



**UNIVERSIDAD
DE ANTIOQUIA**

**IMPACTO DE LOS MÉTODOS DE
ELIMINACIÓN DE ARTEFACTOS SOBRE LA
REPRODUCIBILIDAD EN LAS MÉTRICAS
OBTENIDAS EN IMÁGENES DE RESONANCIA
MAGNÉTICA FUNCIONAL EN ESTADO DE
REPOSO**

Autor(es)

Servio Leonel Mera Jiménez

Universidad de Antioquia

Facultad de ingeniería, Maestría en Ingeniería

Medellín, Colombia

2021



Impacto de los métodos de eliminación de artefactos sobre la reproducibilidad en las métricas obtenidas en imágenes de resonancia magnética funcional en estado de reposo

Servio Leonel Mera Jiménez

Tesis o trabajo de investigación presentada(o) como requisito parcial para optar al título de:

Magister en Ingeniería con Énfasis en Bioingeniería

Asesores (a):

Ph.D. John Fredy Ochoa Gomez

Línea de Investigación:

Bioingeniería

Grupo de Investigación:

Grupo de Investigación en Bioinstrumentación e Ingeniería Clínica

Universidad de Antioquia

Facultad de Ingeniería, Maestría en Ingeniería

Medellín, Colombia

2021.

Agradecimientos

A mi madre Esperanza

Por haberme brindado todo su apoyo, en todo momento. Por darme su aliento su fortaleza y paciencia, además de todas las virtudes que me ha inculcado desde pequeño, porque sin ella, no habría culminado esta pequeña etapa de un gran sueño que apenas comienza.

A mi hermana Adriana

Por haberme brindado su apoyo incondicional, sin importar las adversidades que tuvimos que enfrentar. Por ser la persona que me acompañó en este largo trayecto sin convalecer un solo día y, por último, por confiar en mí en todo momento sin siquiera dudarlo una sola ocasión.

Contenido

1. Introducción	15
2. Marco teórico	18
2.1. Resonancia magnética funcional	18
2.2. Señal BOLD	18
2.3. Métodos de limpieza de la señal BOLD	18
2.3.1. Regresión lineal basado en parámetros de posición	19
2.3.2. Regresión global	19
2.3.3. Depuración de volúmenes	19
2.3.4. Corrección basada en el análisis de componentes principales (CompCor)	19
2.3.5. Eliminación de ruido basado en el análisis de componentes principales (ICA)	20
2.4. Clasificación de componentes independientes mediante técnicas de aprendizaje de máquina	20
2.4.1. Perceptrón multicapa	20
2.4.2. Red neuronal convolucional	21
2.5. Métricas de conectomas funcionales	22
2.5.1. Amplitud de fluctuación de baja frecuencia (ALFF) y amplitud fluctuación de baja frecuencia fraccional (fALFF)	22
2.5.2. Homogeneidad regional (ReHo)	22
2.5.3. Conectividad funcional (FC)	23
2.6. Teoría de grafos	23
2.7. Reproducibilidad	24
2.7.1. Coeficiente de correlación intraclases (ICC)	24
3. Objetivos	25
4. Publicaciones derivadas	26
5. Implementación de un algoritmo para la clasificación automática de componentes artefactuales obtenidos por ICA en datos fMRI	28
5.1. Metodología	30
5.1.1. Sujetos y componentes independientes	30
5.1.2. Adquisición de los datos fMRI	30
5.1.3. Clasificación de las componentes	30
5.1.4. Extracción de características de las series de tiempo y densidad espectral de potencia	33
5.1.5. Diseño de las redes neuronales basadas en MLP	34

5.1.6.	Reducción de dimensiones	35
5.1.7.	Entrenamiento de las redes MLP	35
5.1.8.	Reducción de volúmenes de los mapas espaciales	35
5.1.8.1.	Reducción por PCA	36
5.1.8.2.	Extracción de bordes	37
5.1.8.3.	Reducción en cortes medios	37
5.1.8.4.	Reducción por patrones binarios consecutivos	38
5.1.8.5.	Reducción de patrones binarios consecutivos y filtrado no lineal	41
5.1.9.	Escalado de imágenes	41
5.1.10.	Diseño de las redes CNN	41
5.1.11.	Entrenamiento redes CNN	45
5.2.	Resultados	46
5.2.1.	Clasificación de componentes independientes	46
5.2.2.	Reducción de dimensiones	46
5.2.3.	Entrenamiento de las redes MLP	48
5.2.4.	Reducción de volúmenes	51
5.2.5.	Entrenamiento de las redes CNN	56
5.2.5.1.	Entrenamientos con bordes	56
5.2.5.2.	Entrenamiento con imágenes escaladas	57
5.3.	Discusión	63
5.4.	Conclusión	65
6.	Implementar un flujo de procesamiento automatizado para imágenes fMRI que integre las herramientas de limpieza de señal BOLD y métricas de conectoma	66
6.1.	Metodología	67
6.1.1.	Software	67
6.1.2.	Preprocesamiento	69
6.1.4.	Métricas	74
6.2.	Resultados	77
6.3.	Discusión	78
6.4.	Conclusión	79
7.	Determinar el impacto sobre la reproducibilidad de las métricas de conectoma bajo los diferentes métodos de limpieza en la señal BOLD	80
7.1.	Metodología	80
7.1.1.	Sujetos	80
7.2.	Resultados	82
7.3.	Discusión	93
7.4.	Conclusión	94

8. Conclusiones	95
Bibliografía	97
9. Anexos	104

Lista de Tablas

Tabla I Arquitecturas de las redes neuronales artificiales	34
Tabla II Arquitecturas de las redes neuronales convolucionales de una sola imagen de entrada.....	42
Tabla III Arquitecturas de las redes neuronales convolucionales multimodales para tres imágenes de entrada	43
Tabla IV Porcentaje de acierto de los modelos en la clasificación	51
Tabla V Peso promedio por componente.....	52
Tabla VI Diferencia de medias mediante la prueba la kruskal-wallis sobre las imágenes con y sin contornos.....	57
Tabla VII Diferencia de medias mediante la prueba kruskal-wallis entre imágenes originales escaladas y comparación entre la precisión por validación cruzada y validación por datos de prueba	57
Tabla VIII Máxima precisión de clasificación alcanzada por los modelos entrenados con diferentes imágenes	58
Tabla IX Máxima precisión de clasificación alcanzada por los modelos multimodales entrenados con las tres imágenes	59
Tabla X 48 Esquemas de procesamiento generados por la combinación de las diferentes técnicas de eliminación de artefactos.....	77
Tabla XI Regiones anatómicas usadas en la parcelación de los volúmenes cerebrales y las abreviaciones utilizadas en las gráficas de los resultados	82
Tabla XII Pipelines de mayor reproducibilidad por métrica y regiones asociadas.....	93

Lista de Figuras

Figura 1	Representación básica de un perceptrón y de una arquitectura MLP.....	21
Figura 2	Estructura base de una red neuronal convolucional combinada con un red neuronal convencional para un proceso de clasificación.	21
Figura 3	Conectividad funcional modelada como un grafo.....	23
Figura 4	Mapa espacial de una componente independiente vistas en cortes axiales.31	
Figura 5	Mapa espacial de una componente independiente en cortes axiales..	31
Figura 6	Mapa espacial de una componente independiente en cortes axiales.	31
Figura 7	Series de tiempo de diferente naturaleza..	32
Figura 8	Densidad espectral de diferente naturaleza.....	33
Figura 9	Reducción del volumen mediante análisis de componentes principales..	36
Figura 10	Reducción del volumen en 3 cortes.	38
Figura 11	Proceso de reducción del volumen en una imagen RGB axial..	39
Figura 12	Proceso de reducción del volumen en una imagen RGB coronal. l.....	40
Figura 13	Proceso de reducción del volumen en una imagen RGB sagital.....	40
Figura 14	Matriz de coeficientes de correlación entre las 27 características espectro-temporales.	47
Figura 15	Porcentaje de varianza explicada en función de número de variables latentes obtenidos del análisis de componentes principales de las 27 características espectro-temporales.	47
Figura 16	Promedio del porcentaje de precisión de clasificación en cada una de las variables validadas en los cuatro modelos.....	48
Figura 17	Comportamiento del entrenamiento de los 4 modelos con todas las características del primer grupo de sujetos.....	49
Figura 18	Comportamiento del entrenamiento de los 4 modelos con todas las características para el segundo grupo de sujetos..	50
Figura 19	Precisión en los 4 modelos entrenados con diferente número de componentes principales.....	50
Figura 20	Comportamiento del entrenamiento de los 4 modelos con las cinco características de mayor precisión para el segundo grupo de sujetos.	51
Figura 21	Resultados de cuatro componentes independientes clasificados como actividad neuronal.	53
Figura 22	Resultados de cuatro componentes independientes clasificados como actividad neuronal..	53
Figura 23	Resultados de cuatro componentes independientes clasificados como actividad neuronal.).....	54
Figura 24	Resultados de cuatro componentes independientes clasificados como ruido.....	55
Figura 25	Resultados de cuatro componentes independientes clasificados como ruido.	55

Figura 26 Resultados de cuatro componentes independientes clasificados como ruido.	56
Figura 27 Gráfico de caja y bigotes de precisión en validación cruzada basada en diferentes métodos de reducción de imagen con y sin contornos.	60
Figura 28 Gráfico de caja y bigotes de precisión en el conjunto de prueba basado en diferentes métodos de reducción de imagen con y sin contornos.	61
Figura 29 Diagrama de caja y bigotes de precisión en el conjunto de prueba en función de los diferentes modelos.....	62
Figura 30 Precisión promedio en validación cruzada y datos de entrenamiento en función de los métodos de reducción para los 7 grupos de entrenamiento.	62
Figura 31 Conexión entre funciones de Nipype.	67
Figura 32 Diagrama principal del flujo de preprocesamiento utilizado para reducción de ruido.	69
Figura 33 Flujo de del preprocesamiento hasta la normalización del sujeto con diferentes grados de libertad en la corrección de movimiento.....	71
Figura 34 Flujo de del procesamiento de suavizado y filtrado de la imagen normalizada.....	72
Figura 35 Eliminación automática de artefactos mediante RPBC en las componentes independientes.	73
Figura 36 Flujo de procesamiento de TCompCor	73
Figura 37 Flujo de procesamiento regresión de señal global.....	74
Figura 38 Diagrama principal del flujo de preprocesamiento utilizado para reducción de ruido.	76
Figura 39 ICC sobre la métrica de ReHo con número de vecinos 7, 19 y 27 en las diferentes regiones para los 6 pipelines de mayor reproducibilidad.	85
Figura 40 ICC sobre la métrica de ReHo con número de vecinos 7, 19 y 27 para todos los pipelines en la red modo defecto.	86
Figura 41 ICC sobre las métricas de ALFF y fALFF en las bandas 2 y 3 en las diferentes regiones para los 6 pipelines de mayor reproducibilidad.....	87
Figura 42 ICC sobre las métricas de ALFF y fALFF en las bandas 4 y 5 en las diferentes regiones para los 6 pipelines de mayor reproducibilidad.....	88
Figura 43 ICC sobre la métrica de conectividad funcional para el pipeline de mayor reproducibilidad.....	89
Figura 44 ICC sobre la métrica de conectividad funcional para el pipeline de mayor reproducibilidad en el atlas msdl.	90
Figura 44 ICC sobre la métrica de agrupamiento promedio en los 48 pipelines diferentes.....	90
Figura 45 ICC sobre la métrica de camino característico en los 48 pipelines diferentes.	90
Figura 46 ICC sobre la métrica de grado promedio en los 48 pipelines diferentes.	91
Figura 47 ICC sobre las métricas de agrupamiento y grado en las diferentes regiones para los 6 pipelines de mayor reproducibilidad.....	92

Figura 48 ICC sobre la métrica de ReHo con número de vecinos 7, 19 y 27 en las diferentes regiones para todos los pipelines.	107
Figura 49 ICC sobre las métricas de ALFF y fALFF en banda 2 en las diferentes regiones para todos los pipelines.	108
Figura 50 ICC sobre las métricas de ALFF y fALFF en banda 3 en las diferentes regiones para todos los pipelines.	109
Figura 51 ICC sobre las métricas de ALFF y fALFF en banda 4 en las diferentes regiones para todos los pipelines.	110
Figura 52 ICC sobre las métricas de ALFF y fALFF en banda 5 en las diferentes regiones para todos los pipelines.	111
Figura 53 ICC sobre la métrica de conectividad funcional para el pipeline 0.	112
Figura 54 ICC sobre la métrica de conectividad funcional para el pipeline 1.	112
Figura 55 ICC sobre la métrica de conectividad funcional para el pipeline 2.	113
Figura 56 ICC sobre la métrica de conectividad funcional para el pipeline 3.	113
Figura 57 ICC sobre la métrica de conectividad funcional para el pipeline 4.	114
Figura 58 ICC sobre la métrica de conectividad funcional para el pipeline 5.	114
Figura 59 ICC sobre la métrica de conectividad funcional para el pipeline 6.	115
Figura 60 ICC sobre la métrica de conectividad funcional para el pipeline 7.	115
Figura 61 ICC sobre la métrica de conectividad funcional para el pipeline 8.	116
Figura 62 ICC sobre la métrica de conectividad funcional para el pipeline 9.	116
Figura 63 ICC sobre la métrica de conectividad funcional para el pipeline 10.	117
Figura 64 ICC sobre la métrica de conectividad funcional para el pipeline 11.	117
Figura 65 ICC sobre la métrica de conectividad funcional para el pipeline 12.	118
Figura 66 ICC sobre la métrica de conectividad funcional para el pipeline 13.	118
Figura 67 ICC sobre la métrica de conectividad funcional para el pipeline 14.	119
Figura 68 ICC sobre la métrica de conectividad funcional para el pipeline 15.	119
Figura 69 ICC sobre la métrica de conectividad funcional para el pipeline 16.	120
Figura 70 ICC sobre la métrica de conectividad funcional para el pipeline 17.	120
Figura 71 ICC sobre la métrica de conectividad funcional para el pipeline 18.	121
Figura 72 ICC sobre la métrica de conectividad funcional para el pipeline 19.	121
Figura 73 ICC sobre la métrica de conectividad funcional para el pipeline 20.	122
Figura 74 ICC sobre la métrica de conectividad funcional para el pipeline 21.	122
Figura 75 ICC sobre la métrica de conectividad funcional para el pipeline 22.	123
Figura 76 ICC sobre la métrica de conectividad funcional para el pipeline 23.	123
Figura 77 ICC sobre la métrica de conectividad funcional para el pipeline 24.	124
Figura 78 ICC sobre la métrica de conectividad funcional para el pipeline 25.	124
Figura 79 ICC sobre la métrica de conectividad funcional para el pipeline 26.	125
Figura 80 ICC sobre la métrica de conectividad funcional para el pipeline 27.	125
Figura 81 ICC sobre la métrica de conectividad funcional para el pipeline 28.	126

Figura 82 ICC sobre la métrica de conectividad funcional para el pipeline 29	126
Figura 83 ICC sobre la métrica de conectividad funcional para el pipeline 30	127
Figura 84 ICC sobre la métrica de conectividad funcional para el pipeline 31	127
Figura 85 ICC sobre la métrica de conectividad funcional para el pipeline 32	128
Figura 86 ICC sobre la métrica de conectividad funcional para el pipeline 33	128
Figura 87 ICC sobre la métrica de conectividad funcional para el pipeline 34	129
Figura 88 ICC sobre la métrica de conectividad funcional para el pipeline 35	129
Figura 89 ICC sobre la métrica de conectividad funcional para el pipeline 36	130
Figura 90 ICC sobre la métrica de conectividad funcional para el pipeline 37	130
Figura 91 ICC sobre la métrica de conectividad funcional para el pipeline 38	131
Figura 92 ICC sobre la métrica de conectividad funcional para el pipeline 39	131
Figura 93 ICC sobre la métrica de conectividad funcional para el pipeline 40	132
Figura 94 ICC sobre la métrica de conectividad funcional para el pipeline 41	132
Figura 95 ICC sobre la métrica de conectividad funcional para el pipeline 42	133
Figura 96 ICC sobre la métrica de conectividad funcional para el pipeline 43	133
Figura 97 ICC sobre la métrica de conectividad funcional para el pipeline 44	134
Figura 98 ICC sobre la métrica de conectividad funcional para el pipeline 45	134
Figura 99 ICC sobre la métrica de conectividad funcional para el pipeline 46	135
Figura 100 ICC sobre la métrica de conectividad funcional para el pipeline 47	135
Figura 101 ICC sobre la métrica de conectividad funcional para el pipeline 48	136
Figura 102 ICC sobre las métricas de: grado de asortatividad con el coeficiente de correlación de Pearson, agrupamiento promedio, índice de Estrada, transitividad, conectividad promedio por nodo, camino característico y valor propio.	137
Figura 103 ICC sobre las métricas de centralidad de Katz e intermediación en las diferentes regiones para todos los pipelines.	138
Figura 104 ICC sobre las métricas de grado promedio de los vecinos y agrupamiento en las diferentes regiones para todos los pipelines.	139
Figura 105 ICC sobre las métricas de grado de página e hits de autoridad en las diferentes regiones para todos los pipelines.	140
Figura 106 ICC sobre las métricas de hits de concentrador y grado de centralidad en las diferentes regiones para todos los pipelines.	141
Figura 107 ICC sobre las métricas de centralidad de proximidad y de fracción de posibles cuadrados en las diferentes regiones para todos los pipelines.	142
Figura 108 ICC sobre las métricas de número del núcleo y número máximo de cliques en las diferentes regiones para todos los pipelines.	143
Figura 109 ICC sobre las métricas de clique de mayor tamaño y número de triángulos en las diferentes regiones para todos los pipelines.	144
Figura 110 ICC sobre las métricas de color y grado en las diferentes regiones para todos los pipelines.	145
Figura 111 ICC sobre las métricas centralidad de vector propio y centralidad de carga en las diferentes regiones para todos los pipelines.	146

Lista de Anexos

A 1. Métricas adicionales derivadas de la teoría de grafos.....	104
B 2. Resultados completos de todas las métricas	107
B 3. ICC sobre la métrica de ReHo en todos los pipelines	107
B 4. ICC sobre las métricas de ALFF y fALFF en todos los pipelines.....	108
B 5. ICC sobre la métrica de conectividad funcional en todos los pipelines	112
B 6. ICC sobre las métricas de grafos globales en todos los pipelines.....	137
B 7. ICC sobre las métricas de grafos por región en todos los pipelines.....	137

Glosario

fMRI: Resonancia magnética funcional.

rs-fMRI o R-fMRI: Resonancia magnética funcional en estado de reposo.

BOLD: Dependencia del nivel de oxigenación en la sangre.

PCA: Análisis de componentes principales.

ALFF: Amplitud de fluctuación de baja frecuencia.

fALFF: Amplitud de fluctuación de baja frecuencia fraccional.

ReHo: Homogeneidad regional

FC: Conectividad funcional.

ICC: Coeficiente de correlación intraclases.

DCw: Grados de centralidad ponderado

ECw: Grados de valores propios ponderado.

GLM: Modelo lineal general

CompCor: Corrección de ruido basado en componentes

tCompCor: Corrección de ruido basado en componentes en las series de tiempo.

ICA: Análisis de componentes independientes.

ML: Aprendizaje de máquina.

DL: Aprendizaje profundo.

MLP: Perceptrón multicapa.

CNN: Red neuronal convolucional.

BIDS: Estructura de datos de imágenes cerebrales.

ICC: Coeficiente de correlación intraclases

Resumen

En las últimas décadas, la resonancia magnética funcional en estado de reposo se presenta como una de las herramientas más importantes para la exploración de la actividad neuronal, de la cual se han derivado muchas métricas de segregación e integración de la actividad cerebral y en este mismo sentido, se han desarrollado múltiples métodos de limpieza de la neuroimagen. No obstante, la reproducibilidad de las métricas es un problema que no ha sido abordado con detalle, por lo tanto, en este documento se estudia el impacto de los métodos de eliminación de artefactos sobre la reproducibilidad en dichas métricas. Inicialmente, se buscó integrar una nueva herramienta de eliminación de artefactos basada en la clasificación automática de las componentes independientes derivadas de la descomposición por el análisis de componentes independientes (ICA). Se usaron varias arquitecturas de redes neuronales. Las primeras se diseñaron con estructuras de perceptrón multicapa, las cuales alcanzaron una precisión superior al 97% con características espectro-temporales. Las segundas se generaron como redes neuronales convolucionales, donde se encontró una red con un porcentaje de 99.44, superando todos los métodos existentes basados en ICA. Además, estas últimas se implementaron con un nuevo método de reducción de volúmenes, el cual permitió reducir el peso de las imágenes hasta dos órdenes de magnitud.

Con los hallazgos realizados, se procedió a integrar todas las técnicas en un flujo totalmente automatizado. Se incluyeron las técnicas de procesamiento convencionales, los métodos de eliminación de artefactos, el nuevo método de clasificación y el cálculo de métricas. El desarrollo permitió calcular de manera ininterrumpida 2448 volúmenes de 17 participantes con tres secciones bajo 48 pipelines diferentes.

Finalmente, a los 2448 volúmenes se les calcularon las métricas de ReHo, ALFF, fALFF, conectividad funcional y 24 métricas derivadas de teoría de grafos como grado, agrupamiento, camino característico entre otros. Los resultados entre sesiones se utilizaron para calcular la reproducibilidad mediante el coeficiente de correlación intraclases (ICC). Se estimó que los métodos de eliminación de artefactos tienen un impacto diferente en la reproducibilidad para cada una de las métricas, lo cual fue de gran utilidad para establecer los pipelines de mejor desempeño para cada métrica. ReHo presentó mayor reproducibilidad con la combinación de ICA-AROMA y TcompCor. Las cuatro bandas de ALFF y fALFF presentaron un mejor comportamiento con la eliminación de artefactos a partir de la red neuronal convolucional. La conectividad funcional mostró mejor reproducibilidad con ICA-AROMA y, finalmente, las métricas derivadas de grafos mostraron tuvieron un mejor comportamiento frente a ICA-AROMA, la red neuronal, TcompCor, la regresión global y la depuración de volúmenes.

Palabras clave:

rs-fMRI , métricas, ICC, test-retest, pipeline, preprocesamiento.

Abstract

In the last decades, functional magnetic resonance imaging in resting state is presented as one of the most important tools for the exploration of neuronal activity, from which many metrics of segregation and integration of brain activity have been derived and, in the same way, multiple neuroimaging cleansing methods have been developed. However, the reproducibility of the metrics is a problem that has not been addressed in detail, therefore, this document studies the impact of artifact elimination methods on the reproducibility of these metrics. Initially, we sought to integrate a new artifact elimination tool based on the automatic classification of independent components derived from decomposition by independent component analysis (ICA). Various neural network architectures were used. The first ones were designed with multilayer perceptron structures, which reached a precision higher than 97% with spectro-temporal characteristics. The second was generated as convolutional neural networks, where a network with a percentage of 99.44 was found, surpassing all existing ICA-based methods. Besides, the latter was implemented with a new volume reduction method, which made it possible to reduce the weight of the images by up to two orders of magnitude.

With the findings made, all techniques were integrated into a fully automated flow. Conventional processing techniques, artifact removal methods, the new classification method, and metric calculation were included. The development made it possible to calculate 2448 volumes of 17 participants with three sections under 48 different pipelines.

Finally, the 2448 volumes were calculated with the ReHo, ALFF, fALFF metrics, functional connectivity, and 24 metrics derived from graph theory such as degree, clustering, path length, among others. The results between sessions were used to calculate reproducibility using the Intraclass Correlation Coefficient (ICC). The artifact removal methods were estimated to have a different impact on reproducibility for each of the metrics, which was very useful to establish the best performing pipelines for each metric. ReHo presented greater reproducibility with the combination of ICA-AROMA and TcompCor. In four bands ALFF and fALFF showed better behavior with the convolutional neural network. The functional connectivity showed better reproducibility with ICA-AROMA and, finally, the metrics derived from graphs showed better performance against ICA-AROMA, the neural network, TcompCor, global regression and scrubbing.

Key words:

rs-fMRI, metrics, ICC, test-retest, pipeline, preprocessing.

1. Introducción

Al interior de diferentes grupos e instituciones de investigación alrededor del mundo se está abordando una de las problemáticas de salud pública de mayor impacto a en el siglo XXI como lo son la enfermedad de Alzheimer y las patologías que afectan el sistema nervioso central [1], [2]. En esta medida, hoy en día existe un creciente interés en desarrollar y estudiar herramientas de exploración de la actividad neuronal, como lo son la electroencefalografía (EEG), magneto-encefalografía (MEG) e imágenes de resonancia magnética (MRI) entre las técnicas más modernas [3], sin embargo, esta última es ampliamente utilizada por su carácter no invasivo, por no contar con radiación ionizante y por la alta resolución espacial [4]. Adicionalmente, el enfoque hacia el paradigma en estado de reposo, conocido como resonancia magnética funcional en estado de reposo (rs-fMRI) ha entregado buenos resultados en cuanto a detección de circuitos neuronales, como es el caso de la red de modo defecto [5], permitiendo encontrar diferencias funcionales en patologías tan complejas como el Alzheimer [6] o el Parkinson [7], además de las múltiples métricas o medidas aplicadas sobre el sistema nervioso central y desarrolladas alrededor de dicho paradigma [8], [9]. Por otro lado, la universidad de Antioquia no es una excepción a estos estudios, contribuyendo con aportes desde los diversos grupos adscritos a la institución. De igual manera, desde el ámbito académico se está participando con diferentes enfoques científicos, como es el caso del análisis de las diferentes métricas de conectomas funcionales a escala local y meso escala en las imágenes de resonancia magnética funcional en estado de reposo (abreviado como rs-fMRI o R-fMRI), construidas a partir de la señal BOLD (Blood oxygen level dependent) la cual es una medida indirecta de la actividad neuronal generada en el estado de reposo, similar a la producida durante la ejecución en tareas, pero con la diferencia significativa de no depender de un paradigma, siendo su adquisición más fácil y aplicables a un rango mayor de sujetos de estudio [10].

El enfoque de rs-fMRI, busca encontrar la arquitectura intrínseca del cerebro para la comprensión y análisis de este y las patologías que lo afectan. Además, las métricas de conectomas funcionales, con imágenes de rs-fMRI son un método ampliamente utilizado, en gran parte por ser una técnica no invasiva con alta resolución espacial [11]. Nos obstante, una buena métrica, al igual que todo tipo de medida basada en el método científico, debe ser reproducible bajo las mismas consideraciones, es decir, la medida debe tener una fiabilidad alta para obtener conclusiones acertadas [12]. Para poder llegar a este punto es necesario extraer únicamente la señal BOLD correspondiente a la actividad neuronal generada en el estado de reposo. No obstante, uno de los problemas principales se da porque la actividad BOLD es una señal bastante afectada por diferentes factores como por ejemplo: los procesos metabólicos, procesos vasculares, ruido instrumental, ruido fisiológico y movimientos involuntarios, entre los que presentan mayor influencia sobre la misma, debido a que un pequeño movimiento durante la adquisición produce cambios del campo magnético de excitación sobre el plano de registro del volumen, generando estimaciones erróneas en la interpretación de la actividad neuronal [13]. Esta dificultad ha motivado el desarrollo de múltiples métodos de eliminación de artefactos, como la regresión lineal basada en parámetros de posición, regresión global, depuración de volúmenes, corrección basado en componentes principales (PCA) anatómicas y

temporales, denominados aCompCor y tCompCor respectivamente y la selección automática de componentes independientes (ICA) entre las metodologías más importantes [13].

Los actuales desarrollos han presentado avances significativos en cuanto a limpieza de la señal y métricas de conectomas funcionales, pero, desde un punto de vista más exhaustivo, no son muchos los estudios longitudinales existentes [14], limitando la posibilidad de evaluar la fiabilidad de las métricas de manera concreta y, en ese mismo sentido, restringiendo los estudios de la fiabilidad de las métricas posterior a un método de limpieza. Por tal motivo, nace el requerimiento de evaluar cuál es el impacto de los métodos de eliminación de artefactos sobre las métricas de conectomas funcionales. Igualmente, partiendo de la evaluación de las métricas de mayor confiabilidad, es posible establecer medidas para caracterizar diferencias clínicas entre grupos de sujetos de estudio. Asimismo, el impacto que ha tenido la resonancia magnética funcional en estado de reposo le ha dado una fuerte posición a esta técnica con especial enfoque en estudios clínicos y de desarrollo neurológico, no obstante, los avances en la confiabilidad y la validez de la conectómica funcional hasta ahora han retrasado la aplicación de estos métodos en la práctica. Por lo tanto, la alta confiabilidad es un estándar requerido tanto para investigación como para uso clínico [15].

Finalmente, siendo un poco más estrictos con los métodos de limpieza, se puede prestar especial atención al enfoque del análisis de componentes independientes, ya que es una de las técnicas más utilizadas para el estudio exploratorio de los datos de fMRI y ha demostrado ser una herramienta poderosa para identificar fuentes de fluctuaciones de diferentes naturalezas, tal como la actividad neuronal y el ruido artefactual. Una de las estrategias más empleadas para la eliminación de ruido, se centra ulteriormente a la descomposición, identificando de manera manual las componentes artefactuales y de interés; proceso necesario para la eliminación eficiente de elementos y, por ende, para la limpieza de los datos. Además, la identificación puede ser incluso utilizada para entrenar algoritmos de *machine learning* para la clasificación supervisada y/o para verificar los resultados en los no supervisados [16].

Siguiendo esta misma línea de ideas, en los últimos años el aprendizaje de máquina ha tenido bastante auge en todas las áreas de conocimiento, incluso en campos como las neurociencias, se han presentado aumentos bastante inusitados. El aprendizaje de máquina tiene cuatro potenciales funciones en el área de las neurociencias, a saber, crear soluciones a problemas de ingeniería, identificar variables predictivas, establecer puntos de referencia para modelos simples del cerebro y servir a sí mismo como modelo para el cerebro [17]. La eliminación de ruido hace parte del primero de los casos, partiendo de la premisa de que no es preciso ser un experto o de saber a detalle el proceso a analizar, siendo únicamente necesario disponer de un gran número de bases de datos de mediciones junto con la cantidad de interés asociada para la medición. Los algoritmos de aprendizaje de máquina pueden modelar automáticamente las relaciones e incluso discriminar entre diferentes características [17], como por ejemplo discernir entre ruido y actividad neuronal. “El aprendizaje automático se está convirtiendo en parte integral de la investigación biomédica moderna. Es importante destacar que han surgido enfoques que pueden integrar sobre

grandes conjuntos de datos biomédicos diferentes. Estos enfoques apuntan a cerrar la brecha entre nuestra capacidad de generar vastas cantidades de datos y nuestra comprensión de los sistemas biomédicos y, por lo tanto, reflejar la complejidad intrincada de la biología" [18].

