Universitat Oberta de Catalunya

uoc.edu

Análisis Estadístico de Imágenes Hiperespectrales para la Clasificación de Tumores Cerebrales

Nombre Estudiante: Cristina Lendinez González

Plan de Estudios del Estudiante: Máster Bioinformática con Bioestadística
Área del trabajo final

Director: Edwin Santiago Alférez Baquero

Fecha Entrega 02/06/2022

Cristina Lendinez

Licencias alternativas (elegir alguna de las siguientes y sustituir la de la página anterior)

Cristina Lendinez

A) Creative Commons:

Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivada 3.0 España de Creative Commons

Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-CompartirIgual 3.0 España de Creative Commons

Esta obra está sujeta a una licencia de Reconocimiento-NoComercial 3.0 España de Creative Commons

Esta obra está sujeta a una licencia de Reconocimiento-SinObraDerivada 3.0 España de Creative Commons

Esta obra está sujeta a una licencia de Reconocimiento-CompartirIgual 3.0 España de Creative Commons

Esta obra está sujeta a una licencia de Reconocimiento 3.0 España de Creative Commons

B) GNU Free Documentation License (GNU FDL)

C) Copyright

FICHA DEL TRABAJO FINAL

Título del trabajo:	Análisis Estadístico de Imágenes Hiperespectrales para la Clasificación de Tumores Cerebrales
Nombre del autor:	Cristina Lendinez González
Nombre del consultor/a:	Edwin Santiago Alférez Baquero
Nombre del PRA:	Laura Calvet
Fecha de entrega (mm/aaaa):	06/2022
Titulación:	Máster Bioinformática con Bioestadística
Área del Trabajo Final:	Trabajo Fin de Máster
Idioma del trabajo:	Español
Número de créditos:	15 créditos
Palabras clave	Imágenes Hiperespectrales, Machine Learning Máquinas de Vector Soporte

Cristina Lendinez

Resumen del Trabajo (máximo 250 palabras): *Con la finalidad, contexto de aplicación, metodología, resultados i conclusiones del trabajo.*

El objetivo general del proyecto es la clasificación de diferentes áreas de tumores cerebrales a través de imágenes hiperespectrales. Para clasificar las imágenes usaremos algoritmos de Machine Learning, específicamente la Máquina de Vectores de Soporte (SVM).

En el diagnóstico médico se debe de ser lo más riguroso posible, y el algoritmo puede ser usado en el quirófano, es necesario mostrar al cirujano por medio de una pantalla, una imagen procesada del paciente con los tejidos ya clasificados.

La metodología llevada a cabo se basa en comprobar cómo estos modelos entrenados predicen las diferentes áreas del tumor, obteniendo resultados para determinar la mejor forma para entrenar un modelo. Por ello tendremos en cuenta el entrenamiento y del paciente al que corresponden.

Se concluye que el mejor modelo para realizar este TFM son las SVM en este caso se usaran únicamente los datos propios de cada paciente para entrenar (y predecir)los tipos de tejidos.

Abstract (in English, 250 words or less):

The general objective of the project is the classification of different areas of brain tumors through hyperspectral images. To classify the images we will use Machine Learning algorithms, specifically the Support Vector Machine (SVM).

In the medical diagnosis, it must be as rigorous as possible, and the algorithm can be used in the operating room, it is necessary to show the surgeon through a screen, a processed image of the patient with the tissues already classified.

The methodology carried out is based on checking how these trained models predict other hyperspectral images, obtaining results to determine the best way to train a model. For this reason, we will take into account the training and the corresponding patient.

It is concluded that the best model is the SVM, in this case only the data of each patient will be used to train (and predict) the types of tissue.

Contenido

Índice de Figuras:

Índice de Tablas:

Introducción

Contexto y justificación del Trabajo

Este trabajo de Fin de Máster (TFM) se centra en el desarrollo de un método que permita el uso de imágenes hiperespectrales durante una operación de resección de un tumor cerebral. Los tumores cerebrales se encuentran en una zona muy delicada del cerebro, por ello, es crucial delimitar el tejido sano del canceroso. La práctica habitual a la hora de extirpar un tumor es trazar un amplio margen para asegurar la completa resección del tejido canceroso, pero esta práctica no puede realizarse en el cerebro, ya que puede suponer la afectación de funciones vitales del paciente. Entonces, para afectar lo mínimo posible a la calidad de vida del paciente y conseguir la mejor recuperación posible, es esencial identificar con exactitud la frontera entre tejido canceroso y sano. Para este fin se plantea el uso de aprendizaje de máquina junto con las Imágenes hiperespectrales.

La ciencia del aprendizaje y las redes neuronales juega un papel clave en los campos estadísticos, el análisis de datos e inteligencia artificial, que están destinadas a las áreas de ingeniería y otras disciplinas. Es por eso por lo que este trabajo es parte del proyecto

Helicoid, realizado en el Centro de Investigación sobre Sostenibilidad de la Salud y los Sistemas Multimedia (CITETEM) de la Universidad Politécnica de Madrid (UPM). *Helicoid*, el acrónimo de inglés es la detección de cáncer de imagen Hyper-Name, es un proyecto europeo que comenzó en enero de 2014 y se basa en el estudio de imágenes de hiper-visión para realizar una discriminación real entre tejidos sanos y un paciente tejido por cáncer.

En consecuencia, este proyecto se basa en el estudio o el diseño de un método que nos permita aplicar el aprendizaje supervisado durante una operación quirúrgica para facilitar la resección del tumor al cirujano. Para hacer esto, es necesario utilizar la información obtenida en operaciones anteriores con la información obtenida durante la operación actual, para proporcionar al cirujano una imagen RGB en la que se señalen 4 clases (Sanas, Cáncer, Vena, Arteria, y Duramadre) y de esta forma ayude a tomar la decisión del tipo de tejido que debe ser eliminado del cerebro del paciente.

Con este TFM se pretende desarrollar un modelo, que a través de Imágenes Hiperespectrales permita la clasificación del tejido tumoral y otras estructuras neurales.

Objetivos del Trabajo

El objetivo general de este trabajo es realizar un análisis estadístico de imágenes hiperespectrales para la clasificación de tumores cerebrales.

Para llevar a cabo el propósito general de este trabajo, se realizarán los siguientes objetivos específicos:

- Determinar mediante inferencia estadística si hay diferencias significativas entrelas bandas hiperespectrales.
- Determinar si las bandas hiperespectrales correspondientes están correlacionadas.
- Desarrollar un modelo de clasificación automática, con base en máquinas de vectores de soporte, para reconocer hasta cinco clases de zonas tumorales (sano,cáncer, vena, arteria y duramadre).
- Evaluar el desempeño del modelo de clasificación automática desarrollado.

Enfoque y método seguido

En este trabajo se realizarán los siguientes pasos:

- Estudiar todos los conceptos estadísticos que cualquier modelo predictivo de inferencia estadística debe seguir, para evaluar con calidad científica comprobada los diferentes aspectos a considerar en este trabajo. De la misma manera, se estudiarán las pruebas de diagnóstico (medidas de discriminación y clasificación) necesarias para lograr los objetivos.

- Para hacer esto, se estudiará la literatura científica, particularmente donde se explica cuándo y cómo analizar estos modelos de tipo estadístico en el campo de las ciencias de la vida. Una vez conocido todos los conceptos estadísticos necesarios, definimos los pasos desde un punto de vista estadístico.

- Usar Python o R y sus diferentes paquetes para calcular los parámetros estadísticos a evaluar.

- Se realizarán unas primeras pruebas de los métodos estadísticos del código para conocer en profundidad su implementación. En este paso, será necesario determinar las variables de interés.

- Se realizarán unas pruebas con datos reales obtenidos a partir de los diferentes pacientes. En este paso se busca solventar problemas con el código de programación en el caso que fuera necesario.

- Las etapas 2, 3 y 4 se repetirán varias veces para mejorar el código escrito, como eliminar variables redundantes, eliminar líneas de código, inclusión de explicaciones cortas en el código, modificación de formatos de los diferentes archivos de datos.

TABLA 1: PLANIFICACIÓN A TRAVÉS DE LAS TAREAS

Tareas	Fecha de Inicio	Días	Fecha Fin
Definición de los contenidos del TFM	16/02/2022	6	22/02/2022
PEC 0	23/02/2022	9	04/03/2022
PEC 1	05/03/2022	34	08/04/2022
Búsqueda en Pubmed	17/02/2022	6	23/02/2022
Manejo de Visual Studio	17/02/2022	12	01/03/2022
Manejo de Librerías	18/02/2022	91	20/05/2022
Estudio de Inferencia Estadística	01/03/2022	80	20/05/2022
Extracción de base de datos	28/02/2022	3	03/03/2022
Primeras pruebas con datos de imágenes	04/03/2022	11	15/03/2022
Primeros análisis de Anova de 2 vías	16/02/2022	32	20/03/2022
Seguimiento de pruebas Inferenciales	21/03/2022	9	30/03/2022
PEC 2	03/03/2022	36	08/04/2022
Correlaciones	09/04/2022	7	16/04/2022
KFold Validaciones Cruzadas	20/04/2022	11	01/05/2022

Primeras pruebas con Maquinas de Soporte Vectorial	03/05/2022	17	20/05/2022
PEC 3	11/05/2022	5	16/05/2022
Redacción de memoria del TFM	20/05/2022	13	02/06/2022
Elaboración de la presentación	25/12/2021	6	31/12/2021
Defensa Publica	15/06/2022	6	21/06/2022

La planificación de tareas se muestra en la Figura 1 donde se ilustra el gráfico de Gantt de la planificación realizada. La lista de tareas a realizar se indica a continuación para llegar a la realización del TFM.

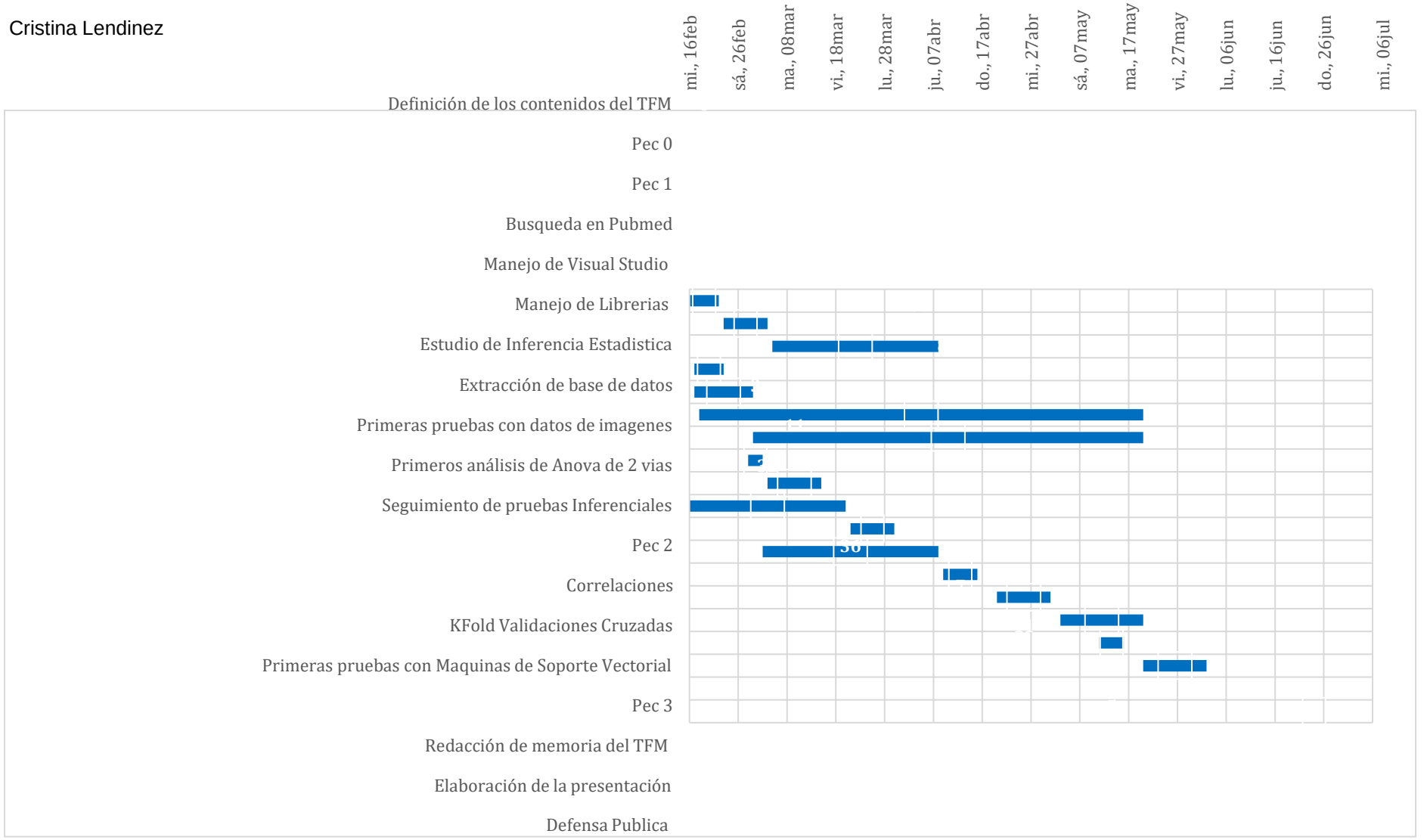


FIGURA 1: DIAGRAMA DE GANTT PARA ILUSTRAR LA PLANIFICACIÓN DEL TRABAJO

Breve sumario de contribuciones y productos obtenidos

- **PEC0:** Pertenece al objetivo principal, definir claramente el tema del trabajo, justifique su interés y / o relevancia y qué se desea lograr al final de TFM
- **PEC1:** Pertenece al objetivo principal, consiste en desglosar todo el contenido que tendrá el futuro trabajo académico.
- **PEC2:** en la parte técnica, el filtrado de secuencia debe haberse completado y evaluado. La introducción y una gran parte de la metodología deben escribirse.
- **PEC3 (17/05):** en la parte técnica, el genoma debe haberse montado en las dos plataformas, así como en las carreras con los dos programas. De la misma manera, y como parte del proceso, la evaluación de la plataforma y la necesidad de usar ambos. La escritura incluye secciones de metodología y resultados completos.
- **PEC4 (18/05):** En la parte técnica, la asamblea debe ser educada. El ensayo se completa con las secciones de discusión y conclusión y el resumen está escrito.
- **PEC5 A (06/13):** la presentación del trabajo se prepara en el formato correspondiente.
- **PEC5 B (21/06/23):** la defensa del trabajo se realiza a través de un videoexplicativo.

Análisis de Riesgo

En la Tabla 2 se detallan los riesgos según la planificación de las actividades·

TABLA 2: ***RIESGOS QUE PUEDEN OCURRIR A LO LARGO DE ESTE TRABAJO.***

Descripcion del proceso	Severidad	Probabilidad	Mitigación
Dataset Extensos	Moderada	Baja	Los análisis se hacen con los datos obtenidos por la Universidad Politécnica de Madrid
Problemas con las librerías	Alta	Baja	Podemos tener problemas con algunas librerías, ya que algunas son nuevas para mí, como pueden ser Trelliscope .

Dataset demasiado voluminosos	Moderado	Alto	Intentaremos optimizar todos los códigos posibles, siempre podremos crear funciones, que nos permitan optimizar más el tiempo.
KFold con validaciones cruzadas	Moderado	Alto	Tenemos problemas con el balanceo de las etiquetas y esta técnica nos ayudara, ya que no podemos quitar variables
Problemas con algoritmos	Moderado	Alto	Buscar ayuda en diferentes páginas en foros como Stackoverflow y pedir ayuda al tutor.

Los resultados esperados son:

- El documento de la memoria del TFM
- Vídeo de la presentación virtual
- Modelos para la clasificación automática de diferentes tipos de tejido cerebral
- Código desarrollado en el anexo

Breve descripción de los otros capítulos de la memoria

La memoria se ha dividido en las siguientes secciones:

- Capítulo 1. Introducción: en este capítulo se describe la motivación del trabajo, los objetivos y las tareas ajustadas a este trabajo,
- Capítulo 2. Imágenes hiperespectrales y de los métodos de inferencia estadística y machine learning.
- Capítulo 3. Materiales y métodos: explicación de los métodos propuestos y desarrollo del trabajo.
- Capítulo 4. Resultados: resumen de los resultados obtenidos de los análisis realizados en los modelos estadísticos y en el modelo de machine learning.
- Capítulo 5. Discusión y conclusiones: valoración del trabajo y futuras líneas de investigación.
- Glosario; definición de termino y acrónimos.
- Bibliografía: bibliografía usada en el TFM.

- Anexo: incluirá todo el código usado en el TFM.

Imágenes Hiperespectrales

Como se ha citado anteriormente, este Trabajo Fin de Máster, procura implementar el algoritmo de Machine Learning de Maquina de Vector Soporte. Por todo esto, explicaremos el concepto de imagen hiperespectral (1) y definiremos su utilización en el campo de la medicina

Una imagen hiperespectral es la representación de un objeto en función de su longitud de onda que está reflejando, es un conjunto de imágenes en el que cada imagen simboliza una longitud de onda distinta. Por otra parte, la reflectancia es el porcentaje de luz que incide sobre un material observado y reflejado por él. Por consiguiente, para un píxel constar de tres valores (bandas de color azul, rojo y verde), como ocurre en las imágenes RGB, En las imágenes hiperespectrales, cada píxel puede contener cientos de valores de reflectancia asociados a distintas bandas distribuido por todo el espectro electromagnético, dichas bandas pueden pertenecer al espectro visible, infrarrojo o ultravioleta.

En este tipo de imágenes, debemos de conocer conceptos básicos sobre las mismas, las métricas usadas en su procesamiento, conocemos que los sensores hiperespectrales tienen capacidad de registrar y cuantificar variaciones de longitud de onda desde el espectro visible (0.4µ, 0.7µm) hasta microondas(30cm) dependiendo de la reflectancia del objeto.

Estos valores de reflectancia medidos en cada banda de la imagen se pueden definir como firma espectral, esta firma espectral es usada para calificar un material de forma evidente.

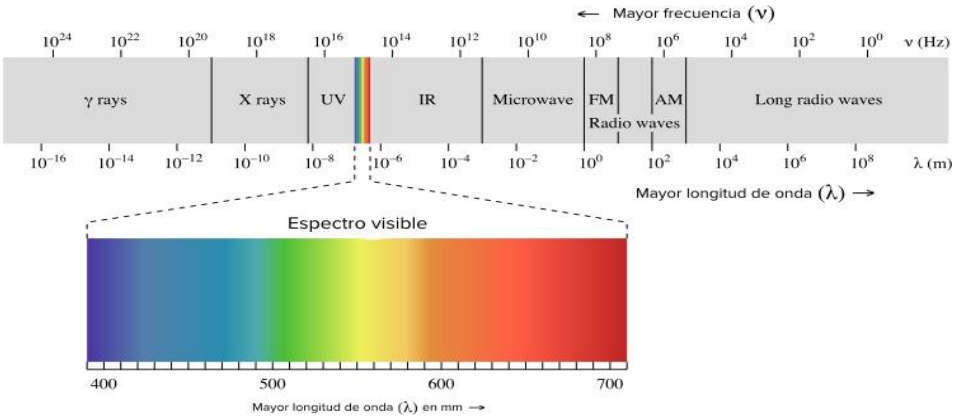


FIGURA 2: ***EL ESPECTRO ELECTROMAGNÉTICO (2)***

Estas imágenes de alta densidad espectral son guardadas en forma de una matriz tridimensional, generándose lo que conocemos como cubo de datos hiperespectral o hipercubo, al que podemos indiciarlo como $X \in R^{n1 \times n2 \times d}$, siendo n1 y n2 el alto y ancho de las imágenes, (número de filas y numero de columnas y d es el número de bandas espectrales (la profundidad de la matriz del hipercubo).

El número de bandas que contienen estás imágenes es muy diferente con las de las imágenes RGB, las imágenes hiperespectrales sus bandas son contiguas. Con todo esto los resultados que se obtendrán proporcionan un espectro continuo. En la Figura 3, se presenta una comparación minuciosa entre las diferentes imágenes RGB y las imágenes hiperespectrales.

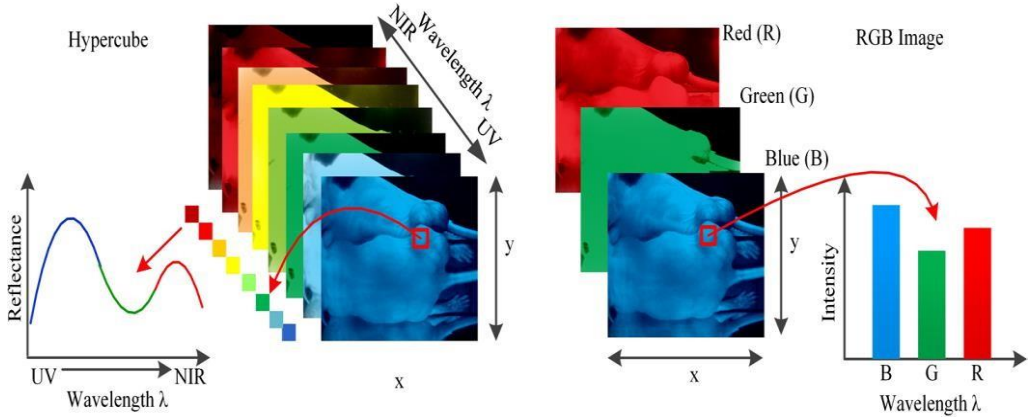


FIGURA 3: COMPARACIÓN ENTRE IMAGEN HIPERESPECTRAL E IMAGEN (RGB)

Para las imágenes hiperespectrales, la firma espectral es un conjunto de valores de luminosidad, o reflectancia, que tiene un píxel para cada una de las bandas que pertenecen a la imagen hiperespectral. La firma espectral perteneciente de cada píxel se usa en los análisis para identificar el material que se hallan en cada píxel. Con todo esto, podemos decir que un píxel no este compuesto por un único material, podrían aparecer varios tipos de material, el cual denominamos cómo píxel mezcla, además podemos hallar lo que se conoce como píxel puro. Estos píxeles se identifican como aquellos que solo contienen un material y que, además, su firma espectral es idéntica con la del material correspondiente.

Podemos ver en la, ejemplos de los diferentes tipos de pixeles, el píxel del agua es un píxel puro, ya que su firma espectral concuerda con la del material, en otros tipos de materiales como son rocas y plantas los pixeles son mezcla, en la Figura 4, podemos ver los diferentes tipos de pixeles según el material.

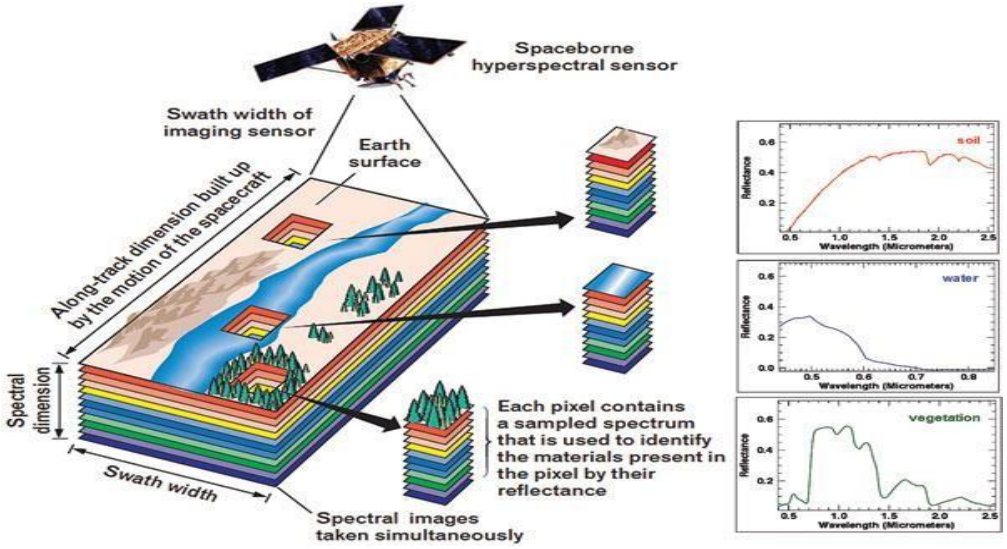


FIGURA 4: ***DIFERENTES PIXELES EN IMÁGENES HIPERESPECTRALES (2)***

Imágenes hiperespectrales en medicina

Las imágenes hiperespectrales han tenido un gran auge en la medicina, particularmente en la tecnología hiperespectral para la detección de tumores (5), siendo de gran utilidad para guiar a los cirujanos en las resecciones de estos tumores en tiempo real. Como ya se ha comentado, esta apuesta se fundamenta en la identificación de la firma espectral de los tumores debidos a los cambios morfológicos y bioquímicos que contienen estos tejidos, estas imágenes son de gran ayuda para clasificar el material de cada píxel que contiene la imagen del tumor cerebral, clasificando las clases con las que se trabaja (en este trabajo las clases corresponden a: Sano, Tumor, Vena, Arteria, Duramadre).

Estas imágenes se obtienen midiendo de cada píxel la intensidad de la luz para una longitud de onda particular, los sensores recogen la información como un conjunto de imágenes. Estas imágenes fueron tomadas en medidas secuenciales de los espectros línea a línea del área del cerebro de interés. Se usaron dos cámaras hiperespectrales una en el rango VNIR 400nm-1000nm y otra en el rango NIR 900nm-1700nm.

Los pacientes del estudio *Helicoid* pertenecen al servicio de Neurocirugía del Hospital Universitario 12 de octubre de Madrid, los pacientes dieron su consentimiento para formar parte del estudio y firmaron el consentimiento informado, con todo ello ya forman parte del proyecto *Helicoid*, cuyo fin principal consiste en el uso de imágenes hiperespectrales para la localización precisa de tumores neurales en procedimientos quirúrgicos y con todo el material procedente de estos pacientes durante su cirugía se pueda usar la inteligencia Artificial para el estudio de predicción de la clasificación de los diferentes tumores y sus áreas neurales.

Las imágenes fueron tomadas durante una intervención quirúrgica cerebral. El cirujano coloca dos anillos para identificar la ubicación del tumor (rodeado con un manguito), y otros anillos ubicados en la región sin tumor durante la cirugía, se tomaron muestras para el Neuropatólogo que será el encargado de diagnosticar mediante histología el tipo de tumor del paciente, con todo esto se pudo cotejar los resultados del Neuropatólogo en el laboratorio de Anatomía Patológica con las imágenes obtenidas de estas ubicaciones.

