) respectivamente. En el caso de que la prueba haya sido rechazada, esto implica que en esa posición hubieron cambios estadísticamente significativos, los cuales desde el punto de vista biológico

se traducen en mutaciones, las cuales son el origen del cáncer. Mutaciones descontroladas en órganos y sistemas producen en sus etapas iniciales alertas químicas, en las que el sistema esta reportando estas anomalías, estas alertas reciben el nombre antígenos. Las características seleccionadas en este tipo de investigaciones producen las primeras definiciones de este tipo de sustancias y proteínas que caracterizan el comportamiento del cáncer.

Esta investigación combina los resultados de las características detectadas con las pruebas t-Student, Wilcoxon y χ^2 en un solo grupo de características, para ser evaluados en la siguiente etapa. Los fundamentos teóricos de las pruebas estadísticas t-Student, Wilcoxon y χ^2 y presentan a continuación en las siguientes secciones.

4.1.1 PRUEBA DE T-STUDENT

La prueba del *t-Student* es una prueba paramétrica que ayuda en la comprobación de una hipótesis, asumiendo algunas propiedades en los datos. Los datos de cada muestra pueden ser vectores o matrices. Esta prueba utiliza la Distribución de *t-Student*, que modela una familia de curvas que varía en función de los grados de libertad g.

Para aplicar la prueba de *t-test* se asume que:

1. La distribución de los datos debe seguir una curva normal.
2. La desviación estándar (s) de cada grupo de análisis son desconocidas pero se asumen que son iguales.

La hipótesis a comprobar en la prueba de t-Student es si las medias de las muestras analizadas son iguales, es este caso, para estas aplicaciones, lo que se busca comprobar es si la media de las mediciones de los grupos de análisis son iguales.

Si el planteamiento inicial de la hipótesis no cambia de ninguna manera se le denomina como hipótesis nula y esto será verdad pero si los datos dicen lo contrario se cumple la hipótesis alternativa.

- **Hipotesis Nula:** No existe una diferencia entre los grupos de análisis.
- **Hipotesis Alternativa:** Si existe una diferencia entre los grupos de análisis.

Para pruebas de procesamiento en datos de espectrometría de masas se establece un intervalo de confianza (CI) del $95\,\%$ y un $5\,\%$ de error. El CI es el rango entre dos valores donde se estima que estará un valor desconocido con una determinada probabilidad de acierto.

Para aplicar la prueba se define el estadístico t que ayuda a aceptar o rechazar la hipótesis al comparar el t calculado con el estadístico teórico disponible en tablas. El estadístico t se calcula mediante la siguiente Ecuación 4.1:

$$t = \frac{\mu_A - \mu_B}{s * \sqrt{\frac{1}{m_A} + \frac{1}{m_B}}} \tag{4.1}$$

con

$$s = \frac{(m_A - 1)s_A^2 + (m_B - 1)s_B^2}{m_A + m_B - 2} \tag{4.2}$$

Donde, μ_A y μ_B son las medias de cada grupo, s la desviación estándar total, s_A y s_B la estimación de la desviación estándar obtenidas de cada grupo de muestras y m_A, m_B son el número de muestras de cada conjunto representadas por todas las intensidades de cada uno de los vectores del conjunto.

La desviación estándar es una medida de dispersión, indica cuanto pueden alejarse los valores de la media. Los grados libertad se calculan como $g = n - 2$, donde $n = m_A + m_B$ es el número de muestras totales.

Con los grados de libertad y el intervalo de confianza se busca en la tabla del *t-test*, el valor del t téorico para construir la distribución de probabilidad.

Si las muestras difieren de otras, se asume que sus medias no son iguales. En consecuencia, dichas muestras nos aportan información de datos altamente discriminantes. Este grupo de información discriminante será seleccionado para el análisis.

EJEMPLO:

Se tiene dos grupos(grupo A y grupo B) de análisis en el cuadro 4.1 con 10 muestras cada uno respectivamente,

GRUPO A	16.85	16.40	13.21	16.35	16.52	17.04	16.96	17.15	16.59	16.57
GRUPO B	17.50	17.63	18.25	18.00	17.86	17.75	17.90	17.96	16.59	18.15

Cuadro 4.1.: Datos: Grupo A y Grupo B

para obtener el estadístico de la fórmula antes mencionada es necesario calcular el valor de las medias de cada grupo es así que luego de calcular la media para cada grupo, tenemos μ_A = $16{,}76$ y μ_B = $17{,}92$.

A continuación se procede a calcular el valor de s, como datos conocidos se tiene el número de muestras para m_A = m_B = 10, luego se calcula el valor de s_A y s_B que se obtiene de la ecuación 4.3.

$$s_m^2 = \frac{1}{m}\sum_{i=1}^{m}(X_i - \bar{X})^2 \tag{4.3}$$

Donde, X_i es cada uno de los datos para cada grupo, $\bar{X}$ es la media aritmética de los datos y m es el número de muestras de cada grupo. Con esto se obtiene S_A = $0{,}316$ y S_B = $0{,}247$. Una vez que se tiene todos los parámetros requeridos se calcula la desviación estándar total s = $0{,}204$.

Se cálcula el $t_{experimental}$ = $-9{,}13$ con su valor absoluto $|t_{exp}|$ =$9{,}13$. Para los grados de libertad obtenidos con $m_A + m_B - 2$ = 18, el intervalo de confianza $(1-\alpha)$ = $95\,\%$ con el valor de $\frac{\alpha}{2}$ = $0{,}025\,\%$. Con estos valores localizamos el t téorico en tabla de Distribución de $t - Student$ ver la tabla 4.2, que muestra el valor de t_{teo} = $2{,}10$.

Entonces se observa que $t_{exp} > t_{teo}$ por lo que se rechaza la H_0.

$\alpha/2$	0.25	0.2	0.15	0.1	0.05	0.025	0.01	0.005	0.0005
g									
1	1.000	1.376	1.983	3.078	6.314	12.706	31.821	63.658	638.578
2	0.816	1.061	1.386	1.886	2.920	4.303	6.965	9.925	31.600
3	0.765	0.978	1.250	1.638	2.353	3.182	4.541	5.841	12.924
4	0.741	0.941	1.190	1.533	2.132	2.776	3.747	4.604	8.610
5	0.727	0.920	1.156	1.476	2.015	2.571	3.365	4.032	6.889
6	0.718	0.906	1.134	1.440	1.943	2.447	3.143	3.707	5.959
7	0.711	0.896	1.119	1.415	1.895	2.365	2.998	3.499	5.408
16	0.690	0.865	1.071	1.337	1.748	2.120	2.583	2.921	4.015
17	0.689	0.863	1.069	1.333	1.740	2.110	2.567	2.898	3.955
18	0.688	0.862	1.067	1.330	1.734	2.101	2.552	2.878	3.922
19	0.688	0.861	1.066	1.328	1.729	2.093	2.539	2.861	3.883
20	0.687	0.860	1.064	1.325	1.725	2.086	2.528	2.845	3.850

Cuadro 4.2.: Tabla de Distribución t-Student

4.1.2 PRUEBA DE SUMA DE RANGOS DE WILCOXON-MANN-WHITNEY

Es una prueba no paramétrica que sirve para comprobar si dos conjuntos de muestras tienen medianas iguales. Las muestras en estudio no siguen una distribución normal. El estadístico Z_A se calcula con el mínimo valor entre U_A y U_B (4.4).

$$U = min(U_A, U_B) \tag{4.4}$$

La prueba de *U-Mann-Whitney* se realiza de la siguiente forma:

1. Dados 2 grupos de análisis A y B. Se ordenan ascendentemente todas las observaciones o muestras en un solo conjunto, sin importar el grupo al que pertenecen.

2. Se asigna un rango de orden a cada valor como se indica en la tabla 4.3 los rangos de operación asignados empiezan en 1 e incrementan en 1. Los problemas de valores repetidos en los datos se corrigen obteniendo la media aritmética de los rangos a los que pertencen los valores repetidos, asi por ejemplo en la tabla se observa dos veces el número 13, que pertenecen a los rangos 3 y 4 cuya media aritmética es $3{,}5$ y luego reemplazamos este valor en la fila de rangos.

GRUPO A	7	13	22	23					
GRUPO B	10	13	21	22	25				
GRUPOA Y GRUPOB	7	10	13	13	21	22	22	23	25
RANGOS DE ORDEN	1	2	3	4	5	6	7	8	9
CORRECCIÓN DE VALORES REPETIDOS	1	2	3.5	3.5	5	6.5	6.5	8	9

Cuadro 4.3.: Asignación de rangos de Operación

Por último se suman los rangos por separado.

R_A sumatoria de los rangos de orden del primer grupo de análisis.

R_B sumatoria de los rangos de orden del segundo grupo de análisis.

3. Mediante (4.5) y (4.6), se calcula U_A y U_B para elegir posteriormente el mínimo valor.

$$U_A = n_A * n_B + \frac{n_A(n_A+1)}{2} - R_A \tag{4.5}$$

$$U_B = n_A * n_B + \frac{n_B(n_B+1)}{2} - R_B \tag{4.6}$$

Donde, n_A y n_B son los tamaños muestrales de cada conjunto.

4. En (4.7) se calcula Z téorico para comparar con el Z experimental y aceptar o rechazar la hipótesis.

$$Z = \frac{U - (n_A * n_B/2)}{\sqrt{\frac{n_A * n_B(n_A+n_B+1))}{12}}} \tag{4.7}$$

En la ecuación (4.8), se acepta o se rechaza la hipótesis.

$$Si\ Z \leq Z_\alpha \Rightarrow\ Se\, acepta\, la\, H_0 \quad Si\ Z > Z_\alpha \Rightarrow\ Se\, rechaza\, la\ H0 \tag{4.8}$$

EJEMPLO:

De la tabla 4.3 se observan dos grupos A y B. De los cuales se asigna los rangos de valor y se corrige los valores repetidos. Se separan los rangos corregidos y se suman. Ver cuadro 4.4:

GRUPO A	7	13	22	23	
RANGOS DE ORDEN A	1	3.5	6.5	8	
GRUPO B	10	13	21	22	25
RANGOS DE ORDEN B	2	3.5	5	6.5	9

Cuadro 4.4.: Asignación de rangos de Operación por grupo

$R_A = 1 + 3{,}5 + 6{,}5 + 8 = 19$

$R_B = 2 + 3{,}5 + 5 + 6{,}5 + 9 = 26$

U_A=$4 * 5 + \frac{4*(4+1)}{2} - 19$ = 11

U_B=$4 * 5 + \frac{5*(5+1)}{2} - 26$ = 9

Remplazando los datos en (4.5) y (4.6), se calcula $U_A = 11$ y $U_B = 9$ cuyo mínimo valor es $U = 9$.

Con la ecuación (4.7) se calcula:
$Z_{experimental}$ = $-0{,}245$ con su valor absoluto
$|Z_{experimental}|$ =$0{,}245$.

En la tabla 4.5 se localiza el $Z_{teorico} = 1$ con el intervalo de confianza $(1 - \alpha)$ = $95\,\%$, donde:
$\frac{\alpha}{2}$ = $0{,}025\,\%$,
$n_A = n = 4$
$n_B = m = 5$.

Z=$\frac{9-((4*5)/2)}{\sqrt{\frac{4*5(4*5+1)}{12}}}$ = $-0{,}245$

Se observa que $Z_{experimental} < Z_{teorico}$ en consecuencia se acepta la hipótesis Ho [26].

α	= 0.025								
				n					
m	2	3	4	5	6	7	8	9	10
2	-	.	.	.	.	.	.	.	.
3	-	-	.	.	.	.	.	.	.
4	-	-	0	.	.	.	.	.	.
5	-	0	1	2	.	.	.	.	.
6	-	1	2	3	5	.	.	.	.
7	-	1	3	5	6	8	.	.	.
8	0	2	4	6	8	10	13	.	.
9	0	2	4	7	10	12	15	17	.
10	0	3	5	8	11	14	17	20	23

Cuadro 4.5.: Tabla de Distribución de U Mann-Whitney

4.1.3 PRUEBA DE CHI CUADRADO (χ^2)

La prueba χ^2 es una prueba no paramétrica que mide la discrepancia entre una distribución de frecuencias observadas y una distribución de frecuencias teóricas o esperadas, indicando el nivel de discrepancia y además prueba la independencia entre los grupos de análisis.

Esta prueba sigue la distribución χ^2, esta distibución es una familia de curvas que varía en función de los grados de libertad $(I-1)(J-1)$. Donde, I es el número de filas y J es el número de columnas.

4.1.3.1 HIPÓTESIS A COMPROBAR:

Hipotesis a comprobar: Comprobar que existe discrepancia entre dos grupos de muestras.

- **Hipótesis Nula:** Las variables en estudio son independientes.
- **Hipótesis Alternativa:** Las variables en estudio están relacionadas.

Para aplicar la prueba se define el estadístico X^2 que ayuda a aceptar o rechazar la hipótesis al comparar el X^2 experimental con el estadístico teórico disponible en tablas. El estadístico se calcula con la ecuación (4.9):

$$X^2{}_{exp} = \sum_{ij} \frac{(fo_{ij} - fe_{ij})^2}{fe_{ij}} \tag{4.9}$$

Donde, fo_{ij} es la frecuencia observada para la ij_esima posición y fe_{ij} es la frecuencia esperada para la ij_esima posición.

En la ecuación (4.10), se calculan las frecuencias esperadas marginales.

$$fe_{ij} = \frac{(Total\, i - esima\, fila) * (Total\, j - esima\, columna)}{TotalGlobal} \tag{4.10}$$

Las discrepancias son medidas calculando la diferencia entre las frecuencias observadas y las frecuencias esperadas según la ecuación (4.9). La hipótesis nula se rechaza cuando $X^2{}_{exp} > X^2{}_{critico}$ [51].

EJEMPLO

En la siguiente tabla 4.6 se muestra las frecuencias observadas para dos grupos de análisis y las suma de filas y columnas correspondientes.

DATOS	GRUPO A	GRUPO B	$\sum$
	1	10	11
	3	2	5
	5	7	12
$\sum$	9	19	28

Cuadro 4.6.: Datos χ^2)

Con la ecuación 4.10 se obtienen las frecuencias esperadas, dicho cálculo se detalla en la ecuación 4.11 y los resultados se aprecian en la tabla 4.7.

f_{i*j}		
	3.535	7.464
	1.607	3.392
	3.857	8.142

Cuadro 4.7.: Resultados χ^2

$$f_{1*1} = \frac{(11 * 9)}{28} = 3{,}535 \tag{4.11}$$

Reemplazando estos valores en la ecuación 4.9 como se muestra en ecuación 4.12 para obtener el estadístico experimental.

$$x^2_{exp} = \frac{(1-3{,}535)^2}{3{,}535} + \frac{(10-7{,}464)^2}{7{,}464} + \frac{(3-1{,}607)^2}{1{,}607} + \frac{(2-3{,}392)^2}{3{,}392} + \frac{(5-3{,}857)^2}{3{,}857} + \frac{(7-8{,}142)^2}{8{,}142} \tag{4.12}$$

$$x^2_{experimental} = 4{,}954$$

El $x^2_{teorico}$ se localiza en la tabla 4.8 con los siguientes datos: $g = (I-1)*(J-1) = (3-1)*(2-1) = 2$ y $\alpha = 0{,}05$. El valor es $x^2_{teorico}$ = $5{,}99$ que define el punto crítico de la distribución de χ^2.

α	0.05	0.01	0.001
g			
1	3.84	6.64	10.83
2	5.99	9.21	13.82
3	7.82	11.34	16.27
4	9.49	13.28	18.47
5	11.07	15.09	20.52
6	12.59	16.81	22.46
7	14.07	18.48	24.32
8	15.51	20.09	26.12
9	16.92	21.67	27.88
10	18.31	23.21	29.59

Cuadro 4.8.: Tabla de Distribucíon $chi2$

La Ho es aceptada ya que el $x^2_{experimental}$ =$4{,}954$ es menor que $x^2_{teorico}$ = $5{,}99$.

4.1.4 FILTRO GEOMÉTRICO DE DISTANCIAS MÍNIMAS

En 1938 Prasanta Chandra Mahalanobis introdujo una nueva métrica para medir la similitud de dos variables aleatorias multidimensionales. Formalmente esta distancia asume que las dos variables aleatorias($\overrightarrow{x}$ y $\overrightarrow{y}$) poseen la misma distribución de probabilidad. En esta tesis se aplica una idea inversa a esta, se asume que las distancias muy cortas entre las variables analizadas dificultan el poder diferenciarlas. Un ejemplo de este tipo de fenómenos en estas aplicaciones se presenta en la Figura 4.1, donde se puede notar claramente el exceso de marcadores en determinadas zonas del espectro.

Para poder eliminar las zonas del espectro redundantes se aplica un concepto simple, eliminar los marcadores ubicados a la distancia mínima de las zonas detectadas.

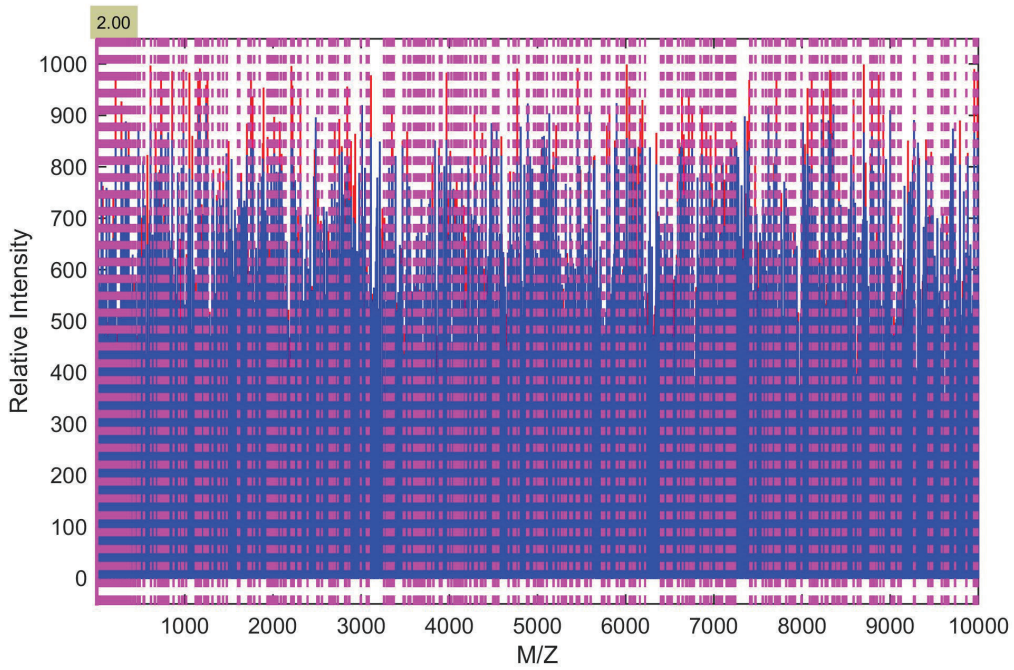


Figura 4.1.: Zonas detectadas con información redundante

Estos es, primeramente se miden todas las distancias de las zonas detectadas en el espectro. Luego se evalua cual fue la menor distancia detectada y finalmente en un proceso iterativo se eliminan secuencialmente las características concecuentes a la característica del marcador inicial.

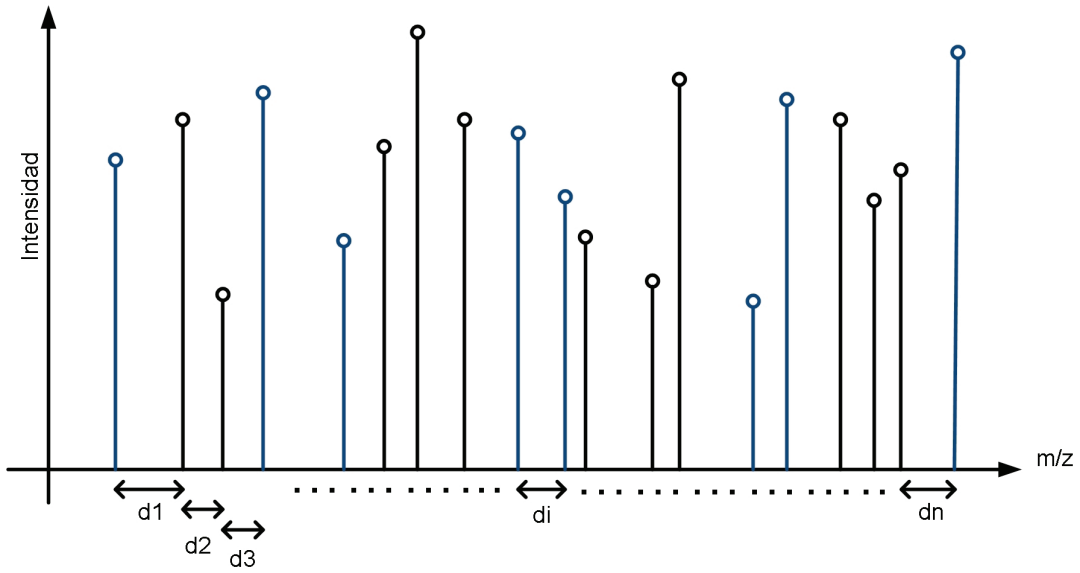


Figura 4.2.: Filtro Geométrico de Distancias Mínimas

En la Figura 4.2 se esquematiza el concepto de medicion de las distancias usado para el proceso de filtrado. Sea $d_1, d_2, d_3, \ldots, d_i, \ldots, d_n$. las distancias entre marcadores detectados, se orden de forma descendente estos valores, para luego ubicar la distancia de referencia y eliminar las ubicadas a esta distancia del espectro. Los resultados de este proceso optimizan la cantidad de marcadores en el espectro reduciendo su número en un factor de 200 a 1. Los resultados con un número reducido de marcadores en el espectro se presentan a nivel general como un elevado rendimiento en los clasificadores, menor evidencia de *overfitting* o *underfitting*, atenuación de los efectos de ruido, mayor grado de generalización y robustez a la detección de nuevas muestras.

4.1.5 ADABOOST

AdaBoost.M1 es un algoritmo de *Machine Learning* muy popular para clasificación binaria $\{G_1, G_2\}$. Este algoritmo supone que se disponen de N observaciones $\left\{x_{(i)}, y_{(i)}\right\}$ donde $x_{(i)}$ representa cada una de las muestras de la poblacion(mediciones de espectrometria de masas) y $y_{(i)}$ representa las clases de un sistema clasificador binario ($\{+1, -1\}$). El algoritmo entrena algoritmos de clasificación simples[7] secuencialmente. Por cada clasificador simple, existe un índice t, con el cual AdaBoost.M1 calcula el error de clasificación ponderada,

$$\varepsilon_i = \sum_{n=1}^{N} d_n^t * I(y_n \neq h_t[x_n]) \tag{4.13}$$

donde:

- x_n es un vector de valores de predicción para la observación n.
- y_n es la etiqueta de clase verdadera.
- $f(x_n)$ ε $(-\infty, +\infty)$ es el puntaje de clasificación previsto.
- I es la función indicadora.
- d_n^t es el peso de la observación n en el paso t.

[7] weak learners

AdaBoost.M1 luego aumenta pesos para las observaciones mal clasificados por clasificador simple t y reduce los pesos para las observaciones correctamente clasificados por clasificador simple t. El siguiente clasificador debil $t+1$ es entonces entrenado en los datos actualizando con los pesos $d_n^{(}t+1)$.

Una vez finalizado el entrenamiento, AdaBoost.M1 calcula la predicción de nuevos datos utilizando:

$$f_{(}x) = \sum_{t=1}^{T} \alpha_t h_t[x] \tag{4.14}$$

Son los pesos de las hipótesis débiles en el conjunto.

A continuación se muestra un ejemplo de la filosofía de clasificación del Algoritmo AdaBoost.M1 en forma gráfica partiendo de la Figura 4.3. El problema propuesto es clasificar los símbolos $+$ y los símbolos $-$, es decir, se trata de un problema binario de clasificación donde hay dos clases o grupos de análisis y el conjunto de datos es $\{+,-\}$. En la Figura 4.3 se muestra el conjunto original de datos, luego se aplica sobre este conjunto un primer intento de clasificacion, mostrado en la Figura 4.4 con un rendimiento de $\epsilon = 0{,}30$ y un $\alpha = 0{,}42$,

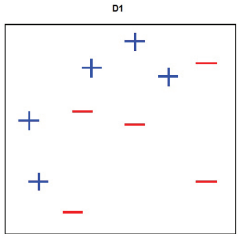


Figura 4.3.: Conjunto original de Datos

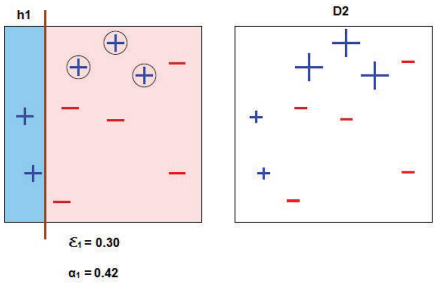


Figura 4.4.: Iteración de Clasificación 1

En la Figura 4.4 se actualizan los pesos a las muestras del conjunto mal clasificadas, para que en la siguiente iteración mostrada en la Figura 4.5, estas se reclasifiquen, obteniendo un valor de $\epsilon = 0{,}21$ y un $\alpha = 0{,}65$ y en una nueva iteración estos valores se actualizan a $\epsilon = 0{,}14$ y un $\alpha = 0{,}92$.

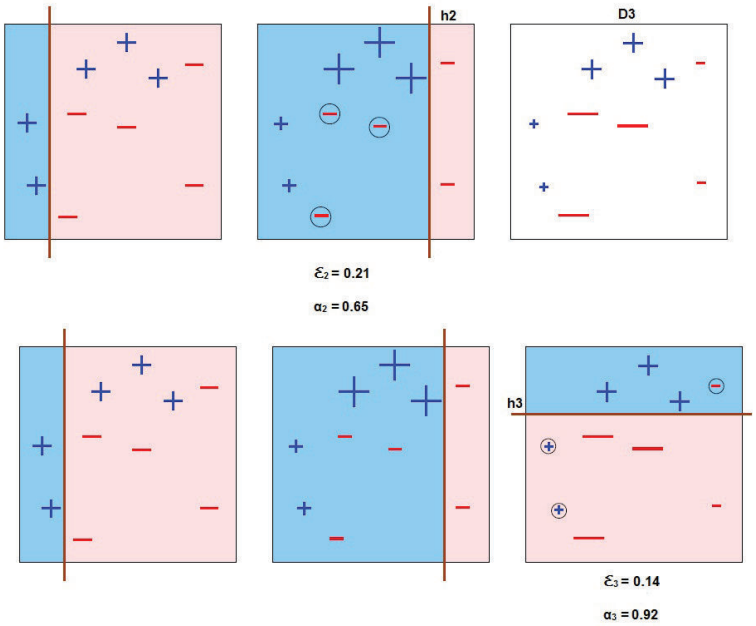


Figura 4.5.: Iteración de Clasificación 2 y 3

Finalmente, se combinan los resultados obtenidos por los tres clasificadores simples en un solo clasificador, ponderando su colaboración en este clasificador con los valores de α. Los aciertos de cada clasificador refuerzan el rendimiento del clasificador general, obteniéndose un mejor resultado que en forma independiente, clasificador por clasificador, este proceso se muestra en la Figura 4.6.

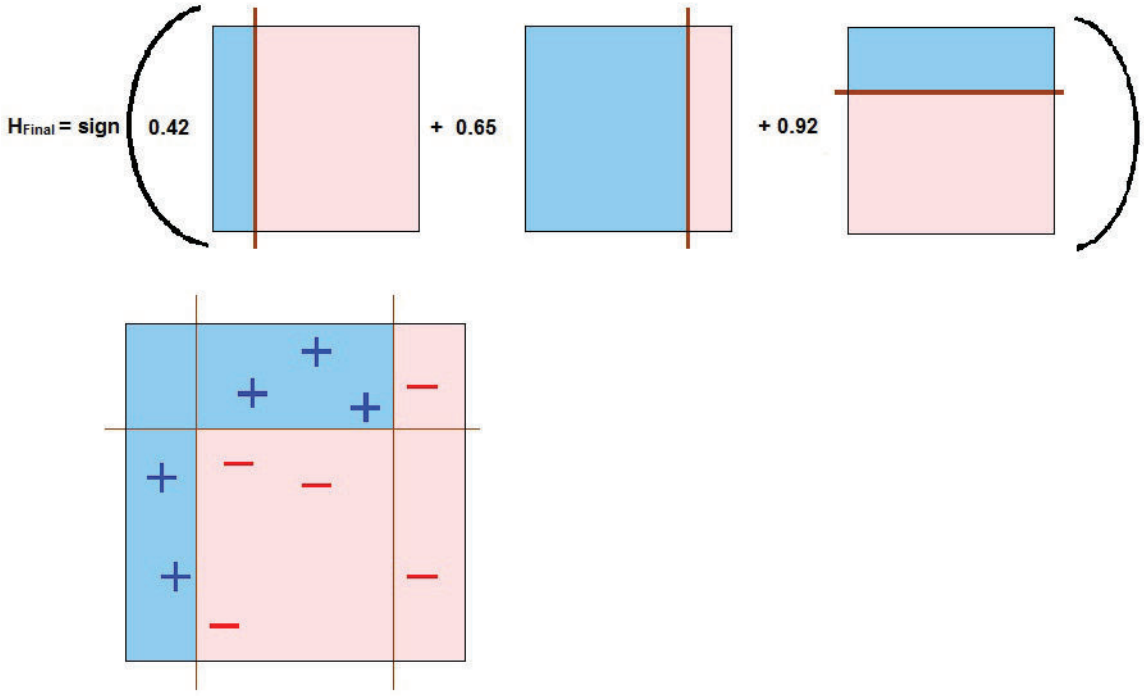


Figura 4.6.: Clasificador Final y Resultados

CAPÍTULO 5

Validación de Resultados

Este capitulo completa el proceso del algoritmo propuesto en este trabajo de titulación. La validación es la etapa crucial de todo este algoritmo desde el punto de vista de marcar los resultados como confiables o no confiables. Hay que tener presente que, todo las etapas anteriores manejan conceptos probabilísticos, entonces, los resultados también estarán sujetos a este escenario. En el conjunto de datos analizado producirá zonas de biomarcación, las cuales serán en mayor o menor grado, probabilisticamente ciertas, como definición de moléculas o antígenos que alerten sobre la presencia de estados cancerígenos en los tejidos analizados. Las definiciones que se validaran en esta etapa serán por tanto sujetas a definir que tan probabilisticamente correcto resulta utilizarlas como conjunto mínimo para caracterizar el grupo de análisis $\{1\}$ o el grupo de análisis $\{2\}$. Las métricas de evaluación serán la curva de aprendizaje del clasificador Adaboost M1 en arboles de decisión y pruebas usando muestras externas. Ninguna de las mediciones usadas en estas dos pruebas han intervenido anteriormente en las etapas descritas en el capitulo anterior, por lo que se garantiza el escenario a doble ciego. La interpretación de los resultados de etapa deberá interpretarse entonces, analizando los niveles de certeza alcanzados con las definiciones de las zonas de biomarcacion anteriores, bajo esto la conclusión debe ser, si existe o no el suficiente nivel de certeza para identificar muestras no etiquetadas en base a ese conjunto de zonas de biomarcacion(caracteristicas).

5.1 TÉCNICAS DE VALIDACIÓN DE CLASIFICADORES

Básicamente existen tres técnicas de evaluación de los clasificadores. La Curva ROC, las pruebas usando muestras externas de las cuales se conoce sus etiquetas y la *crossvalidation*. Las dos primeras técnicas representan evaluaciones superficiales al comportamiento del clasificador, ya que los casos analizados son directamente proporcionales a las muestras disponibles de análisis. En el caso de las aplicaciones

relacionadas con el diagnostico del cáncer usando mediciones de espectrometría de masas, una de las limitaciones mas rígidas es la limitada cantidad de muestras de mediciones disponibles, por tanto estas dos técnicas no representan una opción a la hora de evaluar clasificadores en este tipo de aplicaciones.

La *crossvalidation* en tanto, maneja la idea de crear mucho escenarios de evaluación creados en base a distintas combinaciones las cuales se escogen aleatoriamente. En otras palabras se escogen muchos casos de análisis donde las muestras son agrupadas de manera aleatoria en grupos de entrenamiento y pruebas de manera masiva, todos estos casos de análisis se evalúan independientemente y el error total del sistema, es la media de los errores medidos. Entonces, por razones practicas, la *crossvalidation* es la técnica que mas se ajusta a las necesidades de la evaluación de clasificadores usados para el diagnostico y minería de datos en aplicaciones de espectrometría de masas.

5.2 VALIDACIÓN CRUZADA

La técnica de validación cruzada usada en este proyecto de titulación se denomina, validación cruzada en k iteraciones(k-fold cross-validation). La idea de este método es dividir el conjunto original en un K conjunto de prueba y el resto destinarlo a datos del conjunto de entrenamiento $(K-1)$. Este proceso se repite k veces con todas las posibles combinaciones de los conjuntos para pruebas y entrenamiento, tal como se muestra en la Figura 5.1.

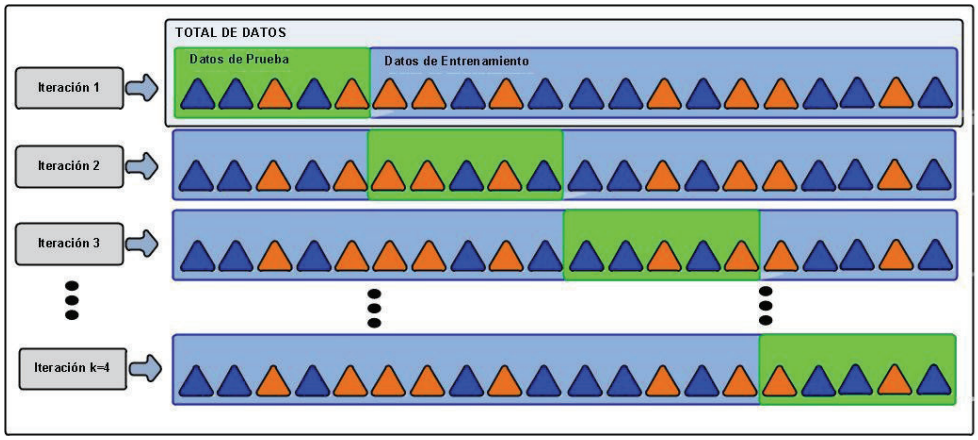


Figura 5.1.: CrossValidation

Finalmente el error total del sistema se estima con la media aritmética de los resultados en cada iteración, este proceso se muestra en la Figura 5.2. Este proceso muestra un alto grado de precisión al evaluar el clasificador modelado ya que se evalúan K combinaciones en el conjunto de datos de manera aletoria para pruebas y entrenamiento, lo

cual a pesar de ser lento computacionalmente hablando, garantiza que se engloben un gran numero de posibilidades de manera aletoria sin influencia externa sobre cuales muestras destinar a las diferentes etapas de modelación del sistema clasificador.

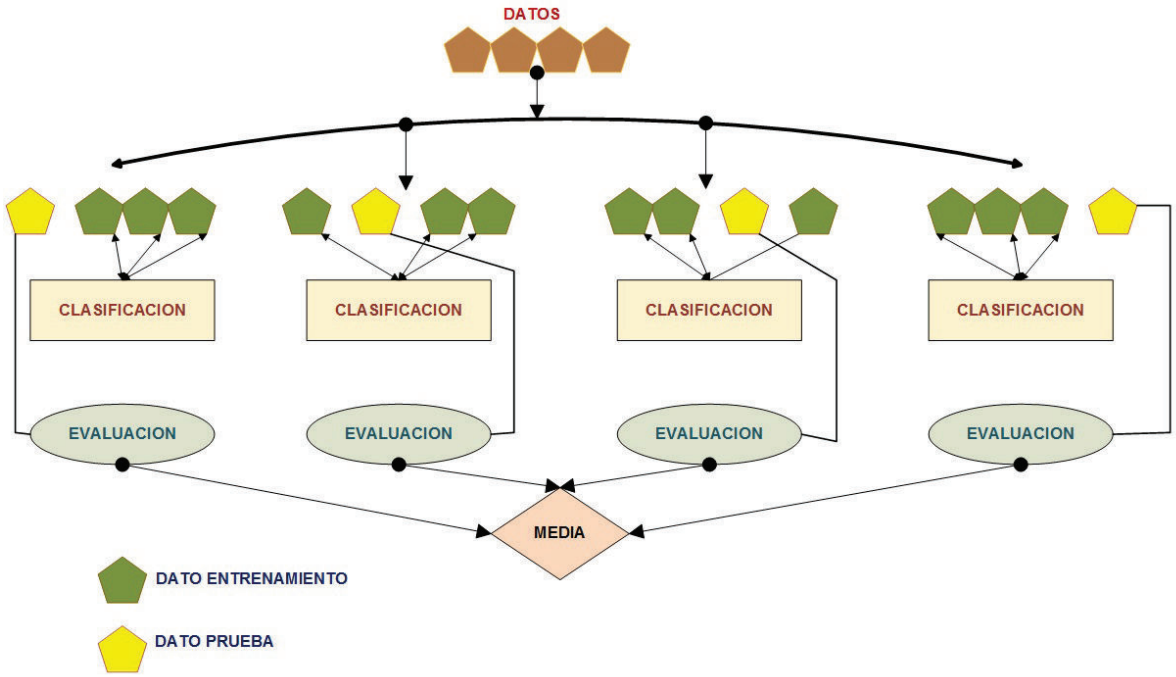


Figura 5.2.: CrossValidation Error

En la implementación presentada este trabajo, a pesar que en la mayoría de referencia bibliográficas, la validación cruzada por medio de k iteraciones aparece como un método lento, al usar Adaboost en este proceso, los tiempos de ejecución y calculo disminuyen radicalmente. Adaboost como tal contrasta el comportamiento natural de este tipo de validación cruzada al ser un algoritmo de rápida convergencia. En la simulaciones realizadas en Matlab, las ejecuciones totales de los scripts, asumiendo como punto de inicio la carga de los datos en memoria no tomaron mas allá de unos cuantos minutos. La validacion cruzada en k iteraciones, por su naturaleza y funcionamiento es fácilmente interpretable a las actividades que realizar un laboratorista al hacer análisis de laboratorio, en este caso, las k iteraciones simbolizan un trabajo realizado en k casos de estudio, en paralelo y con la ventaja de no necesitar trabajar con muestras biológicas, reduciendo el tiempo de ejecución, con resultados altamente precisos.

5.2.1 SIMULACIÓN DE ANÁLISIS EN LABORATORIO VIRTUAL

El uso de la *crossvalidation* para la evaluación de los resultados obtenidos tiene una naturaleza analógico con el comportamiento clásico de un laboratorista el cual evaluá

cultivos en la búsqueda de antígenos o para la identificación de proteínas características en la presencia de enfermedades. El laboratorista toma dos casos de análisis, el cual a su vez desde el punto de vista del *machine learning* puede ser interpretado como un clasificador binario, luego de identificados estos dos casos de análisis, el laboratorista recibe pistas de donde mirar para poder reconocer a que caso pertenecen las muestras analizadas. Estas pistas son los marcadores encontrados en la selección de características, los cuales deben representar información suficiente para poder discriminar entre dos grupos de análisis. El clasificador de igual manera con esta información busca poder discriminar de manera efectiva, sin embargo en investigaciones mas profundas y detalladas este proceso debe repetirse varias veces para poder afirmar que se poseen o no resultados fiables. Los resultados obtenidos deben ser analizados en muchos escenarios de pruebas, bajo diferentes condiciones y donde no interfieran de ninguna manera los resultados anteriores. Este tipo de comportamiento es lo que mas se acerca al procedimiento de la evaluación de clasificadores usando *crossvalidation*.

Existen ciertos criterios que muestran que la validación cruzada solo es fiable con mediciones que han sido tomadas del mismo conjunto de datos del caso estudiado. En este caso teniendo en cuenta que la metodología presentada asume los datos desde la carga en memoria de la plataforma computacional, este problema se ve descartado ya que los marcadores encontrados a la salida del algoritmo se enfocaran en los datos de ingreso como tal, sin tener que verse afectados por otro tipo de parámetros. La aplicación de estos datos a la definición de biomarcadores a su vez es validada de forma aleatoria por tres métodos independientes con resultados similares.

5.2.2 RECEIVER OPERATING CHARACTERISTIC - ROC

La Receiver Operating Characteristic (Característica Operativa del Receptor) es una representación de la sensibilidad de un clasificador binario según se varia el umbral de discriminación. Mientras mas se acerque la curva a sus ejes, el área bajo esta tendera a ser 1, el clasificador por tanto, tendera a tener un error de clasificación igual a cero. Una interpretación alternativa de la ROC es que representa el radio de verdaderos positivos(VPR) frente a la razón de falsos positivos (FPR). Este tipo de herramientas de evaluación de clasificadores permiten tener una visualización mas clara del rendimiento del clasificador usado para realizar diagnósticos. En la Figura 5.4 se muestra una curva ROC y sus zonas de trabajo.

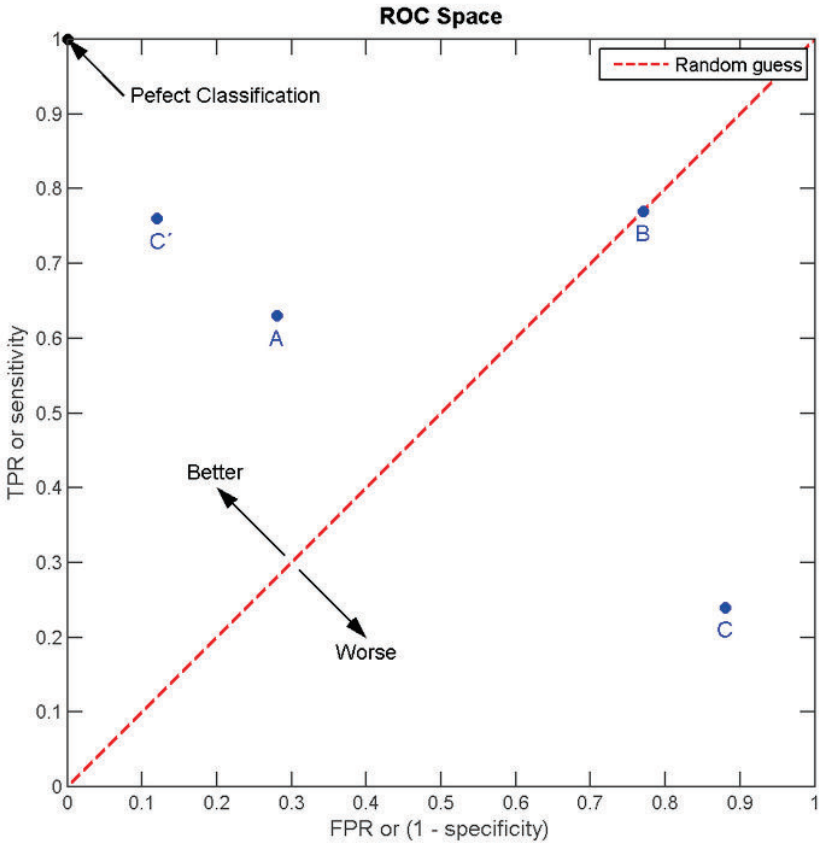


Figura 5.3.: Zonas de trabajo de la Curva ROC

La VPR mide hasta que punto el clasificador es capaz de detectar correctamente a que clase corresponde la muestra analizada, en tanto que la FPR mide cuantos de estos posibles casos son incorrectos. Una forma fácil de interpretar esta curva se presenta a continuación en la Figura 5.4, donde de izquiera a derecha se presenta un buen rendimiento del clasificador evaluado, pasando por un caso medio, hasta un caso de un bajo rendimiento del clasificador evaluado.

5.3 VALIDACIÓN DE RESULTADOS USANDO MUESTRAS EXTERNAS

Para evitar los efectos a doble ciego, adicionalmente a las metodologías explicadas en este capitulo se realizaron también pruebas de muestras externas. Estas pruebas básicamente usan muestras etiquetadas, las cuales son entregadas al clasificador modelado sin sus respectivas etiquetas, una vez que el clasificador ha recibido los datos,

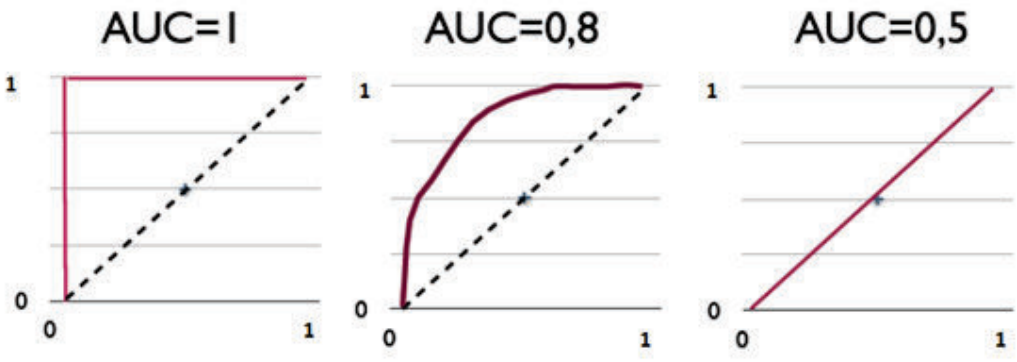


Figura 5.4.: Casos de Análisis - Curvas ROC

se extraen del clasificador las etiquetas resultantes y se comparan con las etiquetas originales evaluando un factor de error por medio de la Ecuación 5.1:

$$\epsilon = \frac{\text{correctamente clasificados}}{\text{no correctamente clasificados} + \text{correctamente clasificados}} \tag{5.1}$$

Por la naturaleza de las aplicaciones del algoritmo implementado se separaron los datos en subconjuntos dejando uno para esta etapa. Esta metodología garantiza malas interpretaciones de resultados excesivamente optimistas, los cuales, en las aplicaciones dadas pueden ser mortales.

CAPÍTULO 6

Pruebas y Resultados del Algoritmo Propuesto

En este capitulo se presentan los resultados de las simulaciones realizadas usando el algoritmo implementado. Se ha dividido este capitulo en tres secciones, una por cada conjunto de datos analizado con el algoritmo. Cada sección a su vez presenta los resultados en tres etapas, una para el procesamiento, otra para la selección de características y una para la validación de los resultados de la selección.