Cabe resaltar que debido a los avances producidos en las últimas décadas existe un gran número de métricas realizadas sobre las imágenes de resonancia magnética, al igual que un gran número de métodos de limpieza de la señal BOLD, por lo que, considerar todas las métricas y todos los métodos sería un proyecto demasiado ambicioso, por estas circunstancias, solo se tomarán en consideración las principales métricas a escala local, como lo son el ALFF, fALFF, ReHo y la conectividad funcional como una métrica a meso escala. Análogo al caso de las métricas, para los métodos de limpieza artefactual, también se considerarán las más relevantes hasta la fecha, a sabiendas, la regresión lineal basado en parámetros de posición, regresión global, corrección basado en componentes principales (PCA) anatómicas y temporales, denominados aCompCor y tCompCor respectivamente, la selección automática de componentes independientes (ICA) y depuración de volúmenes con o sin interpolación.

Partiendo de estas discusiones, este proyecto se enfoca en establecer cuál es el impacto que tienen los métodos de preprocesamiento sobre la reproducibilidad en las métricas sobre las imágenes de resonancia magnética funcional. Sin embargo, en la primera parte de este nos centramos en buscar un algoritmo de eliminación de artefactos basado en ICA y en la clasificación de dichas componentes con estrategias apoyadas en redes neuronales artificiales, posteriormente, debido a la gran cantidad de herramientas de preprocesamiento como de métricas de conectomas, el proyecto se enfocó en integrar todas estas herramientas en un solo entorno de desarrollo que permitiera un flujo de procesamiento automatizado y, finalmente, con la disposición de estas herramientas se estudió la reproducibilidad bajo varios esquemas de procesamiento, llegando al objetivo principal de este proyecto.

El estudio mostró que es posible clasificar componentes independientes a partir de las características temporales, donde se obtuvo un poder de precisión superior al 97%. En este mismo orden de ideas, la integración de nuevos métodos de clasificación de componentes independientes, como las redes neuronales convolucionales, mostraron un poder de clasificación superando todos los métodos ya existentes. La precisión alcanzada fue superior al 99%, pese a que se implementó con un método propio de reducción de volúmenes con una inherente pérdida de información. Por otro lado, fue posible integrar todos estos nuevos desarrollos en un proceso automatizado, generándose 48 diferentes pipelines, los cuales generaron más de 2400 rs-fMRI procesadas, lo cual supone un procesamiento eficiente de los sujetos.

Finalmente, para cada una de las métricas estudiadas, fue posible establecer cual pipeline producía la mayor reproducibilidad, descartándose la idea de un pipeline general para todas las métricas, pero estableciéndose caminos propios por métrica los cuales son de vital interés para estudio posteriores en patologías específicas.

2. Marco teórico

2.1. Resonancia magnética funcional

Una de las herramientas más importantes para la exploración cerebral, tanto anatómica como funcionalmente, son las imágenes por resonancia magnética, las cuales, desde un concepto sencillo, se basa en alinear los núcleos atómicos del hidrógeno mediante un campo magnético externo de alta intensidad con la finalidad de hacer que estos resuenen excitados por una onda de radio frecuencia proporcional a la magnitud del campo magnético externo. La ausencia de la onda de radio frecuencia después de la excitación, hace que dichos núcleos liberen la energía adquirida en un proceso conocido como decaimiento de inducción libre, orientando los núcleos a su posición original en tiempos diferentes dependiendo del tejido donde se encuentran inmersos dichos núcleos, permitiendo generar los diferentes contrastes en las imágenes de resonancia magnética [11].

2.2. Señal BOLD

La funcionalidad del cerebro es un reflejo de la actividad neuronal al interior de este, es decir, es una descripción macroscópica de los potenciales de acción neuronal, procesos que demanda un mayor consumo de oxígeno para metabolizar la glucosa y continuar con el normal funcionamiento neuronal. El oxígeno al interior del cuerpo humano es transportado por la molécula de la hemoglobina, llamada oxihemoglobina cuando está oxigenada, y desoxihemoglobina cuando no está oxigenada. Las dos configuraciones de esta molécula cuentan con dos comportamientos magnéticos diferente, a sabiendas, diamagnetismo y paramagnetismo en la oxihemoglobina y desoxihemoglobina respectivamente. En este sentido, un proceso que implique un aumento focalizado de los potenciales de acción en alguna región específica del cerebro crea un aumento abrupto de oxihemoglobina y de la desoxihemoglobina en esta región, generando inhomogeneidades del campo magnético que afectan los tiempos de relajación y, por lo tanto, posibilitando ver las zonas de mayor actividad neuronal en las imágenes de resonancia magnética, que en este caso se conocen como imágenes de resonancia magnética funcional. Los diferentes tiempos de relajación generados por el oxígeno se conoce como señal BOLD o dependiente del nivel de oxigenación en la sangre.

2.3. Métodos de limpieza de la señal BOLD

Como se mencionó anteriormente, la señal BOLD es una medida indirecta de la actividad neuronal al interior de cerebro debido a la demanda de oxígeno que requieren las zonas de mayor actividad neuronal, sin embargo, el nivel de oxigenación en la sangre no solo depende de la actividad neuronal, sino que depende de otros factores fisiológicos que pueden ocluir la señal de interés. De esta manera, para poder

obtener la señal relacionada con la actividad neuronal se han planteado diferentes estrategias.

2.3.1. Regresión lineal basado en parámetros de posición

El enfoque de corrección de movimiento más común es estimar las series de tiempo de 6 parámetros de realineación, es decir, 3 parámetros de traslación y 3 de rotación para aplicarlos como regresores molestos en el modelo lineal general (GLM). La explicación de la varianza adicional, mediante el uso de los regresores asociados al movimiento, debe equilibrarse con la pérdida de grados de libertad, ya que no es posible saber si se eliminaron todos los efectos producto del movimiento en la señal o si también se están suprimiendo las fluctuaciones relacionadas con la actividad neuronal [14].

2.3.2. Regresión global

La regresión global parte de la conjetura de que una señal registrada en todo el volumen cerebral no puede representar ninguna señal del tipo neuronal, asociándose directamente con toda variabilidad producto de fluctuaciones instrumentales, relacionadas con el movimiento o de procesos de carácter fisiológicos. La señal global también es aplicada como un regresor molesto en el modelo lineal general (GLM) [14].

2.3.3. Depuración de volúmenes

En los registros de rs-fMRI, los movimientos de la cabeza producen los cambios de señal más problemáticos, generando grandes variaciones de intensidad sobre la neuroimagen que no puede ser explicadas adecuadamente por los parámetros de realineación. Para reducir estos efectos residuales en los datos, un enfoque popular es censurar los volúmenes del registro adquiridos durante el período en el que se efectuó el movimiento, es decir, eliminarlo en su totalidad. La censura también se conoce como depuración y es equivalente a agregar regresores de anulación de barrido en el modelo. Las secciones de tiempo afectadas por los fuertes movimientos se pueden identificar en función de los rastros del movimiento, derivados del desplazamiento relativo entre los puntos de tiempo consecutivos de los parámetros de realineación, como la Norma Euclidiana o el desplazamiento de marco (DM) [19].

2.3.4. Corrección basada en el análisis de componentes principales (CompCor)

La implementación del algoritmo de corrección por análisis de componentes principales (PCA), se basa en la suposición de que la señal BOLD en algunas regiones de interés (denominadas ROIs) puede modelar con mayor precisión las fluctuaciones fisiológicas o de movimiento (ruido), por ejemplo, áreas de poca actividad neuronal como ventrículos, vasos sanguíneos grandes o líquido cefalorraquídeo. En los anteriores casos, el método se denomina aCompCor haciendo referencia a la elección de la ROI en una áreas anatómicas, o bien, si se selecciona un voxel de alta desviación estándar temporal el método es conocido como tCompCor, pero en los dos casos a partir de las regiones de interés, se calculan las componentes más significativas mediante el análisis

de componentes principales y estas se introducen en el modelo lineal general (GLM) como variables de la estimación de la señal-ruido [20].

2.3.5. Eliminación de ruido basado en el análisis de componentes principales (ICA)

La estrategia de componentes independientes es un método de análisis multivariante para descomponer una señal en componentes, partiendo de la inferencia de que la señal a analizar está conformada por elementos con independencia estadística y con distribución no gaussiana. Una vez que se calculan las diferentes componentes, es necesario clasificar las que están asociadas con la actividad neuronal y las que proceden de alguna fuente de ruido artefactual. La clasificación se realiza partiendo de las claras diferencias en términos de características espaciales, temporales y espectrales. Después del paso de la clasificación se eliminan las componentes relacionadas con el ruido y se reconstruye el conjunto de datos. Cabe resaltar que, este método también puede ser aplicado posterior a una descomposición por el análisis de componentes principales (PCA) [16].

2.4. Clasificación de componentes independientes mediante técnicas de aprendizaje de máquina

La idea principal de la clasificación de características es eliminar rasgos ruidosos, redundantes o irrelevantes de los datos conservando la mayor parte de la información de interés. Esta medida se ha abordado desde los algoritmos de aprendizaje de máquina, no obstante, las redes neuronales artificiales se han mostrado como métodos prometedores para el aprendizaje automatizado debido a su alta capacidad de aproximar funciones complejas a los datos etiquetados (aprendizaje supervisado) [21], [22].

2.4.1. Perceptrón multicapa (MLP)

El perceptrón multicapa es una clase de red neuronal artificial constituida por unidades básicas o de inferencia conocidas como perceptrones, los cuales asemejan su funcionamiento al modelo biológico más simple de una neurona. El perceptrón suma los valores de entrada de acuerdo con los pesos o parámetros de entramiento asociados a estas y el resultado es introducido a una función de activación no lineal generando el resultado de salida de cada perceptrón. Los pesos en cada una de las unidades base se ajustan en la medida que se efectúa el proceso de aprendizaje y la arquitectura de un perceptrón multi-capas se construye a partir de tres capas principales de diferentes densidades; la capa de entrada, las capas ocultas y la capa de salida, las cuales varían dependiendo de la complejidad del problema [21]. En la *Figura 1* se muestra una esquematización sencilla de una arquitectura MLP (del inglés

Multilayer Percetron) de un clasificador binario (dos perceptrones en la capa de salida) y la estructura interna del perceptrón.

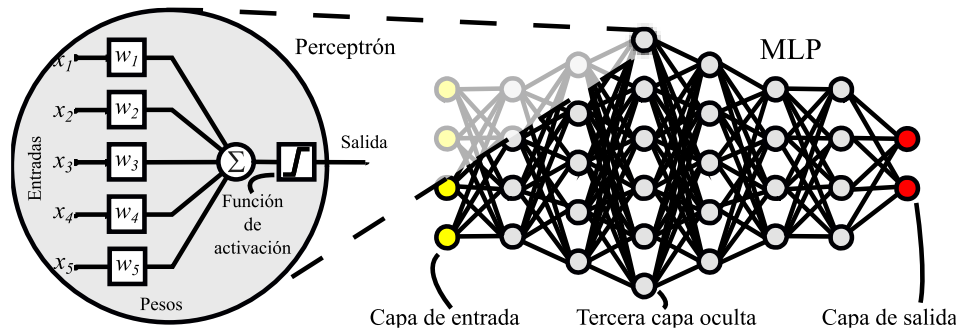


Figura 1 Representación básica de un perceptrón y de una arquitectura MLP.

2.4.2. Red neuronal convolucional

Las redes neuronales convolucionales cuentan con un comportamiento similar a las redes neuronales normales, debido a que estas también cuentan con pesos o parámetros de entrenamiento que se ajustan en el proceso de aprendizaje, sin embargo, la CNN parte de la suposición de que los datos de entrada cuentan con una estructura definida como la de una imagen. Esto implica que los pesos son compartidos en un núcleo para cada fracción de la imagen que se convoluciona y por lo tanto hacen que las neuronas tengan una respuesta local [23], [24]. En la *Figura 2* se muestra la estructura de una red CNN para un clasificador binario.

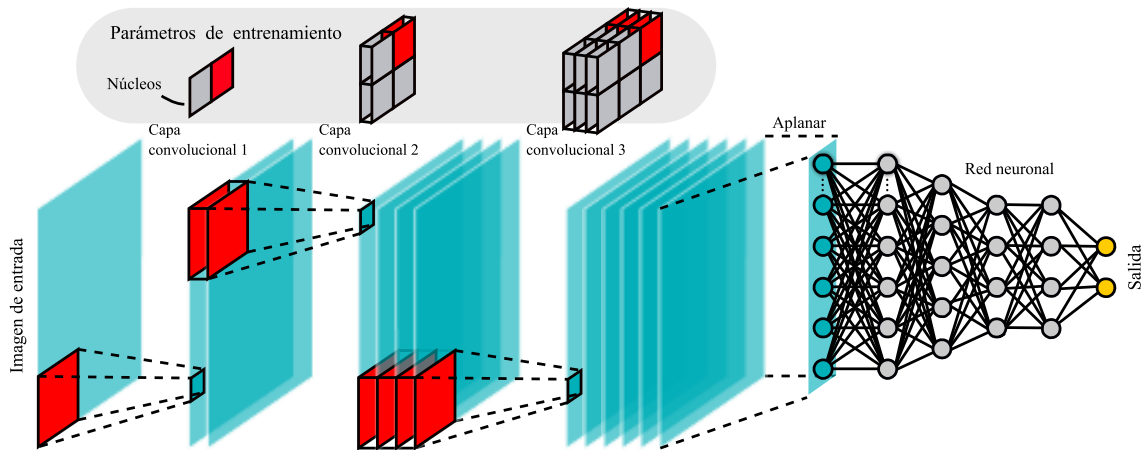


Figura 2 Estructura base de una red neuronal convolucional combinada con una red neuronal convencional para un proceso de clasificación.

2.5. Métricas de conectomas funcionales

2.5.1. Amplitud de fluctuación de baja frecuencia (ALFF) y amplitud fluctuación de baja frecuencia fraccional (fALFF)

La fluctuación de baja frecuencia es un método para detectar cambios de la actividad neuronal regional espontánea, midiendo la intensidad de las oscilaciones en diferentes bandas espectrales.

ALFF mide la intensidad de las oscilaciones de baja frecuencia en diferentes bandas espectrales contenidas entre los 0.01 y 0.1 Hz. Mientras que fALFF es la contribución relativa de las amplitudes en el rango de frecuencias detectables, es decir, la contribución de la banda de interés sobre todo el rango detectable, denotado como $\Omega_0=0.1/2 \cdot \Delta t$, donde Δt representa la tasa de muestreo de la adquisición de la rs-fMRI. Los rangos frecuenciales más comunes son: 0.027 a 0.073 Hz, 0.01 a 0.027 Hz y 0.01 a 0.08 Hz. El cálculo se determina siguiendo las siguientes expresiones matemáticas:

$$ALFF(v_i, \Omega) = \sqrt{\sum_{l: \omega_l \in \Omega} \frac{\alpha(\omega_l)_i^2 + \beta(\omega_l)_i^2}{T/2}}, \quad fALFF(v_i) = \frac{ALFF(v_i, \Omega)}{ALFF(v_i, \Omega_0)} \quad (1)$$

Donde Ω es el rango de frecuencias completo y v_i es el valor del voxel i-ésimo. Además, esta serie temporal puede ser representada como la siguiente serie de Fourier:

$$v_i(t) = \sum_{l=1}^{T/2} \{\alpha_l \cos(\omega_l t) + \beta_l \sin(\omega_l t)\}, \quad \omega_l = \frac{2\pi l}{T} \quad (2)$$

Donde T es el tiempo de muestreo y α_l y β_l son los coeficientes de Fourier.

2.5.2. Homogeneidad regional (ReHo)

Para medir la conectividad local, se emplea la homogeneidad regional (conocida como ReHo), basada en la observación de los K vecinos del vértice de análisis $v_i, N(v_i)$, con lo cual la homogeneidad es calculada a partir de la siguiente ecuación matemática:

$$ReHo(v_i, N(v_i)) = \frac{\sum_{j=1}^T (\gamma_i^j)^2 - T(\underline{\gamma}_i)^2}{\frac{1}{12} K^2 (T^3 - T)} \quad (3)$$

Donde v_i es el voxel i-ésimo, $N(v_i)$ representa los vóxeles vecinos de i-ésimo voxel, γ_i^j es la serie de tiempo de rango $v_i(t_j)$, $\underline{\gamma}_i^j$ es el rango medio entre los vecinos en el j-ésimo punto temporal y $\underline{\gamma}_i$ su rango medio global en todos los vóxeles y T el número de observaciones temporales.

2.5.3. Conectividad funcional (FC)

El método de semilla toma una región de interés con el fin de comparar la actividad neuronal de esa región con el resto del cerebro, permitiendo medir la conectividad funcional entre esas regiones [25]. El cálculo de la conectividad funcional basada en este método se calcula de la siguiente manera:

$$SFC(v_i, s) = \frac{1}{2} \ln \left(\frac{1 + \rho(v_i, s)}{1 - \rho(v_i, s)} \right) \quad (4)$$

Donde, el coeficiente de correlación de Pearson ρ , las series temporales de la semilla S y la serie temporal del volumen cerebral V están dados por:

$$\rho(v_i, s) = SV \quad (5)$$

$$S = (s^j)_{j=1}^T \quad (6)$$

$$V = (v^j) = (v^j)_{j=1}^V \quad (7)$$

Donde v_i es el voxel i -ésimo de análisis, s las series de tiempo asociada a los vóxeles de la semilla, T es el número de observaciones que conforma la semilla, V el total de vóxeles que conforman la región de análisis (volumen) y j el j -ésimo elemento de la semilla o volumen.

2.6. Teoría de grafos

La teoría grafos nacida de la ciencia de redes se ha vuelto un enfoque interdisciplinario y una parte importante del estudio de la conectividad funcional en las neurociencias. En este enfoque las redes se modelan con dos elementos fundamentales, los nodos o vértices y los bordes. El cerebro se puede modelar como una red donde los nodos son las regiones anatómicas de interés y los bordes son las conexiones entre dichas regiones (ver *Figura 3*), por lo tanto, esto permite calcular los diferentes parámetros de la red y caracterizarla [26], [27].

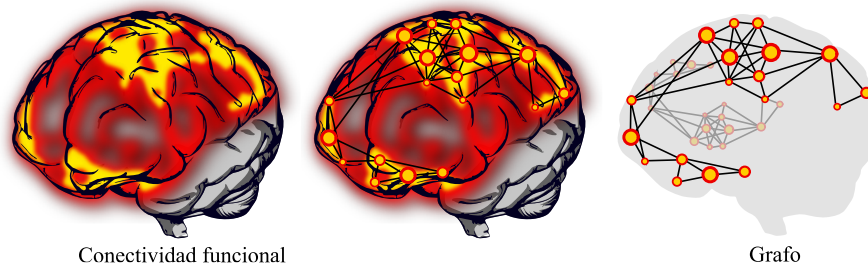


Figura 3 Conectividad funcional modelada como un grafo

2.7. Reproducibilidad

2.7.1. Coeficiente de correlación intraclases (ICC)

La reproducibilidad se calculará a partir del coeficiente de correlación intraclases (ICC), debido a que es uno de los métodos más utilizados en el área médica [13], permitiendo estimar con gran precisión el grado de fiabilidad de cualquier tipo de método o proceso de análisis y por ende la reproducibilidad de la misma. El ICC se basa en la varianza intraindividual e interindividual, como se puede ver en la ecuación más sencilla y representativa [Ver ecuación (1)], en donde σ_b^2 es la variabilidad interindividual (*between*) y σ_w^2 intraindividual (*within*).

$$ICC = \frac{\sigma_w^2}{\sigma_w^2 + \sigma_b^2} \quad (8)$$

3. Objetivos

Objetivo general

Evaluar el impacto de los métodos de eliminación de artefactos sobre la fiabilidad de las métricas de conectómicas funcionales sobre imágenes de resonancia magnética funcional en estado de reposo.

Objetivos específicos

1. Implementar un algoritmo para la clasificación automática de componentes artefactuales obtenidos por ICA en datos fMRI.
2. Implementar un flujo de procesamiento automatizado para imágenes fMRI que integre las herramientas de limpieza de señal BOLD y métricas de conectomas.
3. Medir el impacto sobre la reproducibilidad de las métricas de conectoma bajo los diferentes métodos de limpieza en la señal BOLD usando el Coeficiente de Correlación intraclase.

4. Publicaciones derivadas

- **Mera Jiménez S. M., & Ochoa Gómez J. F.** Machine Learning for the Classification of Independent Components of rs-fMRI

Abstract: At present, functional magnetic resonance imaging at rest (rs-fMRI) is one of the tools of greatest interest in neurosciences due to the advantages presented over other methodologies for acquiring brain activity. However, there is a long way in terms of artifacts cleaning, something necessary to increase the signal-to-noise ratio in the images. Therefore, to make a contribution to the new cleaning methods on rs-fMRI images, in this investigation four deep learning network architectures for the classification of neuronal activity and noise are studied, using 27 temporal spectrum characteristics of the independent components of rs-fMRI images of two groups of healthy subjects. The study allowed to find four architectures with classification accuracy close to 97% and the number of characteristics necessary for the identification of the components associated to noise and neuronal activity were reduced to five, these being the low-frequency fluctuation amplitude (ALFF) in the bands from 0.01 to 0.027 Hz and from 0.027 to 0.073 Hz, the fractional low-frequency amplitude in the same 2 bands and a jump amplitude function.

Estado: Presentada congreso internacional SABI2020 y sometido a la revista SABI.

- **Mera-Jiménez, L., & Ochoa-Gómez, J. F. (2020).** Convolutional Neural Network for the Classification of Independent Components of rs-fMRI. *Tecnológicas*, 24(50), e1626. <https://doi.org/10.22430/22565337.1626>

Abstract: Resting state functional magnetic resonance imaging (rs-fMRI) is one of the most relevant techniques in brain exploration. However, it is susceptible to many external factors that can occlude the signal of interest. In this order of ideas, rs-fMRI images have been studied adopting different approaches, with a particular interest in artifact removal techniques through Independent Component Analysis (ICA). Such an approach is a powerful tool for blind source separation, where elements associated with noise can be eliminated. Nevertheless, such removal is subject to the identification or classification of the components provided by the ICA. In that sense, this study focuses on finding an alternative strategy to classify the independent components. The problem was addressed in two stages. In the first one, the components (3D volumes) were reduced to images by Principal Component Analysis (PCA) and by obtaining the median planes. The methods achieved a reduction of up to two orders of magnitude in the weight of the data size, and they were shown to preserve the spatial characteristics of the independent components. In the second stage, the reductions were used to train six models of convolutional neural networks. The networks analyzed in this study reached accuracies around 98 % in classification, one of them even up to 98.82 %, which reflects the high discrimination capacity of convolutional neural networks.

Estado: publicada en revista tecnológica.

- **Mera Jiménez S. M., & Ochoa Gómez J. F.** Classification of Independent Components Using Convolutional Neural Networks in rs-fMRI and Volume Reduction techniques

Abstract: In the last decade, neurosciences have had an increasing interest in resting state functional magnetic resonance imaging (rs-fMRI) as a result of its advantages, such as high spatial resolution, compared to other brain exploration techniques. To improve the technique, the elimination of artifacts through Independent Components Analysis (ICA) has been proposed, as this can separate neural signal and noise, opening possibilities for automatic classification. The main classification techniques have focused on processes based on machine learning. However, there are currently more robust approaches such as deep learning, which can deal with complex problems directly from the data without feature selection and even with data that does not have a simple interpretation, being limited by the amount of data necessary for training and its high computational cost. This research focused on studying four methods of volume reduction mitigating the computational cost for the training of 10 models based on convolutional neural networks. One of the reduction techniques is a novel approach that we call Reduction by Consecutive Binary Patterns (RCBP), which was shown to preserve the spatial characteristics of the independent components. In addition, the RCBP showed networks in components associated with neuronal activity more clearly. Furthermore, most of the analyzed networks reached accuracies of over 98% in the classification and one network was even found with an accuracy of over 99%, surpassing most of the machine-learning based classification algorithms.

Estado: Sometido revista *Neuroinformatics*

5. Implementación de un algoritmo para la clasificación automática de componentes artefactuales obtenidos por ICA en datos fMRI

La resonancia magnética funcional en estado de reposo (rs-fMRI del inglés *Resting State Functional Magnetic Resonance Imaging*) es una de las técnicas más usadas en la exploración funcional para sujetos sanos, pacientes con alguna fisiopatología e incluso estudios con animales [28]. Esta técnica se basa en medir los cambios espontáneos de señal dependiente del nivel de oxigenación en la sangre, conocida como señal BOLD (Blood Oxygen Level Dependent), relacionada con la actividad neuronal. Además, parte de la relevancia fisiológica del rs-fMRI reside en el hecho de que la energía consumida en estado de reposo representa aproximadamente el 95% de la energía total usada por el cerebro, siendo solo el 5% la aprovechada en la ejecución de tareas explícitas [29]. Asimismo, la fMRI, y por ende la rs-fMRI, es ampliamente utilizada por su carácter no invasivo, por no depender de ningún tipo de radiación ionizante y por su alta resolución espacial frente a otros métodos [11].

Sin embargo, aunque la técnica presenta sus ventajas, existen muchos factores que disminuyen la relación señal/ruido, siendo el ruido instrumental, el aleatorio, el fisiológico y el asociado al movimiento, los que tienen mayor impacto [30]. Por estas razones, es indispensable limpiar la señal capturada por el equipo de resonancia conservando únicamente la señal BOLD que corresponde a la actividad neuronal en estado de reposo, eliminando las variaciones asociadas al ruido [13]. En la actualidad, solo en Estados Unidos, se invierte un estimado de mil cuatrocientos millones de dólares anuales, para evitar la corrupción de datos por movimiento [31].

En ese sentido, se han desarrollado múltiples métodos de análisis estadístico sobre la actividad neuronal espontánea [32]–[34], un sinnúmero de aplicaciones en aprendizaje de máquina [35] y, asimismo, han surgido alternativas prometedoras como el análisis de componentes independientes (ICA), tanto para la identificación de regiones funcionales asociadas, como para la eliminación de artefactos [36], [37]. En este orden de ideas, la relevancia que ha tenido la eliminación de artefactos a través del análisis de componentes independientes ha permitido llegar a desarrollos automatizados como ICA-AROMA [37] o ICA-FIX, basándose en la clasificación mediante características diseñadas y extraídas a partir de las series de tiempo, espectro y distribución espacial de las componentes [38], no obstante, la eliminación de artefactos mediante la clasificación manual de las características espaciales y espectro-temporales aún sigue siendo el estándar de oro [16].

Pese a los buenos resultados entregados por estos dos desarrollos, estos tipos de implementaciones se basan en un gran número de características espaciales, temporales y espectrales de las componentes independientes, lo cual se refleja en un alto costo computacional y un tiempo de procesamiento bastante elevado, como lo es el caso de ICA-FIX, que a pesar de contar con una precisión de clasificación cercana al 99%, su ejecución se basa en 180 características constituidas en 134

espaciales y 46 espectro-temporales [38], siendo la mayoría índices generales utilizados en procesamiento de señales, es decir, índices de tendencia como asimetría, curtosis, entropía, propiedades del espectro como potencia máxima, frecuencia de potencia máxima, funciones derivadas de las series de tiempo, amplitudes de salto de la señal, entre otras. Aun así, entre las numerosas características extraídas, no se considera la amplitud de fluctuación de baja frecuencia (ALFF) y su correspondiente fraccional (fALFF), medidas proporcionales a la intensidad de actividad cerebral espontánea regional [39], o el almacenamiento de información activa (AIS) para cuantificar el procesamiento de información en sistemas neuronales [40], índices asociados al carácter dinámico de la actividad neuronal.

Por otro lado, el aprendizaje profundo es uno de los temas de mayor crecimiento en informática médica [22], ya que ha demostrado no depender de las características diseñadas o extraídas manualmente, superando las técnicas de aprendizaje de máquina convencionales con aplicaciones en reconocimiento de patrones, detección, visión por computadora [41] y clasificación, donde en esta última, incluso se ha logrado superado el rendimiento humano [42]. De hecho, el interés en esta área es tal, que hoy en día es fácil encontrar estudios en temas encaminados al análisis de las imágenes de resonancia magnética [43], [44] e incluso en la clasificación de trastornos bajo el paradigma en estado de reposo [45]. Son claras las ventajas manifestadas en el aprendizaje profundo [46], limitado únicamente por el número de datos necesarios para entrenar los algoritmos, los cuales pueden estar en el orden miles de datos, presentándose como una carga computacional muy elevada, más aún en el caso de las rs-fMRI, donde los datos son arreglos de cuatro dimensiones (3 espaciales y una temporal).

Teniendo en cuenta las anteriores consideraciones, este objetivo cuenta con varios enfoques. El primero de ellos es buscar una solución alternativa y de alta precisión, para la detección automática de artefactos, basado en un número menor de características espectro-temporales, incluyendo índices no considerados hasta el momento como el almacenamiento de información activa (AIS) [40], la fluctuación de baja frecuencia y su correspondiente fraccional [39], donde dicho enfoque se aborda con el diseño de diferentes modelos de redes neuronales artificiales basadas en la arquitectura del perceptrón multicapa (MLP).

El segundo de ellos se encamina en buscar la mejor forma de representar los volúmenes cerebrales de las componentes independientes, abordando algunas técnicas ya existentes de reducción de dimensiones como el análisis de componentes principales (PCA) [47] e incluyendo un nuevo método basado en el concepto de patrones binarios locales [48], pero aplicados a vóxeles de cortes consecutivos, el cual llamaremos a lo largo de este documento como reducción por patrones binarios consecutivos (RPBC). El tercer enfoque, vuelve a retomar la idea del primero, pero basado la clasificación de las componentes mediante una red neuronal convolucional entrenadas sobre las previas representaciones de las reducciones de los volúmenes de las componentes independientes.

5.1. Metodología

5.1.1. Sujetos y componentes independientes

Las redes se entrenaron con 2 grupos de datos, el primero de ellos se generó a partir de las imágenes funcionales y anatómicas de 100 participantes aleatorios del proyecto de acceso abierto *Longitudinal Imaging Multimodal (SLIM) Brain Data Repository* realizado por la universidad *Southwest University* como un estudio longitudinal de adultos en estado de reposo [49]. Los datos están conformados por hombres y mujeres sanos en edades desde los 18 hasta los 27 años.