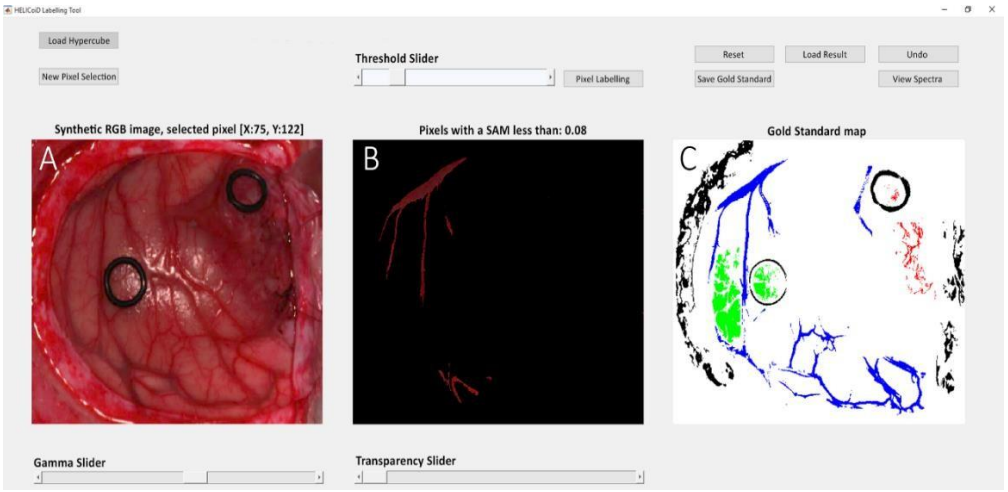


FIGURA 5: ***IMAGEN QUIRÚRGICA COMPARANDO CON IMÁGENES HIPERESPECTRAL***

Como podemos ver en la Figura 5 se seleccionan en el quirófano las áreas tumorales con un manguito de goma y se compara con las imágenes hiperepectrales

Metodología

Comité de ética y obtención de imágenes hiperespectrales

En los proyectos de investigación su principal objetivo es garantizar el respeto, identidad e integridad de los pacientes pertenecientes en el estudio en el caso que el proyecto de investigación conste de muestras biológicas, datos de origen humano o como en este proyecto de investigación de imágenes hiperespectrales de cirugías de extirpación de tumores neurales, como es el caso del proyecto *Helicoid,* El Real Decreto 1090/2015 la formación de un Comité Ético de Investigación Clínica debe estar compuesto por al menos 10 miembros, entre estos miembros a al menos un médico y, en el caso de ser varios, uno de ellos debe ser farmacólogo clínico. Además, debe incluir a un farmacéutico, a un enfermero y a un integrante que represente a los pacientes y a sus intereses, ajeno a la asistencia clínica y a la investigación biomédica. Una vez aprobado el proyecto por el comité de ética, se procede al inicio del proyecto de investigación.

En este trabajo se usaron imágenes hiperespectrales de seis pacientes pertenecientes del servicio de neurocirugía con una patología previa de tumor neural que debía ser eliminado mediante neurocirugía en el hospital 12 de octubre de Madrid, los pacientes previamente fueron informados del proyecto *Helicoid* dieron su autorización y firmaron el consentimiento informado para pertenecer al estudio *Helicoid.* Durante la cirugía, los tumores fueron fotografiados por las cámaras hiperespectrales, con el fin de procesar estas imágenes y mediante algoritmos de machine learning e inteligencia artificial intentar predecir las diferentes áreas del tumor y así ayudar a su resección completa o su mayor eliminación debido a que no se pueda eliminar por completo debido a la complejidad del órgano.

Procesado de las imágenes hiperespectrales

Las imágenes hiperespectrales que se usaron en este TFM fueron capturadas durante una cirugía. Entonces, se creó una base de datos donde se guardaron las imágenes de distintas patologías. Tras la obtención de las imágenes, los neurocirujanos etiquetaron las diferentes áreas tumorales a partir de las imágenes obtenidas en el quirófano, en esta clasificación se han clasificado 5 clases: tejido cerebral sano, tumor, vena, arteria y duramadre.

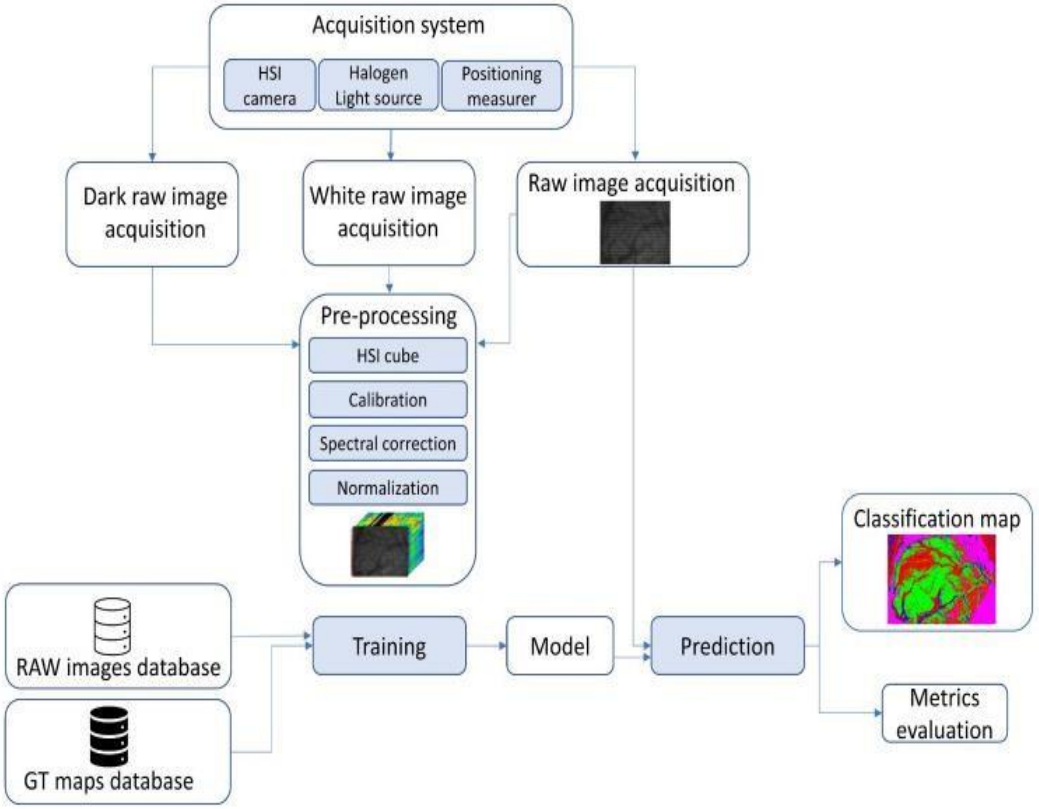
Las imágenes hiperespectrales se procesaron para homogeneizar las firmas espectrales de estas imágenes etiquetadas, mediante los siguientes pasos:

- Crear los cubos hiperespectrales, las imágenes se transforman en cubos hiperespectrales, el área de la imagen hiperespectral contiene una resolución de 2045x1085 pixeles con una composición de 25 longitudes de onda en bloques de mosaicos repetitivos (5 x 5), por todo esto el cubo hiperespectral tiene una resolución espacial de 419 x 27 junto con una resolución de 25 bandas (7).

- Calibración: sirve para la reproducibilidad de estos datos.

- Corrección espectral: debido a la alta sensibilidad de las cámaras, las curvas de respuesta conllevan una diafonía en su longitud de onda máxima de los vecinos. De este modo diferentes curvas poseen armónicos secundarios que no pueden eliminarse con filtros de paso largo o paso corto. Toda esta incidencia puede serquitado por el proceso de corrección espectral (7)

- Normalización de datos: debido a la morfología del cerebro, los pixeles no se encentran todos a la misma altura, por ello, la luz capturada por la cámara no es simétrica en toda la zona, esto puede interferir ya que la clasificación de cada zona en función del de cada píxel. Para atenuar este efecto se utilizan técnicas de estandarización, con el fin de preservar la firma espectral. Los valores cuadráticos medios (RMS) de estas firmas espectrales de todas las bandas se usan como coeficientes con el fin de normalizar el cubo con



corrección espectral(7)

FIGURA 6: ***DIAGRAMA DE METODOLOGÍA DISEÑADO PARA CONVERTIR IMÁGENES EN DATASET***

Como podemos ver en la Figura 6, los datos que han sido introducidos en la base de datos ya han sido preprocesados (se han normalizado y calibrado), por lo cual, no tenemos la capacidad de volver a la imagen original. Por esta razón, en este TFM solo trabajaremos en la realización de los análisis a partir de ese punto.

Exploración de los datos

Vamos a describir los conjuntos de datos de los 6 pacientes con que se trabaja en esteTFM:

- **Paciente 1:** Consta de 25 bandas hiperespectrales, y el área del Tumor se divide en 4 Zonas (1438 píxeles pertenecientes a Arteria, 1707 píxeles pertenecientes a Sano, 120 píxeles pertenecientes a Tumor, y 63 píxeles pertenecientes a Vena).
- **Paciente 2:** Consta de 25 bandas hiperespectrales y el Área del Tumor se divide en 5 Áreas Tumorales (176 píxeles de Arteria, 780 píxeles de Duramadre, 1186 píxeles de Sano, 1464 píxeles de Tumor y 181 píxeles de Vena).
- **Paciente 3:** Consta de 25 bandas hiperespectrales y el Área del Tumor se divide en 4 áreas Tumorales (2823 píxeles de Arteria, 3073 píxeles de Sano, 366 píxeles de Tumor, 795 píxeles de Vena).
- **Paciente 4:** Consta de 25 bandas hiperespectrales y el área del Tumor se divide en 5 Áreas Tumorales (82 píxeles de Arteria, 1106 píxeles de Duramadre, 391píxeles de Sano, 433 píxeles de Tumor, 76 píxeles de Vena).
- **Paciente 5:** Consta de 25 bandas hiperespectrales y el área del Tumor se divide en 5 Áreas Tumorales (256 píxeles de Arteria, 220 píxeles de Duramadre, 3063 píxeles de Sano, 110 píxeles de Tumor y 172 píxeles de Vena).

Librerías y Paquetes

Para la realización de los análisis estadísticos se utilizó el lenguaje de programación estadística **R**, ya que es un software muy valorado en toda la temática de la Ciencia de Datos.

Las librerías que se usaron para realizar estas pruebas de hipótesis fueron:

- **Dplyr**: su principal función de este paquete es realizar operaciones de manipulación de datos, filtrar columnas, añadir columnas nuevas, ordenar filas y añadir datos nuevos.
- **Shapiro.text**: es una de las librerías mejor valoradas para probar la normalidad de una variable, el tamaño de su muestra debe de ser inferior a 5000.
- **Levene.test**: es una librería que se utiliza para valorar la igualdad de las varianzas en una variable determinada para comprobar o en más de una.
- **res.aov**: es una librería que sirve para saber si existe alguna diferencia significativa entre las 25 bandas hiperespectrales y las Áreas del Tumor.
- **Tukey-HSD:** su principal función es realizar pruebas múltiples de comparaciones por pares entre las medias de las 25 bandas hiperespectrales en relación a las Áreas del Tumor

Otras librerías, de Python, que se usaron para el desarrollo de la metodología fueron:

- **Pandas**: es una librería, su principal función es la limpieza y el análisis deDatos.
- **Numpy:** su principal función es trabajar con matrices.
- **Matplotlib**: su principal función es la visualización de datos y trazado degráficos.
- **Sklearn**: está librería dispone de algoritmos de Machine Learning como KNN, SVM, etc.
- **Cv2**: su principal función es la detección de rostros y objetos.

Estadística Descriptiva

En la Tabla 3, podemos ver las estadísticas descriptivas de las bandas hiperespectrales y verificar cómo se comportan las medias y las desviaciones estándar, comprobar si tenemos valores faltantes o si tenemos diferentes tipos de variables. Además, los gráficos boxplot, nos ayudan a comprobar como se comportan la media junto a la desviación estándar, estos gráficos marcan estos 2 componentes de la estadística descriptiva. (En el anexo encontramos los gráficos desarrollados)

TABLA 3: ESTADÍSTICA DESCRIPTIVA

Zona	Banda	Variable	nº de observaciones	media	Desviación estándar
Arteria	Banda_0	medida	111	1.195	0.071
Duramadre	Banda_0	medida	704	1.267	0.070
Sano	Banda_0	medida	1821	1.056	0.038
Tumor	Banda_0	medida	565	0.898	0.039
Vena	Banda_0	medida	166	0.939	0.109
Arteria	Banda_1	medida	111	1.181	0.051
Duramadre	Banda_1	medida	704	1.260	0.051
Sano	Banda_1	medida	1821	1.074	0.030
Tumor	Banda_1	medida	565	0.933	0.026
Vena	Banda_1	medida	166	0.999	0.097

Prueba de Normalidad

El análisis de normalidad, o contrastes de normalidad, la finalidad es analizar cuánto difiere la distribución de los datos observados con respecto a lo esperado si nuestros datos procediesen de una distribución normal con la misma media y desviación típica.

Se pueden realizar diferentes pruebas; representaciones gráficas, y test de hipótesis.

Métodos gráficos:

En estos métodos gráficos encontramos los histogramas y los gráficos de cuantiles (gráficos Q-Q), los histogramas, representan los datos mediante un histograma con una curva de una distribución normal con la media y la desviación estándar, este histograma se superpone sobre nuestros datos

Los gráficos de cuantiles en los que comparamos los cuantiles de la distribución observada de nuestros datos con los cuantiles teóricos de una distribución normal con la misma media y desviación estándar que los datos. Cuanto más próximos estén nuestros datos a la recta de normalidad, esto nos indicara que nuestros datos siguen la normalidad. En este trabajo se usarán los gráficos Q-Q(Anexo)

Test de hipótesis:

Los test de hipótesis son los más empleados para analizar la normalidad de los datos, consideramos como hipótesis nula que los datos sí proceden de una distribución normal y en el caso contrario como hipótesis alternativa que nuestros datos no cumplen la normalidad. El *p-value* nos indica la probabilidad de obtener una distribución como la observada si los datos proceden realmente de una población con una distribución normal.

- H0: La distribución es normal
- H1: La distribución no es normal,

Hacemos la prueba de Shapiro Wilks para comprobar si las bandas siguen la normalidad.

Como podemos comprobar en la Tabla 4 la gran mayoría de los datos siguen la normalidad, ya que su p-valor es superior a 0.05 (8) (Gráficos de normalidad en él anexo).

TABLA 4: *NORMALIDAD DE LOS DATOS*

Zona	Banda	variable	estadístico	valor-p
Arteria	Banda_10	medida	0.9886442	4,80E-01

Duramadre	Banda_10	medida	0.9983109	7,38E-01
Sano	Banda_10	medida	0.9987257	2,03E-01
Tumor	Banda_10	medida	0.9970559	4,04E-01
Vena	Banda_10	medida	0.9915053	4,39E-01
Arteria	Banda_11	medida	0.9846425	4,32E-01
Duramadre	Banda_11	medida	0.9987292	2,34E-01
Sano	Banda_11	medida	0.9995732	9,64E-01
Tumor	Banda_11	medida	0.9966996	3,02E-01
Vena	Banda_11	medida	0.9874062	1,42E-01

Análisis de Anova de 2 vías mixto y Anova Factorial no paramétrico

El estudio ANOVA es una técnica estadística que estudia el efecto que tiene uno o varios factores sobre la media de una variable objetivo. Se trata de una herramienta fundamental para el análisis de diseño de experimentos.

En este trabajo se usa la prueba de Anova de 2 Vías Mixto, y la prueba de Anova de Factores no Paramétrico, como herramientas para confirmar o descartar la existencia de diferencias estadísticas significativas entre varios conjuntos de bandas hiperespectrales y áreas de tumor (9).

El Anova de 2 vías mixto será usado para evaluar simultáneamente el efecto que tenemos en 2 variables categóricas(factores), uno de muestras independientes que en este estudio son las áreas del tumor y otro de muestras relacionadas sobre una variable continua que en este estudio son las 25 bandas hiperespectrales, en el anexo, se puede observar las 2 variables categóricas (En el anexo encontramos desarrollado el análisis)

A pesar de su robustez la prueba de ANOVA de 2 Vías Mixto y la prueba de Anova Factorial no paramétrico tienen una importante desventaja en el caso de rechazar la hipótesis nula, ya que no aporta información sobre cual o cuales son las medias (μ) que difieren del resto de valores. Esta desventaja se puede solucionar aplicando la prueba de Comparaciones Múltiples Post-Hoc (hablaremos de ellas en el siguiente punto).

Comparaciones Múltiples Post-Hoc

Una vez que hemos realizado la prueba de ANOVA y hemos visto que el valor de las medias entre las bandas hiperespectrales y las zonas del tumor son iguales, tenemos que comprobar donde se encuentran esas diferencias, para confirmar que media difiere de

queotra podemos utilizar las comparaciones múltiples *post hoc. Este* tipo de comparación me permite controlar la tasa de error Tipo I.

Obtener un valor positivo en la prueba de Anova de comparaciones múltiples, es decir, cuando el valor-p es inferior a 0.05, significa que tenemos diferencias estadísticas significativas. En nuestro caso particular, representa que hay diferencias significativas entre los píxeles de las bandas hiperespectrales y las áreas del tumor. Por esto, para confirmar si realmente todas las bandas, en comparación con todas las áreas del tumor, son diferentes, debemos de usar estas comparaciones para todos los pares de medias seleccionando las comparaciones múltiples Post-Hoc (10).(En el anexo tenemos desarrollado el análisis)

Análisis de Correlación

Cuando se dispone de múltiples variables numéricas, por ejemplo, en problemas de modelado estadístico y *machine learning*, es conveniente estudiar el grado de correlación entre las variables disponibles.

Una forma de hacerlo es mediante matrices de correlación, en las que se muestra el coeficiente de correlación para cada par de variables. Teniendo esto en cuenta, y para identificar si existe una relación lineal entre las bandas de las imágenes hiperespectrales y las diferentes áreas del tumor (sano, tumor, duramadre, vena y arteria) se emplearon medidas de correlación. Dos variables se correlacionan positivamente si, al aumentarlos valores de la primera variable, los niveles de la segunda aumentan; y la correlación es negativa si, en caso contrario, con el descenso de la primera, la segunda disminuye. Para comprobar si existe una correlación entre las variables mencionadas, se calculó la matriz de correlación sobre las 25 bandas hiperespectrales.

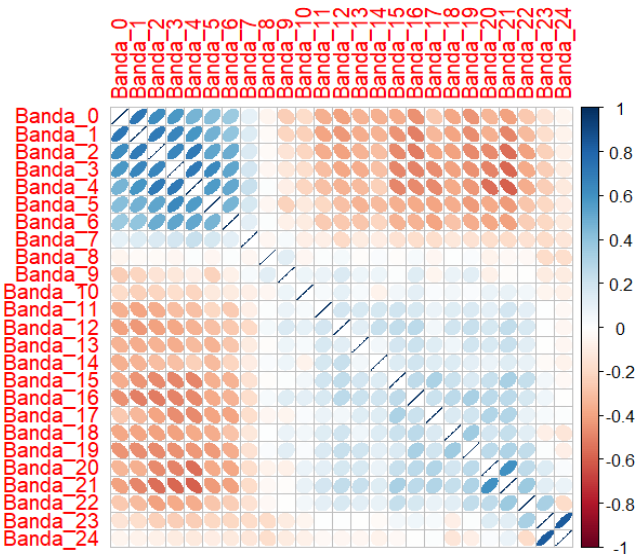


FIGURA 7: *GRAFICO DE CORRELACIÓN*

En la Figura 7, observamos que tenemos varias variables altamente correlacionadas, yaque algunos valores de correlación están por encima de 0.8.

Modelo de clasificación: máquinas de vectores de soporte

En esta sección, se describe cómo se usaron modelos de machine learning paradesarrollar un sistema que clasifique de forma automática los diferentes tipos (áreas)

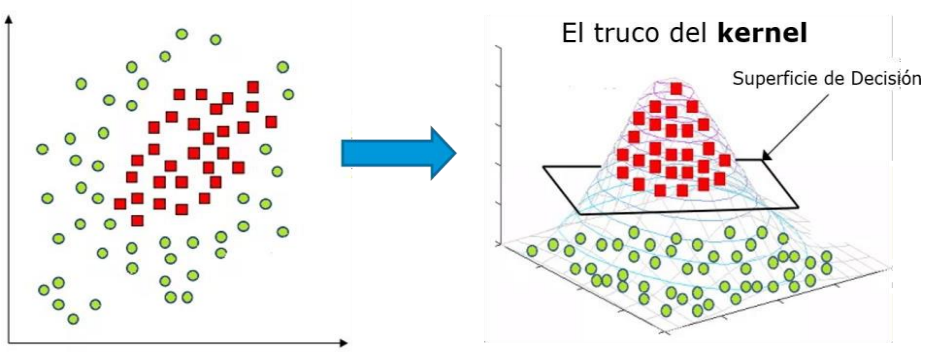


FIGURA 8: MAQUINA DE VECTOR DE SOPORTE CON SEPARACIÓN DE KERNEL (9)

de tejido cerebral. Se implementó principalmente un modelo SVM a través de librerías de Python (*scikit-learn*) y notebooks de *Jupyter* (11).

Las SVM son un tipo de algoritmo de aprendizaje automático supervisado, su función es predecir un hiperplano óptimo en un espacio de mayor dimensión (que puede ser no lineal) con el fin de separar los datos en diversas clases. La Figura 8 muestra un ejemplo del mapeo de un espacio a otro de mayor dimensión usando el truco del Kernel. En este trabajo, se usaron diferentes funciones de Kernel para probar la mejor clasificación multiclase. -

Empezaremos eligiendo la variable de interés que en nuestro caso es la variable zona que indica las áreas con diferentes tipos de tejido (que puede contener las cuatro áreas del tumor).

Se uso un 80% del conjunto de datos para entrenar el modelo SVM y se utilizó el restante 20% de los datos como conjunto de prueba.

Para evaluar el desempeño del modelo y escoger el mejor modelo posible se usó validación cruzada con 5 folds. Además, se realizó una búsqueda de hiperpárametros (por ejemplo, el tipo de Kernel y el parámetro C) con la función de `gridSearch` de *scikit-learn*. El Kernel elegido por la función seleccionó el kernel lineal. Se calcularon diversas métricas de desempeño de clasificación como la matriz de confusión (ver Capítulo de Resultados y Discusión).

Resultados

Los datos con los que se trabajó en este TFM provienen de 6 imágenes hiperespectrales ya procesadas. Estas imágenes son un *ground truth* (conjunto de datos verdaderamente etiquetado), porque conocemos con exactitud el tipo de tejido para cada píxel que la componen. De las imágenes *ground truth* no conocemos todos los píxeles de la imagen hiperespectral, solo una parte de ellos. A continuación, se presentan los resultados obtenidos empezaremos mencionando los resultados de los métodos estadísticos aplicados.

TABLA 5: RESULTADOS DEL ANOVA MIXTO

Columna1	Effect	DFn	DFd	F	p	p<0,05	ges
	Zona	2.26	140.10	992.831	1.11e-86	*	0,023
Paciente 1	bandas	3.76	853.34	2.360.172	0.00e+00	*	0.95
	Zona:bandas	72.00	4464.00	298.727	0.00e+00	*	0.78
	Zona	4.00	3782.00	1.152.612	0	*	0.00077
Paciente 2	bandas	13.09	49520.37	8.754.018	0	*	0.69800
	Zona:bandas	52.37	49520.37	1.013.160	0	*	0.51700
	Zona	3.00	7053.0	100.983	5.3e-64	*	8.54e-05
Paciente 3	bandas	8.62	60819.6	2.386.263	0.0e+00	*	2.52e-01
	Zona:bandas	25.87	60819.6	607.343	0.0e+00	*	2.05e-01
	Zona	4.00	2083.00	1.077.646	0	*	0.006
Paciente 4	bandas	15.50	32276.29	1.663.426	0	*	0.443
	Zona:bandas	61.98	32276.29	511.485	0	*	0.495
	Zona	4.00	3816.00	3.260.774	0	*	0.003
Paciente 5	bandas	16.31	62235.21	3.543.981	0	*	0.481
	Zona:bandas	65.24	62235.21	468.764	0	*	0.329
	Zona	4.00	3362.00	4.407.446	0	*	0.009
Paciente 6	bandas	13.79	46345.34	5.583.881	0	*	0.624
	Zona:bandas	55.14	46345.34	1.190.192	0	*	0.586

La Tabla 5 muestra el análisis anova de dos vías mixto y el anova factorial no paramétrico realizados, obteniéndose un valor-p para los 6 pacientes inferior a 0.05. Esto indica que las diferencias estadísticas significativas en todos los pacientes en las 25 bandas hiperespectrales, en comparación de las áreas tumor, son independientes entre sí y no están relacionadas entre ellas. A continuación, se explica los gráficos respectivos de los 6 pacientes estudiados.

Comparación entre medias

Como podemos ver en el gráfico de violín en la Figura 9, para el paciente 1, las medias entre las diferentes áreas del tumor son diferentes, y hay diferencias significativas entre

las 4 áreas del tumor y las 25 bandas hiperespectrales. La media menor correspondiente a Tumor es de 0.80, seguidamente de la de zona Sano con un valor de 0.89, mientras en vena hay un valor 1.02 de media, y, por último, el tejido de Arteria tiene un valor de

1.27. Estos resultados indican que las cuatro zonas del tumor son completamente diferentes con respecto a las 25 bandas hiperespectrales.

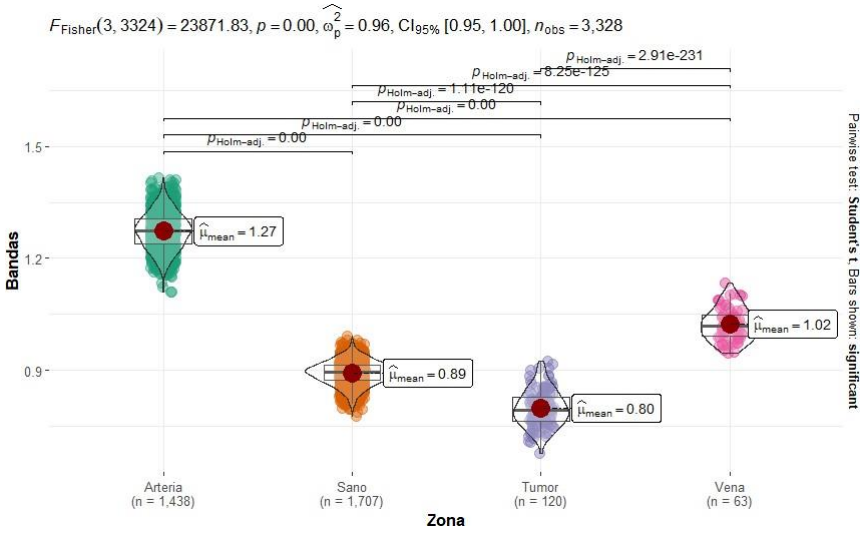


FIGURA 9: GRAFICO DE VIOLÍN PERTENECIENTE AL PACIENTE 1

En el paciente 2 los resultados que obtenemos son similares a los obtenidos para el

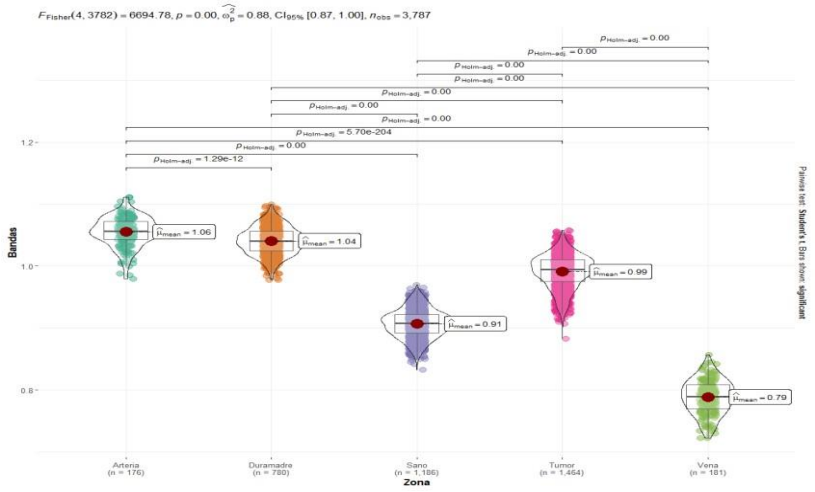


FIGURA 10: *RESULTADO DEL PACIENTE 2*

paciente 1. En este caso, hay diferencias significativas de las medias entre las 25 bandas hiperespectrales y las áreas del tumor. La Figura 9 muestra que hay una media en la zona de Vena de un valor de 0.79, seguido por Sano con un valor de 0.91, el área de Tumor presenta una media con un valor de 0.99, mientras el tejido Duramadre tiene un valor de 1.04 y finalmente, la zona de arteria presenta un valor de 1.06. Esto indica que las 25 bandas hiperespectrales son independientes con respecto a las 5 áreas de tumor para el caso del paciente 2.