6.1 CONJUNTO ARCENE

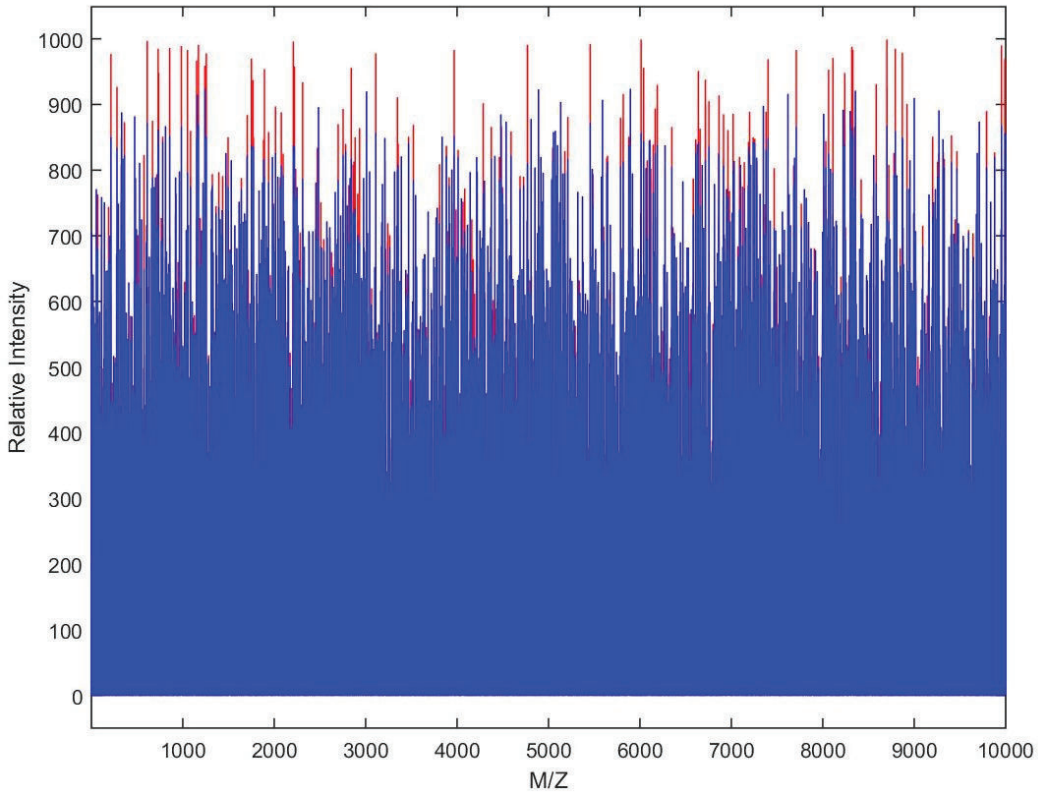


Figura 6.1.: Conjunto de Datos Arcene(I vs m/z)

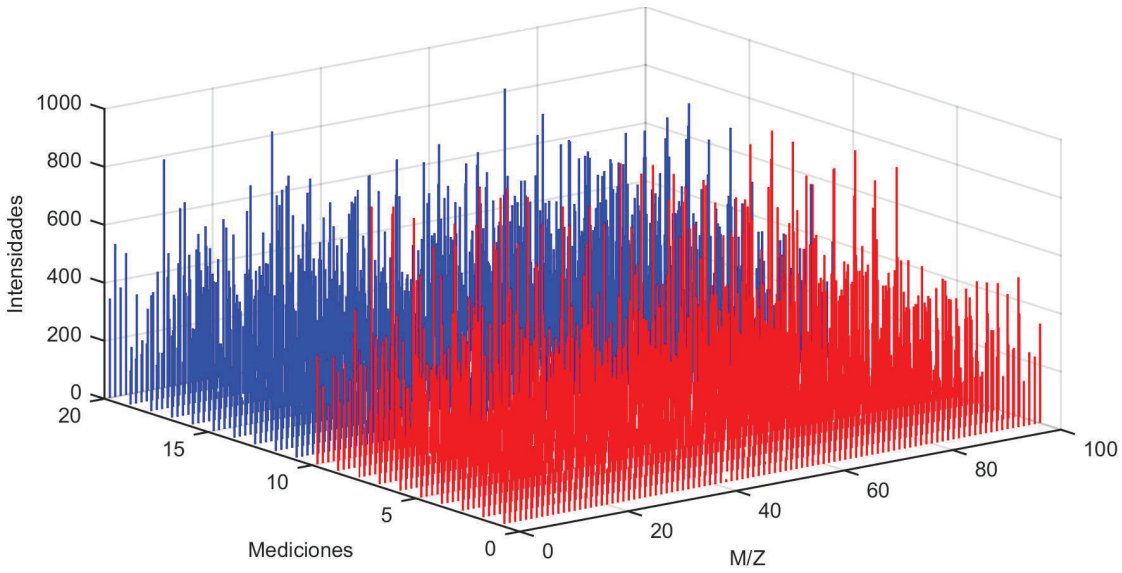


Figura 6.2.: Conjunto de Datos Arcene(I vs m/z - 3D)

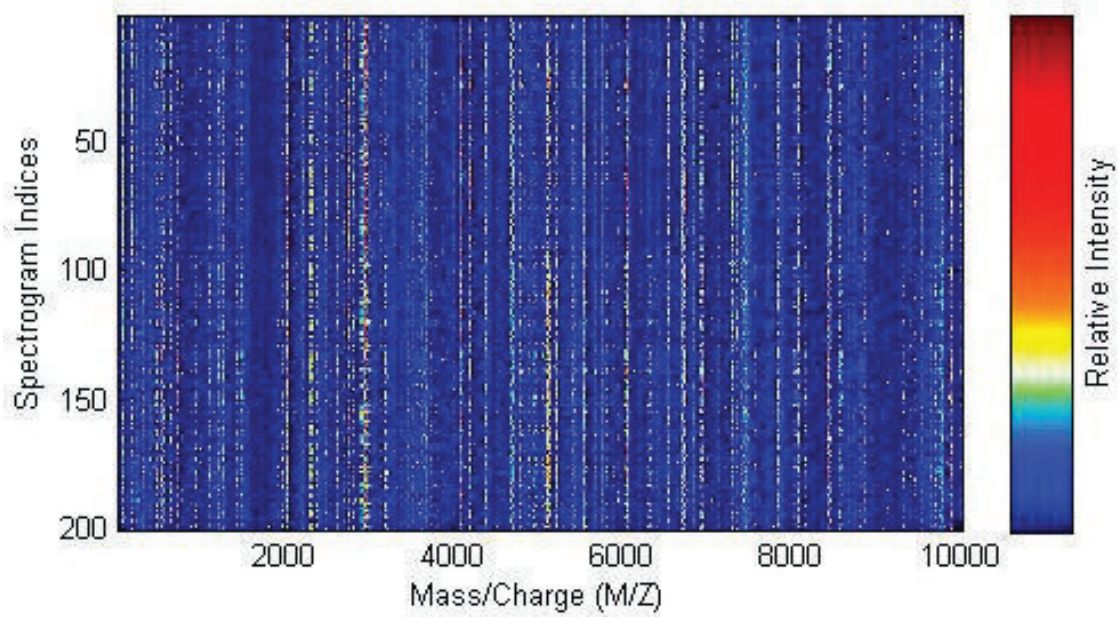


Figura 6.3.: Conjunto de Datos Arcene(Mapa de Calor)

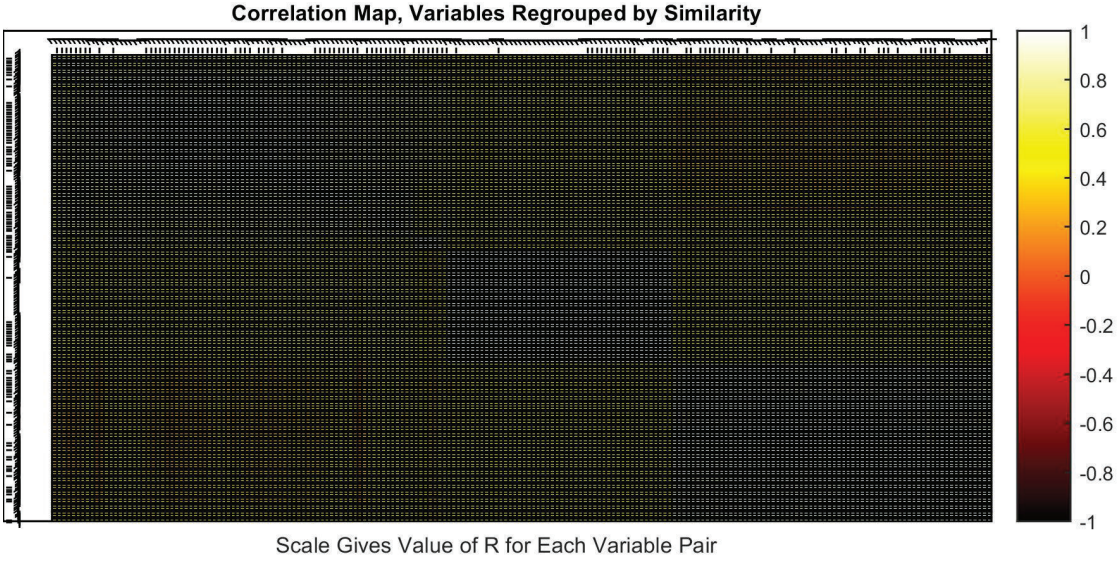


Figura 6.4.: Mapa de Correlación Datos ARCENE (Cancer vs Control)

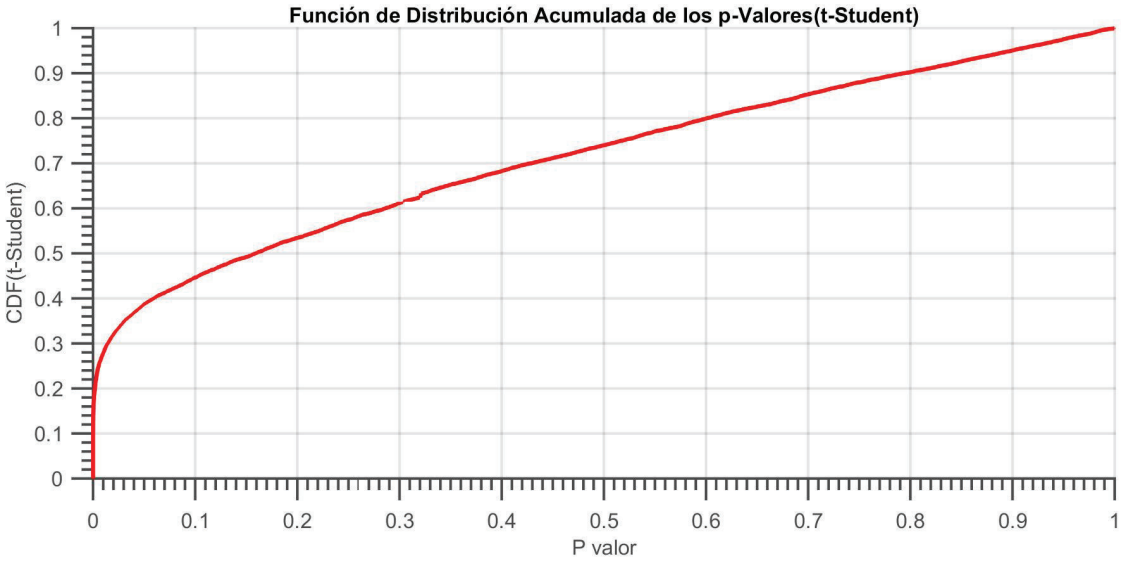


Figura 6.5.: Función Empírica de Probabilidad - Conjunto de Datos ARCENE (t-Student)

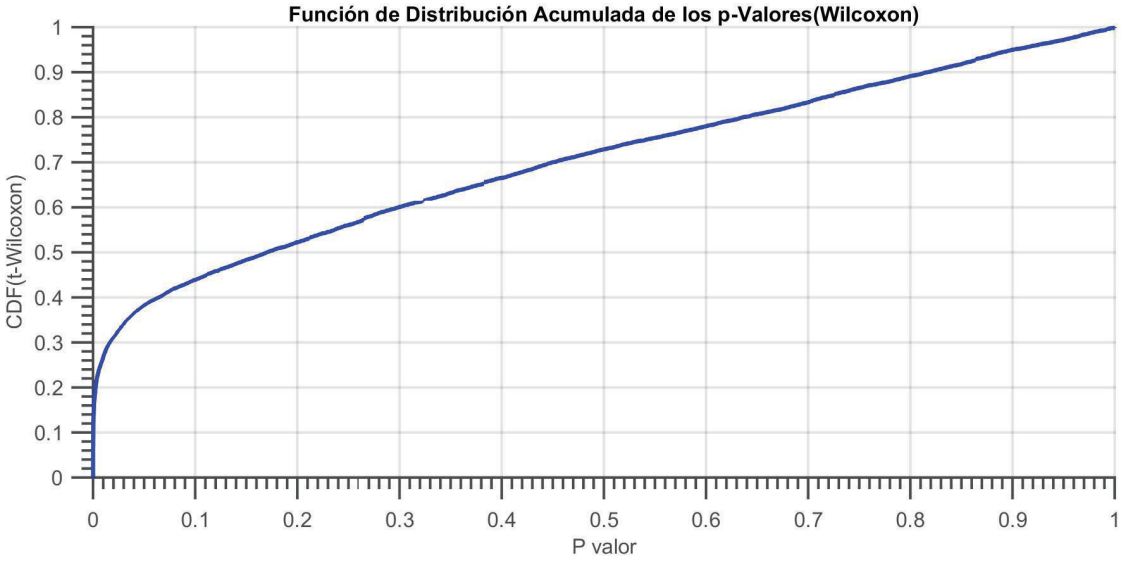


Figura 6.6.: Función Empírica de Probabilidad - Conjunto de Datos ARCENE (Mann–Whitney U test)

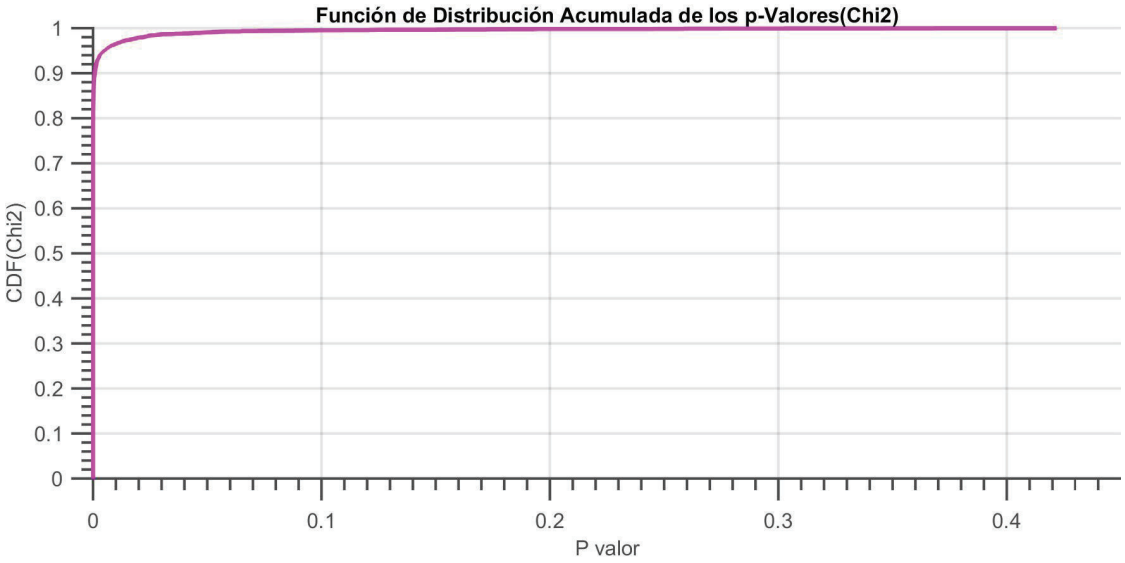


Figura 6.7.: Función Empírica de Probabilidad - Conjunto de Datos ARCENE (χ^2)

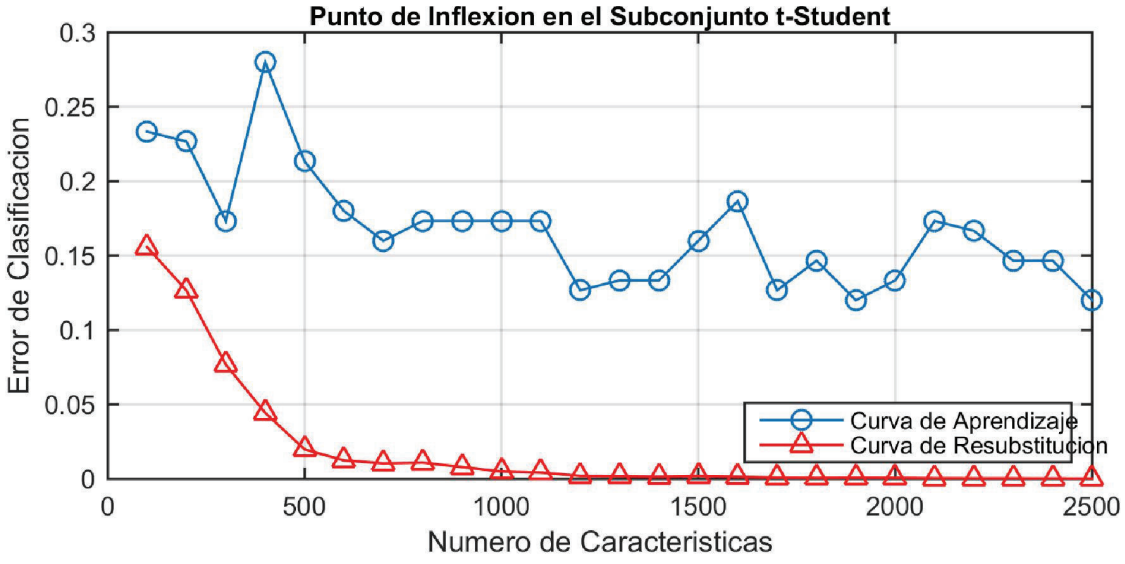


Figura 6.8.: Punto de Inflexión usando filtro t-Student

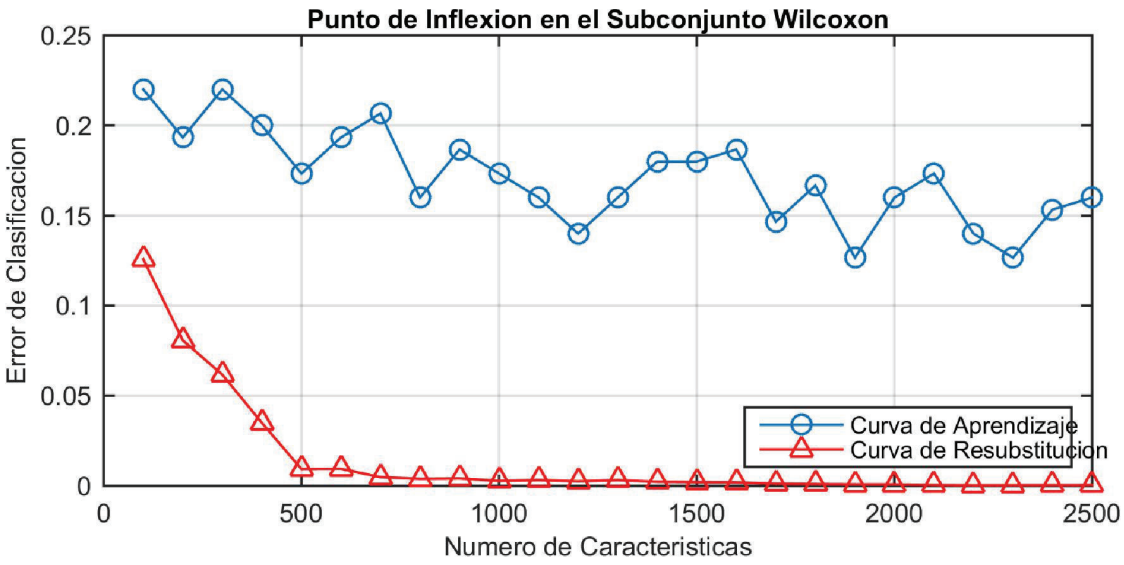


Figura 6.9.: Punto de Inflexión usando filtro Mann–Whitney U test

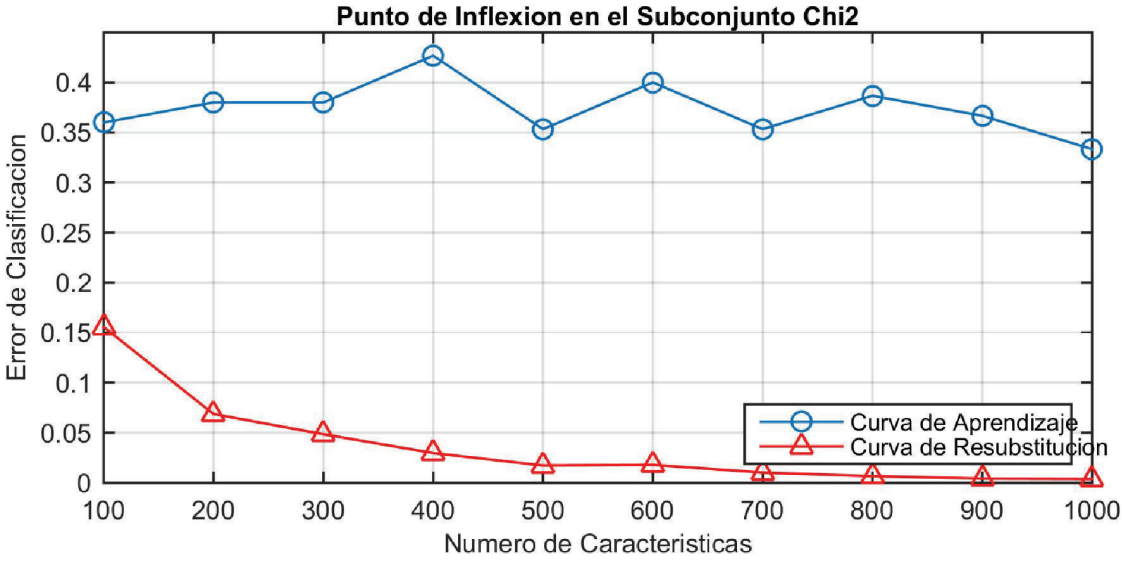


Figura 6.10.: Punto de Inflexion usando filtro χ^2

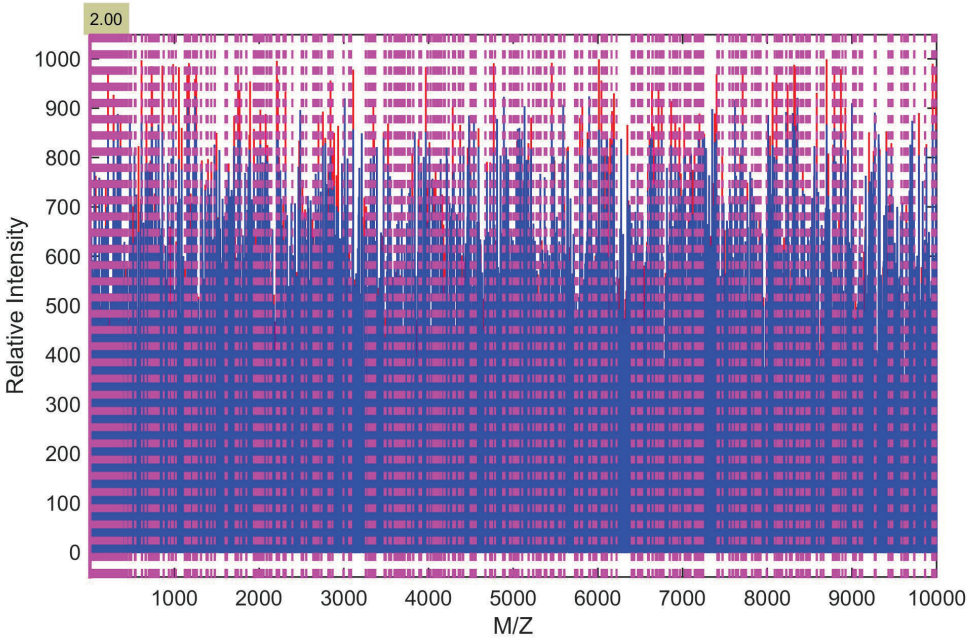


Figura 6.11.: Características detectadas con Marcadores Redundantes Sin Filtrar en Conjunto Arcene(mz vs I)

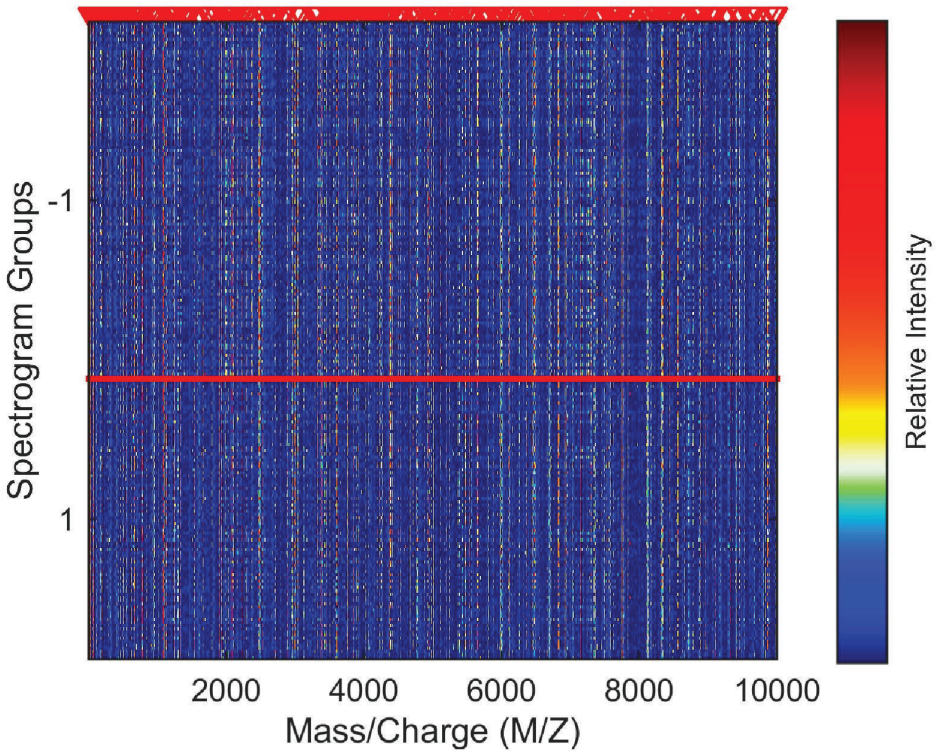


Figura 6.12.: Características detectadas con Marcadores Redundantes Sin Filtrar en Conjunto Arcene(Mapa de Calor)

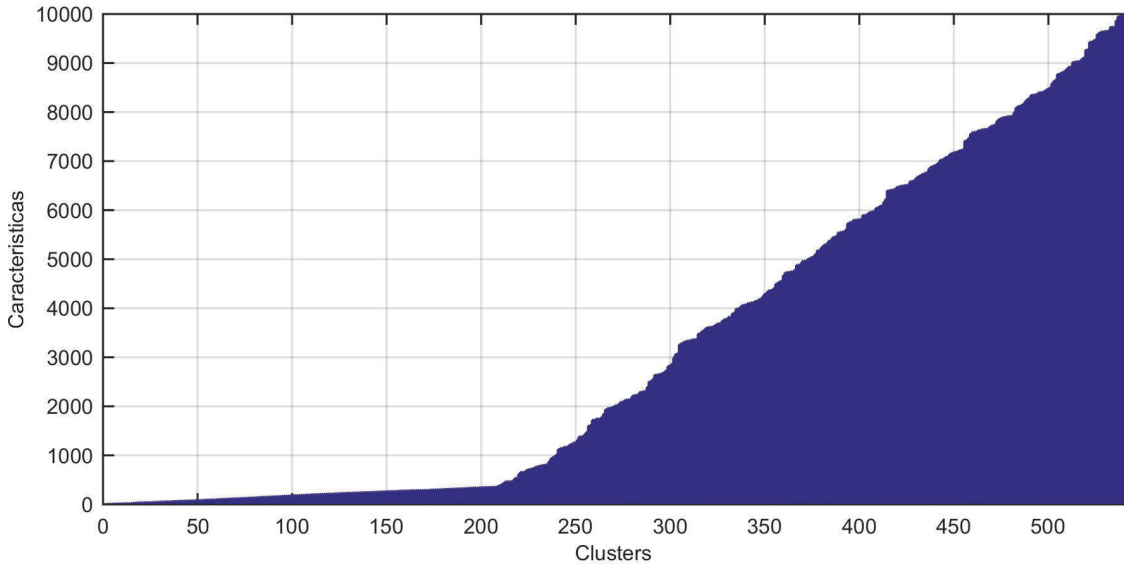


Figura 6.13.: Clusters - Redundancia de Marcadores en Conjunto ARCENE

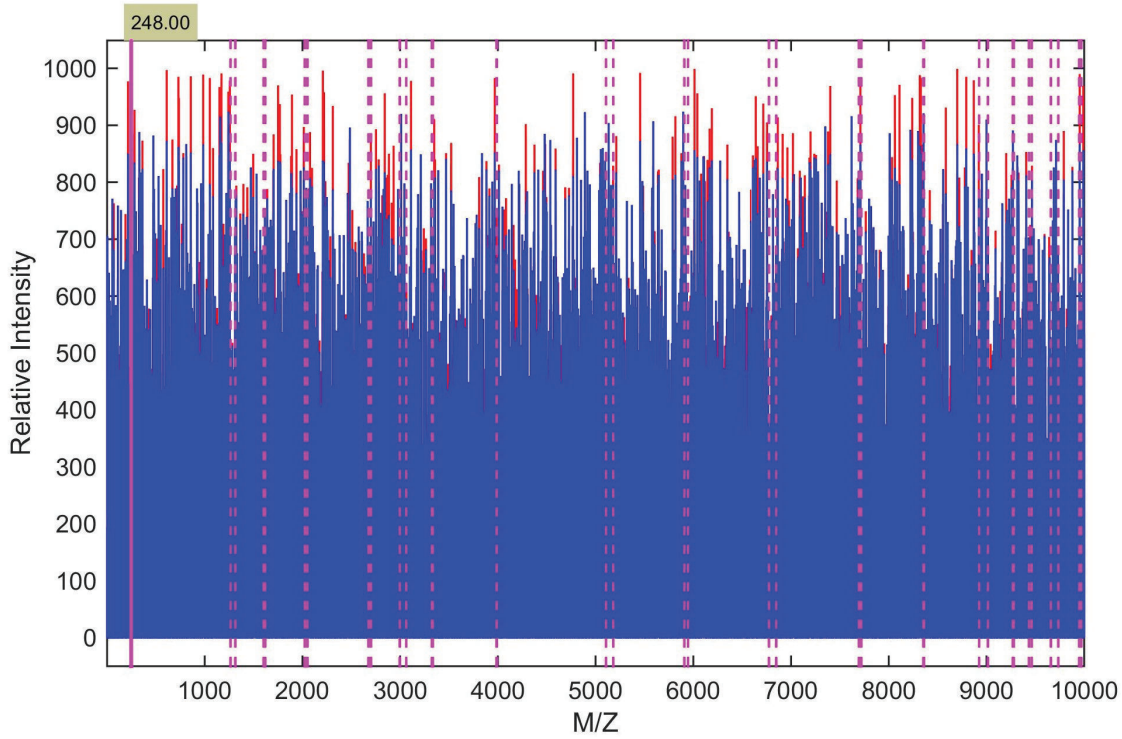


Figura 6.14.: Características Filtras en Conjunto Arcene(mz vs I)

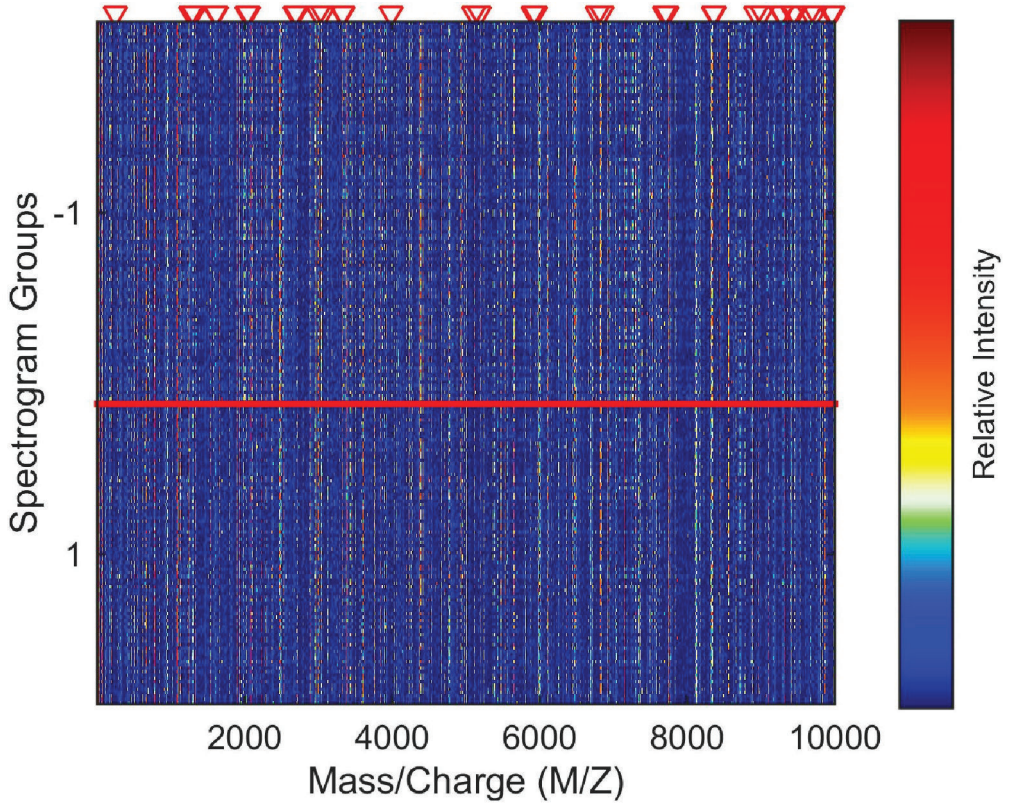


Figura 6.15.: Características Filtradas en Conjunto Arcene(mapa de calor)

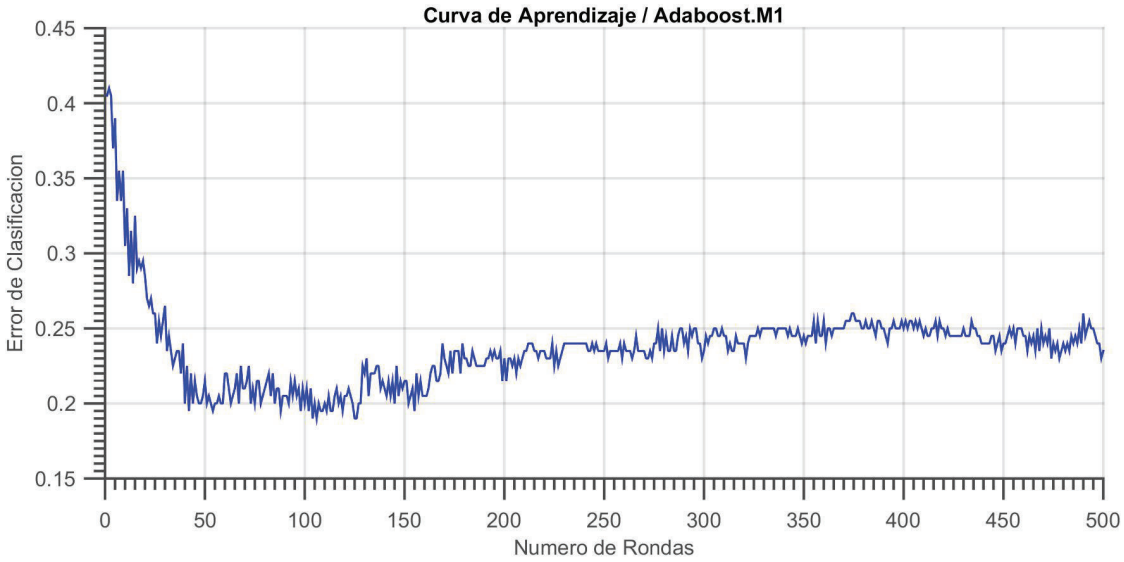


Figura 6.16.: Evaluación de las Características Seleccionadas usando *Crossvalidation* en Adaboost M1

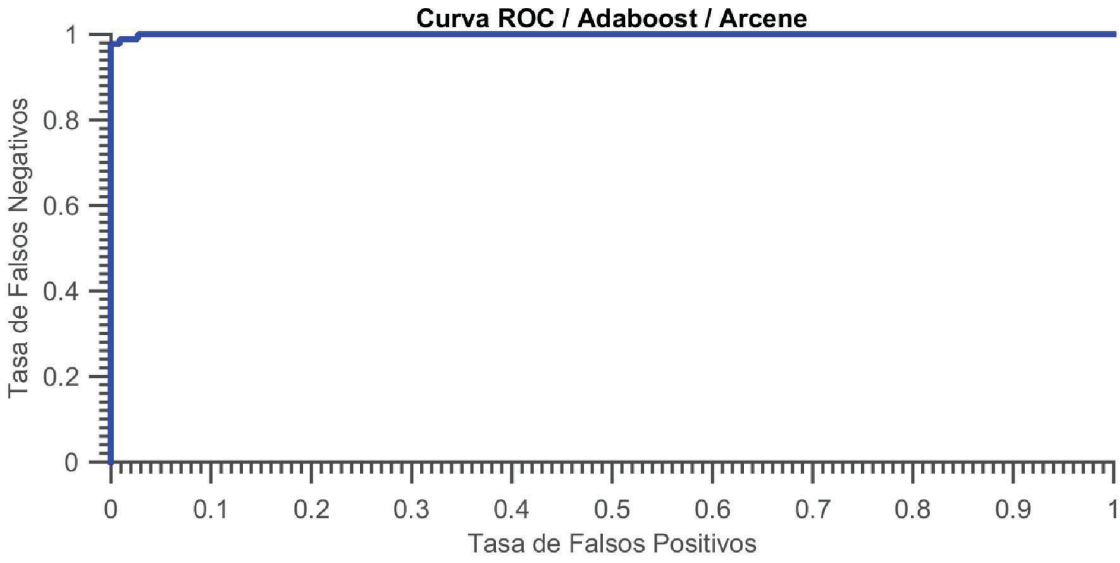


Figura 6.17.: Evaluación del Rendimiento de Clasificación usando las Características Seleccionadas - Curva Receiver Operating Characteristic(ROC)

6.2 CONJUNTO OVARIAN CANCER QA-QC

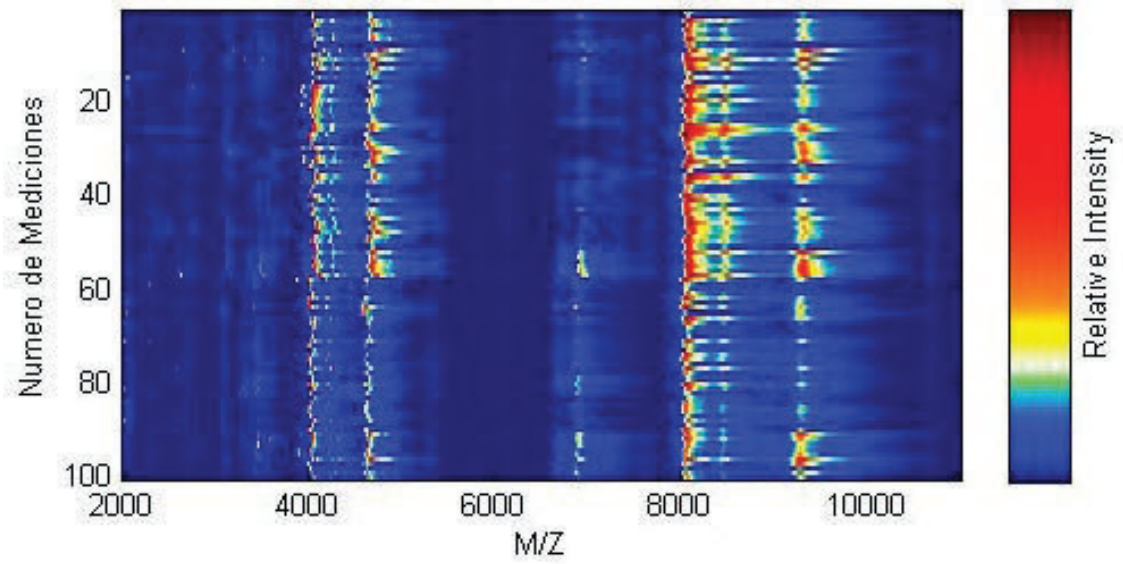


Figura 6.18.: Remuestreo de Mediciones - Conjunto *Ovarian cancer QA-QC*

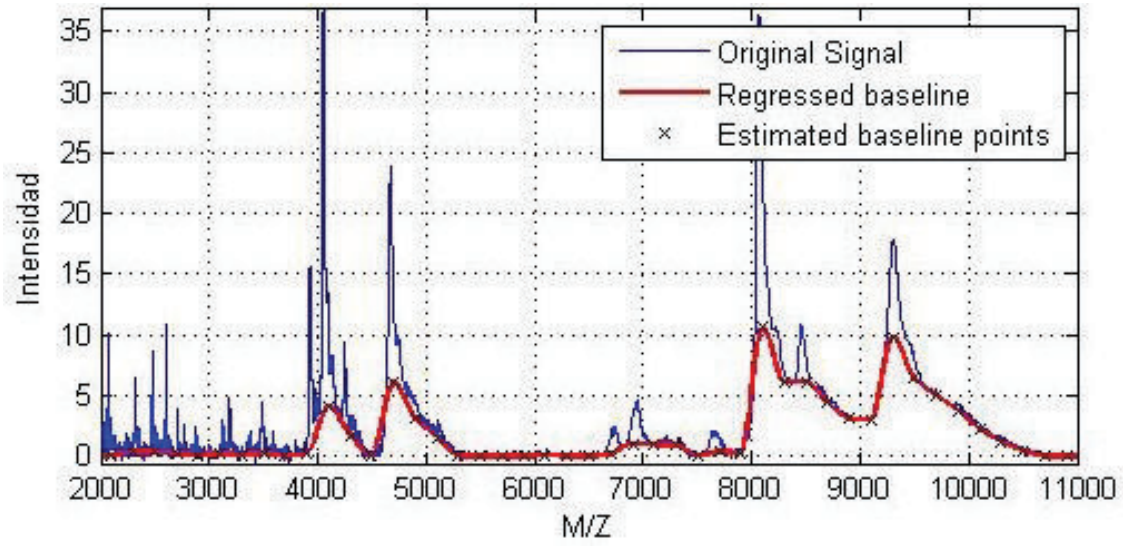


Figura 6.19.: Corrección de Linea de Base en Mediciones - Conjunto *Ovarian cancer QA-QC*

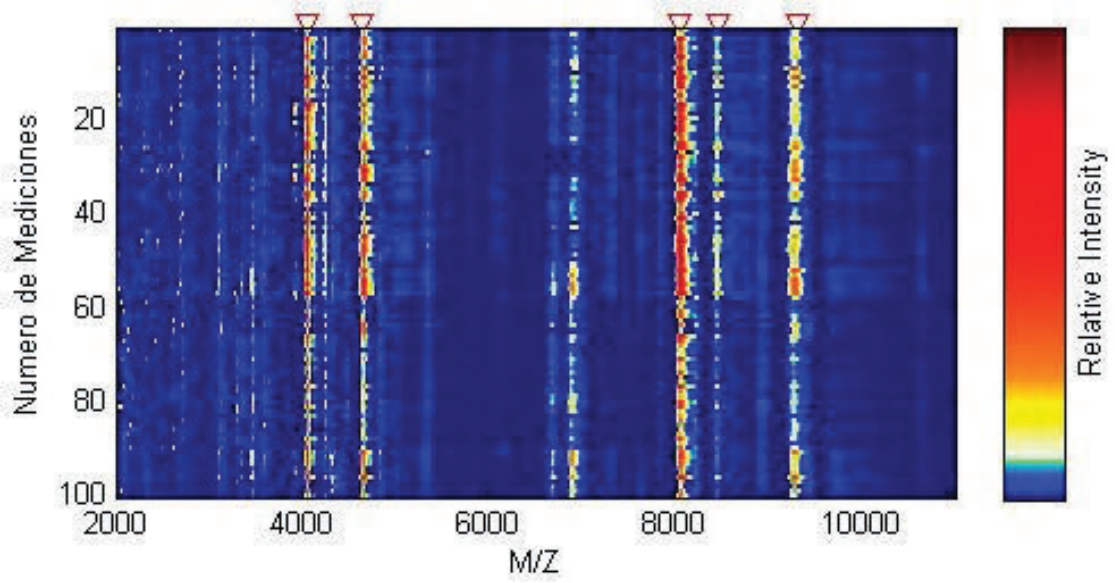


Figura 6.20.: Alineación de Mediciones - Conjunto *Ovarian cancer QA-QC*

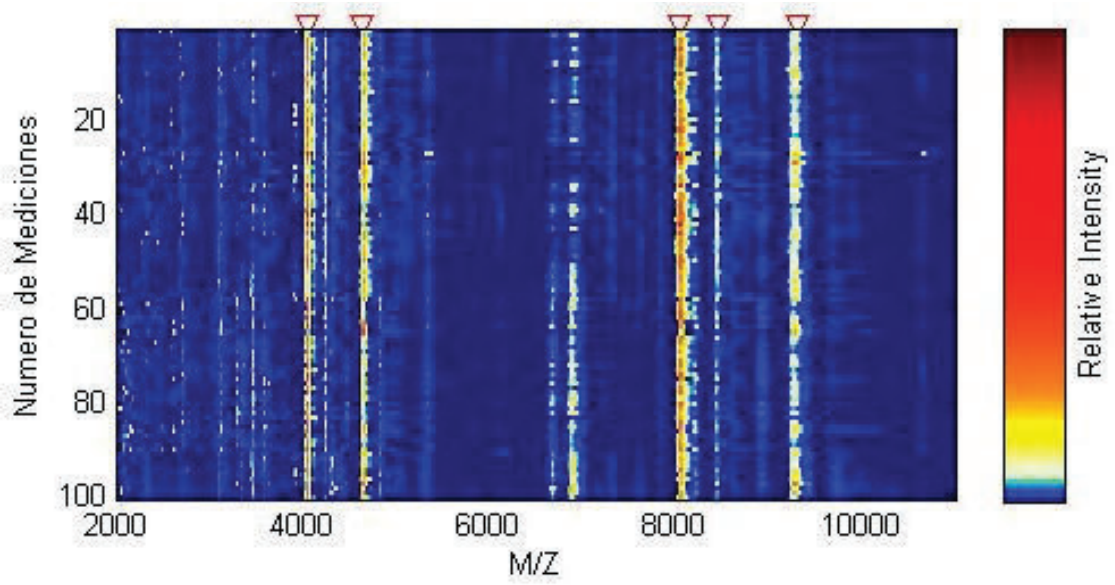


Figura 6.21.: Normalización de Mediciones - Conjunto *Ovarian cancer QA-QC*

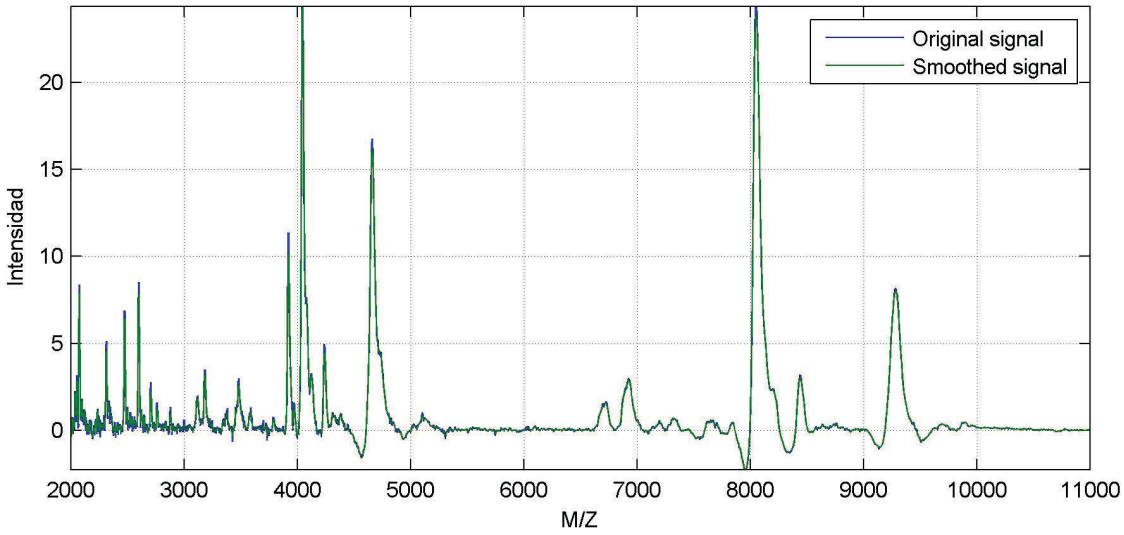


Figura 6.22.: Suavizamiento de Ruido en Mediciones - Conjunto *Ovarian cancer QA-QC*