El segundo grupo de datos se tomó del mismo conjunto de sujetos utilizados en el entrenamiento de ICA-FIX obtenido del proyecto de acceso público *Human Connectome Project* [50], sin embargo, las componentes independientes de estos sujetos ya están clasificadas [51]. Por lo tanto, la metodología descrita a continuación, desde el inciso 5.12 hasta el 5.14, solo se aplica al primer grupo de sujetos.

5.1.2. Adquisición de los datos fMRI

Los estudios funcionales para el primer grupo se obtuvieron como imágenes eco-planares (EPI) mediante un scanner SIEMENS (MAGNETOM Trio) con una intensidad de campo magnético de 3T, una bobina de recepción de 15 canales y con la versión del software syngo MR B17. La secuencia de adquisición contó con los siguientes parámetros: Tiempo de repetición (TR) de 2.0 s, tiempo de eco (TE) de 30 ms, tamaño de voxel de 3.4x3.4x3.0 mm y tiempo de registro aproximado de 8.08 minutos [49]. Además, para la disposición de vigilia, se solicitó a los participantes que mantuvieran los ojos abiertos en todo el proceso de registro.

5.1.3. Clasificación de las componentes

Las componentes independientes (IC) se clasificaron manualmente en dos categorías, como señal de actividad neuronal (S-IC) o ruido (N-IC). La clasificación se fundamentó en tres características complementarias: el mapa espacial de IC, la serie temporal y su densidad espectral de potencia (magnitud de la transformada de Fourier de la serie de tiempo), además, la visualización de estas se realizó mediante el software *FSLeyes* [52], función contenida en FLS-5.0.

Los criterios de clasificación utilizados son [16]:

- a) La distribución espacial de las componentes asociadas al ruido presenta gran cantidad de pequeños grupos como en la *Figura 4*, en contraste, la actividad neuronal se manifiesta como pocos grupos de gran tamaño similar a la *Figura 5*.

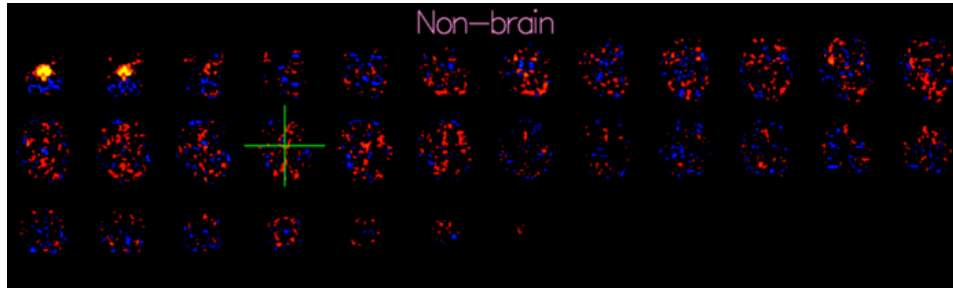


Figura 4 Mapa espacial de una componente independiente vistas en cortes axiales de un mismo sujeto en diferentes cortes a lo largo del eje z. Los colore amarillos representa una mayor correlación entre regiones y la ubicación espacial de la componente. La componente tiene carácter de ruido, ya que cuenta con gran número de pequeños grupos, además, hay una fuerte presencia en el líquido cefalorraquídeo para los primeros cortes. La imagen se deriva del software FSL-Eyes y los cortes aparecen como mapas de calor para los valores positivos (colores cálidos) y mapas fríos para los valores negativos (colores fríos). La cruz verde hace parte del software y únicamente indica la posición del corte que se está seleccionando.

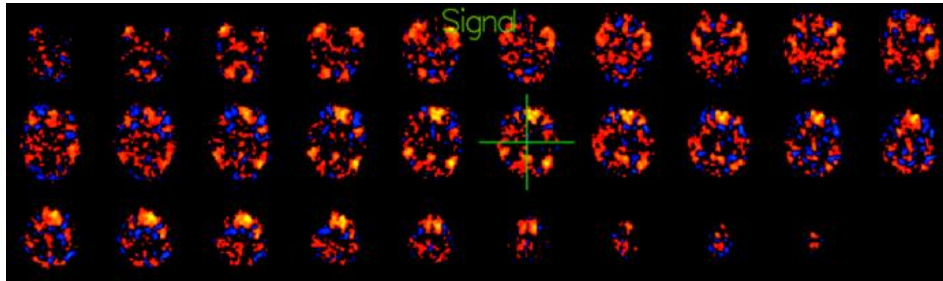


Figura 5 Mapa espacial de una componente independiente en cortes axiales de un mismo sujeto en diferentes cortes a lo largo del eje z. Los colore amarillos representa una mayor correlación entre regiones y la ubicación espacial de la componente. La componente tiene carácter de actividad neuronal, tiene pocos grupos de gran tamaño ubicados mayormente sobre la materia gris y no cuenta con patrones extraños no biológicos. La imagen se deriva del software FSL-Eyes y los cortes aparecen como mapas de calor para los valores positivos (colores cálidos) y mapas fríos para los valores negativos (colores fríos). La cruz verde hace parte del software y únicamente indica la posición del corte que se está seleccionando.

- b) El ruido tiene fuerte incidencia sobre el líquido cefalorraquídeo, se superpone con la materia blanca y con los vasos sanguíneos, además, puede manifestarse en áreas cercanas al campo de visión, áreas dentro de la región de pérdida de la señal, por ejemplo, interfaz tejido-aire o incluso en los límites del cerebro, pero en este caso, se crean anillos, lunas y rayas que recorren el borde como se muestra en la Figura 6.

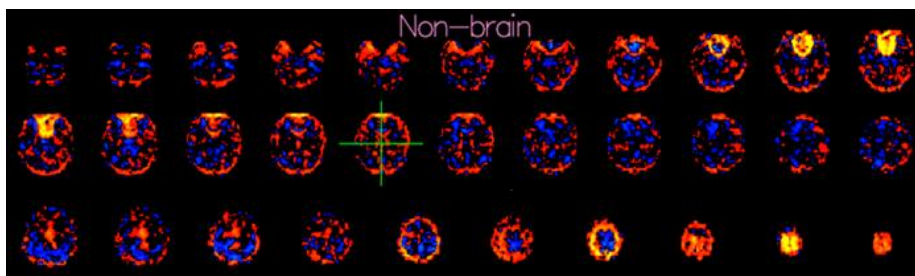


Figura 6 Mapa espacial de una componente independiente en cortes axiales de un mismo sujeto en diferentes cortes a lo largo del eje z. Los colore amarillos representa una mayor correlación entre regiones y la ubicación

espacial de la componente. Nuevamente la componente tiene características de ruido. Tiene fuerte presencia en el líquido cefalorraquídeo, en límites del cerebro generando anillos y lunas. La imagen se deriva del software FSL-Eyes y los cortes aparecen como mapas de calor para los valores positivos (colores cálidos) y mapas fríos para los valores negativos (colores fríos). La cruz verde hace parte del software y únicamente indica la posición del corte que se está seleccionando.

- c) La actividad neuronal se presenta en la materia gris y cuenta con poca superposición o es totalmente ausente sobre la materia blanca, el líquido cefalorraquídeo y vasos sanguíneos.
- d) Cualquier patrón no biológico, por ejemplo, patrones de bandas en dirección de corte, rayas a lo largo de la dirección de codificación de fase o las secuencias aceleradas son características de ruido por la adquisición.
- e) La serie de tiempo de la actividad neuronal es regular y oscilatoria sin presencia de grandes saltos de la señal o cambios repentinos, similar a la primera gráfica de la *Figura 7*. En contraste, el ruido puede manifestarse con cambios rápidos de la señal y grandes saltos como se muestra en la segunda gráfica de la *Figura 7*.

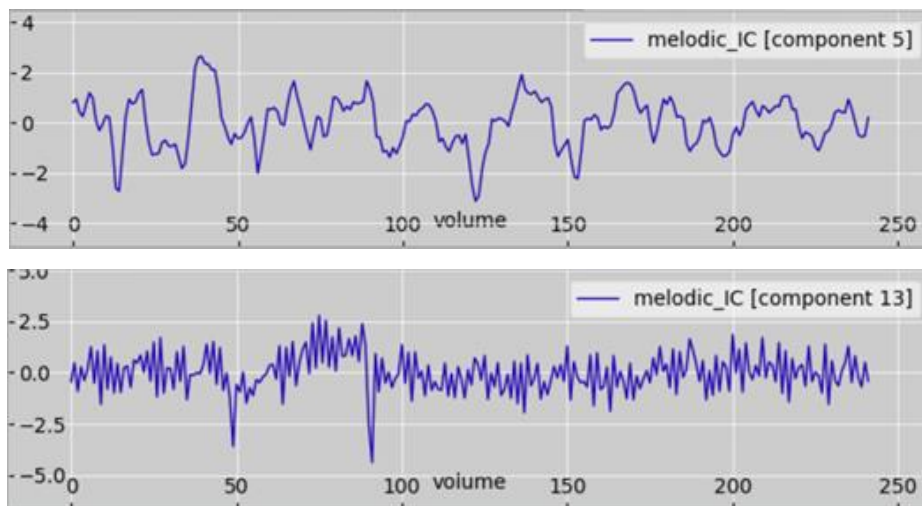


Figura 7 Series de tiempo de diferente naturaleza. Las gráficas corresponden a las series de tiempo de los volúmenes de dos componentes independientes (componente 5 y 13). El eje vertical indica la intensidad de la señal BOLD (nivel de oxigenación en la sangre) y el horizontal está en segundos. La primera gráfica es una señal de actividad neuronal, mientras que la segunda muestra características de ruido.

- f) La distribución espectral de potencia en la actividad neuronal se ubica regularmente entre los rangos de 0.01 a 0.1Hz.

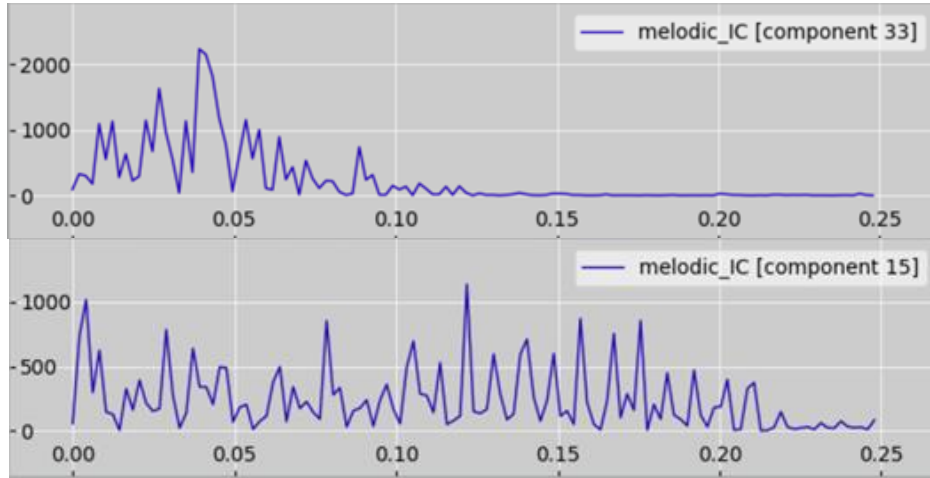


Figura 8 Densidad espectral de diferente naturaleza. Las gráficas corresponden a las series de tiempo de los volúmenes de dos componentes independientes (componente 33 y 15). El eje vertical indica la densidad espectral y el horizontal es la frecuencia en Hz. La primera es el espectro correspondiente a una señal neuronal, mientras que la segunda cuenta con un espectro relacionado con el ruido.

5.1.4. Extracción de características de las series de tiempo y densidad espectral de potencia

Se calcularon 27 características a partir de cada una de las series de tiempo de las componentes independientes para entrenar los modelos de aprendizaje profundo, las cuales se listan a continuación:

1. Amplitud de fluctuación de baja frecuencia (ALFF) para las siguientes cuatro bandas espectrales: de 0.01 a 0.027 Hz, de 0.027 a 0.073 Hz, de 0.073 a 0.198 y de 0.198 a 0.25 Hz [53], las cuales denotamos respectivamente como: ALFF5, ALFF4, ALFF3 y ALFF2.
2. Amplitud de fluctuación de baja frecuencia fraccional (fALFF) en las mismas bandas espectrales que ALFF [53]. De manera análoga al caso anterior, estas serán denotadas a lo largo de este documento como: fALFF5, fALFF4, fALFF3 y fALFF2, para las mismas 4 bandas espectrales.
3. Almacenamiento de información activa (AIS) [40].
4. Curtosis, media, mediana, entropía, negentropía [54].
5. Tres tipos de funciones de amplitud de salto basadas en derivadas temporales [38] como se muestran en las siguientes ecuaciones:

$$salto_1 = \left(\left| \frac{da}{dt} \right| \right) / std(a) \quad (9)$$

$$salto_2 = \left(\left| \frac{da}{dt} \right| \right) / std \left(\left| \frac{da}{dt} \right| \right) \quad (10)$$

$$salto_3 = \left(\left| \frac{da}{dt} \right| \right) / std(a) \quad (11)$$

Donde std es la desviación estándar y a es la serie de tiempo de la componente principal.

6. Frecuencia media (MNF), frecuencia mediana (MDF), potencia total (TTP), potencia promedio (MNP), frecuencia del pico máximo de potencia (PKF), momentos espectrales de orden 1, 2 y 3 (SM1, SM2, SM3), relación entre potencia máxima y potencia total (PSP), varianza de la frecuencia central (VCF) [55].

5.1.5. Diseño de las redes neuronales basadas en MLP

Inicialmente, los modelos de las redes se basaron en la arquitectura del perceptrón multicapa (MLP) debido a su capacidad universal de aproximar problemas de relaciones no lineales entre los datos de entrada-salida y su fácil uso, además de una de las principales arquitecturas en la resolución de problemas [21]. Las cuatro arquitecturas se diseñaron con las librerías de keras en el lenguaje de programación Python 3.0 [56]. Todas las redes contaron con una función de pérdida de crossentropía categórica, un optimizador de gradiente descendiente estocástico, una métrica de precisión del modelo calculado mediante validación cruzada y una regularización mediante abandono al azar de nodos (*dropout*). La regularización por Dropout, o la técnica por deserción de neuronas consiste en ignorar aleatoriamente un porcentaje del total de neuronas durante el entrenamiento por cada época. El proceso permite que los modelos no se sobre-ajusten.

En la Tabla I se muestran con detalle la conformación de los 4 modelos implementados y se especifican el número de parámetros de entrenamiento como una función de número de características de entrada.

Tabla I Arquitecturas de las redes neuronales artificiales

Arquitectura	Número de perceptrones	Función de activación	Regularización
Modelo 1			
Capa de entrada	Co	NA	NA
Primera capa oculta	13	tanh	Dropout = 0.3
Segunda capa oculta	6	tanh	Dropout = 0.3
Capa de salida	2	softmax	NA
Parámetros de entrenamiento: $98+13*(Co+1)$			
Modelo 2			
Capa de entrada	Co	NA	NA
Primera capa oculta	8	tanh	Dropout = 0.3
Segunda capa oculta	4	tanh	Dropout = 0.3
Capa de salida	2	softmax	NA
Parámetros de entrenamiento: $46+8*(Co+1)$			
Modelo 3			
Capa de entrada	Co	NA	NA
Primera capa oculta	16	tanh	Dropout = 0.3
Segunda capa oculta	8	tanh	Dropout = 0.3
Tercera capa oculta	4	tanh	Dropout = 0.3
Capa de salida	2	softmax	NA
Parámetros de entrenamiento: $182+16*(Co+1)$			

Modelo 4			
Capa de entrada	Co	NA	NA
Primera capa oculta	8	tanh	Dropout = 0.3
Capa de salida	2	softmax	NA
Parámetros de entrenamiento: $18+8*(Co+1)$			

NA: No aplica. Co: Número de características de entrada en la red.

Nota: Le número de parámetros de entrenamiento se deja en función Co ya que señales usadas contaban con diferentes dimensiones.

Donde los parámetros de entrenamiento por capa se calculan como $(int + 1) * p$, donde int son las entradas en cada capa y p el número de perceptrones por capa.

5.1.6. Reducción de dimensiones

Se realizó un análisis de correlaciones entre las 27 características, validando la precisión de clasificación de estas en los 4 modelos y se redujo el número de características en un número menor de variables latentes mediante el análisis de componentes principales (PCA), buscando un número menor de características a ser entrenadas con la red. Adicionalmente, también se optó por saturar las entradas de las cuatro redes con cada una de las 27 características, como un método alternativo para calcular el aporte individual de cada variable sobre la precisión de clasificación de la red.

5.1.7. Entrenamiento de las redes MLP

El entrenamiento de las redes se realizó con 80 por ciento de los datos obtenidos en los dos grupos (datos de entrenamiento), un tamaño de lote de 500 muestras, 40 por ciento de regularización en la capa de *Dropout* para evitar el sobreajuste de los modelos y se ejecutó para 1000 épocas con el fin de reducir al máximo la pérdida. El algoritmo se ejecutó varias veces almacenando de manera automática los pesos de los 4 modelos con mayor precisión de clasificación, previendo la variabilidad generada por el carácter estocástico del optimizador. La precisión de los 4 modelos se observó mediante validación cruzada usando una división de los datos 80-20% de los datos de entrenamiento. Sin embargo, la precisión definitiva de los modelos se calculó mediante la concordancia de la clasificación y las etiquetas originales del 20 por ciento de los datos restantes (datos de prueba).

La ejecución se realizó sobre la plataforma Colab de Google y la implementación de los algoritmos está contenida en el siguiente repositorio de GitHub (<https://github.com/LxMera/Deep-Learning--denoising-rs-fMRI>)

5.1.8. Reducción de volúmenes de los mapas espaciales

Con el fin de dar paso al entrenamiento de las redes neuronales convolucionales, todos los volúmenes de las componentes independientes se redujeron, mediante cuatro diferentes métodos, a matrices bidimensionales de tres canales (o imágenes). La reducción se realizó sobre la plataforma Colab de Google utilizando el lenguaje de

programación Python, implementando principalmente las librerías de Nibabel, Numpy y Opencv. Los métodos de reducción utilizados se presentan a continuación.

5.1.8.1. Reducción por PCA

El análisis de componentes principales se utilizó para obtener tres imágenes equivalentes de las secciones axial, coronal y sagital. La imagen axial se generó dividiendo el volumen de los componentes independientes en 24 cortes axiales como se muestra en la *Figura 9 (a)*. Cada corte se consideró como la matriz de observaciones de las $p \times q$ variable, donde p y q son las dimensiones del volumen en los ejes coronal y sagital, respectivamente. Posteriormente, cada matriz se almacenó en un vector de dimensiones equivalentes al número de cortes axiales, para aplicar el análisis de componentes principales, obteniendo el nuevo conjunto de componentes no correlacionados [47]. Los primeros tres componentes principales se almacenaron en los canales RGB para generar la imagen estándar de 8 bits como se ilustra en la *Figura 9 (b)*. Las imágenes de los cortes coronal y sagital se generaron de la misma manera, sin embargo, las matrices de observación se tomaron como cortes consecutivos a lo largo de sus respectivos ejes como se ilustra en la *Figura 9 (c)* y *(e)*, creando imágenes similares a las ilustradas en *Figura 9 (d)* y *(f)*.

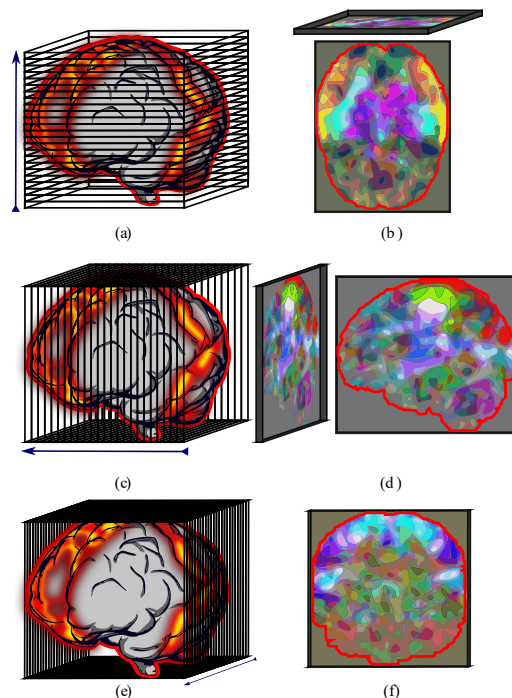


Figura 9 Reducción del volumen espacial mediante análisis de componentes principales. En el volumen cerebral los colores cálidos representan la probabilidad de encontrar la componente independiente (ubicación espacial). (a) volumen cerebral en cortes axiales. (b) Imagen axial obtenida de las tres componentes principales de mayor porcentaje de varianza explicada. (c) volumen cerebral en cortes coronales. (d) Imagen coronal obtenida de las tres componentes principales de mayor porcentaje de varianza explicada. (e) volumen cerebral en cortes sagitales. (f) Imagen sagital obtenida de las tres componentes principales de mayor porcentaje de varianza explicada. Las imágenes de la derecha son la combinación de las tres primeras componentes principales en una imagen RGB, lo que genera esa gran cantidad de colores.

5.1.8.2. Extracción de bordes

Como se indicó anteriormente, en el análisis de los componentes principales se seleccionaron los componentes de mayor varianza de las matrices de observación, en este caso, los cortes a largos del eje en consideración, por lo tanto, de ahí que las componentes creadas conservan la mayor varianza a lo largo de dicho eje, implicando una menor variación en las regiones externas al volumen cerebral debido a la homogeneidad del medio, por lo tanto, esta consideración se utilizó para encontrar el contorno del volumen cerebral en las secciones axial, coronal y sagital.

Las tres imágenes derivadas de PCA se segmentaron para valores de píxeles iguales a los puntos finales de los mismos (píxeles no asociados con el volumen cerebral), obteniendo imágenes binarias, las cuales fueron sometidas al algoritmo 'Suzuki85' para la detección de bordes externos [57]. Los bordes externos originados por esta segmentación delimitan el volumen cerebral como se ilustra en los contornos rojos de las *Figura 9* (b), (d) y (f). Cabe señalar que los contornos se utilizaron para obtener un nuevo conjunto de imágenes, es decir, representaciones de los cortes axial, coronal y sagital con contornos.

5.1.8.3. Reducción en cortes medios

Cada volumen de los componentes independientes se redujo a tres imágenes correspondientes a los cortes intermedios en los planos axial, coronal y sagital como se muestra en la *Figura 10*. Cada imagen se ajustó para los niveles de gris modificando el histograma, ajustando el nivel de intensidad de 2 como el mínimo y el valor de 22 como máximo en la imagen de 8 bits, es decir, los niveles entre 2 y 22 se ajustaron proporcionalmente a una escala de 0 a 255. Cabe resaltar que los cortes perpendiculares no tienen el contorno claramente definido. Por lo tanto, a este grupo de imágenes también se les agregó el contorno obtenido con PCA en el canal rojo, generando dos grupos de imágenes, unas con contorno y otras sin contorno como se esquematiza en la *Figura 10* de (c) a (h).

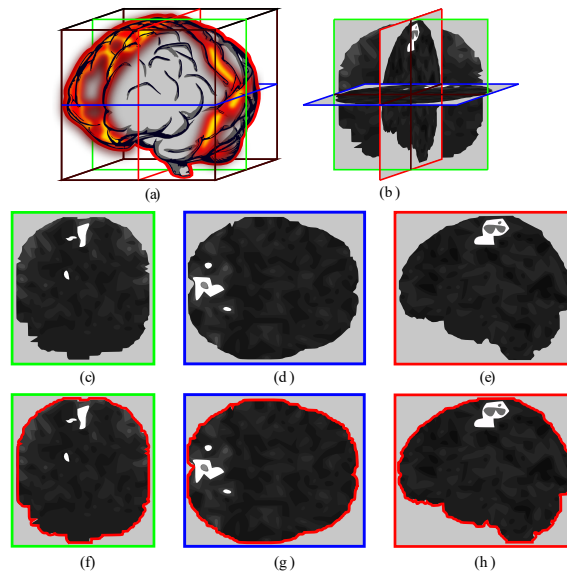


Figura 10 Reducción del volumen espaciales de una componente principal en 3 cortes. En el volumen cerebral los colores cálidos representan la probabilidad de encontrar la componente independiente (ubicación espacial). (a) Esquematización del volumen de una componente independiente, (b) cortes medios del volumen obteniéndose tres imágenes de la actividad neuronal correspondientes a los cortes coronal (c), axial (d) y sagital (e). Cortes de la actividad neuronal coronal (f), axial (g) y sagital (h) delimitadas por el límite del volumen cerebral (límite en color rojo). Los resultados axiales, coronales y sagitales se resaltan en los marcos verde, azul y rojo. Además, los colores blancos representan la posición espacial de la componente independiente.

5.1.8.4. Reducción por patrones binarios consecutivos

Los volúmenes de los componentes independientes se dividieron en 24 cortes axiales asignados a los canales rojo, verde y azul (imagen RGB) distribuidos en 3 grupos de 8 cortes como se muestra en la *Figura 11* (a). A cada grupo se le asignaron pesos ascendentes con valores potenciales de 2, es decir, los valores de 1, 2, 4, 8, 16, 32, 64 y 128. Posteriormente, cada corte de la actividad neuronal (*Figura 11* b y c) se sometió a un proceso de umbralización con un valor de 2 y una transformación morfológica de cierre con un núcleo de tamaño 4×4 , obteniendo las imágenes binarias ilustradas en la *Figura 11* (d). Luego, las imágenes por grupo se multiplicaron con sus respectivos pesos (*Figura 11* (e)) y se agregaron a una imagen de 8 bits de un solo canal como se muestra en la *Figura 11* (f). Finalmente, las tres imágenes generadas (*Figura 11* (g)) se combinaron en una sola imagen RGB estándar como se indica en la *Figura 11* (h).

Es de aclarar que, de forma análoga a los métodos anteriores, en este caso, se generaron 2 conjuntos de imágenes, unas sin contorno y otras con el contorno derivado de la reducción del volumen por PCA. Por otro lado, las componentes contienen más de 24 cortes y la cantidad varía según la resolución del equipo de MRI, por lo tanto, se tomaron cortes equidistantes desde la base del volumen cerebral hasta la parte superior. Además, se tomaron 24 cortes ya que una imagen convencional se conforma de tres canales (canales Rojo, Verde y Azul) y es de 8 bits.

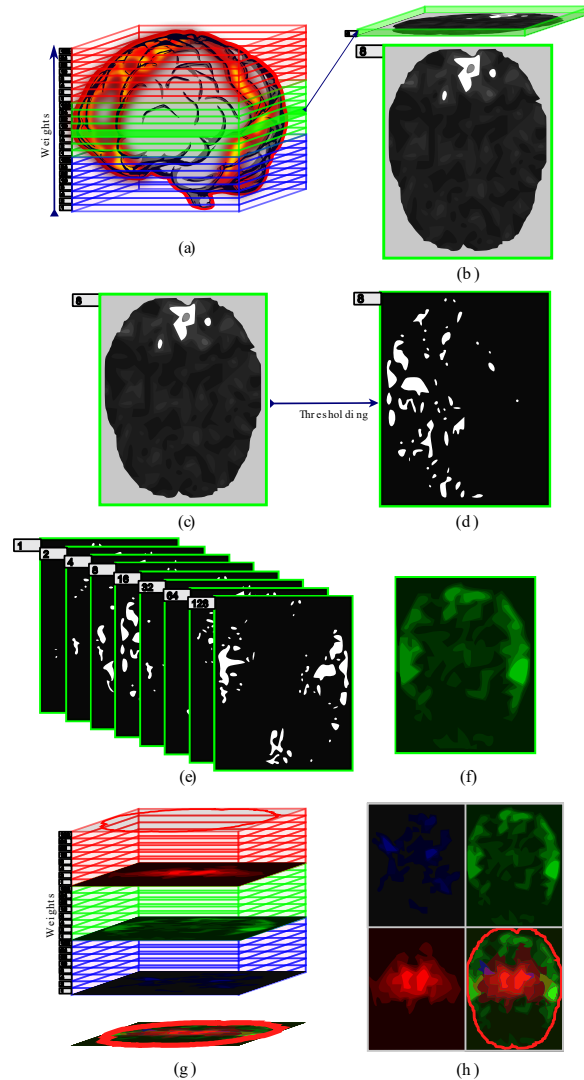


Figura 11 Proceso de reducción del volumen en una imagen RGB axial. (a) Esquematización del volumen de una componente independiente con 24 cortes axiales (8 para cada canal de la imagen RGB). Cada corte tiene asignado un peso con valores potencia de 2. (b, c) Actividad neuronal del doceavo corte axial. (d) Imagen binaria producto de la umbralización de la posición de la actividad neuronal. (e) Umbralización de los 8 cortes de la zona media. (f) suma de las ocho imágenes binarias multiplicadas por sus respectivos pesos. (g) Esquematización de la reducción del volumen en los tres canales de la imagen RGB. (h) Visualización de los tres canales y la imagen resultante delimitada por el límite del volumen cerebral. Cada 8 cortes se reducen en una canal que se almacena en los canales RGB de una imagen convencional.

El proceso anterior se realizó de la misma manera para la reducción del volumen a una imagen RGB coronal, sin embargo, en este caso, el volumen se dividió en 24 cortes coronales como se ilustra en la *Figura 12* (a). Cada corte fue umbralizado bajo las mismas consideraciones, multiplicado por el respectivo peso, sumado y combinado para generar la imagen RGB como se ilustra en la *Figura 12* y, asimismo, se generaron 2 grupos de imágenes con y sin contorno.

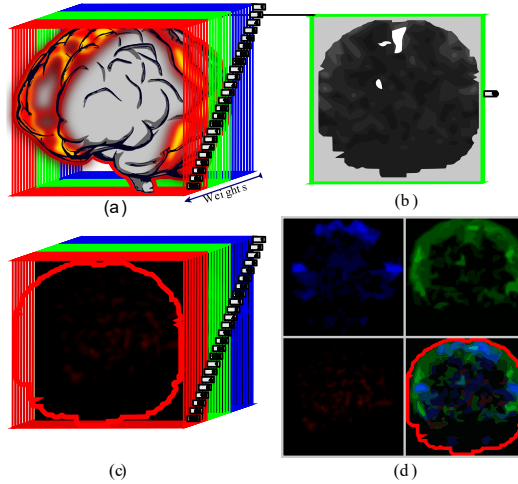


Figura 12 Proceso de reducción del volumen en una imagen RGB coronal. (a) Esquematización del volumen de una componente independiente con 24 cortes coronales (8 para cada canal de la imagen RGB). Cada corte tiene asignado un peso con valores potencia de 2. (b) Actividad neuronal del treceavo corte coronal. (c) Esquematización de la reducción del volumen en los tres canales de la imagen RGB. (d) Visualización de los tres canales y la imagen resultante delimitada por el límite del volumen cerebral.