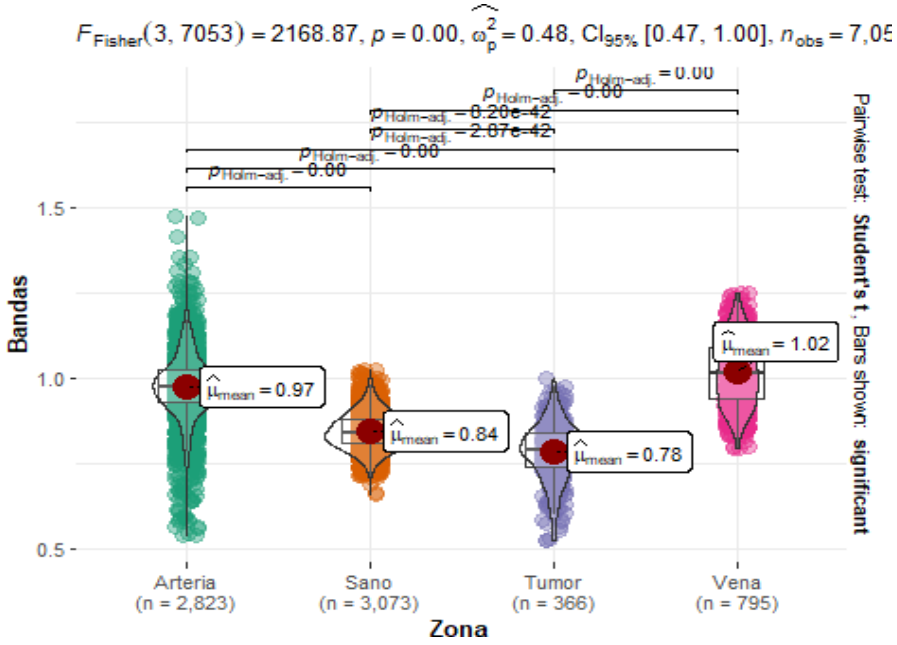


FIGURA 11: *RESULTADO DEL PACIENTE 3*

La Figura 11 muestra el diagrama de violín para el paciente 3, en donde se observa que hay diferencias estadísticas significativas entre las 25 bandas hiperespectrales. A través de ésta última figura, se puede ver que la zona de Tumor tiene un valor de media de 0.78, seguido por la zona Sano con un valor de media de 0.84, la zona de Arteria presenta un valor de 0.97 y por último, Vena con un valor de media de 1.02. Hasta el momento todos los pacientes se comportan de forma similar, es decir tenemos claras evidencias estadísticas que las bandas hiperespectrales y las áreas tumorales son independientes y no están relacionados entre ellos.

La Figura 12 muestra los resultados del paciente 4, confirmándose el comportamiento estadístico similar en los 6 pacientes y obteniéndose diferencias significativas estadísticas entre las 25 bandas hiperespectrales con respecto a las cinco áreas tumorales. Se observa que la zona de Vena tiene un valor de media de 0.75, seguido de la zona de Sano con un valor de media de 0.91, mientras la zona de Tumor tiene 0.984, la zona de Arteria obtiene un valor medio de 1.16 y finalmente, la zona de Duramadre presenta un valor medio de 1.20. Este resultado es similar a los 3 pacientes anteriores donde podemos declarar las diferencias ente las 25 bandas hiperespectrales respecto a las áreas tumorales.

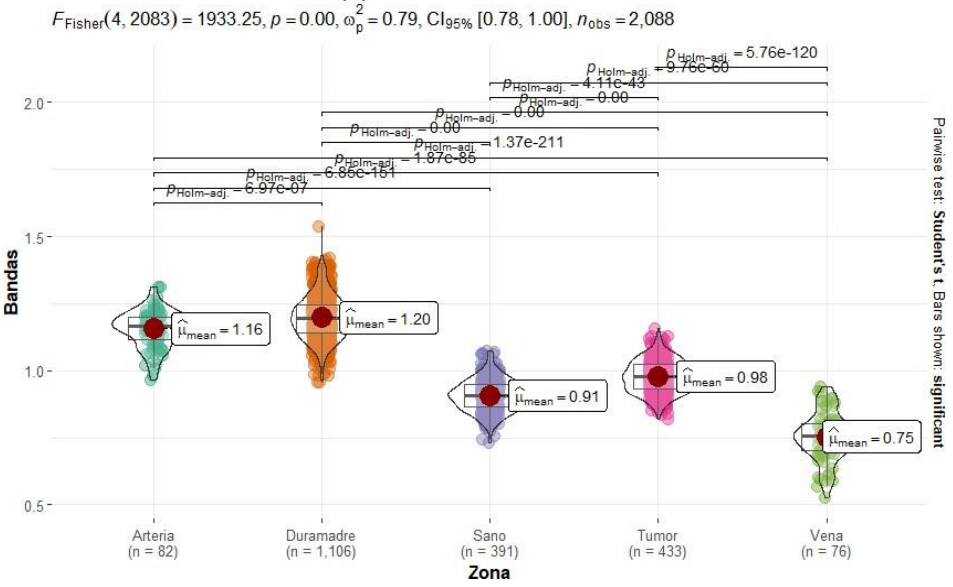


FIGURA 12: *RESULTADO DEL PACIENTE 4*

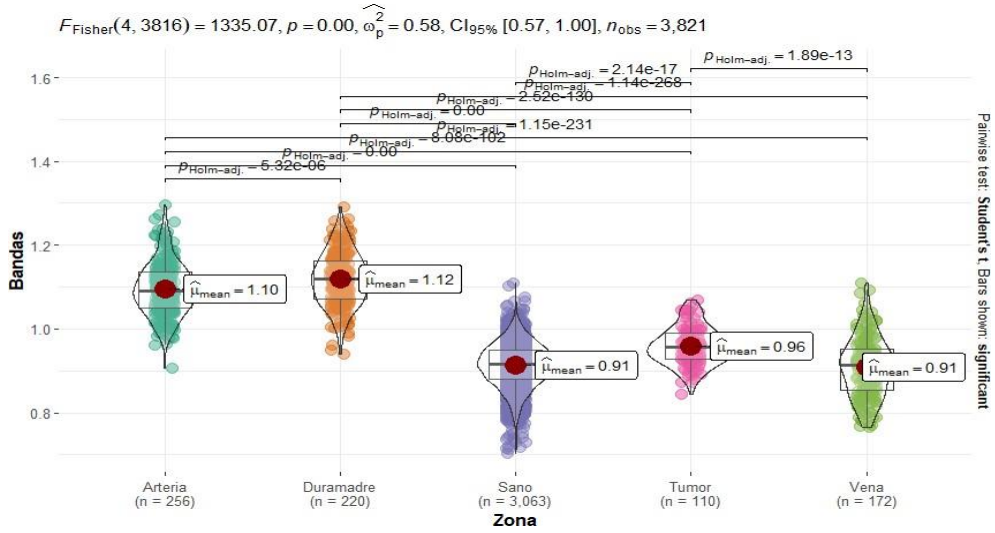


FIGURA 13: RESULTADO PACIENTE 5

En el paciente 5 se observa que la media de la zona vena y la media de la zona sano coincide. De forma similar, la Figura 13 muestra al paciente 5, en donde se observa un valor de media igual entre las áreas de Vena y Sano ambas con un valor de 0.91, seguido de la zona de Tumor con un valor de 0.96, mientras la zona de Arteria presenta un valor de 1.10 de media y, por último, la zona Duramadre con un valor de 1.12. Los resultados anteriores indican que hay diferencias significativas entre las 5 áreas, pero en este

paciente en concreto, no hay diferencias estadísticas significativas entre las áreas de Vena y de Sano, ya que su valor es idéntico.

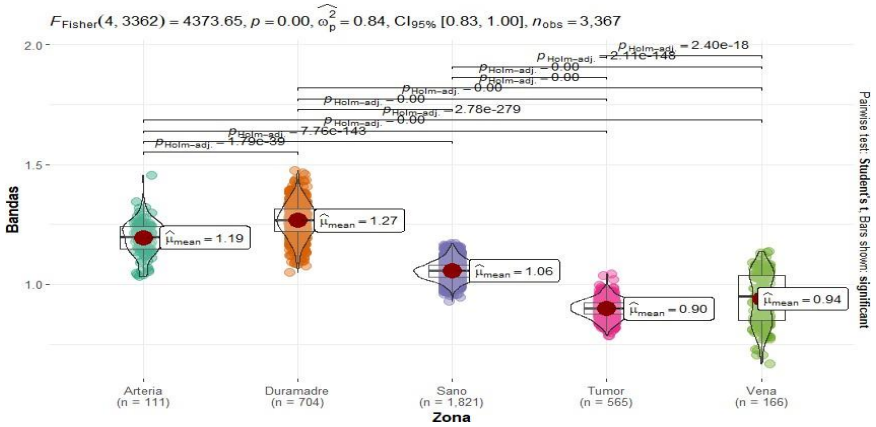


FIGURA 14: *RESULTADO DEL PACIENTE 6*

Finalmente, para los resultados del paciente 6 que se muestran en la Figura 14, se observa que su comportamiento es similar a los primeros cuatro pacientes. Podemos ver claramente que este paciente presenta diferencias significativas en las 5 áreas tumorales: la zona de Tumor tiene un valor de media 0.90, a continuación, la zona de Vena tiene un valor de media de 0.94, la zona Sano presenta un valor de media de 1.06. La zona de Arteria tiene un valor de 1.19 y, por último, la zona de Duramadre presenta un valor de 1.27. se puede declarar que en este paciente las 5 áreas son completamente independientes a las 25 bandas hiperespectrales.

Resultados de las correlaciones

Para los resultados de las correlaciones en los seis pacientes que se estudian en este trabajo, se mira la asociación entre las 25 bandas hiperespectrales. La forma ideal de medir la relación (lineal) entre varias variables es mediante una matriz de correlación y posteriormente un gráfico de la matriz ayuda a diferenciar de forma más clara el resultado obtenido de las correlaciones. Describiremos a continuación los resultados de los 6 pacientes.

Paciente 1

En el paciente 1, las variables más correlacionadas son:

- La banda 0 junto a la banda 1 tenemos un R^2 de 0.98, banda 0 respecto a banda 2 con un R^2de 0.92
- La banda 1 con respecto a la banda 5 con un R^2 de0.90
- La banda 1 con la banda 3 tenemos un R^2 de 0.91

- La banda 2 con la banda 0 con un R^2 de 0.96,
- Y la banda 3 con la banda 2 con un R^2 de 0.92 (13).

Este paciente tiene bandas altamente correlacionadas. Inicialmente, cada paciente está representada por una imagen en la que tenemos etiquetadas cuatro o cinco zonas de

un tumor neural, y como no se deben de eliminar variables, eliminaría este paciente del estudio de machine learning.

Paciente 2

En el paciente 2 tenemos pocas bandas que estén altamente correlacionadas, estasbandas son:

- La banda 0 con respecto a la banda 1 con un R^2 de 0.91
- La banda 1 con respecto a la banda 2 con un R^2 de 0.9
- La banda 2 con respecto a la banda 0 con un R^2 de 0.87, y;
- La banda 23 con respecto a la banda 24 con un R^2 de 0.9

Cómo se observó en el paciente 1, el paciente está siendo representado por una imagen y no podemos eliminar las bandas ya que estamos eliminando información de los píxeles. Por lo tanto, no es recomendable eliminar bandas con el fin de obtener una mejor predicción.

Paciente 3

En el paciente 3 tenemos pocas correlaciones entre las bandas. Están correlacionadas la banda 0 con respecto a la banda 1 con un valor de R^2 de 0.88 y la banda 1 con respecto a la banda 3 con un R^2 de 0.89. Como ya se ha comentado anteriormente, no es conveniente eliminar las bandas hiperespectrales para construir el modelo de clasificación automática, ya que estas bandas aportan información relevante para la imagen hiperespectral.

Paciente 4

Para el paciente 4, se tienen varias bandas correlacionadas:

- La banda 0 con respecto a la banda 1 con un R^2 de 0.82
- La banda 1 con la banda 2 con un R^2 de 0.88
- La banda 3 con respecto a la banda 1 con un R^2 de 0.85
- La banda 3 con la banda 2 con un R^2 0.87; y,
- La banda 34 con la banda 24 con un R^2 de 0.82

No es conveniente eliminar las variables (bandas) para los algoritmos de clasificación como se ha venido recalcando.

Paciente 5

En este paciente la correlación más alta pertenece a la banda 23 con la banda 24 con un R^2 de 0.84.

Paciente 6

En el paciente 6 las bandas que están más correlacionadas son:

- La banda 0 con respecto a la banda 1 con un R^2 de 0.93
- La banda 1 con la banda 3 con un R^2 de 0.91
- La banda 2 con la banda 4 con un R^2 de 0.83,
- La banda 23 con la banda 24 con un R^2 de 0.87.

Resultados de la máquina de vectores de soporte

A continuación, expondremos los resultados del algoritmo de machine learning de SVM.

Se han usado 6 pacientes con tumores cerebrales que fueron extirpados en el quirófano, se utiliza este algoritmo, porque las SVM se pueden usar como un modelo de clasificación automática, y el interés de este trabajo es intentar clasificar las estructuras de la imagen que consta de 4 o 5 etiquetas (sano, tumor, vena, arteria y duramadre).

Una vez se ha construido el modelo SVM, se procede a evaluar el desempeño sobre el conjunto de datos de prueba (test set). En la Table 6, se muestran los resultados de la exactitud (accuracy) para cada uno de los pacientes. Es importante aclarar, que en este caso se usaron únicamente los datos propios de cada paciente para entrenar (y predecir) los tipos de tejido. Por esta razón, la exactitud es mayor, ya que existe un sesgo añadido porque toda la información (valores de píxel) pertenece a la misma muestra del mismo paciente.

TABLA 6: *DATOS DEL EXACTITUD DE LOS PACIENTES*

Máquina de Vector Soporte	
Pacientes	Exactitud (Accuracy)
Paciente 1	0.99
Paciente 2	1.00
Paciente 3	0.82
Paciente 4	0.99
Paciente 5	0.83
Paciente 6	0.99

En el estudio de los 6 pacientes la exactitud oscila entre un 83% y un 100%. También, se calcularon las matrices de confusión correspondientes a la evaluación del modelo SVM sobre los 6 pacientes con el fin de observar los aciertos y los errores de este algoritmo de clasificación.

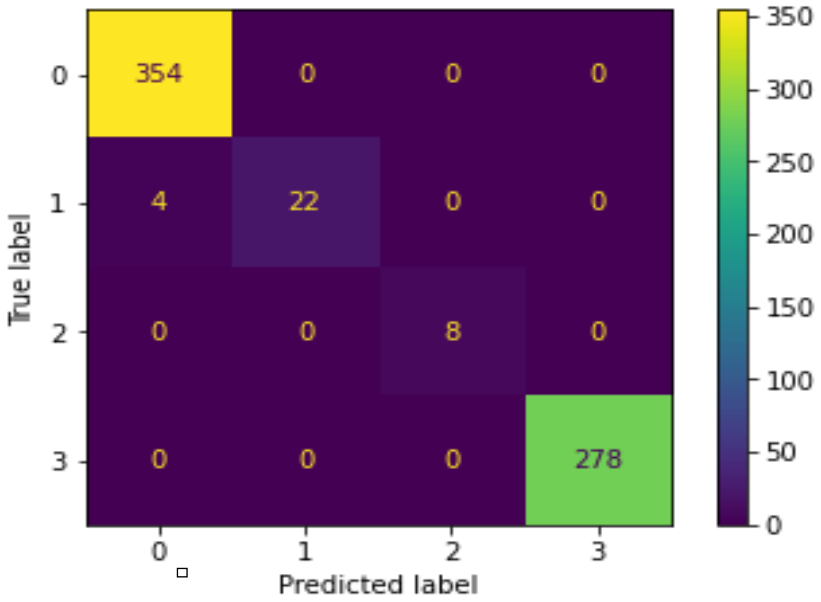


FIGURA 15: MATRIZ DE CONFUSIÓN DEL PACIENTE 1

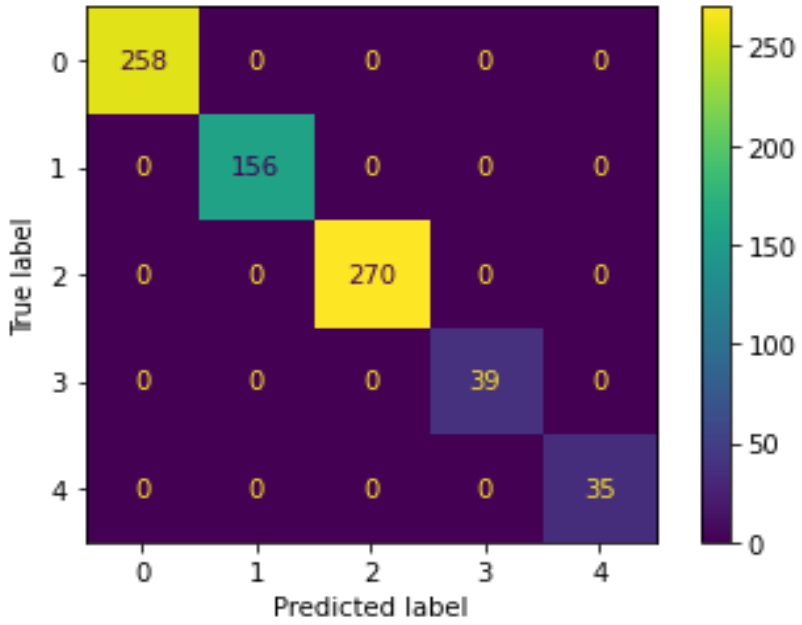


FIGURA 16: *MATRIZ DE CONFUSIÓN DEL PACIENTE 2*

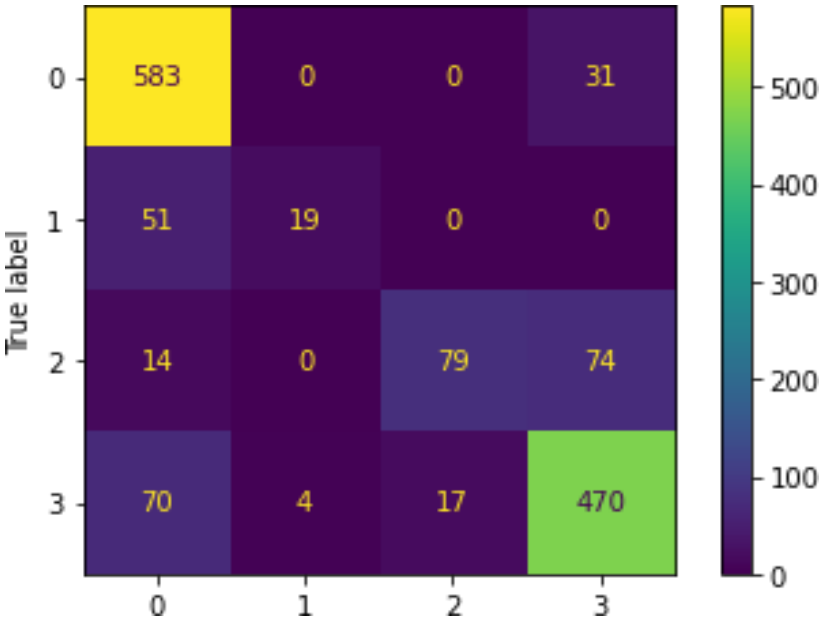


FIGURA 17: *MATRIZ DE CONFUSIÓN PACIENTE 3*

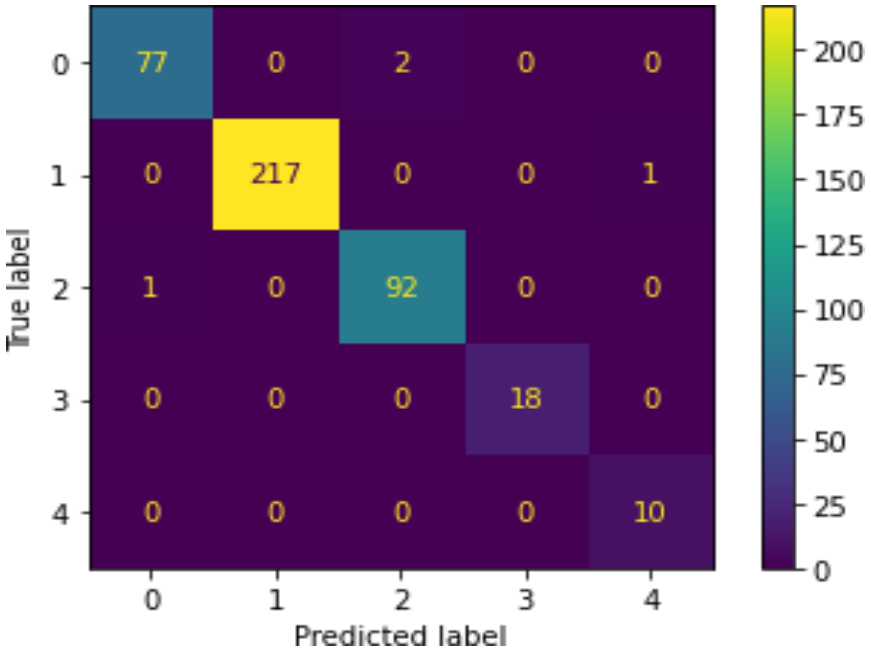


FIGURA 18: MATRIZ DE CONFUSIÓN DEL PACIENTE 4

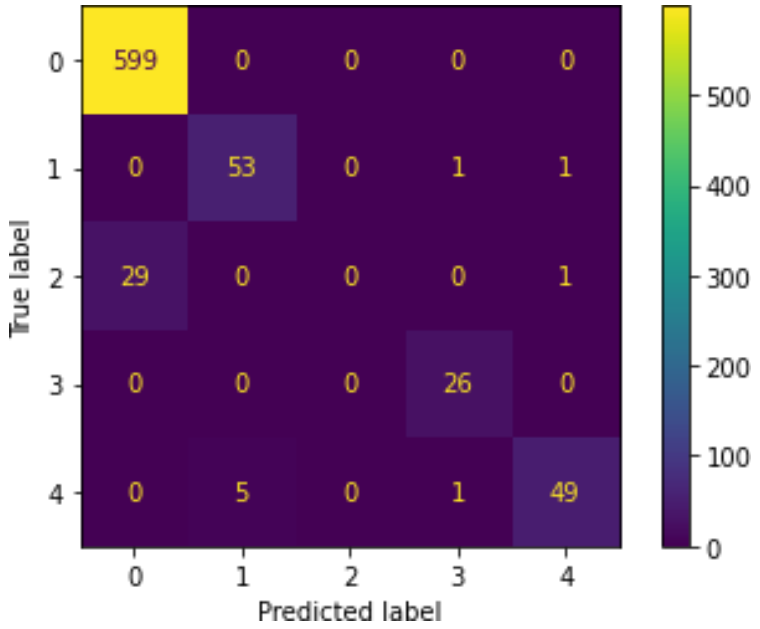


FIGURA 19: *MATRIZ DE CONFUSIÓN PACIENTE 5*

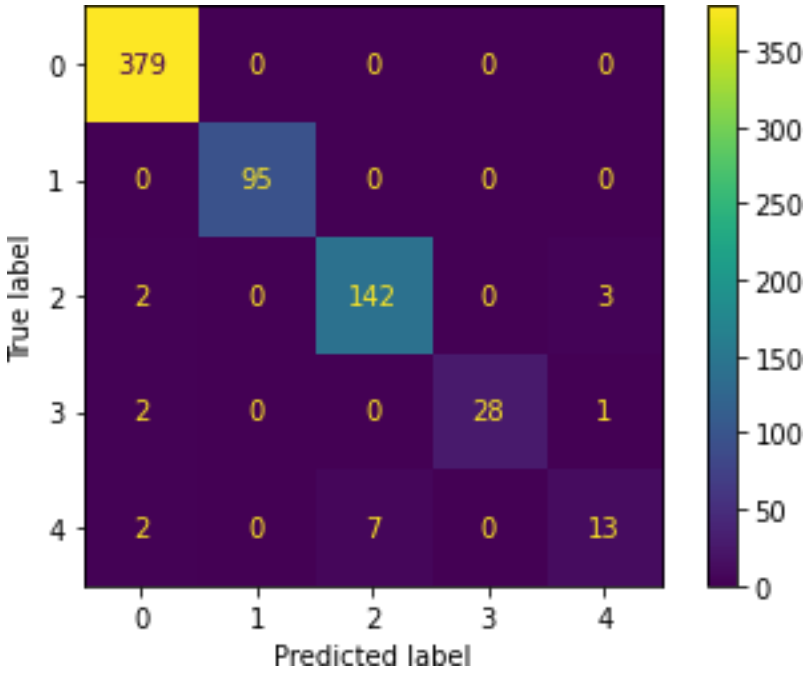


FIGURA 20: *MATRIZ DE CONFUSIÓN PACIENTE 6*

Discusión:

En este trabajo de fin de master se ha realizado un modelo de machine learning con las máquinas de vector soporte, con el fin de poder clasificar las diferentes áreas de tumores neurales, en este trabajo además se ha profundizado en la inferencia estadística entre los píxeles de las 25 bandas hiperespectrales en relación a las diferentes áreas del tumor neural, se realizan pruebas de inferencias estadísticas, para comprobar si las medias de los pixeles de las 25 bandas hiperespectrales están relacionadas con las áreas del tumor neural, y miramos si las bandas hiperespectrales están relacionadas entre ellas. La finalidad de este trabajo como ya hemos comentado consiste en usar el algoritmo de machine learning de la máquina de vector soporte, para predecir las diferentes áreas del tumor neural, con este algoritmo podemos predecir si la imagen que tenemos el algoritmo clasifica correctamente cuales son las áreas tumorales que tenemos y la extensión de cada una y así el neurocirujano podrá resecar mejor el área tumoral, evitando lesiones, o en el caso en que no se puedan evitar lesiones, resecar la menor zona para así dejar la menor secuela posible. Primero se realizaron análisis de inferencia estadísticas donde miramos como se comportaban las medias con la desviación estándar, seguidamente realizamos la anova mixto, miraremos si los pixeles de las 25 bandas hiperespectrales están relacionadas entre ellas, y el paso más importante probaremos el algoritmo de machine learning de la máquina de vector soporte. En los resultados que tenemos con relación al anova mixto y al anova factorial no paramétrico, todos los pacientes tienen diferencias entre las bandas excepto en el paciente 5 que observamos que las bandas pertenecientes a las bandas de la zona sana y de la zona vena que tienen el mismo valor, en relación con las correlaciones, vemos que, en 5 pacientes, la correlación entre las bandas es baja, excepto en el paciente 1 que está altamente correlacionas las bandas.