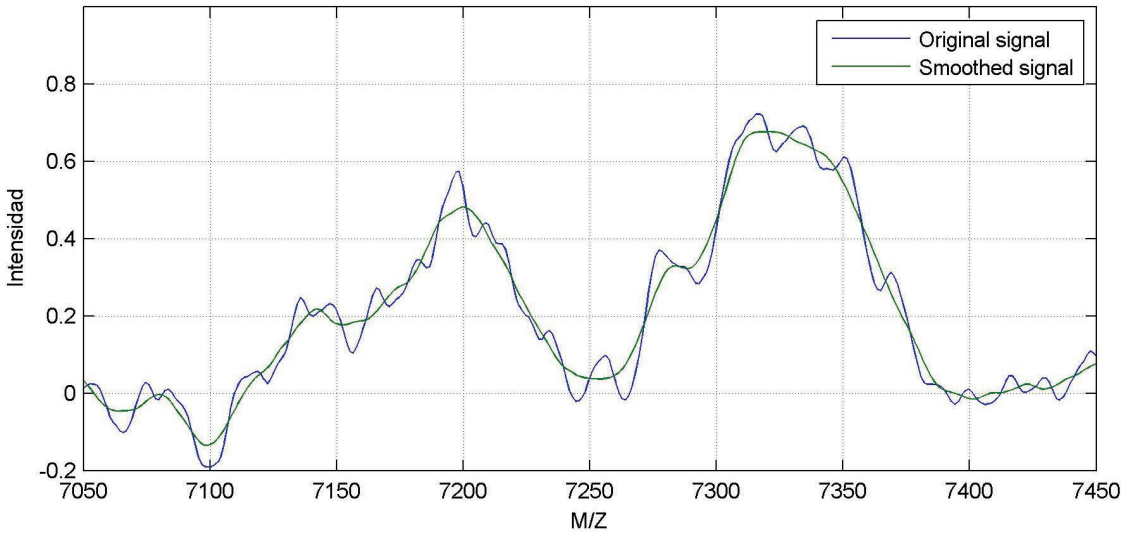


Figura 6.23.: Suavizamiento de Ruido en Mediciones - Zoom - Conjunto *Ovarian cancer QA-QC*

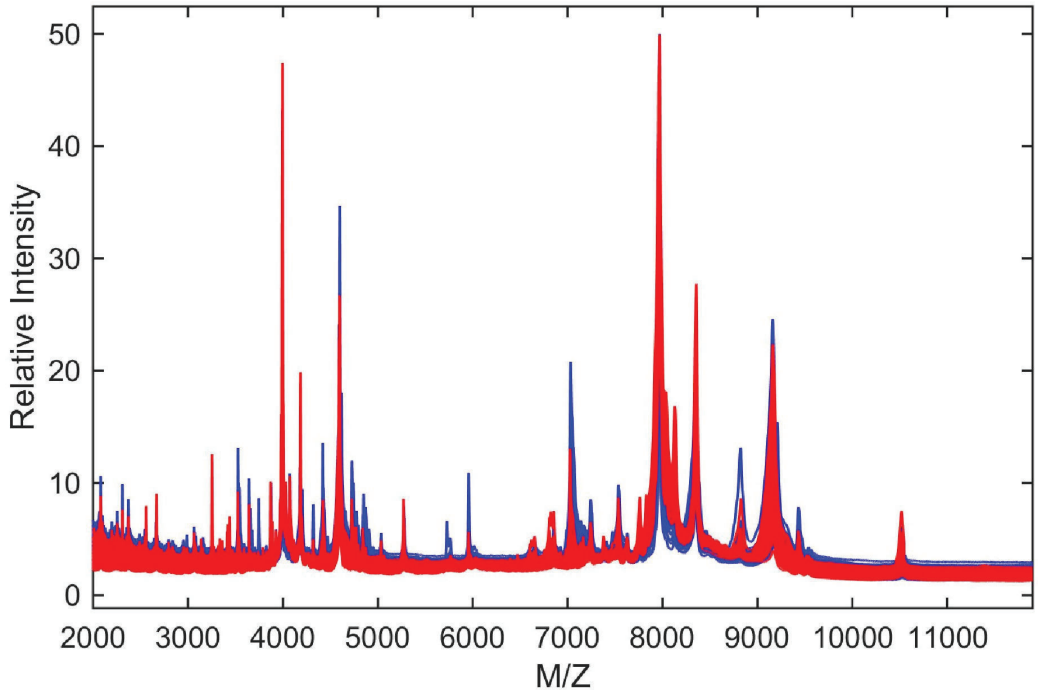


Figura 6.24.: Conjunto *Ovarian cancer QA-QC* - Visualización en 2D

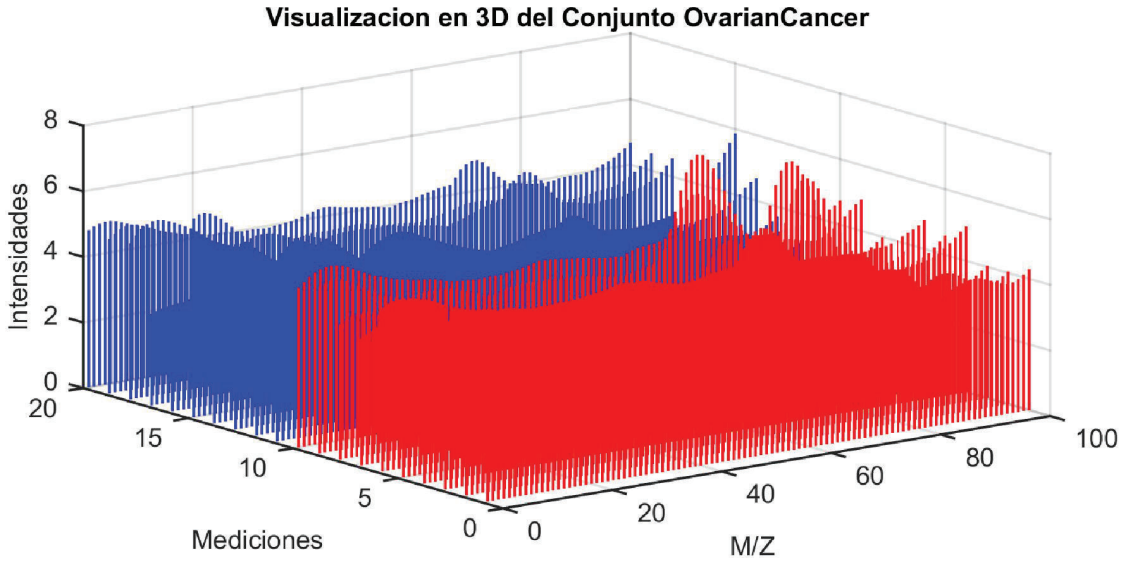


Figura 6.25.: Conjunto *Ovarian cancer QA-QC* - Visualización en 3D

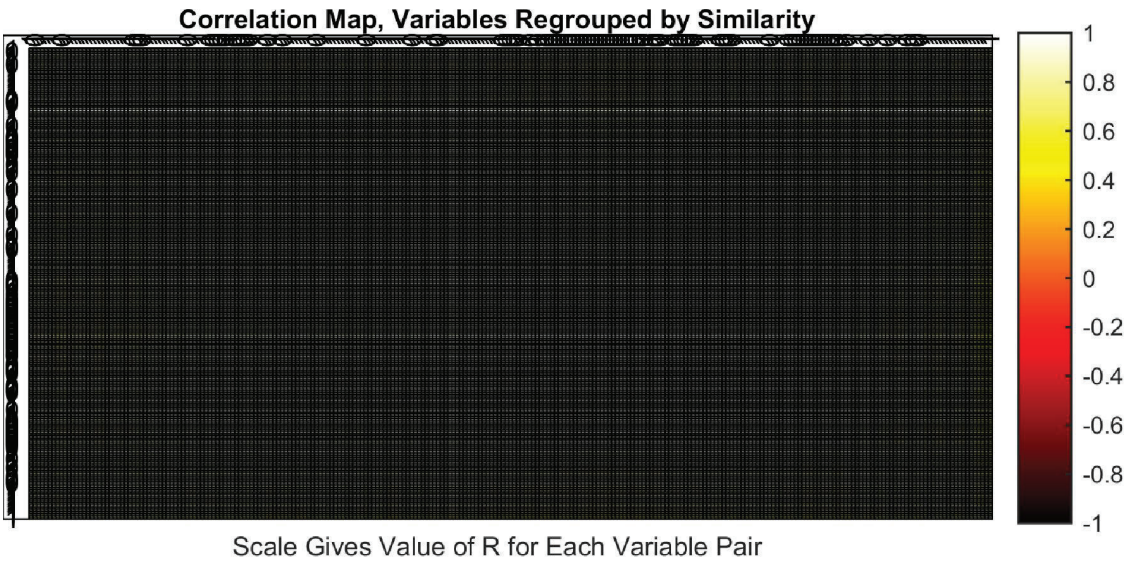


Figura 6.26.: Correlación - Conjunto *Ovarian cancer QA-QC*(Cancer vs Control)

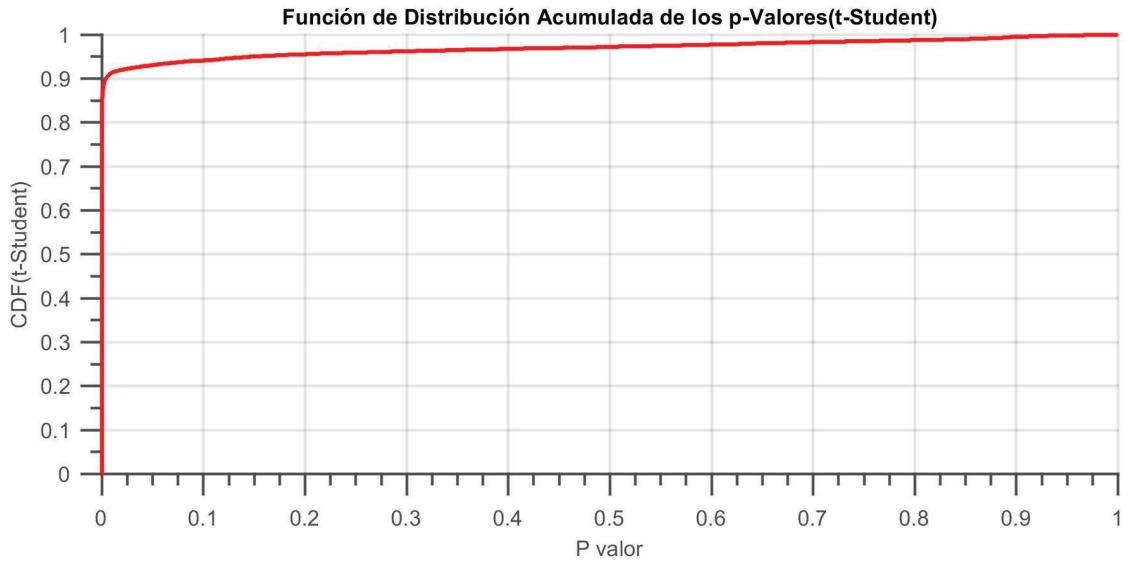


Figura 6.27.: Función Empírica de Probabilidad - Conjunto *Ovarian cancer QA-QC* (t-Student)

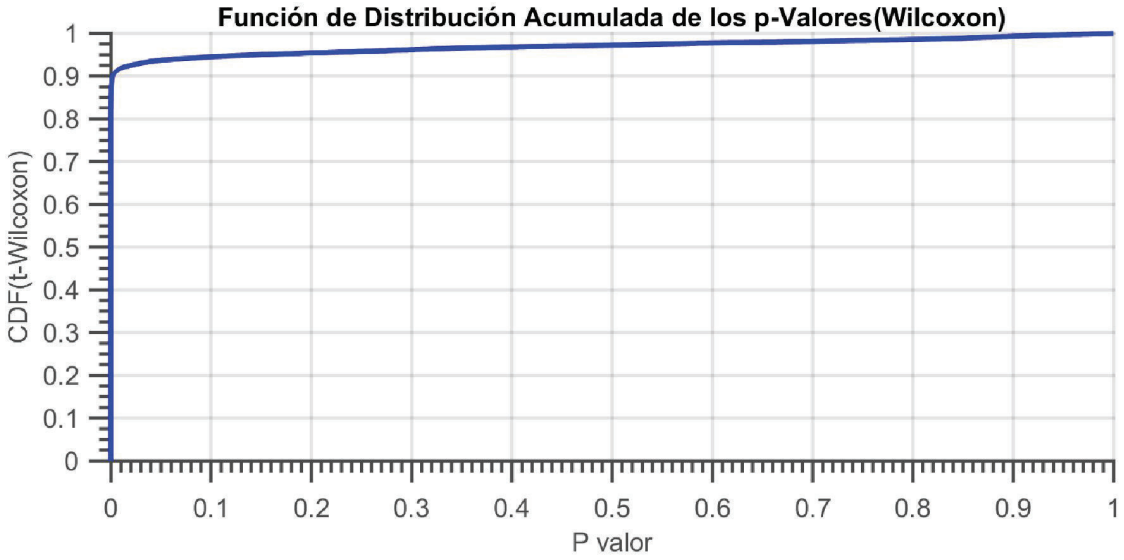


Figura 6.28.: Función Empírica de Probabilidad - Conjunto *Ovarian cancer QA-QC* (Mann–Whitney U test)

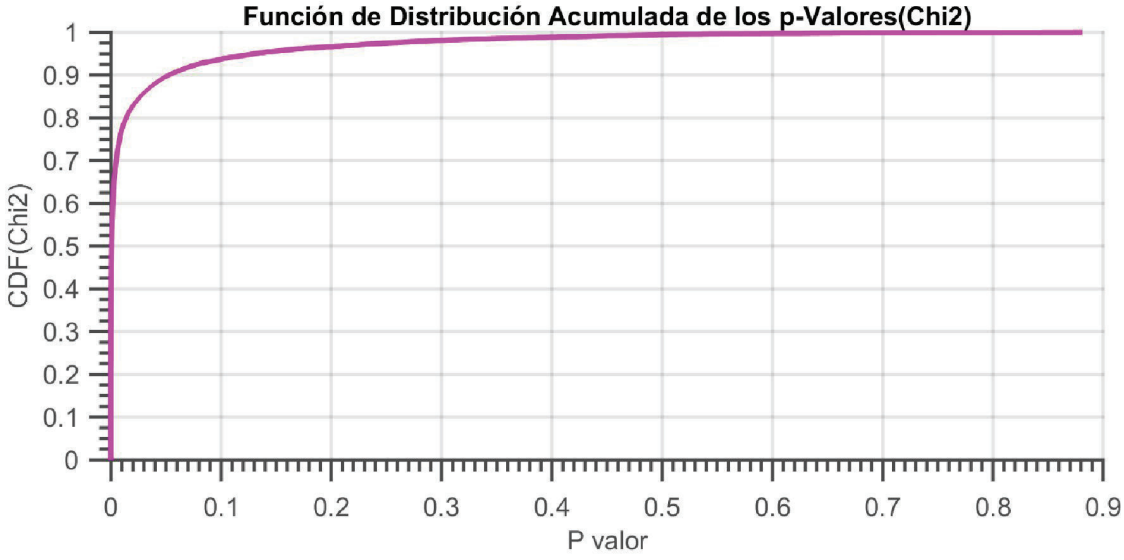


Figura 6.29.: Función Empírica de Probabilidad - Conjunto *Ovarian cancer QA-QC* (χ^2)

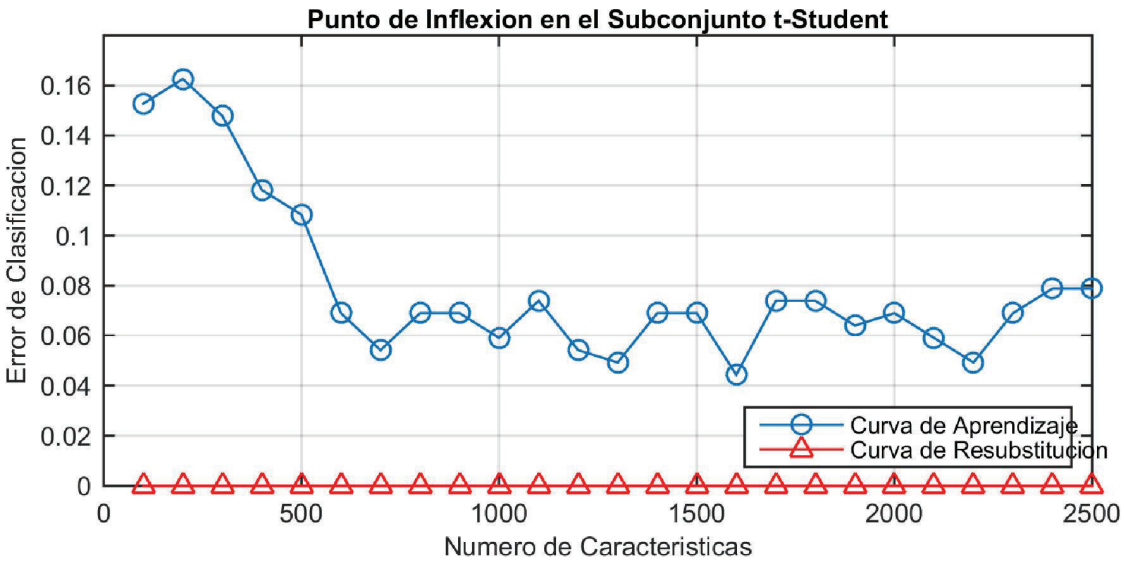


Figura 6.30.: Punto de Inflexión usando filtro t-Student

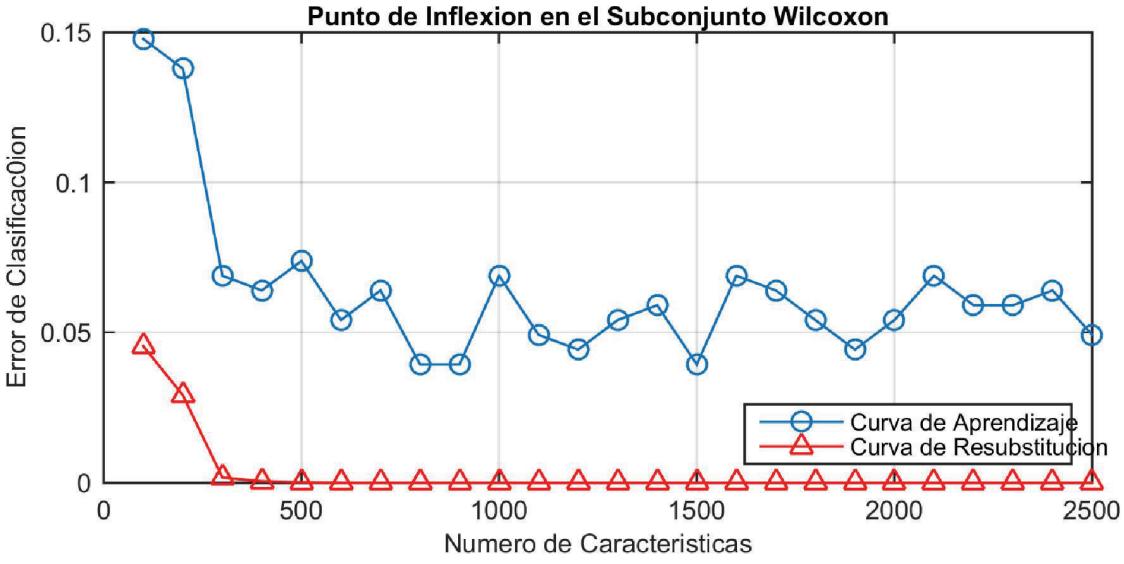


Figura 6.31.: Punto de Inflexión usando filtro Mann–Whitney U test

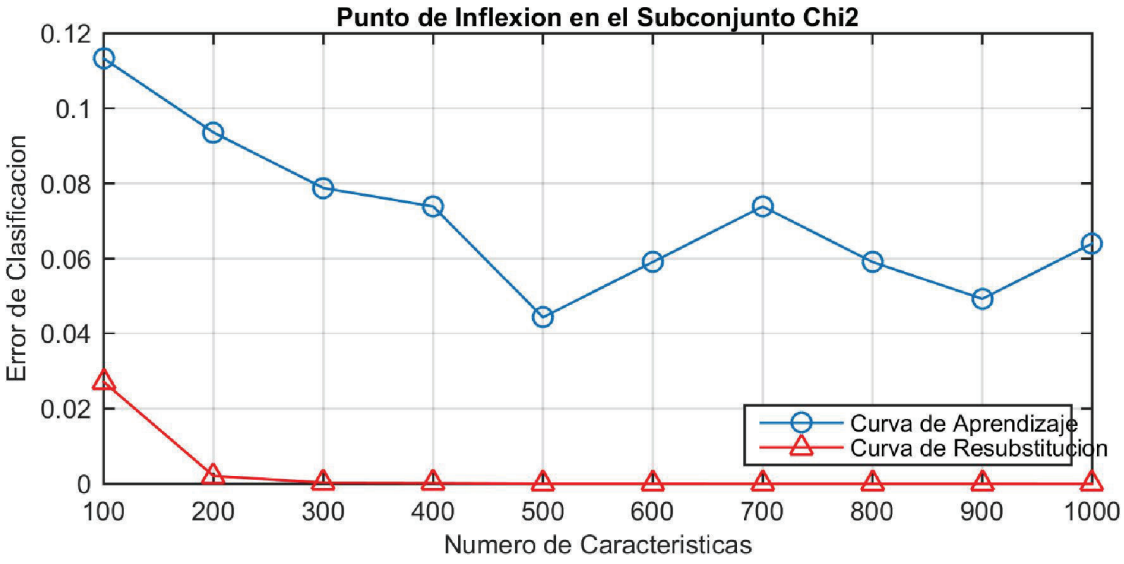


Figura 6.32.: Punto de Inflexión usando filtro χ^2

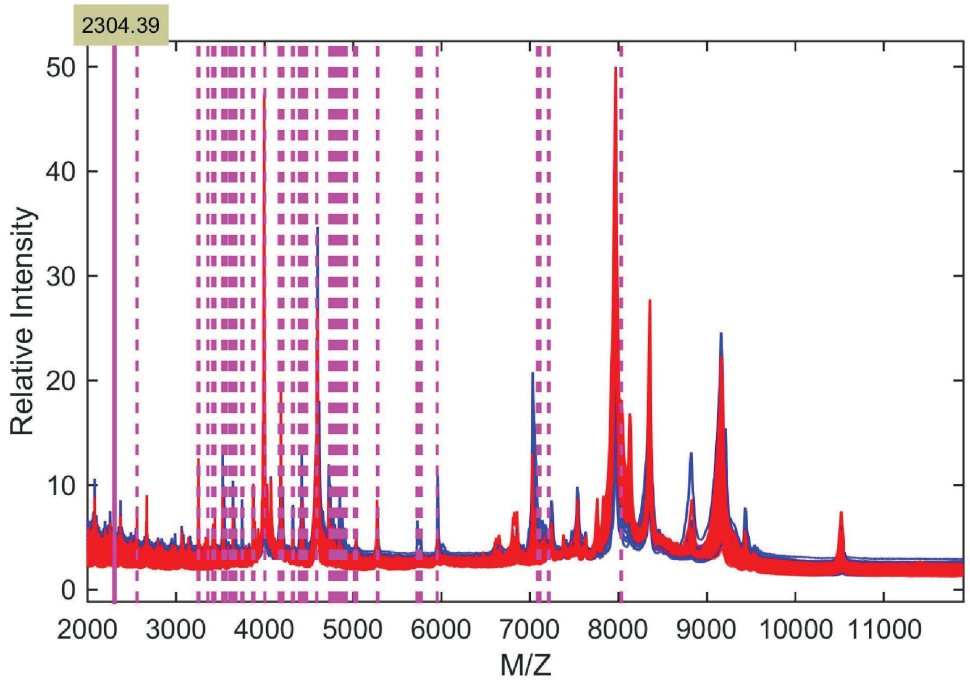


Figura 6.33.: Características detectadas con Marcadores Redundantes Sin Filtrar en Conjunto *Ovarian cancer QA-QC*(mz vs I)

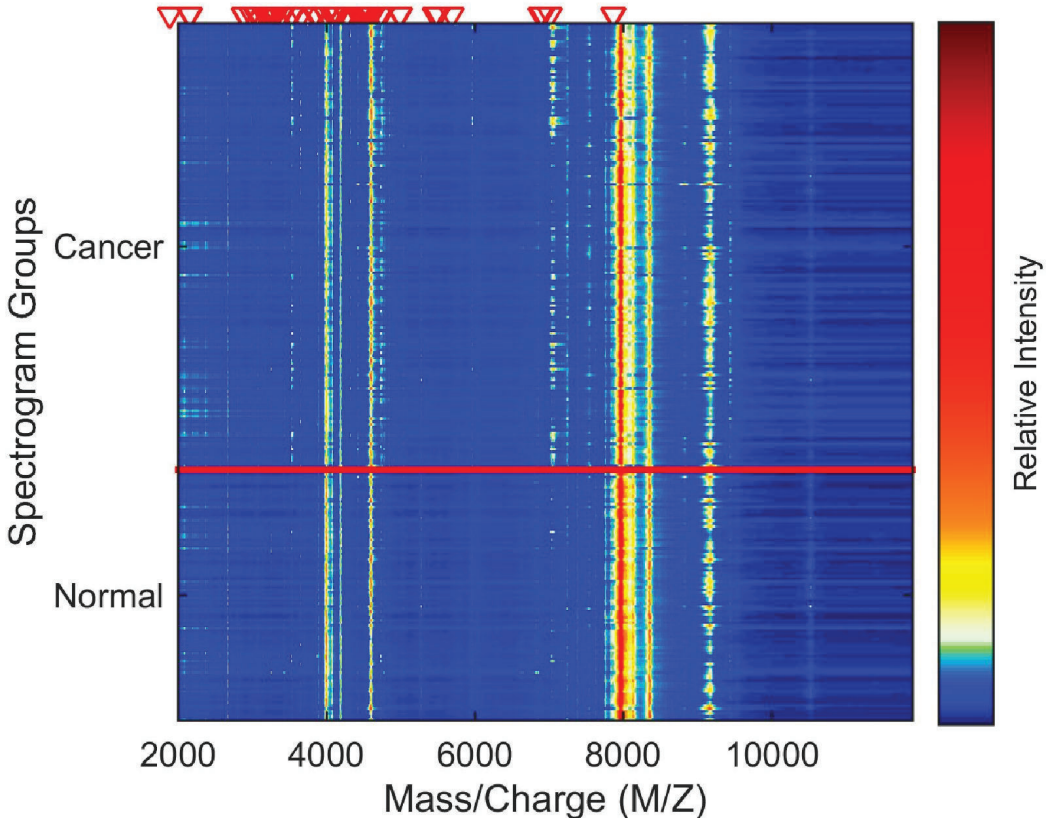


Figura 6.34.: Conjunto *Ovarian cancer QA-QC* sin filtrar en mapa de calor

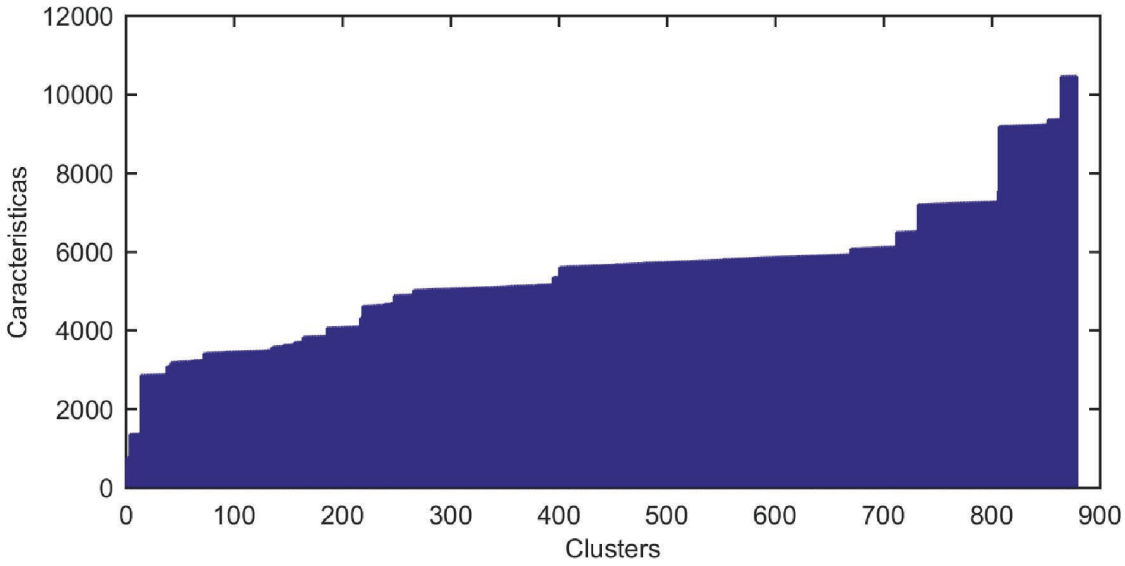


Figura 6.35.: Clusters - Redundancia de Marcadores en Conjunto *Ovarian cancer QA-QC*

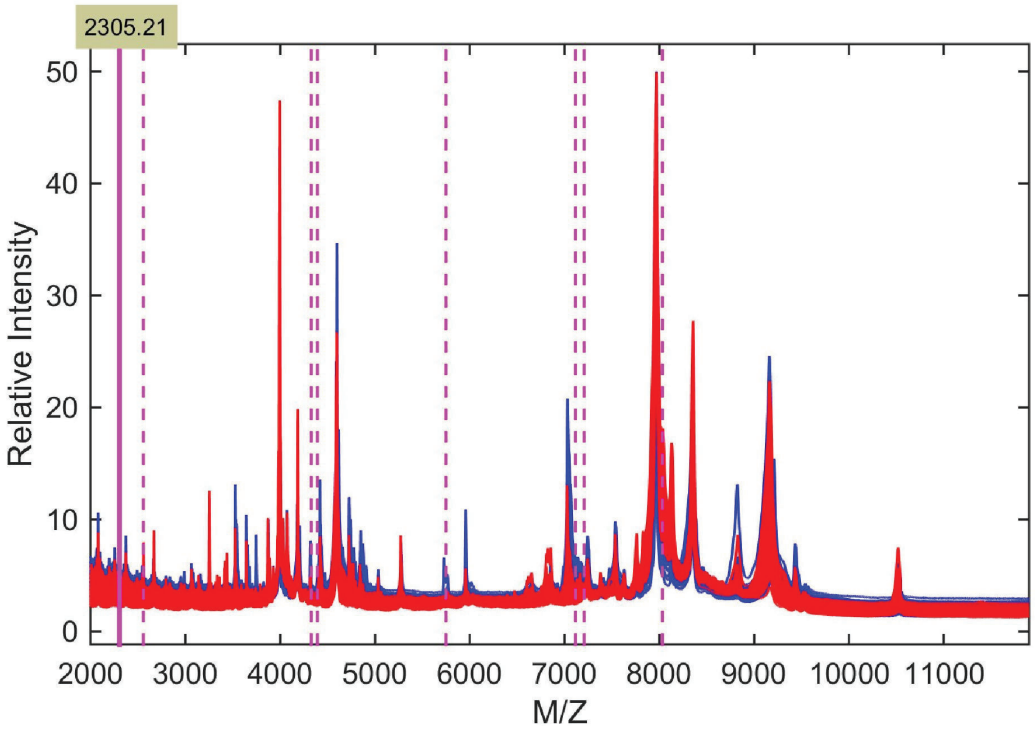


Figura 6.36.: Características detectadas filtradas en Conjunto *Ovarian cancer QA-QC*(mz vs I)

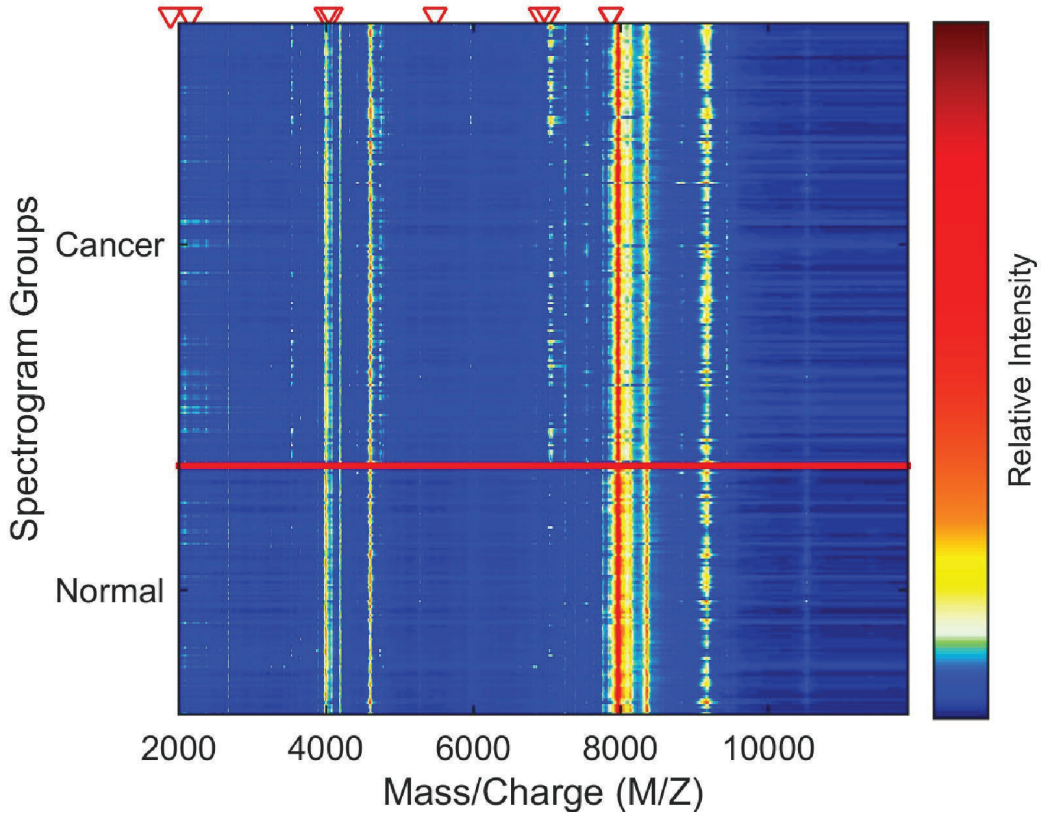


Figura 6.37.: Conjunto *Ovarian cancer QA-QC* filtradas en mapa de calor

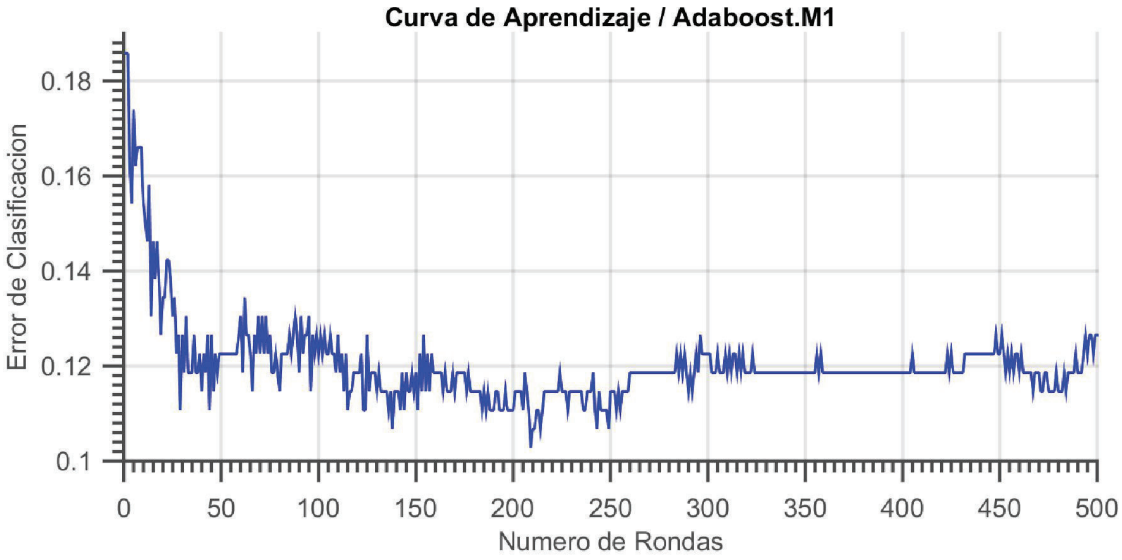


Figura 6.38.: Evaluación de las Características Seleccionadas usando *Crossvalidation* en Adaboost M1

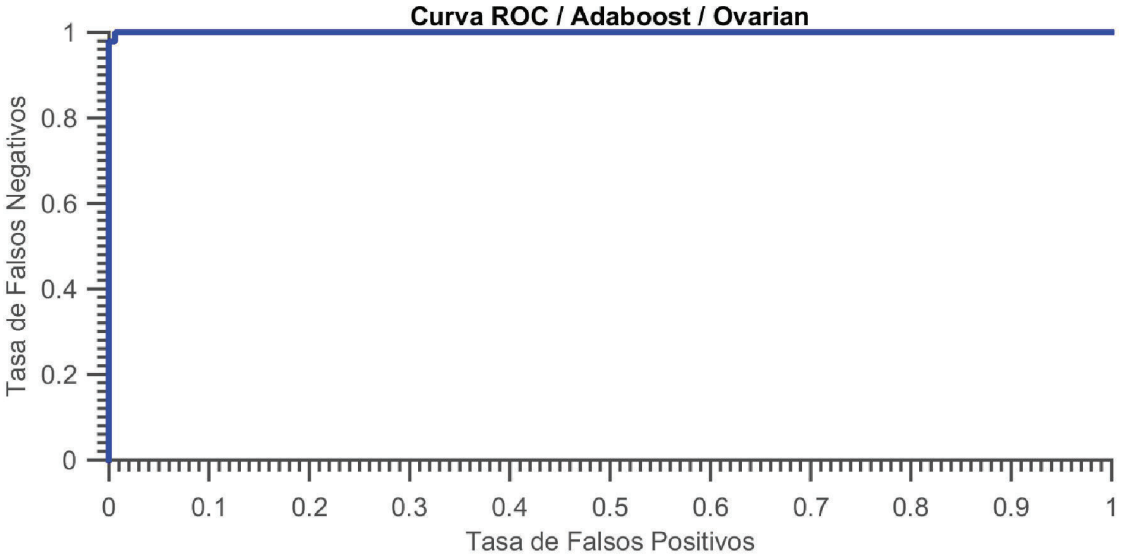


Figura 6.39.: Evaluación del Rendimiento de Clasificación usando las Características Seleccionadas - Curva Receiver Operating Characteristic(ROC)

6.3 CONJUNTO OVARIANDATASET8-7-02

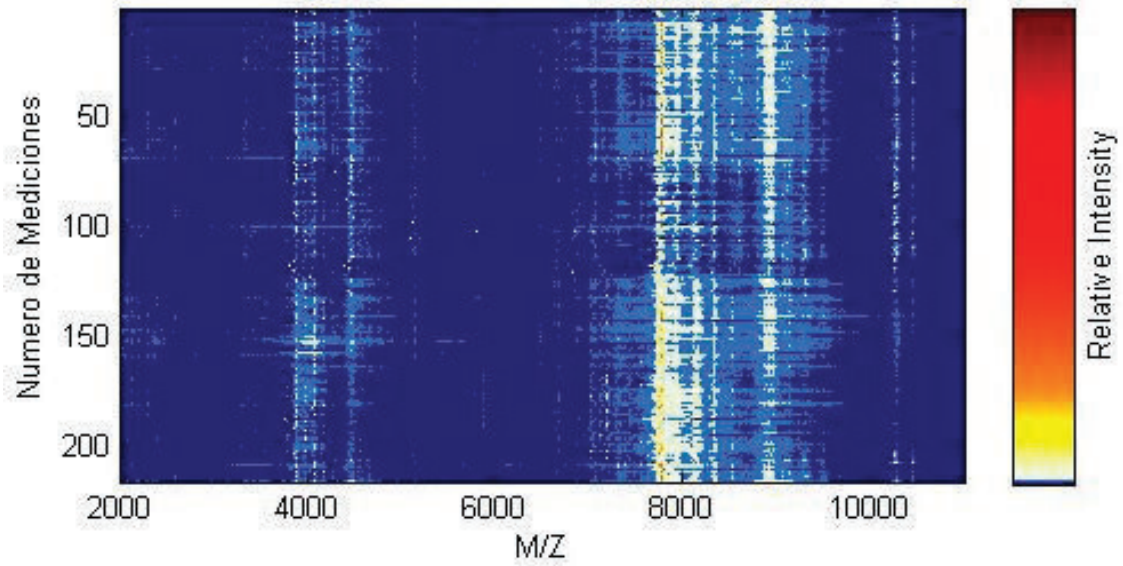


Figura 6.40.: Remuestreo de Mediciones - Conjunto OvarianDataset8-7-02

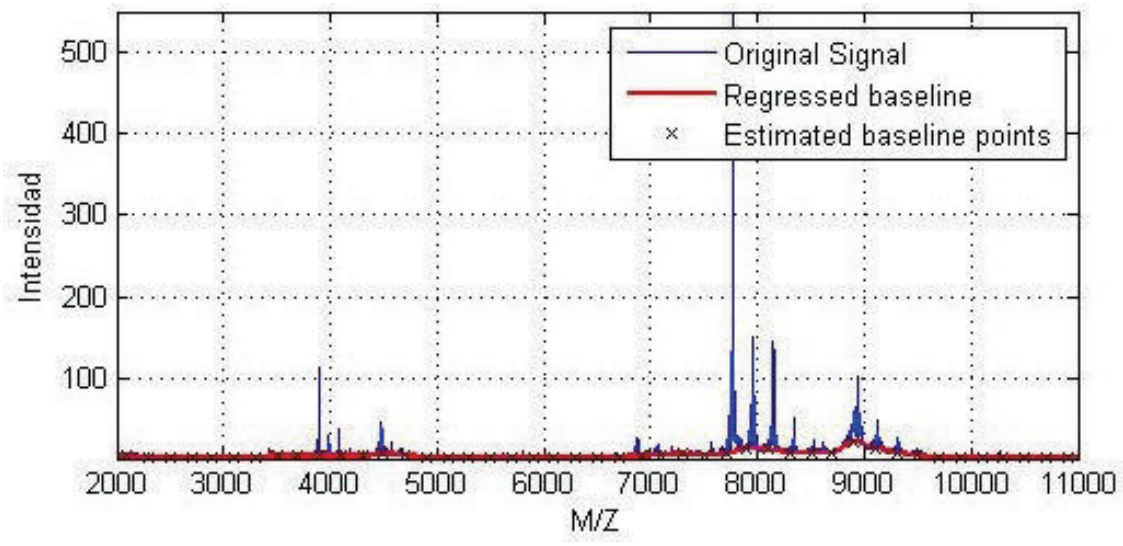


Figura 6.41.: Corrección de Linea de Base en Mediciones - Conjunto OvarianDataset8-7-02