Como en el caso anterior, la reducción a una imagen sagital RGB siguió el mismo proceso. Sin embargo, en este caso, el volumen se dividió en 24 cortes sagitales como se ilustra en la *Figura 13* (a). Cada corte fue umbralizado bajo las mismas consideraciones, multiplicado por el respectivo peso, sumado y combinado para generar la imagen RGB como se ilustra en la *Figura 13*, además, también se generaron 2 grupos de imágenes con y sin contorno.

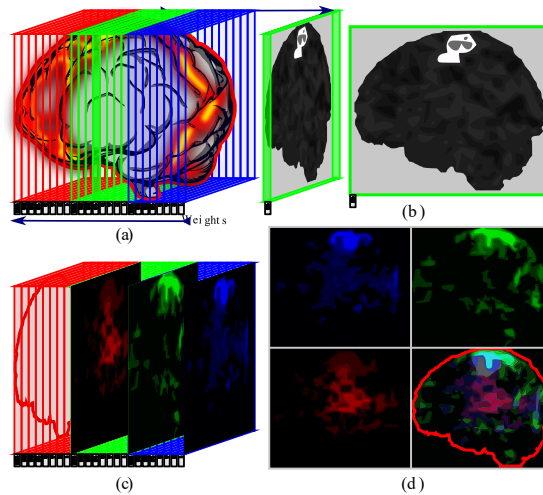


Figura 13 Proceso de reducción del volumen en una imagen RGB sagital. (a) Esquematización del volumen de una componente independiente con 24 cortes sagitales (8 para cada canal de la imagen RGB). Cada corte tiene asignado un peso con valores potencia de 2. (b) Actividad neuronal del treceavo corte sagital. (c) Esquematización de la reducción del volumen en los tres canales de la imagen RGB. (d) Visualización de los tres canales y la imagen resultante delimitada por el límite del volumen cerebral.

5.1.8.5. Reducción de patrones binarios consecutivos y filtrado no lineal

Este proceso tiene los mismos pasos utilizados en el método anterior, pero se aplicó un filtro no lineal constituido por un filtro de desenfoque mediano de tamaño 7×7 a las imágenes resultantes con el fin de eliminar los pequeños elementos aislados considerados como ruido en el fMRI [16].

5.1.9. Escalado de imágenes

Las matrices 3D de los componentes independientes provienen de diferentes conjuntos de datos y, en consecuencia, tienen diferentes tamaños de voxel como se indica en la Tabla I, por lo tanto, todas las imágenes se redujeron mediante la interpolación de la relación de área de píxeles en los tamaños de 73×57 , 57×57 y 57×73 para corte sagital, coronal y axial respectivamente.

5.1.10. Diseño de las redes CNN

Al igual que en las secciones previas, en este apartado se aborda la clasificación de artefactos para la posterior eliminación de estos. No obstante, en este caso se usaron las características espaciales de las componentes independientes. Además, resaltamos que las primeras redes fueron diseñadas a partir de las series de tiempos y en este caso únicamente se utilizan redes neuronales convolucionales con las características espaciales de las componentes independientes.

El diseño de las redes neuronales convolucionales (CNN del inglés Convolutional Neural Network) se generaron en la plataforma Colab de Google, implementando las bibliotecas Keras y TensorFlow en el lenguaje de programación Python 3.0, donde se crearon diez modelos de redes neuronales convolucionales [23], seis de ellos con una sola imagen de entrada y cuatro de ellos con estructura multimodal para la recepción de 3 imágenes. Todos los modelos fueron diseñados con la función de pérdida de *sparse categorical crossentropy*, el optimizador *Adam* basado en un gradiente estocástico de primer orden, se regularizaron mediante la técnica de deserción de neuronas y se validaron utilizando la métrica de precisión del modelo, calculada a partir del 10% de los datos de entrenamiento.

Las Tabla II y Tabla III muestran en detalle las arquitecturas de las 10 redes. Todas cuentan con estructuras convolucionales en la entrada para extraer las características de las imágenes, luego, se aplanan y se entregan a una estructura de perceptrón multicapa para terminar con la clasificación en dos tipos de componentes, neuronal o ruido. Las estructuras en la Tabla II están diseñadas para recibir solo una imagen de entrada, es decir, la imagen axial, coronal o sagital. Por otro lado, las redes multimodales se diseñaron para recibir los 3 cortes y extraer características individualmente usando las diferentes capas convolucionales como se muestra en la Tabla III, las cuales están marcadas con los superíndices 1, 2 y 3 indicando el proceso sobre las imágenes sagital, coronal y axial respectivamente.

Tabla II Arquitecturas de las redes neuronales convolucionales de una sola imagen de entrada

Modelo 1					
Arquitectura	Función	Filtros o perceptrones	Tamaño de filtro	Función de activación	Deserción (Dropout)
Función Lambda	Lambda	-	-	-	-
Convolutacional	Conv2D	16	(3, 3)	elu	0.4
Convolutacional	Conv2D	16	(3, 3)	elu	-
Submuestreo	MaxPooling2D	-	(2, 2)	-	-
Aplanar	Flatten	-	-	-	-
Capa densa	Dense	16	-	relu	0.4
Capa densa	Dense	8	-	relu	0.4
Capa densa	Dense	4	-	relu	0.4
Capa de salida	Dense	2	-	softmax	-
Parámetros de entrenamiento* 261.014					
Modelo 2					
Arquitectura	Función	Filtros o perceptrones	Tamaño de filtro	Función de activación	Deserción (Dropout)
Función Lambda	Lambda	-	-	-	-
Convolutacional	Conv2D	32	(3, 3)	elu	-
Submuestreo	MaxPooling2D	-	(2, 2)	-	0.4
Aplanar	Flatten	-	-	-	-
Capa densa	Dense	128	-	relu	0.4
Capa de salida	Dense	2	-	softmax	-
Parámetros de entrenamiento* 4.130.050					
Modelo 3					
Arquitectura	Función	Filtros o perceptrones	Tamaño de filtro	Función de activación	Deserción (Dropout)
Función Lambda	Lambda	-	-	-	-
Convolutacional	Conv2D	32	(3, 3)	elu	-
Submuestreo	MaxPooling2D	-	(2, 2)	-	0.4
Convolutacional	Conv2D	64	(3, 3)	elu	-
Submuestreo	MaxPooling2D	-	(2, 2)	-	0.4
Aplanar	Flatten	-	-	-	-
Capa densa	Dense	128	-	relu	0.4
Capa de salida	Dense	2	-	softmax	-
Parámetros de entrenamiento* 2.084.162					
Modelo 4					
Arquitectura	Función	Filtros o perceptrones	Tamaño de filtro	Función de activación	Deserción (Dropout)
Función Lambda	Lambda	-	-	-	-
Convolutacional	Conv2D	32	(3, 3)	elu	-
Submuestreo	MaxPooling2D	-	(2, 2)	-	0.4
Convolutacional	Conv2D	64	(3, 3)	elu	-
Submuestreo	MaxPooling2D	-	(2, 2)	-	0.4
Convolutacional	Conv2D	128	(3, 3)	elu	-
Submuestreo	MaxPooling2D	-	(2, 2)	-	0.4
Aplanar	Flatten	-	-	-	-
Capa densa	Dense	128	-	relu	0.4
Capa de salida	Dense	2	-	softmax	-
Parámetros de entrenamiento* 1.125.826					
Modelo 5					
Arquitectura	Función	Filtros o perceptrones	Tamaño de filtro	Función de activación	Deserción (Dropout)

Función Lambda	Lambda	-	-	-	-
Convolutacional	Conv2D	32	(3, 3)	elu	-
Submuestreo	MaxPooling2D	-	(2, 2)	-	0.4
Convolutacional	Conv2D	32	(3, 3)	elu	-
Submuestreo	MaxPooling2D	-	(2, 2)	-	0.4
Convolutacional	Conv2D	64	(3, 3)	elu	-
Submuestreo	MaxPooling2D	-	(2, 2)	-	0.4
Convolutacional	Conv2D	128	(3, 3)	elu	-
Submuestreo	MaxPooling2D	-	(2, 2)	-	0.4
Aplanar	Flatten	-	-	-	-
Capa densa	Dense	512	-	relu	0.4
Capa densa	Dense	128	-	relu	0.4
Capa de salida	Dense	2	-	softmax	-

Parámetros de entrenamiento* 955.362

Modelo 6

Arquitectura	Función	Filtros o perceptrones	Tamaño de filtro	Función de activación	Deserción (Dropout)
Función Lambda	Lambda	-	-	-	-
Convolutacional	Conv2D	20	(5, 5)	relu	-
Submuestreo	MaxPooling2D	-	(2, 2)	-	0.4
Convolutacional	Conv2D	50	(5, 5)	relu	-
Submuestreo	MaxPooling2D	-	(2, 2)	-	0.4
Aplanar	Flatten	-	-	-	-
Capa densa	Dense	500	-	relu	0.4
Capa de salida	Dense	2	-	softmax	-

Parámetros de entrenamiento* 6.328.072

*Parámetros de entrenamiento de las capas internas

Tabla III Arquitecturas de las redes neuronales convolucionales multimodales para tres imágenes de entrada

Modelo 7					
Arquitectura	Función	Filtros o perceptrones	Tamaño de filtro	Función de activación	Deserción (Dropout)
Función lambda ¹	Lambda	-	-	-	-
Convolutacional ¹	Conv2D	16	(3, 3)	elu	0.4
Convolutacional ¹	Conv2D	16	(3, 3)	elu	-
Submuestreo ¹	MaxPooling2D	-	(2, 2)	-	-
Aplanar ¹	Flatten	-	-	-	-
Función lambda ²	Lambda	-	-	-	-
Convolutacional ²	Conv2D	16	(3, 3)	elu	0.4
Convolutacional ²	Conv2D	16	(3, 3)	elu	-
Submuestreo ²	MaxPooling2D	-	(2, 2)	-	-
Aplanar ²	Flatten	-	-	-	-
Función lambda ³	Lambda	-	-	-	-
Convolutacional ³	Conv2D	16	(3, 3)	elu	0.4
Convolutacional ³	Conv2D	16	(3, 3)	elu	-
Submuestreo ³	MaxPooling2D	-	(2, 2)	-	-
Aplanar ³	Flatten	-	-	-	-
Concatenar	Concatenate	-	-	-	-
Capa densa	Dense	16	-	relu	0.4
Capa densa	Dense	8	-	relu	0.4
Capa densa	Dense	4	-	relu	0.4

Capa de salida	Dense	2	-	softmax	-
----------------	-------	---	---	---------	---

Parámetros de entrenamiento* 725.302

Modelo 8					
Arquitectura	Función	Filtros o perceptrones	Tamaño de filtro	Función de activación	Deserción (Dropout)
Función lambda ¹	Lambda	-	-	-	-
Convolutacional ¹	Conv2D	32	(3, 3)	elu	-
Submuestreo ¹	MaxPooling2D	-	(2, 2)	-	0.4
Convolutacional ¹	Conv2D	64	(3, 3)	elu	-
Submuestreo ¹	MaxPooling2D	-	(2, 2)	-	0.4
Aplanar ¹	Flatten	-	-	-	-
Función lambda ²	Lambda	-	-	-	-
Convolutacional ²	Conv2D	32	(3, 3)	elu	-
Submuestreo ²	MaxPooling2D	-	(2, 2)	-	0.4
Convolutacional ²	Conv2D	64	(3, 3)	elu	-
Submuestreo ²	MaxPooling2D	-	(2, 2)	-	0.4
Aplanar ²	Flatten	-	-	-	-
Función lambda ³	Lambda	-	-	-	-
Convolutacional ³	Conv2D	32	(3, 3)	elu	-
Submuestreo ³	MaxPooling2D	-	(2, 2)	-	0.4
Convolutacional ³	Conv2D	64	(3, 3)	elu	-
Submuestreo ³	MaxPooling2D	-	(2, 2)	-	0.4
Aplanar ³	Flatten	-	-	-	-
Concatenar	Concatenate	-	-	-	-
Capa densa	Dense	128	-	relu	0.4
Capa densa	Dense	32	-	relu	0.4
Capa de salida	Dense	2	-	softmax	-

Parámetros de entrenamiento* 5.796.898

Modelo 9					
Arquitectura	Función	Filtros o perceptrones	Tamaño de filtro	Función de activación	Deserción (Dropout)
Función lambda ¹	Lambda	-	-	-	-
Convolutacional ¹	Conv2D	32	(3, 3)	elu	-
Submuestreo ¹	MaxPooling2D	-	(2, 2)	-	0.4
Convolutacional ¹	Conv2D	32	(3, 3)	elu	-
Submuestreo ¹	MaxPooling2D	-	(2, 2)	-	0.4
Convolutacional ¹	Conv2D	64	(3, 3)	elu	-
Submuestreo ¹	MaxPooling2D	-	(2, 2)	-	0.4
Convolutacional ¹	Conv2D	128	(3, 3)	elu	-
Submuestreo ¹	MaxPooling2D	-	(2, 2)	-	0.4
Aplanar ¹	Flatten	-	-	-	-
Función lambda ²	Lambda	-	-	-	-
Convolutacional ²	Conv2D	32	(3, 3)	elu	-
Submuestreo ²	MaxPooling2D	-	(2, 2)	-	0.4
Convolutacional ²	Conv2D	32	(3, 3)	elu	-
Submuestreo ²	MaxPooling2D	-	(2, 2)	-	0.4
Convolutacional ²	Conv2D	64	(3, 3)	elu	-
Submuestreo ²	MaxPooling2D	-	(2, 2)	-	0.4
Convolutacional ²	Conv2D	128	(3, 3)	elu	-
Submuestreo ²	MaxPooling2D	-	(2, 2)	-	0.4
Aplanar ²	Flatten	-	-	-	-

Función lambda ³	Lambda	-	-	-	-
Convolutacional ³	Conv2D	32	(3, 3)	elu	-
Submuestreo ³	MaxPooling2D	-	(2, 2)	-	0.4
Convolutacional ³	Conv2D	32	(3, 3)	elu	-
Submuestreo ³	MaxPooling2D	-	(2, 2)	-	0.4
Convolutacional ³	Conv2D	64	(3, 3)	elu	-
Submuestreo ³	MaxPooling2D	-	(2, 2)	-	0.4
Convolutacional ³	Conv2D	128	(3, 3)	elu	-
Submuestreo ³	MaxPooling2D	-	(2, 2)	-	0.4
Aplanar ³	Flatten	-	-	-	-
Concatenar	Concatenate	-	-	-	-
Capa densa	Dense	512	-	relu	0.4
Capa densa	Dense	128	-	relu	0.4
Capa densa	Dense	32	-	relu	0.4
Capa de salida	Dense	2	-	softmax	-

Parámetros de entrenamiento* 2.540.546

Modelo 10					
Arquitectura	Función	Filtros o perceptrones	Tamaño de filtro	Función de activación	Deserción (Dropout)
Función lambda ¹	Lambda	-	-	-	-
Convolutacional ¹	Conv2D	20	(5, 5)	elu	-
Submuestreo ¹	MaxPooling2D	-	(2, 2)	-	0.4
Convolutacional ¹	Conv2D	50	(5, 5)	elu	-
Submuestreo ¹	MaxPooling2D	-	(2, 2)	-	0.4
Aplanar ¹	Flatten	-	-	-	-
Función lambda ²	Lambda	-	-	-	-
Convolutacional ²	Conv2D	20	(5, 5)	elu	-
Submuestreo ²	MaxPooling2D	-	(2, 2)	-	0.4
Convolutacional ²	Conv2D	50	(5, 5)	elu	-
Submuestreo ²	MaxPooling2D	-	(2, 2)	-	0.4
Aplanar ²	Flatten	-	-	-	-
Función lambda ³	Lambda	-	-	-	-
Convolutacional ³	Conv2D	20	(5, 5)	elu	-
Submuestreo ³	MaxPooling2D	-	(2, 2)	-	0.4
Convolutacional ³	Conv2D	50	(5, 5)	elu	-
Submuestreo ³	MaxPooling2D	-	(2, 2)	-	0.4
Aplanar ³	Flatten	-	-	-	-
Concatenar	Concatenate	-	-	-	-
Capa densa	Dense	500	-	relu	0.4
Capa densa	Dense	32	-	relu	0.4
Capa de salida	Dense	2	-	softmax	-

Parámetros de entrenamiento* 17.596.308

5.1.11. Entrenamiento redes CNN

El entrenamiento de las redes neuronales convolucionales se realizó con el 80% de los datos de cada grupo, un tamaño de lote de 16 muestras y se ejecutó durante 50 épocas (número de veces que se entren la red) para alcanzar la máxima precisión posible. Sin embargo, se implementó la función 'EarlyStopping' para detener el

entrenamiento en ausencia de una mejora en la pérdida del modelo, además, todos estos modelos se almacenaron automáticamente con la función 'ModelCheckpoint'. Cabe señalar que el comportamiento de los modelos se observó con validación cruzada, sin embargo, su precisión también se calculó por concordancia entre las etiquetas originales y clasificadas por la red neuronal, utilizando el 20% de los datos restantes.

En primer lugar, se entrenaron los seis modelos de una sola imagen (ver Tabla II) con cada conjunto de datos obtenidos de los cuatro métodos, cada representación de los tres cortes, con contornos, sin contornos, con sus tamaños originales y escalados. En segundo lugar, las cuatro redes multimodales de tres imágenes (ver Tabla III) se entrenaron de manera similar a las anteriores, pero con la única distinción de que en este caso las imágenes de los cortes ya no se usaban individualmente, sino con las tres simultáneamente.

La información detallada de todos los algoritmos implementados, desde los diferentes métodos de reducción de volumen, los diseños de red y hasta la capacitación de todas las arquitecturas están disponibles en GitHub (<https://github.com/LxMera/Convolutional-Neural-Network-for-the-classification-of-independent-components-of-rs-fMRI>)

5.2. Resultados

5.2.1. Clasificación de componentes independientes

La descomposición ICA de un solo sujeto para el primer grupo, entregó en promedio 52 ± 15.92 componentes independientes por sujeto con sus respectivos mapas espaciales y series de tiempo, encontrándose que el 58.07% correspondían a componentes asociadas a ruido y 41.92% contaban con características de actividad neuronal. Para el segundo grupo de sujetos, se contaban con 228 ± 35 componentes independientes, donde el 89.83% correspondían a componentes asociadas a ruido y solo el 10.17% se etiquetaron como actividad neuronal.

5.2.2. Reducción de dimensiones

Con las series de tiempo obtenidas por la descomposición de ICA se calcularon las 27 características descritas en la metodología y se encontró alta correlación entre algunas de ellas. En la *Figura 14* se observa la matriz de correlaciones obtenida con los coeficientes de correlación de Pearson, para la combinación de todas las posibles variables.

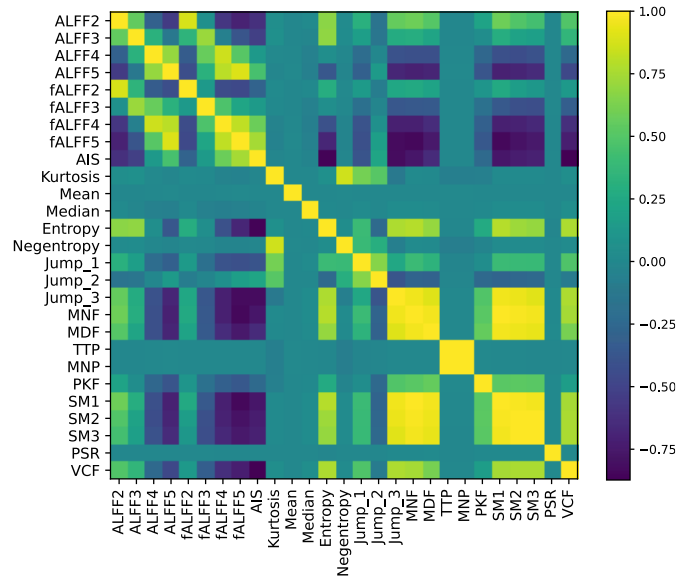


Figura 14 Matriz de coeficientes de correlación entre las 27 características espectro-temporales.

A las 27 características se les aplicó el análisis de componentes principales, donde se encontró que el 95.73% de la varianza de los datos se puede explicar mediante 11 componentes principales y hasta el 99.99% con 24. En la *Figura 15* se grafica el porcentaje de variabilidad explicada en función del número de componentes principales en la reducción de dimensiones por PCA.

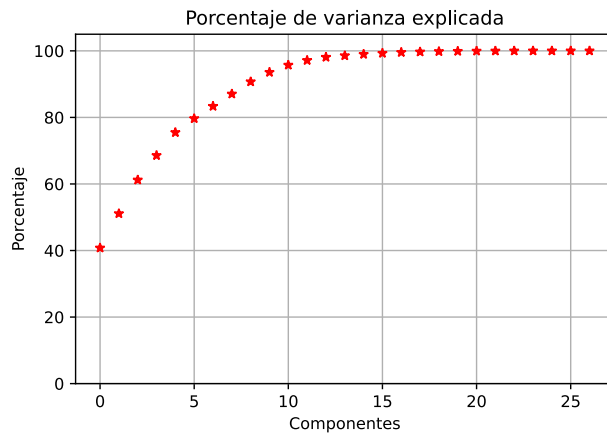


Figura 15 Porcentaje de varianza explicada en función de número de variables latentes obtenidos del análisis de componentes principales de las 27 características espectro-temporales.

Por otro lado, se validó la precisión para cada característica de manera individual, mediante la saturación de las redes. En la *Figura 16* se muestra la precisión promedio de los cuatro modelos, donde se observa que 5 de las características tuvieron valores superiores al 90%, siendo estas: la amplitud de fluctuación de baja frecuencia (ALFF) en las bandas de 0.01 a 0.027 Hz y de 0.027 a 0.073 Hz, la amplitud de baja frecuencia fraccional en las mismas 2 bandas y función de amplitud de salto descrita en la ecuación (1).

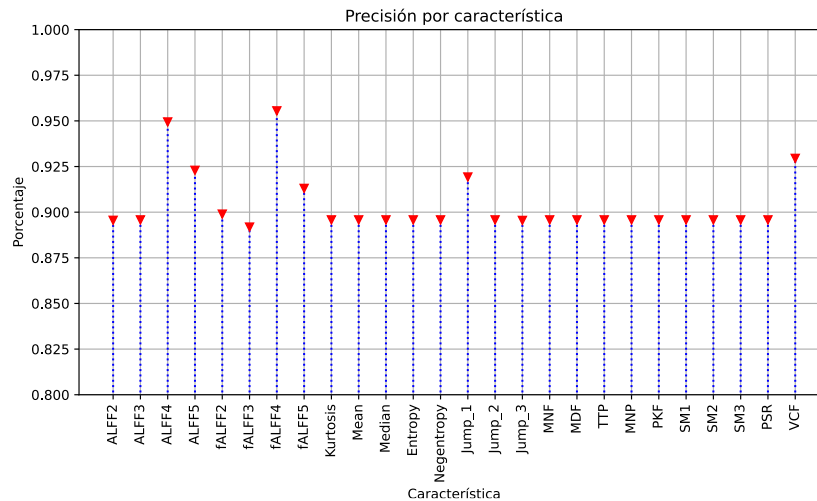


Figura 16 Promedio del porcentaje de precisión de clasificación en cada una de las variables validadas en los cuatro modelos.

5.2.3. Entrenamiento de las redes MLP

En primer lugar, los modelos fueron entrenados con el primer grupo de sujetos clasificados de manera manual y con todas las características. En segundo lugar, se entrenaron con el segundo grupo de sujetos con todas las características, con las variables latentes obtenidas en la reducción mediante el análisis de componentes principales y con las 5 características de mayor precisión de clasificación individual.

Los resultados del entrenamiento para el grupo 1 y 2, con todas las características, se muestran en *Figura 17* y *Figura 17* respectivamente, donde la gráfica superior representa el comportamiento de la pérdida de los cuatro modelos en función de las épocas de entrenamiento, observándose un comportamiento descendiente tanto para datos de entrenamiento como con los de prueba y en la gráfica inferior se muestra el incremento en la precisión de los 4 modelos para los mismos conjuntos de datos, es decir, datos de entrenamiento y de validación.

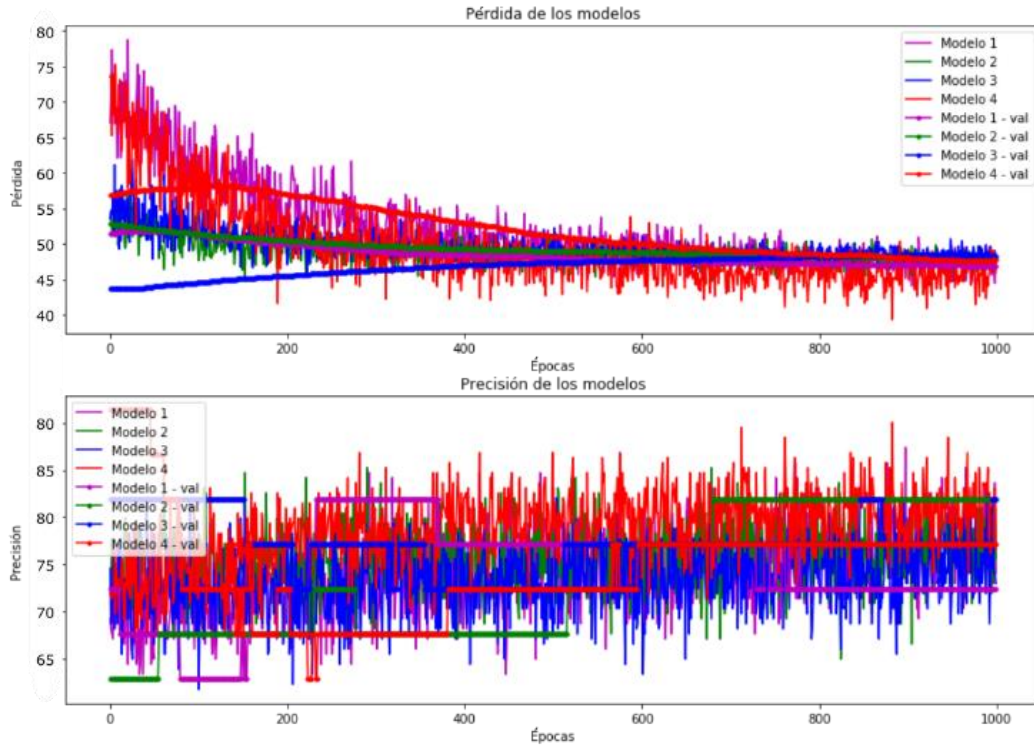


Figura 17 Comportamiento del entrenamiento de los 4 modelos con todas las características del primer grupo de sujetos. La primera gráfica presenta la pérdida de los 4 modelos (ver *Tabla 1*) en función de las épocas para los datos de entrenamiento y el conjunto de validación cruzada. La segunda de ellas muestra la precisión en función de las épocas.

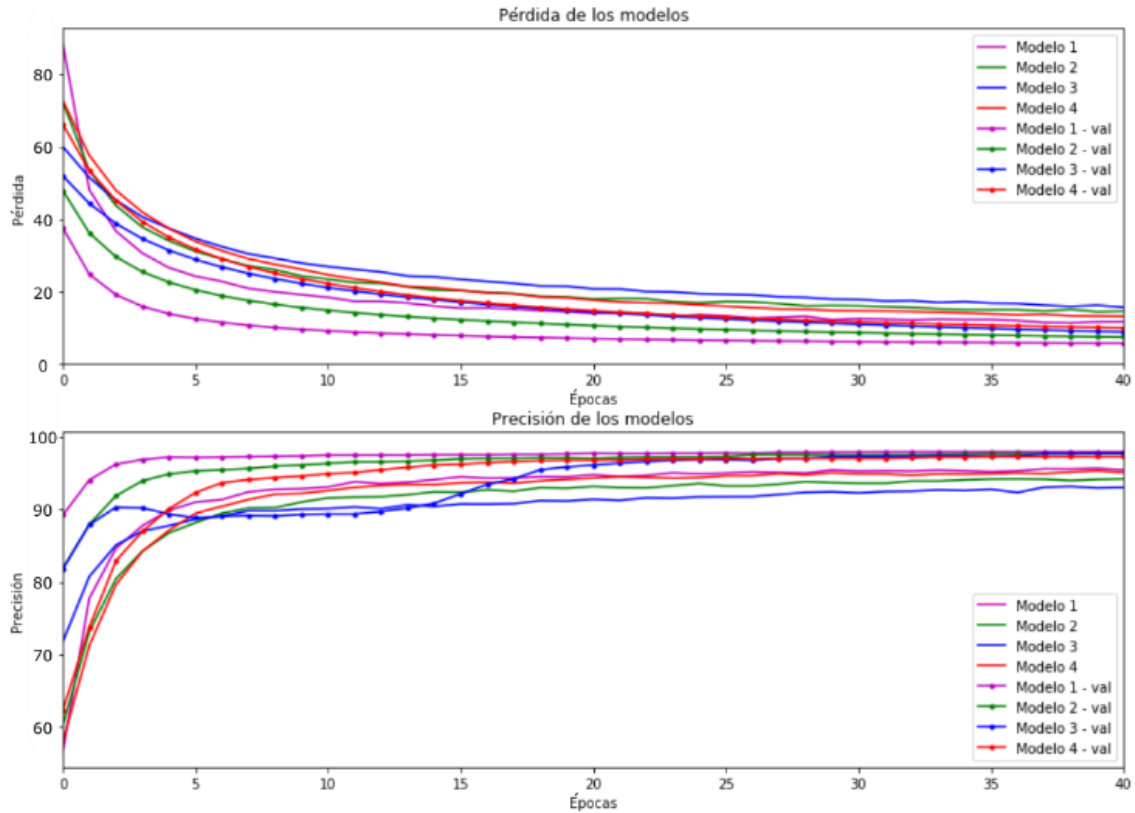


Figura 18 Comportamiento del entrenamiento de los 4 modelos con todas las características para el segundo grupo de sujetos. La primera gráfica presenta la pérdida de los 4 modelos (ver *Tabla I*) en función de las épocas para los datos de entrenamiento y el conjunto de validación. La segunda de ellas muestra la precisión en función de las épocas.