Por ultimo hablaremos del resultado en el algoritmo de predicción de machine learning con las máquinas de vector soporte, como se ha mencionado a lo largo de este trabajo, se usaron el mismo dataset de cada paciente para realizar la predicción, con esto el problema que me encuentro es que las predicciones son muy altas y los errores que comente el algoritmo son muy bajos, esto me indica como menciono en el apartado de trabajo futuro, la mejor opción hubiera sido realizar la predicción añadiendo una nueva etiqueta con el número de paciente y realizando la predicción con los 6 pacientes juntos y así comprobar cómo se comporta el algoritmo. Pero los resultados obtenidos realizando la predicción sobre el propio dataset el realmente bueno, como se menciona en las líneas futuras probare cómo se comporta el algoritmo con los 6 pacientes juntos.

Conclusiones y líneas de futuro

Conclusiones

A lo largo de este Trabajo de Fin de Máster, se han realizado diferentes análisis con la intención de clasificar y de afianzar todos los conocimientos que se han ido adquiriendo no solo durante la realización de este trabajo, sino además de toda la formación recibida a lo largo del estudio de este máster. Se han cumplido los objetivos planeados, aunque debido a la limitación del número de pacientes y al alcance de este trabajo, se ha dejado la implementación del modelo SVM utilizando varios pacientes de forma simultánea para un trabajo futuro.

En este trabajo fin de máster, se han presentado diferentes metodologías desde los análisis estadísticos de 6 pacientes de neurocirugía, pasando por correlaciones para evaluar la asociación entre las 25 bandas hiperespectrales y para desarrollar un modelo de clasificación automática de los tipos de tejidos cerebrales.

En las pruebas inferenciales de Anova de 2 vías y de Anova Factorial hemos podido comprobar la diferencia entre estas 25 bandas hiperespectrales y las cuatro o cinco etiquetas tumorales de las que está compuesta la imagen. Con ello se ha observado que las medias de las 25 bandas son completamente independientes al área tumoral de cada paciente. Mientras que, respecto a las correlaciones, hemos visto que pocas bandas están asociadas entre ellas, excepto para el paciente 1, el cual tiene una gran correlación entre las variables (bandas espectrales).

Con respecto al modelo de máquinas de vectores de soporte los resultados de clasificación han sido excelentes. Sin embargo, esto es posiblemente a que el modelo se ha entrenado con datos del mismo paciente para predecir datos del mismo paciente. Para un trabajo futuro, sería interesante desarrollar un modelo mezclando todos los pacientes y prediciendo pacientes de forma independiente (en un conjunto de prueba). No se ha realizado está implementación, dado el alcance de este trabajo y por la poca cantidad de pacientes disponibles en el momento de ejecución del TFM.

Líneas de futuro

Sería interesante probar modelos más potentes como redes neuronales profundas directamente sobre las imágenes hiperespectrales, no solo para enfermedades neurales, sino para la clasificación de diferentes enfermedades, como, por ejemplo, cáncer de pulmón o cáncer de colon, ya que son tumores sumamente agresivos y con un mal pronóstico. Esta tecnología podría ayudar a los radiólogos a detectar este tipo de patología con anticipación sobre los pacientes y así se podría actuar con mayor brevedad en el tratamiento quirúrgico o farmacológico, especialmente en pacientes con pronóstico muy reservado.

La metodología desarrollada ha sido la adecuada, aunque no se ha podido aumentar el número de algoritmos o incluso mirar variables nuevas pertenecientes a los pacientes de neurocirugía dado el alcance y el tiempo disponible para el desarrollo de este trabajo. Esta nueva información y estos métodos podrían informar sobre la esperanza de vida de estos pacientes respecto a una patología concreta y compleja como es el tumor cerebral.

Glosario

HSI: Imagen hiperespectral

ML: Machine Learning

AI: Inteligencia Artificial

DL: Deep Learning

ACC: Accuracy

SH: Prueba de Shapiro Wiks

AOV; Anova de 2 vías Mixto

COR: Correlación

Bibliografía

1. Selci S. The Future of Hyperspectral Imaging. Journal of Imaging. .
2. Academy k. [Online]. Available from: https://es.khanacademy.org/science/ap-chemistry/electronic-structure-of-atoms-ap/bohr-model-hydrogen-ap/a/light-and-the-electromagnetic-spectrum.
3. LIBRARY SD. [Online]. Available from: ttps://www.spiedigitallibrary.org/journals/journal-of-biomedical-optics/volume-19/issue-01/010901/Medical-hyperspectral-imaging-a-review/10.1117/1.JBO.19.1.010901.full?SSO=1.
4. TELEMATICA. [Online].; 2019. Available from: https://www.telematica.com.pe/envi-imagenes-hiperespectrales-una-herramienta-emergente-para-la-disponibilidad-en-las-misiones/.
5. j.Ricardo MFF. Brain Tumors. The American Journal of Medicine. 2018 Agosto; 131(8).

6. Fabelo , Ortega S, Ravi D, Ravi Kiran B, Sosa C, Bulters D, et al. Spatio-spectral classification of hyperspectral images for brain cancer detection during surgical operations. PLOS ONE. 2018.

7. Urbanos G, Martin A, Vazquez G, Villanueva M, Villa M, Jimenez-Roldan L, et al. Supervised Machine Learning Methods and Hyperespectral Imaging Techniques Jointly Appied for Brain Cancer Classification. Sensors. .

8. Huertas JP. Infeencia estadistica y aproximacion al valor p. Parte I. Revista Española de Podologia. .

9. Albano G, Giorno V, Roman-Roman P, Roman-Roman S, Serrano-Perez JJ, Torres-Ruiz F. Inference on an heterocedastic Gompertz tumor growth model. Mathematical Biosciences. 2020 Julio 28.

10. [Online]. Available from: https://campus.datacamp.com/courses/free-introduction-to-r/chapter-1-intro-to-basics-1?ex=4.

11. Carpenter KA, Huang X. Machine Learning-based Virtual Screening and Its Applications to Alzheimer's Drug Discovery: A Review. Current Pharmaceutical Design. 2018.

12. Heras JM. IArtifical.net. [Online].; 2019. Available from: https://www.iartificial.net/maquinas-de-vectores-de-soporte-svm/.

13. Gonzalez DC. Calculo de la potencia de una lente intraocular mediante tecnicas de machine learning. Trabajo Fin de Master. Barcelona: UOC; 2018. Report No.: 43.

Anexos:

Anexo 1: Código de 1 de los pacientes, exponemos el análisis completo de uno de los 6 pacientes, ya que el tratamiento de los datos se ha realizado de la misma forma.

Dataset del paciente número 3

Introducimos el archivo csv dónde encontramos los píxeles en las 25 bandas hiperespectrales y sus áreas tumorales que pertenecen a las etiquetas.

```
paciente_3 <- read.csv("mi_Paciente_3.csv")

head(paciente_3)
```

	X <int>	Banda_0 <dbl>	Banda_1 <dbl>	Banda_2 <dbl>	Banda_3 <dbl>	Banda_4 <dbl>	Banda_5 <dbl>	Banda_6 <dbl>	Banda_7 <dbl>
1	0	0.7830897	0.9057773	0.8835342	1.0127974	1.0644996	1.058529	1.049941	1.0573903
2	1	0.7592152	0.8943781	0.9487288	0.9806058	1.0586792	1.040291	1.048497	0.9589957
3	2	0.6952866	0.8199529	0.9158324	1.0114678	0.9688775	1.047099	1.040481	1.0671906
4	3	0.8043979	0.7821254	0.9019680	0.9609478	1.0738894	1.056217	1.051217	0.9918915
5	4	0.7739124	0.8540516	0.9698070	0.9840448	1.0120853	1.060073	1.018433	1.1048777
6	5	0.8543717	0.9052514	0.9777586	1.0183298	1.0282095	1.138200	1.046582	1.0939791

6 rows | 1-10 of 29 columns

Importante, cambiaremos nuestros datos a formato largo y así podré usar las pruebas estadísticas del anova de 2 vías mixto, ya que tendré 2 factores, un factor dependiente que son mis 25 bandas, y un factor independiente que son las áreas tumorales, calculamos la media con la desviación estándar, así comprobamos cómo se comportan los datos

```
#install.packages("dplyr")
library(dplyr)
paciente_3_long<- Paciente_3 %>%
  gather(key = "Banda", value = "medida",
         Banda_0,Banda_1, Banda_2, Banda_3, Banda_4, Banda_5, Banda_6, Band
a_7, Banda_8,Banda_9,Banda_10,
Banda_11, Banda_12, Banda_13, Banda_14, Banda_15, Banda_16, Banda_17, Banda
_18, Banda_19, Banda_20,
Banda_21, Banda_22, Banda_23, Banda_24) %>%
  convert_as_factor(X, Zona)
head(paciente_3_long)

##   X Zona   Banda    medida
## 1 0 Sano Banda_0 0.7830897
## 2 1 Sano Banda_0 0.7592152
## 3 2 Sano Banda_0 0.6952866
## 4 3 Sano Banda_0 0.8043979
## 5 4 Sano Banda_0 0.7739124
## 6 5 Sano Banda_0 0.8543717
```

Estadísticos básicos

Sacamos la media y la desviación estándar del dataframe en formato largo, con estos datos podré ver cómo se distribuyen los datos.

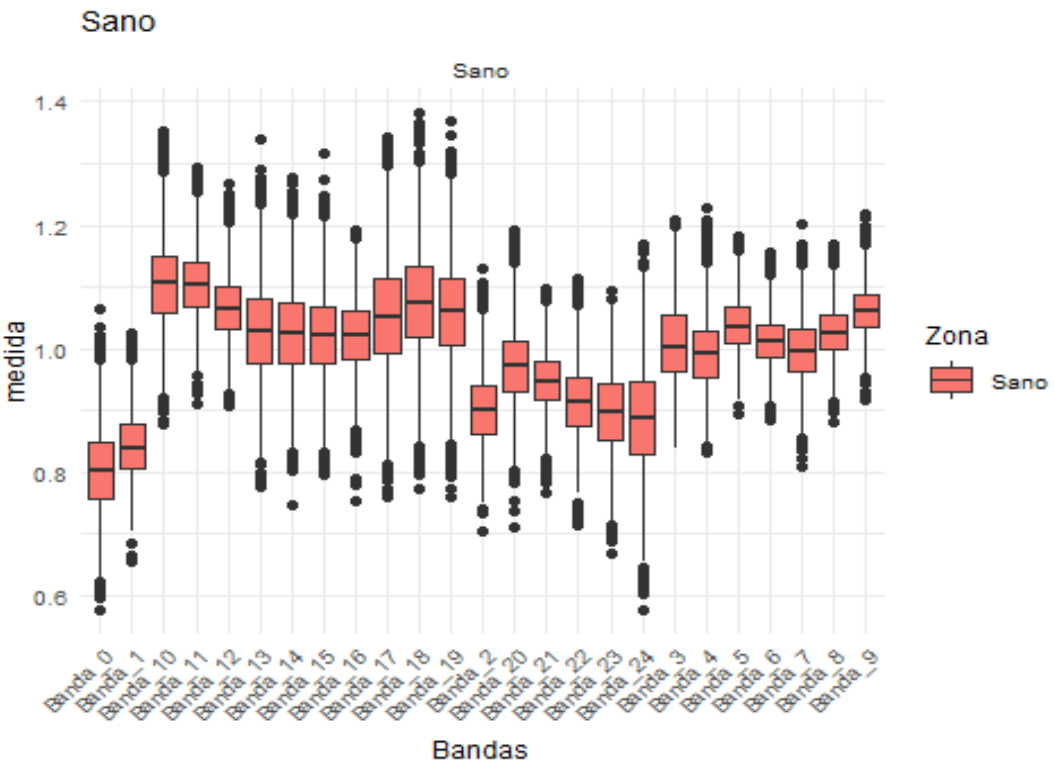
```
paciente_3_long %>%
group_by(Banda, Zona) %>%
get_summary_stats(medida, type = "mean_sd")

## # A tibble: 100 x 6
##    Zona    Banda    variable     n  mean    sd
##    <fct>   <chr>    <chr>    <dbl> <dbl> <dbl>
```

```
##  1 Arteria Banda_0  medida    2823 0.953 0.129
##  2 Sano    Banda_0  medida    3073 0.804 0.067
##  3 Tumor   Banda_0  medida     366 0.739 0.105
##  4 Vena    Banda_0  medida     795 0.987 0.115
##  5 Arteria Banda_1  medida    2823 0.973 0.095
##  6 Sano    Banda_1  medida    3073 0.843 0.052
##  7 Tumor   Banda_1  medida     366 0.783 0.082
##  8 Vena    Banda_1  medida     795 1.02  0.096
##  9 Arteria Banda_10 medida    2823 1.08  0.09
## 10 Sano    Banda_10 medida    3073 1.11  0.07
## # ... with 90 more rows
```

Gráfico Boxplot

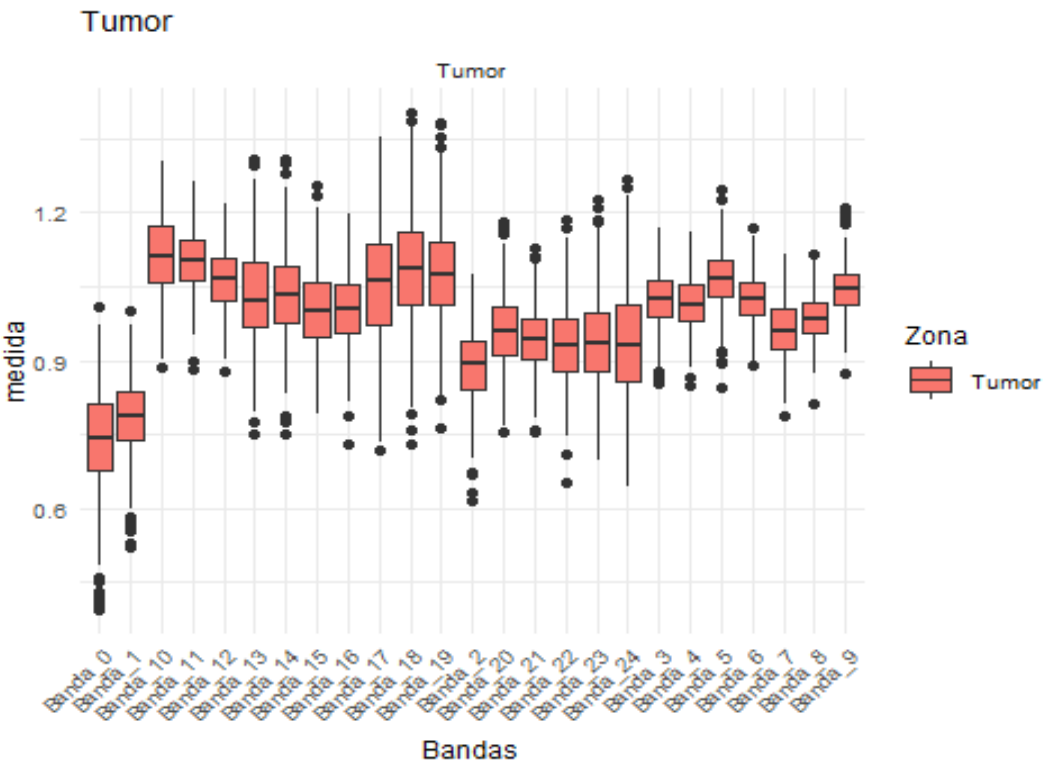
Empezamos graficando las características de la variable Zona seleccionando la zona sana del tumor, con estos gráficos podremos ver cómo se comporta la media con la desviación estándar

```
plotdata <- dplyr::filter(paciente_3_long,
                          Zona == "Sano")
ggplot(plotdata, aes(x = Banda, y = medida)) +
geom_boxplot(aes(fill = Zona), width = 0.8) +
  facet_wrap(~Zona) +
  theme_minimal(base_size = 9) +
  theme(axis.text.x = element_text(angle = 45,
                                   hjust = 1)) +
  labs(title = "Sano",
       x = "Bandas",
       y = "medida")
```



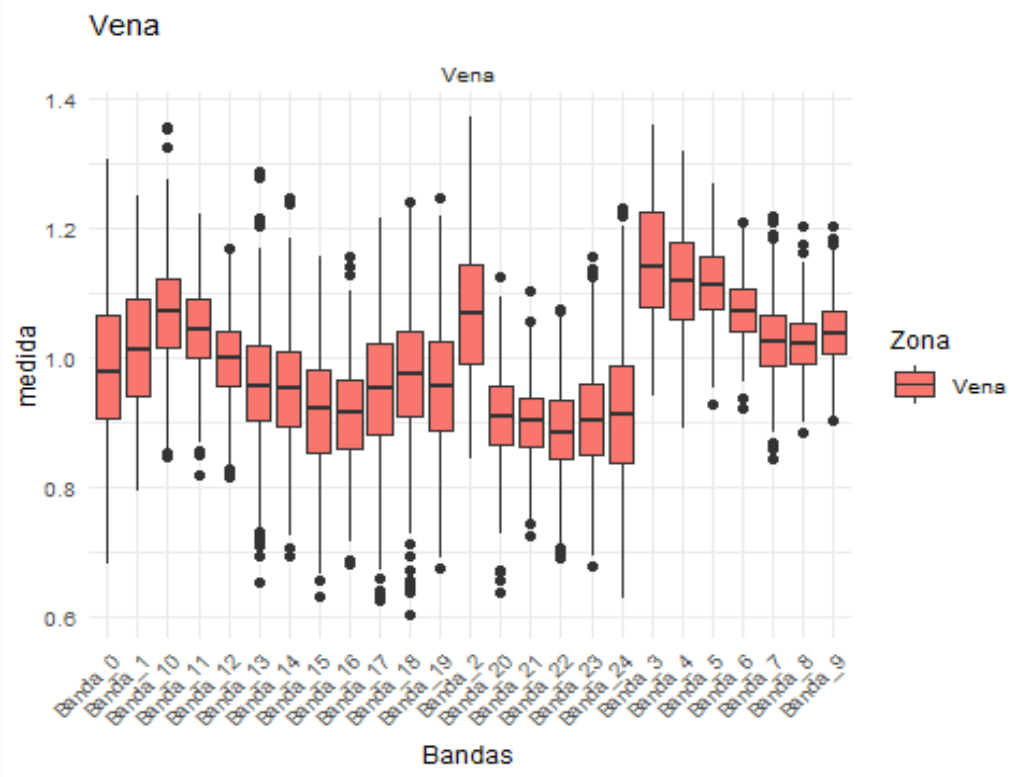
Hacemos lo mismo con las otras zonas:

```
plotdata <- dplyr::filter(paciente_3_long,
                          Zona == "Tumor")
ggplot(plotdata, aes(x = Banda, y = medida)) +
geom_boxplot(aes(fill = Zona), width = 0.8) +
  facet_wrap(~Zona) +
  theme_minimal(base_size = 9) +
  theme(axis.text.x = element_text(angle = 45,
                                   hjust = 1)) +
  labs(title = "Tumor",
       x = "Bandas",
       y = "medida")
```



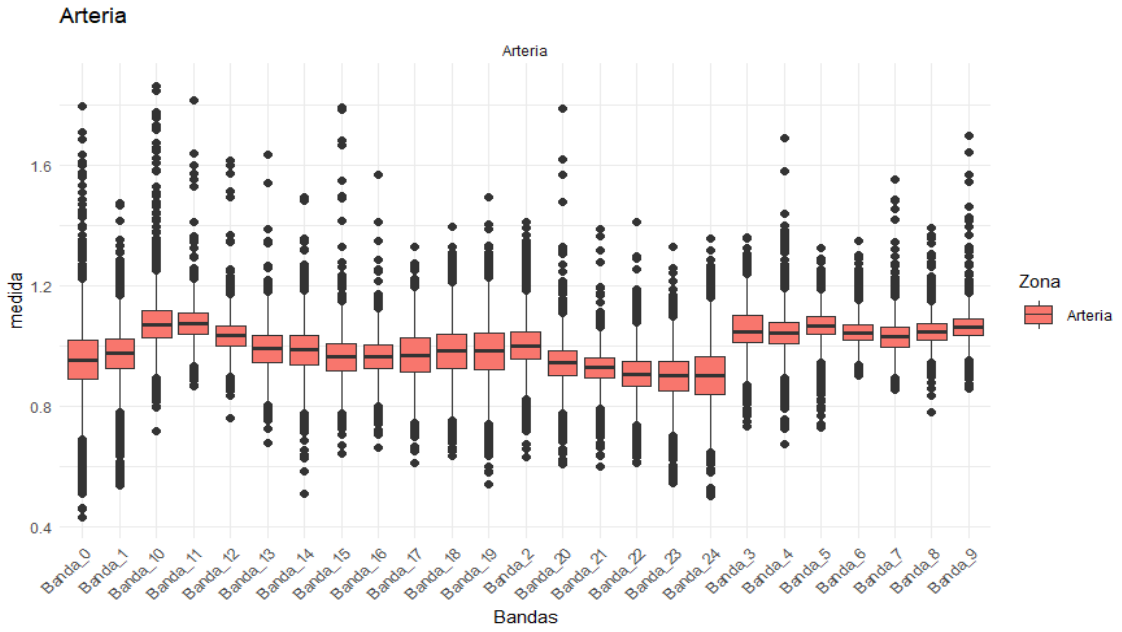
Seguidamente graficamos la zona de Vena

```
plotdata <- dplyr::filter(paciente_3_long,
                          Zona == "Vena")
ggplot(plotdata, aes(x = Banda, y = medida)) +
geom_boxplot(aes(fill = Zona), width = 0.8) +
  facet_wrap(~Zona) +
  theme_minimal(base_size = 9) +
  theme(axis.text.x = element_text(angle = 45,
                                   hjust = 1)) +
  labs(title = "Vena",
       x = "Bandas",
       y = "medida")
```



Por último, graficamos la zona de arteria

```
plotdata <- dplyr::filter(paciente_3_long,
                          Zona == "Arteria")
ggplot(plotdata, aes(x = Banda, y = medida)) +
geom_boxplot(aes(fill = Zona), width = 0.8) +
  facet_wrap(~Zona) +
  theme_minimal(base_size = 9) +
  theme(axis.text.x = element_text(angle = 45,
                                   hjust = 1)) +
  labs(title = "Arteria",
       x = "Bandas",
       y = "medida")
```



Comprobación de outlier

Vamos a comprobar sí tenemos outliers, en algunas webs recomiendan no quitarlo, ya que al quitar los outlier, varia tanto la media como la mediana y eso puede afectar a la Inferencia, la información aparece en el siguiente link. https://www.maximaformacion.es/blog-dat/como-lidiar-con-los-datos-atipicos-outliers/

```
paciente_3_long %>%
  group_by(Banda, Zona) %>%
  identify_outliers(medida)
```

```
## # A tibble: 2,903 x 6
##    Zona    Banda   X     medida is.outlier is.extreme
##    <fct>   <chr>   <fct>  <dbl> <lgl>      <lgl>
##  1 Arteria Banda_0 36      1.56 TRUE       TRUE
##  2 Arteria Banda_0 38      1.23 TRUE       FALSE
##  3 Arteria Banda_0 41      1.23 TRUE       FALSE
##  4 Arteria Banda_0 50      1.26 TRUE       FALSE
##  5 Arteria Banda_0 58      1.23 TRUE       FALSE
##  6 Arteria Banda_0 82      1.29 TRUE       FALSE
##  7 Arteria Banda_0 83      1.27 TRUE       FALSE
##  8 Arteria Banda_0 96      1.24 TRUE       FALSE
##  9 Arteria Banda_0 109     1.23 TRUE       FALSE
## 10 Arteria Banda_0 114     1.31 TRUE       FALSE
## # ... with 2,893 more rows
```

Verificamos los supuestos de normalidad

Vamos a realizar la normalidad con la prueba de shapiro Wilks, es una prueba bastante potente, y como el dataset al estar en formato largo, puedo usarlo ya que el número máximo de observaciones, para realizar dicha prueba es de 3000 observaciones.