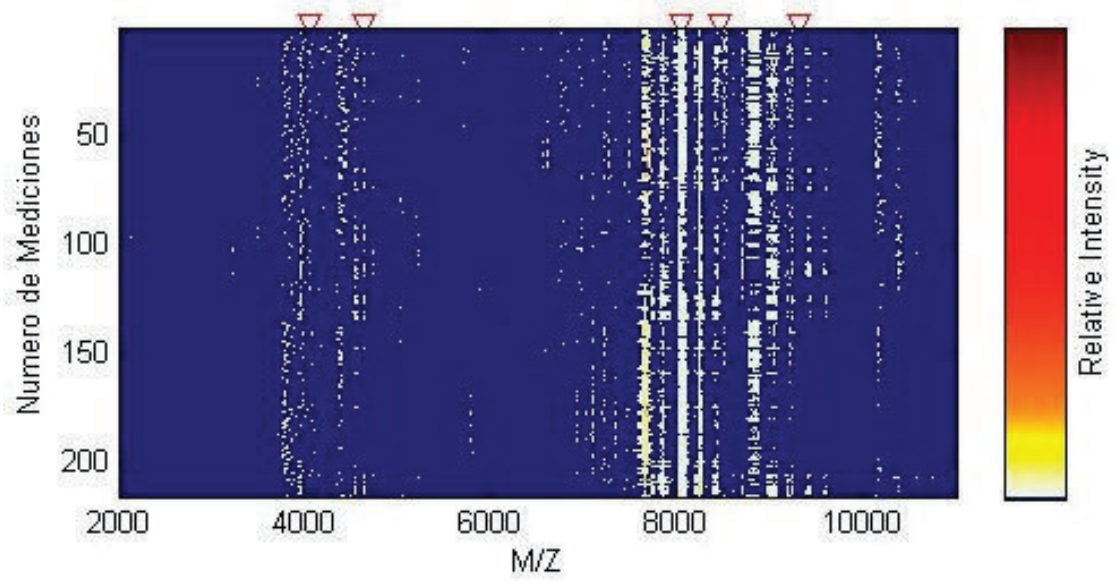


Figura 6.42.: Alineación de Mediciones - Conjunto OvarianDataset8-7-02

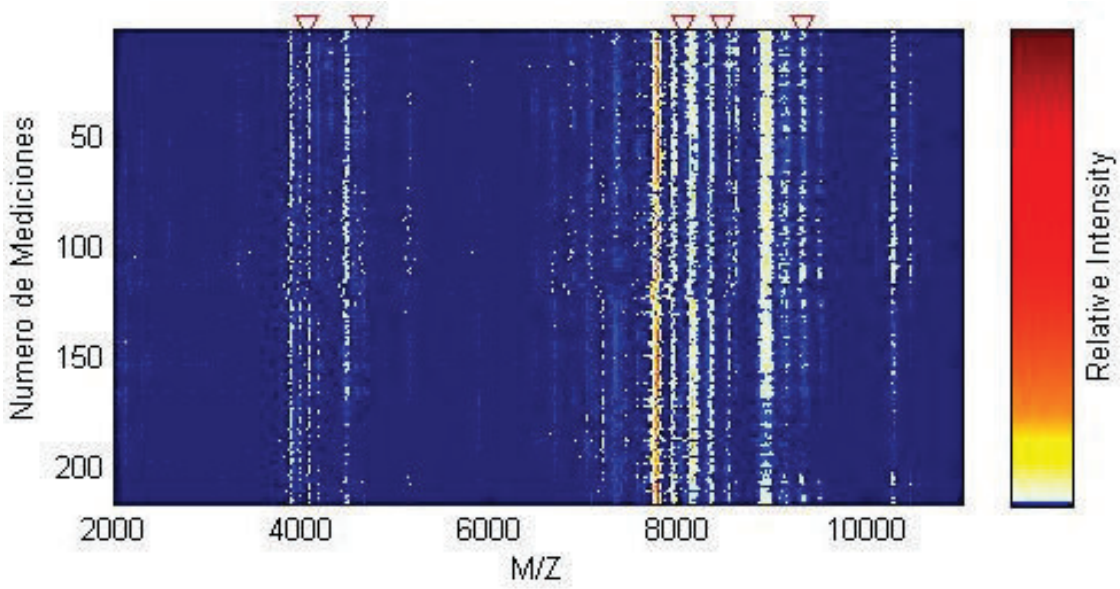


Figura 6.43.: Normalización de Mediciones - Conjunto OvarianDataset8-7-02

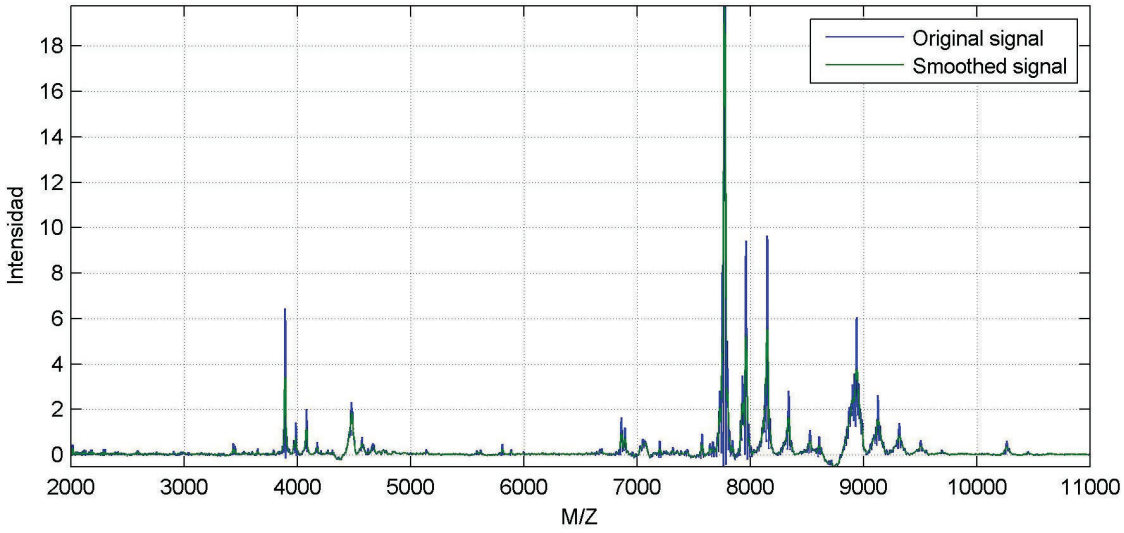


Figura 6.44.: Suavizamiento de Ruido en Mediciones - Conjunto OvarianDataset8-7-02

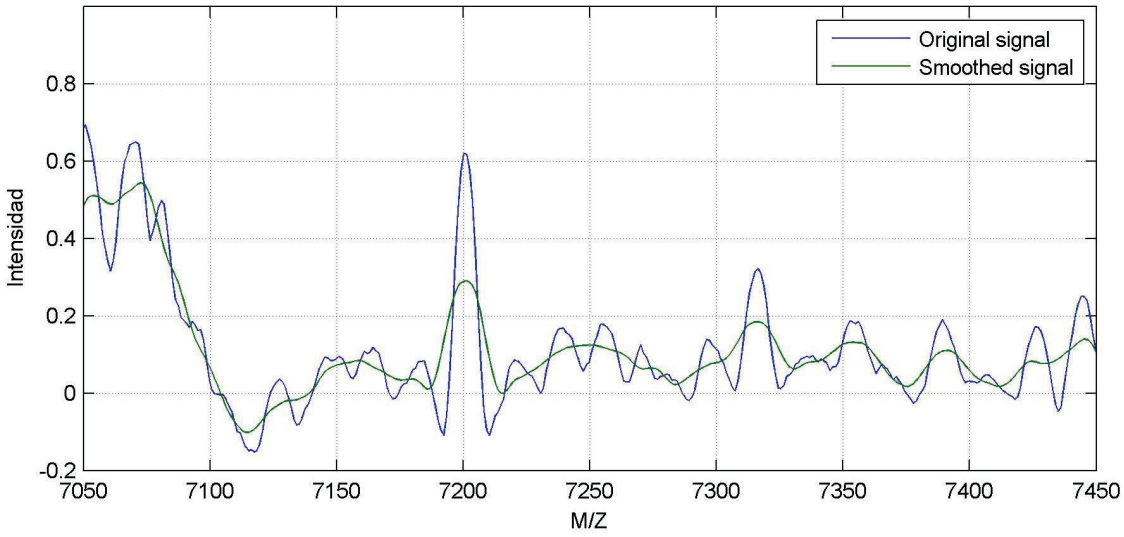


Figura 6.45.: Suavizamiento de Ruido en Mediciones - Zoom / Conjunto OvarianDataset8-7-02

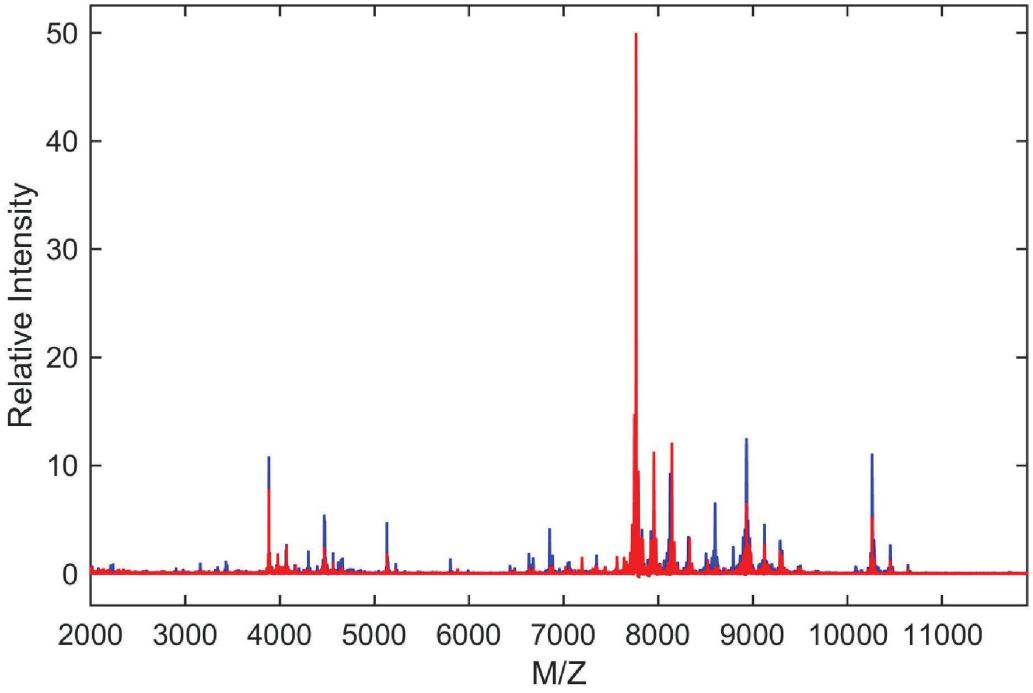


Figura 6.46.: Conjunto OvarianDataset8-7-02 - Visualización de Datos 2D

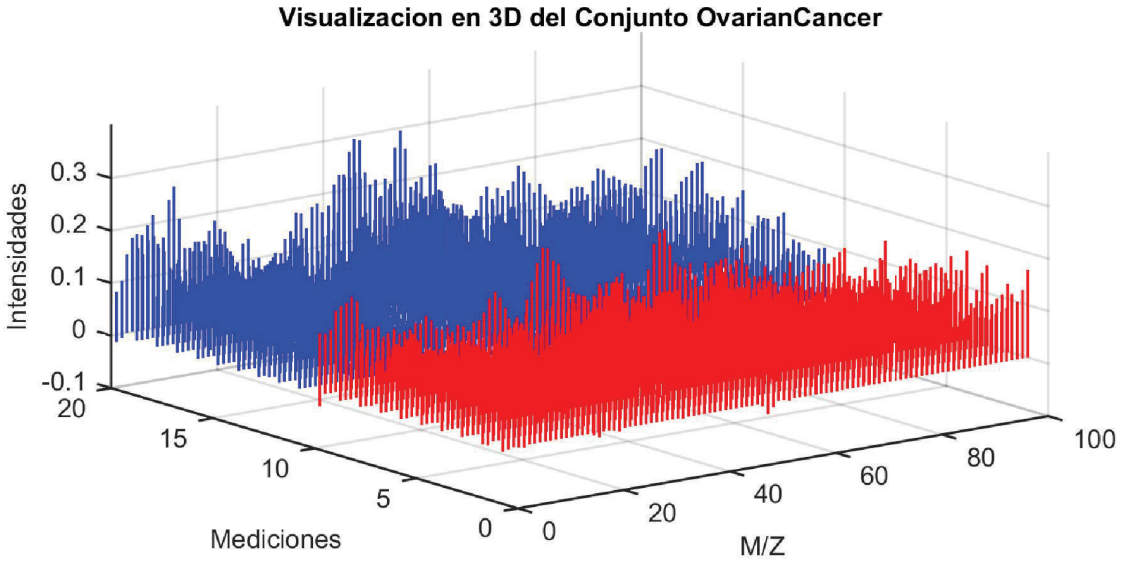


Figura 6.47.: Conjunto OvarianDataset8-7-02 - Visualización de Datos 3D

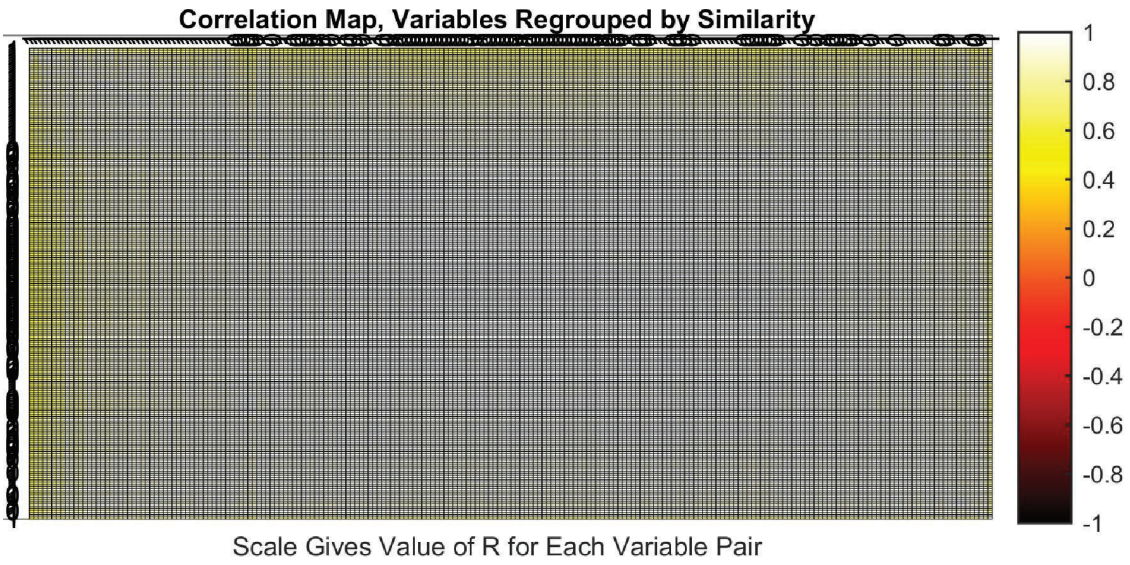


Figura 6.48.: Correlacion - Conjunto OvarianDataset8-7-02

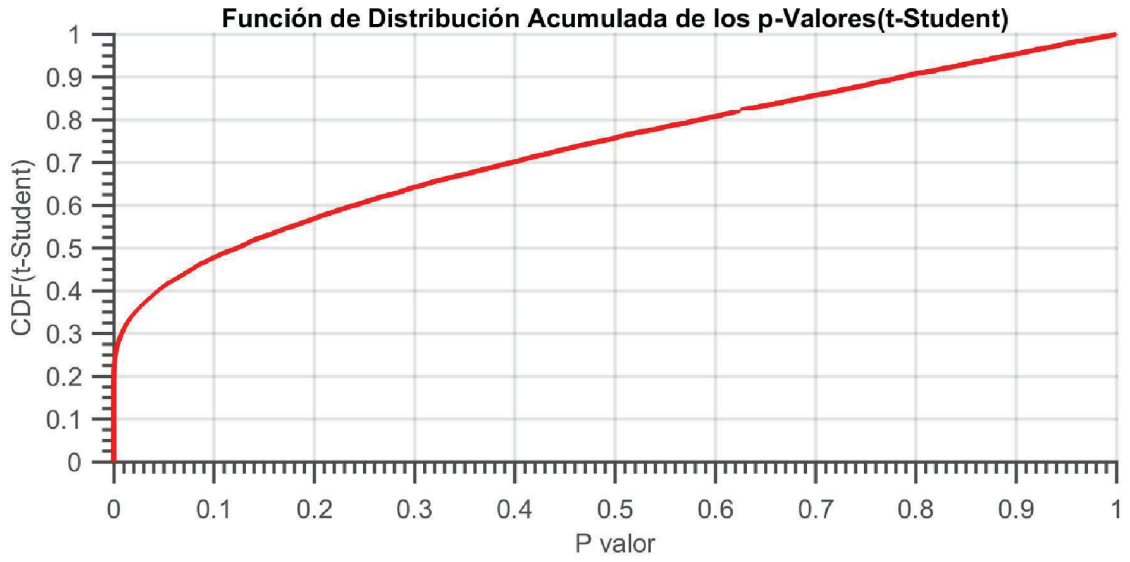


Figura 6.49.: Función Empírica de Probabilidad - Conjunto *Conjunto OvarianDataset8-7-02* (t-Student)

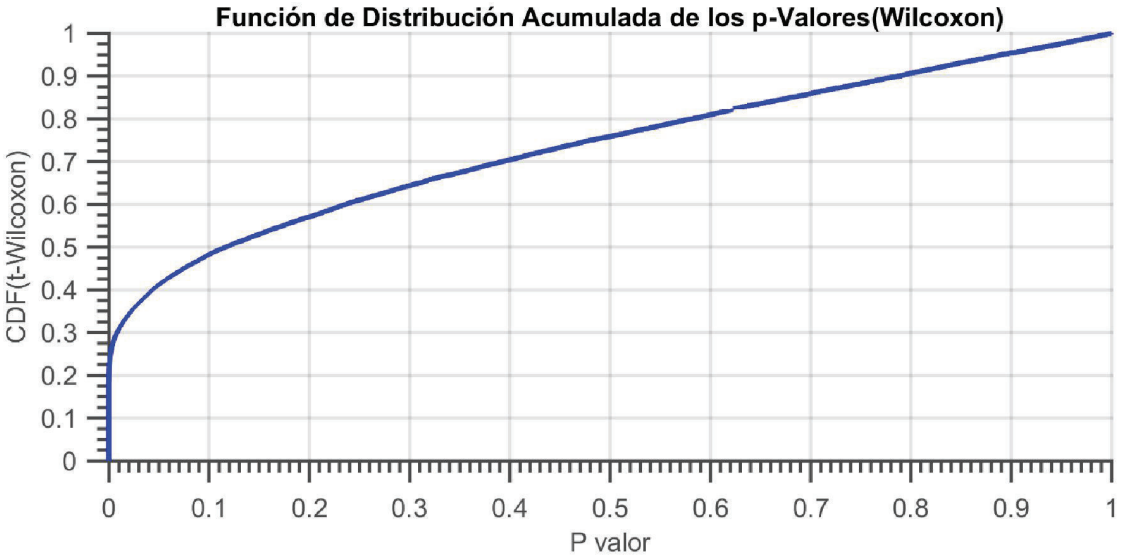


Figura 6.50.: Función Empírica de Probabilidad - Conjunto *OvarianDataset8-7-02* (Mann–Whitney U test)

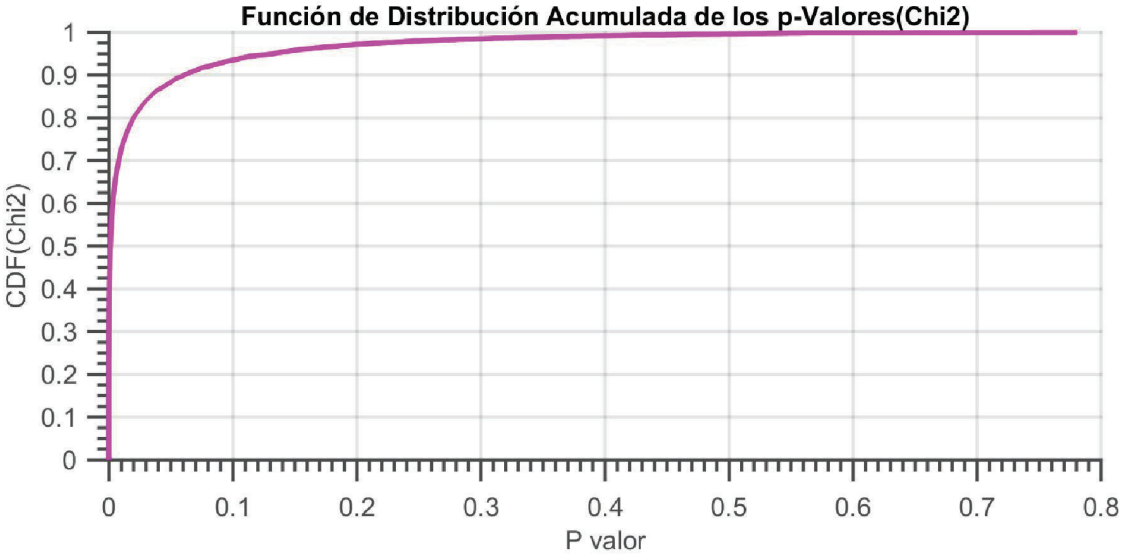


Figura 6.51.: Función Empírica de Probabilidad - Conjunto *OvarianDataset8-7-02* (χ^2)

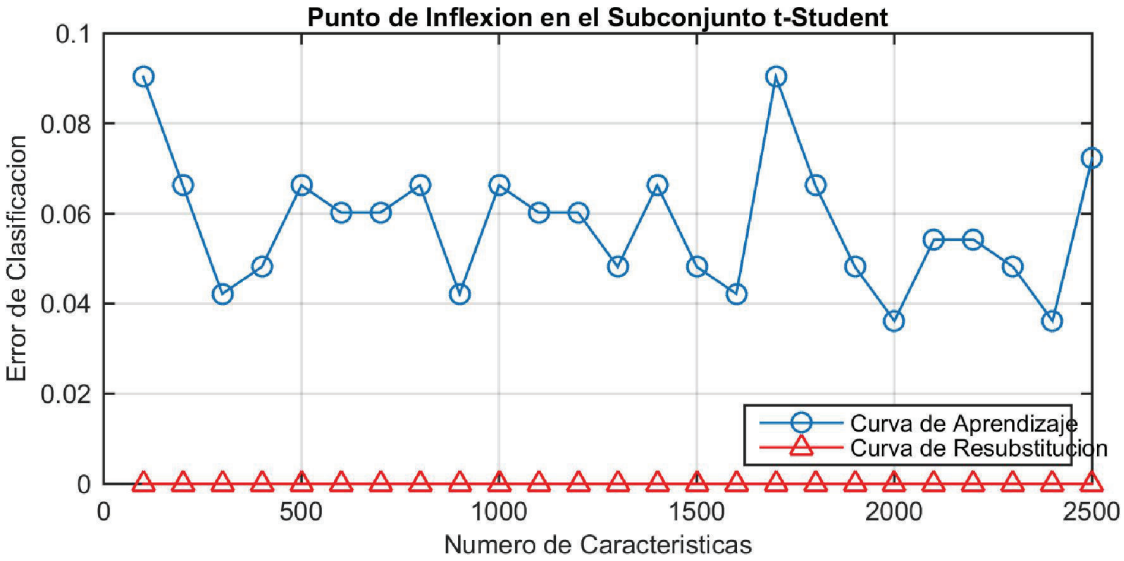


Figura 6.52.: Punto de Inflexión usando filtro t-Student

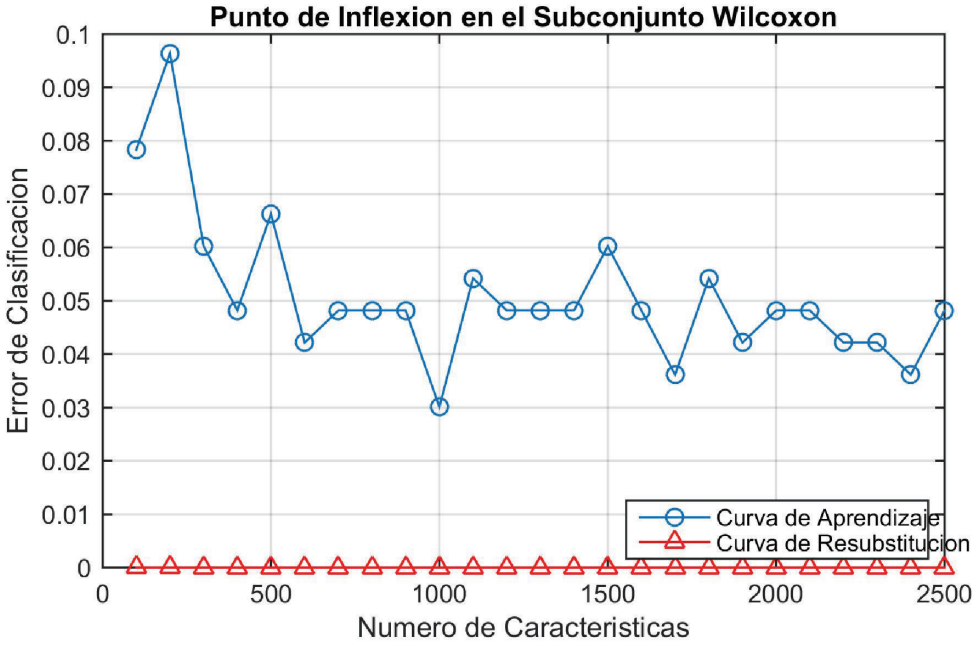


Figura 6.53.: Punto de Inflexión usando filtro Mann–Whitney U test

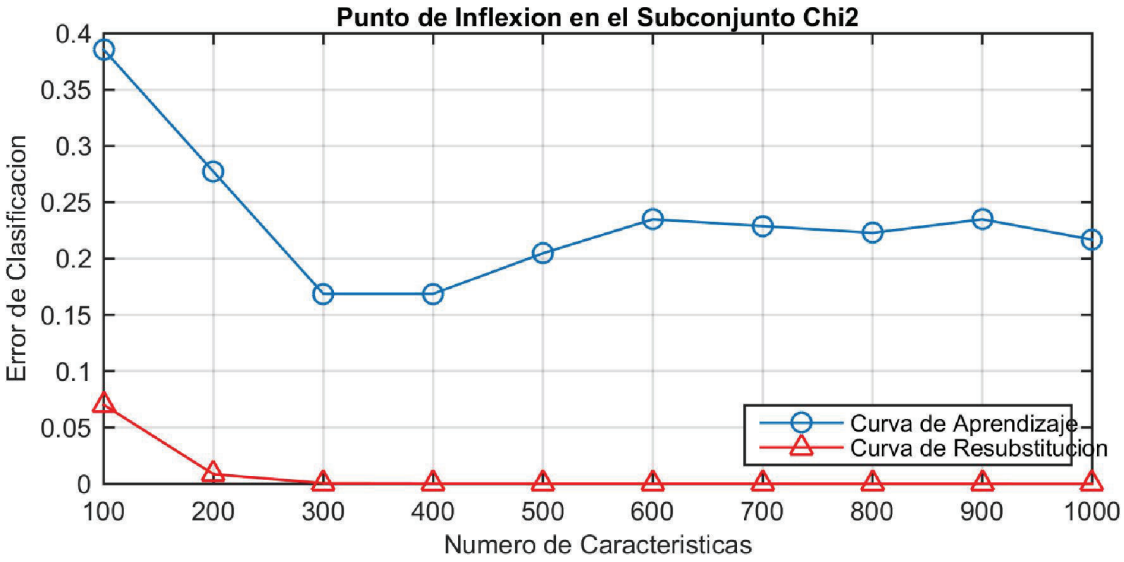


Figura 6.54.: Punto de Inflexión usando filtro χ^2

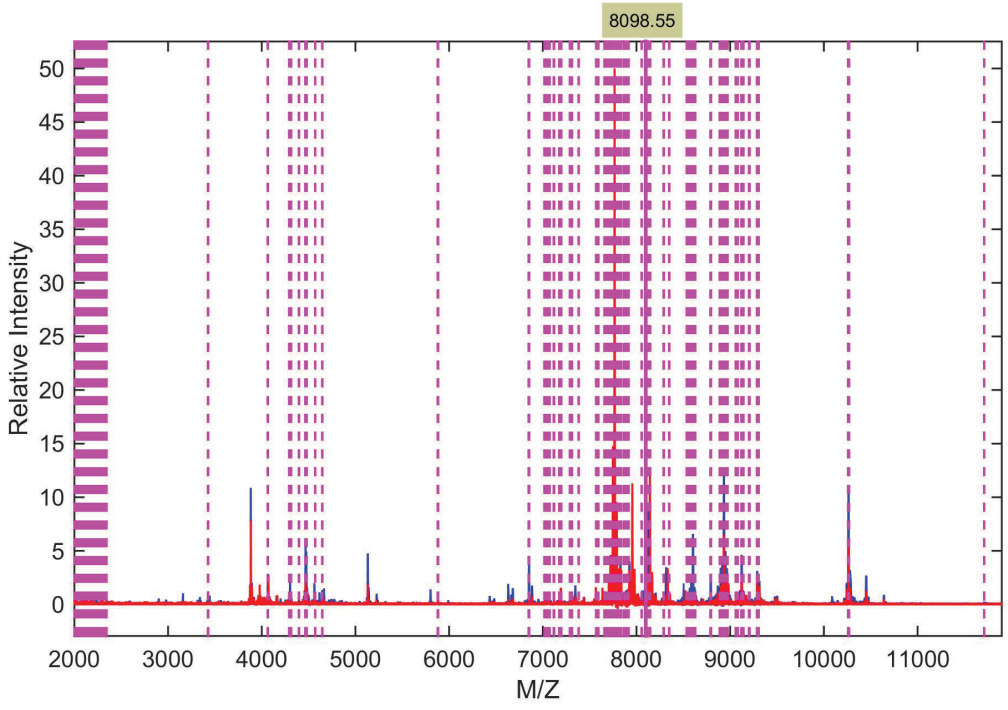


Figura 6.55.: Características detectadas con Marcadores Redundantes Sin Filtrar en Conjunto *OvarianDataset8-7-02*(mz vs I)

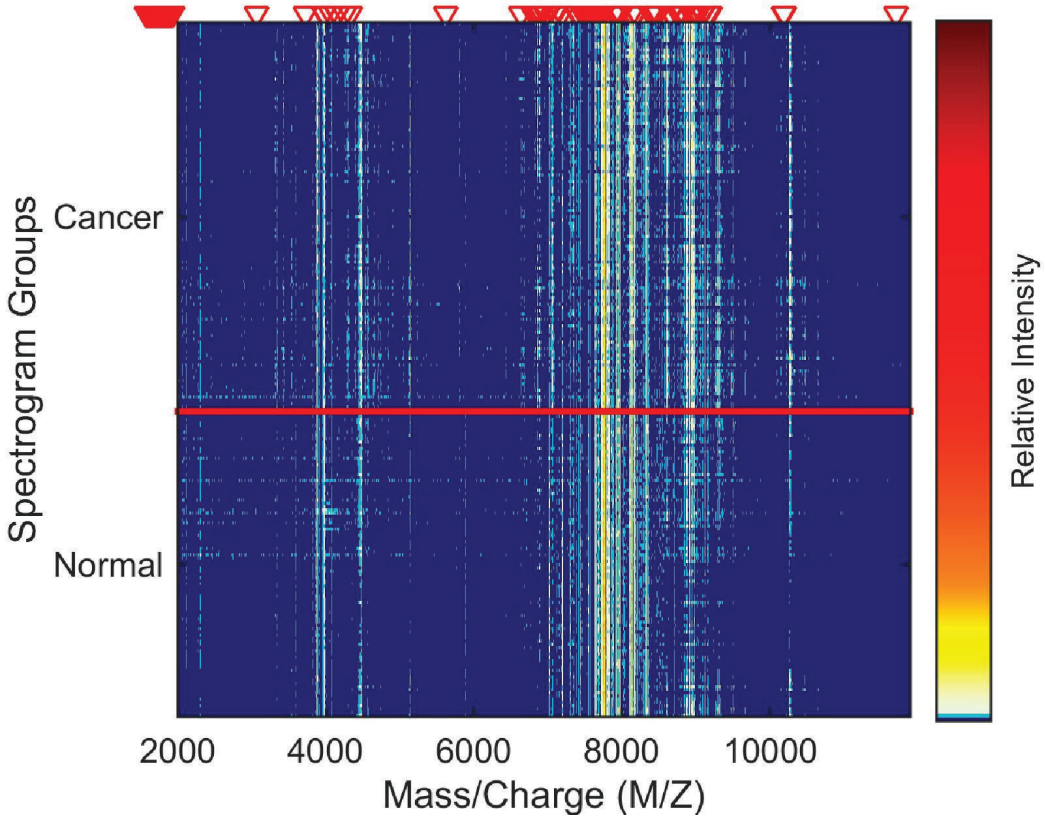


Figura 6.56.: Conjunto *OvarianDataset8-7-02* sin filtrar en mapa de calor

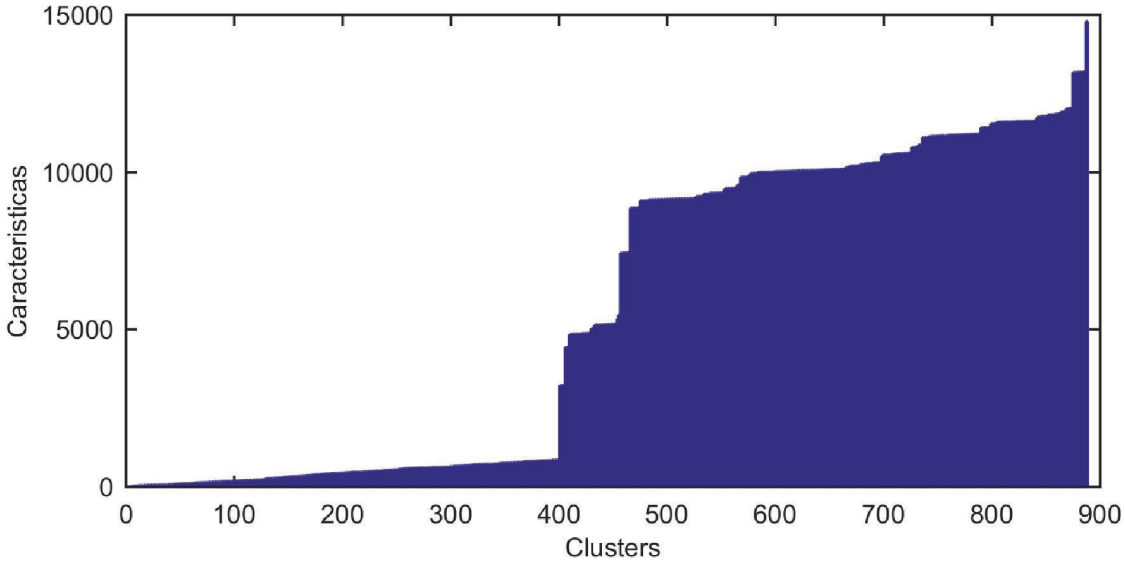


Figura 6.57.: Clusters - Redundancia de Marcadores en Conjunto *OvarianDataset8-7-02*(mz vs I)

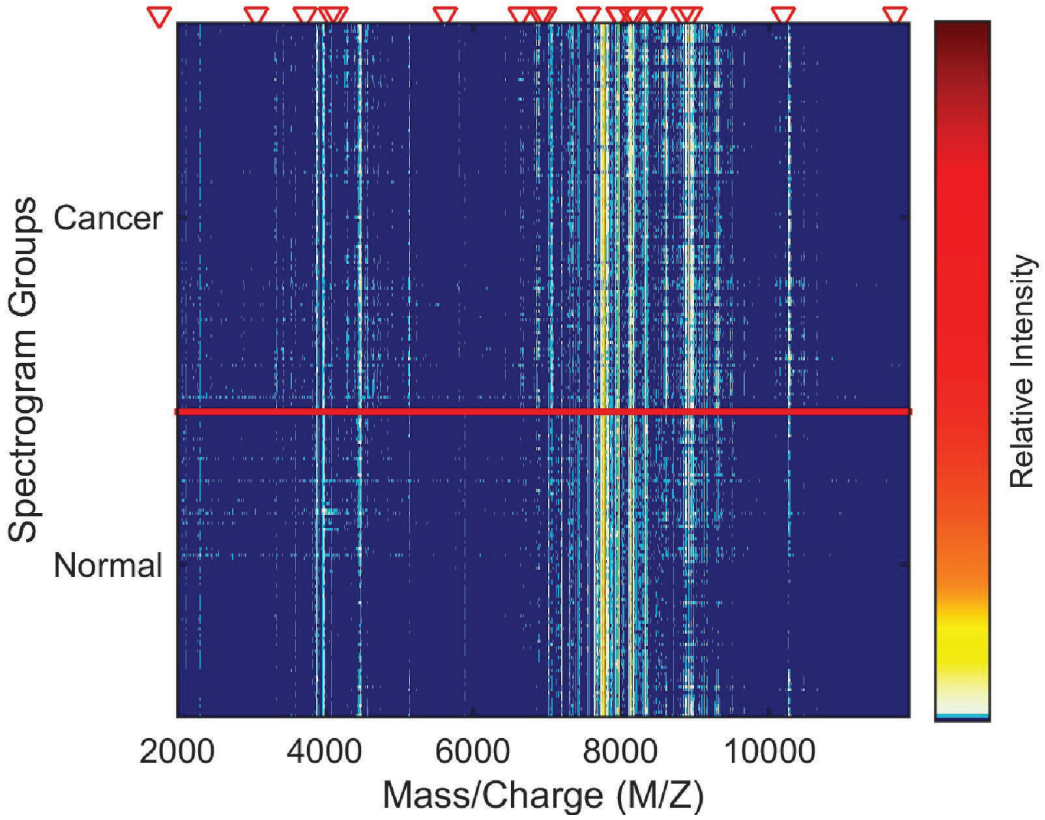


Figura 6.58.: Conjunto *OvarianDataset8-7-02* filtradas en mapa de calor

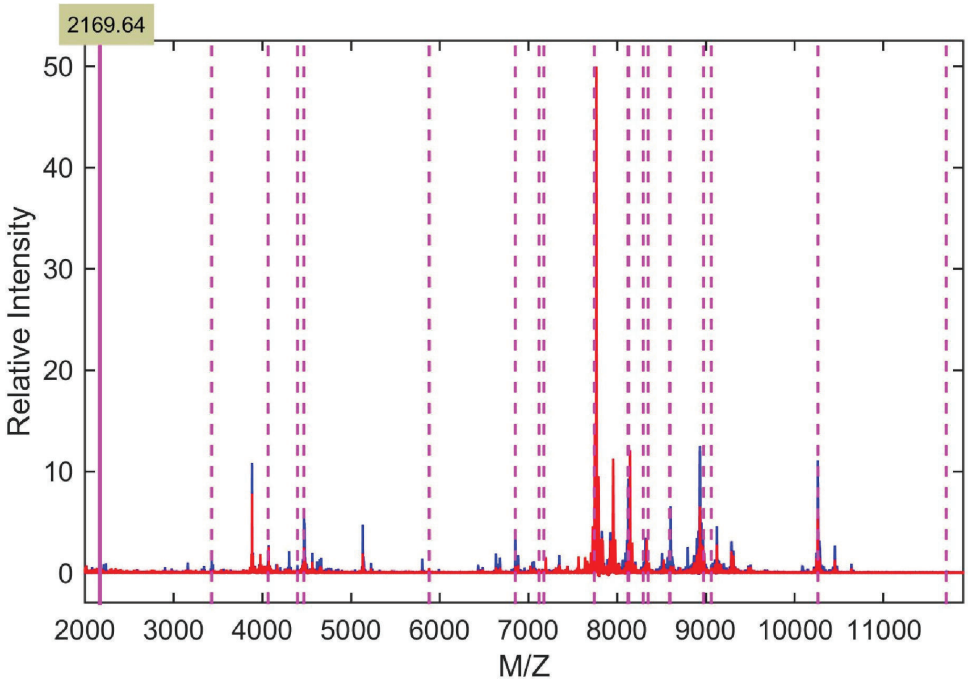


Figura 6.59.: Características Filtradas en Conjunto *OvarianDataset8-7-02*((mz vs I))

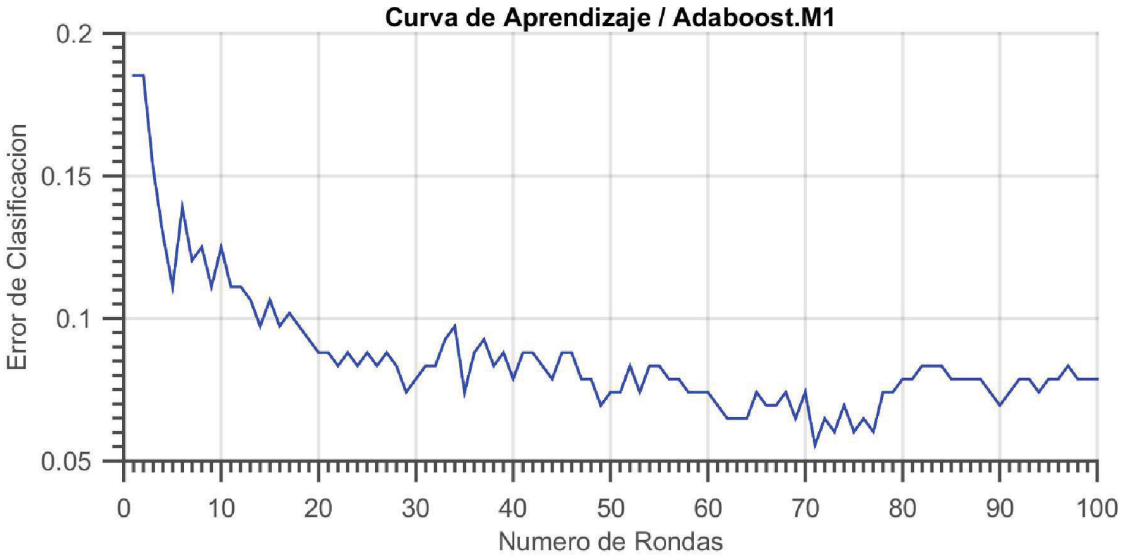


Figura 6.60.: Evaluación de las Características Seleccionadas usando *Crossvalidation* en Adaboost M1

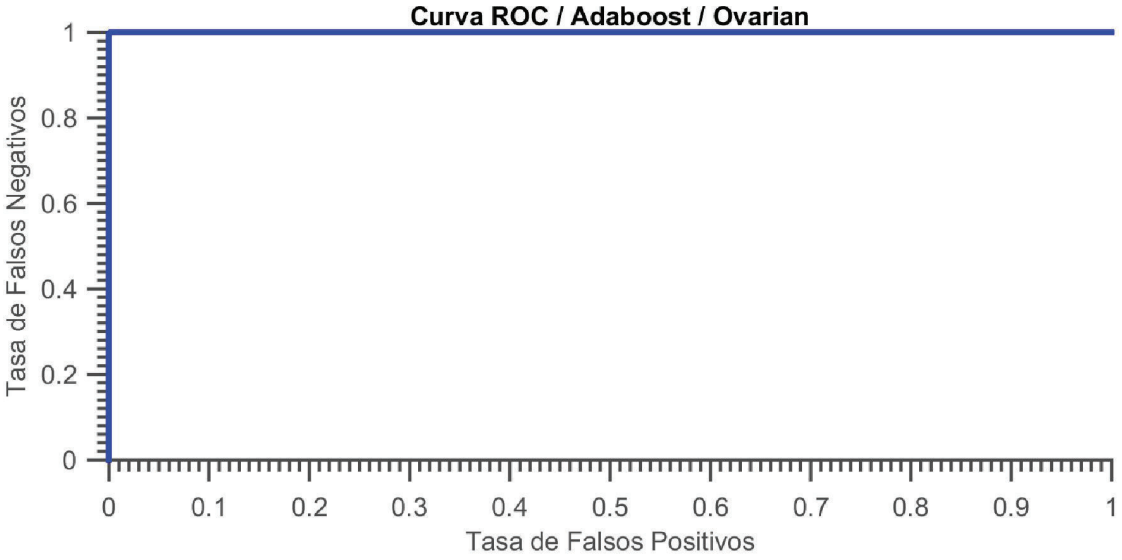


Figura 6.61.: Evaluación del Rendimiento de Clasificación usando las Características Seleccionadas - Curva Receiver Operating Characteristic(ROC)

CAPÍTULO 7

CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

SOBRE EL PROCESAMIENTO DE LAS MEDICIONES

- Los algoritmos de procesamiento de mediciones de espectrometría de masas tienen características enfocadas en solucionar los problemas característicos de la adquisición de datos a través de un espectrómetro de masas.
- Las mediciones de espectrometría de masas sufren modificaciones considerables en su estructura si no se ajustan correctamente los parámetros de entrada de los algoritmos de procesamiento.
- En las simulaciones realizadas, la etapa de remuestreo introduce discontinuidades en los valores de la medición, estos valores representan un problema en la etapa de validación de resultados ya que los algoritmos no clasifican correctamente la ausencia de valores en los puntos donde se produjeron estas discontinuidades.
- En la etapa de eliminación de la linea de base y suavizamiento de ruido, es muy importante calibrar correctamente los parámetros de entrada de estas etapas, ya que el exceso o valores muy valores muy bajos de las ventanas usadas para el análisis, los parámetros de regresión, o grados polinomiales usados en la construcción de curvas diferenciables, producen discontinuidades y alteran de manera radical la forma de los espectros, echando a perder todo el trabajo. Una buena aplicación de estos algoritmos no cambia la forma de los espectros y se concentra exclusivamente en la mitigación de los problemas propios de las mediciones de datos.

SOBRE LA SELECCIÓN DE CARACTERÍSTICAS

- La etapa de selección de características implementada en este proyecto de titulación hacer referencia al paradigma de analizar la muestra, para luego incluirla o descartarla, no se generan nuevas muestras ni valores.

- El no generar nuevos valores numéricos para la representación dimensional reducida de las mediciones de espectrometría de masas, representa una carga computacional alta sobre la plataforma de computacional de cálculo, debido al número de puntos a analizar y a los cálculos numéricos involucrados, en promedio decenas de miles.

- El filtro estadístico basado en la Prueba de *t-Student* muestra un mejor rendimiento en complemento con la *Mann–Whitney U test*, que en forma aislada. Al combinar las pruebas estadísticas de *t-Student, Mann–Whitney U test y* χ^2 en un filtro estadístico, el rendimiento no presenta mejoras notables en cuanto a habilidades de reconocer una muestra de otra muestra, sin embargo, al aplicar el filtro geométrico, la cantidad de muestras detectadas se ve reducida dramáticamente llegando a determinar como muestras discriminantes de grupos de 50 marcadores de entre 10000 marcadores originales, en el caso de arcene, y menores a 40 en los otros dos conjuntos de entre 15000 y 10000 muestras respectivamente.

- El filtro geométrico implementado toma como idea el eliminar características muy cercanas con el objetivo de brindar mayor información discriminante. Primeramente se miden todas las distancias entre las características detectadas en el filtro estadístico, luego se determina la menor distancia entre estas, para finalmente eliminar las características que tengas estas distancias de su característica antecesora y su predecesora, los resultados se muestras como un patrón limpio de información redundante sin perder la estructura básica de las zonas de masa a radio detectadas.