Se validó la precisión de las redes realizando la reducción de dimensiones mediante PCA, variando el número de componentes principales desde 11 hasta 27 componentes principales para los cuatro modelos. En la *Figura 19* se observan estos valores de precisión, sin embargo, no se aprecia ningún cambio significativo relevante y la precisión se reduce para un número de componentes menor.

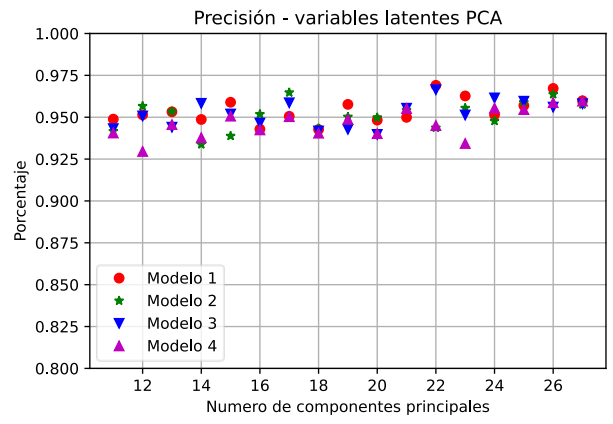


Figura 19 Precisión en los 4 modelos entrenados con diferente número de componentes principales, abarcando un porcentaje de variabilidad desde el 95 hasta el 100% de la varianza del modelo.

Finalmente, se entrenaron los 4 modelos con las 5 características de mayor precisión (ver *Figura 16*), observándose un comportamiento similar al entrenamiento con todas las características como se muestra en la *Figura 20*, sin embargo, la precisión aumentó por encima del 97%. En la *Tabla IV* se resume los resultados más relevantes obtenidos en este estudio.

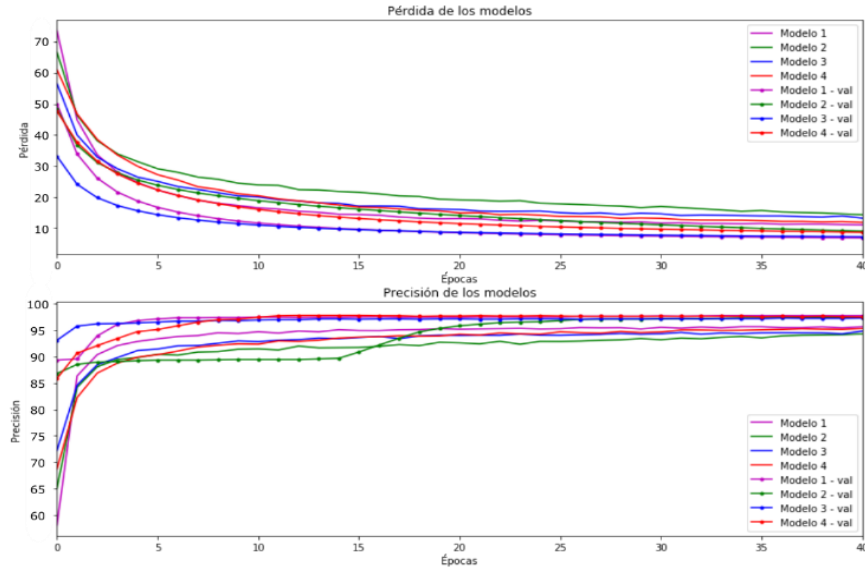


Figura 20 Comportamiento del entrenamiento de los 4 modelos con las cinco características de mayor precisión para el segundo grupo de sujetos. La primera gráfica presenta la pérdida de los 4 modelos (ver *Tabla I*) en función de las épocas para los datos de entrenamiento y el conjunto de validación. La segunda de ellas muestra la precisión en función de las épocas

Tabla IV Porcentaje de acierto de los modelos en la clasificación

Grupo	Precisiones por modelo			
	1	2	2	2
Características	25	25	11*	5**
modelo 1	75.38%	97.07%	94.60%	97.14%
modelo 2	77.31%	96.96%	93.95%	97.12%
modelo 3	79.23%	97.25%	94.12%	97.28%
modelo 4	75.38%	96.90%	93.33%	97.16%

* Componentes principales

**Características de mayor precisión

5.2.4. Reducción de volúmenes

Se redujeron 36.828 volúmenes de los componentes independientes de los siete grupos considerados (Ver **tabla V**). En cada volumen se encontró una representación de los cortes axial, coronal y sagital generados por los 4 métodos diferentes, además, se generó el contorno máximo de cada corte derivado de la reducción por PCA, agregado posteriormente en el canal rojo de la correspondiente imagen. Cabe señalar que las redes fueron entrenadas con las imágenes originales, las reducciones, con y sin contornos. Sin embargo, todas las imágenes que se presentarán a continuación tienen estos contornos para tener una mejor interpretación de la posición espacial de los componentes independientes.

El proceso de reducción de volumen alcanzó una disminución en los pesos de hasta 2 órdenes de magnitud, tanto en las imágenes con las dimensiones originales equivalentes a las dimensiones del volumen, como con las imágenes escaladas. Los resultados se muestran en la Tabla V.

Tabla V Peso promedio por componente

Grupo	Puntos de tiempo	Componentes totales	Peso promedio por componente (Kilobits)		
			volumen	imagen (Original)	imagen (Escalada)
Hcp_HP2000**	1200	22877	851.03	9.97	7.97
Whii_MB6*	460	3346	233.45	9.86	7.85
Whii_MB6*	1000	1797	239.65	10.48	7.7
Whii_Standard*	200	3205	125.1	5.85	7.45
Standard*	145	1678	802.72	4.22	6.85
Standard*	180	329	868.39	5.7	7.35
Standard*	200	3596	242.35	5.81	7.43

**Human Connectome Project

*Whitehall II imaging study

La *Figura 21* muestra cuatro componentes independientes asociados con la actividad neuronal, seleccionados al azar y generados por los cuatro métodos diferentes de reducción de volumen. Los métodos entregaron imágenes similares donde se exhiben regiones con algunas características sobresalientes con respecto al resto de la región del cerebro y, por lo tanto, donde esto supone la correspondencia de la posición espacial del componente asociado con dicha actividad. Por otra parte, el método 2 es el mapa espacial real del componente en su corte axial medio, en consecuencia, debido a la correlación espacial manifestada por los otros métodos, se puede inferir que estos retienen parte de las características espaciales de los componentes, por ejemplo, el componente A en el método 2 muestra que esta se encuentra en la parte posterior del cerebro, región que también se destaca en las otras imágenes, como en el caso de PCA, donde la parte posterior del cerebro tiene un color homogéneo entre el cian y el magenta, en contraste con las otras regiones que tienen una distribución más irregular de colores. Asimismo, el método 3 también presenta este comportamiento en la parte posterior, sin embargo, en este caso la región donde se revela el componente independiente se visualiza mejor. Finalmente, el método 4 no es una excepción y presenta este mismo comportamiento, sin embargo, debido a la incorporación del filtro no lineal de tamaño 7×7 todos los elementos pequeños dentro de la región del cerebro fueron eliminados generando una imagen más homogénea.

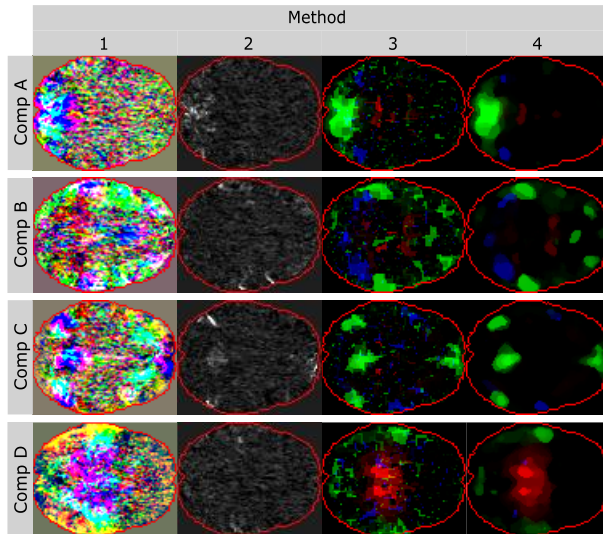


Figura 21 Resultados de cuatro componentes independientes clasificados como actividad neuronal. Reducción a imágenes axiales mediante los métodos de (1) análisis de componentes principales, (2) corte axial medio, (3) patrones binarios consecutivos y (4) patrones binarios consecutivos con un filtro no lineal (desenfoque medio). Las imágenes de los métodos 1, 3 y 4 son la combinación de los canales, producto de las reducciones, en una imagen RGB, lo que genera esa gran cantidad de colores.

Las Figura 22 y Figura 23 muestran las imágenes coronales y sagitales de los mismos 4 componentes independientes. En este caso, nuevamente se presentan comportamientos similares, regiones sobresalientes con alguna característica especial en relación con el resto de la región cerebral, además, estas regiones concuerdan con la posición espacial de las imágenes axiales, por ejemplo, el componente A en la Figura 21. se manifestó en la parte posterior del cerebro y, en la Figura 22, este mismo componente muestra algunas regiones en color azul, que coincide con la parte posterior del cerebro según lo establecido en el método de compresión de volumen (ver Figura 12 (a)). Además, en la Figura 23, este mismo componente se muestra en la parte posterior del cerebro.

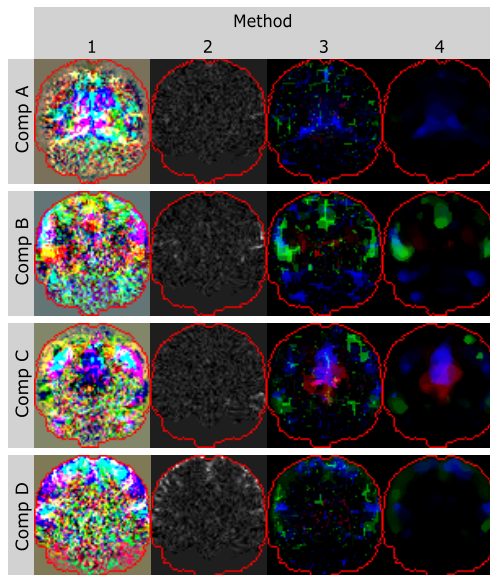


Figura 22 Resultados de cuatro componentes independientes clasificados como actividad neuronal. Reducción a imágenes coronales mediante los métodos de (1) análisis de componentes principales, (2) corte coronal

medio, (3) patrones binarios consecutivos y (4) patrones binarios consecutivos con un filtro no lineal (desenfoque medio). Las imágenes de los métodos 1, 3 y 4 son la combinación de los canales, producto de las reducciones, en una imagen RGB, lo que genera esa gran cantidad de colores.

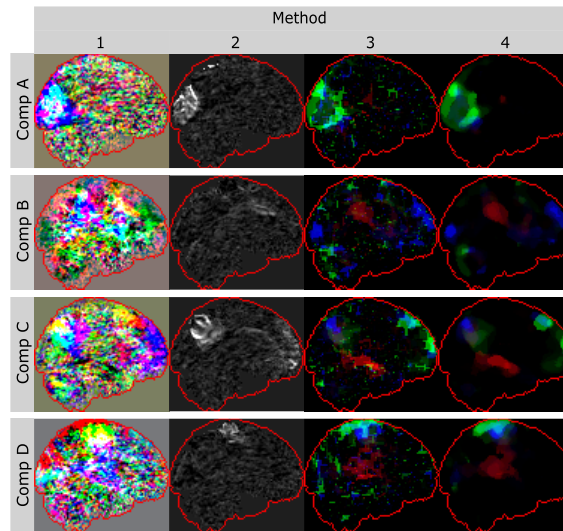


Figura 23 Resultados de cuatro componentes independientes clasificados como actividad neuronal. Reducción a imágenes sagitales mediante los métodos de (1) análisis de componentes principales, (2) corte sagital medio, (3) patrones binarios consecutivos y (4) patrones binarios consecutivos con un filtro no lineal (desenfoque medio). Las imágenes de los métodos 1, 3 y 4 son la combinación de los canales, producto de las reducciones, en una imagen RGB, lo que genera esa gran cantidad de colores.

Aunque las imágenes muestran coherencia espacial con respecto al componente independiente, vale la pena señalar que en estos cuatro casos las formas se asemejan a redes de conexión ya estudiadas de antemano, en otras palabras, los componentes A, B, C y D Tiene la forma de redes: sistema visual medial, saliencia, red de modo defecto y red sensoriomotora respectivamente [58].

Finalmente, para observar las diferencias entre los dos tipos de componentes, se seleccionaron al azar 4 componentes con ruido o carácter no neuronal. Las *Figura 24*, *Figura 25* y *Figura 26* muestran las representaciones axial, coronal y sagital obtenidas de los cuatro métodos de reducción, pero al contrario del caso anterior, estas imágenes ya no muestran el mismo comportamiento en los cuatro métodos, por ejemplo, en el primero de ellos, los componentes E y G tienen una distribución irregular en toda la región cerebral, en contraste con los dos últimos métodos que se manifiestan en la parte posterior del cerebro, además, a pesar de la coherencia espacial entre los tres cortes, las redes de conexión no corresponden cualquier red funcional conocida.

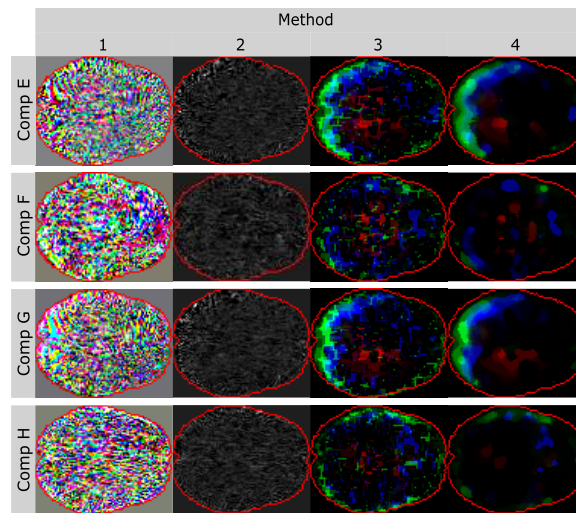


Figura 24 Resultados de cuatro componentes independientes clasificados como ruido. Reducción a imágenes axiales mediante los métodos de (1) análisis de componentes principales, (2) corte axial medio, (3) patrones binarios consecutivos y (4) patrones binarios consecutivos con un filtro no lineal (desenfoque medio). Las imágenes de los métodos 1, 3 y 4 son la combinación de los canales, producto de las reducciones, en una imagen RGB, lo que genera esa gran cantidad de colores.

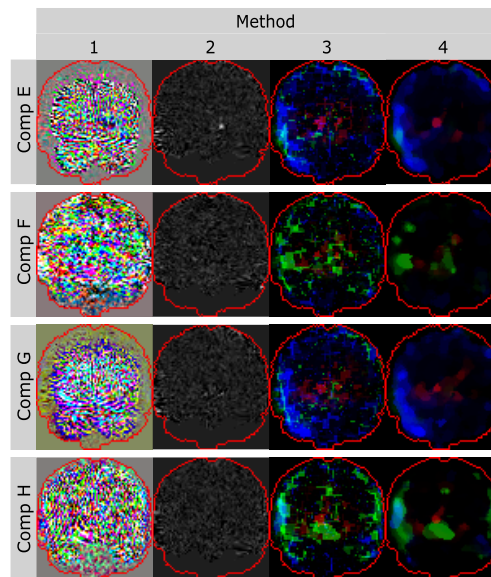


Figura 25 Resultados de cuatro componentes independientes clasificados como ruido. Reducción a imágenes coronales mediante los métodos de (1) análisis de componentes principales, (2) corte coronal medio, (3) patrones binarios consecutivos y (4) patrones binarios consecutivos con un filtro no lineal (desenfoque medio). Las imágenes de los métodos 1, 3 y 4 son la combinación de los canales, producto de las reducciones, en una imagen RGB, lo que genera esa gran cantidad de colores.

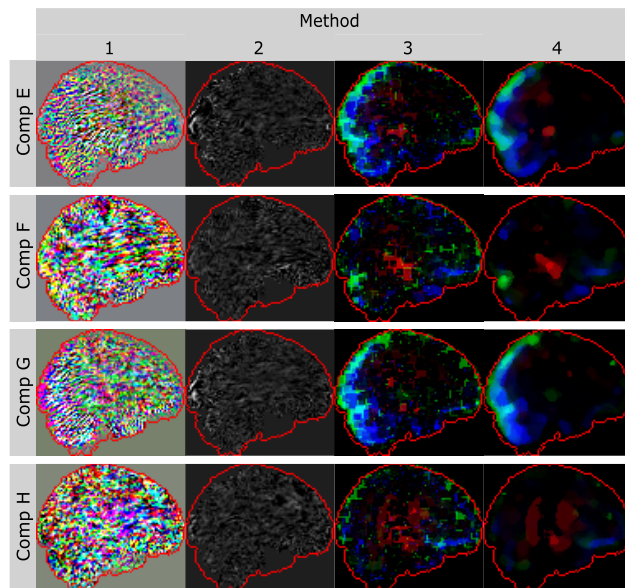


Figura 26 Resultados de cuatro componentes independientes clasificados como ruido. Reducción a imágenes sagitales mediante los métodos de (1) análisis de componentes principales, (2) corte sagital medio, (3) patrones binarios consecutivos y (4) patrones binarios consecutivos con un filtro no lineal (desenfoque medio). Las imágenes de los métodos 1, 3 y 4 son la combinación de los canales, producto de las reducciones, en una imagen RGB, lo que genera esa gran cantidad de colores.

5.2.5. Entrenamiento de las redes CNN

Se llevaron a cabo tres corridas de entrenamiento para las 2,464 redes de los 10 modelos base, de las cuales 352 redes corresponden a cada grupo, donde 176 fueron entrenadas con los tamaños originales y 176 con las imágenes escaladas. Por lo tanto, debido a una gran cantidad de redes, solo los resultados más relevantes se muestran en este documento, sin embargo, todos los detalles se almacenan en el repositorio de GitHub (<https://github.com/LxMera/Deep-Learning---denoising-rs-fMRI>).

5.2.5.1. Entrenamientos con bordes

Los criterios de clasificación de ruido se basan en la posición espacial de los componentes con respecto al límite entre la interfaz de la materia gris y el líquido cefalorraquídeo. Por lo tanto, se decidió entrenar las redes con y sin los contornos para establecer si estos límites generaban mayor poder de clasificación en redes neuronales convolucionales. Los valores de precisión de los datos de prueba entre las posibles combinaciones de los modelos y métodos se evaluaron mediante la prueba no paramétrica de Kruskal-Wallis entre las imágenes con y sin contorno. La Tabla VI muestra los valores de p obtenidos para todas las combinaciones posibles, donde todos dieron valores por debajo de 0.05 excepto el par Modelo 1-Método 4 y Modelo 9-Método 4, siendo claro que las combinaciones no tienen una diferencia estadística significativa a excepción de los pares mencionados, no obstante, estos están cerca de 0.05, lo que podría deberse a los valores atípicos generados en el entrenamiento.

Tabla VI Diferencia de medias mediante la prueba la kruskal-wallis sobre las imágenes con y sin contornos.

Prueba Kruskal-Wallis (valor p)				
Diferencias de medias entre:	Con y sin contornos			
	Método 1	Método 2	Método 3	Método 4
Modelo 1	0.54	0.25	0.59	0.05
Modelo 2	0.54	0.33	0.35	0.29
Modelo 3	0.63	0.82	0.72	0.53
Modelo 4	0.72	0.82	0.96	0.33
Modelo 5	0.76	0.79	0.09	0.86
Modelo 6	0.18	0.48	0.72	0.45
Modelo 7	0.81	0.12	0.81	0.08
Modelo 8	0.27	0.38	0.83	0.18
Modelo 9	0.49	0.28	0.66	0.05
Modelo 10	0.38	0.82	0.8	0.82

5.2.5.2. Entrenamiento con imágenes escaladas

Debido a las diferencias en las dimensiones de los volúmenes de los componentes independientes en los diferentes grupos, todas las redes fueron entrenadas con los tamaños originales y con las imágenes escaladas a los tamaños de 73×57 , 57×57 y 57×73 para los cortes axial, coronal y sagital respectivamente. El entrenamiento arrojó diferentes valores de precisión en validación cruzada y datos de prueba, que se compararon usando la prueba de Kruskal-Wallis entre las imágenes originales y escaladas. La Tabla VII muestra los valores de p obtenidos en la comparación previa, además, se validó la precisión entre la validación cruzada y los datos de prueba, donde, en ninguno de los casos el valor de p generó valores inferiores a 0.05, significando que no hay diferencia estadística significativa.

Tabla VII Diferencia de medias mediante la prueba kruskal-wallis entre imágenes originales escaladas y comparación entre la precisión por validación cruzada y validación por datos de prueba

Prueba de Kruskal-Wallis (Valor p)			
Diferencia de medias entre:	Original y escalada (Validación cruzada)	Original y escalada (Datos de prueba)	Validación de cruzada y datos de prueba
Modelo 1	0.23	0.19	0.24
Modelo 2	0.11	0.2	0.28
Modelo 3	0.37	0.06	0.06
Modelo 4	0.13	0.06	0.28
Modelo 5	0.56	0.13	0.07
Modelo 6	0.88	0.62	0.21
Modelo 7	0.73	0.83	0.94
Modelo 8	0.15	0.41	0.97
Modelo 9	0.35	0.52	0.4
Modelo 10	0.92	0.67	0.42

Las Tabla VIII y Tabla IX muestran las 172 precisiones máximas alcanzadas por los datos de prueba en todas las combinaciones posibles entre los modelos y los métodos de reducción de imagen. El entrenamiento proporcionó valores entre 0.8957 y 0.9944, lo equivalente a 89.57% y 99.44% en la precisión de la clasificación, siendo este último el valor generado por el quinto modelo a través del corte sagital derivado del tercer modelo de reducción de imagen con su respectivo contorno.

Tabla VIII Máxima precisión de clasificación alcanzada por los modelos entrenados con diferentes imágenes

Modelo 1	Sin contorno				Con contorno			
Corte	Met 1	Met 2	Met 3	Met 4	Met 1	Met 2	Met 3	Met 4
Axial	0.9744	0.8958	0.9803	0.9792	0.8969	0.9486	0.9751	0.9784
Coronal	0.9795	0.9371	0.9738	0.9812	0.9639	0.9347	0.9731	0.976
Sagital	0.9653	0.941	0.9736	0.9847	0.9047	0.9467	0.9751	0.9779
Modelo 2	Sin contorno				Con contorno			
Corte	Met 1	Met 2	Met 3	Met 4	Met 1	Met 2	Met 3	Met 4
Axial	0.9696	0.9572	0.9873	0.9823	0.9736	0.9462	0.9854	0.9801
Coronal	0.9764	0.9456	0.9847	0.9812	0.9777	0.9292	0.9845	0.9845
Sagital	0.9677	0.9521	0.9862	0.9871	0.965	0.9567	0.986	0.9862
Modelo 3	Sin contorno				Con contorno			
Corte	Met 1	Met 2	Met 3	Met 4	Met 1	Met 2	Met 3	Met 4
Axial	0.9825	0.9696	0.9882	0.9838	0.9806	0.9677	0.9882	0.9836
Coronal	0.9849	0.9639	0.9854	0.9849	0.9878	0.958	0.984	0.9838
Sagital	0.983	0.9631	0.9882	0.9884	0.9851	0.9707	0.9878	0.9878
Modelo 4	Sin contorno				Con contorno			
Corte	Met 1	Met 2	Met 3	Met 4	Met 1	Met 2	Met 3	Met 4
Axial	0.9847	0.9722	0.9875	0.984	0.9827	0.9722	0.9861	0.9843
Coronal	0.988	0.9642	0.9856	0.9847	0.9856	0.962	0.9847	0.9843
Sagital	0.986	0.9694	0.9889	0.9889	0.9865	0.9722	0.9884	0.9889
Modelo 5	Sin contorno				Con contorno			
Corte	Met 1	Met 2	Met 3	Met 4	Met 1	Met 2	Met 3	Met 4
Axial	0.9825	0.9778	0.9861	0.9814	0.9795	0.9677	0.9849	0.9827
Coronal	0.9858	0.9618	0.9825	0.9819	0.9856	0.9618	0.9814	0.983
Sagital	0.986	0.975	0.9893	0.9917	0.9869	0.9722	0.9944	0.9889
Modelo 6	Sin contorno				Con contorno			
Corte	Met 1	Met 2	Met 3	Met 4	Met 1	Met 2	Met 3	Met 4
Axial	0.9849	0.9712	0.9865	0.9838	0.9884	0.9727	0.9865	0.9843
Coronal	0.9882	0.9585	0.9847	0.9871	0.9884	0.9572	0.9867	0.9875
Sagital	0.9875	0.9692	0.9886	0.9895	0.9889	0.972	0.988	0.9897

Met: Método

Tabla IX Máxima precisión de clasificación alcanzada por los modelos multimodales entrenados con las tres imágenes

	Redes multimodales							
	Sin contorno				Con contorno			
	Met 1	Met 2	Met 3	Met 4	Met 1	Met 2	Met 3	Met 4
Modelo 7	0.8958	0.8958	0.9801	0.9871	0.979	0.8958	0.9843	0.9812
Modelo 8	0.9884	0.9792	0.9899	0.9899	0.9875	0.9816	0.9908	0.991
Modelo 9	0.9897	0.983	0.9917	0.9893	0.9908	0.9814	0.9897	0.9878
Modelo 10	0.9913	0.983	0.9908	0.9895	0.9921	0.9834	0.991	0.9889

Met: Método

Las *Figura 27* y *Figura 28* muestran la distribución de precisión en los diferentes cortes para validación cruzada y datos de prueba respectivamente, en función de los métodos de reducción de imagen. En los gráficos es posible observar que los métodos 3 y 4 tenían una distribución más compacta, tanto para imágenes con y sin contorno, además, es notorio que los rangos intercuartiles están por encima de 0.98 en contraste con los otros métodos, que incluso obtuvieron valores por debajo de 0.9. Además, estas cifras mostraron un comportamiento similar al descrito anteriormente, confirmando el enunciado derivado de la Tabla VII, donde el estadístico de prueba indicaba que no hay diferencias estadísticamente significativas entre estas precisiones de validación cruzada y los datos de prueba, también, esto refleja que no hubo sobreajuste en las redes.

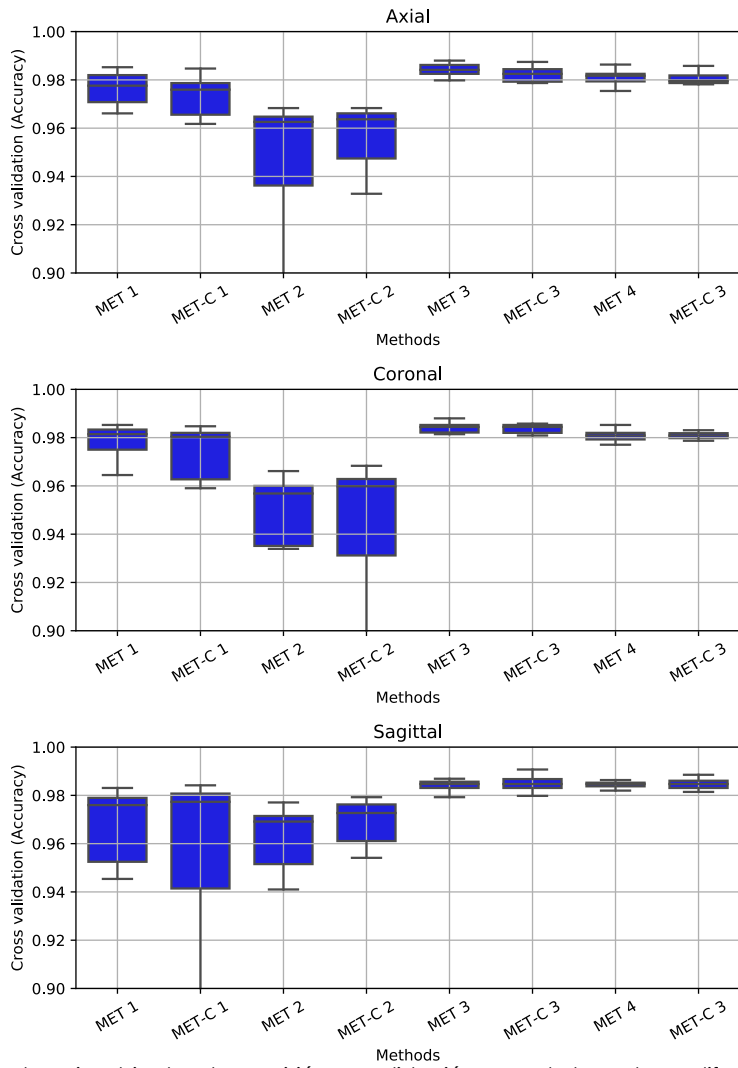


Figura 27 Gráfico de caja y bigotes de precisión en validación cruzada basada en diferentes métodos de reducción de imagen con y sin contornos.