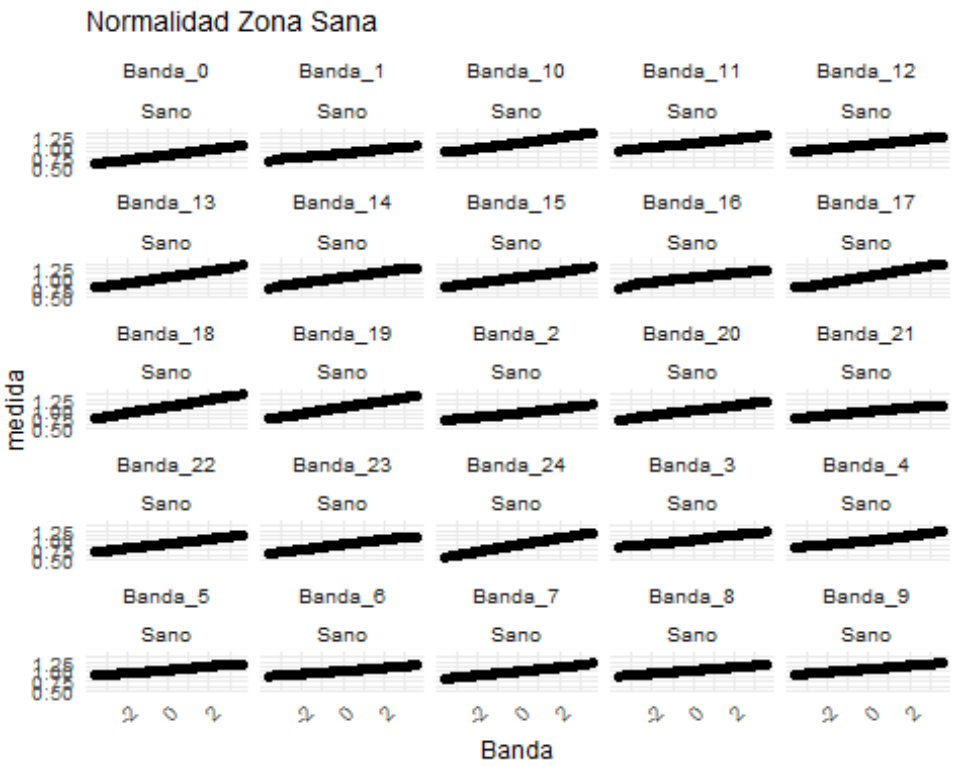
```
paciente_3_long %>%
group_by(Banda, Zona) %>%
shapiro_test(medida)
```

```
## # A tibble: 100 x 5
##    Zona    Banda    variable statistic        p
##    <fct>   <chr>    <chr>        <dbl>    <dbl>
##  1 Arteria Banda_0  medida       0.950 6.14e-30
##  2 Sano    Banda_0  medida       0.999 2.94e- 2
##  3 Tumor   Banda_0  medida       0.984 5.46e- 4
##  4 Vena    Banda_0  medida       0.992 3.77e- 4
##  5 Arteria Banda_1  medida       0.959 2.09e-27
##  6 Sano    Banda_1  medida       0.996 2.60e- 7
##  7 Tumor   Banda_1  medida       0.982 1.86e- 4
##  8 Vena    Banda_1  medida       0.986 9.78e- 7
##  9 Arteria Banda_10 medida       0.840 8.06e-47
## 10 Sano    Banda_10 medida       0.998 1.41e- 3
## # ... with 90 more rows
```

Graficos QQ-Plot (confirmar normalidad)

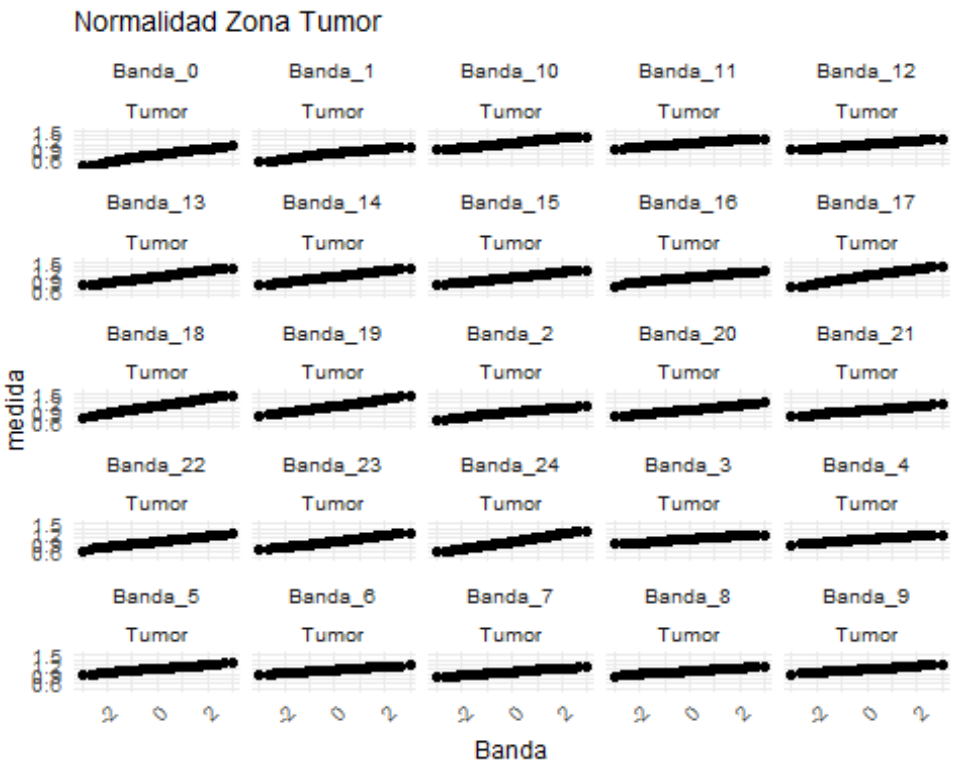
Empezamos graficando la normalidad de la Zona selecionando la zona sana del tumor, con este gráfico podremos ver cómo se comportan los datos en esa parte del tumor.

```
plotdata <- dplyr::filter(paciente_3_long,
                          Zona == "Sano")
ggqqplot(plotdata, "medida", ggtheme = theme_bw()) +
  facet_wrap(Banda ~ Zona) +
  theme_minimal(base_size = 9) +
  theme(axis.text.x = element_text(angle = 45,
                                   hjust = 1)) +
  labs(title = "Normalidad Zona Sana",
       x= "Banda",
       y= "medida")
```



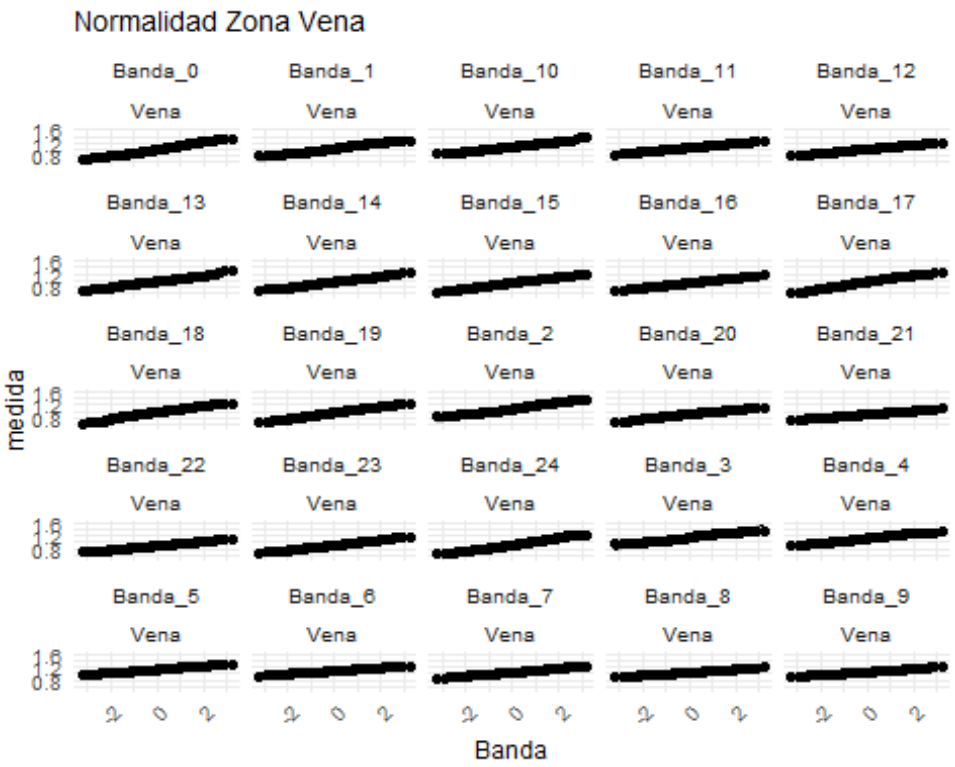
El siguiente grafico qqplot es la zona tumoral:

```
plotdata <- dplyr::filter(paciente_3_long,
                          Zona == "Tumor")
ggqqplot(plotdata, "medida", ggtheme = theme_bw()) +
  facet_wrap(Banda ~ Zona) +
  theme_minimal(base_size = 9) +
  theme(axis.text.x = element_text(angle = 45,
                                   hjust = 1)) +
  labs(title = "Normalidad Zona Tumor",
       x= "Banda",
       y= "medida")
```



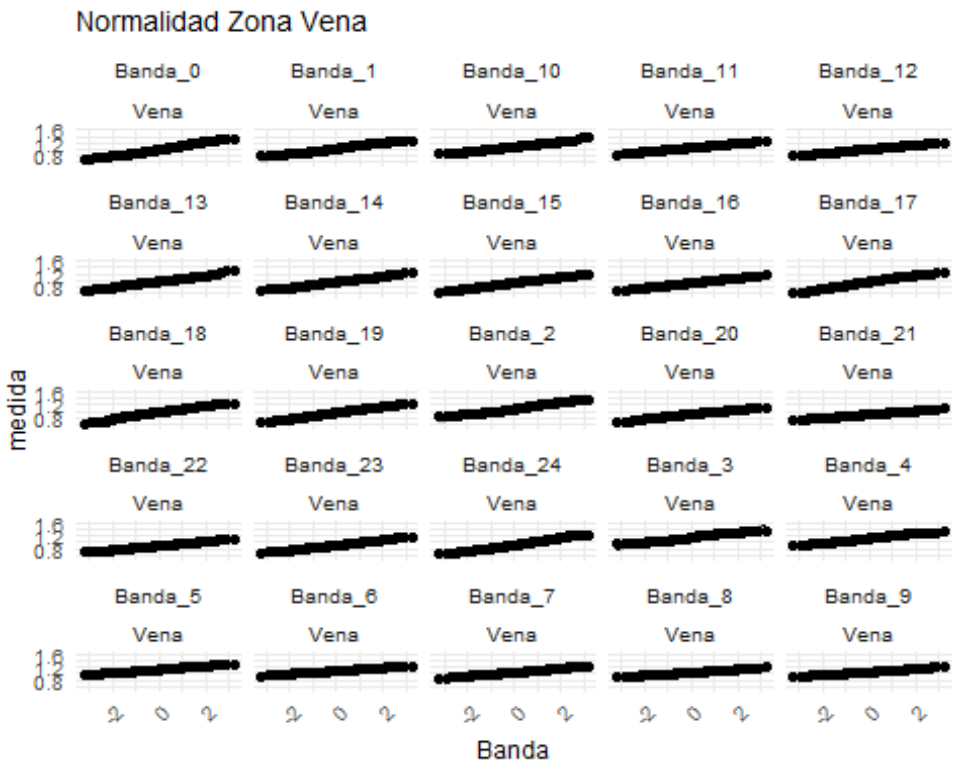
Realizamos el gráfico qqplot de la zona vena:

```
plotdata <- dplyr::filter(paciente_3_long,
                          Zona == "Vena")
ggqqplot(plotdata, "medida", ggtheme = theme_bw()) +
  facet_wrap(Banda ~ Zona) +
  theme_minimal(base_size = 9) +
  theme(axis.text.x = element_text(angle = 45,
                                   hjust = 1)) +
  labs(title = "Normalidad Zona Vena",
       x= "Banda",
       y= "medida")
```



Por último, haremos el gráfico qqplot de la zona vena:

```
plotdata <- dplyr::filter(paciente_3_long,
                          Zona == "Vena")
ggqqplot(plotdata, "medida", ggtheme = theme_bw()) +
  facet_wrap(Banda ~ Zona) +
  theme_minimal(base_size = 9) +
  theme(axis.text.x = element_text(angle = 45,
                                   hjust = 1)) +
  labs(title = "Normalidad Zona Vena",
       x= "Banda",
       y= "medida")
```



Como podemos observar, todas las áreas tumorales siguen la normalidad, así que procederemos a realizar el anova mixto de 2 vías

Anova de 2 vías mixto

Realizo el anova de 2 vías mixto, ya que nuestras muestras siguen la normalidad, podemos realizar pruebas paramétricas.

```
res.aov <- anova_test(data = paciente_3_long,
dv = medida, wid = X, between = Zona, within = Banda )

## Warning: The 'wid' column contains duplicate ids across between-subjects
## variables. Automatic unique id will be created

get_anova_table(res.aov) # aplica corrección automática

## ANOVA Table (type III tests)
##
##       Effect   DFn     DFd        F       p p<.05      ges
## 1       Zona  3.00  7053.0  100.983 5.3e-64     * 8.54e-05
## 2      Banda  8.62 60819.6 2386.263 0.0e+00     * 2.52e-01
## 3 Zona:Banda 25.87 60819.6  607.343 0.0e+00     * 2.05e-01
```

Como puedo comprobar las iteraciones entre bandas y zonas, obtengo un resultado significativo, ya que el valor que obtenemos en el p-valor es inferior a 0.05, con todo esto podemos decir que tenemos diferencias significativas entre los píxeles de las bandas y las

áreas del tumor. Con esto seguiré con la prueba Post-Hoc de comparaciones múltiples, con esta prueba podemos ver las diferencias entre todas las bandas y las áreas tumorales.

Hacemos las comparaciones múltiples

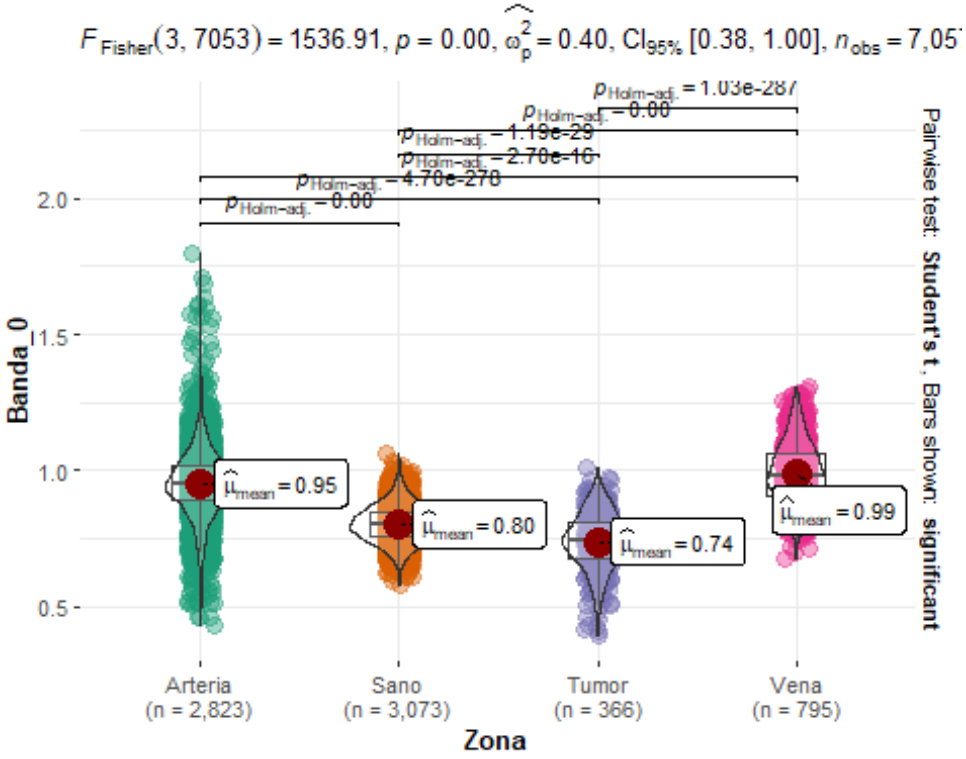
```
paciente_3_long %>%
group_by(Zona) %>%
pairwise_t_test(medida ~ Banda, paired = TRUE,
p.adjust.method = "holm") %>%
filter(p.adj < 0.05)
```

```
## # A tibble: 1,112 x 11
##    Zona    .y.    group1  group2    n1    n2 statistic    df         p
   p.adj
##    <fct>   <chr>  <chr>   <chr>  <int> <int>     <dbl> <dbl>     <dbl>
   <dbl>
##  1 Arteria medida Banda_0 Banda~  2823  2823    -15.4   2822 1.75e- 51 1
.5 e- 49
##  2 Arteria medida Banda_0 Banda~  2823  2823    -49.1   2822 0         0

##  3 Arteria medida Banda_0 Banda~  2823  2823    -44.7   2822 0         0

##  4 Arteria medida Banda_0 Banda~  2823  2823    -28.5   2822 8.84e-157 1
.26e-154
##  5 Arteria medida Banda_0 Banda~  2823  2823    -12.3   2822 5.98e- 34 3
.89e- 32
##  6 Arteria medida Banda_0 Banda~  2823  2823    -10.4   2822 1.11e- 24 6
.33e- 23
##  7 Arteria medida Banda_0 Banda~  2823  2823     -3.31  2822 9.47e-  4 1
.9 e-  2
##  8 Arteria medida Banda_0 Banda~  2823  2823     -4.49  2822 7.32e-  6 1
.98e-  4
##  9 Arteria medida Banda_0 Banda~  2823  2823     -8.02  2822 1.57e- 15 7
.38e- 14
## 10 Arteria medida Banda_0 Banda~  2823  2823     -7.73  2822 1.52e- 14 6
.99e- 13
## # ... with 1,102 more rows, and 1 more variable: p.adj.signif <chr>
```

Realizamos gráficos de comparaciones múltiples, con los que comprobaremos la diferencia entre las bandas y las áreas del tumor:

Banda 0

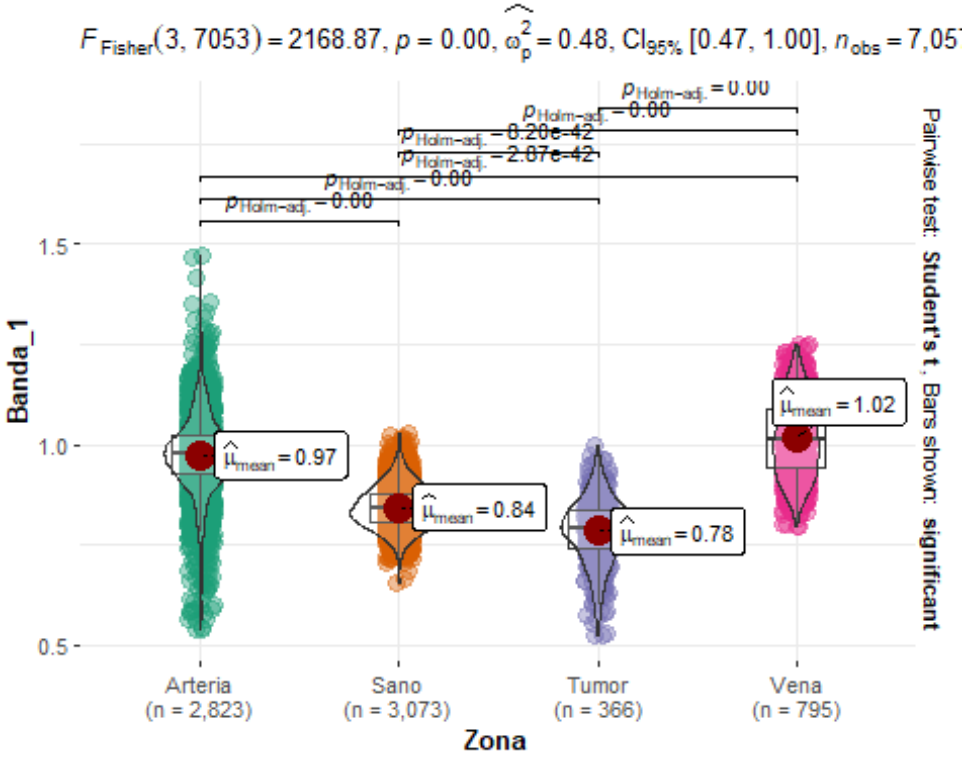
```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_0,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



Cómo podemos ver en la banda 0 tenemos diferencias significativas entre las 4 zonas del tumor.

Banda 1:

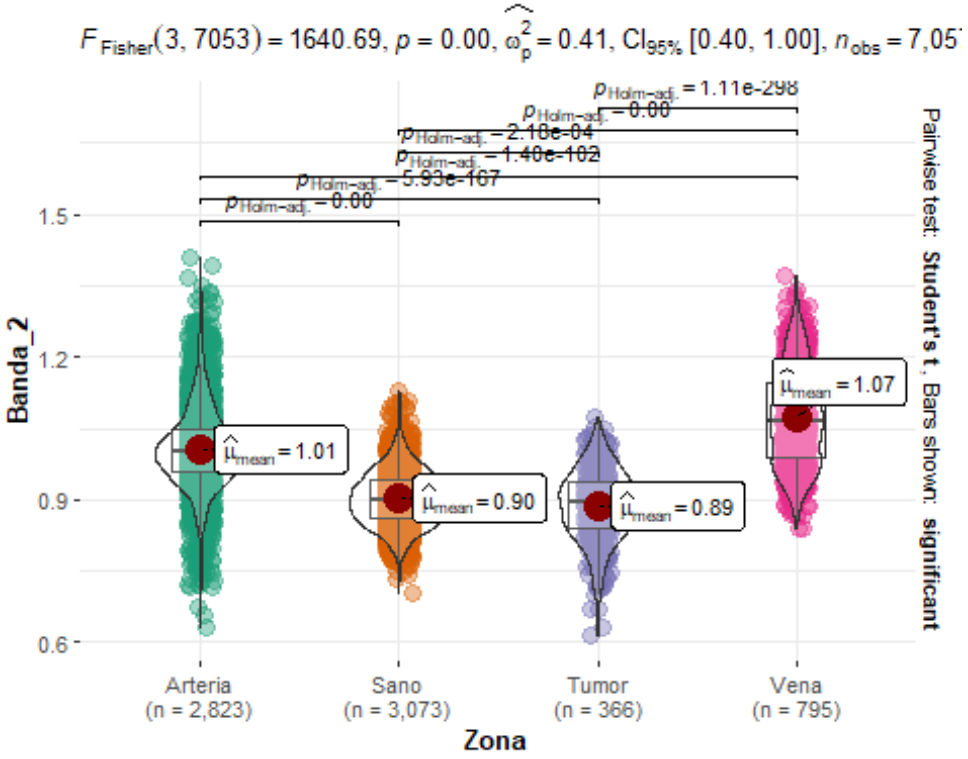
```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_1,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



Obtenemos el mismo resultado tenemos diferencias significativas entre os píxels de la banda 1 y las 4 áreas del tumor

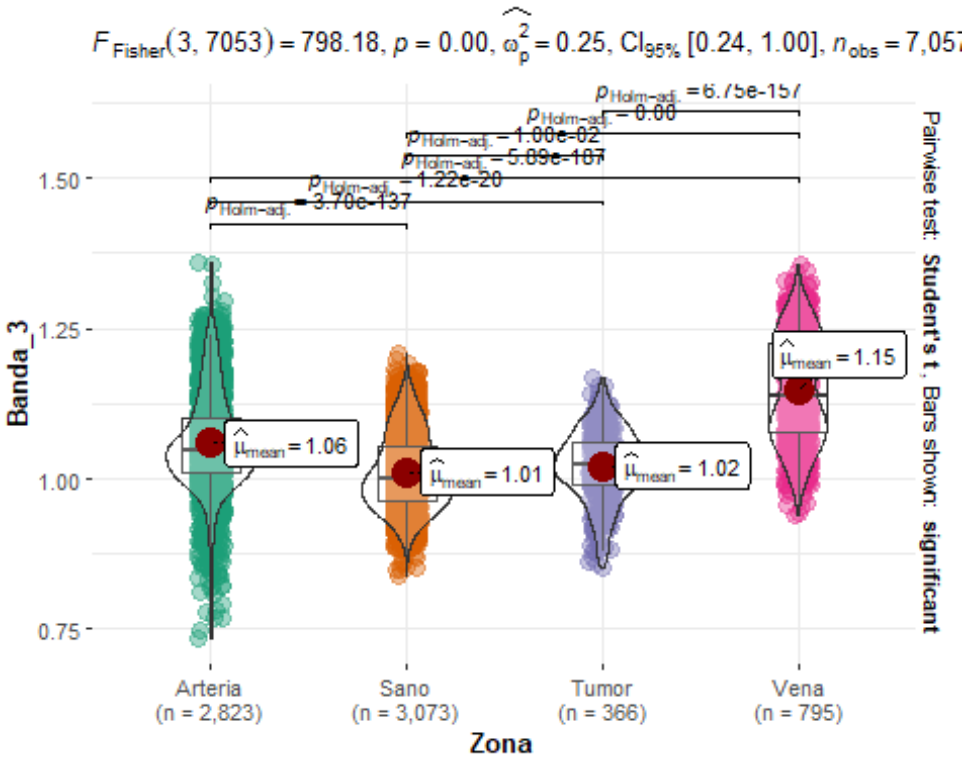
Banda 2:

```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_2,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



Banda 3:

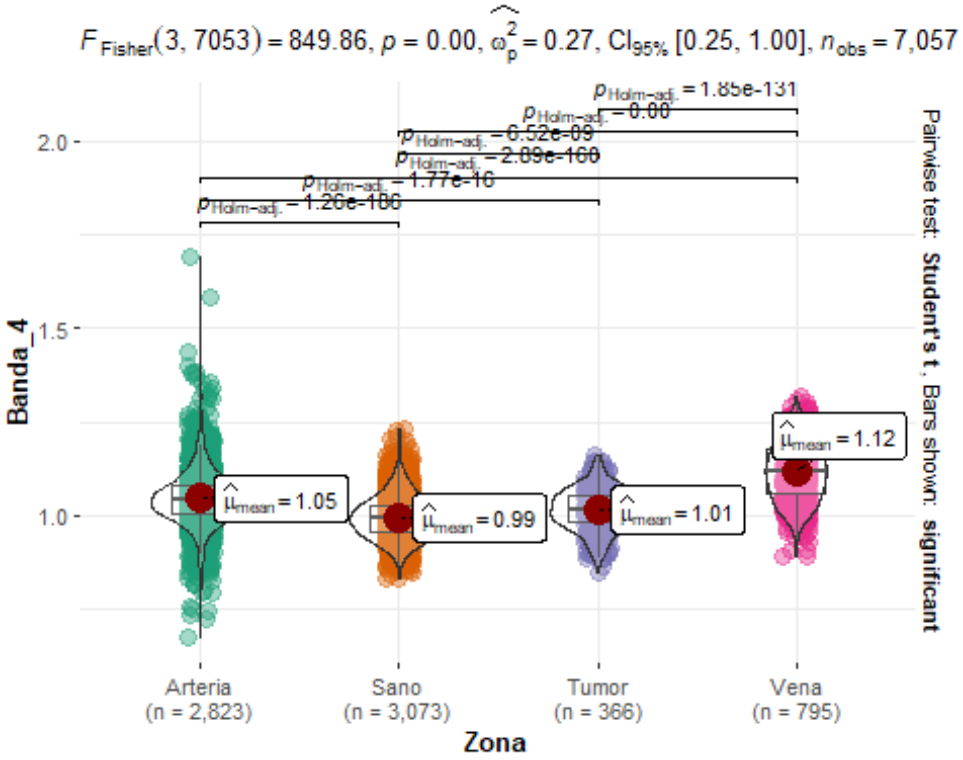
```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_3,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



Como podemos observar tenemos diferencias significativas entre las medias de los píxeles de las 4 áreas tumorales

Banda 4:

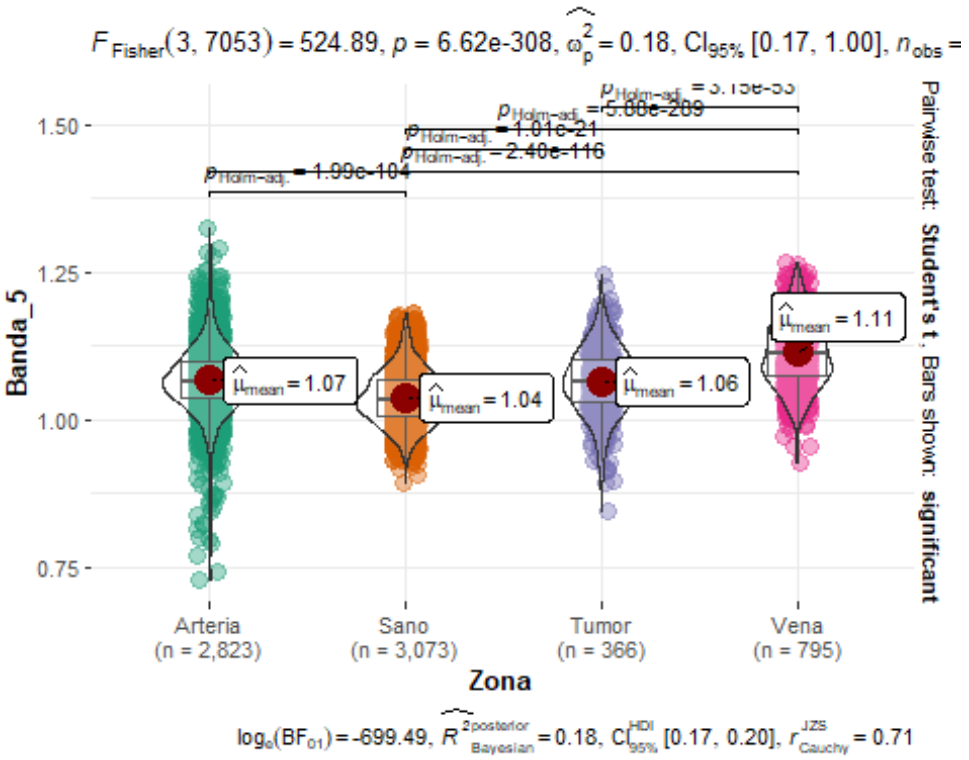
```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_4,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



Tenemos diferencias significativas entre las medias de los píxeles de las 4 áreas tumorales

Banda 5:

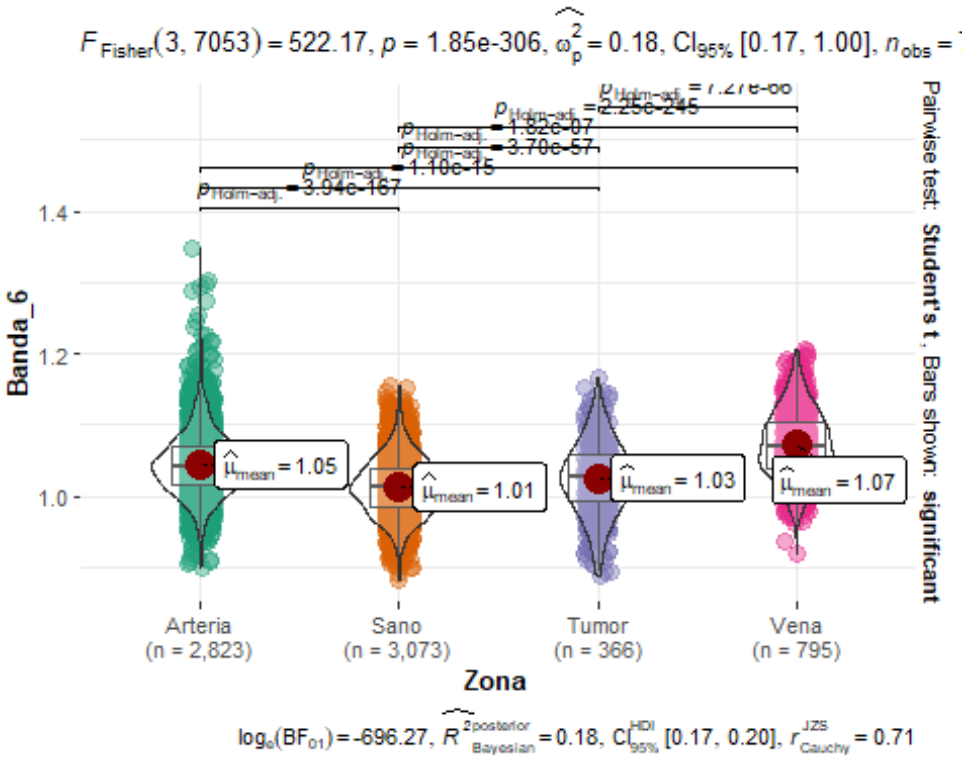
```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_5,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