- En las simulaciones se utilizaron diferentes muestras de semilla en los generadores de números pseudoaleatorios para comprobar la certeza de las características seleccionadas, encontrándose en cada búsqueda heuristica las mismas posiciones en el eje de masa a cargas, lo que respalda los resultados expuestos en este trabajo y del algoritmo propuesto.

SOBRE LA VALIDACIÓN DE RESULTADOS

- La etapa de validación de características discriminantes tiene a ser una de las etapas mas trascendentales en estas aplicaciones, ya que si bien, en la selección de características, ya se obtuvieron resultados, en la validación es posible decidir si esos resultados fueron o no los mas adecuados.

- La aplicación del algoritmo de Adaboost.M1 a la etapa de modelación de un clasificador usando mediciones para el entrenamiento y pruebas reducidas dimensionalmente en función de las características seleccionadas, simplifica radicalmente la validación de resultados, debido a la practicidad en la implementación del algoritmo.

- Como prueba adicional a la medición del rendimiento del algoritmo a través de las curvas de aprendizaje, se usan simulaciones en validación cruzada y mediciones con pruebas externas. El rendimiento en todas las pruebas respalda el buen rendimiento del algoritmo.

- La aplicacion del Algoritmo Adaboost.M1 en arboles de decisión en este tipo de aplicaciones demuestra la capacidad y versatilidad del algoritmo, su fortaleza al *Overfitting, Underfitting* y al ruido estadístico, así como también su alta capacidad de generalización. De esta manera, el algoritmo propuesto puede ser escalado a problemas multiclase con el algoritmo Adaboost.M2, para diagnósticos y estudios mas dedicados a determinadas situaciones donde existan estados intermedios de entre cáncer y control(no patológico).

SOBRE LA SIMULACIÓN Y RESULTADOS

- Se cumplieron los objetivos planteados al inicio de este trabajo implementando un algoritmo completo para la minería de datos y determinación de biomarcadores en mediciones de espectrometria de masas. Los rendimientos obtenidos desde el punto de vista del aprendisaje no evidencian la presencia de *overfitting* ni *underfitting* y el rendimiento de clasificacion sobrepasa el $98\,\%$ de efectividad, lo cual muestra como prometedor el método propuesto.

- En la simulación se abordaron varias estrategias para optimizar los tiempos de computo de los valores numéricos, sin embargo, no se logro disminuir el tiempo total de procesamiento a valores menores de varios minutos. Desde este punto de vista, la implementación del algoritmo en etapas de procesamiento, selección

de características y validación, cada una de las etapas de manera independiente, resulta mas prometedora, ya que así, se pueden ir evaluando en el camino, el nivel de certeza de los resultados, recalibrando etapas intermedias y mejorando el rendimiento de la plataforma computacional sin saturar los bancos de memoria.

- La simulación abarca meticulosamente las etapas descritas en el alcance del plan de titulación aprobado, cada una de las etapas se muestran en las gráficas del capitulo de simulación y resultados. En este sentido hay que recalcar las facilidades que Matlab brinda para la modelación de algoritmos de procesamiento de datos. El modelar este tipo de procesos, en esta herramienta informática, no requiere de conocimientos de programación avanzados ni manejo de librerías especializadas.

- El uso de computadoras, minería de datos y aprendizaje de maquina para aplicaciones medicas, abre nuevas puertas en la búsqueda de curas para enfermedades crónicas. En este trabajo, los resultados obtenidos, resultan prometedores y motivan el trabajo en estas lineas de investigación. Las curvas de aprendizaje muestran un alto rendimiento y un buen nivel de rechazo al *underfitting* y *overfitting*.

RECOMENDACIONES

- La implementación de algoritmos en plataformas computacionales exige de mucha habilidad en la codificación de algoritmos. En este trabajo se dio un primer paso en estos trabajos haciendo uso de toolboxes especializados para la modelación del algoritmo. Debido a la complejidad del algoritmo, para implementaciones bajo lenguajes especializados en calculo númerico, se recomienda abordar cada subetapa del algoritmo de manera aislada y probarla con datos conocidos, que permitan conocer los resultados a esperar.

- Otro aspecto a tener en cuenta es los recursos computacionales. En el caso de las mediciones del conjunto de cáncer de ovario, la gran cantidad de puntos, cargarlos en memoria dependía de la memoria RAM disponible, si la memoria estaba por debajo de los $4[GB]$, Matlab reportaba saturación de memoria y no se podía cargar los datos. Para poder optimizar la modelación de algoritmos se recomienda contar con bancos de memoria superiores a los $8[GB]$.

- Finalmente se recomienda motivar la investigación multidisciplinar, para en futuros desarrollos, en colaboración con profesionales en medicina y biología, poder

obtener resultados más puntuales, traduciendo los patrones numéricos a sus equivalentes químicos en proteínas, antígenos, o a su vez, en nuevos biomarcadores que orienten a definir una cura para el cáncer.

TRABAJOS FUTUROS

- El presente trabajo de titulación presenta una primera etapa al desarrollo de algoritmos complejos en aplicaciones medicas basados en plataformas computacionales. Esta primera etapa comprende el procesamiento, búsqueda y validación de patrones de regiones de biomarcación en mediciones de espectrometría de masas. El alcance de este proyecto se limita a la identificación de los patrones y su validación, dejándose para trabajos futuros su traducción a su equivalente químico.
- La implementación del presente algoritmo se realizo usando puramente programación en lineas de código de Matlab y las funciones especializadas de sus toolboxes. Para trabajos futuros quedan abiertas las puertas a la implementación del algoritmo propuesto en este trabajo de titulación, en lenguajes más optimizados, como `C` o `C++`.
- Los tiempos de procesamiento exigieron recursos computacionales algo elevados para plataformas comerciales, por lo que la optimización del rendimiento en tiempos de cálculo deja aun detalles por pulir en la implementación del algoritmo para que este sea usado como una herramienta de diagnóstico.
- Este trabajo de titulación deja abiertas lineas de investigación para escalar el algoritmo propuesto a problemas multiclase donde se analicen múltiples grupos de estados patológicos, lo que permita mayor precisión a la hora de determinar biomarcadores.

Bibliografía

[1] Ingvar Eidhammer, Kristian Flikka, Lennart Martens, Svein Ole Mikalsen. *Computational Methods for Mass Spectrometry Proteomics*. Amdur M, Doull J, Klaassen C, 40 edition, 2007. ISBN 978-1-111-42737-5.

[2] M. Cannataro, P. H. Guzzi, T. Mazza, and P. Veltri. *Preprocessing, Management, and Analysis of Mass Spectrometry Proteomics Datam*. University Magna Grecia di Catanzaro, Italy., 2005. ISBN 9781605663746.

[3] Cesar Nombela y Concha Gil Aida Pitarch. *Enfermedades Infecciosas y Microbiología Clínica*. Enferm Infecc Microbiol Clin., 2010. ISBN 0213-005X.

[4] Alegre, E., Sánchez, L., Fernández, R. A., Mostaza, J. C. . *Procesamiento Digital de Imágenes. Fundamentos y Prácticas con Matlab*. Universidad de León, 2003. ISBN 84-9773-052-6.

[5] Gil Alterovitz and Marco F. Ramoni. *Systems Bioinformatics An Engineering Case-Based Approach*. ARTECH HOUSE, INC. Boston-London, 2007. ISBN 978-1-59693-124-4.

[6] Mohammad Hassan Moradi Amin Assareh and Vahid Esmaeili. *A Novel Ensemble Strategy for Classification of Prostate Cancer Protein Mass Spectra*. Conference of the IEEE EMBS, 2007. ISBN 1-4244-0788-5.

[7] Jeremie Bigot Anestis Antoniadis and Sophie Lambert-Lacroix. *Peaks Detection and Alignment for Mass Spectrometry Data*. Journal de la Société Francaise de Statistique, April 2010. ISBN 9781848211551.

[8] Mahadev Murthy Barker Peter E. *Biomarker Validation for Aging: Lessons from mt DNA Heteroplasmy Analyses in Early Cancer Detection*. Biomark Insights, 2003. ISBN 978-953-307-987-5.

[9] Roberts Berkow. *The Merck Manual: A Century of Medical Publishing and Practice*. Home Segunda Edition, 2008. ISBN 84-494-1184-X.

[10] Fernando Berzal. *Clasificación y Predicción*. Departamento de Ciencias de la Computación, 2006. ISBN 978-84-691-8159-1.

[11] Karin Sundfeldt Bjorg Kristijansdottir, Kristina Levan. *Potential Tumor Biomarkers Identified in Ovarian Cyst Fluid by Quantitative Proteomic Analysis, Itraq*. Clinical Proteomics, 2013. ISBN 978-91-628-8727-8.

[12] Scott E. Van Bramer. *An Introduction to Mass Spectrometry*. Departament of Chemistry One Universit y Place, 1997. ISBN 0708308104.

[13] Paul Brassard, Gilles; Bratley. *Fundamentos de Algoritmia*. PRENTICE HALL, 1997. ISBN 84-89660-00-X.

[14] C. Baumgartner, C. Boehm, D. Baumgartner, G. Marini, K. Weinberger, B. Olgemöller, B. Liebl and A. A. Roscher. *Supervised machine learning techniques for the classification of metabolic disorders in newborns*. BIOINFORMATICS, 2004. ISBN 978-1-59593-926-5.

[15] C. Martin Gomez y M. Ballesteros Gonzales. *Espectrometría de Masas y Análisis de Biomarcadores*. Real Academia Real de Farmacia, 2011. ISBN 798-84-937389-3-8.

[16] Robert A.W Christopher G. Herbert. *Mass Spectrometry Basics*. CRC Press, 2002. ISBN 0-8493-1354-6.

[17] Chung-Chih Lin,Yuh-Show Tsai,Yu-Shi Lin,Tai-Yu Chiu,Chia-Cheng Hsiung, May-I. Lee Jeremy C. Simpson and Chun-Nan Hsu. *Boosting multiclass learning with repeating codes and weak detectors for protein subcellular localization*. BIOINFORMATICS, 2007. ISBN 3-540-44038-0.

[18] Geoerey I. Webb Claude Sammut. *Encyclopedia of Machine Learning*. Library of Congress Control Number, 2011. ISBN 978-0-387-30768-8.

[19] Jason Corso. *Boosting and AdaBoost*. Suny at Buffalo, . ISBN 0-387-. 94559.

[20] Da Elene van der Merwe, K. Oikonomopoulou, J. Marshall and E. P. Diamandis. *Mass Spectrometry: Uncovering the Cancer Proteome for Diagnostics*. Elsevier Inc., 2007. ISBN 978-0-12-006696-4.

[21] MD David A. fishman. *National Ovarian Cancer Early Detection Program*. Mount Sinai School of Medicine, 2014. ISBN 9781475735895.

[22] Richard Maclin David Opitz. *Popular Ensemble Methods: An Empirical Study*. Journal of Artificial Intelligence Research 11, 1999. ISBN 978-1-439-83003-1.

[23] Luis de Alba. *Boosting and AdaBoost*. JCAI 99 Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence, 2008. ISBN 978-0-12-381479-1.

[24] Edmond de Hoffmann and Vincent Stroobant. *Mass Spectrometry Principles and Applications*. John Wiley and Sons Ltd, The Atrium, Southern Gate, England, 2007. ISBN 978-0-470-03310-4.

[25] Jose Angel Cocho de Juan. *Desarrollo de un método por espectrometría de masas en tándem para la de determinación de acilcarnitinas y la detección neonatal de alteraciones del metabolismo de ácidos orgánicos y ácidos grasos*. Universidad de Santiago de Compostela, 2007. ISBN 978-84-9750-974-9.

[26] María Durban. *Modelos Aditivos Generalizados con PSplines*. Hastie y Tibshirani, 1990. ISBN 978-607-28-0154-7. 49

[27] José Ángel García Pérez Ernesto Barrios Zamudio. *Formulario y Tablas de Probabilidad para el Curso de Estadística*. Instituto Tecnológico Autónomo de México, 2000. ISBN 9781107029873.

[28] Pedro J. Rodríguez Esquerdo. *Pruebas t*. Universidad de Puerto Rico, 2009. ISBN 0-7817-8258-9.

[29] Jose Maria Matias Fernandez. *Boosting con redes neuronales RBF. Análisis sesgo-varianza en un problema de clasificación*. VI Congreso Galego de Estadística e Investigación de Operaciones, 2003. ISBN 0-387-95457-0.

[30] Gabriel Hernández Sierra y José Calvo de Lara Flavio J. Reyes Diaz. *Métodos de clasificación de locutores utilizando clasificadores Boosting*. Cenatav, 2011. ISBN 2072-6287.

[31] Giorgio Sberveglieri Francesco Masulli, Matteo Pardo and Giorgio Valentini. *Boosting and Classification of Electronic Nose Data*. Kluwer Academic, 2002. ISBN 3-540-43818-1.

[32] John E. Freund and Gary A. Simon. *Estadística Elemental*. Prentice Hall, 1992. ISBN 968-880-433-9.

[33] Phillip I. Good. *Introduction to Statistics Through Resampling Methods and R/S-PLUS*. Springer, 2012. ISBN 9780471715757.

[34] Jurgen H. Gross. *Mass Spectrometry*. Springer, 2011. ISBN 978-3-642-10709-2.

[35] Roberto Behar Gutierrez. *55 Respuestas a dudas típicas de estadística*. Diaz de Santos, 2004. ISBN 84-7978-643-4.

[36] Isabelle Guyon and Andre Elisseeff. *An Introduction to Variable and Feature Selection*. Journal of Machine Learning Research, 2003. ISBN 94708-1501.

[37] Alex G. Harrison. *Chemical Ionization Mass Spectrometry*. CRC Press, 2000. ISBN 0-8493-4254-6.

[38] José M. Hernández. *ESPECTROMETRIA DE MASAS. APLICACIONES CLINICAS*. SEQC, 2007. ISBN 1887-6463.

[39] Herrera Luis Herrera Roberto. *Una Nueva Metodología para identificación de Patrones de Biomarcación aplicados al Estudio, Prevención y Tratamiento Temprano de Enfermedades Crónicas*. Revista Politécnica, Febrero 2015. ISBN 9780470041963.

[40] Yelena Mukomel Hong Tang and Eugene fink. *Diagnosis of Ovarian Cancer Based on Mass Spectra of Blood Samples*. IEEE, 2004. ISBN 0-7803-8566-7.

[41] Frank C. Porter Ilya Narsky. *Statistical Analysis Techniques in Particle Physics: fits, Density Estimation and Supervised Learning*. WILEY-VCH, 2013. ISBN 978-3-527-41086-6.

[42] Vinay K. Ingle and John G. Proakis. *Digital Signal Processing Using MATLAB*. CENGAGE Learning, 2012. ISBN 978-1-111-42737-5.

[43] Ingvar Eidhammer, Harald Barsnes,Geir Egil Eide,Lennart Martens. *Computational and Statistical Methods for Protein Quantification by Mass Spectrometry*. No Starch Press, 2013. ISBN 9781119964001.

[44] Isabelle Guyon, Jiwen Li, Theodor Mader, Patrick A. Pletscher, Georg Schneider, Markus Uhr. *Competitive baseline methods set new standards for the NIPS 2003 feature selection benchmark*. Pattern Recognition Letters, 2007. ISBN 1438–1444.

[45] Isabelle Guyon, Steve Gunn, Massoud Nikravesh, Lotfi A. Zadeh. *Feature Extraction-Foundations and Applications*. Springer, 2003. ISBN 1434-9922.

[46] Isis Bonet, Abdel Rodríguez, María M. García y Ricardo Grau. *Combinación de Clasificadores para Bioinformática*. TDGScholar, 2013. ISBN 9783642451102.

[47] Nasir Ahmad Jehad Ali, Rehanullah Khan and Imran Maqsood. *Random Forests and Decision Trees*. International Journal of Computer Science Issues, 2012. ISBN 1694-0814.

[48] Jan Sochman Jiri Matas. *AdaBoost*. Centre for Machine Perception Czech Technical University, Prague, 2007. ISBN 3-540-29620-4.

[49] Johan O. R. Gustafsson, Martin K. Oehler, Andrew Ruszkiewicz, Shaun R. McColl and Peter Hoffmann. *MALDI Imaging Mass Spectrometry (MALDI-IMS) Application of Spatial Proteomics for Ovarian Cancer Classification and Diagnosis*. International Journal of Molecular Sciences, 2011. ISBN 1422-0067.

[50] Kees Jong. *Machine Learning for Human Cancer Research*. Printed by Universal Press, The Netherlands, 2006. ISBN 2320- 9798.

[51] Juan Pérez. Juan Francisco Monje. *Estadística no Paramétrica Prueba Chi Cuadrado*. Alianza editorial, 1996. ISBN 84-206-8110-5. 51

[52] Adem Karahoca. *Data Mining applications in engineering and medicine*. Janeza Trdine 9, 51000 Rijeka, Croatia, 2012. ISBN 978-953-51-0720-0.

[53] Balazs Kegl. *Introduction to AdaBoost*. Palikir, 2009. ISBN 0-7695-2122-3.

[54] Kermit K. Murray, Robert K. Boyd, Marcos N. Eberlin, G. John Langley, Liang Li and Yasuhide Naito. *Definitions of terms relating to mass spectrometry (IUPAC Recommendations 2013)*. Pure Appl. Chem, 2013. ISBN 0195059298.

[55] S. B. Kotsiantis. *Supervised Machine Learning: A Review of Classification Techniques*. University of Peloponnese, Greece, 2007. ISBN 0-13273350-1.

[56] Bjorg Kristjansdottir. *Early Diagnosis of Epithelial Ovarian Cancer Analysis of Novel Biomarkers*. Gothenburg 2013, 2013. ISBN 978-91-628-8727-8.

[57] Eva Lange. *Analysis of mass spectrometric data: peak picking and map alignment*. Freie Universit at Berlin, 2008. ISBN 978-3-659-33320-0.

[58] Niklas Lavesson. *Evaluation and Analysis of Supervised Learning Algorithms and Classifiers*. Blekinge Institute of Technology, 2006. ISBN 91-7295-083-8.

[59] Leong, PP; Rezai, B; Koch, WM; Reed, A; Eisele, D; Lee, DJ; Sidransky, D; Jen, J; Westra, WH . *HPV Analysis Distinguishing Second Primary Tumors from Lung Metastases in Patients with Head and Neck Squamous Cell Carcinoma*. Am J Surg Pathol, 1998. ISBN 978-953-51-0228-1.

[60] Angela Nebot Luis Carlos Molina, Lluís Belanche. *Feature Selection Algorithms: A Survey and Experimental Evaluation*. Departament de Languages and Systems Informatics, 2002. ISBN 0-7695-1754-4.

[61] Víctor Robles Forcada Luis Pelayo Guerra Velasco, José María Peña Sánchez. *Curso de Minería de datos*. Facultad de Informática, 2008. ISBN 0-471-66657-2.

[62] Michal Lysek and Tobias Persson. *Diagnostics using SELDI-TOF Mass Spectrometry*. Technical report, IDE0530, 2005. ISBN 81-7764-707-5.

[63] M. Ceccarelli, A. Dacierno y A. Facchiano. *A Machine Learning Approach To Mass Spectra Classification With Unsupervised Feature Selection*. BibSonomy, 2011. ISBN 978-3-642-02503-7.

[64] Rune Matthiesen. *Mass Spectrometry Data Analysis in Proteomics*. Humana Press, 2007. ISBN 978-1-58829-563-7.

[65] Matthiesen Rune. *Bioinformatics Methods In Clinical Research : Methods, Applications, and Tools*. Humana Press Inc, U.S., 2009. ISBN 1603271937/9781603271936.

[66] L. Joyanes. McGraw-Hill. *Fundamentos de Programación*. McGraw-Hill Interamericana de España S.L.; Edición: 4 (24 de marzo de 2008), 2008. ISBN 8448161114.

[67] Ron Meir and Gunnar Ratsch. *An introduction to boosting and leveraging*. Springer Verlag New York, Inc. New York, 2003. ISBN 3-540-00529-3.

[68] Melanie Hilaro, Alexandros Kalousis, Christian Pellegrini and Markus Muller. *Processing and Classification of Protein Mass Spectra*. Wiley InterScience DOI 10.1002/mas.20072, 2005. ISBN 0470090138.

[69] James N. Miller and Jane C. Miller. *Estadística y Quimiometría para Química Analítica*. Prentice Hall, 2000. ISBN 84-205-3514-1.

[70] Cedazo Minguez. *Biomarkers for Alzheimer's Disease and Other Forms of Dementia*. Experimental Gerontology, 2010. ISBN 0-13-010492-2.

[71] A. D. Mc Naught and A. Wilkinson. *IUPAC. Compendium of Chemical Terminology*. 2 ed. the Gold Book, 1998. ISBN 0-9678550-9-8.

[72] Oleg Okun. *Feature Selection and Ensemble Methods for Bioinformatics*. SMARTTECCO, Sweden, 2011. ISBN 978-1-60960-558-2.

[73] Matzinger P. *The Danger Model: a Renewed Sense of Self*. National Institutes of Health Bethesda, 2002. ISBN 9783527320844.

[74] Bing B. Zhou Pengyi Yang, Yee Hwa Yang and Albert Y. Zomaya. *A review of ensemble methods in bioinformatics*. Current Bioinformatics, 2006. ISBN 978-0-9893193-1-7.

[75] Pita Fernández S. Pertega Diaz S. *Métodos Paramétricos para la Comparación de Dos Medias T de Student*. Unidad de Epidemiología Clínica y Bioestadística, 2001. ISBN 978-84-695-1074-2.

[76] Li Yu Phuong Pham and Minh Nguyen. *A Novel Algorithm for Multi-Class Cancer Diagnosis on MALDI-TOF Mass Spectra*. 2011 IEEE International Conference on Bioinformatics and Biomedicine, 2011. ISBN 978-1-4577-1799-4.

[77] HE Ping. *Classification Methods and Applications to Mass Spectral Data*. Hong Kong Baptist University, 2007. ISBN 0542428431.

[78] ROBI POLIKAR. *PATTERN RECOGNITION*. Rowan University Glassboro, New Jersey, 2006. ISBN 0387310738.

[79] Remi Longuespee, Charlotte Boyon, Olivier Kerdraon, Denis Vinatier, Isabelle Fournier, Robert Day and Michel Salzet . *MALDI-MSI and Ovarian Cancer Biomarkers*. Advances in Cancer Management, 2012. ISBN 978-953-307-870-0.

[80] David Stork Richard Duda, Peter Hart. *Pattern Classification*. Wiley Second Edition, 2001. ISBN 978-0-471-05669-0.

[81] John Walker Rune Matthiesen. *METHODS IN MOLECULAR BIOLOGY*. Institute of Molecular Pathology and Immunology Universidad do Porto, Portugal, 2013. ISBN 978-1-62703-391-6.

[82] Saketh. *Foundations of Machine Learning*. The MIT Press, 2001. ISBN 978-0262018258.

[83] Rob Schapire. *The Boosting Approach to Machine Learning*. Princeton University, 2001. ISBN 978-0-387-21579-2.

[84] Robert J. Schilling and Sandra L. Harris. *Fundamentals of digital signal processing*. Clarkson University Potsdam, NY, 2012. ISBN 0-8400-6909-X.

[85] Shan He, Huanhuan Chen, Xiaoli Li, and Xin Yaol. *Profiling of mass spectrometry data for ovarian cancer detection using negative correlation learning*. School of Computer Science University of Birmingham, 2009. ISBN 978-3-642-04277-5.

[86] Rosie Shier. *The Wilcoxon signed rank sum test*. Mathematics Learning Support Centre, 2004. ISBN 1111746583.

[87] Holler F. James SKOOG, D.A. James. *PRINCIPIOS DE ANALISIS INSTRUMENTAL*. Editorial McGraw-Hill, 1998. ISBN 0-03-002078-6.

[88] Christopher C. Atkeson Stefan Schaal. *From Isolation to Cooperation: An Alternative View of a System of Experts*. College of Computing, Georgia Tech, 2011. ISBN 0-262-20107-0.

[89] S. Theodoridis and K. Koutroumbas. *An Introduction to Pattern Recognition A MATLAB Approach*. Library of Congress Cataloging-in-Publication Data, 2009. ISBN 978-0-12-374486-9.

[90] Jerome Friedman Trevor Hastie, Robert Tibshirani. *The Elements of Statistical Learning*. Springer Series in Statistics, 2008. ISBN 978-0-387-84858-7.

[91] Noelia Sanchez Veronica Bolon Canedo and Amparo Alonso Betanzos. *Feature Selection for High-Dimentional Data*. Springer, 2007. ISBN 978-3-319-21857-1.

[92] Angel R. Martinez Wendy L. Martinez. *Computational Statistics Handbook with MATLAB*. CHAPMAN and HALL CRC, 2002. ISBN 9781584885665.

[93] Werner Dubitzky, Martin Granzow, and Daniel Berrar. *Introduction to Genomic and Proteomic Data Analysis*. Hardcover, Springe, 2007. ISBN 0-387-47508-7.

[94] markus Kostrzewa Wolfgang Pusch, Sau Mei Leung. *Mass Spectrometry Based Clinical Proteomics*. PubMed, 2006. ISBN 18732364.

[95] Robert E. Schapire Yoav Freund. *Experiments with a New Boosting Algorithm*. Proceedings of the Thirteenth International Conference, 1996. ISBN 1-55860-419-7.

[96] Yvan Saeys, Inaki Inza and Pedro Larranaga. *A Review of Feature Selection Techniques in Bioinformatics*. US National Library of Medicine National Institutes of Health, 2007. ISBN 978-1-4503-0616-4.

[97] Mohammed J. Zaki. *Advances in Knowledge Discovery and Data Mining*. June 2010, 2010. ISBN 3-642-13656-7.

[98] Ingrid Russell Zdravko Markov. *An Introduction to the WEKA Data Mining System*. University of Hartford., . ISBN 1-59593-055-8.

[99] Zhenyu Zhang. *Research on AdaBoost.M1 with Random Forest*. Department of Development Strategy China Mobile Group Guizhou Co., Ltd Guiyang, China, 2008. ISBN 978-1-4244-6 350-3.

[100] Zhi Hua Zhou and Yang Yu. *The Top Ten Algorithms in Data Mining*. CRC Press, 2009. ISBN 9781420089646.

APÉNDICE A

Reportes de Simulación de Matlab

A.1 CÓDIGOS DE MATLAB PARA LA ETAPA DE DE PROCESAMIENTO DE MEDICIONES

A.1.1 CONJUNTO ARCENE

```
%%=====================================================================%%
%% BUSQUEDA DE ZONAS DE BIOMARCACION EN EL CONJUNTO DE DATOS ARCENE
% El presente codigo de la busqueda de zonas de biomarcacion en el
% conjunto de datos  ARCENE. Este script descarga los datos
% directamente del sitio web de Web Center for Machine Learning and
% Intelligent Systems, los procesa, analiza, extrae resultados, los
% valida y reporta las curvas de analisis.
%%=====================================================================%%
%
%%=====================================================================%%
%% Descarga de Archivos
%%=====================================================================%%
%
% Descargando Archivos del Sitio Web Center for Machine Learning and
% Intelligent Systems - Machine Learning Repository
% https://archive.ics.uci.edu/ml/datasets/Arcene
clc; clear;  % Limpiar el espacio de memoria Workspace
% Subphases -> Etapas de descarga en el Menu grafico
% Stage -> Etapa alcanzada
subphases = 14; stage = 0;
h = waitbar(0,'Inicializando descarga...','Name', ...
```

```
22      'Descargando Conjunto de Datos');
23  stage = stage + 1;
24
25  % Crear menu grafico de barra de avance, waitbar
26  waitbar(stage/subphases,h,'Descargando arcene.param');
27  direccion = ['https://archive.ics.uci.edu/ml/machine-learning' ...
28      '-databases/arcene/ARCENE/arcene.param'];
29  urlwrite(direccion,'arcene.param');
30  stage = stage + 1;
31  waitbar(stage/subphases,h,'arcene.param descargado...');
32  stage = stage + 1;
33  waitbar(stage/subphases,h,'Descargando arcene train.data');
34  direccion = ['https://archive.ics.uci.edu/ml/machine-learning' ...
35      '-databases/arcene/ARCENE/arcene_train.data'];
36  urlwrite(direccion,'arcene_train.data');
37  stage = stage + 1;
38  waitbar(stage/subphases,h,'arcene train.data descargado...');
39  stage = stage + 1;
40  waitbar(stage/subphases,h,'Descargando arcene test.data ');
41  direccion = ['https://archive.ics.uci.edu/ml/machine-learning' ...
42      '-databases/arcene/ARCENE/arcene_test.data'];
43  urlwrite(direccion,'arcene_test.data');
44  stage = stage + 1;
45  waitbar(stage/subphases,h,'arcene test.data descargado...');
46  stage = stage + 1;
47  waitbar(stage/subphases,h,'Descargando arcene train.labels');
48  direccion = ['https://archive.ics.uci.edu/ml/machine-learning' ...
49      '-databases/arcene/ARCENE/arcene_train.labels'];
50  urlwrite(direccion,'arcene_train.labels');
51  stage = stage + 1;
52  waitbar(stage/subphases,h,'arcene train.labels descargado...');
53  stage = stage + 1;
54  waitbar(stage/subphases,h,'Descargando arcene valid.data');
55  direccion = ['https://archive.ics.uci.edu/ml/machine-learning-' ...
56      'databases/arcene/ARCENE/arcene_valid.data'];
57  urlwrite(direccion,'arcene_valid.data');
58  stage = stage + 1;
```

```
59 waitbar(stage/subphases,h,'arcene valid.data descargado...');
60 stage = stage + 1;
61 waitbar(stage/subphases,h,'Descargando arcene arcene valid.labels');
62 direccion = ['https://archive.ics.uci.edu/ml/machine-learning' ...
63     '-databases/arcene/arcene_valid.labels'];
64 urlwrite(direccion,'arcene_valid.labels');
65 urlwrite('http://www.eigenvector.com/MATLAB/PC_M_files/corrmap.m',...
66     'corrmap.m');
67 stage = stage + 1;
68 waitbar(stage/subphases,h,'arcene valid.labels descargado...');
69 stage = stage + 1;
70 waitbar(stage /subphases,h,'Sin errores en la Descarga 100 % [OK] !!');
71 pause(1);
72 stage = stage + 1;
73 waitbar(stage/subphases,h,'Proceso de Descarga de Archivos Terminado');
74 pause(2);
75 delete(h);  % Cerrar Menu grafico
76 %
77 %%=====================================================================%%
78 %% Procesamiento de Mediciones
79 %%=====================================================================%%
80 %
81 % En esta etapa se empaquetaran las mediciones en forma matricial.
82 % No se realizan operaciones adicionales de procesamiento ya
83 % que las medicionesdel conjunto de datos descargado estan
84 %ya sometidas a un procesamiento previo.
85 %
86 subphases = 4; stage = 0;
87 h = waitbar(0,'Cargando datos en Memoria ...','Name',...
88     'Procesamiento de Mediciones');
89 stage = stage + 1;
90 waitbar(stage/subphases,h, ...
91     'Eliminando extensiones de los archivos descargados');
92 %
93 % Elimina las extensiones de los archivos descargados
94 % codificandolos en texto plano a archivos sin extensiones
95 % en estandar UTF-8
```

```
96 %
97 movefile('arcene_train.data','ArceneTrainDatos');
98 movefile('arcene_train.labels','ArceneTrainEtiquetas');
99 movefile('arcene_valid.data','ArceneTestDatos');
100 movefile('arcene_valid.labels','ArceneTestEtiquetas');
101 movefile('arcene_test.data','ArceneTestsData');
102 stage = stage + 1;
103 waitbar(stage/subphases,h,'Cargando datos de mediciones en Memoria');
104 %
105 % Carga los datos en memoria Workspace
106 TrainDatos = load('ArceneTrainDatos');
107 TrainLabels = load('ArceneTrainEtiquetas');
108 TestDatos = load('ArceneTestDatos');
109 TestLabels = load('ArceneTestEtiquetas');
110 stage = stage + 1;
111 waitbar(stage/subphases,h, ...
112     'Particion de Conjuntos en Prueba y Entrenamiento');
113 %
114 % Separar los datos en grupos de analisis
115 %
116 ConjuntoDatos = vertcat(TrainDatos,TestDatos);
117 EtiquetasDatos = vertcat(TrainLabels,TestLabels);
118 IndiceGrupo1 = find(EtiquetasDatos==1);
119 IndiceGrupo2 = find(EtiquetasDatos==-1);
120 DatosGrupo1 = ConjuntoDatos(IndiceGrupo1,:);
121 DatosGrupo2 = ConjuntoDatos(IndiceGrupo2,:);
122 EtiquetasGrupo2 = EtiquetasDatos(IndiceGrupo2,:);
123 EtiquetasGrupo1 = EtiquetasDatos(IndiceGrupo1,:);
124 DatosReagrupados = vertcat(DatosGrupo1,DatosGrupo2);
125 EtiquetasGrupo1Grupo2 = vertcat(EtiquetasGrupo1,EtiquetasGrupo2);
126 MZ = 1:1:length(DatosReagrupados(1,:)); MZ = MZ';
127 stage = stage + 1;
128 waitbar(stage/subphases,h, ...
129     'Visualizacion de los Datos del Conjunto Arcene');
130 delete(h);   % Borrar menu de Waitbar
131 %
132 %%=====================================================================%%
```

```
%% Visualizacion en 2D  de las mediciones de los conjuntos de Datos
%%=========================================================================%%
msviewer(MZ,DatosReagrupados','Group',EtiquetasGrupo1Grupo2)

%%=========================================================================%%
%% Visualizacion en 3D  de las mediciones de los conjuntos de Datos
%%=========================================================================%%
EjeZ=repmat([1:100],10); j=0;
for i=1:1:10
    mattz= i*ones(1,1000); GrupoP(i,:) = mattz;
end
for i=11:1:20
    j=j+1;
    mattz= i*ones(1,1000); GrupoN(j,:) = mattz;
end

EspectrosDemoP=DatosGrupo1([1:10],[1:1000]); % 10 muestras por grupo
EspectrosDemoN=DatosGrupo2([1:10],[1:1000]); % 10 muestras por grupo
h = stem3(EjeZ,GrupoP,EspectrosDemoP,'r');
set(h, 'Marker', 'none');
hold on
h = stem3(EjeZ,GrupoN,EspectrosDemoN,'b');
hr = rotate3d;
hr.RotateStyle = 'box';
hr.Enable = 'on';
set(h, 'Marker', 'none');
ylabel('Mediciones'); xlabel('M/Z'); zlabel('Intensidades');
hold off
print('-dtiff','-r500','3DArcene.jpg')
%
%%=========================================================================%%
%% Visualizacion en Mapa de Calor de las mediciones de los conjuntos ...
    de Datos
%%=========================================================================%%
msheatmap(MZ,DatosReagrupados','Group',EtiquetasGrupo1Grupo2)
```

A.1.2 CONJUNTO OVARIAN CANCER QA-QC

```
%% Descarga de Archivos
%
% Descargando Archivos del Sitio Web Center for Machine Learning and
% Intelligent Systems - Machine Learning Repository
% https://archive.ics.uci.edu/ml/datasets/Arcene

clc;
clear;

% subphases = 14; stage = 0;
% h = waitbar(0,'Inicializando descarga...','Name', ...
%     'Descargando Conjunto de Datos');
% stage = stage + 1;
% waitbar(stage/subphases,h,'Descargando OvarianCD-PostQAQC.zip');
% direccion = ['http://home.ccr.cancer.gov/' ...
%     'ncifdaproteomics/OvarianCD_PostQAQC.zip'];
% urlwrite(direccion,'OvarianCD_PostQAQC.zip');
% delete(h);
%
% % Descomprimir archivos
% unzip('OvarianCD_PostQAQC.zip')

%% Procesamiento de Mediciones
% En esta etapa se empaquetaran las mediciones en forma matricial. ...
    No se
% realizan operaciones adicionales de procesamiento ya que las ...
    mediciones
% del conjunto de datos descargado estan ya sometidas a un procesamiento
% previo.

local_repository  = 'OvarianCD_PostQAQC\';

cancerFiles = dir([local_repository '\Cancer\*.txt'])
normalFiles = dir([local_repository '\Normal\*.txt'])
```

```
33
34 files = [ strcat('Cancer/',{cancerFiles.name}) ...
35           strcat('Normal/',{normalFiles.name})];
36 N = numel(files)    % total number of files
37
38 repository = local_repository;
39 K = N;  % change to N to do all
40
41 [MZ,Y] = msbatchprocessing(repository,files(1:K));
42
43 disp(sprintf('Sequential time for %d spectrograms',K))
44
45 beep
46
47 Y = msnorm(MZ,Y,'QUANTILE',0.5,'LIMITS',[3500 11000],'MAX',50);
48 grp = [repmat({'Cancer'},size(cancerFiles));...
49        repmat({'Normal'},size(normalFiles))];
50
51 indxGrp  = [repmat(1,size(cancerFiles));...
52        repmat(0,size(normalFiles))];
53
54 cancerIdx = find(strcmp(grp,'Cancer'));
55 numel(cancerIdx)  % number of files in the "Cancer" subdirectory
56
57 normalIdx = find(strcmp(grp,'Normal'));
58 numel(normalIdx)  % number of files in the "Normal" subdirectory
59
60 hPlot = plot(MZ,Y(:,cancerIdx(1:5)),'b',MZ,Y(:,normalIdx(1:5)),'r');
61 axis([7650 8200 -2 50])
62 xlabel('Mass/Charge (M/Z)');ylabel('Relative Intensity')
63 legend(hPlot([1 end]),{'Ovarian Cancer','Control'})
64 title('Region of the pre-processed spectrograms')
65
66 DatosReagrupados = Y';
67 EtiquetasGrupo1Grupo2 = grp;
68
69 DatosGrupo1 = DatosReagrupados(find(indxGrp==1),:);
```

```
DatosGrupo2 = DatosReagrupados(find(indxGrp==0),:);

% Visualizacion de los Datos del Conjunto Arcene
msviewer(MZ,DatosReagrupados','Group',EtiquetasGrupo1Grupo2)

% Visualizacion en 3D de los Datos del Conjunto Arcene
EjeZ=repmat([1:100],10); j=0;
for i=1:1:10
    mattz= i*ones(1,1000); GrupoP(i,:) = mattz;
end
for i=11:1:20
    j=j+1;
    mattz= i*ones(1,1000); GrupoN(j,:) = mattz;
end

EspectrosDemoP=DatosGrupo1([1:10],[1:1000]); % 10 muestras por grupo
EspectrosDemoN=DatosGrupo2([1:10],[1:1000]); % 10 muestras por grupo
h = stem3(EjeZ,GrupoP,EspectrosDemoP,'r');
set(h, 'Marker', 'none');

hold on
h = stem3(EjeZ,GrupoN,EspectrosDemoN,'b');
hr = rotate3d;
hr.RotateStyle = 'box';
hr.Enable = 'on';
set(h, 'Marker', 'none');
ylabel('Mediciones')
xlabel('M/Z')
zlabel('Intensidades')
title('Visualizacion en 3D del Conjunto OvarianCancer')
hold off

print('-dtiff','-r500','3DArcene.jpg')

% Visualizacion en Mapa de Calor
msheatmap(MZ,DatosReagrupados','Group',EtiquetasGrupo1Grupo2)
```

A.1.3 CONJUNTO OVARIANDATASET8-7-02

```
%% BUSQUEDA DE ZONAS DE BIOMARCACION EN EL CONJUNTO DE DATOS OVARIAN
% El presente documento es un reporte de la busqueda de zonas de
% biomarcacion en el conjunto de datos ARCENE. Este script descarga los
% datos directamente del sitio web de Web Center for Machine ...
    Learning and
% Intelligent Systems, los procesa, analiza, extrae resultados, los ...
    valida
% y reporta las curvas de analisis directamente en formato html.
%%
tic
%% Descarga de Archivos
%
% Descargando Archivos del Sitio Web Center for Machine Learning and
% Intelligent Systems - Machine Learning Repository
% https://archive.ics.uci.edu/ml/datasets/Arcene

clc;
clear;

% subphases = 14; stage = 0;
% h = waitbar(0,'Inicializando descarga...','Name', ...
%     'Descargando Conjunto de Datos');
% stage = stage + 1;
% waitbar(stage/subphases,h,'Descargando OvarianCD-PostQAQC.zip');
% direccion = ['http://home.ccr.cancer.gov/' ...
%     'ncifdaproteomics/OvarianCD_PostQAQC.zip'];
% urlwrite(direccion,'OvarianCD_PostQAQC.zip');
% delete(h);
%
% % Descomprimir archivos
% unzip('OvarianDataset8-7-02.zip')

%% Procesamiento de Mediciones
```

```
% En esta etapa se empaquetaran las mediciones en forma matricial. ...
    No se
% realizan operaciones adicionales de procesamiento ya que las ...
    mediciones
% del conjunto de datos descargado estan ya sometidas a un procesamiento
% previo.

local_repository  = pwd;

cancerFiles = dir([local_repository '\Ovarian Cancer\*.csv'])
normalFiles = dir([local_repository '\Control\*.csv'])

files = [ strcat('Cancer/',{cancerFiles.name}) ...
          strcat('Normal/',{normalFiles.name})];
N = numel(files)    % total number of files

% d = dir(['*.csv']);
nCancer = length(cancerFiles);
% datos{1} = [0 0];

root_path = pwd;
cd([local_repository '\Ovarian Cancer'])
for i=1:nCancer
    [~,name,ext] = fileparts(cancerFiles(i).name)
    movefile(cancerFiles(i).name,[name '.txt'])
end
cd(root_path); clear root_path;

nNormal = length(normalFiles);

root_path = pwd;
cd([local_repository '\Control'])
for i=1:nNormal
    [~,name,ext] = fileparts(normalFiles(i).name)
    movefile(normalFiles(i).name,[name '.txt'])
end
cd(root_path); clear root_path;
```

```
67
68 % file = 'H:\user4\matlab\myfile.txt';
69 % [pathstr,name,ext] = fileparts(file)
70
71 K = numel(files);
72 Y = zeros(15000,K); % need to preset the size of Y for memory ...
     performance
73 MZ = zeros(15000,1);
74 parfor k = 1:3
75
76     file = [repository files{1}];
77
78     % read the two-column text file with mass-charge and intensity ...
         values
79     fid = fopen(file,'r');
80     ftext = textscan(fid, '%s %u8 %s');
81     fclose(fid);
82     signal = ftext{1};
83     intensity = ftext{2};
84
85     % resample the signal to 15000 points between 2000 and 11900
86     mzout = ...
         (sqrt(2000)+(0:(15000-1))'*diff(sqrt([2000,11900]))/15000).^2;
87     [mz,YR] = msresample(signal,intensity,mzout);
88
89     % align the spectrograms to two good reference peaks
90     P = [3883.766 7766.166];
91     YA = msalign(mz,YR,P,'WIDTH',2);
92
93     % estimate and adjust the background
94     YB = msbackadj(mz,YA,'STEP',50,'WINDOW',50);
95
96     % reduce the noise using a nonparametric filter
97     Y(:,k) = mslowess(mz,YB,'SPAN',5);
98
99     % the mass/charge vector is the same for all spectra after the ...
         resample
```

```
100     if k==1
101         MZ(:,k) = mz;
102     end
103 end
104
105 repository = local_repository;
106 K = N;  % change to N to do all
107
108 [MZ,Y] = msbatchprocessing(repository,files(1:K));
109
110 disp(sprintf('Sequential time for %d spectrograms',K))
111
112 beep
113
114 Y = msnorm(MZ,Y,'QUANTILE',0.5,'LIMITS',[3500 11000],'MAX',50);
115
116 grp = [repmat({'Cancer'},size(cancerFiles));...
117        repmat({'Normal'},size(normalFiles))];
118
119 indxGrp  = [repmat(1,size(cancerFiles));...
120        repmat(0,size(normalFiles))];
121
122 cancerIdx = find(strcmp(grp,'Cancer'));
123 numel(cancerIdx)  % number of files in the "Cancer" subdirectory
124
125 normalIdx = find(strcmp(grp,'Normal'));
126 numel(normalIdx)  % number of files in the "Normal" subdirectory
127
128 hPlot = plot(MZ,Y(:,cancerIdx(1:5)),'b',MZ,Y(:,normalIdx(1:5)),'r');
129 axis([7650 8200 -2 50])
130 xlabel('Mass/Charge (M/Z)');ylabel('Relative Intensity')
131 legend(hPlot([1 end]),{'Ovarian Cancer','Control'})
132 title('Region of the pre-processed spectrograms')
133
134 DatosReagrupados = Y';
135 EtiquetasGrupo1Grupo2 = grp;
136
```

```
137 DatosGrupo1 = DatosReagrupados(find(indxGrp==1),:);
138 DatosGrupo2 = DatosReagrupados(find(indxGrp==0),:);
139
140 % Visualizacion de los Datos del Conjunto Arcene
141 msviewer(MZ,DatosReagrupados','Group',EtiquetasGrupo1Grupo2)
142
143 % Visualizacion en 3D de los Datos del Conjunto Arcene
144 EjeZ=repmat([1:100],10); j=0;
145 for i=1:1:10
146     mattz= i*ones(1,1000); GrupoP(i,:) = mattz;
147 end
148 for i=11:1:20
149     j=j+1;
150     mattz= i*ones(1,1000); GrupoN(j,:) = mattz;
151 end
152 EspectrosDemoP=DatosGrupo1([1:10],[1:1000]); % 10 muestras por grupo
153 EspectrosDemoN=DatosGrupo2([1:10],[1:1000]); % 10 muestras por grupo
154 h = stem3(EjeZ,GrupoP,EspectrosDemoP,'r');
155 set(h, 'Marker', 'none');
156
157 hold on
158 h = stem3(EjeZ,GrupoN,EspectrosDemoN,'b');
159 hr = rotate3d;
160 hr.RotateStyle = 'box';
161 hr.Enable = 'on';
162 set(h, 'Marker', 'none');
163 ylabel('Mediciones')
164 xlabel('M/Z')
165 zlabel('Intensidades')
166 title('Visualizacion en 3D del Conjunto OvarianCancer')
167 hold off
168
169 print('-dtiff','-r500','3DArcene.jpg')
170
171 % Visualizacion en Mapa de Calor
172 msheatmap(MZ,DatosReagrupados','Group',EtiquetasGrupo1Grupo2)
```