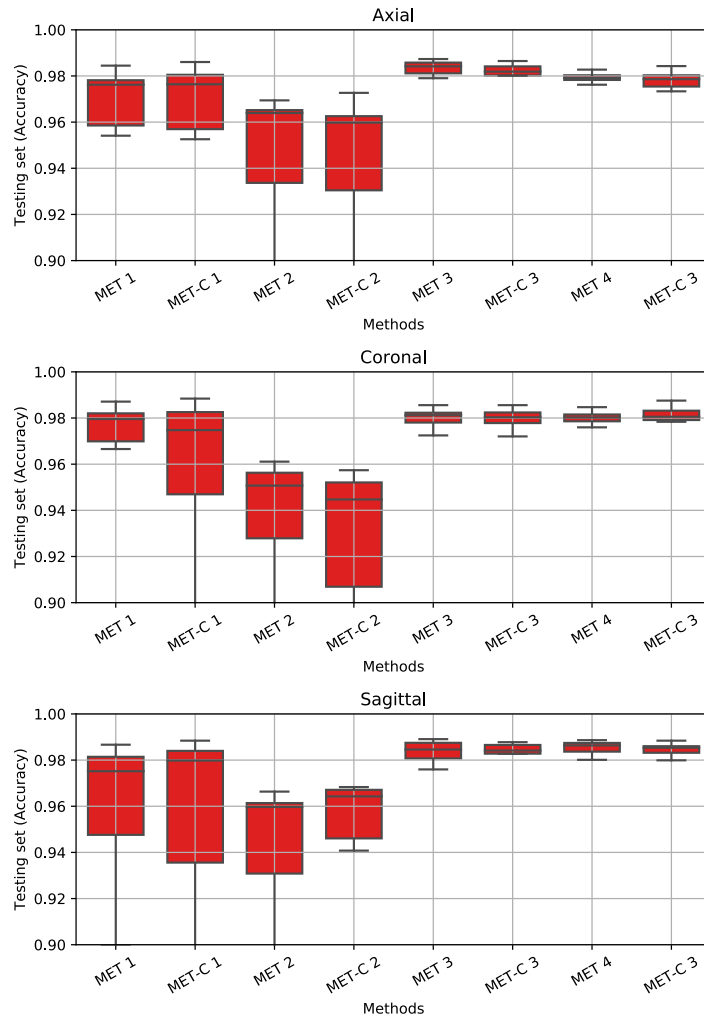


Figura 28 Gráfico de caja y bigotes de precisión en el conjunto de prueba basado en diferentes métodos de reducción de imagen con y sin contornos.

La *Figura 29* muestra nuevamente la distribución de precisión, sin embargo, en este caso, se realizó de acuerdo con los 10 modelos. Los primeros tres gráficos contienen los 6 modelos de una imagen en los diferentes sectores, mientras que el último gráfico muestra los 4 modelos multimodales. En este caso, los modelos 3, 4, 5 y 6 tuvieron comportamientos similares, presentando su rango intercuartil por encima de 0.95, de la misma manera, los modelos 8, 9 y 10 llevaron estos rangos por encima de 0.975, por el contrario, los modelos restantes mostraron una distribución más irregular alcanzando valores por debajo de 0.9.

Finalmente, la *Figura 30* muestra la precisión promedio alcanzada por los diferentes grupos seleccionados para el entrenamiento. El gráfico muestra que el grupo HP_hcp2000 alcanzó valores más altos en comparación con el resto de los grupos, tanto para la validación cruzada como para los datos de prueba, por el contrario, los grupos 145-Standard y 180-Standard tuvieron valores más bajos que los otros grupos en la validación cruzada, lo cual podría deberse en gran medida a los pocos componentes de estos grupos para el entrenamiento.

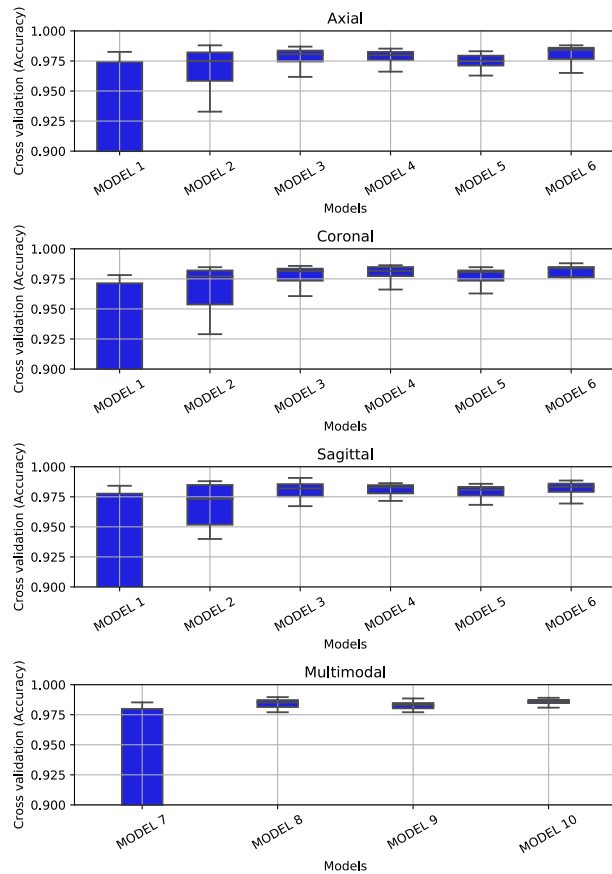


Figura 29 Diagrama de caja y bigotes de precisión en el conjunto de prueba en función de los diferentes modelos.

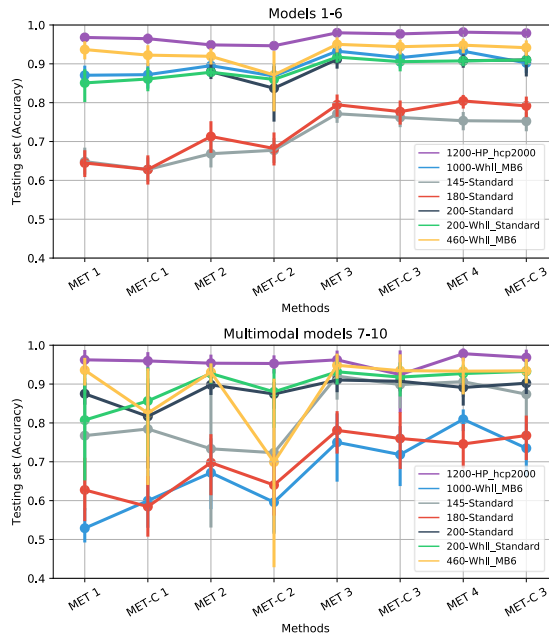


Figura 30 Precisión promedio en validación cruzada y datos de entrenamiento en función de los métodos de reducción para los 7 grupos de entrenamiento.

5.3. Discusión

La clasificación de artefactos, mediante el análisis de componentes independientes, es un tema activo dentro de los métodos de limpieza en imágenes de resonancia magnética funcional. Las estrategias propuestas van desde clasificaciones manuales [59], k vecinos más cercanos, árboles de decisión, máquinas de soporte vectorial, regresiones logísticas [35] e incluso combinaciones de estas. Un claro ejemplo de esto es ICA-FIX, clasificador que cuenta con una precisión cercana al 99%, pero su ejecución se basa en 180 características espaciales, espectrales y temporales [38]. Por lo anterior, se propuso una nueva estrategia para la eliminación de ruido automático mediante redes de aprendizaje profundo sobre las series de tiempo de imágenes de rs-fMRI, determinando cuáles de las componentes están asociadas a actividad neuronal y ruido a partir de 27 características espectro-temporales, de las cuáles las amplitudes de fluctuación de baja frecuencia (ALFF y fALFF) y almacenamiento de información activa no habían sido consideradas hasta el momento en eliminación de artefactos.

Las cuatro redes neuronales artificiales basadas en arquitecturas MLP lograron una precisión de clasificación superior al 97% utilizando únicamente 5 características de las 27 planteadas inicialmente, las cuales fueron: la amplitud de fluctuación de baja frecuencia (ALFF) en las bandas de 0.01 a 0.027 Hz y de 0.027 a 0.073 Hz, la amplitud de baja frecuencia fraccional (fALFF) en las mismas 2 bandas y una de funciones de amplitud de salto (Ver ecuación (9)).

Adicionalmente, en el caso de los diferentes entrenamientos, logramos observar que el primer grupo no obtuvo una precisión superior al 80% para ninguno de los 4 modelos, además, la precisión y la pérdida se mostraron bastante erráticas con respecto al segundo grupo, esto se generó por los pocos datos presentes en el primer grupo, lo cual no permite que la red ajuste sus pesos de manera eficiente en cada una de las épocas de entrenamiento, sin embargo, desde este punto las redes mostraron un porcentaje de precisión prometedor, confirmando la alta capacidad de aprendizaje de las redes *Deep Learning* [43] a pesar del número reducido de componentes usadas en el entrenamiento.

En el segundo grupo, la precisión y pérdida se mostraron más estables y, a pesar de ser entrenados con 1000 épocas, la estabilización ya se había manifestado después la época 40. No obstante, se continuó con esta configuración para obtener el menor valor de pérdida y la mayor precisión posibles con las 27 características, encontrándose valores cercanos al 97% de precisión, siendo una mejora significativa en la clasificación de las redes. Cabe resaltar, que a pesar utilizarse un número elevado de épocas de entrenamiento, estas redes no presentaron un sobreajuste, ya que contaban con un porcentaje de 40% en el abandono aleatorio de los nodos, método de regularización que se implementó con estas redes.

Por otro lado, el análisis de componentes principales PCA, no arrojó ningún resultado relevante en cuanto a mejorar o conservar el porcentaje de precisión en la reducción de dimensiones de las características, esto puede deberse a que el PCA genera nuevas componentes como una combinación lineal de las variables originales, donde estas abarcan la mayor varianza de las variables. Por lo tanto, en este caso la mayor varianza de estas 27 características no refleja totalmente el carácter de ruido o señal de las

componentes independientes, atenuando la capacidad de clasificación de las redes para un número menor de componentes principales.

Abordar la precisión individual de cada una de las características validada con la saturación de las mismas redes, resultó ser el método con mejores resultados para encontrar variables con mejor carácter de clasificación y reducir el número de características a ser implementadas con las redes. La amplitud de fluctuación de baja frecuencia fraccional, por sí solo, para la banda de 0.027 a 0.073 Hz, entregó una precisión superior al 95% y la implementación con las 5 variables de mayor precisión aumentó la precisión de todas las redes por encima del 97%. Cabe resaltar que a pesar de no alcanzarse la misma precisión obtenida por ICA-FIX el porcentaje de precisión es bastante alto en contraste con el número de características utilizadas para clasificar, es decir, 5 utilizadas en nuestra estrategia en contraste con las 180 utilizadas por ICA-FIX, mostrando el alto poder de discriminación de estas y confirmando la relevancia de incluir estas cuatro nuevas características de amplitud fluctuación de baja frecuencia en los métodos de eliminación de artefactos.

Por otro lado, aunque las redes neuronales sobre las series de tiempo, es decir las redes basadas en arquitecturas MLP entregaron buenos resultados, hoy en día existen técnicas prometedoras con redes neuronales convolucionales (CNN) para la tarea de clasificación que incluso ha excedido la capacidad humana [42], por lo tanto, también propusimos abordar esta estrategia de aprendizaje profundo para la clasificación de artefactos usando neuronal convolucional redes aplicadas en la representación de los mapas espaciales de los componentes, incluido un nuevo método que diseñamos, basado en el principio de patrones binarios locales y que denominamos reducción por patrones binarios consecutivos (RCBP).

En primer lugar, las cuatro reducciones o representaciones diferentes de los mapas espaciales tuvieron una disminución de hasta dos órdenes de magnitud y, aunque esto no fue validado por nosotros, ya implica una reducción en la carga computacional en el caso de la clasificación con los datos puros como se suele hacer con algoritmos de aprendizaje profundo [34]. Además, la representación con el nuevo método RCBP que sugerimos, nos permitió identificar fácilmente las redes de conexión en los componentes independientes asociados con la actividad neuronal, un proceso que no es tan fácil de interpretar cuando se trata de la extracción de características, como en el caso de ICA-AROMA, que tiene características como el porcentaje de superposición del componente sobre la materia gris o el líquido cefalorraquídeo [37].

En segundo lugar, todas las redes neuronales convolucionales implementadas alcanzaron una precisión de clasificación de 89.57 a 99.44%, alcanzando este valor máximo con la ayuda de la imagen sagital con contorno generado por nuestro método RPBC, del mismo modo, esto demostró tener una mayor estabilidad frente a los otros métodos ya que todos sus valores, bajo las diferentes configuraciones, generaron distribuciones con sus rangos intercuartiles por encima del 98% de precisión, superando incluso el 96% alcanzado por ICA-AROMA [37]. Del mismo modo, la red de mayor precisión, es decir, el modelo de 4 capas convolucionales y tres capas densas superó la precisión de clasificación máxima alcanzada hasta ahora por ICA-FIX [38], entrenada con los mismos datos del proyecto del conectoma humano.

Adicionalmente, la comparación mediante el estadístico de prueba de Kruskal-Wallis permitió evidenciar que no hay diferencia estadística significativa entre la validación cruzada y los dos datos de prueba, lo que indica claramente que no hubo un problema

de sobre entrenamiento durante el aprendizaje de los modelos. También se encontró que no hay diferencia estadísticamente significativa entre las imágenes con y sin contorno y entre el tamaño original y las imágenes escaladas, condiciones que dan paso a una red neuronal convolucional de tamaño estándar.

5.4. Conclusión

Hemos diseñado cuatro redes neuronales artificiales con arquitecturas MLP, estudiado 4 métodos de reducción o representación de los volúmenes de las componentes independientes y hemos combinamos dicha reducción con 10 redes CNN, todo esto con el fin de clasificar automáticamente artefactos en imágenes de resonancia magnética funcional en estado de reposo mediante el análisis de componentes independientes (ICA), en donde en el primero de los casos logramos precisiones de clasificación superior al 97%. Además, las redes MLP se basaron en la descomposición ICA y se emplearon únicamente las características temporales de: Amplitud de fluctuación de baja frecuencia (ALFF) en las bandas de 0.01 a 0.027 Hz y de 0.027 a 0.073 Hz, amplitud de baja frecuencia fraccional en las mismas 2 bandas y una de funciones de amplitud de salto (Ver ecuación (1)). En el segundo de los casos, introducimos un nuevo método de reducción de volúmenes, el cual llamamos reducción por patrones binarios consecutivos (RPBC), donde se demostró que este redujo hasta en 2 órdenes de magnitud los pesos originales y logró conservar las características espaciales de los componentes para la clasificación con las redes CNN. La implementación de esta reducción con las diez arquitecturas CNN, mostraron precisiones cercanas al 98% de clasificación en la mayoría de las redes, e incluso se encontró una red con una precisión del 99% excediendo el poder de clasificación de los métodos existentes hasta ahora basados en aprendizaje automático.

6. Implementar un flujo de procesamiento automatizado para imágenes fMRI que integre las herramientas de limpieza de señal BOLD y métricas de conectoma

Todo los procesos implicados en la adquisición de la fMRI, generar cambios o diferencias inherentes al proceso, por ejemplo, el paradigma en estado de reposo no cuenta con un diseño de tarea estructurado, por lo tanto, el inicio del registro puede no ser consistente debido a la simple experiencia del registro en cuestión [60], siendo necesario eliminar estos primeros volúmenes. La secuencia corte intercalada que se suele utilizar para evitar la excitación de cortes consecutivos genera un desfase temporal en la adquisición de los volúmenes de tiempo que idealmente deberían corresponder a un mismo punto de tiempo [61]. El registro temporal para un estudio funcional con tiempo de repetición en el orden de los segundo requiere exploraciones que van desde los 4 minutos e incluso hasta los 30 o 40 minutos, donde inevitablemente existe la posibilidad de generarse algún movimiento durante la exploración, la cual es una de las principales razones de oclusión de la señal de interés, ya que el movimiento en consecuencia evita que los cortes se magnetice de manera adecuada, generando inhomogeneidades de campo magnético que alteran la señal BOLD [14]. Los enfoques más comunes para corregir este efecto se basa en la transformación de cuerpo rígido de diferentes grados de libertad o parámetros de movimiento, utilizándose hasta 12 parámetros, 3 de traslación, 3 de rotación y 6 generados a partir de las derivadas temporales de estos mismos [62].

Por otro lado, para que el estudio o sujeto sea comparable con otros sujetos, este debe estar contenido en un mismo espacio, como el espacio estándar del Instituto Neurológico de Montreal (MNI) ampliamente utilizado en estudio funcionales y anatómicos [63], siendo necesario, primero eliminar regiones del registro que no tienen interés, como por ejemplo el cráneo del sujeto, segundo identificar o segmentar el encéfalo en los principales tejidos, líquido cefalorraquídeo, materia blanca y materia gris, para posteriormente llevar el registro del sujeto al espacio MNI mediante una transformación lineal o una no lineal .

En ese sentido, eliminar el ruido inherente a las imágenes de resonancia magnética es una necesidad vigente que se ha venido abordado desde diferentes técnicas combinadas en esquemas de preprocesamiento (conocidos como pipelines) [64], que a su vez se han implementado desde diferentes entornos o software, abordando uno o varios problemas específicos que depende de las características de adquisición de la imagen, como por ejemplo el paradigma que se está utilizando para evocar la actividad neuronal. En consecuencia, hoy en día existen muchos esquemas de preprocesamiento y herramientas tales: DPABI, REST, C-PAC, GREYNA, CCS, CONN, SPM, FSL, AFNI, Freesurfer, Nilearn, Nibabel y muchos otros más [65]–[74], [75]–[77].

Por otro lado, una vez obtenida la imagen de resonancia magnética con la menor cantidad de ruido posible, se procede a calcular las métricas que permitan caracterizar la actividad neuronal o la estructura de los sujetos de estudio a fin de establecer diferencias manifestadas por alguna fisiopatología. Las métricas al igual que

los esquemas de preprocesamiento son abordados desde diferentes herramientas y modelos que mejor se ajusten a la patología de interés o al estudio en cuestión, donde es posible encontrar métricas para determinar la amplitud de fluctuación de baja frecuencia, la homogeneidad regional de la actividad neuronal, la conectividad funcional entre regiones, grados de centralidad, centralidad de vectores propios y muchas otras medidas de la actividad neuronal.

En este orden de ideas, generar métricas de caracterización de la actividad neuronal se torna en una tarea titánica al existir tantos y tan diversos softwares, además, no existe un procesamiento ya establecido para las imágenes de resonancia magnética funcional en estado de reposo, lo que supone la existencia de infinidad de procesamiento que pueden limpiar la señal, pero no sin repercutir o alterar parcialmente la señal de interés. Adicionalmente, tratar de asumir esta tarea sin un proceso automatizado supone un empleo de tiempo enorme y un gasto de recursos humanos innecesario. Por lo tanto, en esta sección se abordan estos problemas, diseñando e implementando diferentes flujos de procesamiento automatizado integrando las más actuales y más novedosas herramientas e incluyendo los desarrollos propios de este trabajo de investigación con el fin de estudiarlos en este trabajo.

6.1. Metodología

6.1.1. Software

Todos los procesos y funciones a implementar se desarrollaron en el entorno de desarrollo spyder bajo el lenguaje de programación python, integrando los software y librerías de FSL, AFNI, Nilearn, Nibabel, ICA_AROMA, networkx, Keras y Tensorflow, mediante el software de código abierto Nipype [78], estableciéndose el flujo de procesamiento automatizado para limpiar y calcular las métricas sobre las imágenes de resonancia magnética funcional. Cada función o proceso se definió como un nodo en el flujo de trabajo, siendo las aristas de conexión los datos de entrada y salida de cada nodo como se esquematiza en la *Figura 31*.

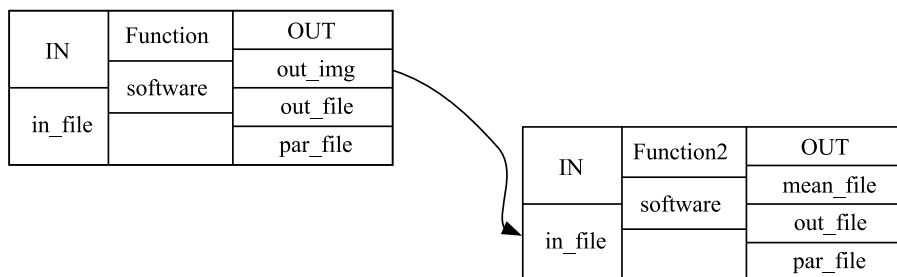


Figura 31 Conexión entre funciones de Nipype.

A continuación, se da un fragmento de código donde se describe como es el lenguaje y la sintaxis con la librería de Nipype. Como en todo código en Python, inicialmente se cargan las librerías principales como se muestra en las siguientes líneas de código.

```
1. from nipype import Workflow, Node, Function, MapNode
2. from os.path import join as opj
```

Nipype tiene muchas funciones propias que pueden ser llamadas en los nodos. Sin embargo, esta también permite usar otras funciones que no estén en las librerías. Por ejemplo, a continuación, se crean dos funciones que crean un script en lenguaje bash y lo ejecutan respectivamente.

```
1. def createScript():
2.     import os
3.     path_p=os.getcwd()
4.     out_path=path_p+'/Reference.sh'
5.     file = open(out_path, "w")
6.     file.write('echo "Running the bash script"' + os.linesep)
7.     file.close()
8.     return out_path

1. def runExternalScript(in_path):
2.     import subprocess
3.     result = subprocess.check_output('bash '+in_path, shell=True)
4.     print('\033[92m')
5.     print(result)
6.     print('\033[0m')
```

Con las funciones de Nipype o las funciones creadas es posible crear los nodos. Todos ellos cuentan con los mismos elementos; las entradas, las salidas, la función que usa y el nombre del nodo. Este último generalmente se usa para rotular los nodos en las gráficas automáticas de flujo de Nipype. En las siguientes líneas de código, se crean dos nodos con las funciones establecidas anteriormente.

```
1. Node1=Node(Function(input_names=[],
2.                     output_names=['out_path'],
3.                     function=createScript),
4.             name='create_script')

1. Node2=Node(Function(input_names=['in_path'],
2.                     output_names=[],
3.                     function=runExternalScript),
4.             name='run_external_script')
```

Nipype permite usar los nodos en diferentes partes del flujo o en diferentes flujos. Por lo tanto, es necesario crear un flujo de trabajo, darle una dirección y un nombre como se describe a continuación.

```
1. flow=Workflow(name='pipeline_flow')
2. flow.base_dir = '/content/workingdir'
```

Posteriormente, es necesario establecer todos los nodos al interior del flujo de trabajo y la conexión entre los nodos como se muestra a continuación.

```
1. flow.connect([(Node1, Node2, [('out_path', 'in_path')])])
```

Finalmente, el flujo puede ser ejecutado en su totalidad como se indica en la siguiente línea de código.

```
1. flow.run('MultiProc', plugin_args={'n_procs': 4})
```

Nipype crea de manera automática las carpetas de ejecución de cada nodo, en donde se almacena el reporte de ejecución de cada uno. El siguiente enlace GitHub cuenta con este ejemplo base el cual se puede ejecutar directamente sobre Colab. (https://github.com/LxMera/Pre-pipelines/blob/master/Flujo_de_ejemplo.ipynb)

6.1.2. Preprocesamiento

El preprocesamiento utilizado cuenta con una estructura principal como se muestra en la *Figura 32*, sin embargo, esta estructura se modificó a fin de contar con diferentes pipelines las cuales mostraremos a lo largo de este capítulo. Cada una de las funciones utilizadas en el preprocesamiento y el cálculo de las métricas de caracterización de la actividad neuronal se describen a continuación como nodos del flujo de trabajo definido en Nipype.

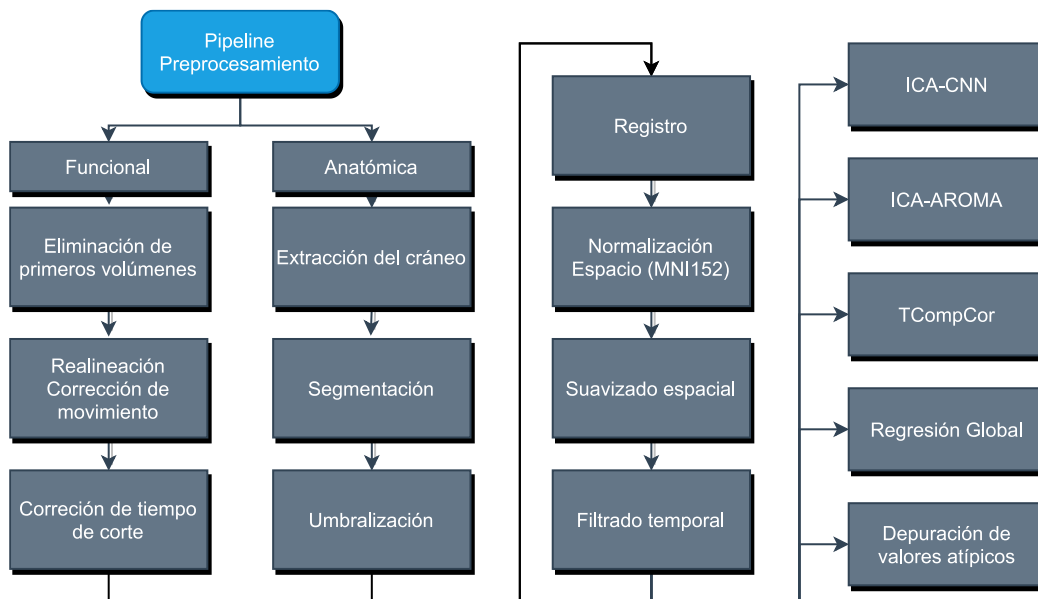


Figura 32 Diagrama principal del flujo de preprocesamiento utilizado para reducción de ruido.

El flujo de procesamiento se diseñó con uno nodo principal de iteración entre los diferentes sujetos de estudio y la sesiones por sujeto, pasando a un nodo secundario de selección de las imágenes funcionales y anatómicas de cada sujeto. Los nodos se titularon 'infosource' y 'selecfiles', además, estos funcionan con las estructuras de datos de imágenes cerebrales BIDS [79] como convencionalmente se suele usar en este tipo de datos.

A la imagen funcional original se le depuraron los primeros 4 volúmenes de tiempo para eliminar las inconsistencias presentes al principio del registro de los sujetos. El proceso se realizó mediante la integración de la función 'fslroi' de FSL a el nodo rotulado como

ExtractROI, generándose una imagen equivalente con el mismo formato nifti de la imagen original de entrada.

La corrección de movimiento se realizó mediante la función 'mcflir' de FSL implementada como el siguiente nodo en la parte funcional y aplicado como una transformación de cuerpo rígido con 12 parámetros de alineación, correspondientes a: tres parámetros de traslación, tres de rotación y los seis generados por las derivadas de estos 6 primeros. Cada volumen de la fMRI se alineó con el primer volumen del registro. El nodo se rotuló como 'mcflir2' y se diseñó para entregar la imagen funcional corregida, los parámetros de movimiento y la imagen media del movimiento, las cuales son utilizadas en nodos subsecuentes.

La corrección de tiempo de corte fue realizada a través de la función 'slicetimer' de FSL utilizando la adquisición intercalada correspondiente al registro de las imágenes MRI generando únicamente la imagen corregida a la salida de este nodo rotulado como 'slicetimer'.

A la imagen estructural se le realizó: la extracción del cráneo mediante la función BET, la segmentación de los diferentes tejidos (líquido cefalorraquídeo, materia blanca y materia gris) con la función 'FAST' y se generó un mapa de probabilidad de la materia blanca mediante la umbralización del tejido con la función 'Threshold', utilizando FSL en cada uno de los nodos los cuales se nombraron respectivamente como 'bet_anat', 'segmentation' y 'threshold'.

Posteriormente, estos resultados se combinaron con los datos funcionales, para generar la matriz de registro de la imagen funcional hacia la imagen anatómica del sujeto. El proceso se hizo como un pre-registro afín (12 grados de libertad) y una optimización basado en los límites de materia blanca [80], utilizando la función FLIRT en dos nodos llamados 'coreg_pre' y 'coreg_bbr' correspondientes a los mismos. La matriz de transformación obtenida se utilizó para generar una matriz de 'normalización' ajustando el espacio del sujeto al espacio estándar MNI, nuevamente a través de la función FLIRT de FSL y utilizando la plantilla 'MNI152_T1_1mm' en el nodo titulado 'Normalization'. Finalmente, la transformación se implementó en el nodo llamado 'applyward' con la función FLIRT sobre la imagen funcional alineada y con corrección de tiempo de corte. La *Figura 33* muestra con detalle las funciones descritas hasta el momento con todos los datos de entrada y salida generados en el proceso.

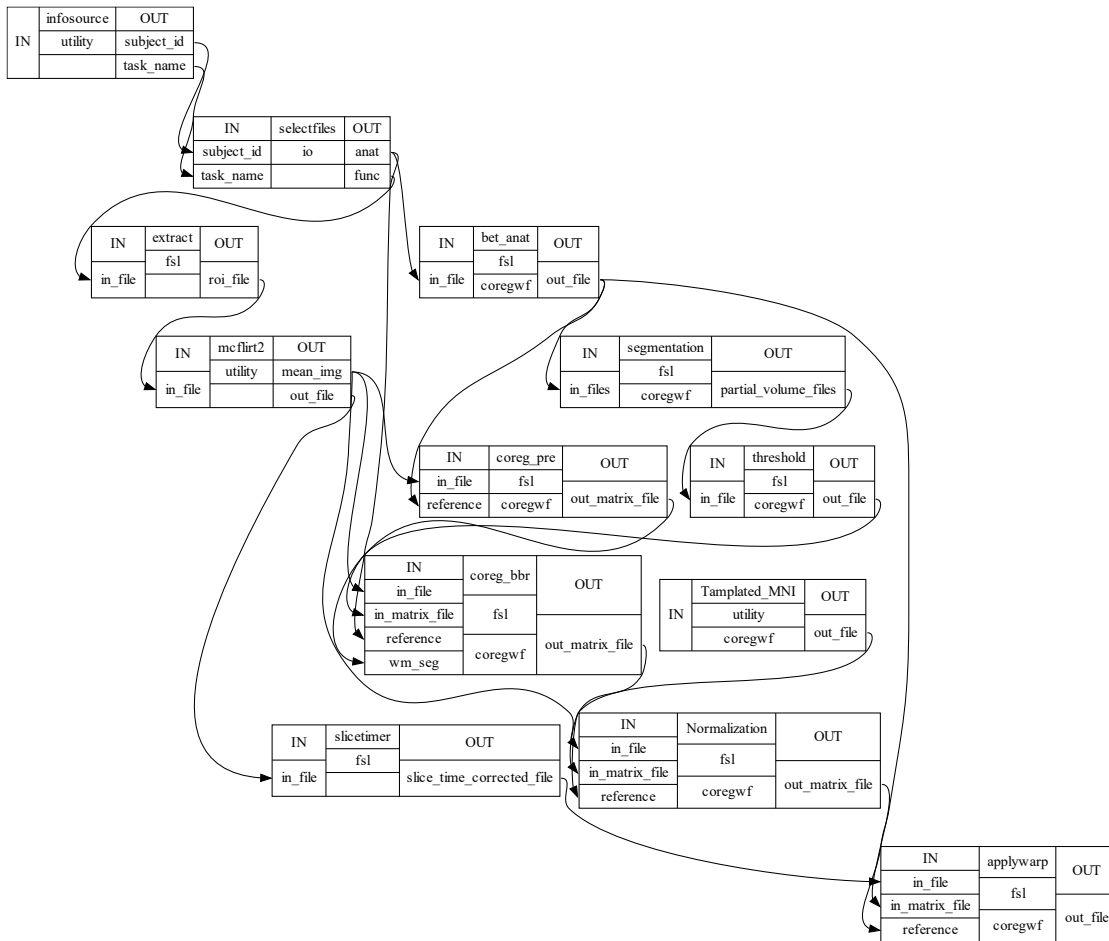


Figura 33 Flujo de del preprocesamiento hasta la normalización del sujeto con diferentes grados de libertad en la corrección de movimiento

Con la imagen registrada en el espacio MNI se realizó el suavizado espacial mediante un filtro Gaussiano de tamaños de filtro de 4. El filtro se implementó mediante la función 'smooth_img' de la librería Nilearn en el nodo titulado 'smooth', seguidamente, la imagen funcional fue filtrada con un filtro pasa altas de 0.01 Hz eliminando los movimientos lentos de la señal, aplicado como una función en python basada en la transformada de rápida de Fourier de la librería de Numpy, la cual se rotuló como 'filter_pass_high' (ver Figura 34).