Podemos ver que la media de los píxeles de las 4 áreas tumorales difiere entre sí

Banda 6:

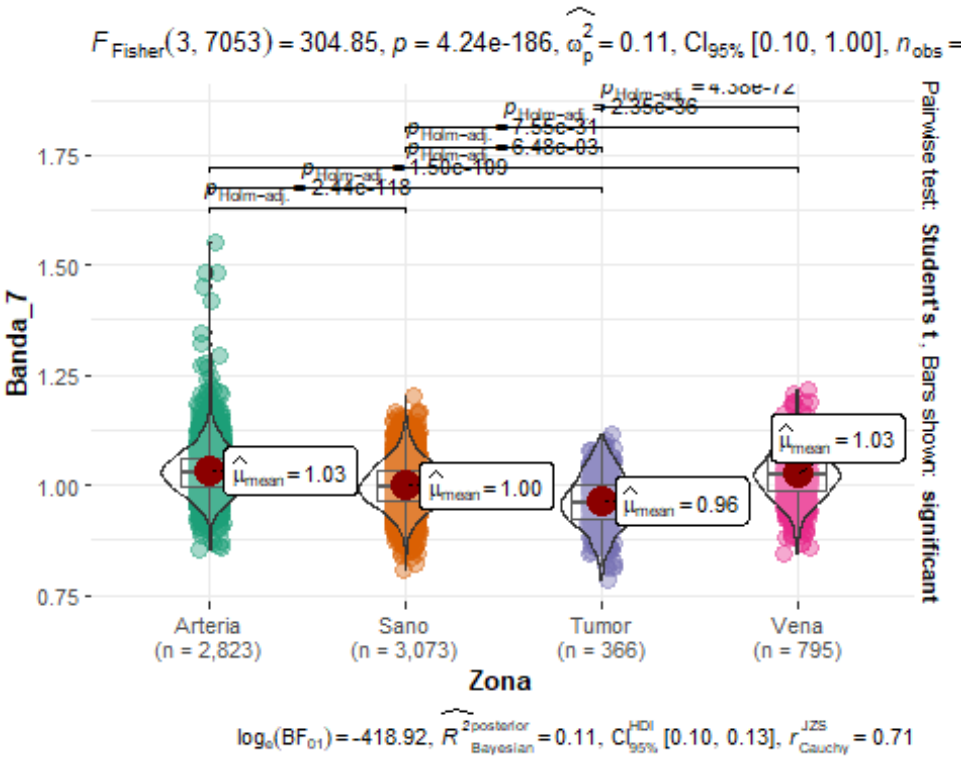
```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_6,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



Como podemos observar tenemos diferencias significativas entre las medias de los píxeles de las 4 áreas tumorales

Banda 7:

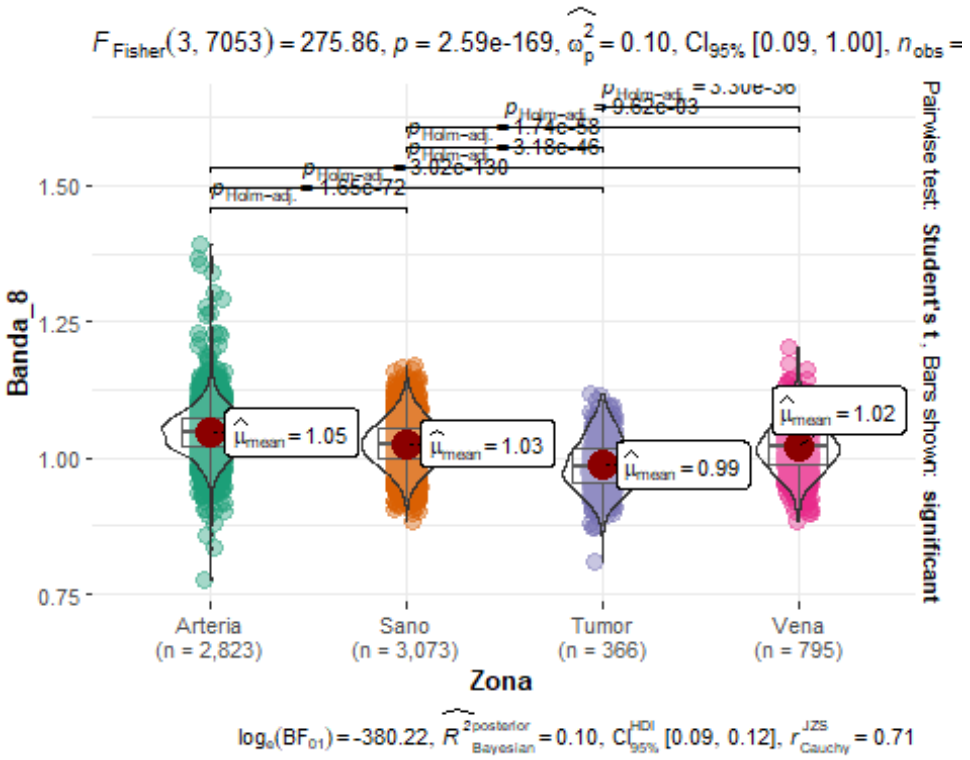
```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_7,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



Tenemos diferencias significativas entre las medias de los píxeles de las 4 áreas tumorales

Banda 8:

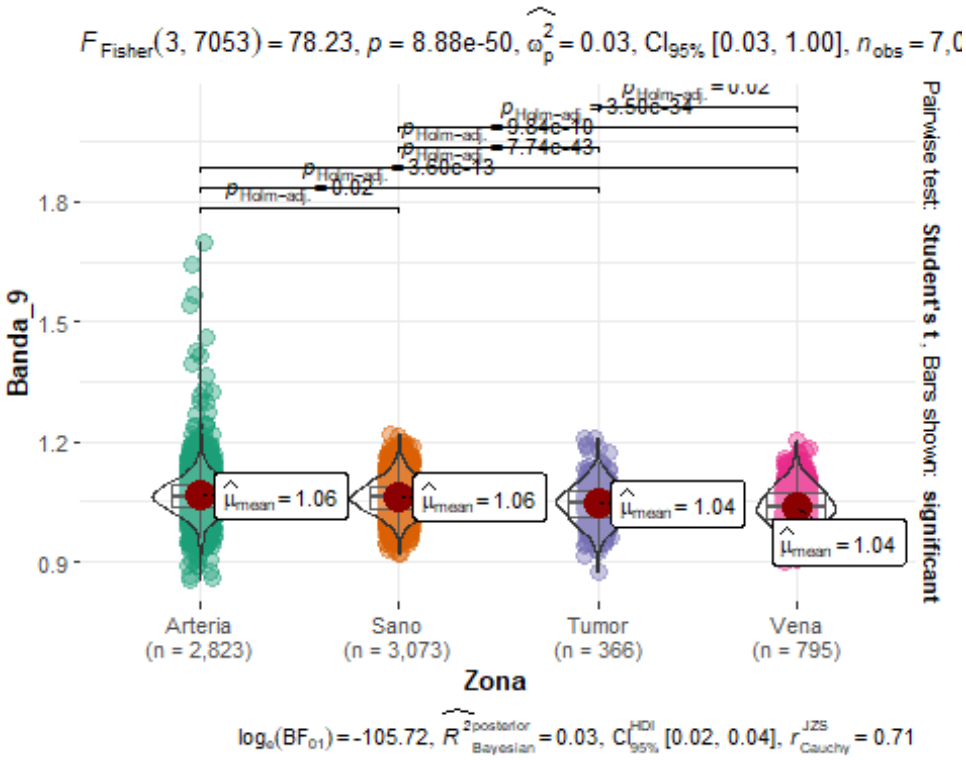
```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_8,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



Tenemos diferencias significativas entre las medias de los píxeles de las 4 áreas tumorales

Banda 9:

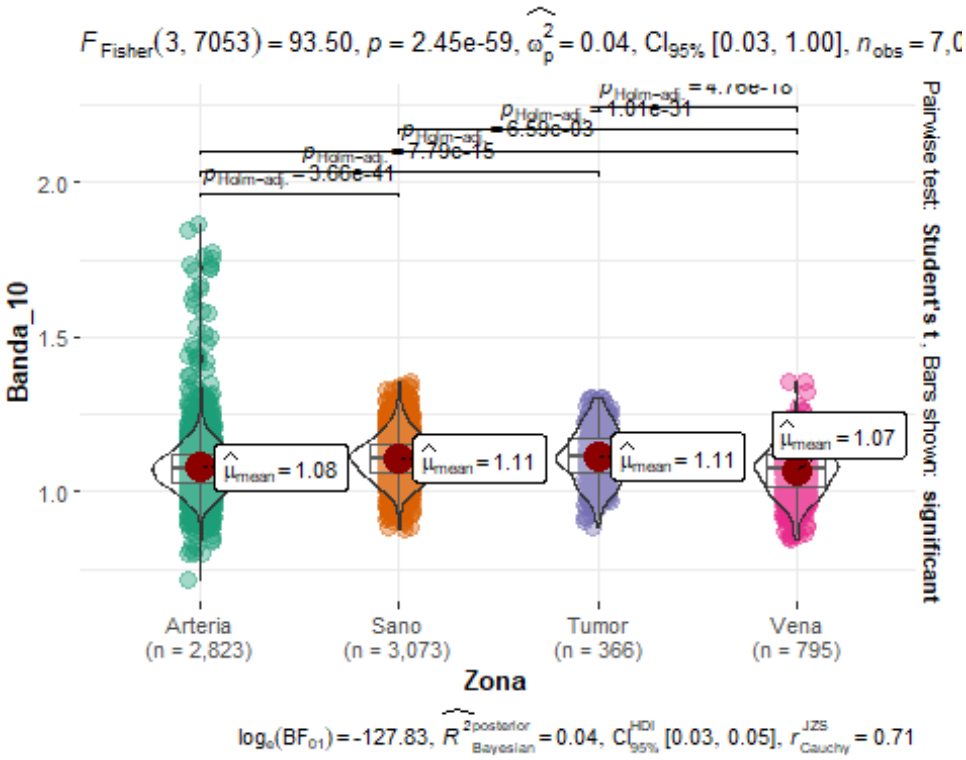
```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_9,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



En esta banda podemos ver que los píxeles de las zonas arteria con sano coinciden en su media igual que los píxeles de las zonas tumor con vena.

Paciente 10:

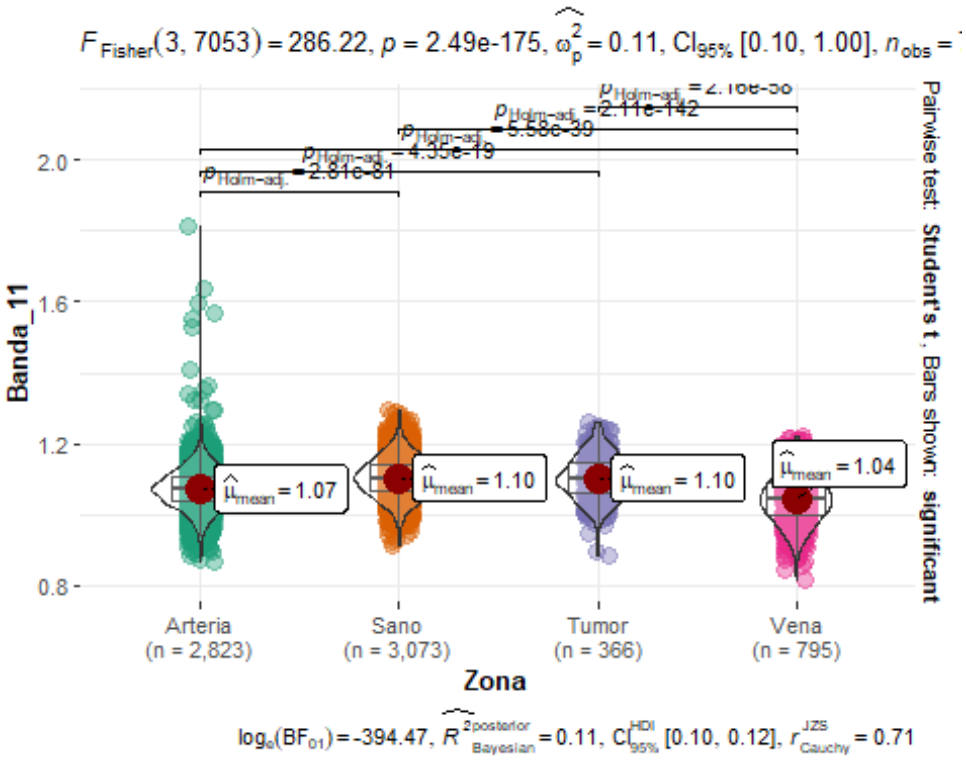
```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_10,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



En la banda 10 encontramos que los píxeles de la zona sana y tumor vemos que son iguales y no hay diferencias significativas entre ambas bandas, sin embargo si hay diferencias significativas entre las medias de los píxeles de arteria con vena.

Banda 11:

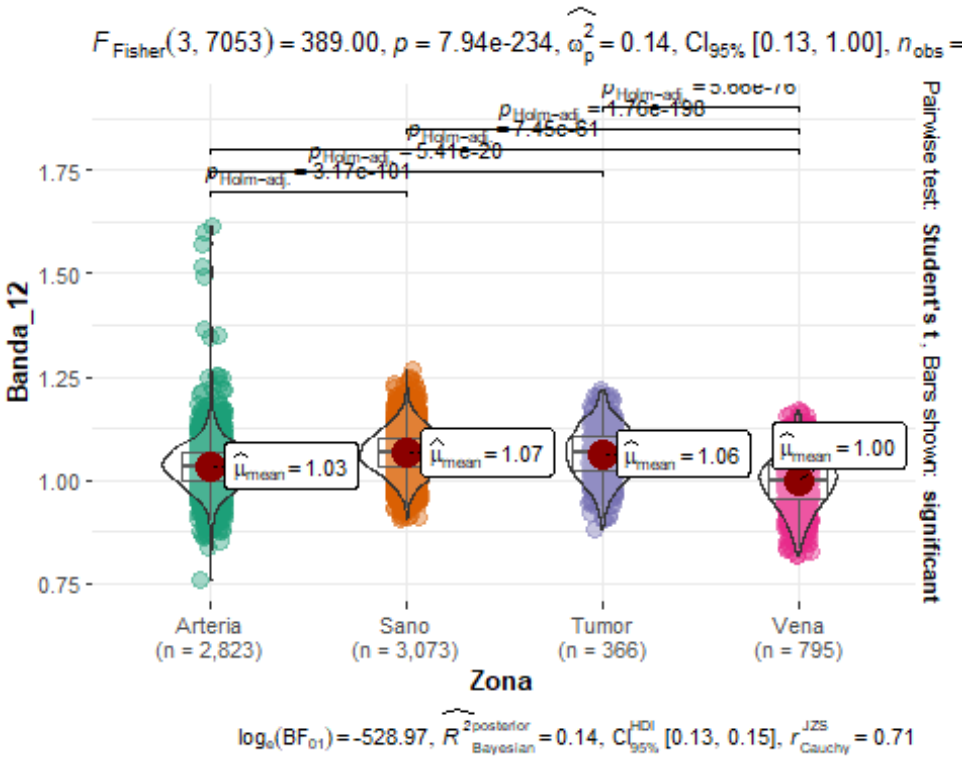
```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_11,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



En la banda 11 vemos que no tenemos diferencias significativas en los píxeles de las zonas tumor y sano

Banda 12:

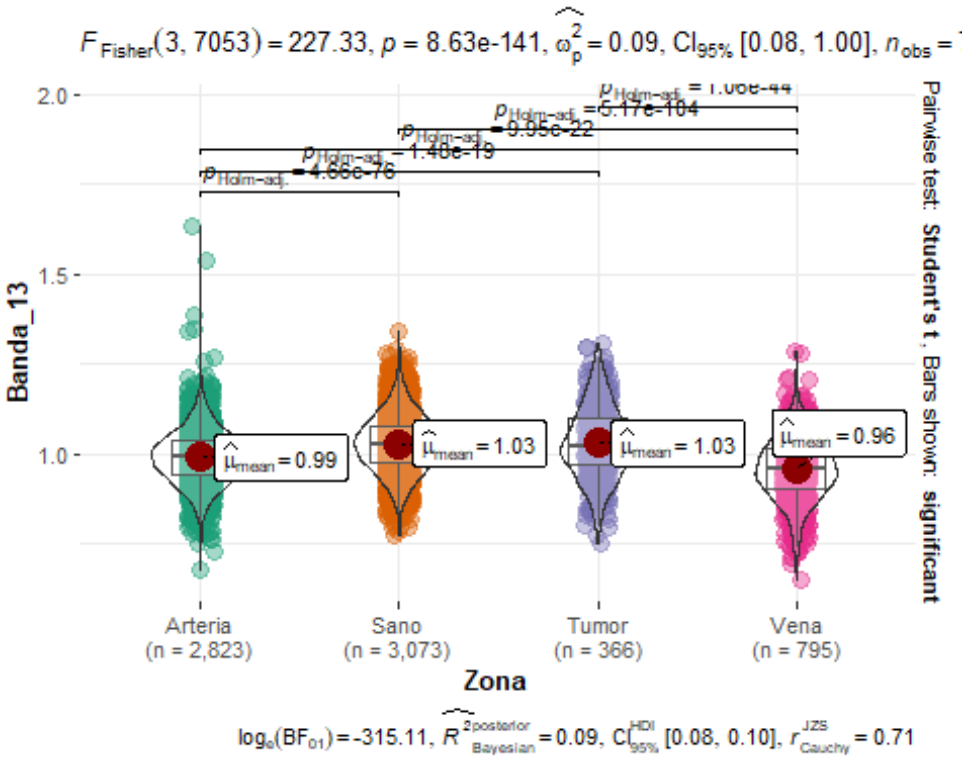
```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_12,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



En los píxeles de la banda 12 todas las áreas de la banda son diferentes entre sí.

Banda 13:

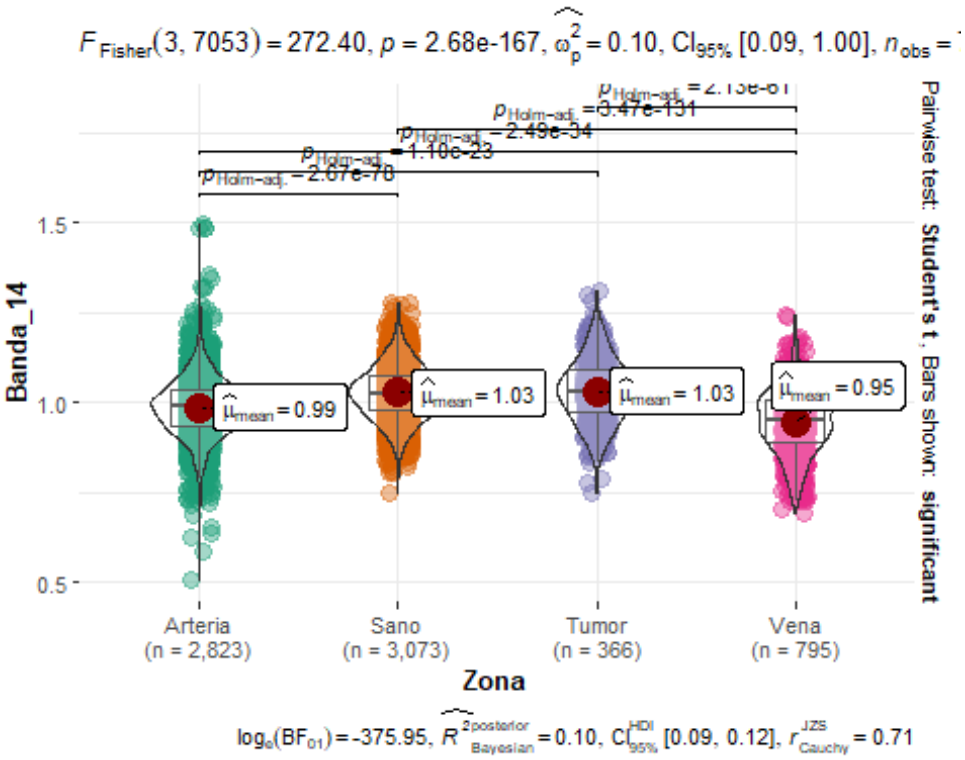
```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_13,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



En los píxeles de la banda 13 podemos ver que las zonas tumor y sano no hay diferencias significativas entre esas 2 zonas, sin embargo, si tenemos diferencias significativas entre los píxeles de las zonas vena y arteria.

Banda 14:

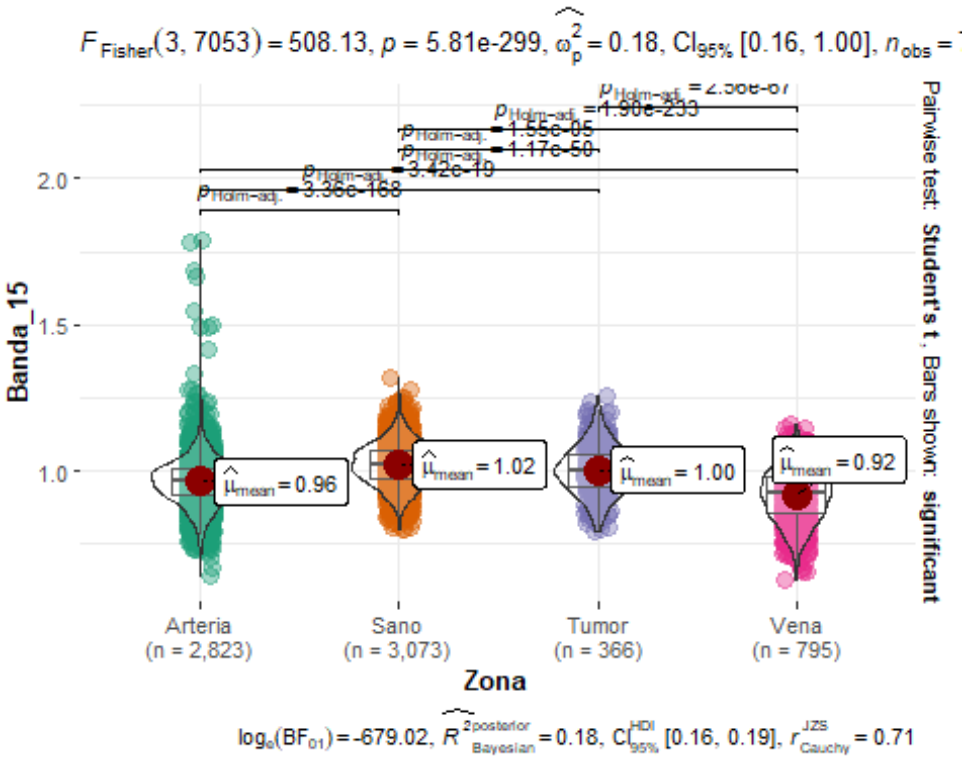
```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_14,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



Vemos qué los píxeles de la zona de tumor y sano tienen la misma media, sin embargo hay diferencias significativas entre los píxeles de vena y arteria

Banda 15:

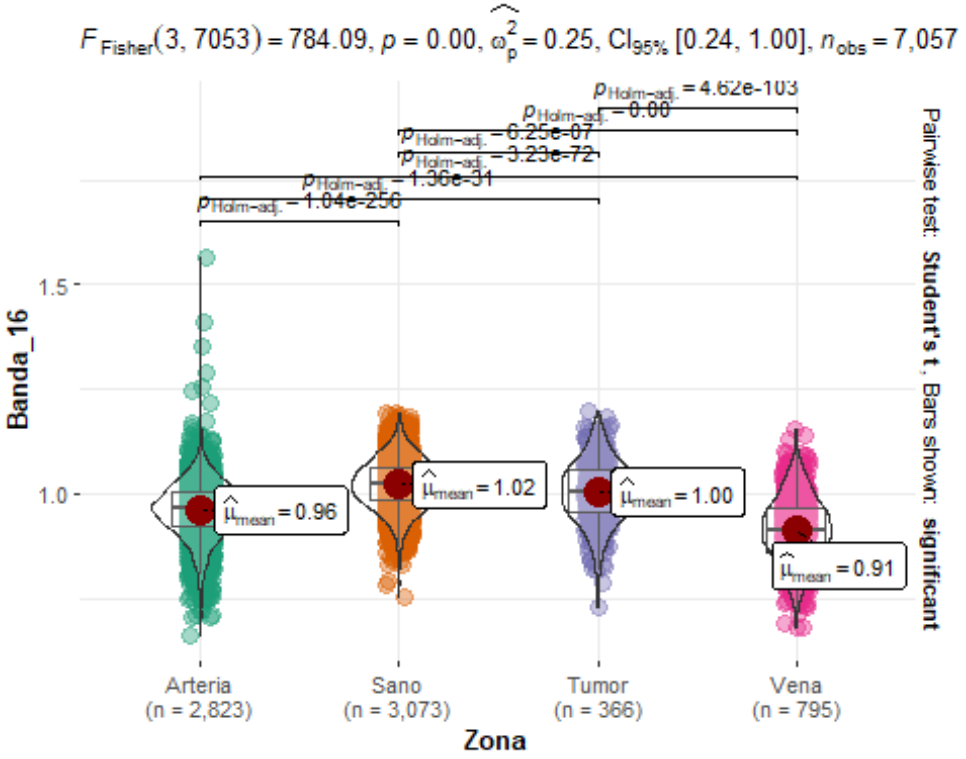
```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_15,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