A.2 CÓDIGOS DE MATLAB PARA SELECCIÓN DE CARACTERÍSTICAS DISCRIMINANTES DE LOS GRUPOS DE MEDICIONES

A.2.1 CONJUNTO ARCENE

```
%%=====================================================================%%
%% Seleccion de Caracteristicas Discriminantes / Filtro Estadistico
%%=====================================================================%%

% Esta etapa carga los datos empaquetados en como ConjuntoDatos en la
% memoria de Matlab, sobre los cuales se realizara la busqueda de
% caracteristicas discriminantes usando las pruebas estadisticas de
% t-student, wilcoxon y chi2 de manera independiente. Los resultados se
% agrupan y filtran al final, eliminando la informacion redundante.
%%=====================================================================%%
%
subphases = 13; stage = 0;
h = waitbar(0,'Cargando datos en Memoria ...','Name', ...
    'Procesamiento de Mediciones');
stage = stage + 1;
waitbar(stage/subphases,h, ...
    'Seleccionando de Caracteristicas Discriminantes');
%
% Mapa de Correlacion
LabelsC = num2str(EtiquetasGrupo1Grupo2);
corrmap(DatosReagrupados',LabelsC,1)
%
% Particion del Conjunto de Datos en Entrenamiento y Pruebas
rng(5000,'multFibonacci');
IndxParticion = cvpartition(EtiquetasDatos,'holdout',50);

TrainDataG1 = ConjuntoDatos(IndxParticion.training,:);
TestDataG2 = ConjuntoDatos(IndxParticion.test,:);
LblTrainG1 = EtiquetasDatos(IndxParticion.training);
LblTestG2 = EtiquetasDatos(IndxParticion.test);
```

```
%
stage = stage + 1;
waitbar(stage/subphases,h,'Separacion de los grupos de analisis');
%
% Separacion de los grupos de analisis en DatosTrainGt1 y DatosTrainGt2
DatosTrainGt1 = TrainDataG1(grp2idx(LblTrainG1)==1,:);
LblDatosTrainGt1 = LblTrainG1(grp2idx(LblTrainG1)==1,:);
DatosTrainGt2 = TrainDataG1(grp2idx(LblTrainG1)==2,:);
LblDatosTrainGt2 = LblTrainG1(grp2idx(LblTrainG1)==2,:);
%
stage = stage + 1;
waitbar(stage/subphases,h,'Pruebas estadisticas, busqueda de ...
    p-valores');
%
% Pruebas estadisticas
[ h_ttest, p_ttest, ci_ttest] = ttest2(DatosTrainGt1,DatosTrainGt2, ...
    'Vartype','unequal');
[ p_Wilcoxon, h_Wilcoxon ] = ...
    PruebaWilcoxon(DatosTrainGt1,DatosTrainGt2);
[ p_chi2, h_chi2 ] = ...
    ChiCuadradoBondadAjuste(DatosTrainGt1,DatosTrainGt2);
%
% Impresion de las Curvas de la funcion distribucion acumulada F(x)
%
stage = stage + 1;
waitbar(stage/subphases,h, ...
    'Impresion de las Curvas de la funcion distribucion acumulada ...
        F(x)');
%
stage = stage + 1;
waitbar(stage/subphases,h, ...
    'Distribucon Acumulada de los p-Valores(t-Student)');
%
figure
ecdf(p_ttest);
h1 = get(gca,'children');
set(h1,'LineWidth',1.5,'Color','r');
```

```
64 xlabel('P valor');
65 ylabel('CDF(t-Student)');
66 title('Funcion de Distribucion Acumulada de los p-Valores(t-Student)')
67 set(gca, ...
68   'Box'         , 'off'     , ...
69   'TickDir'     , 'out'     , ...
70   'TickLength'  , [.02 .02] , ...
71   'XMinorTick'  , 'on'      , ...
72   'YMinorTick'  , 'on'      , ...
73   'YGrid'       , 'on'      , ...
74   'XGrid'       , 'on'      , ...
75   'XColor'      , [.3 .3 .3], ...
76   'YColor'      , [.3 .3 .3], ...
77   'YTick'       , 0:1/10:1, ...
78   'XTick'       , 0:1/10:1, ...
79   'LineWidth'   , 1         );
80 print('-dtiff','-r500','tStudentArcene.jpg')
81 stage = stage + 1;
82 waitbar(stage/subphases,h, ...
83     'Distribucon Acumulada de los p-Valores(Wilcoxon)');
84 %
85 figure
86 ecdf(p_Wilcoxon)
87 h1 = get(gca,'children');
88 set(h1,'LineWidth',1.5,'Color','b');
89 xlabel('P valor');
90 ylabel('CDF(t-Wilcoxon)');
91 title('Funcion de Distribucion Acumulada de los p-Valores(Wilcoxon)')
92 set(gca, ...
93   'Box'         , 'off'     , ...
94   'TickDir'     , 'out'     , ...
95   'TickLength'  , [.02 .02] , ...
96   'XMinorTick'  , 'on'      , ...
97   'YMinorTick'  , 'on'      , ...
98   'YGrid'       , 'on'      , ...
99   'XGrid'       , 'on'      , ...
100  'XColor'      , [.3 .3 .3], ...
```

```
101   'YColor'      , [.3 .3 .3], ...
102   'YTick'       , 0:1/10:1, ...
103   'XTick'       , 0:1/10:1, ...
104   'LineWidth'   , 1         );
105 print('-dtiff','-r500','WilcoxonArcene.jpg')
106 %
107 stage = stage + 1;
108 waitbar(stage/subphases,h, ...
109     'Distribucion Acumulada de los p-Valores(Chi2)');
110 %
111 figure
112 ecdf(p_chi2)
113 h1 = get(gca,'children');
114 set(h1,'LineWidth',1.5,'Color','m');
115 xlabel('P valor');
116 ylabel('CDF(Chi2)');
117 title('Funcion de Distribucion Acumulada de los p-Valores(Chi2)')
118 set(gca, ...
119   'Box'         , 'off'     , ...
120   'TickDir'     , 'out'     , ...
121   'TickLength'  , [.02 .02] , ...
122   'XMinorTick'  , 'on'      , ...
123   'YMinorTick'  , 'on'      , ...
124   'YGrid'       , 'on'      , ...
125   'XGrid'       , 'on'      , ...
126   'XColor'      , [.3 .3 .3], ...
127   'YColor'      , [.3 .3 .3], ...
128   'YTick'       , 0:1/10:1, ...
129   'XTick'       , 0:1/10:1, ...
130   'LineWidth'   , 1         );
131 print('-dtiff','-r500','ChiArcene.jpg')
132 %
133 stage = stage + 1;
134 waitbar(stage/subphases,h, 'Busqueda de las mutaciones intergrupales');
135 %
136 % Busqueda de las mutaciones intergrupales en funcion de los valores de
137 % probabilidad de los p valores por cada una de las pruebas estadisticas
```

```
% aplicadas

% Ordenar los p-valores de la prueba t-test
[¬,CoeficientesIndicesP_ttest]= sort(p_ttest,2,'ascend');
% Ordenar los p-valores de la prueba Wilcoxon
[¬,CoeficientesIndicesP_Wilcoxon]= sort(p_Wilcoxon,2,'ascend');
% Ordenar los p-valores de la prueba chi2
[¬,CoeficientesIndicesP_chi2]= sort(p_chi2,2,'descend');
%
% Vector de Indices para la Busqueda, el valor de 2500 representa el 25%
% del total de las caracteristicas del conjunto original Arcene
%
GruposDeBusquedaSobreentrenamiento = 100:100:2500;
%
% Control de Numeros Aleatorios
rng(7000,'multFibonacci');
%
% En las siguientes lineas de codigo se busca un punto de inflexion ...
    entre
% las curvas de aprendisaje y la curva de resubstitucion
%
stage = stage + 1;
waitbar(stage/subphases,h, ...
    'Busqueda en las caracteristicas de la t-Student');
%
% Busqueda en 25 iteraciones

% t-Student
for i=1:25
   fs_ttest = CoeficientesIndicesP_ttest( ...
       1:GruposDeBusquedaSobreentrenamiento(i));

   ClassTreeEnsT = fitensemble(TrainDataG1(:,fs_ttest),...
       LblTrainG1,'AdaBoostM1',100,'Tree','CrossVal','on');
   testMCE_ttest(i) = kfoldLoss(ClassTreeEnsT);
   ClassTreeEnsR = fitensemble(TrainDataG1(:,fs_ttest), ...
       LblTrainG1,'AdaBoostM1',100,'Tree');
```

```
174    resubMCE_ttest(i) = resubLoss(ClassTreeEnsR,'lossfun','exponential');
175 end
176 %
177 figure
178  plot(GruposDeBusquedaSobreentrenamiento(1:25), testMCE_ttest,'-o', ...
179      GruposDeBusquedaSobreentrenamiento(1:25),resubMCE_ttest,'-r^');
180  xlabel('Numbero de Caracteristicas');
181  ylabel('Error de Clasification');
182
183  grid on;
184  legend({'Curva de Aprendizaje' 'Curva de Resubstitucion'}, ...
185      'location','SE');
186  title('Punto de Inflexion en el Subconjunto t-Student');
187 print('-dtiff','-r500','StudentInflexionArcene.jpg')
188 %
189 stage = stage + 1;
190 waitbar(stage/subphases,h, ...
191     'Busqueda en las caracteristicas de la Wilcoxon');
192 %
193 % Vector de Indices para la Busqueda
194 GruposDeBusquedaSobreentrenamiento = 100:100:2500;
195
196  % Wilcoxon
197 for i=1:25
198    fs_wilcoxon = CoeficientesIndicesP_Wilcoxon( ...
199        1:GruposDeBusquedaSobreentrenamiento(i));
200
201    ClassTreeEnsT = fitensemble(TrainDataG1(:,fs_wilcoxon),...
202        LblTrainG1,'AdaBoostM1',100,'Tree','CrossVal','on');
203    testMCE_wilcoxon(i) = kfoldLoss(ClassTreeEnsT);
204    ClassTreeEnsR = fitensemble(TrainDataG1(:,fs_wilcoxon), ...
205        LblTrainG1,'AdaBoostM1',100,'Tree');
206    resubMCE_wilcoxon(i) = resubLoss(ClassTreeEnsR,'lossfun',...
207        'exponential');
208 end
209 %
210 figure
```

```
211  plot(GruposDeBusquedaSobreentrenamiento(1:25), ...
        testMCE_wilcoxon,'-o', ...
212      GruposDeBusquedaSobreentrenamiento(1:25),resubMCE_wilcoxon,'-r^');
213  xlabel('Numbero de Caracteristicas');
214  ylabel('Error de Clasification');
215
216  grid on;
217  legend({'Curva de Aprendizaje' 'Curva de Resubstitucion'},...
218      'location','SE');
219  title('Punto de Inflexion en el Subconjunto Wilcoxon');
220 print('-dtiff','-r500','WilcoxonInflexionArcene.jpg')
221 %
222 stage = stage + 1;
223 waitbar(stage/subphases,h, 'Busqueda en las caracteristicas de la ...
       Chi2');
224 %
225 % Vector de Indices para la Busqueda
226 GruposDeBusquedaSobreentrenamiento = 100:100:1000;
227
228 % Chi2
229 for i=1:10
230    fs_chi2 = CoeficientesIndicesP_chi2( ...
231        1:GruposDeBusquedaSobreentrenamiento(i));
232
233    ClassTreeEnsT = fitensemble(TrainDataG1(:,fs_chi2),...
234        LblTrainG1,'AdaBoostM1',100,'Tree','CrossVal','on');
235    testMCE_chi2(i) = kfoldLoss(ClassTreeEnsT);
236    ClassTreeEnsR = fitensemble(TrainDataG1(:,fs_chi2), ...
237        LblTrainG1,'AdaBoostM1',100,'Tree');
238    resubMCE_chi2(i) = resubLoss(ClassTreeEnsR,'lossfun','exponential');
239 end
240 %
241 figure
242 plot(GruposDeBusquedaSobreentrenamiento(1:10), testMCE_chi2,'-o', ...
243      GruposDeBusquedaSobreentrenamiento(1:10),resubMCE_chi2,'-r^');
244 xlabel('Numbero de Caracteristicas');
245 ylabel('Error de Clasification');
```

```
246
247 grid on;
248 legend({'Curva de Aprendizaje' 'Curva de Resubstitucion'},...
249     'location','SE');
250 title('Punto de Inflexion en el Subconjunto Chi2');
251 print('-dtiff','-r500','ChiInflexionArcene.jpg')
252 %
253 stage = stage + 1;
254 waitbar(stage/subphases,h, 'Seleccion de Subconjuntos');
255 %
256 % Caracteristicas Seleccionadas
257 fs_ttestSelect = fs_ttest(1:300);
258 fs_wilcoxonSelect = fs_wilcoxon(1:200);
259 fs_chi2Select = fs_chi2(1:200);
260 %
261 fs_Total = horzcat( fs_ttestSelect, fs_wilcoxonSelect,fs_chi2Select);
262 length(fs_Total);
263 fs_Total = unique(fs_Total);
264 length(fs_Total);
265 stage = stage + 1;
266 waitbar(stage/subphases,h, ...
267     'Visualizacion de Caracteristicas Seleccionadas');
268 delete(h);
269 %
270 % Visualizacion de los Datos del Conjunto Arcene
271 msviewer(MZ,DatosReagrupados','Group',EtiquetasGrupo1Grupo2,...
272     'Markers',MZ(fs_Total))
273 %
274 % Visualizacion en Mapa de Calor
275 msheatmap(MZ,DatosReagrupados','Group',EtiquetasGrupo1Grupo2,...
276     'Markers',MZ(fs_Total))
277 %
278 % Filtrado
279 fs = fs_Total;
280 %
281 format long g
282 %
```

```
figure
bar(1:length(fs),sort(fs));   %226,7454,
 %
fs; [a,b] = size(fs);
fsord = sort(fs,2);

for i = 1:b-1;
    aux(i) = fsord(i)-fsord(i+1);
    aux = unique(abs(aux));
    size(abs(aux));
end

fss=sort(fs,2);
for j=1:b;
    for i=1:length(aux);
        zonas(i)=length(fss(diff(fss)==aux(i)));
    end
    zonas = unique(zonas);
end

[xx1,xx2] = size(zonas);
agrupar = xx2;
indx = clusterdata(fs',agrupar);
indx = indx';
featMean = [];

for i = 1:agrupar; % 25 numero de zonas
featMean(i) = mean(fs(find(indx==i)));
end

featMean = featMean';
featRed = [];
 %
for i = 1:agrupar;
    prueba = featMean(i);
    for j = 1:length(fs);
        indxSup = find(fs>prueba);
```

```
320         indInf = find(fs<prueba);
321         %clear prueba
322     end
323
324 prueba;
325     vectOrdenadosS = sort(fs(indxSup),'ascend');
326     indxSupM = find(fs==vectOrdenadosS(1));
327
328     vectOrdenadosI = sort(fs(indInf),'descend');
329     indInfM = find(fs==vectOrdenadosI(1));
330
331     featRed(2*i-1) = indxSupM(1);
332     featRed(2*i) = indInfM(1);
333     fs(featRed);
334 end
335 %
336 fss = fs(unique(featRed)); % toma valores q no se repiten
337 %
338 feattFinal = fss;
339 %
340 fs_TotalRed = feattFinal;
341 %
342 length(fs_TotalRed);
343 %
344 % Visualizacion de datos ya filtrado
345
346 % Visualizacion de los Datos del Conjunto Arcene
347 msviewer(MZ,DatosReagrupados','Group',EtiquetasGrupo1Grupo2,...
348     'Markers',MZ(fs_TotalRed))
349 %
350 % Visualizacion en Mapa de Calor
351 msheatmap(MZ,DatosReagrupados','Group',EtiquetasGrupo1Grupo2,...
352     'Markers',MZ(fs_TotalRed))
```

A.2.2 CONJUNTO OVARIAN CANCER QA-QC

```
%% Seleccion de Caracteristicas Discriminantes
% Esta etapa carga los datos empaquetados en como ConjuntoDatos en la
% memoria de Matlab, sobre los cuales se realizara la busqueda de
% caracteristicas discriminantes usando las pruebas estadisticas de
% t-student, wilcoxon y chi2 de manera independiente. Los resultados se
% agrupan y filtran al final, eliminando la informacion redundante.
% Mapa de Correlacion

LabelsC = num2str(indxGrp);
corrmap(DatosReagrupados',LabelsC,1)
ConjuntoDatos = DatosReagrupados;
EtiquetasDatos = EtiquetasGrupo1Grupo2;

% Particion del Conjunto de Datos en Entrenamiento y Pruebas
rng(5000,'multFibonacci');
IndxParticion = cvpartition(EtiquetasDatos,'holdout',50);
TrainDataG1 = ConjuntoDatos(IndxParticion.training,:);
TestDataG2 = ConjuntoDatos(IndxParticion.test,:);
LblTrainG1 = EtiquetasDatos(IndxParticion.training);
LblTestG2 = EtiquetasDatos(IndxParticion.test);

% Separacion de los grupos de analisis en DatosTrainGt1 y DatosTrainGt2
DatosTrainGt1 = TrainDataG1(grp2idx(LblTrainG1)==1,:);
LblDatosTrainGt1 = LblTrainG1(grp2idx(LblTrainG1)==1,:);
DatosTrainGt2 = TrainDataG1(grp2idx(LblTrainG1)==2,:);
LblDatosTrainGt2 = LblTrainG1(grp2idx(LblTrainG1)==2,:);

% Pruebas estadisticas
[ h_ttest, p_ttest, ci_ttest] = ttest2(DatosTrainGt1,DatosTrainGt2, ...
    'Vartype','unequal');
[ p_Wilcoxon, h_Wilcoxon ] = ...
    PruebaWilcoxon(DatosTrainGt1,DatosTrainGt2);
[ p_chi2, h_chi2 ] = ...
    ChiCuadradoBondadAjuste(DatosTrainGt1,DatosTrainGt2);
```

```
33
34 % Impresion de las Curvas de la funcion distribucion acumulada F(x)
35 figure
36 ecdf(p_ttest);
37 h1 = get(gca,'children');
38 set(h1,'LineWidth',1.5,'Color','r');
39 xlabel('P valor');
40 ylabel('CDF(t-Student)');
41 title('Funcion de Distribucion Acumulada de los p-Valores(t-Student)')
42 set(gca, ...
43   'Box'         , 'off'     , ...
44   'TickDir'     , 'out'     , ...
45   'TickLength'  , [.02 .02] , ...
46   'XMinorTick'  , 'on'      , ...
47   'YMinorTick'  , 'on'      , ...
48   'YGrid'       , 'on'      , ...
49   'XGrid'       , 'on'      , ...
50   'XColor'      , [.3 .3 .3], ...
51   'YColor'      , [.3 .3 .3], ...
52   'YTick'       , 0:1/10:1, ...
53   'XTick'       , 0:1/10:1, ...
54   'LineWidth'   , 1         );
55 print('-dtiff','-r500','tStudentArcene.jpg')
56
57 figure
58 ecdf(p_Wilcoxon)
59 h1 = get(gca,'children');
60 set(h1,'LineWidth',1.5,'Color','b');
61 xlabel('P valor');
62 ylabel('CDF(t-Wilcoxon)');
63 title('Funcion de Distribucion Acumulada de los p-Valores(Wilcoxon)')
64 set(gca, ...
65   'Box'         , 'off'     , ...
66   'TickDir'     , 'out'     , ...
67   'TickLength'  , [.02 .02] , ...
68   'XMinorTick'  , 'on'      , ...
69   'YMinorTick'  , 'on'      , ...
```

```
  'YGrid'       , 'on'      , ...
  'XGrid'       , 'on'      , ...
  'XColor'      , [.3 .3 .3], ...
  'YColor'      , [.3 .3 .3], ...
  'YTick'       , 0:1/10:1, ...
  'XTick'       , 0:1/10:1, ...
  'LineWidth'   , 1         );
print('-dtiff','-r500','WilcoxonArcene.jpg')

figure
ecdf(p_chi2)
h1 = get(gca,'children');
set(h1,'LineWidth',1.5,'Color','m');
xlabel('P valor');
ylabel('CDF(Chi2)');
title('Funcion de Distribucion Acumulada de los p-Valores(Chi2)')
set(gca, ...
  'Box'         , 'off'     , ...
  'TickDir'     , 'out'     , ...
  'TickLength'  , [.02 .02] , ...
  'XMinorTick'  , 'on'      , ...
  'YMinorTick'  , 'on'      , ...
  'YGrid'       , 'on'      , ...
  'XGrid'       , 'on'      , ...
  'XColor'      , [.3 .3 .3], ...
  'YColor'      , [.3 .3 .3], ...
  'YTick'       , 0:1/10:1, ...
  'XTick'       , 0:1/10:1, ...
  'LineWidth'   , 1         );
print('-dtiff','-r500','ChiArcene.jpg')

% Busqueda de las mutaciones intergrupales en funcion de los valores de
% probabilidad de los p valores por cada una de las pruebas estadisticas
% aplicadas

% Ordenar los p-valores de la prueba t-test
[¬,CoeficientesIndicesP_ttest]= sort(p_ttest,2,'ascend');
```

```
% Ordenar los p-valores de la prueba Wilcoxon
[¬,CoeficientesIndicesP_Wilcoxon]= sort(p_Wilcoxon,2,'ascend');
% Ordenar los p-valores de la prueba chi2
[¬,CoeficientesIndicesP_chi2]= sort(p_chi2,2,'descend');

% Vector de Indices para la Busqueda, el valor de 2500 representa el 25%
% del total de las caracteristicas del conjunto original Arcene
GruposDeBusquedaSobreentrenamiento = 100:100:2500;

% Control de Numeros Aleatorios
rng(7000,'multFibonacci');

% En las siguientes lineas de codigo se busca un punto de inflexion ...
    entre
% las curvas de aprendisaje y la curva de resubstitucion

% Busqueda en 25 iteracionesTrainDataG1

% t-Student
for i=1:25
    i
   fs_ttest = CoeficientesIndicesP_ttest( ...
       1:GruposDeBusquedaSobreentrenamiento(i));
   ClassTreeEnsT = fitensemble(TrainDataG1(:,fs_ttest),...
       LblTrainG1,'AdaBoostM1',50,'Tree','CrossVal','on');
   testMCE_ttest(i) = kfoldLoss(ClassTreeEnsT);
   ClassTreeEnsR = fitensemble(TrainDataG1(:,fs_ttest), ...
       LblTrainG1,'AdaBoostM1',50,'Tree');
   resubMCE_ttest(i) = ...
       resubLoss(ClassTreeEnsR,'lossfun','classiferror');
end

figure
 plot(GruposDeBusquedaSobreentrenamiento(1:25), ...
      testMCE_ttest(1:25),'-o', ...
     GruposDeBusquedaSobreentrenamiento(1:25),resubMCE_ttest(1:25),'-r^');
 xlabel('Numbero de Caracteristicas');
```

```
141  ylabel('Error de Clasification');
142  grid on;
143  legend({'Curva de Aprendizaje' 'Curva de Resubstitucion'}, ...
144      'location','SE');
145  title('Punto de Inflexion en el Subconjunto t-Student');
146 print('-dtiff','-r500','StudentInflexionArcene.jpg')
147
148 % Vector de Indices para la Busqueda
149 GruposDeBusquedaSobreentrenamiento = 100:100:2500;
150  % Wilcoxon
151 for i=1:25
152    fs_wilcoxon = CoeficientesIndicesP_Wilcoxon( ...
153        1:GruposDeBusquedaSobreentrenamiento(i));
154    ClassTreeEnsT = fitensemble(TrainDataG1(:,fs_wilcoxon),...
155        LblTrainG1,'AdaBoostM1',100,'Tree','CrossVal','on');
156    testMCE_wilcoxon(i) = kfoldLoss(ClassTreeEnsT);
157    ClassTreeEnsR = fitensemble(TrainDataG1(:,fs_wilcoxon), ...
158        LblTrainG1,'AdaBoostM1',100,'Tree');
159    resubMCE_wilcoxon(i) = resubLoss(ClassTreeEnsR,'lossfun',...
160        'exponential');
161 end
162
163 figure
164  plot(GruposDeBusquedaSobreentrenamiento(1:25), ...
       testMCE_wilcoxon,'-o', ...
165     GruposDeBusquedaSobreentrenamiento(1:25),resubMCE_wilcoxon,'-r^');
166  xlabel('Numbero de Caracteristicas');
167  ylabel('Error de Clasification');
168  grid on;
169  legend({'Curva de Aprendizaje' 'Curva de Resubstitucion'},...
170      'location','SE');
171  title('Punto de Inflexion en el Subconjunto Wilcoxon');
172 print('-dtiff','-r500','WilcoxonInflexionArcene.jpg')
173
174 % Vector de Indices para la Busqueda
175 GruposDeBusquedaSobreentrenamiento = 100:100:1000;
176
```

```
% Chi2
for i=1:10
    fs_chi2 = CoeficientesIndicesP_chi2( ...
        1:GruposDeBusquedaSobreentrenamiento(i));
    ClassTreeEnsT = fitensemble(TrainDataG1(:,fs_chi2),...
        LblTrainG1,'AdaBoostM1',100,'Tree','CrossVal','on');
    testMCE_chi2(i) = kfoldLoss(ClassTreeEnsT);
    ClassTreeEnsR = fitensemble(TrainDataG1(:,fs_chi2), ...
        LblTrainG1,'AdaBoostM1',100,'Tree');
    resubMCE_chi2(i) = resubLoss(ClassTreeEnsR,'lossfun','exponential');
end

figure
plot(GruposDeBusquedaSobreentrenamiento(1:10), testMCE_chi2,'-o', ...
     GruposDeBusquedaSobreentrenamiento(1:10),resubMCE_chi2,'-r^');
xlabel('Numbero de Caracteristicas');
ylabel('Error de Clasification');
grid on;
legend({'Curva de Aprendizaje' 'Curva de Resubstitucion'},...
    'location','SE');
title('Punto de Inflexion en el Subconjunto Chi2');
print('-dtiff','-r500','ChiInflexionArcene.jpg')

% Caracteristicas Seleccionadas
fs_ttestSelect = fs_ttest(1:400); %300
fs_wilcoxonSelect = fs_wilcoxon(1:400);
fs_chi2Select = fs_chi2(1:400);

fs_Total = horzcat( fs_ttestSelect, fs_wilcoxonSelect,fs_chi2Select);
length(fs_Total);
fs_Total = unique(fs_Total);
length(fs_Total)

% Visualizacion de los Datos del Conjunto Arcene
msviewer(MZ,DatosReagrupados','Group',EtiquetasGrupo1Grupo2,...
    'Markers',MZ(fs_Total))

```

```
214 % Visualizacion en Mapa de Calor
215 msheatmap(MZ,DatosReagrupados','Group',EtiquetasGrupo1Grupo2,...
216     'Markers',MZ(fs_Total))
217
218 % Filtrado
219 fs = fs_Total;
220 format long g
221 figure
222 bar(1:length(fs),sort(fs));   %226,7454,
223
224 fs;
225 [a,b] = size(fs);
226 fsord = sort(fs,2);
227 for i = 1:b-1;
228     aux(i) = fsord(i)-fsord(i+1);
229     aux = unique(abs(aux));
230     size(abs(aux));
231 end
232 fss=sort(fs,2);
233 for j=1:b;
234     for i=1:length(aux);
235         zonas(i)=length(fss(diff(fss)==aux(i)));
236     end
237     zonas = unique(zonas);
238 end
239 [xx1,xx2] = size(zonas);
240 agrupar = xx2;
241 indx = clusterdata(fs',agrupar);
242 indx = indx';
243 featMean = [];
244
245 for i = 1:agrupar;  % 25 numero de zonas
246 featMean(i) = mean(fs(find(indx==i)));
247 end
248 featMean = featMean';
249 featRed = [];
250
```

```
for i = 1:agrupar;
    prueba = featMean(i);
    for j = 1:length(fs);
        indxSup = find(fs>prueba);
        indInf = find(fs<prueba);
        end
    prueba;
    vectOrdenadosS = sort(fs(indxSup),'ascend');
    indxSupM = find(fs==vectOrdenadosS(1));

    vectOrdenadosI = sort(fs(indInf),'descend');
    indInfM = find(fs==vectOrdenadosI(1));

    clear prueba
    featRed(2*i-1) = indxSupM(1);
    featRed(2*i) = indInfM(1);
    fs(featRed);
end

fss = fs(unique(featRed)); % toma valores q no se repiten
feattFinal = fss;
fs_TotalRed = feattFinal;
length(fs_TotalRed);

% Visualizacion de datos ya filtrados
% Visualizacion de los Datos del Conjunto Arcene
msviewer(MZ,DatosReagrupados','Group',EtiquetasGrupo1Grupo2,...
    'Markers',MZ(fs_TotalRed))

% Visualizacion en Mapa de Calor
msheatmap(MZ,DatosReagrupados','Group',EtiquetasGrupo1Grupo2,...
    'Markers',MZ(fs_TotalRed))
```

A.2.3 CONJUNTO OVARIANDATASET8-7-02

```
%% Seleccion de Caracteristicas Discriminantes
% Esta etapa carga los datos empaquetados en como ConjuntoDatos en la
% memoria de Matlab, sobre los cuales se realizara la busqueda de
% caracteristicas discriminantes usando las pruebas estadisticas de
% t-student, wilcoxon y chi2 de manera independiente. Los resultados se
% agrupan y filtran al final, eliminando la informacion redundante.

% Mapa de Correlacion
LabelsC = num2str(indxGrp);
corrmap(DatosReagrupados',LabelsC,1)
ConjuntoDatos = DatosReagrupados;
EtiquetasDatos = EtiquetasGrupo1Grupo2;

% Particion del Conjunto de Datos en Entrenamiento y Pruebas
rng(5000,'multFibonacci');
IndxParticion = cvpartition(EtiquetasDatos,'holdout',50);

TrainDataG1 = ConjuntoDatos(IndxParticion.training,:);
TestDataG2 = ConjuntoDatos(IndxParticion.test,:);
LblTrainG1 = EtiquetasDatos(IndxParticion.training);
LblTestG2 = EtiquetasDatos(IndxParticion.test);

% Separacion de los grupos de analisis en DatosTrainGt1 y DatosTrainGt2
DatosTrainGt1 = TrainDataG1(grp2idx(LblTrainG1)==1,:);
LblDatosTrainGt1 = LblTrainG1(grp2idx(LblTrainG1)==1,:);
DatosTrainGt2 = TrainDataG1(grp2idx(LblTrainG1)==2,:);
LblDatosTrainGt2 = LblTrainG1(grp2idx(LblTrainG1)==2,:);

% Pruebas estadisticas
[ h_ttest, p_ttest, ci_ttest] = ttest2(DatosTrainGt1,DatosTrainGt2, ...
    'Vartype','unequal');
[ p_Wilcoxon, h_Wilcoxon ] = ...
    PruebaWilcoxon(DatosTrainGt1,DatosTrainGt2);
```

```
[ p_chi2, h_chi2 ] = ...
    ChiCuadradoBondadAjuste(DatosTrainGt1,DatosTrainGt2);

% Impresion de las Curvas de la funcion distribucion acumulada F(x)

figure
ecdf(p_ttest);
h1 = get(gca,'children');
set(h1,'LineWidth',1.5,'Color','r');
xlabel('P valor');
ylabel('CDF(t-Student)');
title('Funcion de Distribucion Acumulada de los p-Valores(t-Student)')
set(gca, ...
  'Box'         , 'off'     , ...
  'TickDir'     , 'out'     , ...
  'TickLength'  , [.02 .02] , ...
  'XMinorTick'  , 'on'      , ...
  'YMinorTick'  , 'on'      , ...
  'YGrid'       , 'on'      , ...
  'XGrid'       , 'on'      , ...
  'XColor'      , [.3 .3 .3], ...
  'YColor'      , [.3 .3 .3], ...
  'YTick'       , 0:1/10:1, ...
  'XTick'       , 0:1/10:1, ...
  'LineWidth'   , 1         );
print('-dtiff','-r500','tStudentArcene.jpg')

figure
ecdf(p_Wilcoxon)
h1 = get(gca,'children');
set(h1,'LineWidth',1.5,'Color','b');
xlabel('P valor');
ylabel('CDF(t-Wilcoxon)');
title('Funcion de Distribucion Acumulada de los p-Valores(Wilcoxon)')
set(gca, ...
  'Box'         , 'off'     , ...
  'TickDir'     , 'out'     , ...
```

```
69    'TickLength'   , [.02 .02] , ...
70    'XMinorTick'   , 'on'      , ...
71    'YMinorTick'   , 'on'      , ...
72    'YGrid'        , 'on'      , ...
73    'XGrid'        , 'on'      , ...
74    'XColor'       , [.3 .3 .3], ...
75    'YColor'       , [.3 .3 .3], ...
76    'YTick'        , 0:1/10:1, ...
77    'XTick'        , 0:1/10:1, ...
78    'LineWidth'    , 1         );
79 print('-dtiff','-r500','WilcoxonArcene.jpg')
80
81 figure
82 ecdf(p_chi2)
83 h1 = get(gca,'children');
84 set(h1,'LineWidth',1.5,'Color','m');
85 xlabel('P valor');
86 ylabel('CDF(Chi2)');
87 title('Funcion de Distribucion Acumulada de los p-Valores(Chi2)')
88 set(gca, ...
89    'Box'          , 'off'     , ...
90    'TickDir'      , 'out'     , ...
91    'TickLength'   , [.02 .02] , ...
92    'XMinorTick'   , 'on'      , ...
93    'YMinorTick'   , 'on'      , ...
94    'YGrid'        , 'on'      , ...
95    'XGrid'        , 'on'      , ...
96    'XColor'       , [.3 .3 .3], ...
97    'YColor'       , [.3 .3 .3], ...
98    'YTick'        , 0:1/10:1, ...
99    'XTick'        , 0:1/10:1, ...
100   'LineWidth'    , 1         );
101 print('-dtiff','-r500','ChiArcene.jpg')
102
103 % Busqueda de las mutaciones intergrupales en funcion de los valores de
104 % probabilidad de los p valores por cada una de las pruebas estadisticas
105 % aplicadas
```

```
106
107 % Ordenar los p-valores de la prueba t-test
108 [¬,CoeficientesIndicesP_ttest]= sort(p_ttest,2,'ascend');
109 % Ordenar los p-valores de la prueba Wilcoxon
110 [¬,CoeficientesIndicesP_Wilcoxon]= sort(p_Wilcoxon,2,'ascend');
111 % Ordenar los p-valores de la prueba chi2
112 [¬,CoeficientesIndicesP_chi2]= sort(p_chi2,2,'descend');
113
114 % Vector de Indices para la Busqueda, el valor de 2500 representa el 25%
115 % del total de las caracteristicas del conjunto original Arcene
116 GruposDeBusquedaSobreentrenamiento = 100:100:2500;
117
118 % Control de Numeros Aleatorios
119 rng(7000,'multFibonacci');
120
121 % En las siguientes lineas de codigo se busca un punto de inflexion ...
      entre
122 % las curvas de aprendisaje y la curva de resubstitucion
123 % Busqueda en 25 iteracionesTrainDataG1
124
125 % t-Student
126 for i=1:25
127     i
128    fs_ttest = CoeficientesIndicesP_ttest( ...
129        1:GruposDeBusquedaSobreentrenamiento(i));
130    ClassTreeEnsT = fitensemble(TrainDataG1(:,fs_ttest),...
131        LblTrainG1,'AdaBoostM1',50,'Tree','CrossVal','on');
132    testMCE_ttest(i) = kfoldLoss(ClassTreeEnsT);
133    ClassTreeEnsR = fitensemble(TrainDataG1(:,fs_ttest), ...
134        LblTrainG1,'AdaBoostM1',50,'Tree');
135    resubMCE_ttest(i) = ...
         resubLoss(ClassTreeEnsR,'lossfun','classiferror');
136 end
137
138 figure
139  plot(GruposDeBusquedaSobreentrenamiento(1:25), ...
       testMCE_ttest(1:25),'-o', ...
```

```
140      GruposDeBusquedaSobreentrenamiento(1:25),resubMCE_ttest(1:25),'-r^');
141  xlabel('Numbero de Caracteristicas');
142  ylabel('Error de Clasification');
143  grid on;
144  legend({'Curva de Aprendizaje' 'Curva de Resubstitucion'}, ...
145      'location','SE');
146  title('Punto de Inflexion en el Subconjunto t-Student');
147 print('-dtiff','-r500','StudentInflexionArcene.jpg')
148
149  % Vector de Indices para la Busqueda
150 GruposDeBusquedaSobreentrenamiento = 100:100:2500;
151
152  % Wilcoxon
153 for i=1:25
154    fs_wilcoxon = CoeficientesIndicesP_Wilcoxon( ...
155        1:GruposDeBusquedaSobreentrenamiento(i));
156    ClassTreeEnsT = fitensemble(TrainDataG1(:,fs_wilcoxon),...
157        LblTrainG1,'AdaBoostM1',100,'Tree','CrossVal','on');
158    testMCE_wilcoxon(i) = kfoldLoss(ClassTreeEnsT);
159    ClassTreeEnsR = fitensemble(TrainDataG1(:,fs_wilcoxon), ...
160        LblTrainG1,'AdaBoostM1',100,'Tree');
161    resubMCE_wilcoxon(i) = resubLoss(ClassTreeEnsR,'lossfun',...
162        'exponential');
163 end
164
165 figure
166  plot(GruposDeBusquedaSobreentrenamiento(1:25), ...
       testMCE_wilcoxon,'-o', ...
167      GruposDeBusquedaSobreentrenamiento(1:25),resubMCE_wilcoxon,'-r^');
168  xlabel('Numbero de Caracteristicas');
169  ylabel('Error de Clasification');
170  grid on;
171  legend({'Curva de Aprendizaje' 'Curva de Resubstitucion'},...
172      'location','SE');
173  title('Punto de Inflexion en el Subconjunto Wilcoxon');
174 print('-dtiff','-r500','WilcoxonInflexionArcene.jpg')
175
```

```
% Vector de Indices para la Busqueda
GruposDeBusquedaSobreentrenamiento = 100:100:1000;

% Chi2
for i=1:10
    fs_chi2 = CoeficientesIndicesP_chi2( ...
        1:GruposDeBusquedaSobreentrenamiento(i));
    ClassTreeEnsT = fitensemble(TrainDataG1(:,fs_chi2),...
        LblTrainG1,'AdaBoostM1',100,'Tree','CrossVal','on');
    testMCE_chi2(i) = kfoldLoss(ClassTreeEnsT);
    ClassTreeEnsR = fitensemble(TrainDataG1(:,fs_chi2), ...
        LblTrainG1,'AdaBoostM1',100,'Tree');
    resubMCE_chi2(i) = resubLoss(ClassTreeEnsR,'lossfun','exponential');
end

figure
plot(GruposDeBusquedaSobreentrenamiento(1:10), testMCE_chi2,'-o', ...
     GruposDeBusquedaSobreentrenamiento(1:10),resubMCE_chi2,'-r^');
xlabel('Numbero de Caracteristicas');
ylabel('Error de Clasification');
grid on;
legend({'Curva de Aprendizaje' 'Curva de Resubstitucion'},...
    'location','SE');
title('Punto de Inflexion en el Subconjunto Chi2');
print('-dtiff','-r500','ChiInflexionArcene.jpg')

% Caracteristicas Seleccionadas
fs_ttestSelect = fs_ttest(1:400); %300
fs_wilcoxonSelect = fs_wilcoxon(1:400);
fs_chi2Select = fs_chi2(1:400);
 fs_Total = horzcat( fs_ttestSelect, fs_wilcoxonSelect,fs_chi2Select);
length(fs_Total);
fs_Total = unique(fs_Total);
length(fs_Total)

% Visualizacion de los Datos del Conjunto Arcene
msviewer(MZ,DatosReagrupados','Group',EtiquetasGrupo1Grupo2,...
```

```
213     'Markers',MZ(fs_Total))
214
215 % Visualizacion en Mapa de Calor
216 msheatmap(MZ,DatosReagrupados','Group',EtiquetasGrupo1Grupo2,...
217     'Markers',MZ(fs_Total))
218
219 % Filtrado
220 fs = fs_Total;
221 format long g
222 figure
223 bar(1:length(fs),sort(fs));   %226,7454,
224 fs;
225 [a,b] = size(fs);
226 fsord = sort(fs,2);
227 for i = 1:b-1;
228     aux(i) = fsord(i)-fsord(i+1);
229     aux = unique(abs(aux));
230     size(abs(aux));
231 end
232
233 fss=sort(fs,2);
234 for j=1:b;
235     for i=1:length(aux);
236         zonas(i)=length(fss(diff(fss)==aux(i)));
237     end
238     zonas = unique(zonas);
239 end
240
241 [xx1,xx2] = size(zonas);
242 agrupar = xx2;
243 indx = clusterdata(fs',agrupar);
244 indx = indx';
245
246 featMean = [];
247 for i = 1:agrupar;  % 25 numero de zonas
248 featMean(i) = mean(fs(find(indx==i)));
249 end
```

```
250 featMean = featMean';
251 featRed = [];
252
253 for i = 1:agrupar;
254     prueba = featMean(i);
255     for j = 1:length(fs);
256         indxSup = find(fs>prueba);
257         indInf = find(fs<prueba);
258     end
259     prueba;
260     vectOrdenadosS = sort(fs(indxSup),'ascend');
261     indxSupM = find(fs==vectOrdenadosS(1));
262
263     vectOrdenadosI = sort(fs(indInf),'descend');
264     indInfM = find(fs==vectOrdenadosI(1));
265
266     clear prueba
267     featRed(2*i-1) = indxSupM(1);
268     featRed(2*i) = indInfM(1);
269     fs(featRed);
270 end
271
272 fss = fs(unique(featRed)); % toma valores q no se repiten
273 feattFinal = fss;
274 fs_TotalRed = feattFinal;
275 length(fs_TotalRed);
276
277 % Visualizacion de datos ya filtrados
278 % Visualizacion de los Datos del Conjunto Arcene
279 msviewer(MZ,DatosReagrupados','Group',EtiquetasGrupo1Grupo2,...
280     'Markers',MZ(fs_TotalRed))
281
282 % Visualizacion en Mapa de Calor
283 msheatmap(MZ,DatosReagrupados','Group',EtiquetasGrupo1Grupo2,...
284     'Markers',MZ(fs_TotalRed))
```

A.3 CÓDIGOS DE MATLAB PARA LA ETAPA DE VALIDACIÓN DE RESULTADOS

A.3.1 CONJUNTO ARCENE

```
1
2  %%=====================================================================%%
3  %% Validacion de Resultados usando Crossvalidation
4  %%=====================================================================%%
5  %
6  % Se cargan los datos en memoria, en tres grupos, entrenamiento,
7  % pruebas y validacion externa. Los datos de entrenamiento y pruebas se
8  % mezclan en un conjunto el cual evalua un clasificador ...
       independiente de
9  % Adaboost M1 usando arboles de clasificacion en Crossvalidation.
10 % La curva de aprendizaje evidenciara la presencia de overfitting o
11 % underfitting como criterio  % de continuar o abortar el proceso.
12 % Si no existe ninguno de los dos fenomenos, la siguiente etapa de
13 % prueba es la clasificacion usando muestras externas, donde
14 % el tercer conjunto usado no ha tenido contacto con ninguna de las
15 % etapas anteriores garantizando que no existe ningun tipo de ...
       influencia.
16 % Los resultados obtenidos son prometedores desde el punto de vista de
17 % la eficacia y aprendizaje del sistema.
18 %
19 %%=====================================================================%%
20 %
21 ConjuntoDatosRed = ConjuntoDatos(:,fs_TotalRed);
22 %
23 rng(5000,'multFibonacci');
24 %
25 % AdaBoostM1
26 ClassTreeEnsAda = fitensemble(ConjuntoDatosRed,EtiquetasDatos, ...
27     'AdaBoostM1',500,'Tree','KFold',10);
28 %
29 figure
```

```
plot(kfoldLoss(ClassTreeEnsAda,'mode','cumulative', ...
    'lossfun','classiferror'),'b');
xlabel('Numero de Rondas'); grid on
ylabel('Error de Clasificacion');
title('Curva de Aprendizaje / Adaboost.M1')
set(gca, ...
  'Box'         , 'off'     , ...
  'TickDir'     , 'out'     , ...
  'TickLength'  , [.02 .02] , ...
  'XMinorTick'  , 'on'      , ...
  'YMinorTick'  , 'on'      , ...
  'XColor'      , [.3 .3 .3], ...
  'YColor'      , [.3 .3 .3], ...
  'LineWidth'   , 1         );

print('-dtiff','-r500','CrossvalidationAdaArcene.jpg')
%
ClassTreeEnsAda = fitensemble(ConjuntoDatosRed,EtiquetasDatos, ...
    'AdaBoostM1',100,'Tree');
TestsDatos = load('ArceneTestsData');
%
[¬, Pesos] = predict(ClassTreeEnsAda,ConjuntoDatosRed);
%
[X,Y,T,AUC,OPTROCPT] = perfcurve(EtiquetasDatos',Pesos(:,2),1); % 1 ...
    clase
%
fprintf('El area bajo la curva es: %f\n\n',AUC)
%
figure

plot(X,Y)
h1 = get(gca,'children');

set(h1,'LineWidth',1.75,'Color','b');
xlabel('Tasa de Falsos Positivos'); ylabel('Tasa de Falsos Negativos');
title('Curva ROC / Adaboost / Arcene')
set(gca, ...
```

```
66    'Box'         , 'off'     , ...
67    'TickDir'     , 'out'     , ...
68    'TickLength'  , [.02 .02] , ...
69    'XMinorTick'  , 'on'      , ...
70    'YMinorTick'  , 'on'      , ...
71    'XColor'      , [.3 .3 .3], ...
72    'YColor'      , [.3 .3 .3], ...
73    'LineWidth'   , 1         );
74 print('-dtiff','-r500','CurvaROC.jpg')
75 %
76 [GruposMuestrasClasificadasT, PesosT] =  ...
77     predict(ClassTreeEnsAda,TestsDatos(:,fs_TotalRed));
78 Positivas = ...
79     GruposMuestrasClasificadasT(find(GruposMuestrasClasificadasT==1));
80 Negativas = ...
81     GruposMuestrasClasificadasT(find(GruposMuestrasClasificadasT==-1));
82 %
83 mensaje = ['Los valores teoricos de mediciones de los grupos son' ...
84     ' 310 \npara muestras positivas y 390 para muestras negativas\n\n'];
85 fprintf(mensaje)
86 %
87 mensaje = ['\nEl numero de muestras positivas clasificadas es de ...
      %d,' ...
88     '\ny el numero de muestras negativas es  %d\n'];
89
90 fprintf(mensaje,...
91     length(Positivas),length(Negativas))
92 %
93 fprintf('Existe un total de muestras mal clasificadas es   %d\n', ...
94     abs(390-length(Negativas)))
95 %
96 fprintf('El Error de Clasificacion es del (%d/700) --> %f%s  \n', ...
97     abs(390-length(Negativas)), ...
98     100*0.5*(abs(390-length(Negativas))+ ...
99     abs(310-length(Positivas)))/700,37)
```