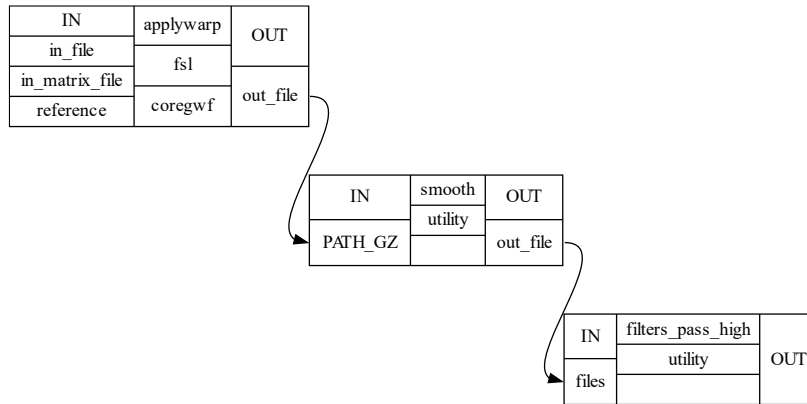


Figura 34 Flujo de del procesamiento de suavizado y filtrado de la imagen normalizada.

El pre-proceso descrito hasta este punto se utilizó sobre todos los esquemas de preprocesamiento. Los procesos de eliminación de artefactos que se presentan a continuación se implementaron en todas las combinaciones posibles sobre el resultado de esta primera parte.

6.1.3. Reducción de ruido

Se implementó la eliminación automática de artefactos mediante el enfoque de componentes independientes, usando la descomposición lineal exploratoria multivariada en componentes independientes (MELODIC) de FSL, donde se estimaron automáticamente el número de componentes independientes (Nodo 'Descomposition_ICA') reducidas mediante el método RPBC (ver [sección 5.1.8.4](#)) y se clasificaron a través la red neuronal convolucional de máxima precisión previamente entrenada (ver [sección 5.2.5.2](#)). Las componentes clasificadas como ruido se eliminaron del sujeto usando una regresión simple de mínimos cuadrados ordinarios de la función 'fsl_regfilt' ejecutada en el nodo 'Denoising_ICA'.

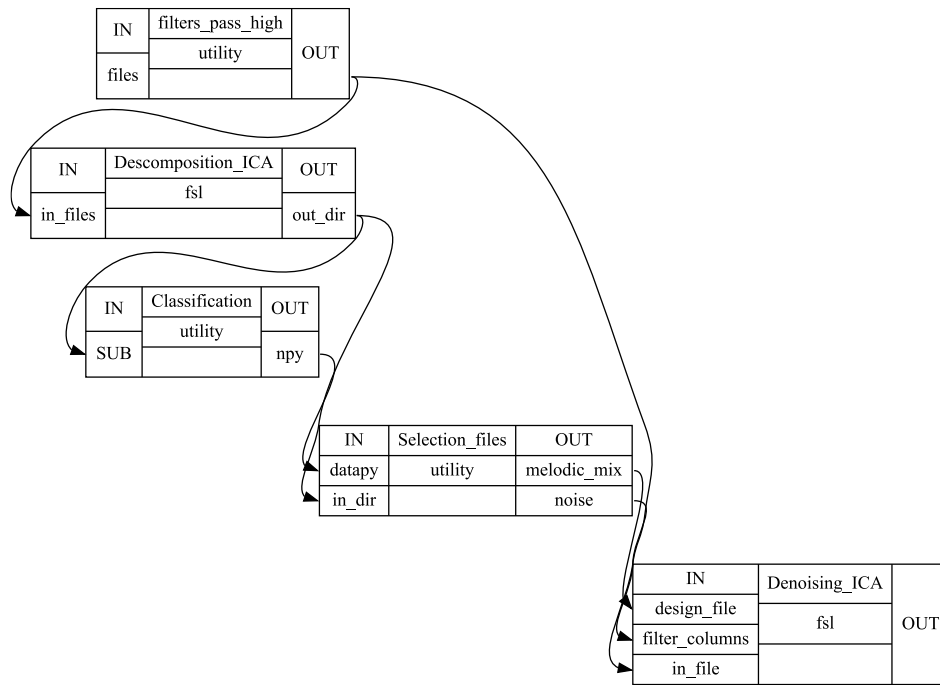


Figura 35 Eliminación automática de artefactos mediante RPBC en las componentes independientes.

Se utilizó la corrección por componentes principales de las regiones con desviación estándar temporal alta (tCompCor) para estimar 6 componentes [20] de ruido fisiológico en el nodo 'TCompCor', haciendo uso de la función TCompCor de FSL. Cada una de las componentes independientes se eliminaron como regresores molestos aplicados sobre la imagen funcional en un modelo lineal general a través de la función GLM en el nodo 'Regressor_CompCor2'.

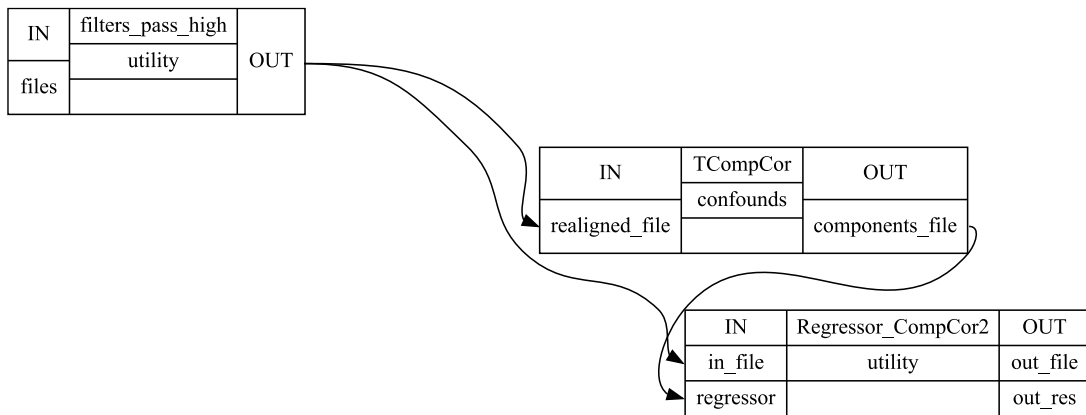


Figura 36 Flujo de procesamiento de TCompCor

La señal global de la imagen funcional se calculó como el promedio de todos los vóxeles correspondientes al volumen cerebral de un punto de tiempo, obteniéndose con la ayuda de las librerías Numpy y Nibabel. La señal global se generó en el nodo

'Global_signal' a la cual se le eliminó la tendencia temporal y se estandarizó para posteriormente eliminarse como un regresor molesto sobre la imagen funcional en un modelo lineal general mediante la función GLM de FSL en el nodo 'Regressor_Global2'.

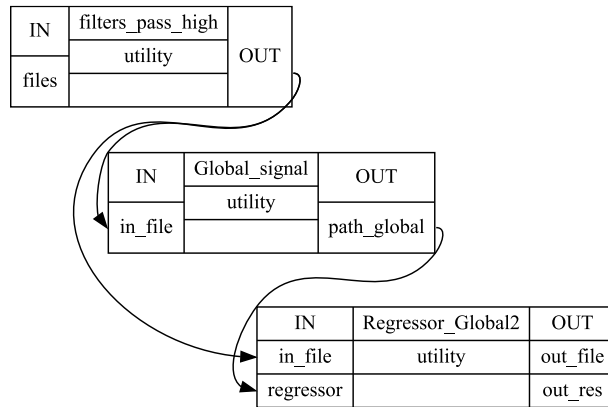


Figura 37 Flujo de procesamiento regresión de señal global.

Finalmente, se usaron las coordenadas afines y los parámetros de movimientos para estimar los valores atípicos generados por movimientos grandes en el registro de la imagen funcional y se eliminaron usando la selección de puntos temporales con las librerías de Nibabel y Numpy en el nodo 'scrub'.

6.1.4. Métricas

A las diferentes imágenes funcionales 'limpias' obtenidas en la sección anterior, se les calcularon las siguientes métricas de manera paralela, por lo tanto, en este apartado se describen los cálculos realizados sin entrar en detalle en el flujo posterior al preprocesamiento. Inicialmente, se calculó la homogeneidad regional con el coeficiente de concordancia de Kendall [81] para cada uno de los vóxeles usando vecindades de 7, 19 y 27 con la ayuda de la función '3dReHo' de software AFNI [82]. Seguidamente, con el uso de la función 'NiftiMapsMasker' de Nilearn, la imagen funcional se dividió en las series de tiempo de las regiones de interés en las regiones anatómicas establecidas por el atlas de etiquetado anatómico automatizado 3 (AAL3 de sus siglas en inglés *Automated anatomical labelling atlas 3*) [83]. A dichas series de tiempo se les calculó la amplitud de fluctuación de baja frecuencia y su correspondiente normalizada o fraccional (ecuación (1)) en las diferentes bandas espectrales. El cálculo se hizo considerando la serie de tiempo como una aproximación a una serie de Fourier así como se indica en la ecuación (2) [53].

Se calculó la conectividad funcional de todas las combinaciones posibles de las ROIs mediante el coeficiente de correlación de Pearson de las series de tiempo usando la función 'ConnectivityMeasure' de Nilearn, resultando en una matriz simétrica de 160x160 con valores de -1 a 1, indicando la fuerza la conectividad entre las regiones, donde 1 corresponde a una correlación perfecta, 0 correlación nula y -1 una correlación inversa. La matriz de conectividad funcional se estableció como la matriz de adyacencia para modelar la conectividad como un grafo ponderado limitando el

modelo al 30 por ciento de las conexiones con mayor peso. El grafo se generó con la librería 'networkx' a través de la función 'Graph' y partiendo de esta librería, se calcularon las siguientes métricas:

- El grado ponderado del grafo no dirigido, calculado como la suma de las conexiones con los nodos adyacentes mediante la función 'degree' para cada nodo.
- Con la función 'average_shortest_path_length' se estimó la ruta más corta promedio entre las combinaciones de todos los pares de nodos posibles, siguiendo la siguiente relación matemática:

$$a = \sum_{s,t \in V} \frac{d(s,t)}{n(n-1)} \quad (12)$$

donde V es el conjunto de todos los nodos en el grafo, $d(s,t)$ es la ruta más corta entre el nodo s y el nodo t .

- El coeficiente de agrupamiento por nodo c_u , se calculó mediante la función 'clustering' en un grafo ponderado no dirigido basado en el concepto de intensidad de subgrafos, definido como el promedio geométrico de los pesos del borde del subgrafo expresado por la ecuación (13).

$$c_u = \frac{1}{deg(u)(deg(u)-1)} \sum_{vw} (\hat{w}_{uv}\hat{w}_{uw}\hat{w}_{vw})^{1/3} \quad (13)$$

donde u es el nodo de interés, $deg(u)$ grado del nodo y \hat{w} el peso normalizado entre los correspondientes nodos u , v y w , es decir, el peso dividido entre el máximo peso presente en el grafo [84].

- El coeficiente de agrupamiento promedio para el grafo de tamaño n se calculó con el uso de la 'average_clustering' establecida por la ecuación (14) [84], [85].

$$c = \frac{1}{n} \sum_v c_v \quad (14)$$

En la *Figura 38* se muestran todas las métricas que se calcularon sobre los datos preprocesados, donde todas estas se calcularon sobre las regiones de interés derivadas del atlas anatómico AAL3 [83], a excepción de las métricas globales, las cuales arrojaron un valor puntual por volumen cerebral.

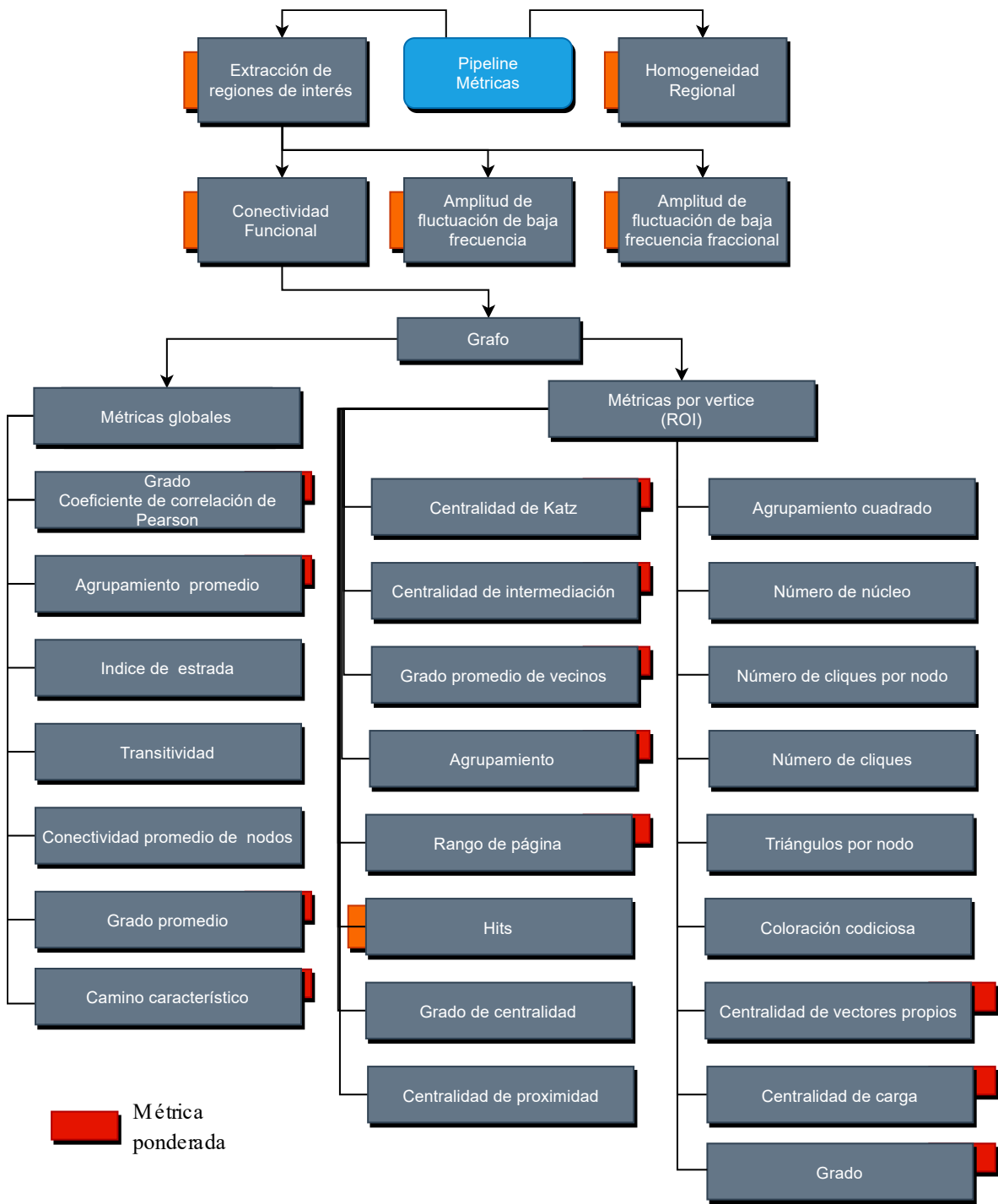


Figura 38 Diagrama principal del flujo de preprocesamiento utilizado para reducción de ruido.

6.2. Resultados

La combinación de los diferentes métodos de eliminación de artefactos dio como resultado un total 48 esquemas de pre-procesamiento (pipeline) como se indica en la **Tabla X**, siendo el pipeline 0 la imagen original, el pipeline 1 la imagen pre-procesada hasta el suavizado espacial y los pipelines del 2 al 48 los pre-procesos con el suavizado espacial más los indicados en dicha tabla.

En total se procesaron 17 sujetos con estudios longitudinales de 3 sesiones con un tiempo de separación de registro de aproximadamente 6 meses. En consecuencia, se generaron 144 volúmenes cerebrales por sujeto, es decir, 2448 volúmenes para todos los sujetos con tiempos promedio de generación de volúmenes de 123 segundos, para un tiempo de ejecución continuo de 83.64 horas.

El desarrollo de los scripts de los pipelines, hasta última versión, están contenidos en el siguiente repositorio GitHub (<https://github.com/LxMera/Pre-pipelines>).

Tabla X 48 Esquemas de procesamiento generados por la combinación de las diferentes técnicas de eliminación de artefactos

Pipeline	Procesos				
0	Puro				
1					
2	aroma agresivo				
3	aroma No agresivo				
4	ICA CNN				
5	aroma agresivo	ICA CNN			
6	aroma No agresivo	ICA CNN			
7	TcompCor				
8	aroma agresivo	TcompCor			
9	aroma No agresivo	TcompCor			
10	ICA CNN	TcompCor			
11	aroma agresivo	ICA CNN	TcompCor		
12	aroma No agresivo	ICA CNN	TcompCor		
13	RG				
14	aroma agresivo	RG			
15	aroma No agresivo	RG			
16	ICA CNN	RG			
17	aroma agresivo	ICA CNN	RG		
18	aroma No agresivo	ICA CNN	RG		
19	TcompCor	RG			
20	aroma agresivo	TcompCor	RG		
21	aroma No agresivo	TcompCor	RG		
22	ICA CNN	TcompCor	RG		
23	aroma agresivo	ICA CNN	TcompCor	RG	
24	aroma No agresivo	ICA CNN	TcompCor	RG	
25	Depuración				
26	aroma agresivo	Depuración			
27	aroma No agresivo	Depuración			
28	ICA CNN	Depuración			

29	aroma agresivo	ICA CNN	Depuración		
30	aroma No agresivo	ICA CNN	Depuración		
31	TcompCor	Depuración			
32	aroma agresivo	TcompCor	Depuración		
33	aroma No agresivo	TcompCor	Depuración		
34	ICA CNN	TcompCor	Depuración		
35	aroma agresivo	ICA CNN	TcompCor	Depuración	
36	aroma No agresivo	ICA CNN	TcompCor	Depuración	
37	RG	Depuración			
38	aroma agresivo	RG	Depuración		
39	aroma No agresivo	RG	Depuración		
40	ICA CNN	RG	Depuración		
41	aroma agresivo	ICA CNN	RG	Depuración	
42	aroma No agresivo	ICA CNN	RG	Depuración	
43	TcompCor	RG	Depuración		
44	aroma agresivo	TcompCor	RG	Depuración	
45	aroma No agresivo	TcompCor	RG	Depuración	
46	ICA CNN	TcompCor	RG	Depuración	
47	aroma agresivo	ICA CNN	TcompCor	RG	Depuración
48	aroma No agresivo	ICA CNN	TcompCor	RG	Depuración

Puro*: Datos funcionales del sujeto sin ningún preproceso.

Nota: Todos los pipelines, a excepción del 0, cuentan con el preproceso base hasta el filtrado temporal descrito en este capítulo.

6.3. Discusión

Muchas de las librerías utilizadas se implementan bajo diferentes lenguajes de programación, como es el caso de AFNI, software desarrollado inicialmente en C y hoy en día implementado con secuencias de comando Shell o Bash dependiendo del sistema operativo [65]. De igual manera, FSL ha sido desarrollado como software con secuencia de comandos y actualmente también cuenta con una interfaz gráfica que ha permitido una implementación más sencilla [68], [69], sin embargo, la ejecución de estos en mismo entorno es una tarea que incluso a la fecha puede presentar inconvenientes. En este sentido, la integración de estas herramientas con la librería de Nipype [78] muestra la versatilidad de la herramienta, ya que, aunque esta se basa en el lenguaje de programación Python, las múltiples librerías permitieron integrar todos los métodos de procesamiento de neuroimágenes, incluso con diferentes lenguajes de programación.

En contraste, pese a que existen muchas herramientas ampliamente usadas como DPARSAF, FSL o CONN [68], [69], [72], [86], ninguna de ellas ha incluido desarrollos usados en el aprendizaje profundo como son las redes neuronales convolucionales. Asimismo, las herramientas están limitadas, ya que las mismas fueron pensadas para el fácil uso de un usuario final, restringiendo la adición o incorporación de nuevos desarrollos y, por lo tanto, limitando el número de pipelines a los establecidos por el software. Cabe resaltar, que, aunque en esta sección se describieron en total 48 pipelines, la implementación mediante Nipype permite modificar parámetros, adicionar o eliminar elementos para generar nuevos pipelines, característica que puede ser de utilidad para estudios posteriores.

6.4. Conclusión

Se logró unificar diferentes softwares de neuroimágenes, con el fin del integrar las diferentes herramientas de preprocesamiento y del cálculo de las métricas, en un solo interfaz, lo cual permite encontrar características de los sujetos de manera ininterrumpida, reflejándose en una administración más eficiente del proceso y, por lo tanto, en una reducción del tiempo de ejecución de las máquinas. El método mostró su eficiencia, generando 2448 volúmenes de las diferentes métricas para los 17 sujetos con 3 registros temporales para el estudio en cuestión.

7. Determinar el impacto sobre la reproducibilidad de las métricas de conectoma bajo los diferentes métodos de limpieza en la señal BOLD

La resonancia magnética funcional proporciona métricas para comprender las diferentes fisiopatologías que afectan el sistema nervioso central y como se ha mencionado anteriormente, el paradigma en estado de reposo es una de las técnicas prometedoras en exploración funcional, siendo métricas muy comunes en este paradigma la amplitud de fluctuación de baja frecuencia, la amplitud de fluctuación de baja frecuencia fraccional, la homogeneidad regional de la actividad neuronal, la conectividad funcional y algunas métricas derivadas de la teoría de grafos como el grado, la centralidad ponderada y agrupamiento [87], [88].

Sin embargo, para que estas métricas tengan validez en el ámbito clínico y puedan ser estudiadas como biomarcadores en las diferentes patologías, respuestas a tratamientos específicos y/o indicadores del progreso de una enfermedad, estas deben ser altamente reproducibles, es decir, las diferentes métricas deben presentar una baja variabilidad entre sujetos y un alto acuerdo entre mediciones [89], además, estudios anteriores han mostrado que la confiabilidad varía mucho entre construcciones siendo difícil crear generalidades en el estudio de las neurociencias [90]. Por lo tanto, en esta sección nos centramos estudiar la reproducibilidad bajo el efecto de 48 esquemas de procesamiento sobre imágenes rs-fMRI, donde se incluyen tanto las métricas más comunes en rs-fMRI como los diferentes índices derivados de la teoría de grafos que se describieron en los pipelines de la sección anterior.

7.1. Metodología

7.1.1. Sujetos

Se utilizaron las imágenes estructurales y funcionales del proyecto 'Datos longitudinales de test-retest de neuroimágenes en adultos jóvenes y sanos del suroeste de china'. Las imágenes se obtuvieron mediante un resonador SIEMENS MAGNETOM TrioTim syngo MR B17, con un campo magnético de 3T. La imagen estructural se registró con una secuencia de eco de gradiente rápido, con un tiempo de repetición de 1.9ms, tiempo de eco de 2.52ms, tiempo de inversión de 900 ms, tamaño de matriz de 256x256 con 176 cortes a un grosor de 1.0mm y tamaño de vóxel de 1x1x1mm³. La imagen funcional se obtuvo en estado de reposo en una exploración de 8 minutos para un total de 242 volúmenes de tiempo, con una secuencia de imagen de gradiente eco-planar, 32 cortes axiales, tiempo de repetición de 2000 ms, tiempo de eco de 30 ms y tamaño de vóxel de 3.4x3.4x3mm³ [49].

7.1.2. Preprocesamiento

Todas las imágenes de rs-fMRI se sometieron a la eliminación de los primeros 4 volúmenes de tiempo, la realineación o corrección de movimiento con 12 parámetros de cuerpo rígido, la corrección de tiempo de corte, el registro de imagen funcional a la imagen anatómica, la normalización al espacio MNI, el suavizado espacial, el filtrado temporal y a partir de este preproceso las imágenes fueron sometidas a la combinación de los diferentes técnicas de eliminación de artefactos como se mostró en la sección **6.1.3.** generándose los 48 pipelines mostrados en la **Tabla X.**

7.1.3. Métricas

Los 49 resultados, es decir, las imágenes puras y las 48 imágenes generados por los esquemas de preprocesamiento, se parcelaron en 170 regiones definidas por el atlas AAL3, sin embargo, debido a la baja resolución de las rs-fMRI las regiones más pequeñas fueron eliminadas. En las regiones restantes se calcularon las siguientes métricas:

- Homogeneidad regional con 7, 19 y 27 vecinos.
- Amplitud de fluctuación de baja frecuencia y su parte fraccional (ALFF y fALFF) en bandas espectrales 2, 3, 4, y 5 definidas en los rangos frecuenciales de 0.25-0.198, 0.198-0.073, 0.073-0.027 y 0.027-0.01 Hz respectivamente.
- Conectividad funcional.
- Agrupamiento promedio.
- Camino característico promedio.
- Grado promedio.
- Agrupamiento por región (nodo).
- Grado por región (nodo).

Adicionalmente, de la teoría de grafos se estimaron 17 métricas más que se describen en los **Anexos** junto con los resultados, generados con la metodología descrito en esta sección.

7.1.4. Cálculo de la fiabilidad mediante ICC

La confiabilidad se estimó para todas las métricas en las parcelaciones del atlas AAL3 con el coeficiente de correlación intraclases, el cual se calcula típicamente como la relación entre la varianza entre sujetos y la varianza total [91] como se describe en la ecuación (8). Sin embargo, esta es una definición general y el ICC se puede calcular de diferentes maneras y la elección del mismo se rige por tres consideraciones principales: el modelo, el tipo y la definición del coeficiente [89], que, para este caso particular, el estudio se basa en estimar el grado de consistencia entre las mediciones de 3 tiempos diferentes, lo cual supone un modelo de dos factores con efectos mixtos, siendo los sujetos las selecciones aleatorias y las medidas de tiempo los efectos fijos, es decir, el comportamiento se describe matemáticamente por las siguientes expresiones:

$$y_{ijz} = \alpha_i + \beta_j + \varepsilon_{ijz} \quad (15)$$

$$i = 1,2,3,\dots \dots m \quad j = 1,2,3,\dots \dots l \quad z = 1,2,3,\dots \dots p \quad (16)$$

Donde y representa el valor de la métrica, α el factor que contiene la variabilidad intra-sujetos y de efecto fijo (medidas repetidas), β el factor que contiene la variabilidad inter-sujetos además de ser el factor de efectos aleatorios, ε el error aleatorio, i el i -

ésimo nivel del factor α , j el j -ésimo nivel del factor β , z la z -ésima observación, m el número total de sujetos, l el número de medidas repetidas y p el número de observaciones.

Partiendo de este modelo, el ICC tiene dos definiciones conocidas como de acuerdo absoluto y de consistencia. El primero de ellos se usa en la búsqueda de diferencias de asignación de puntajes para un mismo sujeto entre evaluadores o si un instrumento mide exactamente los mismos valores en un mismo sujeto, mientras que el segundo de ellos hace refiere a si los puntajes de los evaluadores para el mismo grupo de sujetos se correlacionan aditivamente. En este caso tenemos un modelo de efectos mixtos de dos factores, en acuerdo absoluto sin replicas [89], [91], [92], el cual se rige por un ICC definido por la siguiente ecuación [93]:

$$ICC = \frac{MSR - MSE}{MSR + (k - 1) \cdot MSE + \frac{k}{n}(MSC - MSE)} \quad (17)$$

Donde MSR es la media de cuadrados de las filas o inter-sujetos, MSC es la media de cuadrados de las columnas o intra-sujetos, MSE la media de cuadrados del error, n el número de sujetos y k el número de medidas de tiempo. MSR y MSC se calcularon a partir de los matrices de datos sujetos-mediciones y MSE se calculó como:

$$MSE = \frac{SStotal - MSR \cdot (n - 1) - MSC \cdot (k - 1)}{(n - 1) \cdot (k - 1)} \quad (18)$$

$$SStotal = \sigma_t^2 \cdot (k \cdot n - 1) \quad (19)$$

Donde $SStotal$ es la suma de cuadrados total y σ_t^2 la varianza total. Adicionalmente, el ICC se calculó utilizando las pruebas de permutaciones o prueba exacta ya que es un enfoque no paramétrico basado en datos [94], [95] y este a su vez se combinó usando el método de Monte Carlo para reducir el número de permutaciones [96].

7.2. Resultados

La resolución de las imágenes y el tamaño reducido de algunas regiones tuvo como resultado una parcelación con un número menor de las regiones de interés respecto al número total de regiones originales establecidas por el atlas anatómico AAL3. En la tabla **Tabla XI** se muestran las abreviaciones de las 141 regiones obtenidas, además, estas abreviaciones son utilizadas en todas las figuras con métricas por regiones.