Tenemos diferencias significativas entre los píxeles de las 4 áreas del tumor

Banda 16:

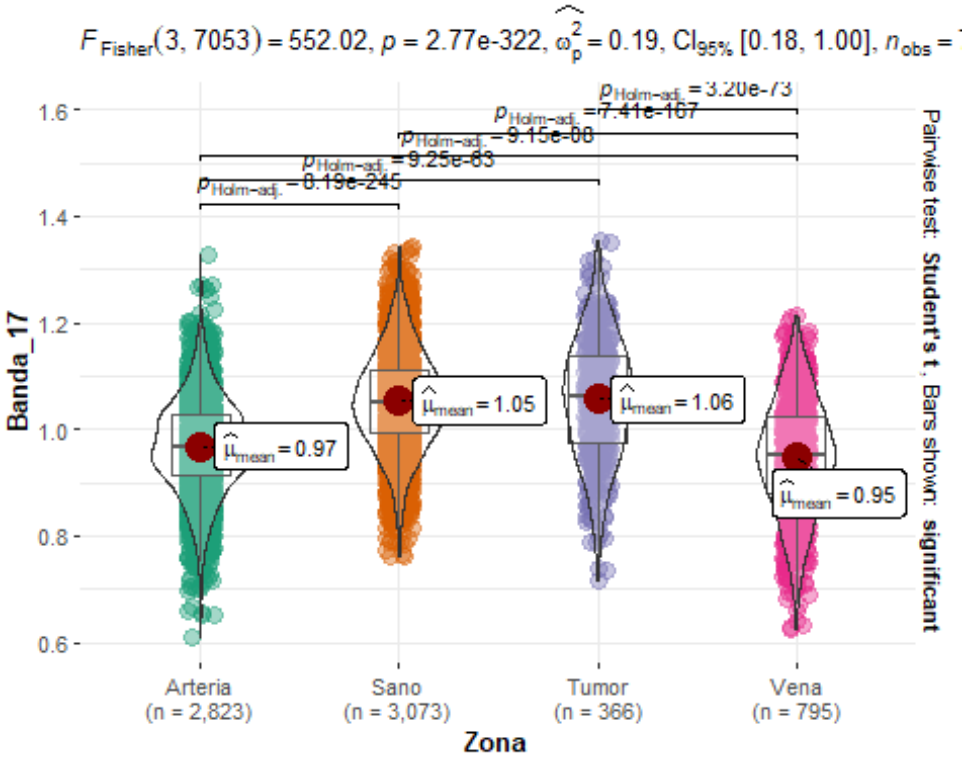
```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_16,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



Los píxeles de la banda 16 son diferentes con respect a las 4 areas del tumor.

Banda 17:

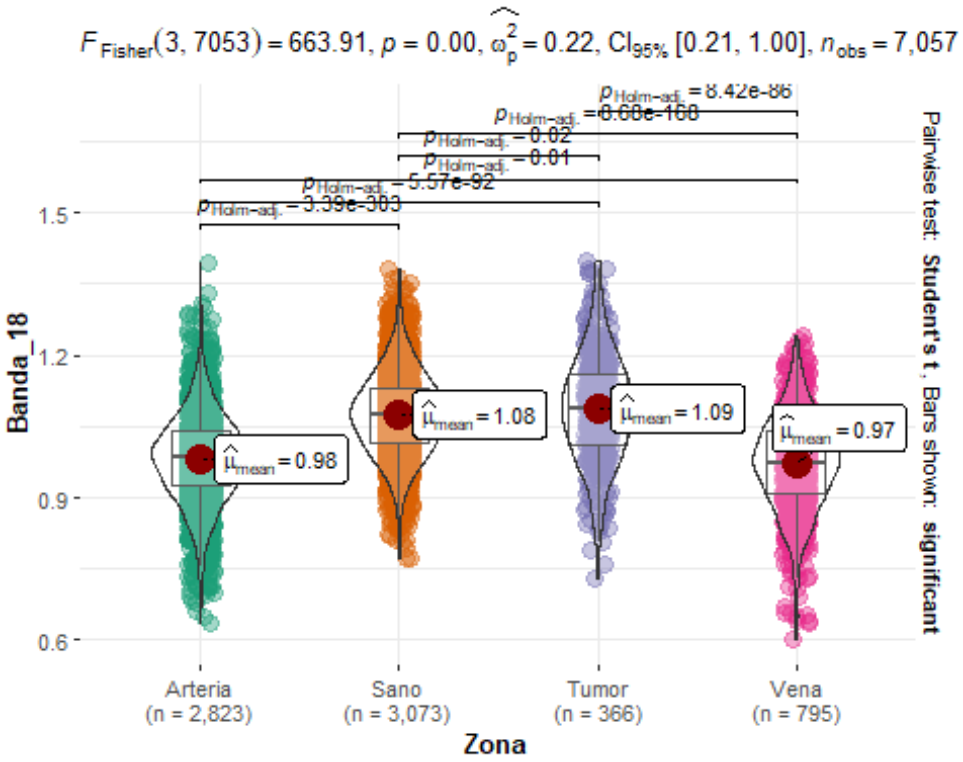
```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_17,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



Los píxeles de la banda 17 son diferentes con respecto a las 4 áreas

Banda 18:

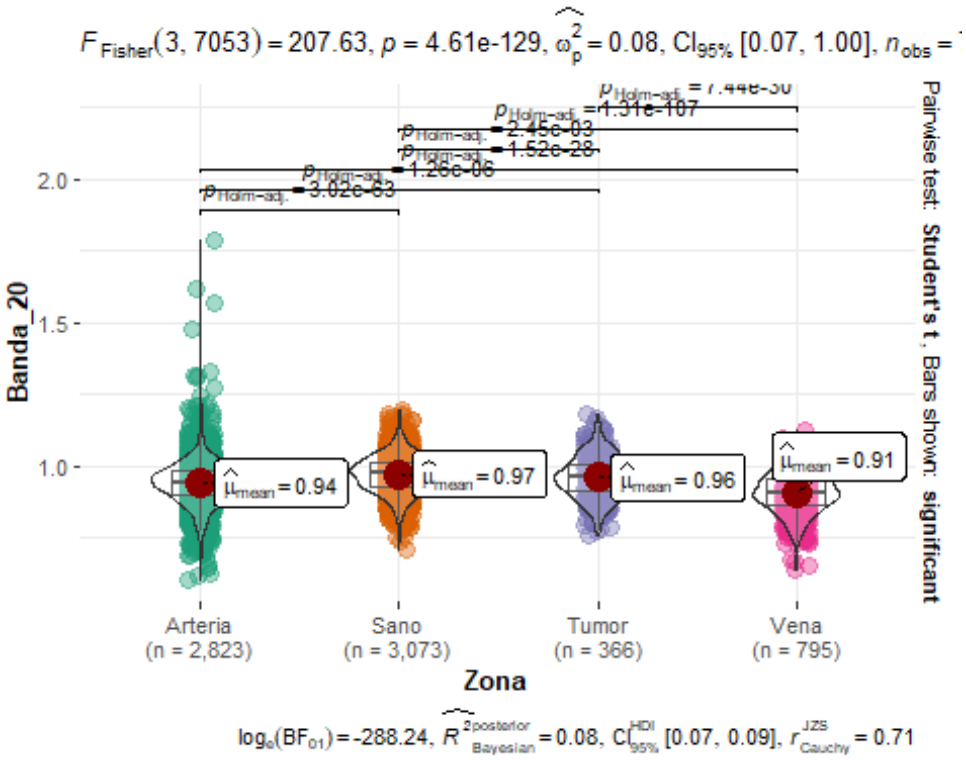
```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_18,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



Hay diferencias significativas entre los píxeles de las 4 áreas tumorales

Banda 19:

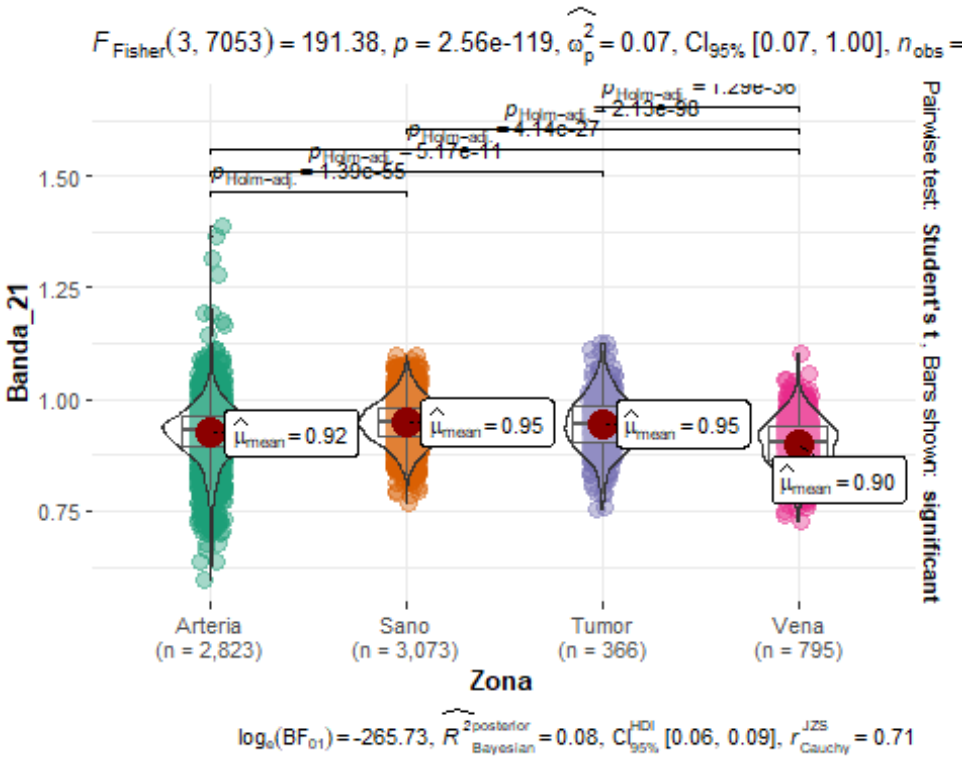
```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_20,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



Vemos que los píxeles de las áreas tumorales las medias de las áreas tumorales son diferentes entre sí.

Banda 21

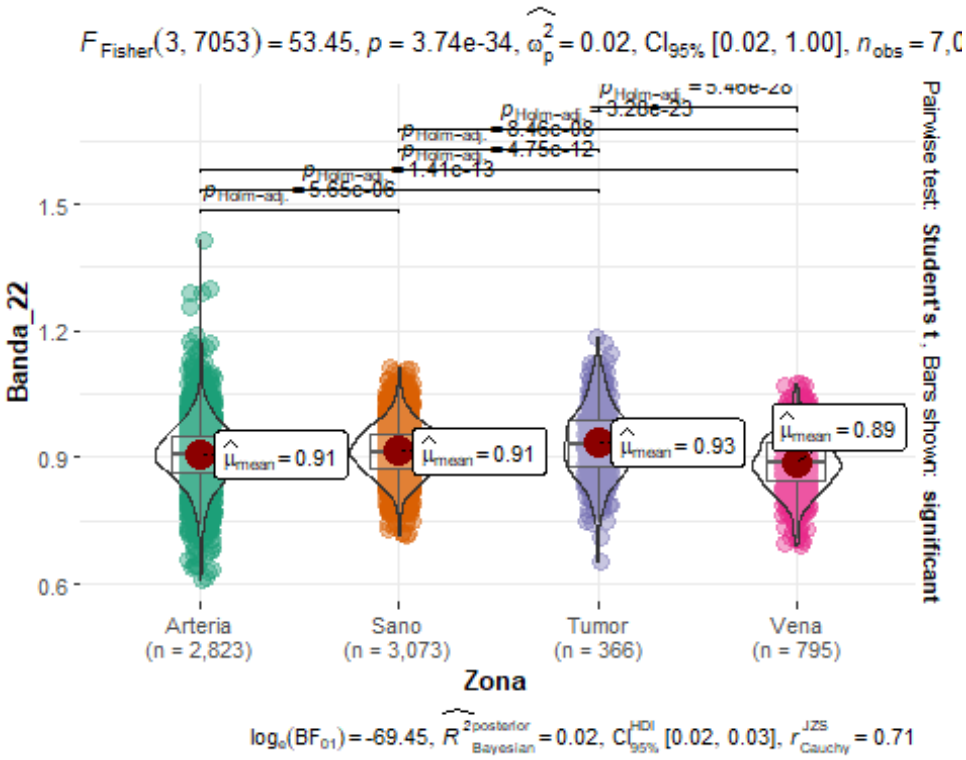
```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_21,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



En los píxeles de las áreas de sano y tumor son iguales, sin embargo tenemos diferencias significativas entre las medias de vena y arteria

Banda 22:

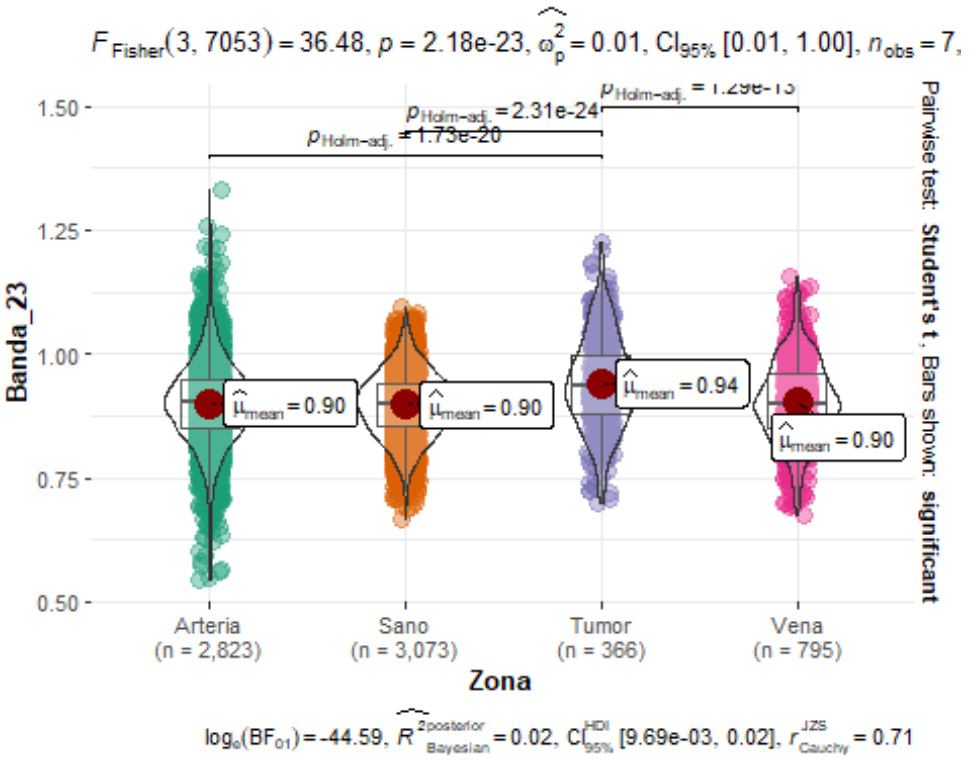
```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_22,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



En los píxeles de la banda 22, son iguales las medias en la zona de arteria y sano, sin embargo tenemos diferencia significativas en la medias de vena y tumor

Banda 23:

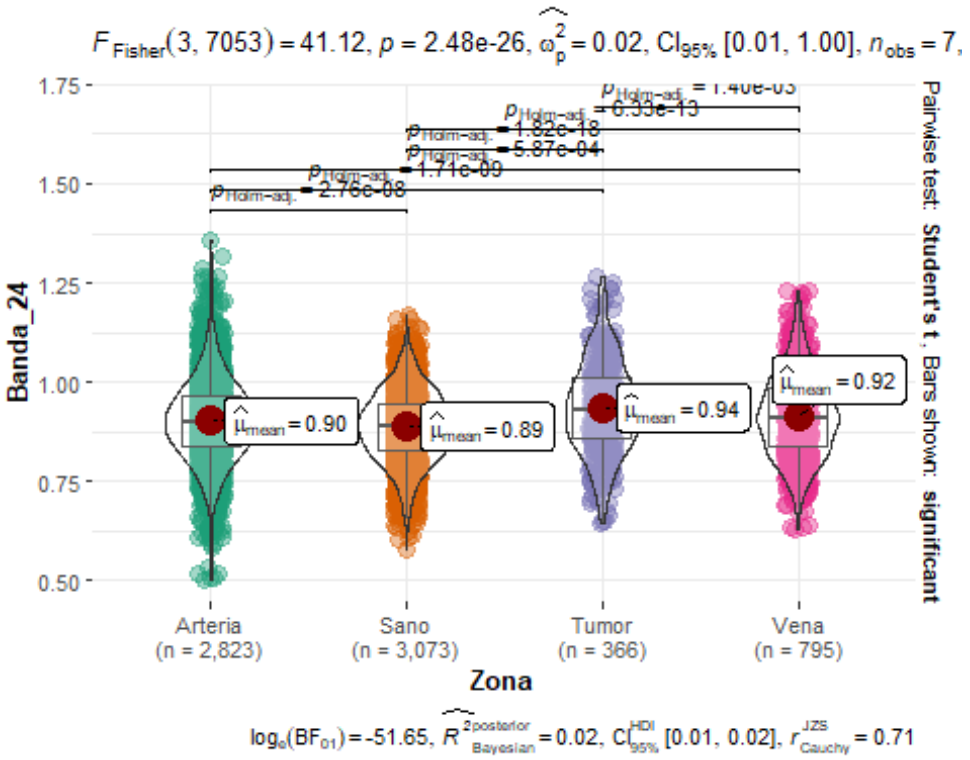
```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_23,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



En la banda 23 nos encontramos que no hay diferencias entre las médias entre la zona de arteria y sano, y vena.

Banda 24:

```
ggbetweenstats(data = Paciente_3, x = Zona, y = Banda_24,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)
```



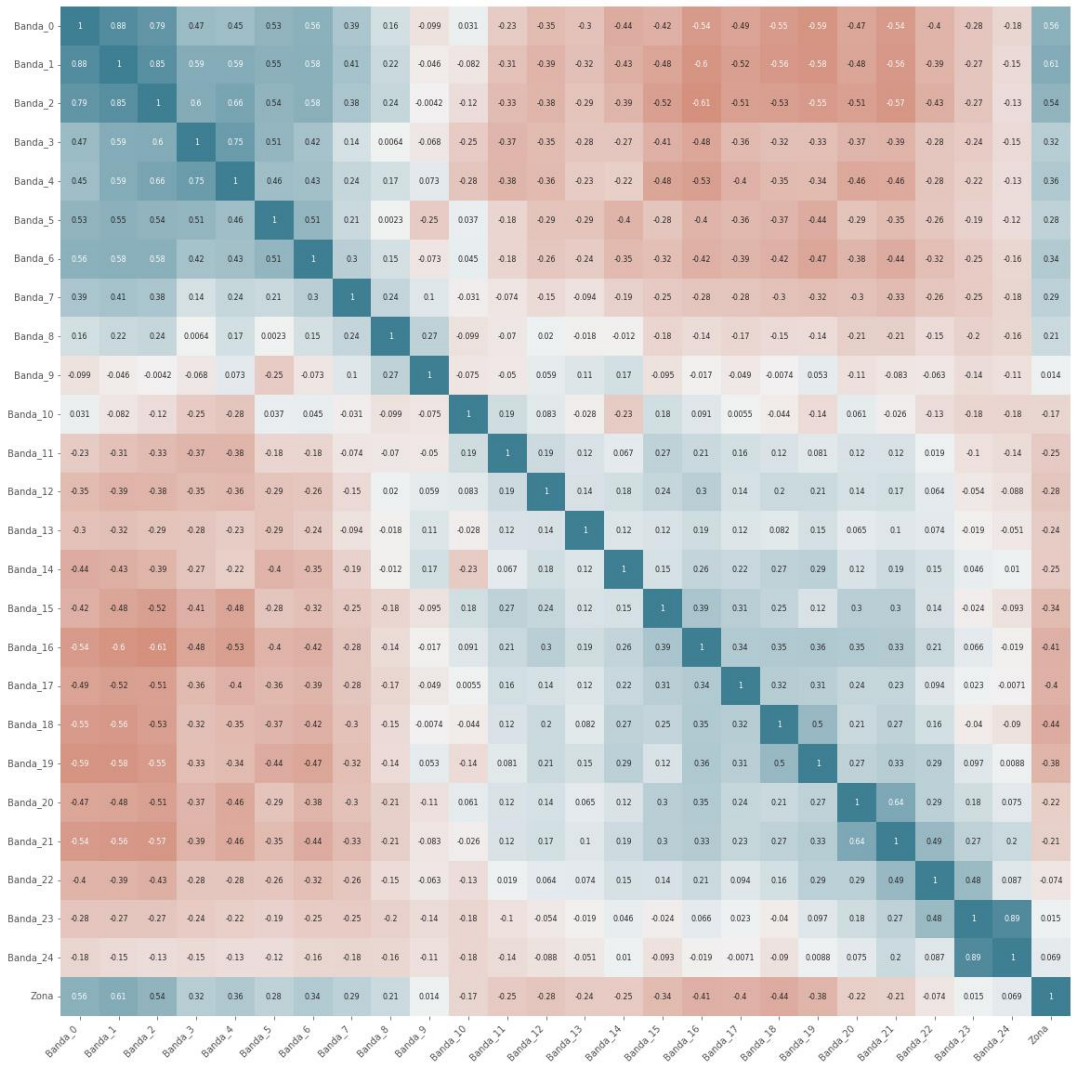
En la última banda, encontramos que las medias de los píxeles de la banda 24 hay diferencias entre las 4 áreas.

Correlación Paciente 3:

```
1           # Tratamiento de datos
 2          # ============================================
 3          ==================================
 4          importar  pandas  como  pd
 5          importar  numpy  como  np
 6          de  sklearn.datasets  importar load_diabetes
 7
 8          # Gráficos
 9          # ============================================
10          ==================================
11          importar  matplotlib.pyplot  como  plt
12          desde  matplotlib  importar estilo
13           importar  seaborn  como  sns
14
15          # Preprocesado y análisis
dieciséis   # ============================================
17          ==================================
18          importar  statsmodels.api  como  sm
19          importar  pingouin  como  pg
20          de  scipy  importar estadísticas
21           de  scipy. estadísticas  importar pearsonr
22
23          # Configuración matplotlib
24          # ============================================
25          ==================================
26          plt.style.use( 'ggplot' )
```

```
27
28          # Advertencias de configuración
29          # ============================================
30          ================================
31          importar  advertencias
32          advertencias.filterwarnings( 'ignorar' )
33
34          df= pd.read_csv( './paciente3SVM.csv' )
35
36          # Matriz de conexiones
37          # ============================================
38          ================================
39          corr_matrix = df.corr(method= 'pearson' )
40          corr_matriz
41
42          def  tidy_corr_matrix (corr_mat):
43               '''
44              Función para convertir una matriz de coincidencias de pandas
45          en formato tidy.
46              '''
47              mat_corr = mat_corr.pila().reset_index()
48              corr_mat.columnas = [ 'variable_1' , 'variable_2' , 'r' ]
49              mat_corr = mat_corr.loc[mat_corr[ 'variable_1' ] !=
50          mat_corr[ 'variable_2' ], :]
51              corr_mat[ 'abs_r' ] = np.abs(corr_mat[ 'r' ])
52              corr_mat = corr_mat.sort_values( 'abs_r' , ascendente= False
53          )
54
55              devolver (corr_mat)
56
57          ordenado_corr_matriz(corr_matriz).cabeza( 10 )
58
59          # Heatmap matriz de correlaciones
60          # ============================================
61          ================================
62          fig, ax = plt.subplots(nrows= 1 , ncols= 1 , tamaño de figura=(
63          24 , 20 ))
64
sesenta y   sns.mapa de calor(
cinco           matriz_corregida,
66              annot = Verdadero ,
67              cbar = Falso ,
68              annot_kws = { "tamaño" : 8 },
69              vmin = - 1 ,
70              vmáx = 1 ,
71              centro = 0 ,
72              cmap = sns.paleta_divergente( 20 , 220 , n= 200 ),
73              cuadrado = Verdadero ,
                hacha = hacha
            )

            hacha.set_xticklabels(
                ax.get_xticklabels(),
                rotación = 45 ,
                alineación horizontal = 'derecha' ,
            )
```

```
ax.tick_params(tamaño de etiqueta = 10 )
```



Podemos ver como hemos descrito en los resultados, apenas tenemos variables relacionadas, esto es perfecto para que no tengamos solapamientos en las bandas en las máquinas de vector soporte.

Maquinas de Vector Soporte del paciente 3:

Exponemos los resultados de la máquina de vector soporte del paciente 3, los datos de todos los pacientes se realizáron del mismo método:

```
# Este entorno de Python 3 viene con muchas bibliotecas de
análisis útiles instaladas
```

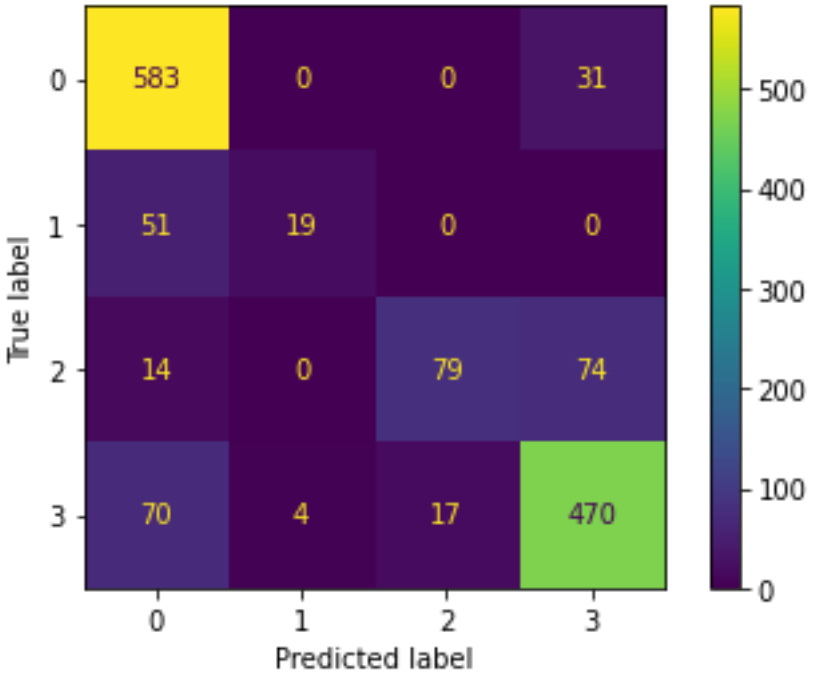
```
2       # Está definido por la imagen acoplable de kaggle/python:
3       https://github.com/kaggle/docker-python
4       # Por ejemplo, aquí hay varios paquetes útiles para cargar en
5       pandas de importación como pd import numpy as np # álgebra lineal
6       import pandas as pd # procesamiento de datos, E/S de archivo CSV
7       (por ejemplo, pd.read_csv) import matplotlib.pyplot as plt # para
8       visualización de datos import seaborn as sns # para visualización
9       de datos estadísticos
10
11
12
13
14      % matplotlib en línea
15      de  sklearn.model_selection  importar train_test_split
         de  sklearn.model_selection  importar cross_val_score
dieciséis de  sklearn.model_selection  importar KFold
17       de  sklearn  importar svm
18       de  sklearn.model_selection  importar GridSearchCV
19
20
21      ## Cargamos el dataset del paciente 3
22
23      df = pd.read_csv( './paciente3SVM.csv' )
24      df.cabeza()
25
26      # comprobar la distribución de la columna target_class
27
28      df[ 'Zona' ].value_counts()
29
30      # dibujar diagramas de caja para visualizar valores atípicos
31
32      plt.figura(tamañofig=( 24 , 20 ))
33
34
35      plt.subtrama( 5 , 2 , 1 )
36      fig = df.boxplot(columna= 'Banda_0' )
37      fig.set_title( '' )
38      fig.set_ylabel( 'Banda_0' )
39
40
41      plt.subtrama( 5 , 2 , 2 )
42      fig = df.boxplot(columna= 'Banda_1' )
43      fig.set_title( '' )
44      fig.set_ylabel( 'Banda_1' )
45
46
47      plt.subtrama( 5 , 2 , 3 )
48      fig = df.boxplot(columna= 'Banda_2' )
49      fig.set_title( '' )
50      fig.set_ylabel( 'Banda_2' )
51
52
53      plt.subtrama( 5 , 2 , 4 )
54      fig = df.boxplot(columna= 'Banda_3' )
55      fig.set_title( '' )
56      fig.set_ylabel( 'Banda_3' )
```