A.3.2 CONJUNTO OVARIAN CANCER QA-QC

```
1
2  %% Validacion de Resultados
3  % DESCRIPTIVE TEXT
4  ConjuntoDatosRed = ConjuntoDatos(:,fs_TotalRed);
5  rng(5000,'multFibonacci');
6
7  % AdaBoostM1
8  ClassTreeEnsAda = fitensemble(ConjuntoDatosRed,EtiquetasDatos, ...
9      'AdaBoostM1',100,'Tree','KFold',10);
10
11 figure
12
13 plot(kfoldLoss(ClassTreeEnsAda,'mode','cumulative', ...
14     'lossfun','classiferror'),'b');
15
16 xlabel('Numero de Rondas'); grid on
17 ylabel('Error de Clasificacion');
18 title('Curva de Aprendizaje / Adaboost.M1')
19
20 set(gca, ...
21   'Box'         , 'off'     , ...
22   'TickDir'     , 'out'     , ...
23   'TickLength'  , [.02 .02] , ...
24   'XMinorTick'  , 'on'      , ...
25   'YMinorTick'  , 'on'      , ...
26   'XColor'      , [.3 .3 .3], ...
27   'YColor'      , [.3 .3 .3], ...
28   'LineWidth'   , 1         );
29
30 print('-dtiff','-r500','CrossvalidationAdaArcene.jpg')
31
32 ClassTreeEnsAda = fitensemble(ConjuntoDatosRed,EtiquetasDatos, ...
33     'AdaBoostM1',100,'Tree');
34
```

```
35 TestsDatos = TestDataG2;
36
37 [¬, Pesos] = predict(ClassTreeEnsAda,ConjuntoDatosRed);
38
39 [X,Y,T,AUC,OPTROCPT] = perfcurve(indxGrp',Pesos(:,2),0); % 1 clase
40
41 fprintf('El area bajo la curva es: %f\n\n',AUC)
42
43 figure
44
45 plot(X,Y)
46
47 h1 = get(gca,'children');
48 set(h1,'LineWidth',1.75,'Color','b');
49 xlabel('Tasa de Falsos Positivos'); ylabel('Tasa de Falsos Negativos');
50 title('Curva ROC / Adaboost / Ovarian')
51
52 set(gca, ...
53   'Box'         , 'off'     , ...
54   'TickDir'     , 'out'     , ...
55   'TickLength'  , [.02 .02] , ...
56   'XMinorTick'  , 'on'      , ...
57   'YMinorTick'  , 'on'      , ...
58   'XColor'      , [.3 .3 .3], ...
59   'YColor'      , [.3 .3 .3], ...
60   'LineWidth'   , 1         );
61 print('-dtiff','-r500','CurvaROC.jpg')
62
63 %================================================%
64
65 [GruposMuestrasClasificadasT, PesosT] =  ...
66     predict(ClassTreeEnsAda,TestsDatos(:,fs_TotalRed));
67
68 % Busqueda de muestras mal clasificadas
69     countxd = 0;
70     CountCancer = 0;
71     CountNormal = 0;
```

```
72
73 for i=1:length(GruposMuestrasClasificadasT)
74     if GruposMuestrasClasificadasT{i} ≠ LblTestG2{i}
75         countxd = 1 + countxd;
76     end
77
78     if GruposMuestrasClasificadasT{i} == 'Cancer'
79         CountCancer = 1 + CountCancer;
80     else
81         CountNormal = 1 + CountNormal;
82     end
83 end
84
85 % Indices numericos para la prueba externa
86 indxGrpTestExterna = indxGrp(IndxParticion.test);
87
88 mensaje = ['\nEl numero de muestras etiquetadas como Cancer es  %d,' ...
89     '\ny el numero de muestras etiquetadas como Normal es  %d\n'];
90
91 fprintf(mensaje,...
92     CountCancer,CountNormal)
93
94 fprintf('Existe un total de muestras mal clasificadas es   %d\n', ...
95     abs(countxd))
96
97 fprintf('El Error de Clasificacion es del (%d/700) --> %f%s  \n', ...
98     countxd/700,countxd/50,37)
99
100 toc
101
102 beep
```

A.3.3 CONJUNTO OVARIANDATASET8-7-02

```
%% Validacion de Resultados
% DESCRIPTIVE TEXT

ConjuntoDatosRed = ConjuntoDatos(:,fs_TotalRed);

rng(5000,'multFibonacci');

% AdaBoostM1
ClassTreeEnsAda = fitensemble(ConjuntoDatosRed,EtiquetasDatos, ...
    'AdaBoostM1',100,'Tree','KFold',10);

figure
plot(kfoldLoss(ClassTreeEnsAda,'mode','cumulative', ...
    'lossfun','classiferror'),'b');
xlabel('Numero de Rondas'); grid on
ylabel('Error de Clasificacion');
title('Curva de Aprendizaje / Adaboost.M1')
set(gca, ...
  'Box'         , 'off'     , ...
  'TickDir'     , 'out'     , ...
  'TickLength'  , [.02 .02] , ...
  'XMinorTick'  , 'on'      , ...
  'YMinorTick'  , 'on'      , ...
  'XColor'      , [.3 .3 .3], ...
  'YColor'      , [.3 .3 .3], ...
  'LineWidth'   , 1         );
print('-dtiff','-r500','CrossvalidationAdaArcene.jpg')

ClassTreeEnsAda = fitensemble(ConjuntoDatosRed,EtiquetasDatos, ...
    'AdaBoostM1',100,'Tree');
TestsDatos = TestDataG2;

[¬, Pesos] = predict(ClassTreeEnsAda,ConjuntoDatosRed);

```

```
35 [X,Y,T,AUC,OPTROCPT] = perfcurve(indxGrp',Pesos(:,2),0);  % 1 clase
36
37 fprintf('El area bajo la curva es: %f\n\n',AUC)
38
39 figure
40 plot(X,Y)
41 h1 = get(gca,'children');
42 set(h1,'LineWidth',1.75,'Color','b');
43 xlabel('Tasa de Falsos Positivos'); ylabel('Tasa de Falsos Negativos');
44 title('Curva ROC / Adaboost / Ovarian')
45 set(gca, ...
46   'Box'         , 'off'     , ...
47   'TickDir'     , 'out'     , ...
48   'TickLength'  , [.02 .02] , ...
49   'XMinorTick'  , 'on'      , ...
50   'YMinorTick'  , 'on'      , ...
51   'XColor'      , [.3 .3 .3], ...
52   'YColor'      , [.3 .3 .3], ...
53   'LineWidth'   , 1         );
54 print('-dtiff','-r500','CurvaROC.jpg')
55
56 %=================================================%
57
58 [GruposMuestrasClasificadasT, PesosT] =  ...
59     predict(ClassTreeEnsAda,TestsDatos(:,fs_TotalRed));
60
61 % Busqueda de muestras mal clasificadas
62     countxd = 0;
63     CountCancer = 0;
64     CountNormal = 0;
65 for i=1:length(GruposMuestrasClasificadasT)
66     if GruposMuestrasClasificadasT{i} ≠ LblTestG2{i}
67         countxd = 1 + countxd;
68     end
69     if GruposMuestrasClasificadasT{i} == 'Cancer'
70         CountCancer = 1 + CountCancer;
71     else
```

```
72          CountNormal = 1 + CountNormal;
73      end
74  end
75
76  % Indices numericos para la prueba externa
77  indxGrpTestExterna = indxGrp(IndxParticion.test);
78
79  mensaje = ['\nEl numero de muestras etiquetadas como Cancer es  %d,' ...
80      '\ny el numero de muestras etiquetadas como Normal es  %d\n'];
81  fprintf(mensaje,...
82      CountCancer,CountNormal)
83
84  fprintf('Existe un total de muestras mal clasificadas es   %d\n', ...
85      abs(countxd))
86
87  fprintf('El Error de Clasificacion es del (%d/700) --> %f%s  \n', ...
88      countxd/700,countxd/50,37)
89  toc
90  beep
```

APÉNDICE B

Publicaciones y Artículos

N°104
Mayo
2015

INFORMATIVO POLITÉCNICO

Publicación oficial de la Escuela Politécnica Nacional | Quito - Ecuador

El Observatorio Astronómico de Quito de la EPN

Conozca a los nuevos miembros
de la Academia de Ciencias

Nuevos algoritmos
para la detección temprana del cáncer

ÍNDICE

Producido por: Dirección de Relaciones Institucionales | EPN | Quito - Ecuador | www.epn.edu.ec

César Herrera, Pablo Posso, Diana Jaramillo, Paulina Fonseca, Esteban Allán, Valeria Hernández, Patricio Castro, Priscila Medina, Ivonne Platzer, Andrés Torres, Darío Rojas, Marcelo Castillo, Bryan Ortiz.
Diseño: Dirección de Relaciones Institucionales
Oficina: Av. Ladrón de Guevara E 11-253, edificio de Administración Central, Tercer Piso.

Teléfonos: (+593) 2 2976 300 ext. 1300, 1305.
Si deseas comentar y darnos tus sugerencias escríbenos al correo: info@epn.edu.ec

Nuevos Algoritmos para la Detección y Diagnóstico Temprano del Cáncer

Algoritmos basados en plataformas computacionales y matemática aplicada buscan nuevas pautas para la detección y diagnostico temprano del cáncer.

El cáncer es una enfermedad asintomática en una etapa temprana y muy difícil de diagnosticar en muchos de los casos, no es percibida hasta que ya ha alcanzado la etapa de metástasis difuminándose en otros órganos del organismo. El Instituto Nacional de Estadística y Censos(INEC) en una Infografia disponible en su sitio de internet resume de forma porcentual los datos de los Egresos Hospitalarios 2011, (Códigos CIE-10), donde en nuestro país de 54809 casos estudiados, las variantes de este padecimiento se manifiestan en afecciones al órganos respiratorios e intratorácicos, ojo, encéfalo y sistema nervioso central, glándula tiroides y otras glándulas endocrinas, Tejidos mesoteliales y blandos, labio, cavidad bucal y faringe, sitios mal definidos, secundarios y no especificados, un 32,9% en hombres y un 67,1% en mujeres. Este mal, es también, según datos del INEC, la segunda causa de muerte para los ecuatorianos, luego de las enfermedades hipertensivas y cerebrovasculares. Los logros que la medicina en su esfuerzo por reducir la muerte debida al cáncer han sido hasta la actualidad muy modestos, a pesar de que los avances conseguidos en los últimos tiempos han sido revolucionarios, existen aun casos en donde el cáncer es detectado en su etapa terminal. En donde aun no se ha encontrado ninguna metodología científica ni empírica que indique la presencia de esta patología.

Las metodologías tradicionales de diagnóstico del cáncer en cualquiera de sus tipos aciertan a un número relativamente bajo de casos en sus etapas tempranas, usando métodos invasivos y que corren el riesgo de ser detectados como falsos positivos o falsos negativos, ambos casos fatales para el paciente. Una detección temprana y adecuada de esta enfermedad, en cualquiera de sus tipo, abre las puertas y brinda esperanzas al paciente, a través de tratamientos de quimioterapias, radioterapias, terapias inmunológicas, terapias biológicas, entre otras.

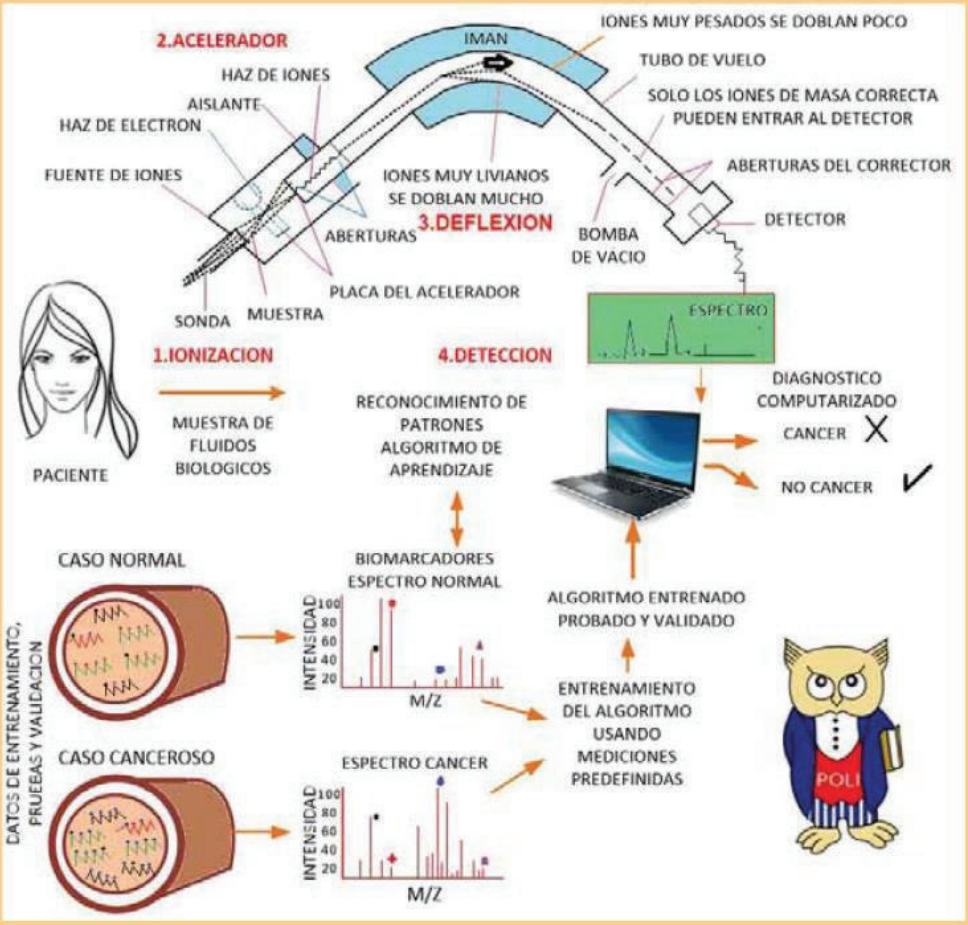
En este sentido centros de investigación y universidades han juntado esfuerzos para buscar alternativas de diagnóstico y tratamiento del cáncer usando métodos que permitan mejorar la eficacia del diagnóstico y detección de cáncer en sus primeras etapas. Dentro de estas alternativas están, los algoritmos para la detección del cáncer usando plataformas computacionales.

Los algoritmos para la detección del cáncer usando plataformas computacionales, están fundamentados en la extracción de información de conjuntos masivos de datos, sean estos matrices de números o imágenes. Como proyecto de titulación, Sofía Calle y Silvia Chasiluisa bajo la dirección de Roberto Herrera-Lara y Jorge Carvajal, han incursionado en este campo de investigación abriendo nuevas ideas sobre investigación de primer nivel en la Escuela Politécnica Nacional. Este proyecto busca definir nuevos algoritmos y metodologías de análisis de datos de mediciones de espectrometría de masas abordando el caso básico de reconocimiento entre pacientes enfermos y pacientes sanos.

El termino pacientes enfermos incluye las diferentes etapas del cáncer, desde sus primeras etapas hasta las etapas terminales. Todo este amplio espectro de casos de análisis es posible gracias a la versatilidad y precisión de la espectrometría de masas. El análisis de datos junto a técnicas quimiometricas permite a su vez la información numérica traducirla en información química y viceversa, siendo posible asi, la identificación de proteínas, antígenos y diferentes biomarcadores involucrados en el cáncer.

Este proyecto se encuentra actualmente en desarrollo en la Facultad de Ingeniería Eléctrica y Electrónica de la Escuela Politécnica Nacional con resultados prometedores. La filosofía del desarrollo de este algoritmo se basa en la economía de recursos computacionales versus sus resultados. Lo que se busca es crear una herramienta informática que pueda ser usada en un ordenador común, sin ninguna limitación.

Cuadro ilustrativo de los nuevos Algoritmos para la Detección y Diagnóstico Temprano del Cáncer



Los datos usados para la modelación y prueba del algoritmo proceden bases de datos en Internet del Centro para la Investigación del Cáncer del Instituto Nacional de la Salud en Estados Unidos. A estos datos se les aplica una serie de etapas de mejoramiento de la calidad de las mediciones como procesamiento (procesamiento digital y estadístico de señales) previo de análisis, luego se buscan de manera heurística (filtros estadísticos) las mejores características que definen cada grupo del conjunto de datos analizado, estas características son luego validadas(algoritmos de machine learning) de forma casuística midiendo en cada caso los resultados obtenidos. El rendimiento promedio de cada uno de estos casos mide de forma directa el rendimiento del algoritmo. La información resultante se presenta como zonas de información en la medición donde se encuentran los compuestos químicos causantes del cáncer. Trabajos futuros sobre los resultados de este proyecto buscan aplicar esta metodología al análisis en pacientes. IP

Autoras: Sofía Calle, Silvia Chasiluisa
Dirigido por: Roberto Herrera-Lara con colaboración de Jorge Carvajal

MEMORIAS

Tercer congreso ecuatoriano de Tecnologías de la información y Comunicación
TIC.EC 2015

2 - 4 de Diciembre 2015
Universidad Técnica Particular de Loja

Maskana

Número Especial ISSN No. 1390-6143

Universidad de Cuenca

Dirección de Investigación - DIUC

MASKANA • Número Especial • 2-4 diciembre 2015
Actas del Congreso TIC.EC
Indexada en Latindex

Revista semestral de Ciencias Humanas y Sociales, Biológicas y de la Salud, Exactas y Tecnologías de la Universidad de Cuenca (UC). Publicación internacional, bilingüe, revista electrónica con acceso abierto (http://diuc.ucuenca.edu.ec/index.php/revista-maskana). En este sitio web se puede descargar la guía para autores (en español o inglés). Las ideas y opiniones expresadas en las colaboraciones, son de exclusiva responsabilidad de los autores y autoras.

Bajo el auspicio de
Ing. Fabián Carrasco C., Rector
Ing. Silvana Larriva G., Vicerrectora

Impresión: Ediloja

Contenidos

Actas del Congreso TIC.EC 2015

Revista semestral de la Dirección de Investigación de la Universidad de Cuenca (DIUC), Cuenca, Ecuador
ISSN: 1390-6143

Revista semestral de la Dirección de Investigación de la Universidad de Cuenca (DIUC), Cuenca, Ecuador
ISSN: 1390-6143

Revista semestral de la Dirección de Investigación de la Universidad de Cuenca (DIUC), Cuenca, Ecuador
ISSN: 1390-6143

Procesamiento de datos de espectrometría de masas: Algoritmos y metodologías

Sofía Calle, Silvia Chasiluisa, Jorge Carvajal, Roberto Herrera.
Facultad de Ingeniería Eléctrica y Electrónica, Escuela Politécnica Nacional, Quito, Ecuador.
Autores para correspondencia: jazmina.calle.jordan@gmail.com
Fecha de recepción: 28 de septiembre 2015 - Fecha de aceptación: 12 de octubre 2015.

RESUMEN

El cáncer es una enfermedad asintomática en una etapa temprana y muy difícil de diagnosticar. En muchos de los casos no es percibida hasta que ya alcanza la metástasis. Es la segunda causa de muerte en el Ecuador a pesar de que los avances conseguidos en los últimos tiempos han sido revolucionarios, existen casos en donde el cáncer es detectado en su etapa terminal y aún no se ha encontrado ninguna metodología científica ni empírica que indique la presencia de esta patología. Las metodologías tradicionales de diagnóstico en cualquiera de sus tipos aciertan a un número relativamente bajo de casos en sus etapas tempranas, usando métodos invasivos con el riesgo de ser falsos positivos o falsos negativos. Centros de investigación y universidades han juntado esfuerzos para buscar alternativas de diagnóstico y tratamiento del cáncer usando métodos que permitan mejorar la eficacia del diagnóstico. En este trabajo se aborda un análisis de las diferentes etapas implicadas en el procesamiento de datos de muestras de tejidos cancerosos y saludables usando espectrometría de masas, usando plataformas computacionales, aplicados al mejoramiento de la calidad de las mediciones para posteriores aplicaciones de definición de biomarcadores.

Palabras clave: Procesamiento de datos, espectrometría de masas, biomarcadores, plataformas computacionales, procesamiento digital de señales.

ABSTRACT

Cancer is an asymptomatic disease at an early stage and very difficult to diagnose in many cases it is not perceived until it has already reached the stage of metastasis, spreading in other organs of the body. It is the second cause of death in Ecuador despite progress made in recent years have been revolutionary, there are cases where the cancer is detected in its terminal stage and still has not found any scientific or empirical methodology to indicate the presence of this pathology. Traditional methods of diagnosing cancer in any of its types are relatively ineffective in early stages, using invasive methods and at risk of being detected as false positive or false negative. Research centers and universities have joined forces to seek alternative diagnosis and treatment of cancer using methods to improve the efficiency of diagnosis and detection of cancer in its early stages. This paper discusses a group of algorithms and methodologies for processing data sets of mass spectrometry measurements of cancer and normal analyzed samples using computing platforms aimed at improving the quality of measurements for biomarkers definition applications.

Keywords: Data processing, mass spectrometry, biomarkers, computing platforms, digital signal processing.

1. INTRODUCCIÓN

La Espectrometría de Masas (EM) es una técnica de adquisición de datos muy utilizada en investigaciones de enfermedades como el cáncer por ser capaz de extraer información y presenta una

gran facilidad de adaptación en plataformas computacionales, donde utilizando algoritmos de minería de datos y aprendizaje de máquina se están definiendo continuamente nuevas metodologías de diagnóstico del cáncer empleando biomarcadores (BM) como técnicas de detección temprana de cáncer.

La adquisición de datos se realiza a través de las técnicas de ionización como: *Electron Ionizations* (EI), *Fast Atom Bombardment* (FAB), *ElectroSpray Ionization* (ESI), *Matrix-assisted laser desorption/ionization* (MALDI) *y Surface-enhanced laser desorption/ionization* (SELDI) aplicada sobre una muestra de fluidos biológicos como saliva, orina o sangre y de esta forma se obtiene la información requerida (Kristjansdottir *et al*., 2013; Fishman, 1991; Cedazo-Minguez & Winblad, 2010). Esta técnica presenta sus mediciones en forma de señales discretas con una limitación al momento de procesar un gran volumen de datos representados en vectores y matrices con software computacional y análisis estadístico, razón por la cual se aplican algoritmos para eliminar datos redundantes y datos que carecen de información relevante. Además, los espectros obtenidos de la EM presentan varios problemas como heterogeneidad y tienen una mezcla de ruido de tipo eléctrico, químico, de procesamiento y ruido debido a la mala calibración de equipos por lo que es necesaria una etapa previa que elimine la mayor parte de este ruido. Las etapas a utilizar son: Remuestreo, Corrección de Línea de Base, Alineación, Normalización y Suavizado de Ruido (Kristjansdottir *et al.,* 2013; Ping, 2007).

Este documento se divide en varias secciones que ayuden al esclarecimiento de esta técnica de los cuales se mencionan así: la Sección 2 se habla de la importancia de la información de EM y como se definen los BM. Sección 3 la Adquisición de los Datos, conceptos básicos y necesarios como espectrómetro de masas y sus cuatro funciones internas principales. Sección 4 analiza las diferentes metodologías de procesamiento, objetivos, problemas y limitaciones. Sección 5 se mencionan las herramientas computaciones usadas para el procesamiento de datos. Al final se adjuntan las conclusiones de este trabajo.

2. INFORMACIÓN DE ESPECTROMETRÍA DE MASAS

La EM es una técnica analítica que permite medir de manera precisa el peso molecular de un compuesto, es un método muy versátil ya que permite identificar la estructura de varios tipos de compuestos. También se aplica a todo tipo de muestras de fluidos biológicos, volátiles, no volátiles, sólidos, líquidos o gaseosas. Los fluidos biológicos contienen proteínas que sirven para la identificación y búsqueda de BM. Los BM son cambios medibles provocados por sustancias ajenas al organismo, que indican el estado patológico o no patológico del ser humano. Para medir estos cambios se analizan los patrones de abundancias obtenidos de las mediciones de EM que definen proteínas y antígenos. Los antígenos son sustancias que produce el sistema inmunológico para la producción de anticuerpos contra virus, químicos o toxinas. Una aplicación importante del uso de BM es el diagnóstico, tratamiento y prevención de enfermedades de tipo cancerígeno en etapa temprana (Cedazo-Minguez & Winblad, 2010; van der Merwe *et al*., 2007; Martín Gómez & Ballesteros González, 2008; Diamandisi, 2004).

3. ADQUISICIÓN DE LOS DATOS

La EM produce información a partir de los iones generados de moléculas orgánicas en fase gaseosa. Estos iones producidos se separan de acuerdo a la relación masa/carga (*m/z*) y se contabiliza su intensidad (abundancia relativa) (Martín Gómez & Ballesteros González, 2008; Ping, 2007). A partir de los datos obtenidos se genera el espectro de masas, en el eje horizontal se representa la relación *m/z* [Th] (*Thomsons*) y en el eje vertical la abundancia relativa (Kristjansdottir *et al*., 2013; Ping, 2007).

El instrumento denominado espectrómetro integra en su funcionamiento el proceso de adquisición de datos en formato digital. Dicho proceso se detalla en la Fig. 1, empieza con la

introducción de la muestra, ionización, analizador de masas, detección de iones. Los datos adquiridos son valores numéricos que al graficarlos toman la forma de una señal discreta.

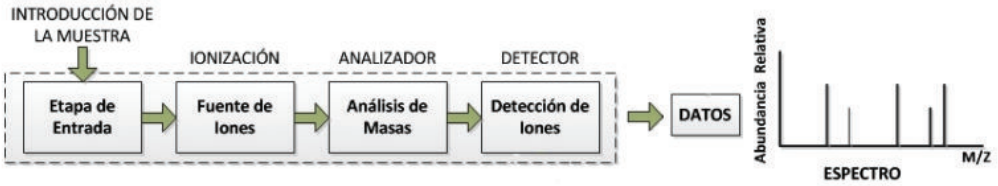


Figura 1. Esquema de un Espectrómetro de Masas**.**

3.1. Etapa de introducción de muestras

En esta etapa se toman pequeñas muestras como tejido, sangre o saliva para ser introducida en una cámara de volatilización en el vacío, donde con una fuente de calor se procede a cambiar el estado natural de la muestra (sólido o líquido) a estado gaseoso (Kristjansdottir *et al*., 2013; Fishman, 1991; Cedazo-Minguez & Winblad, 2010).

3.2. Etapa de ionización

La muestra en estado gaseosa es bombardeada con electrones, iones, moléculas o fotones, esto dependerá de la naturaleza de la muestra y el tipo de información que se desee obtener (Ping, 2007). Existen los siguientes métodos de ionización: Ionización en fase gaseosa e Ionización por desorción. En la ionización en fase gaseosa primero se volatiliza la muestra para luego ionizarla mientras que en la ionización por desorción la muestra se transforma directamente en iones. Se citan a continuación las técnicas por Desorción aplicadas en EM: Ionización por *electrospray* (ESI), Bombardeo con átomos rápidos (FAB) (van der Merwe *et al*., 2007; Martín Gómez & Ballesteros González, 2008; Gomis, 2008) y Ionización/Desorción por Láser (LDI).

3.3. Analizador de masas

Los iones atraviesan unos platos aceleradores que incrementan la energía cinética y pasan por un campo magnético que cambia la dirección de cada ion describiendo una curva que varía en función de su masa. Luego, dichos iones chocan sobre el detector de masas que contabiliza el número de colisiones en un punto específico. El número de colisiones se denomina abundancia relativa, de esta manera se obtiene el denominado espectro de masas con una resolución que varía en un rango de 10000 a 1000000 de puntos (van der Merwe *et al*., 2007; Gomis, 2008; Hilario *et al*., 2005). En la Fig. 2 se muestra un espectro de masas del conjunto de datos *Ovarian_Data_WCX2_CSV.zip*.

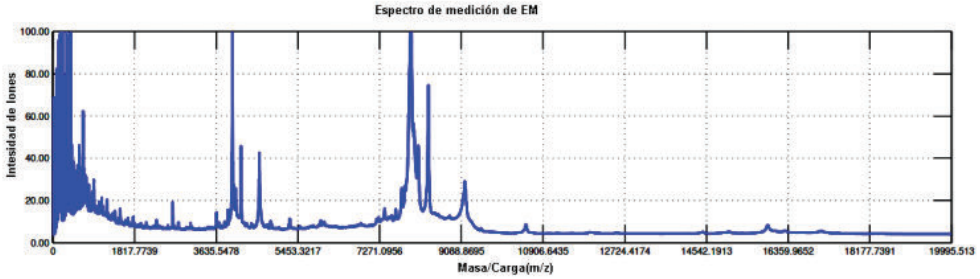


Figura 2. Ejemplo de un espectro de mediciones del Conjunto Ovarian Data WCX2 CSV.zip.

4. TIPOS DE METODOLOGÍAS

Durante el proceso de adquisición de los datos se introducen contaminantes debido a mala calibración de equipos, la preparación de la muestra, la inserción de la muestra en el instrumento y la saturación de iones. Estas variaciones y errores se traducen en ruido de diferentes características introducidos en los espectros, estos son: ruido térmico, eléctrico y químico. Además, los datos obtenidos tienen una heterogeneidad dimensional debido al tamaño de la muestra y la resolución de cada espectrómetro. Por ello una etapa previa al procesamiento de datos es extremadamente importante para extraer la señal de interés (Alterovitz & Ramoni, 2007). La metodología usada para procesar los datos son: Remuestreo, Corrección de Línea de Base, Alineación, Normalización y Suavizado de Ruido.

4.1. Remuestreo

Las aplicaciones prácticas con procesamiento digital de señales enfrentan el problema de cambiar la tasa de muestreo de la señal, ya sea aumentando o disminuyéndola, este paso se llama conversión de frecuencias de muestreo o remuestreo y en este caso se traduce como el aumento o disminución de la resolución de cada vector de *m/z*. El remuestreo es un proceso en el que se obtiene una nueva señal con valores controlados de *m/z*, esta nueva señal debe ser en lo posible similar a la original. Valores controlados significa que el número de puntos pueden ser mayores, iguales o menores al de la señal original. Esta etapa busca homogeneizar los vectores *m/z* para esto se aplica un factor *I/D*, donde I se denomina *Interpolation* que logra el incremento en la resolución de cada vector, y *D* se denomina *Decimation* este logra la disminución de la resolución de cada vector (Alterovitz & Ramoni, 2007). En la Fig. 3, ilustra el espectro de masas de un conjunto de datos Ovarian_Data_WCX2_CSV.zip remuestreado entre los valores de 2000 y 11000 (Ingle & Proakis, 2012).

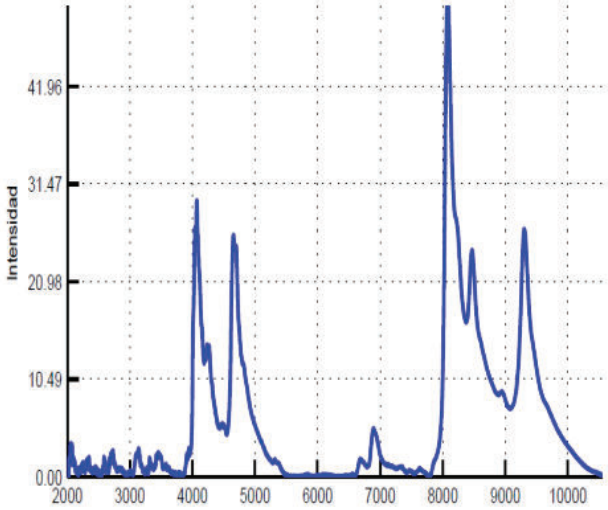


Figura 3. Remuestreo de los espectros.

4.2. Corrección de línea de base

Los datos de manera general muestran una línea de base variable consecuencia del ruido químico en la matriz o sobrecarga de iones que se origina en el detector de iones cuando este se satura. La línea de base es un desplazamiento de los iones en el eje vertical que eleva los valores de *m/z* bajos mientras que los valores altos de *m/z* no se ven tan afectados tal como se muestra en la Fig. 4. Para corregir este problema se estima la línea de base y se resta del espectro original es decir se halla el punto más bajo del espectro y se arrastra hasta cero en el eje vertical (Alterovitz & Ramoni, 2007; Antoniadis *et al.*, 2010).

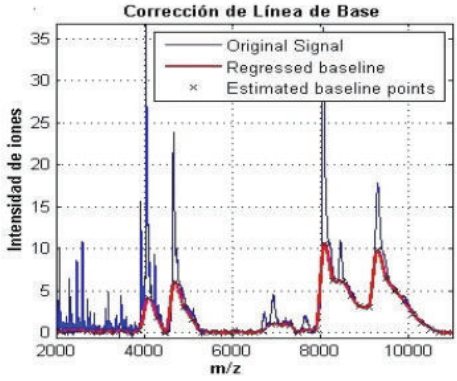


Figura 4. Estimación Línea de Base.

Se han desarrollado varias técnicas que ayudan a estimar la línea de base como: Filtros pasa altos implementados con transformada rápida de Fourier, teoría de *wavelets* y filtros digitales. Los métodos mencionados son poco usuales ya que estos distorsionan la señal y sería necesario modelar un filtro para cada espectro. El algoritmo más utilizado para estimar la línea de base en EM se basa en interpolación *spline* o suavizado (Alterovitz & Ramoni, 2007; Hilario *et al*., 2005, Eidhammer & Mikalsen, 2007).

Spline es una función polinómica a trozos de grado *p*, siendo la más práctica el polinomio de grado 3. Se definen el número y posición de los nodos, los nodos dividen al espectro en regiones en el eje de *m/z* (intervalos). El modelo matemático para un polinomio cúbico *Spline* se muestra en la ecuación 1.

$$s(x) = A_i(x - x_i)^3 + B_i(x - x_i)^2 + C_i(x - x_i) + D_i \quad para \quad i = 0,...,n-1 \tag{1}$$

Donde, A_i, B_i, C_i, D_i, ϵ $\mathbb{R}$, x $\epsilon(x,x_i)$ son las frontera de cada intervalo y n es el grado del polinomio. Se trata de estimar A_i, B_i, C_i, D_i, principalmente con condiciones de continuidad (s, s', s'') y condiciones de interpolación en los nodos. Se repite el paso anterior para cada intervalo de la señal. Y se construye la línea de base. Se resta la línea de base estimada del espectro original (Alterovitz & Ramoni, 2007; Gustafsson *et al*., 2011; Capelo-Martínez *et al*., 2015).

4.3. Alineación

La etapa de alineación de picos es de suma importancia debido a que existe una variación significativa entre las muestras de la intensidad y la ubicación de los picos en *m/z* por la mala calibración de los espectrómetros de masas. La idea es reemplazar los valores originales de *m/z* por valores calibrados o alineados, definiendo un vector de picos con los valores máximos de intensidades. Los picos no alineados se desplazan hacia las zonas donde hay más alineación, obteniendo así nuevos vectores de intensidades (Alterovitz & Ramoni, 2007; Bachmayer, 2007). En la Fig. 5 en a) Se muestra el mapa de calor de los datos con sus picos distorsionados sobre el eje *x* y en la parte b) Se observa el espectro con los picos alineados, estos de concentran en una línea vertical para valores de *m/z* de 4000, 8000 y 9000 aproximadamente.

4.4. Normalización

La normalización se realiza para que los diferentes espectros sean comparables entre sus intensidades relativas. Este método se utiliza para identificar y eliminar las variaciones aleatorias en la amplitud de cada intensidad causadas por la mala calibración de los instrumentos. En esta etapa busca reducir las diferencias de las intensidades para cambiar la escala. Se localiza el valor máximo de intensidad para

asignarle un valor y el resto de valores se ajusten proporcionalmente a este (Alterovitz & Ramoni, 2007). En la siguiente ecuación 2, se muestra el factor de intensidad normalizada.

$$I_{N\,m*n} = \frac{I_{i*n}}{I_{MAX\,i*n}} * N_f \tag{2}$$

Donde $I_{N\,m*n}$ es la matriz normalizada, I_{i*n} es cada intensidad a re-escalar, $I_{MAX\,i*n}$ es la intensidad máxima en cada espectro y N_f es el factor de normalización (Eidhammer & Mikalsen, 2007).

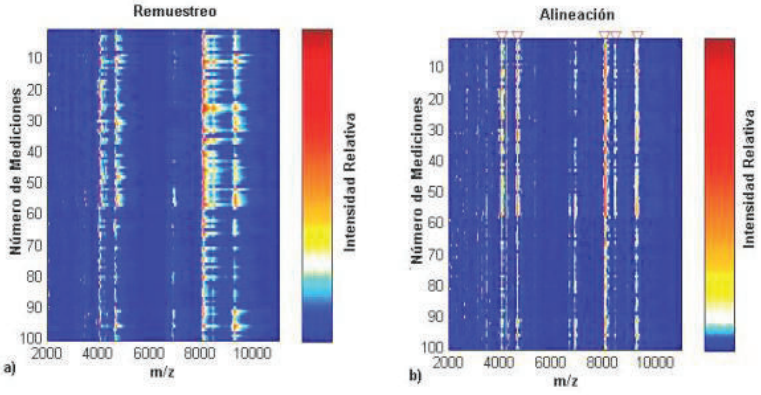


Figura 5. Alineación de picos del Conjunto Ovarian_Data_WCX2_CSV.zip.

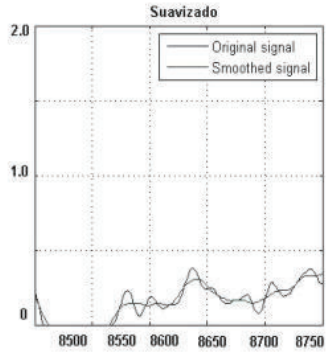


Figura 6. Suavizado de un espectro.

4.5. Suavizado de ruido

En esta etapa se busca reducir el ruido producido en etapas anteriores. Para ello se aplica técnicas de suavizado que produce una curva más suave del espectro reduciendo al máximo los picos falsos. Esta proceso se lleva a cabo utilizando el Filtro de *Savitzky y Golay*, consiste en suavizar muestra a muestra la señal basándose en una regresión polinomial (Alterovitz & Ramoni, 2007). Este tipo de filtro se adapta a la variación de frecuencia de muestreo y conserva la agudeza de los picos. El método de suavizado polinómico de mayor aceptación es el de *Savitzky-Golay* y *Káiser* que emplea un filtro digital de polinomio de mínimos cuadrados el cual conserva la mayor parte de las características de la señal así como la resolución entre picos de iones y la altura de los picos. Sin embargo este tipo de algoritmos requiere un análisis más exhaustivo de software (Alterovitz & Ramoni, 2007; Bachmayer,

2007). En la Fig. 6, se muestra el suavizamiento de picos, eliminando cambios bruscos entre cada pico para mayor apreciación se amplía la zona de interés para valores de *m/z* de 8500 a 8750. En la Fig. 7 se muestra el espectro de masas de un conjunto de datos *Ovarian_Data_WCX2_CSV.zip*, producto del procesamiento. En lo posible se ha reducido el ruido.

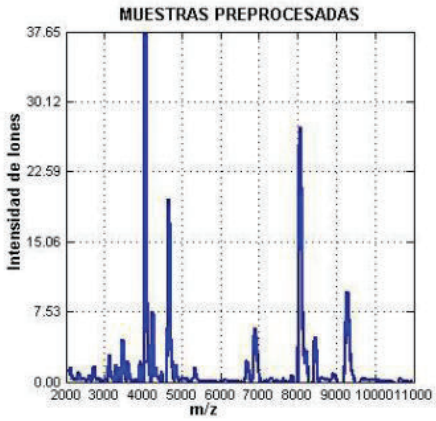


Figura 7. Espectro de masas después del procesamiento.

5. HERRAMIENTAS COMPUTACIONALES

5.1. Matlab bioinformatics toolbox

Esta poderosa herramienta ofrece varios algoritmos para el análisis de microarrayas, espectrometría de masas y la oncología de los genes. En el campo de la EM incluye el procesamiento con corrección de línea de base, suavizado, calibración y remuestreo. Adicional, proporciona funciones para la clasificación e identificación de biomarcadores potenciales en datos adquiridos a través de SELDI y MALDI. Además ofrece funciones como *heatmap* espaciales, navegadores de secuencias y *clustergrams* para visualizar los datos (Guide, 2003).

5.2. Weka3

Este es un software exclusivo de minería de datos desarrollado en JAVA. Integra en su plataforma un conjunto de algoritmos de aprendizaje automático como pre-procesamiento, clasificación de características, regresiones, \textit{clustering} y reglas de asociación. Las principales características de Weka son:

- Es un software libre publicado bajo licencia GNU con plataforma amigable para personas que no conocen a fondo la minería de datos (Cannataro *et al*., 2005; Markov & Russell, 2011).
- Para el pre-procesamiento de los datos es necesario definir el origen de los datos, weka 3 admite los siguientes formatos: .arff por defecto, csv (archivos separados por comas o tabuladores), c4.5 (conformado por el fichero .names y el fichero .data) (Cannataro *et al*., 2005; Markov & Russell, 2011).

5.3. Mass-Up

Es un software de *Open Source* para el análisis de datos obtenidos con la técnica de EM por MALDI desarrollada en java. Además de ser un software libre es capaz de cargar datos de MALDI desde diferentes formatos como mzML, mzXML y CSV. Esta herramienta posee 4 secciones para mejorar la

interacción con el usuario las cuales son: menú de carga, menú de preproceso, menú de análisis y principales tipos de datos.

En la sección del pre procesamiento destacan las opciones de método de suavizado por el método de *Savitzky Golay*, método de corrección de línea de base, detección de picos e intensidad mínima de pico. Adicional, en el sitio web de descarga del software se encuentran varios archivos de datos con los que se puede realizar varias pruebas para probar su funcionamiento (Capelo-Martínez *et al*., 2015).

6. CONCLUSIONES

- Las mediciones de espectrometría de masas poseen características que requieren algoritmos y metodologías de procesamiento específicas para mejorar los problemas de ruido, contaminantes, efecto de línea de base y diferencias dimensionales de las mediciones.
- El procesamiento digital de señales de mediciones de espectrometría de masas engloba una sinergia de técnicas estadísticas, algoritmos computacionales y conceptos de procesamiento digital de señales, lo que lo hace específico y dedicado para estas aplicaciones. Las características de estas mediciones difieren sustancialmente de las señales tradicionalmente analizadas en teoría de comunicaciones.

REFERENCIAS

Alterovitz, G., M.F. Ramoni, 2007. *Systems bioinformatics: An engineering case-based approach.* Boston-London: TRECH House, Inc.

Antoniadis, A., J. Bigot, S. Lambert-Lacroix, 2010. *Peaks detection and alignment for mass spectrometry data*. Disponible en http://membres-timc.imag.fr/Sophie.Lambert/papier/Spectrometry.pdf, 19 pp.

Bachmayer, S., 2007. *Preprocessing of mass spectrometry data in the field of proteomics*. Disiponible en http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.486.6689&rep=rep1&type=pdf, 17 pp.

Cannataro, M., P.H. Guzzi, T. Mazza, P. Veltri, 2005. *Preprocessing, management, and analysis of mass spectrometry proteomics data*. Università Magna Græcia di Catanzaro, Italy. Disponible en http://www.researchgate.net/profile/Tommaso_Mazza/publication/255586127_Preprocessing_Management_and_Analysis_of_Mass_Spectrometry_Proteomics_Data/links/0a85e5350cae8411ec000000.pdf, 5 pp.

Capelo-Martínez, J.L., F. Fdez-Riverola, D. Glez-Peña, A. Gutiérrez Jácome, H. López-Fernández, E. Lorenzo Iglesias, J.R. Méndez Reboredo, R. Pavón Rial, M. Reboiro-Jato, 2015. *Manual de mass up tEAM*. Disponible en http://sing.ei.uvigo.es/mass-up/manual.

Cedazo-Minguez, A., B. Winblad, 2010. Biomarkers for Alzheimer's disease and other forms of dementia. *Exp Gerontol*., 45(1), 5-14.

Cheng, Y., 2009. *Analysis of Seldi mass spectra for biomarker discovery and cancer classification*. PhD dissertation, CRUK Cancer Studies, Medical School, University of Birmingham. Disponible en http://etheses.bham.ac.uk/317/1/Cheng09Phd.pdf, 244 pp.

Diamandisi, E.P., 2004. Mass spectrometry as a diagnostic and a cancer biomarker discovery tool. *Mol. Cell Proteomics,* 3(4), 367-78.

Eidhammer, I., K. Flikka, L. Martens, S-O. Mikalsen, 2007. Computational methods for mass spectrometry proteomics. Wiley Online Library. Disponible en http://onlinelibrary.wiley.com/book/10.1002/9780470724309, 284 pp.

Fishman, D.A., 1991. *National ovarian cancer early detection program*. Mount Sinai School of Medicine. Disponible en https://www.mountsinai.org/static_files/MSMC/Files/Patient%20Care/OBGYN%20and%20Reproductive%20Services/NOCEDP%20Brochure.pdf, 20 pp.

Gomis, V.Y., 2008. *Espectrometría de masas*. Universidad de Alicante. Departamento de Ingeniería Química. Disponible en http://hdl.handle.net/10045/8249.

Guide, B.T.U., 2003. *Statistics Toolbox User's Guide*. The MathWorks Inc., Natick, MA, 816 pp.

Gustafsson, J.O.R., M.K. Oehler, A. Ruszkiewicz, S.R. McColl, P. Hoffmann, 2011. MALDI Imaging Mass Spectrometry (MALDI IMS) - Application of Spatial Proteomics for Ovarian Cancer Classification and Diagnosis. *Int. J. Mol. Sci*., 12(1), 773-794.

Hilario, M., A. Kalousis, C. Pellegrini, M. Müller, 2005.Processing and classification of protein mass spectra. *Mass Spectrom Rev.,* 25(3), 409-49.

Ingle, V. K., J.G. Proakis, 2012. Digital signal processing using MATLAB. CENGAGE Learning. BookWare Companion Series. Disponible en http://www.ece.iit.edu/~biitcomm/Yarmouk/Digital%20Signal%20Processing%20Using%20Matlab%20v4.0%20(John%20G%20Proakis).pdf, 816 pp.

Kristjansdottir, B., K. Levan, K. Partheen, E. Carlsohn, K. Sundfeldt, 2013. Potential tumor biomarkers identified in ovarian cyst fluid by quantitative proteomic analysis, iTRAQ. *Clin. Proteomics*, 10(1), 4.

Markov, Z., I. Russell, 2011. *An introduction to the WEKA data mining system*. Central Connecticut State University. Disponible en http://www.cs.ccsu.edu/~markov/weka-tutorial.pdf, 60 pp.

Martín Gómez, C., M. Ballesteros González, 2008. *Espectrometría de masas y análisis de biomarcadores*. Monografías de la Real Academia Nacional de Farmacía. Disponible en http://www.analesranf.com/index.php/mono/article/view/1066, 56 pp.

Ping, H., 2007. *Classification methods and applications to mass spectral data*. Hong Kong Baptist University. Restricted Access Theses and Dissertations. Paper 593.

van der Merwe, D.E., K. Oikonomopoulou, J. Marschall, E.P. Diamandis, 2007. Mass spectrometry: uncovering the cancer proteome for diagnostics. *Adv. Cancer Res*., 96, 23-50.

A Novel Methodology for the identification of biomarkers using statistical filters with validation through Adaboost

Sofia Calle, Silvia Chasiluisa, Roberto Herrera-Lara, and Jorge Carvajal
National Polytechnic School - Ecuador

***Abstract*—Today, the application of mathematical and computational methods have allowed investigations related to revolutionize medicine, chemometrics, proteomics and genomics. Mathematical algorithms of data mining and machine learning tests are opening up new hopes in the fight against chronic diseases such as cancer, diabetes, Alzheimer's disease, cirrhosis, cardiovascular disease and adrenal diseases. This paper presents a new method of selection and validation of biomarker patterns applied to early treatment of chronic diseases. The methodology proposed in this paper deals with the analysis of data from instruments directly medici'on until the de fi nition of biomarcacion patterns. This operation is based on the combination of t-test and Mann-Whitney U test in a filter statistic, which defines areas of interest in the spectra of measurements, then these areas of interest are grouped and reduced to nearby features than average of each of these, thus eliminating redundant information. The contribution of this work lies in the structure of the statistical fi lter, which has an enormous capacity for extraction of information through simple calculations, compared to complex algorithms presented in similar jobs. Finally selected this fi lter characteristics are validated using Adaboost classi fi ers, LPBoost TotalBoost and cross-validated and tested with external test samples. The results reflect an efficiency greater than 95%, plus robustness against Overfitting and Underfitting.**

***Index Terms*—IEEEtran, journal, LaTeX, paper, template.**

I. Introduction

Mass spectrometry (MS) is a very popular technique data acquisition in research on chronic diseases such as cancer, adrenal diseases, diabetes, cardiovascular disease, cirrhosis, and Alzheimer's. Its popularity is based on the high capacity that has this technique to extract information, and the easiness to be integrated into computational methodologies for analysis of massive data sets. Computational methodologies used in these medical applications have received tremendous interest from researchers because through them you can make analysis and experimentation in a much more comprehensive than traditional methods. The combination of statistics, probability and computational simulation can address a number of cases much more varied than traditional analysis laboratory experimentation. This technique called mass spectra measurements, these spectra, for applications analyzed in this paper come from the analysis of samples of biological fluid (FB) as saliva or blood serum are performed. These FB have a huge amount of information on the presence of pathologies in the human body. This information is represented by biomarkers (BM), which are substances used in the analysis and diagnosis of medical conditions. These substances have the ability to indicate the presence of a disease state, as well as the response to chemical treatments. The acquired mass spectra consist of a vector of intensity values, where the abundance of ions in the gas phase of the sample analyzed and a vector of mass relationships expressed radius m / z expressed in thomsons [Th]. Figure 1 shows various measurements of a data set $OvarianCDPostQAQC.zip$, available from the National Cancer Institute of the United States shown. The process of analysis of EM measurement data for identification of BM consists basically of data processing, selection and validation of patterns with a high degree of intergroup discrimination. Once validated these patterns, the next step is to translate them into chemical expressions that can be used as diagnostic tools and tuning peptides and antigens for the treatment of the diseases mentioned above. This paper presents a new method for the identification of patterns of BM through the combination of two statistical tests t-test and Mann-Whitney U test in a filter statistical and iterative removal characteristics of the areas of interest is presented using for grouping the middle of each zone reducing redundant information. These results are validated using three separate classification AdaBoost ers, TotalBoost and LPBoost. These classifiers are especially robust against the effects of overtraining infraentrenamiento and, besides being easily adaptable to such applications. With these characteristics it avoids falling into the misinterpretation of false results and false positives or negatives. Performance measurement is done through cross-validation and external tests using independent samples that have not previously participated in the modeling of the classifiers. This methodology shows promising results with an efficiency greater than 95% correct in the tests carried out. This work is limited to the identification of patterns of biomarkers, leaving for later developments and their translation into protein chemical compounds. In the following sections algorithmically the proposed methodology described in this paper, the tests using two data sets, the first OvarianCD PostQAQC.zip Clinical Proteomics Program of the Center for Research on Cancer belonging to the National Cancer Institute the United States and the second Arcene UCI Machine Learning Repository. Analyzes and compares the results with previous work. In the last section raises possible extensions to the research presented and the final part of the section is attached conclusions.

II. DEFINITION OF THE METHODOLOGY PROPOSAL

The methodology proposed in this work part of the base case data analysis for medical diagnostic applications, a set D consists of the $G1$ and $G2$, where $G1$ represents a normal biological condition and $G2$ a biological pathological state. Assembly D patterns to differentiate between $G1$ and $G2$ so reliable, these patterns validate and once validated may be used as biomarkers for tuning proteins that can be used in the treatment of diseases will be extracted. This work is limited to the de fi nition of patterns, synthesizing proteins allowing for future work.

A. Data processing

Measurements of groups $G1, G2$ of the set D are expressed as sets of measurements matrices form

$$M = \{((m/z)_1, i_1), ((m/z)_2, i_2), ((m/z)_3, i_3), \ldots, ((m/z)_n, i_n)\} \quad (1)$$

where values $((m/z)_k, i_k)\ \epsilon\ \mathbb{R}\ y\ 1 \leq k \leq n$.

Due to the heterogeneity of the lengths of the M measurements, the first thing to do is to process this data. The processing of the data set is based on D and consists of the following stages: resampling, baseline correction, alignment measurements, standardization and filter [**?**], [**?**], [**?**]. Resampling is based on the concept from the signal processing, where given a discrete time signal, applying this concept can reduce or increase its sampling frequency, in this case, it is understood to increase or decrease the resolution of the measurement and therefore the elements of each vector of measurements. For data processing is assumed that the measurements in the data set D are defined by M_m, a matrix formed by two vectors $I_m, (M/Z)_m$, two discrete signals in time defined as $I_m = \{i_1, i_2, i_3, \ldots, i_n\}$ y $(M/Z)_m = \{((m/z)_1), ((m/z)_2), ((m/z)_3), \ldots, ((m/z)_n)\}$ where the index n indicates the resolution of the measurement m is the number of the measurement. Generally, in these measurements that the resolution of the measurement $m-1 \neq m \neq m+1$ is met. Resampling aims these resolutions to a common value of n_h homogenization by fixing the dimensionality of the data set D in $m \times n_h$.

The algorithm used is based on the combination of an interpolator signals, a filter and a decimating low pass. The interpolator increases the resolution of the measurements of a factor $n_1 > n$ after this low-pass the filter attenuates the effects of aliasing and imaging produced in step interpolation finally the decimator reduces the resolution of a factor $n_2 < n_1$. The combination of these two operations allows to control the homogenization $\frac{n_1}{n_2}$rational value as needed. t this stage the factor $\frac{n_1}{n_2}$ defines a n_h.In certain applications it is necessary to reduce radically the resolution 10-1 factors, however, in other applications, it is necessary to increase the resolution in certain segments of the spectrum for more specific analysis.

After processing is necessary to remove a typical noise in the data called effect of the baseline. This anomaly occurs due to contaminants present in the analyzed sample and is present in all measurements in the initial segment of the measurement. The algorithm used in this step estimates a minimum frequency of baseline level using the intensity and frequency of each noise measurement. An estimated this frequency baseline by regression of these values once a vector of displacement values of baseline of each of the intensities of the measurement processed, which is finally subtracted from the original intensity of the measurement is obtained obtaining a new vector $I^{bc}_{m\times n}$ with the effect of baseline and corrected(bc) without changing the resolution.

With measurements and homogenized and removed from each of these the effect of baseline, the next step is to perform the alignment of measures. The measurement phase alignment aims to correct errors of calibration of measuring instruments in the axis M/Z.For this fixing alignment reference peaks $p = \{p_1, p_2, ..., p_k\}$, where k takes values for practical applications $3 \leq k \leq 5 \forall k \in Z^+$. Basically at this stage new vectors of intensities $I^{alig}_{m\times n}$reference to the peaks of greater intensity measurements are reconstructed. This reconstruction is based on use of temporary deformation functions, whereby peaks peaks unaligned reference to adapting its position in the M/Z axis move..

The normalization step complements the alignment measurements in correcting errors calibration of instruments, but is instead working on the axis of the intensities of abundance. This stage aims at reducing differences in the intensities of the measurements set D with respect to a normalization factor. The process is to identify the maximum intensities of each of the measurements, to which it is assigned a value of normalization$norm_M$. hen, all remaining currents measurements with respect to these maximum intensities normalized value $norm_M$ getting normalized intensity factor $I^{norm}_{i\times n} = \frac{I^{alig}_{i\times n}}{max(I^{alig}_{i\times n})} \times norm_M$. The choice of the normalization factor depends on the nature of the sample, its value is essential for a correct translation of biomarkers into protein.

The final stage of processing of measurements is the filter noise measurements. While an earlier stage performed partly an filter noise in the measurements, all previous steps to noise filter they tend to produce errors that occur as a new presence of noise in the measurements. From a practical standpoint, this filter process aims to smooth the curve of the spectrum by eliminating as many variations of random character of measurements. This step is carried out using the filter *Savitzky-Golay*, which performs a regression polynomial of degree with at α least $\alpha+1$ equidistant points to determine the value of each new point. The result of the application of this filter EM measurements are the same measurements but smoothed, retaining its dimensionality. The approximation calculated through this fi lter tends to preserve the original data distributions fi ltrates, which means that the maxima and minima relative or spikes are not altered.

The entire process of processing the measurement matrices is presented in Figure 2.

B. Discriminant Feature Selection

Once the data set D available, the next step to make is between all measurements, find a subset of values $D^{red} =$

$\{(m/z)_k, i_k\}$, $1 < k < n$ here the intensities and relationships mass radius k-esimas must be statistically cant based on criteria that prove how high is the degree of intergroup discrimination that these possess.

For this stage we have designed a statistical filter combining *t-Student* and *U-Mann-Whitney*. The criteria of these tests is to be complementary, as the first assumes that the analyzed groups have a Gaussian probability distribution, while the *Mann-Whitney* assumes that the probability distribution of the analyzed groups is the same, but not It imposes the condition that this is Gaussian[**?**]. The process is performed by subjecting fi lter data analysis groups $G1$ and $G2$ of set D separately for each of these statistical tests. The application of these tests p values that define the probability of variation in the $i_k - esimos$ values of intensities of the M_m , measurements are obtained p-values whose probability tends to zero indicate variations in the values of intensity lies a high power of discrimination Intergroup. It is very difficult to define how many of these values are needed to define the patterns of biomarkers, however it is possible to estimate how many of these values decrese the maximum error of intergroup discrimination based on the use of algorithms classified supervised cation. Algorithms supervised classi fi cation have been widely used in applications relating to the analysis of mass spectrometry data.[**?**], [**?**], [**?**], [**?**]. However algorithms based on Bayes theory, support vector machine, k nearest neighbors, and even neural networks require additional processing to change the original characteristics of group D to those required as input parameters of these algorithms. This additional processing limits the amount of information that can be extracted from the sets, as groups should be statistically analyzed fi cient matrix of the data set must be square, not negative, and reversed. These limitations are solved through algorithms based on *Boosting-Learning* on binary decision trees.

In each test two vectors independent of p-values, p_1 y p_2 , of dimension n. Then, with each of these vectors are estimated probability functions over which is fixed an approximate value η pairs of $\{(m/z)_k, i_k\}$, whose value p tends to zero. LThen for p-values of the two statistical tests two binary classification ers are modeled using algorithm *AdaboostM.1* based on the analysis groups $G1$ and $G2$, where randomly obtained measurements that were part of the set of training and testing of classified modelinger. In each of the tests, once the structure of the classifier established iteratively be changing the dimension of joint training and testing from 1 to η, to find a point of inflection in ζ pares de $\{(m/z)_k, i_k\}$ dwhere classi fi cation error is minimized by the presence of infraentrenamiento or overtraining in classifiers. The resolution step depends iteration of the available computing power, however, it can be from a certain resolution and gradually increase, the point of inflection error classi fi cation does not change, this way you can get an accurate number of pairs $\{(m/z)_k, i_k\}$ ith a high degree of intergroup discrimination. Here comes a new subset, fixed by the inflection point performance of classifiers. In this new subset of features that cause errors of classi fi cation, usually caused by noise problems that could not be fi lter desecharan in previous stages. At this point they have been defined areas biomarcacion where pairs $\{(m/z)_k, i_k\}$ tend to be redundant, to eliminate this redundancy in addition to fi lter a grouping of data was conducted based on the average of the indices $\{(m/z)_k\}$ defined by the point of inflection in pairs ζ. The grouping is done looking for a number of groups based on the averages of each of the areas marked above.n the last part the two closest to the average of the areas marked discarding the other indexes are taken. Thus it is eliminating redundant information and modeling classi fi ers of a number of pairs $\{(m/z)_k$ of a high degree of intergroup discrimination is strengthened. Finally the indices of the pairs $\{(m/z)_k, i_k\}$ of each of the statistical tests are combined, thereby obtaining the reduced data set $D^{red} = \{I_{m\times k}, M/Z\}$. This procedure is shown in Figures 3 and 4.

C. Discriminant Validations Characteristics

Validation of the patterns detected in the previous section are made using three separate classifiers, *Adaboost*, *Total-Boost* and *LPBoost*,where in each of these errors classification using cross-validation and external further testing samples will be measured. This stage is basically the modeling of classifiers using the D^{red}, measure their performance, eliminate predictors deficient, then re-evaluate the performance of classifiers. The modeling of the classifiers and the elimination of predictors deficient improve the performance of the classification modeleders. The final performance is compared among the three classifiers to measure the effectiveness of pairs $\{(m/z)_k, i_k\}$ selected in the previous section.

Adaboost is defined basically as a methodology learning whereby you take an algorithm to classify simple cation and iteratively applies a certain number of times in sequence, where each iteration error classification is improved, achieving superior returns the application of complex algorithms for classification. The algorithm used in this step is the variant used *Adaboost.M1* data sets in two groups. In this paper, this algorithm is applied according to the m available, for which, first disconnect the entire set D^{red} in 3 subsets randomly, training (m_T) cross-validation tests (m_{VC}) and external tests (m_{VE}). Once this separation, assuming the (m_T) measurements has $\{I_1, I_2, ..., I_{m_T}\}$ strength vectors associated with a predictor p, said intensities are labeled with a vector $y = \{+1, -1\}$, where +1 tag group $G1$ measurements and measurements -1 group $G2$. El clasifier $h(m_T)$ tag group G1 measurements and measurements -1 group G2. $h(m_T)$ will be defined $\epsilon = \frac{1}{m_T}\sum_{i=1}^{m_T}\Gamma(y_i \neq h(m_T))$. Yhe function $\Gamma(y_i \neq h(m_T))$ is 1 if success in the classification and it was 0 on error.

This process is repeated β times, where the final classifier $\mathbf{H}$ will be the combination of all classifiers $h((m_T)_\beta)$ as a function of a vector of weighting $\mathbf{B}_\beta$. The classifier final would be defined by $H(h(m_T)) = (\sum_{i=1}^{\beta}\mathbf{B}_i h(m_T)_i)$, in this case the function sign defines whether the element classi fi ed belongs to $G1$ or $G2$ by mapping $y = \{+1, -1\}$[**?**], [**?**]. *TotalBoost* y *LPBoost* are two variants of *Adaboost*, which do not require the parameter $\beta$for training as they seek an optimal solution and limit the number of interactions in training automatically. These algorithms are ideal for limited

sets of data, making them useful in applications studied in this paper, which is very di ffi cult to have huge amounts of EM measurements. *LPBoost* compared to *Adaboost* and *Totalboost* for training as they seek an optimal solution and limit the number of interactions in training automatically. These algorithms are ideal for limited sets of data, making them useful in applications studied in this paper, which is very di ffi cult to have huge amounts of EM measurements [**?**]

A common tendency to decrease the error classification in the three classifiers, indicates that the chosen data have a high degree of intergroup discrimination, otherwise any of these tests will show sufficient overfiftting and underfiftting in behavior the classifiers. Also showing the effects of underfiftting and overfiftting, the combination of these two benchmarks limiting find false positives or false negatives in samples analyzed, resulting tragic factor in this type of application. In Figure 5, the application of these algorithms is presented in the methodology developed in this paper.

III. TESTING METHODOLOGY PROPOSAL

Testing the methodology described in this paper were performed in Matlab using the *Bioinformatics Toolbox, Statistics Toolbox* and *Optimization Toolbox*, Toolbox in a hardware platform with Windows 7, processing speed of 2.4 [GHz] and 12 [GB] of RAM

The results are shown below in the test dividos cross-validation and external testing samples respectively. In each of the error curves shown classi fi cation, you can clearly see that the results show promising results on the methodology, whose performance is above 95% in cross-validated simulations.

IV. RESULTS

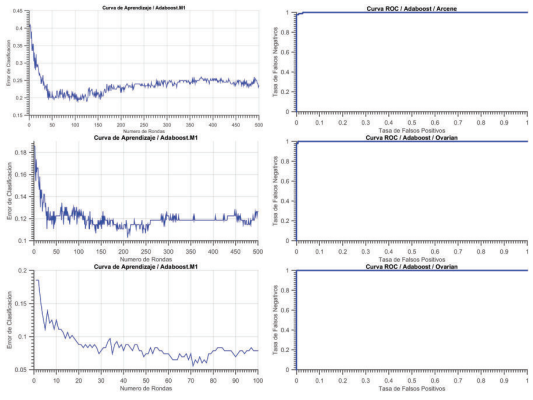


Fig. 1. Learning and ROC Curves of Arcene, Ovarian27-15 and Ovarian-QAQC Datasets

V. FUTURE WORK

Because of the versatility of MS as data acquisition technique, as well as the metodolgia proposed in this paper, the following lines of research based on this article arise:

1) Complementing the methodology described in this paper with the translation of biomarkers into proteins and antigens.
2) analysis of the chemical composition of metal dicinales plants used in the treatment of chronic diseases [**?**], [**?**]. MS allows for very precise information about the chemical composition of the samples analyzed, based on libraries of chemical compounds available on the Internet can be applied the methodology proposed in this paper attempted exhaustive searches for these compounds in endemic medicinal plants of Ecuador [**?**].
3) Implement the methodology used in supercomputing platforms where they can control the dimension of the dataset analyzed, can raise the resolution of these measurements without sacrificing the processing time, memory, and other computer resources.

VI. CONCLUSION

- The methodology proposed in this paper is simple and requires minimal supervision. No need to arbitrarily define a number of possible characteristics of the analyzed together, since the methodology defined for itself the number of possible characteristic patterns of biomarkers.
- The combination of the *t-test* and *Mann-Whitney U test* in the proposed in this article statistical filter is powerful when it comes to extracting information. Both statistical tests define a smaller number of areas of interest as compared to when combined. The number of areas of interest defined by the combination of these techniques provides a greater amount of information on possible patterns of biomarkers, which have greater resistance to the phenomena of infraentrenamiento and overtraining in modeling data classification systems.
- The simplicity of the methodology presented in this paper has qualities to be optimized and implemented in graphics processor cards based platforms. The numerical calculations are basically massive operations on matrices.

REFERENCES

[1] BAOLIN Wu,ABBOTT Tom , FISHMAN David , McMURRAY Walter, MOR Gil,STONE Kathryn, WARD David, WILLIAMS Kenneth y ZHAO Hongyu, Comparison of statistical methods for classification of ovarian cancer using mass spectrometry data, Bioinformatics Journal, Print ISSN 1367-4803.Online ISSN 1460-2059.

[2] WRIGHT Michael, HAN David, AEBERSOLDRuedi, Mass spectrometry-based expression profiling of clinical prostate cancer, Molecular & Cellular Proteomics Journal, Print ISSN 1535-9476, Online ISSN 1535-9484.

[3] PAULO A. Joao, KADIYALA Vivek, BANKS A. Peter, CONWELL L. Darwinl, STTEN Hanno, Mass Spectrometry-based Quantitative Proteomic Profiling of Human Pancreatic and Hepatic Stellate Cell Lines, Genomics, Proteomics & Bioinformatics Journal, ISSN: 1672-0229.

[4] CHO William C. S.,YIP Timothy T. C.,YIP Christine ,YIP Victor ,THULASIRAMAN Vanitha, NGAN Roger K. C., YIP Tai-Tung, LAU Wai-Hon,AU Joseph S. K., LAW Stephen C. K., CHENG Wai-Wai , MA Victor W. S., y LIM Cadmon K. P., Identification of Serum Amyloid A Protein As a Potentially Useful Biomarker to Monitor Relapse of Nasopharyngeal Cancer by Serum Proteomic Profiling, Clinical Cancer Research (CCR) Journal, Print ISSN: 1078-0432; Online ISSN: 1557-3265.

[5] ZHANG Z, BAST RC Jr, YU Y, LI J, SOKOLL LJ, RAI AJ, ROSENZWEIG JM, CAMERON B, WANG YY, MENG XY, BERCHUCK A, VAN Haaften-Day C, HACKER NF, HW Bruijn DE, VAN der Zee AG, IJ Jacobs , ET Fung,DW Chan , Three biomarkers identified from serum proteomic analysis for the detection of early stage ovarian cancer, Cancer Research (CanRes) Journal, Print ISSN: 0008-5472; Online ISSN: 1538-7445.

[6] Dr. PETRICOIN Emanuel F. PhD., ARDEKANI Ali M. PhD., HITT Ben A. PhD., LEVIANE Peter J. , FUSARO Vincent A., STEINBERG Seth M. PhD., MILLS Gordon B. MD., SIMONE Charles MD., FISHMAN David A. MD., KOHN Elise C. MD., LIOTTA Lance A. MD., Use of proteomic patterns in serum to identify ovarian cancer, The Lancet Journal (Vol. 359, Issue 9306, Pages 572-577) , ISSN: 0140-6736.

[7] WADSWORTH J. Trad , MD.;SOMERS Kenneth D. PhD.; BRENDAN C. Stack, Jr, MD.;CAZARES Lisa, BS.; GUNJAN Malik, PhD.; BAO Ling Adam, PhD.; WRIGHT George L. Jr, PhD.; O. John Semmes, PhD., Identification of Patients With Head and Neck Cancer Using Serum Protein Profiles, JAMA OtolaryngologyHead & Neck Surgery Journal, Print: ISSN 2168-6181, Online: ISSN 2168-619X.

[8] O. J. Semmes, L. H. Cazares, M. D. Ward, L. Qi, M. Moody, E. Maloney, J. Morris, M. W. Trosset, M. Hisada, S. Gygi y S. Jacobson, Discrete serum protein signatures discriminate between human retrovirus-associated hematologic and neurologic disease, Leukemia Journal, ISSN: 0887-6924, EISSN: 1476-5551.

[9] FERRARI Lorenza , SERAGLIA Roberta , ROSSI Carlo Riccardo , BERTAZZO Antonella ,LISE Mario , ALLEGRI Graziella y TRALDI Pietro ,Protein profiles in sera of patients with malignant cutaneous melanoma Rapid Communications in Mass Spectrometry Journal, ISSN: 1097-0231.

[10] WOODING Kerry M. y AUCHUS Richard J., Mass spectrometry theory and application to adrenal diseases, Molecular and Cellular Endocrinology Journal, ISSN: 0303-7207.

[11] McDONALD Jeffrey G., MATTHEW Susan ,AUCHUS Richard J., Steroid profiling by gas chromatography-mass spectrometry and high performance liquid chromatography-mass spectrometry for adrenal diseases, Hormones and Cancer Journal, ISSN: 1868-8497 (print version) ISSN: 1868-8500 (electronic version).

[12] LAPOLLA Annunziata, MOLIN Laura , and TRALDI Pietroi, Protein Glycation in Diabetes as Determined by Mass Spectrometry, International Journal of Endocrinology, ISSN: 1687-8337.

[13] LAPOLLA Annunziata,FEDELE1 D. y TRALDI Pietroi, Diabetes and mass spectrometry, Diabetes/Metabolism Research and Reviews Journal, ISSN: 1520-7560.

[14] LI Xiang , LUO Xiangxia , LU Xin, DUAN Junguo y XU Guowang , Metabolomics study of diabetic retinopathy using gas chromatography-mass spectrometry: a comparison of stages and subtypes diagnosed by Western and Chinese medicine, Molecular BioSystems Journal, ISSN: 1742-206X (print).

[15] FERNANDEZ Llama P., Aportaciones de la proteómica al estudio de las enfermedades cardiovasculares, Revista Hipertensión y Riesgo Vascular, ISSN: 1889-1837.

[16] CAO Yuan, HE Kun , CHENG Ming , SI Hai-Yani,ZHANG He-Lin, SONG Wei , LI Ai-Ling, HU Cheng-Jin , y WANG Na, Two Classifiers Based on Serum Peptide Pattern for Prediction of HBV-Induced Liver Cirrhosis Using MALDI-TOF MS, BioMed Research International Journal, ISSN: 2314-6133.

[17] A. K. Batta, R. Arora, G. Salen, G. S. Tint, D. Eskreis y S. Katz, Characterization of serum and urinary bile acids in patients with primary biliary cirrhosis by gas-liquid chromatography-mass spectrometry: effect of ursodeoxycholic acid treatment, Journal of Lipid Research, ISSN 0022-2275.

[18] MUSUNURI Sravanii , WETTERHALL Magnusl ,INGELSSON Martin , LANN-FELT Lars , ARTEMENKO Konstantin ,BERGQUIST Jonas , Kúltima Kim , and SHEVCHENKO Ganna, Quantification of the Brain Proteome in Alzheimers Disease Using Multiplexed Mass Spectrometry, Journal of Proteome Research, ISSN: 1535-3893.

[19] MATTHIESEN Rune and MUTENDA Kudzai E., Introduction to Proteomics, pp. 1-37, Mass spectrometry data analysis in proteomics / edited by Rune Matthiesen, ISBN-13: 978-1-58829-563-7.

[20] FUSHIKI Tadayoshii, FUJISAWA Hironori y EGUCHI Shinto, Identification of biomarkers from mass spectrometry data using a common peak approach, BMC Bioinformatics Journal, ISSN 1471-2105.

[21] PHAM P., A Novel Algorithm for Multi-class Cancer Diagnosis on MALDI-TOF Mass Spectra, Bioinformatics and Biomedicin IEEE Journal, pages 398-401, ISBN 978-1-4577- 1799-4, 12-15 Nov. 2011.

[22] JELONEK Karol , ROS Malgorzata ,PIETROWSKA Monika, WIDLAK Piotr, Cancer biomarkers and mass spectrometry-based analyses of phospholipids in body fluids, Clinical Lipidology Journal, ISSN 1746-0875, pages 137-150, 2013/2.

[23] PIETROSWSKA M., JELONEK K., MICHALAK M., ROS M.,RODZIEWICZ P.,CHMIELEWSKA K ,POLAMSKI K ,POLANSKA J,KLOSOK A Gdowicz,GIGLOK M,SUWINSKI R,TARNAWSKI R , DZIADZIUSZKO R, RZYMAN W ,WIDLAK P, Identification of serum proteome components associated with progression of non-small cell lung cancer, Acta biochimica Polonica Journal, 2014/5.

[24] G. A. GOWDA Nagana , ZHANG Shucha,GU Haiwei , ASIAGO Vincent , SHANAIAH Narasimhamurthy, y RAFTERY Daniel, Metabolomics-Based Methods for Early Disease Diagnostics - A Review, Expert Review of Molecular Diagnostics Journal, Sep 2008; 8(5): 617633, ISSN 1473-7159.

[25] TARAWNEH Sandra K. Al,BORDER Michael B.,DIBBLE Christopher F., y BEN-CHARIT Sompop,Defining Salivary Biomarkers Using Mass Spectrometry-Based Proteomics - A systematic review, OMICS A Journal of Integrative Biology, ISSN: 1536-2310.

[26] Dr. LEE Yu Hsiang, Phd. y Dr.WONG David T., DMD., DMSC. Saliva - An emerging biofluid for early detection of diseases, Am J Dent 2009;22:241-8.

[27] KHADIR Abdelkrim and TISS Ali, Proteomics Approaches towards Early Detection and Diagnosis of Cancer, Carcinogenesis & Mutagenesis Journal, ISSN: 2157-2518.

[28] GIL Alterovitz, RAMONI Marco F., Systems Bioinformatics: An Engineering Case-based Approach, cap. 4, Editorial: Artech House; Edición: Har/Cdr (1 de marzo de 2007), ISBN-10: 1857431820.

[29] EIDHAMMER Ingvar, FLIKKA Kristian, MARTENS Lennart, MIKALSEN Svein-Ole, Computational Methods for Mass Spectrometry Proteomics , Wiley & Sons Publications, January 2008, ISBN: 978-0-470-51297-5.

[30] EIDHAMMER Ingvar, BARSNES Harald, EGIL EIDE Geir, MARTENS Lennart, Computational and Statistical Methods for Protein Quantification by Mass Spectrometry, Wiley & Sons Publications, February 2013, ISBN: 978-1-119-96400-1.

[31] TESSITORE Alessandra, GAGGIANO Agata, CICCIARELLI Germanai, VERZELLA Daniela, CAPECE Daria, FISCHIETTI Mariafausta, ZAZZERONI Francesca, y ALESSE Edoardo, Serum Biomarkers Identification by Mass Spectrometry in High-Mortality Tumors, International Journal of Proteomics, Volume 2013 (2013), Article ID 125858, 15 pages, ISSN 1874-3919.

[32] SAEYS Yvan, INZA Inaki y LARRANAGA Pedro, A review of feature selection techniques in bioinformatics, Bioinformatics Journal, ISSN 1460-2059, 2007.

[33] HE Ping, Classification Methods and Appplications to Mass Spectrometry Data, PhD. Thesis, Hong Kong Baptist University, 2005.

[34] XU Q. ,MOHAMED S.S. ,SALAMA M.M.A.,KAMEL M. y RIZKALLA K., Mass Spectrometry-Based Proteomic Pattern Analysis for Prostate Cancer Detection Using Neural Networks with Statistical Significance Test-Based Feature Selection , Science and Technology for Humanity (TIC-STH), 2009 IEEE Toronto International Conference.

[35] GUYON, I., GUNN, S., NIKRAVESH, M., ZADEH, L.A., Feature Extraction Foundations and Applications, pp. 90, Studies in Fuzziness and Soft Computing, Vol. 207, Springer Publications, ISBN 978-3-540-35488-8.

[36] SINGH Ajit P. , HALLORAN John , BILMES Jeff A. , KIRCHOFF Katrin , NOBLE William S. , Spectrum Identification using a Dynamic Bayesian Network Model of Tandem Mass Spectra, Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence (UAI2012), ISSN 2159-5399.

[37] BJM Webb-Robertson , Support vector machines for improved peptide identification from tandem mass spectrometry database search, Mass Spectrometry of Proteins and peptides: Methods in Molecular Biology Journal, Vol 146. Humana Press, New York, NY, ISSN 1064-3745.

[38] WU Baolin, ABBOTT Tom, FISHMAN David, MCMURRAY Walter, MOR Gil, STONE Kathryn, WARD David, WILLIAMS Kenneth and ZHAO Hongyu,, Comparison of statistical methods for classification of ovarian cancer using mass spectrometry data, March 6, 2003, Bioinformatics Journal, ISSN 1367-4803.

[39] QU Yinsheng, ADAM Bao-Ling, YUTAKA Yasui, WARD Michael D., CAZARES Lisa H., SCHELLHAMMER Paul F., FENG Ziding, SEMMES O. John, and WRIGHT JR. George L., Boosted Decision Tree Analysis of Surface-enhanced Laser Desorption/Ionization Mass Spectral Serum Profiles Discriminates Prostate Cancer from Noncancer Patients, October 2002 vol. 48 no. 10 1835-1843, Clinical Chemistry Journal, ISSN 0009-9147.

[40] NARSKY Ilya , PORTER Frank C., Statistical Analysis Techniques in Particle Physics: Fits, Density Estimation and Supervised Learning, Wiley-VCH; 1 edition (October 24, 2013), ISBN: 9783527677290 - 3527677291.

[41] HETHELYI E., TETENYI P.,DABI E, DANOS B., The role of mass spectrometry in medicinal plant research, Biological Mass Spectrometry Journal, Online ISSN: 1096-9888.

[42] IDOYAGA Moliona Natilde y LUXARDO Natalia, Medicinas no convencionales en cancer, Medicina (B. Aires) [online]. 2005, vol.65, n.5, pp. 390-394. ISSN 1669-9106.

[43] MANZANO SANTANA Patricia , ORELLANA LEON Tulio , MARTINEZ MIGDALIA Miranda C., ABREU PAYROL C. Juan , RUIZ Omar , PERALTA GARCIA C. Esther L., Algunos parámetros farmacognósticos de Vernonanthura patens (Kunth) H. Rob. (Asteraceae) endémica de Ecuador, Rev Cubana Plant Med vol.18 no.1 Ciudad de la Habana ene.-mar. 2013, ISSN 1028-4796.

APÉNDICE C

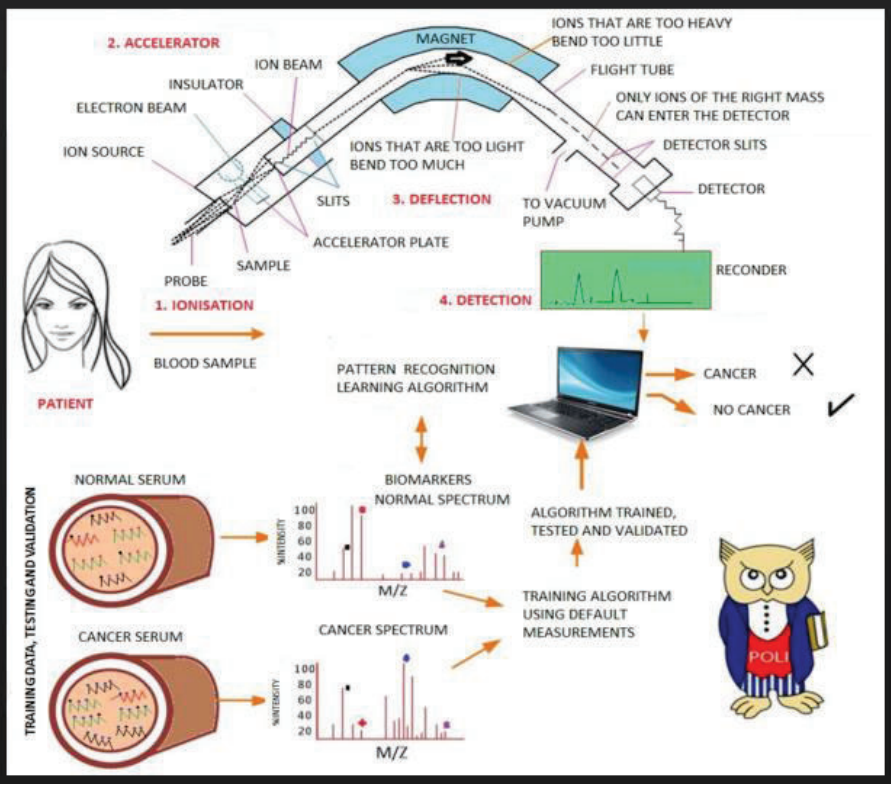
Pagina Web del Proyecto de Titulación

MS-ALGORITHM-ROI

by Sofia Calle & Silvia Chasiluisa

WEB SITE DEVELOPMENT THESIS

WELCOME MASS SPECTROMETRY.



RETURN

Introduction

Mass spectrometry (MS) produces data represented in vectors and matrices enormous. The measurements are represented by real numbers.

Detecting compound it may be performed with a very small sample of saliva, urine or blood and thus obtain the required information.

This technique represents your measurements as discrete signals having a major limitation when processing a large volume of data using computer software and statistical analysis, why apply algorithms to eliminate redundant data, lack of relevant data and information reduce noise instrumentation.

The information obtained is stored in digital files ready to be processed computationally, this processing is presented as a set of steps that seeks to eliminate the errors of calibration and measurement errors. At the end of this process a set of smaller size data is obtained and completely homogenous.

Mass spectrometry measurements produced containing many peaks that are useful for identifying compounds and detecting the different anomalies or asymptomatic diseases such as cancer in humans using spectra.

RETURN

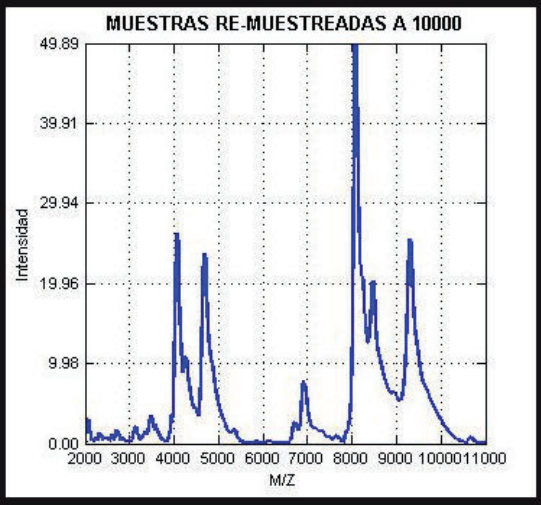
Download .zip Download .tar.gz

RETURN

Processing Data of Mass Spectrometry

Signal Resampling

Resampling is a process in which a new signal is obtained with controlled m / z values, this new signal should be as far as possible similar to the original. Controlled values means that the number of points may be higher, at or below the original signal. When the signal is resampled higher resolution down sampling is performed when the signal is of lower resolution sampling is above and to obtain the same resolution only syncs. The EM data are high resolution so it is difficult to work with computational intensive algorithms that can reach the limit of your computer. An acceptable measure to obtain a decision manageable data by eliminating redundant data signals thus obtaining uniform length. To reduce or increase the resolution of the spectra using a sampling frequency converter with the factor I / D. Where, I represents interpolation and D decimation.



Correcting the Background

The data generally show a variable line based on a result of chemical noise or overload matrix ions. It

Download .zip Download .tar.gz

RETURN

Feature Selection

This section describes the methods used to reduce data dimencionalidad explained. This process is to select the percentage of samples that will be discarded based on the results of statistical tests performed. The dimensionality reduction is essential in terms of computational time optimization and reduction processing the effects of over-training and under-training at the stage of pattern classification.

T-test

Test T-test is a parametric test that helps in checking a property assuming some hypotheses on data. This test using the Student t-distribution, this dristibución is a family of curves that varies depending on the degrees of freedom. To apply the t-test is assumed that:
1.The distribution of data should follow a Gaussian or normal curve.
2. The standar deviation of the data are unknown but are assumed to be equal.
Hypotheses to test: Verify that the means of two groups of samples are equal. Where the data of each sample can be vectors or matrices. To apply the test is necesary define the statistical "t", that helps to accept or reject the hypothesis by comparing the "t" calculated with "t" defined Statistical tables available. Calculating the statistic is computed as follows:

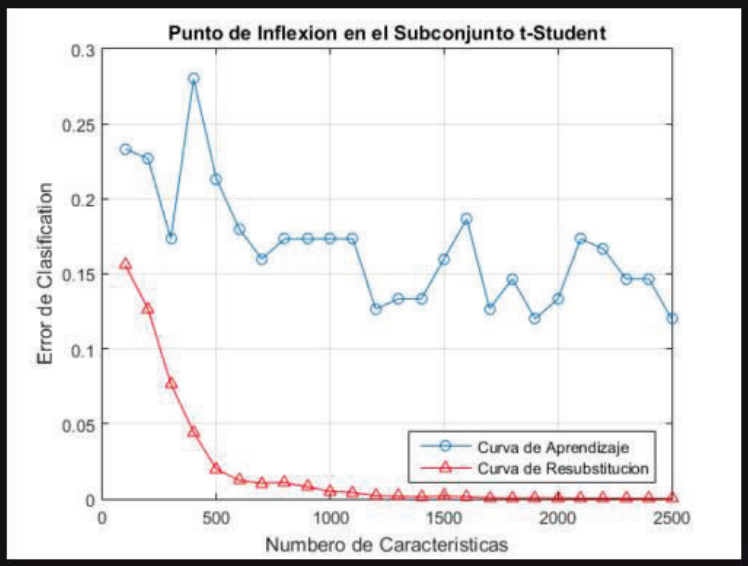
$$t = \frac{\mu_A - \mu_B}{s\sqrt{\frac{1}{m_A} + \frac{1}{m_B}}},$$

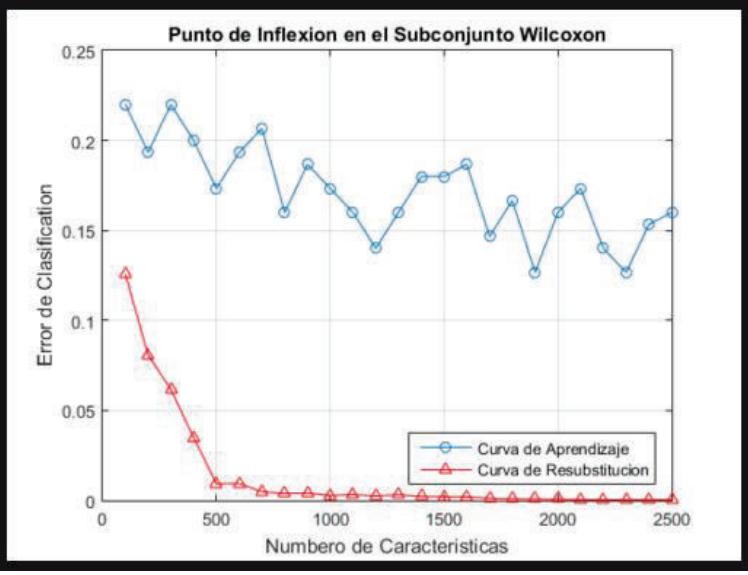
with:

$$s = \frac{(m_A - 1)s_A^2 + (m_B - 1)s_B^2}{m_A + m_B - 2},$$

Validation

Conjuno Arcene





RETURN

Results and Publications

PUBLICATION: NEW ALGORITHMS FOR EARLY DETECTION OF CANCER

Last Politecnico Information in pdf on pages 24 and 25 of the link:

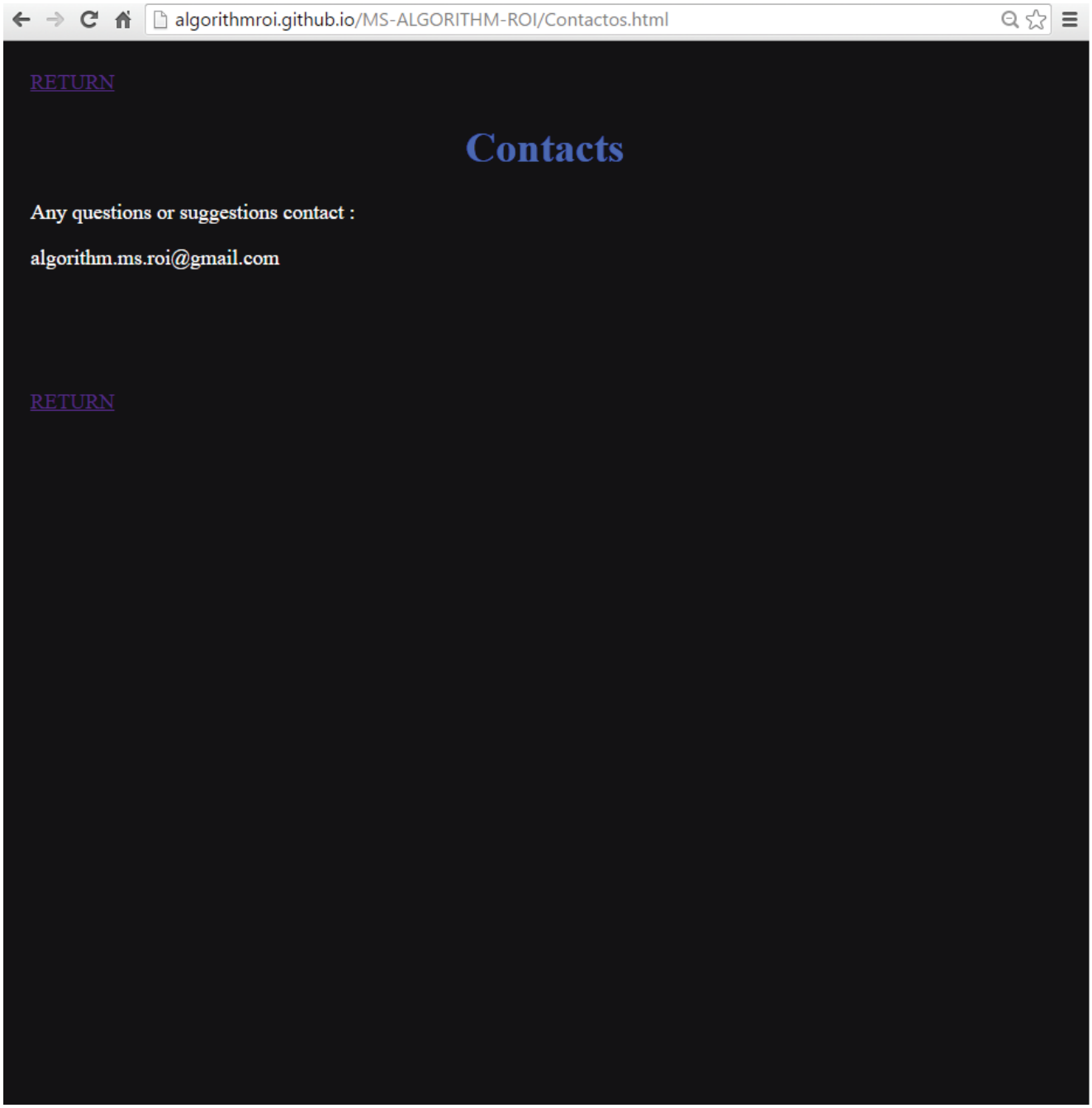
POLITECNICO INFORMATION GO TO PAGE

PUBLICATION: MASKANA MAGAZINE

MASKANA MAGAZINE GO TO PAGE

Download .zip Download .tar.gz

RETURN


algorithmroi.github.io/MS-ALGORITHM-ROI/Contactos.html
RETURN
Contacts
Any questions or suggestions contact :
algorithm.ms.roi@gmail.com
RETURN