```
 57
 58
 59        plt.subtrama( 5 , 2 , 5 )
 60        fig = df.boxplot(columna= 'Banda_4' )
 61        fig.set_title( '' )
 62        fig.set_ylabel( 'Banda_4' )
 63
 64
 sesenta   plt.subtrama( 5 , 2 , 6 )
y cinco    fig = df.boxplot(columna= 'Banda_5' )
 66        fig.set_title( '' )
 67        fig.set_ylabel( 'Banda_5' )
 68
 69
 70        plt.subtrama( 5 , 2 , 7 )
 71        fig = df.boxplot(columna= 'Banda_6' )
 72        fig.set_title( '' )
 73        fig.set_ylabel( 'Banda_6' )
 74
 75
 76        plt.subtrama( 5 , 2 , 8 )
 77        fig = df.boxplot(columna= 'Banda_7' )
 78        fig.set_title( '' )
 79        fig.set_ylabel( 'Banda_7' )
 80
 81        plt.subtrama( 5 , 3 , 1 )
 82        fig = df.boxplot(columna= 'Banda_8' )
 83        fig.set_title( '' )
 84        fig.set_ylabel( 'Banda_8' )
 85
 86        plt.subtrama( 5 , 3 , 2 )
 87        fig = df.boxplot(columna= 'Banda_10' )
 88        fig.set_title( '' )
 89        fig.set_ylabel( 'Banda_10' )
 90
 91        plt.subtrama( 5 , 3 , 3 )
 92        fig = df.boxplot(columna= 'Banda_11' )
 93        fig.set_title( '' )
 94        fig.set_ylabel( 'Banda_11' )
 95
 96        plt.subtrama( 5 , 3 , 4 )
 97        fig = df.boxplot(columna= 'Banda_12' )
 98        fig.set_title( '' )
 99        fig.set_ylabel( 'Banda_12' )
100
101        plt.subtrama( 5 , 3 , 5 )
102        fig = df.boxplot(columna= 'Banda_13' )
103        fig.set_title( '' )
104        fig.set_ylabel( 'Banda_13' )
105
106        plt.subtrama( 5 , 3 , 6 )
107        fig = df.boxplot(columna= 'Banda_14' )
108        fig.set_title( '' )
109        fig.set_ylabel( 'Banda_14' )
110
111        plt.subtrama( 5 , 3 , 7 )
```

```
112         fig = df.boxplot(columna= 'Banda_15' )
113         fig.set_title( '' )
114         fig.set_ylabel( 'Banda_15' )
115 
116         plt.subtrama( 5 , 3 , 8 )
117         fig = df.boxplot(columna= 'Banda_16' )
118         fig.set_title( '' )
119         fig.set_ylabel( 'Banda_16' )
120 
121         plt.subtrama( 5 , 3 , 9 )
122         fig = df.boxplot(columna= 'Banda_17' )
123         fig.set_title( '' )
124         fig.set_ylabel( 'Banda_17' )
125 
126         plt.subtrama( 5 , 4 , 1 )
127         fig = df.boxplot(columna= 'Banda_18' )
128         fig.set_title( '' )
129         fig.set_ylabel( 'Banda_18' )
130 
131         plt.subtrama( 5 , 4 , 2 )
132         figura = df.boxplot(columna= 'Banda_19' )
133         fig.set_title( '' )
134         fig.set_ylabel( 'Banda_19' )
135 
136         plt.subtrama( 5 , 4 , 3 )
137         fig = df.boxplot(columna= 'Banda_20' )
138         fig.set_title( '' )
139         fig.set_ylabel( 'Banda_20' )
140 
141         plt.subtrama( 5 , 4 , 4 )
142         fig = df.boxplot(columna= 'Banda_21' )
143         fig.set_title( '' )
144         fig.set_ylabel( 'Banda_21' )
145 
146         plt.subtrama( 5 , 4 , 5 )
147         fig = df.boxplot(columna= 'Banda_22' )
148         fig.set_title( '' )
149         fig.set_ylabel( 'Banda_22' )
150 
151         plt.subtrama( 5 , 4 , 6 )
152         fig = df.boxplot(columna= 'Banda_23' )
153         fig.set_title( '' )
154         fig.set_ylabel( 'Banda_23' )
155 
156         plt.subtrama( 5 , 4 , 7 )
157         fig = df.boxplot(columna= 'Banda_24' )
158         fig.set_title( '' )
159         fig.set_ylabel( 'Banda_24' )
160 
161         ## Dividimos el dataset en entrenamiento y prueba
162 
163         X = df.drop([ 'Zona' ], eje= 1 )
164 
165         y = df[ 'Zona' ] ## Separamos la parte de predicción y la de
166         prueba
167 
```

```
168         # dividir X e y en conjuntos de entrenamiento y prueba
169
170         de  sklearn.model_selection  import train_test_split
171
172         X_train, X_test, y_train, y_test = train_test_split(X, y,
173         test_size = 0.2 , random_state = 0 ) ## separado
174
175         X_train.shape,X_test.shape ## se muestra
176
177         X_tren.describe()
178
179         ## usamos kfold porque los datos estan desvalanceados
180
181         desde  sklearn.model_selection  importar KFold
182         kf = KFold(n_splits= 5 ) ## sacamos el kfold
183
184         ## LLamamos a las SVM
185
186         svc = svm.SVC()
187
188         ## Elegimos los predictores
189
190         paramgrid = { "núcleo" : ( "lineal" , "rbf" ),
191                       "C" : [ 1 , 10 , 100 , 1000 ]}
192         cv = KFold(n_splits = 5 )
193
194         ## añadimos e kfold
195
196         clf = GridSearchCV(estimador = svc,
197                            param_grid = paramgrid,
198                            puntuación = Ninguno ,
199                            cv=cv)
200         ## Elegimos el predictor
201
202         clf.fit(X, y)
203
204         ##resultados
205
206         pd.DataFrame(clf.cv_results_)
207
208         ## elegimos el mejor estimador
209         clf.mejor_estimador_
210
211         ##resultado
212         clf.mejor_puntuación_
213
214         ## elegimos el mejor estimador
215         clf.mejores_parámetros_
216
217         clf.goleador_
218         clf.n_splits_
219
220         ## LLamamos a la funcion
221
222         y_pred = clf.predict(X_test)
223
```

```
224         ## <Matriz de correlacion y confusion
225         from  sklearn.model_selection  import train_test_split,
226         GridSearchCV
227          from  sklearn.metrics  import Classification_report from
228         sklearn.utils
229          import shuffle
230          from sklearn.metrics import confusion_matrix
231
232         imprimir (informe_clasificación(y_test, y_pred))
             imprimir (confusion_matrix(y_test, y_pred))

            de  sklearn.metrics  importar confusion_matrix,
            ConfusionMatrixDisplay
            cm = matriz_confusión(y_test, y_pred)
            disp = ConfusionMatrixDisplay(confusion_matrix=cm)
            disp.plot()
            plt.mostrar()
```



En la matriz de confusión del paciente 3 podemos ver que tiene un alto porcentaje de aciertos con un porcentaje de más podemos ver que falla 262 de 5381, así que su porcentaje de predicción es de un 82% bastante alto.

Añadiremos el codigo de R de los 6 paciente con la Inferencia estadistica

CODIGO:

```
# ------------------------------
# Instalación de paquetes
# ------------------------------


install.packages("car")
install.packages("compute.es")
install.packages("ggplot2")
install.packages("multcomp")
install.packages("pastecs")
install.packages("reshape")
install.packages("WRS", repos="http://R-Forge.R-project.org")
install.packages("reshape")

library(car)
library(compute.es)
library(ggplot2)
library(multcomp)
library(pastecs)
library(reshape)
library(WRS)
library(reshape)


# -------------------------------------
# ANOVA  de 2 vias mixto de paciente 1.
# -------------------------------------
# Cargamos el paciente 1

paciente_1 <- read.csv("Elpacien1.csv")

## pasamos a formato largo los datos del paciente

library(dplyr)
paciente_1_long<- paciente_1 %>%
  gather(key = "Banda", value = "medida",
         Banda,Banda_1, Banda2, Banda_3, Banda_4, Banda_5, Banda_6, Banda_7
, Banda_8,Banda_9,Banda_10,
Banda_11, Banda_12, Banda_13, Banda_14, Banda_15, Banda_16, Banda_17, Banda
_18, Banda_19, Banda_20,
Banda_21, Banda_22, Banda_23, Banda_24) %>%
  convert_as_factor(X, Zona)
head(paciente_1_long)

## Miramos si tienen outlier
```

```
paciente_1_long%>%
group_by(Banda, Zona) %>%
identify_outliers(medida)

## Verificamos los supuestos de normalidad


paciente_1_long %>%
group_by(Banda, Zona) %>%
shapiro_test(medida)

## Miramos la homogenenidad de las varianzas


paciente_1_long %>%
group_by(Banda) %>%
levene_test(medida ~ Zona)

## Realizamos el anova de 2 vias

res.aov <- anova_test(data = paciente_1_long,
dv = medida, wid = X, between = Zona, within = Banda )
get_anova_table(res.aov) # aplica corrección automática

## Hacemos las comparaciones multiples

paciente_1_long %>%
group_by(Zona) %>%
pairwise_t_test(medida ~ Banda, paired = TRUE,
p.adjust.method = "holm") %>%
filter(p.adj < 0.05)

## Graficamos

ggbetweenstats(data = paciente_1, x = Zona, y = Bandas,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)

# -------------------------------------
# ANOVA  Facorial No Parametrico
# -------------------------------------
# Cargamos el paciente 1

datos <- read.csv("data_frame_Paciente_1.csv")

## Pasamos a formato largo los datos

datos2 <- datos2 %>%
  gather(key= "bandas", value= "score", X0, X1, X2,  X3, X4, X5, X6, X7, X8
, X9, X10, X11,
         X12,X13, X14, X15, X16, X17, X18, X19, X20, X21, X22, X23, X24) %>
%
  convert_as_factor(X, bandas)
```

```
## Miramos la media con la desviacion estandar

datos2 %>%
  group_by(Zona,bandas) %>%
  get_summary_stats(score, type = "mean_sd")

## Visualizacion de datos

bxp <- ggboxplot(data = datos2, x="bandas", y= "score",
                 color= "Zona", palette= "jco")

bxp

## Miramos los outlier

datos2%>%
  group_by(bandas, Zona) %>%
  identify_outliers("score")


## Normalidad

datos2 %>%
  group_by(Zona ,bandas) %>%
  shapiro_test(score)

## Grafico qqplot
ggqqplot(datos2, "score", ggtheme= theme_bw())+
   facet_grid(bandas~ Zona, labeller="label_both")

## realizamos el anova factorial no parametrico


res.aov <- anova_test(data = datos2, dv= score, wid = X,
                      within = c(Zona, bandas))

get_anova_table(res.aov)

## Pruebas post hoc


one.way <- datos2 %>%
  group_by(bandas) %>%
  anova_test(dv= score, wid= X, within= Zona) %>%
  adjust_pvalue(method = "bonferroni")

one.way

# -------------------------------------
# ANOVA  de 2 vias mixto de paciente 2.
# -------------------------------------
# Cargamos el paciente 2

## Cargamos las librerias
```

```
library(tidyverse) # manipulación y visualización de datos
library(ggpubr) # gráficos sencillos
library(ggstatsplot) # gráficos listos para publicar
library(openintro) # datos de ejemplo
library(rstatix) # pruebas P/NP y estadísticos con tuberías.
library(DescTools) # prueba robusta de Yuen
library(WRS2) # tama

## Cargamos el paciente 2

Paciente_2 <- read.csv("Paciente_2")

## Pasamos a formato largo los datos

library(dplyr)
paciente_2_long<- Paciente_2 %>%
  gather(key = "Banda", value = "medida",
         Banda_0,Banda_1, Banda_2, Banda_3, Banda_4, Banda_5, Banda_6, Band
a_7, Banda_8,Banda_9,Banda_10,
Banda_11, Banda_12, Banda_13, Banda_14, Banda_15, Banda_16, Banda_17, Banda
_18, Banda_19, Banda_20,
Banda_21, Banda_22, Banda_23, Banda_24) %>%
  convert_as_factor(X, Zona)
head(paciente_2_long)

## Estadisticos basicos

paciente_2_long %>%
group_by(Banda, Zona) %>%
get_summary_stats(medida, type = "mean_sd")

## Miramos si tienen outlier

paciente_2_long %>%
  group_by(Banda, Zona) %>%
  identify_outliers(medida)

##  Verificamos los supuestos de normalidad

paciente_2_long %>%
group_by(Banda, Zona) %>%
shapiro_test(medida)

## Miramos la homocedasticidad

paciente_2_long %>%
group_by(Banda) %>%
levene_test(medida ~ Zona)

## Realizamos el anova de 2 vias mixto
res.aov <- anova_test(data = paciente_2_long,
dv = medida, wid = X, between = Zona, within = Banda )
get_anova_table(res.aov) # aplica corrección automática
```

```r
## hacemos las pruebas post hoc
paciente_2_long %>%
group_by(Zona) %>%
pairwise_t_test(medida ~ Banda, paired = TRUE,
p.adjust.method = "holm") %>%
filter(p.adj < 0.05)

## Graficamos

ggbetweenstats(data = Paciente_2, x = Zona, y = Bandas,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)


# ------------------------------------
# ANOVA  Facorial No Parametrico
# ------------------------------------
# Cargamos el paciente 2

library(cowplot)
library(DescTools)
library(stringr)
library(MASS)
library(reshape)
library(tibble)
library(lmtest)
library(splitstackshape)
library(emmeans)
library(nortest)
library(car)
library(agricolae)
library(ggplot2)
library(ggpubr)
library(ART)
library(ARTool)
library(tidyverse) # manipulación y visualización de datos
library(ggpubr) # gráficos sencillos
library(ggstatsplot) # gráficos listos para publicar
library(openintro) # datos de ejemplo
library(rstatix) # pruebas P/NP y estadísticos con tuberías.
library(DescTools) # prueba robusta de Yuen
library(WRS2)
library(doBy)# tamaño de efecto robustos
library(onewaytests)

## Cargamos al paciente 2

paciente <- read.csv("data_frame_Paciente_2.csv")

## Pasamos a formato largo los datos

library(dplyr)
paciente_2_long<- Paciente_2 %>%
```

```
  gather(key = "Banda", value = "medida",
         Banda_0,Banda_1, Banda_2, Banda_3, Banda_4, Banda_5, Banda_6, Band
a_7, Banda_8,Banda_9,Banda_10,
Banda_11, Banda_12, Banda_13, Banda_14, Banda_15, Banda_16, Banda_17, Banda
_18, Banda_19, Banda_20,
Banda_21, Banda_22, Banda_23, Banda_24) %>%
  convert_as_factor(X, Zona)
head(paciente_2_long)

## Calculamos los estadisticos basicos

paciente_2_long %>%
group_by(Banda, Zona) %>%
get_summary_stats(medida, type = "mean_sd")

# Miramos los outlier

paciente_2_long%>%
group_by(Banda, Zona) %>%
identify_outliers(medida)

## Miramos la normalidad

paciente_2_long %>%
group_by(Banda, Zona) %>%
shapiro_test(medida)

## Hacemos el ARTools para el anova factorial no parametrico

paciente_2_long$Banda = as.factor(paciente_2_long$Banda)
paciente_2_long$Zona=as.factor(paciente_2_long$Zona)
str(paciente_2_long)

library(ARTool)

model = art(medida ~ Banda + Zona + Banda:Zona,
               data = paciente_2_long)
model

### Comparaciones post-hoc para interacciones en un modelo bidireccional


marginal = art.con(model, "Banda:Zona", adjust="none")

marginal
### eta parcial al cuadrado

Result = anova(model)

Result$part.eta.sq = with(Result, `Sum Sq`/(`Sum Sq` + `Sum Sq.res`))

Result
```

```r
# ------------------------------------
# ANOVA  de 2 vias mixto de paciente 3.
# ------------------------------------
# Cargamos el paciente 3

## Cargamos las librerias
library(tidyverse) # manipulación y visualización de datos
library(ggpubr) # gráficos sencillos
library(ggstatsplot) # gráficos listos para publicar
library(openintro) # datos de ejemplo
library(rstatix) # pruebas P/NP y estadísticos con tuberías.
library(DescTools) # prueba robusta de Yuen
library(WRS2)

Paciente_3 <- read.csv("mi_Paciente_3.csv")

## Pasar a formato largo los datos

library(dplyr)
paciente_3_long<- paciente_3_1 %>%
  gather(key = "Banda", value = "medida",
         Banda_0,Banda_1, Banda_2, Banda_3, Banda_4, Banda_5, Banda_6, Band
a_7, Banda_8,Banda_9,Banda_10,
Banda_11, Banda_12, Banda_13, Banda_14, Banda_15, Banda_16, Banda_17, Banda
_18, Banda_19, Banda_20,
Banda_21, Banda_22, Banda_23, Banda_24) %>%
  convert_as_factor(X, Zona)
head(paciente_3_long)

## Miramos la media con la desviacionestandar

paciente_3_long %>%
group_by(Banda, Zona) %>%
get_summary_stats(medida, type = "mean_sd")

## Miramos si tenenemos outliers

paciente_3_long %>%
  group_by(Banda, Zona) %>%
  identify_outliers(medida)

## Miramos la normalidad delos datos

paciente_3_long %>%
group_by(Banda, Zona) %>%
shapiro_test(medida)

## Hacemos el anova de 2 vias mixto
res.aov <- anova_test(data = paciente_3_long,
dv = medida, wid = X, between = Zona, within = Banda )
get_anova_table(res.aov) # aplica corrección automática

## Comparaciones Post hoc
```

```
library(dplyr)
paciente_3_long%>%
group_by(Zona) %>%
pairwise_t_test(medida ~ Banda, paired = TRUE,
p.adjust.method = "holm") %>%
filter(p.adj < 0.05)

## Graficamos

ggbetweenstats(data = Paciente_3, x = Zona, y = Bandas,
results.subtitle = T, messages = F, var.equal = T, p.adjust.method = "holm"
)

# ------------------------------------
# ANOVA  Facorial No Parametrico
# ------------------------------------
# Cargamos el paciente 3 y las librerias

library(cowplot)
library(DescTools)
library(stringr)
library(MASS)
library(reshape)
library(tibble)
library(lmtest)
library(splitstackshape)
library(emmeans)
library(nortest)
library(car)
library(agricolae)
library(ggplot2)
library(ggpubr)
library(ART)
library(ARTool)
library(tidyverse) # manipulación y visualización de datos
library(ggpubr) # gráficos sencillos
library(ggstatsplot) # gráficos listos para publicar
library(openintro) # datos de ejemplo
library(rstatix) # pruebas P/NP y estadísticos con tuberías.
library(DescTools) # prueba robusta de Yuen
library(WRS2)
library(doBy)# tamaño de efecto robustos
library(onewaytests)

Paciente_3 <- read.csv("mi_Paciente_3.csv")

## Pasamso a formato largo los datos

library(dplyr)
paciente_3_long<- Paciente_3 %>%
gather(key = "Banda", value = "medida",
Banda_0,Banda_1, Banda_2, Banda_3, Banda_4, Banda_5, Banda_6, Banda_7, Band
a_8, Banda_9, Banda_10,
Banda_11, Banda_12, Banda_13, Banda_14, Banda_15, Banda_16, Banda_17, Banda
```

```
_18, Banda_19, Banda_20,
Banda_21, Banda_22, Banda_23, Banda_24)
head(paciente_3_long)

## Miramos la media conla desviacion estandar

paciente_3_long %>%
group_by(Banda, Zona) %>%
get_summary_stats(medida, type = "mean_sd")

## Miramos la normalidad

paciente_3_long%>%
group_by(Banda, Zona) %>%
identify_outliers(medida)

paciente_3_long %>%
group_by(Banda, Zona) %>%
shapiro_test(medida)

## Realizamos el anova facorial no parametrico

paciente_3_long$Banda = as.factor(paciente_3_long$Banda)
paciente_3_long$Zona=as.factor(paciente_3_long$Zona)
str(paciente_3_long)

library(ARTool)

model = art(medida ~ Banda + Zona + Banda:Zona,
                data = paciente_3_long)
model
anova(model)

## Comparaciones post-Hoc

marginal = art.con(model, "Banda:Zona", adjust="none")

marginal

## miramos los resultados

Result = anova(model)

Result$part.eta.sq = with(Result, `Sum Sq`/(`Sum Sq` + `Sum Sq.res`))

Result


# -------------------------------------
# ANOVA  de 2 vias mixto de paciente 4.
# -------------------------------------
# Cargamos el paciente 4

Paciente_4 <- read.csv("miPaciente_4.csv")
```

```
## Pasamos a formato largo el paciente

library(dplyr)
paciente_4_long<- Paciente_4 %>%
  gather(key = "Banda", value = "medida",
         Banda_0,Banda_1, Banda_2, Banda_3, Banda_4, Banda_5, Banda_6, Band
a_7, Banda_8,Banda_9,Banda_10,
Banda_11, Banda_12, Banda_13, Banda_14, Banda_15, Banda_16, Banda_17, Banda
_18, Banda_19, Banda_20,
Banda_21, Banda_22, Banda_23, Banda_24) %>%
  convert_as_factor(X, Zona)
head(paciente_4_long)

## miramos la media conla desv iacionestandar

paciente_4_long %>%
group_by(Banda, Zona) %>%
get_summary_stats(medida, type = "mean_sd")

## Miramos la normalidad

paciente_4_long %>%
group_by(Banda) %>%
levene_test(medida ~ Zona)

## Realizamos el anova
res.aov <- anova_test(data = paciente_4_long,
dv = medida, wid = X, between = Zona, within = Banda )
get_anova_table(res.aov) # aplica corrección automática

## Coparacion multiple spost hoc

library(dplyr)
paciente_4_long%>%
group_by(Zona) %>%
pairwise_t_test(medida ~ Banda, paired = TRUE,
p.adjust.method = "holm") %>%
filter(p.adj < 0.05)


# -------------------------------------
# ANOVA  Facorial No Parametrico
# -------------------------------------
# Cargamos el paciente 4 y las librerias

Paciente_4 <- read.csv("miPaciente_4.csv")

## pasamos a formato largo el paciente 4

library(dplyr)
paciente_4_long<- Paciente_4 %>%
  gather(key = "Banda", value = "medida",
         Banda_0,Banda_1, Banda_2, Banda_3, Banda_4, Banda_5, Banda_6, Band
```

```
a_7, Banda_8,Banda_9,Banda_10,
Banda_11, Banda_12, Banda_13, Banda_14, Banda_15, Banda_16, Banda_17, Banda
_18, Banda_19, Banda_20,
Banda_21, Banda_22, Banda_23, Banda_24) %>%
  convert_as_factor(X, Zona)
head(paciente_4_long)

## Miramos la media conla desiacion estandar

paciente_4_long %>%
group_by(Banda, Zona) %>%
get_summary_stats(medida, type = "mean_sd")

## Miramos la normalidad
paciente_4_long %>%
group_by(Banda, Zona) %>%
shapiro_test(medida)

## Hacemos la prueba ARTools
library(ARTool)

model = art(medida ~ Banda + Zona + Banda:Zona,
               data = paciente_4_long)
model

anova(model)

## Comparaciones Post hoc

marginal = art.con(model, "Banda:Zona", adjust="none")

marginal

# -------------------------------------
# ANOVA  de 2 vias mixto de paciente 5.
# -------------------------------------
# Cargamos el paciente 5

Paciente_5 <- read.csv("miPaciente_5.csv")



## Pasamos a formato largo


library(dplyr)
paciente_5_long<- Paciente_5 %>%
  gather(key = "Banda", value = "medida",
         Banda_0,Banda_1, Banda_2, Banda_3, Banda_4, Banda_5, Banda_6, Band
a_7, Banda_8,Banda_9,Banda_10,
Banda_11, Banda_12, Banda_13, Banda_14, Banda_15, Banda_16, Banda_17, Banda
_18, Banda_19, Banda_20,
Banda_21, Banda_22, Banda_23, Banda_24) %>%
  convert_as_factor(X, Zona)
```

```
head(paciente_5_long)

## Miramos los outlier

paciente_5_long %>%
group_by(Banda, Zona) %>%
get_summary_stats(medida, type = "mean_sd")

## Mirmos la normalidad de los datos

paciente_5_long %>%
group_by(Banda, Zona) %>%
shapiro_test(medida)

## Realizamos el anova de 2 vias mixto

res.aov <- anova_test(data = paciente_5_long,
dv = medida, wid = X, between = Zona, within = Banda )
get_anova_table(res.aov) # aplica corrección automática

## Hacems las comparaciones porthoc

library(dplyr)
paciente_5_long%>%
group_by(Zona) %>%
pairwise_t_test(medida ~ Banda, paired = TRUE,
p.adjust.method = "holm") %>%
filter(p.adj < 0.05)


# -------------------------------------
# ANOVA  Facorial No Parametrico
# -------------------------------------
# Cargamos el paciente 5

Paciente_5 <- read.csv("miPaciente_5.csv")

## Pasamos a formato largo los datos

library(dplyr)
paciente_5_long<- Paciente_5 %>%
gather(key = "Banda", value = "medida",
Banda_0,Banda_1, Banda_2, Banda_3, Banda_4, Banda_5, Banda_6, Banda_7, Band
a_8, Banda_9, Banda_10,
Banda_11, Banda_12, Banda_13, Banda_14, Banda_15, Banda_16, Banda_17, Banda
_18, Banda_19, Banda_20,
Banda_21, Banda_22, Banda_23, Banda_24)
head(paciente_5_long)

## Miramos la media con la desviacion estandar

paciente_5_long %>%
group_by(Banda, Zona) %>%
get_summary_stats(medida, type = "mean_sd")
```

```
## Miramos la normalidad

paciente_5_long %>%
group_by(Banda, Zona) %>%
shapiro_test(medida)

## Realizamos el ARTools

paciente_5_long$Banda = as.factor(paciente_5_long$Banda)
paciente_5_long$Zona=as.factor(paciente_5_long$Zona)
str(paciente_5_long)

library(ARTool)

model = art(medida ~ Banda + Zona + Banda:Zona,
```