Tabla XI Regiones anatómicas usadas en la parcelación de los volúmenes cerebrales y las abreviaciones utilizadas en las gráficas de los resultados

Indices*	Abreviación	Descripción Anatómica	Indices*	Abreviación	Descripción Anatómica
1	Precentral_L	Precentral circunvolución o giro precentral	73	Paracentral_Lobule_L	Lóbulo paracentral
2	Precentral_R		74	Paracentral_Lobule_R	
3	Frontal_Sup_2_L	Giro frontal superior, dorsolateral	75	Caudate_L	Núcleo caudado
4	Frontal_Sup_2_R		76	Caudate_R	
5	Frontal_Mid_2_L	Giro frontal medio	77	Putamen_L	Núcleo lenticular, Putamen
6	Frontal_Mid_2_R		78	Putamen_R	
7	Frontal_Inf_Oper_L		79	Pallidum_L	

8	Frontal_Inf_Oper_R	Giro frontal inferior, parte opercular	80	Pallidum_R	Núcleo lenticular, pálido
9	Frontal_Inf_Tri_L	Giro frontal inferior, parte triangular	83	Heschl_L	La circunvolución de Heschl
10	Frontal_Inf_Tri_R		84	Heschl_R	
11	Frontal_Inf_Orb_2_L	Giro frontal inferior, parte orbital	85	Temporal_Sup_L	Giro temporal superior
12	Frontal_Inf_Orb_2_R		86	Temporal_Sup_R	
13	Rolandic_Oper_L	Opérculo rolandico	87	Temporal_Pole_Sup_L	Polo temporal: giro temporal superior
14	Rolandic_Oper_R		88	Temporal_Pole_Sup_R	
15	Supp_Motor_Area_L	Área motora suplementaria	89	Temporal_Mid_L	Giro temporal medio
16	Supp_Motor_Area_R		90	Temporal_Mid_R	
17	Olfactory_L	Corteza olfatoria	91	Temporal_Pole_Mid_L	Polo temporal: giro temporal medio
18	Olfactory_R		92	Temporal_Pole_Mid_R	
19	Frontal_Sup_Medial_L	Giro frontal superior, medial	93	Temporal_Inf_L	Giro temporal inferior
20	Frontal_Sup_Medial_R		94	Temporal_Inf_R	
21	Frontal_Med_Orb_L	Giro frontal superior, orbital medial	113	Vermis_1_2	Lóbulo I, II de vermis
22	Frontal_Med_Orb_R		114	Vermis_3	Lóbulo III de vermis
23	Rectus_L	Giro recto	115	Vermis_4_5	Lóbulo IV, V de vermis
24	Rectus_R		116	Vermis_6	Lóbulo VI de vermis
25	OFCmed_L	Giro orbital medial	117	Vermis_7	Lóbulo VII de vermis
26	OFCmed_R		118	Vermis_8	Lóbulo VIII de vermis
27	OFCant_L	Giro orbitario anterior	119	Vermis_9	Lóbulo IX de vermis
28	OFCant_R		120	Vermis_10	Lóbulo X de vermis
29	OFCpost_L	Giro orbital posterior	121	Thal_AV_L	Tálamo, núcleo anteroventral
30	OFCpost_R		122	Thal_AV_R	
31	OFClat_L	Giro orbital lateral	123	Thal_LP_L	Lateral posterior
32	OFClat_R		124	Thal_LP_R	
33	Insula_L	Ínsula	125	Thal_VA_L	Ventral anterior
34	Insula_R		126	Thal_VA_R	
37	Cingulate_Mid_L	Circunvolución media cingulada y paracingulada	127	Thal_VL_L	Lateral ventral
38	Cingulate_Mid_R		128	Thal_VL_R	
39	Cingulate_Post_L	Giro cingulado posterior	129	Thal_VPL_L	Posterolateral ventral
40	Cingulate_Post_R		130	Thal_VPL_R	
41	Hippocampus_L	Hipocampo	131	Thal_IL_L	Intralaminar
42	Hippocampus_R		132	Thal_IL_R	
43	ParaHippocampal_L	Giro parahippocampal	135	Thal_MDm_L	Mediodorsal medial magnocelular
44	ParaHippocampal_R		136	Thal_MDm_R	
45	Amygdala_L	Amígdala	137	Thal_MDI_L	Mediodorsal lateral parvocelular
46	Amygdala_R		138	Thal_MDI_R	
47	Calcarine_L		139	Thal_LGN_L	Geniculado lateral

48	Calcarine_R	Fisura de calcarina y corteza circundante	140	Thal_LGN_R	
49	Cuneus_L	Cuneo	141	Thal_MGN_L	Geniculado medial
50	Cuneus_R		142	Thal_MGN_R	
51	Lingual_L	Giro lingual	147	Thal_PuA_L	Pulvinar lateral
52	Lingual_R		148	Thal_PuA_R	
53	Occipital_Sup_L	Giro occipital superior	145	Thal_PuM_L	Pulvinar medial
54	Occipital_Sup_R		146	Thal_PuM_R	
55	Occipital_Mid_L	Giro occipital medio	149	Thal_PuL_L	Pulvinar inferior
56	Occipital_Mid_R		150	Thal_PuL_R	
57	Occipital_Inf_L	Giro occipital inferior	143	Thal_PuI_L	Pulvinar anterior
58	Occipital_Inf_R		144	Thal_PuI_R	
59	Fusiform_L	Giro fusiforme	151	ACC_sub_L	Corteza cingulada anterior, subgenual
60	Fusiform_R		152	ACC_sub_R	
61	Postcentral_L	Giro poscentral	153	ACC_pre_L	Corteza cingulada anterior, pregenual
62	Postcentral_R		154	ACC_pre_R	
63	Parietal_Sup_L	Giro parietal superior	155	ACC_sup_L	Corteza cingulada anterior, supracallosal
64	Parietal_Sup_R		156	ACC_sup_R	
65	Parietal_Inf_L	Circunferencia parietal inferior, excluyendo los giros supramarginal y angular	157	N_Acc_L	Núcleo accumbens
66	Parietal_Inf_R		158	N_Acc_R	
67	SupraMarginal_L	Giro supramarginal	161	SN_pc_L	Sustancia negra, pars compacta
68	SupraMarginal_R		162	SN_pc_R	
69	Angular_L	Giro angular	163	SN_pr_L	Sustancia negra, pars reticulata
70	Angular_R		164	SN_pr_R	
71	Precuneus_L	Precuneo	165	Red_N_L	Núcleo rojo
72	Precuneus_R		166	Red_N_R	
			169	Raphe_D	Núcleo del Raphe, dorsal

Nota: Las abreviaciones terminadas en R y L corresponden a las descripciones anatómicas descritas en la tabla, pero de los hemisferios derecho e izquierdo respectivamente.

*los índices mostrados no corresponden al orden, si no a los índices seleccionada por AAL3.

Se analizó la confiabilidad para las diferentes métricas en función de los 48 esquemas de procesamiento y de las 141 regiones parceladas con el atlas anatómico AAL3, no obstante, debido a la gran cantidad de datos los resultados presentados a continuación se resumen en los 6 esquemas de procesamiento de mayor reproducibilidad por métrica. Cabe aclarar, que los índices numéricos de las figuras corresponden a los indicados en las **Tabla X**. Además, todos los resultados obtenidos en este capítulo están consignados en el apéndice de anexos.

7.2.1. ReHo

La Figura 39 muestra el coeficiente de correlación intraclass de la homogeneidad regional para vecindades de 7, 19 y 27 respectivamente. En las figuras se exponen los valores mayores a 0.4 debido a que valores por debajo de este son considerados como regulares o leves. Además, los valores entre 0.4 a 0.6, 0.6 a 0.8 y de 0.8 a 1.0 son tomados como moderados, sustanciales y casi perfectos respectivamente [97]. Es de aclarar que las regiones de interés están organizadas por el promedio de los valores de ICC entre pipelines de menor a mayor.

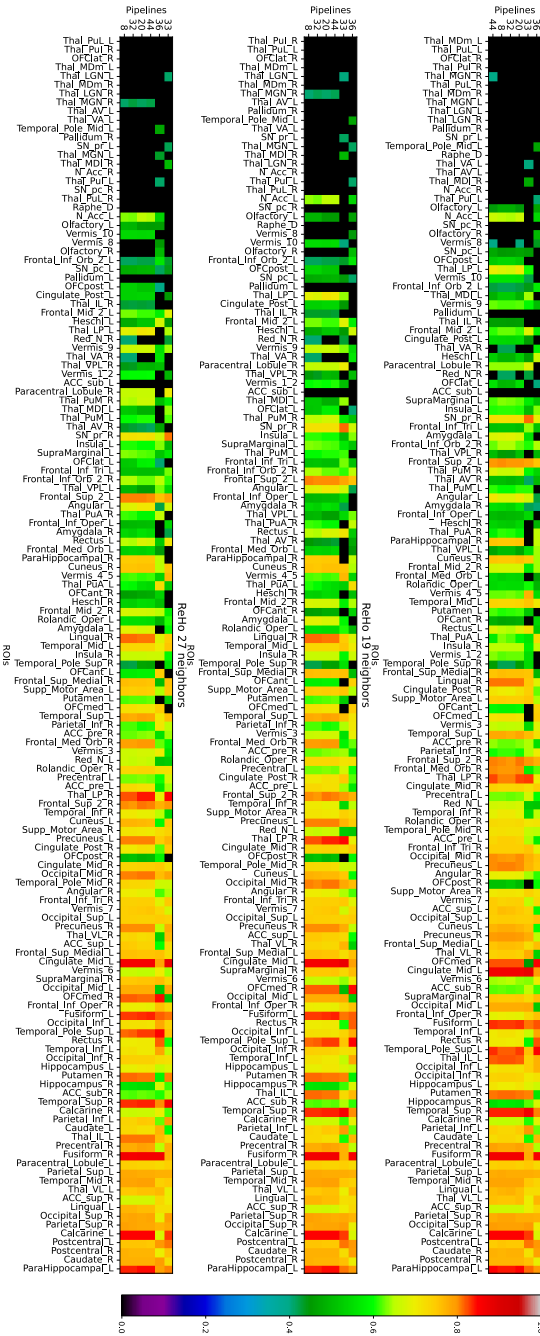


Figura 39 ICC sobre la métrica de ReHo con número de vecinos 7, 19 y 27 en las diferentes regiones para los 6 pipelines de mayor reproducibilidad.

En la *Figura 40* nuevamente, se muestra la reproducibilidad para la homogeneidad regional. Sin embargo, en este caso se usó el atlas anatómico msdl [98], [99] para obtener la red modo defecto o red neuronal por defecto. Análogo al caso anterior, la métrica se validó para las vecindades de 7, 19 y 27, donde se logra ver un comportamiento similar entre las tres métricas. Además, es claro que la reproducibilidad alcanzó valores por encima de moderado para la mayoría de pipelines, e incluso hay valores que están cerca de casi perfecto.

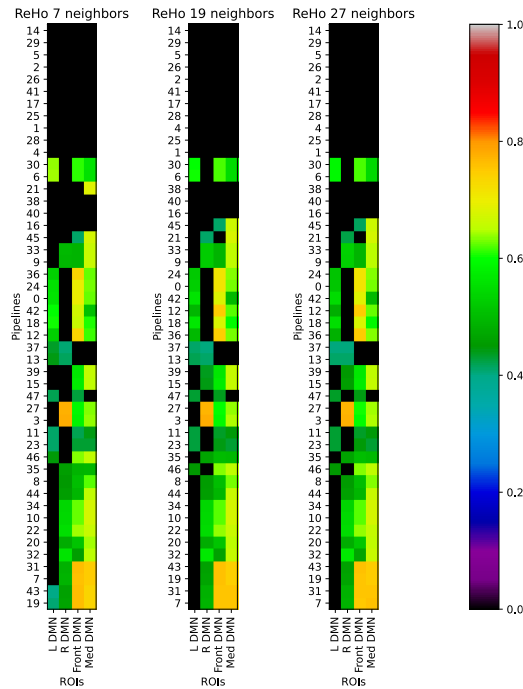


Figura 40 ICC sobre la métrica de ReHo con número de vecinos 7, 19 y 27 para todos los pipelines en la red modo defecto.

7.2.2. ALFF y fALFF

La *Figura 41* muestra el ICC para las métricas de ALFF y fALFF para las bandas 2 y 3, las cuales están organizadas de la misma manera que las métricas anteriores, es decir, están de mayor a menor entre regiones y se muestran los 6 pipelines de mayor reproducibilidad.



Figura 41 ICC sobre las métricas de ALFF y fALFF en las bandas 2 y 3 en las diferentes regiones para los 6 pipelines de mayor reproducibilidad.

En la *Figura 42* se muestra el ICC para las métricas de ALFF y fALFF en las bandas 4 y 5, organizadas de la misma manera que las métricas anteriores, organizadas de mayor a menor entre regiones y se muestran los 6 pipelines de mayor reproducibilidad.

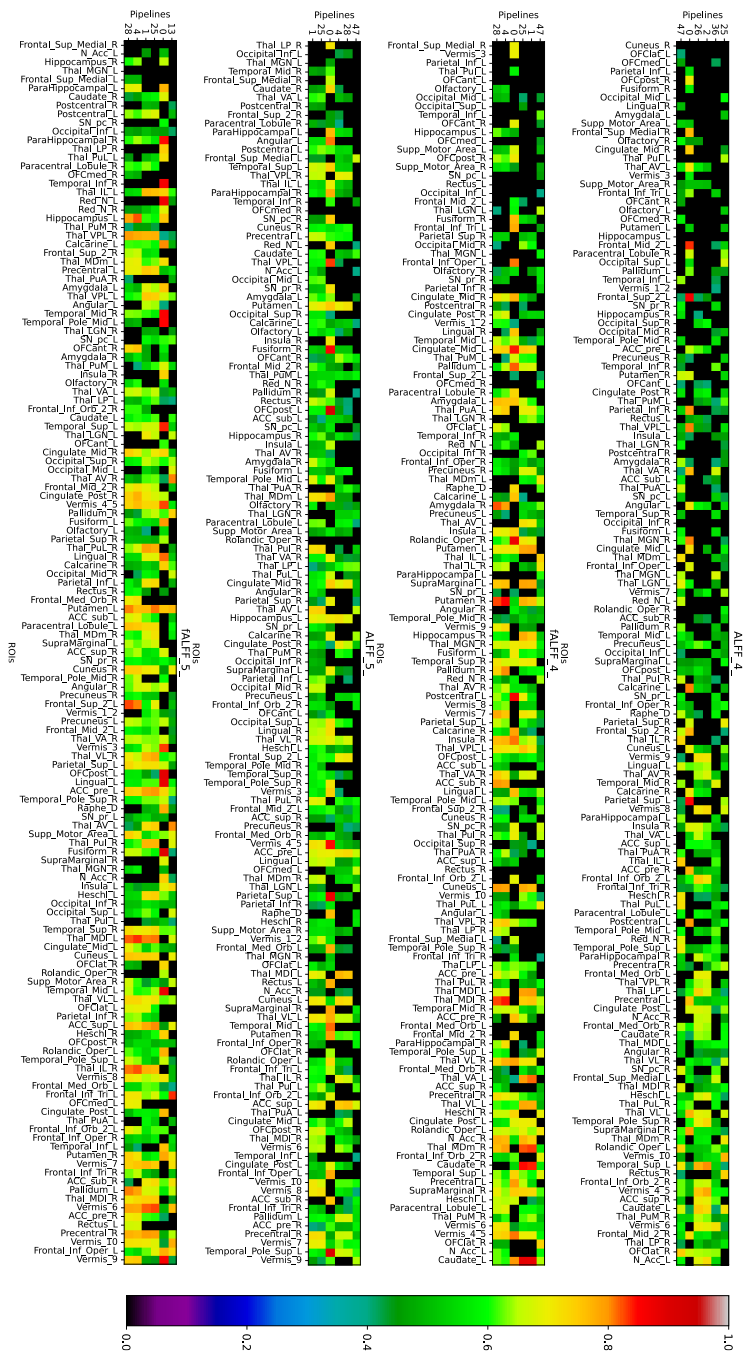


Figura 42 ICC sobre las métricas de ALFF y fALFF en las bandas 4 y 5 en las diferentes regiones para los 6 pipelines de mayor reproducibilidad.

Es de resaltar que los resultados entre las métricas ALFF y fALFF no tuvieron una diferencia aparente, además, los ICC entre bandas se mantuvieron con valores moderados, a excepción de fALFF en la banda 3, el cual contó con valores sustanciales e incluso algunos casi perfectos, sin embargo, también hubo un aumento de las regiones con valores regulares y leves respecto a la banda 2, que, en ese caso, conservó la mayoría de las regiones por encima de moderado.

7.2.3. Conectividad funcional

En las *Figura 43* se muestra el ICC de todas las posibles combinaciones de las regiones para el pipeline 3, el cual presentó el promedio más alto de valores de ICC. Para esta métrica, los pipelines que mejores resultados arrojaron fueron el 3, 27, 2, 26, 13 y 37.

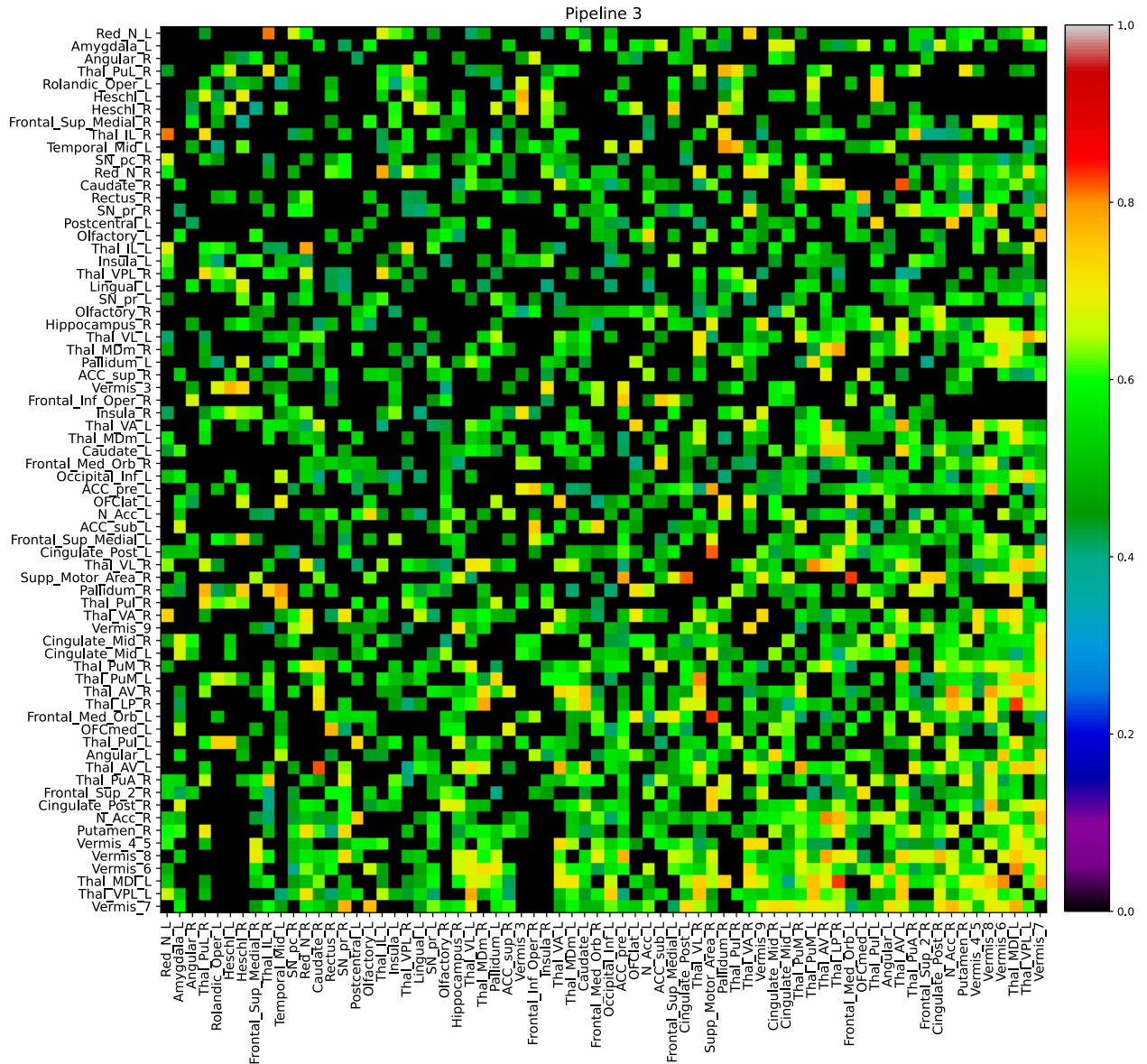


Figura 43 ICC sobre la métrica de conectividad funcional para el pipeline de mayor reproducibilidad.

Análogo al caso anterior, La *Figura 44* muestra el pipeline de mayor reproducibilidad sobre la métrica de conectividad funcional. Sin embargo, para este caso se usó el atlas msdl a fin de obtener la red modo defecto. En este caso, los pipelines de mayor reproducibilidad fueron el 28, 4, 16, 40, 3 y 27, en ese respectivo orden decreciente. Además, es de resaltar que la red modo defecto contó con las regiones de mayor reproducibilidad. De hecho, la parte derecha de esta (R DMN) fue la de mayor reproducibilidad.

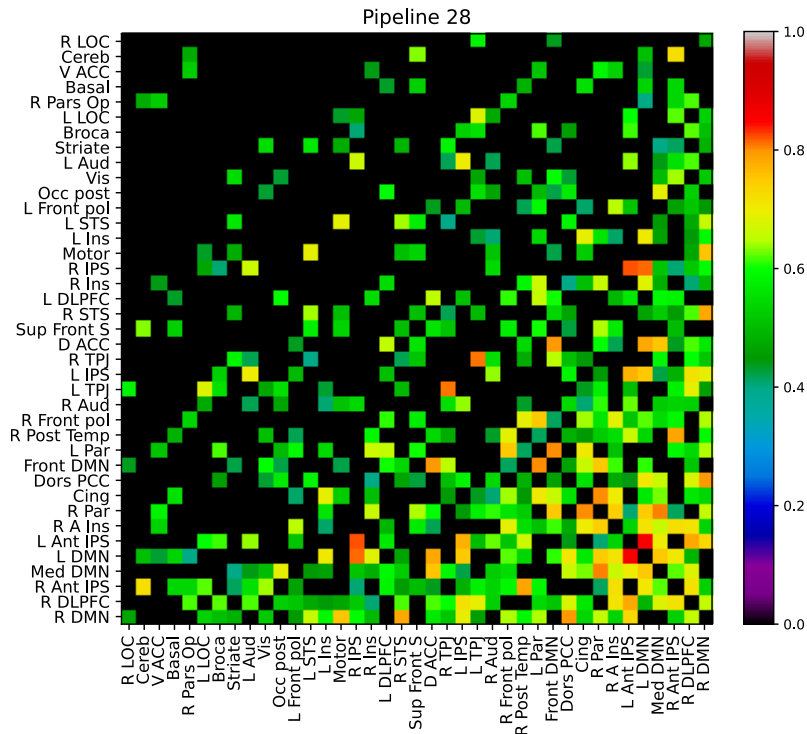


Figura 44 ICC sobre la métrica de conectividad funcional para el pipeline de mayor reproducibilidad en el atlas msdl.

7.2.4. Grafos globales

Las *Figura 45, 46 y 74* muestran el promedio y la distribución de los resultados de las métricas de agrupamiento, camino característico y grado respectivamente, en función de los pipelines. Cabe resaltar que las métricas se derivan de la conectividad funcional en el modelo de grafos.

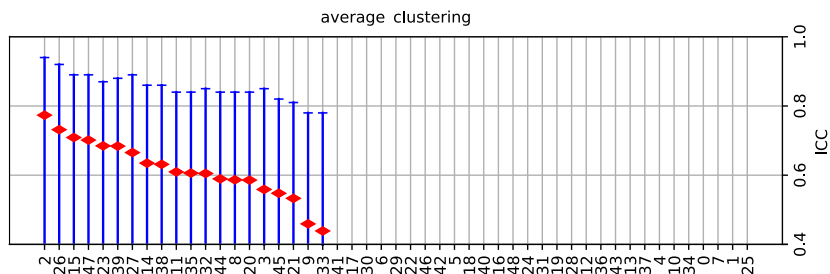


Figura 45 ICC sobre la métrica de agrupamiento promedio en los 48 pipelines diferentes.

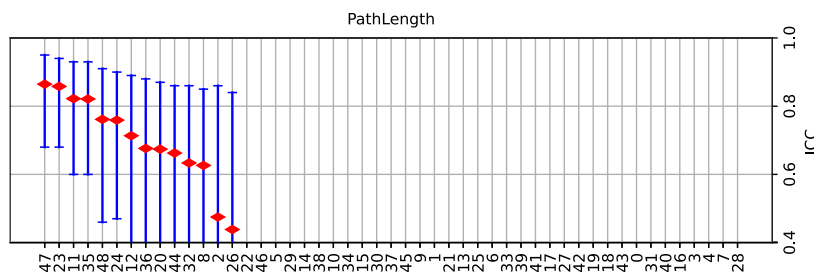


Figura 46 ICC sobre la métrica de camino característico en los 48 pipelines diferentes.

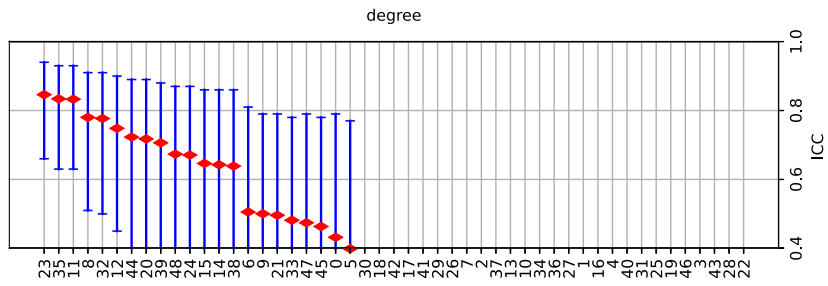


Figura 47 ICC sobre la métrica de grado promedio en los 48 pipelines diferentes.

7.2.5. Grafos por nodos

Nuevamente la *Figura 48* muestra las ICC para las métricas de agrupamiento y grado, pero en las diferentes regiones de interés. En total se calcularon 18 métricas por regiones derivadas del modelo de grafo, sin embargo, en este apartado se muestran estas dos ya que son las métricas más reportadas en neurociencias. Como se mencionó anteriormente, todas estas métricas están contenidas en el apartado de anexos.

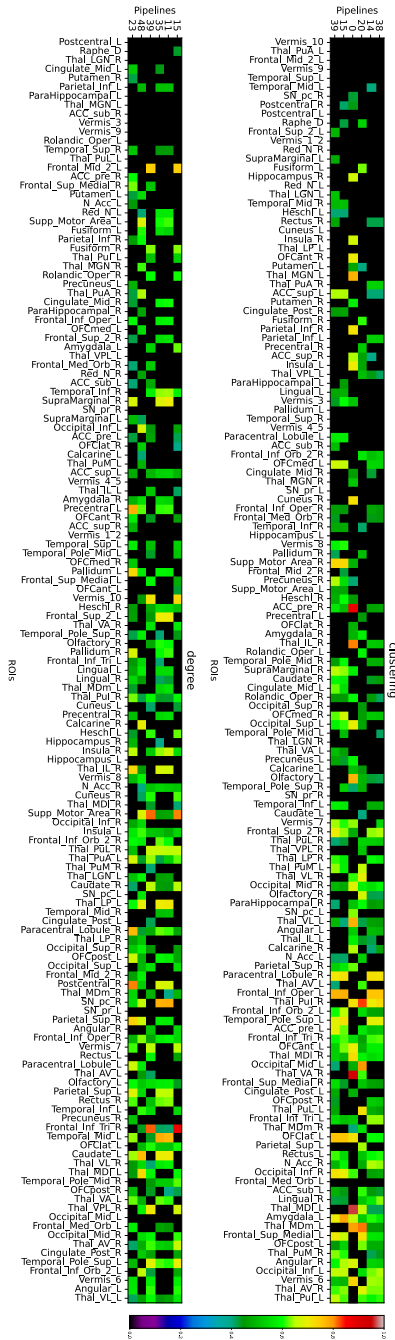


Figura 48 ICC sobre las métricas de agrupamiento y grado en las diferentes regiones para los 6 pipelines de mayor reproducibilidad.

La **Tabla XII** muestra los pipelines de mayor ICC, es decir, los de mayor reproducibilidad para cada una de las métricas consideradas en esta sección. Además, en esta se muestra los procesos de eliminación de artefactos que se incluyeron en cada uno de los pipelines junto con las tres regiones de mayor reproducibilidad para los mismos.

Tabla XII Pipelines de mayor reproducibilidad por métrica y regiones asociadas

Métricas	Procesos							Regiones parceladas con AAL3 de mayor ICC		
	No. Pipeline	aroma agresivo	aroma_No agresivo	ICA_CNN	TcompCor	RG	Depuración	Región de mayor ICC	Segunda región de mayor ICC	Tercera región de Mayor ICC
ReHo (7 vecinos)	44	X			X	X	X	ParaHippocampal_L	Postcentral_R	Caudate_R
ReHo (19 vecinos)	8	X			X			ParaHippocampal_L	Postcentral_R	Caudate_R
ReHo (27 vecinos)	8	X			X			ParaHippocampal_L	Caudate_R	Postcentral_R
ALFF (banda 2)	47	X		X	X	X	X	Caudate_L	Cingulate_Post_L	Frontal_Med_Orb_L
ALFF (banda 3)	47	X		X	X	X	X	Caudate_L	ThaT_Pul_R	Cingulate_Post_L
ALFF (banda 4)	47	X		X	X	X	X	Cingulate_Post_L	Rectus_R	ThaT_VL
ALFF (banda 5)	4			X				Cingulate_Post_L	Temporal_Sup_L	ThaT_Pul_R
fALFF (banda 2)	47	X		X	X	X	X	N_Acc_L	OFClat_R	Thal_LP_R
fALFF (banda 3)	28			X			X	Caudate_L	N_Acc_L	OFClat_R
fALFF (banda 4)	1							Vermis_9	Temporal_Pole_Sup_L	Vermis_7
fALFF (banda 4)	28			X			X	Vermis_9	Frontal_Inf_Oper_L	Vermis_10
Conectividad Funcional	3		X					Vermis_7	Thal_VPL_L	Thal_MDI_L
Agrupamiento promedio	2	X								
Camino Característico	47	X		X	X	X	X			
Grado	23	X		X	X	X				

7.3. Discusión

El objetivo principal de este estudio fue estimar el impacto de los métodos de eliminación de artefactos sobre la reproducibilidad usando las métricas de conectomas funcionales y preprocesos más comunes. Se logró identificar que no hay un conceso muy definido de los pipelines por métrica, es decir, los pipelines con mejor reproducibilidad para una métrica dada no necesariamente arrojará los mismo resultados para otra métrica, sin embargo, aquí se logró identificar cuáles son los pipelines que mejor se ajustan a una métrica específica, por ejemplo, ReHo presentó entre los pipelines de mayor reproducibilidad el TCompCor, seguido de la eliminación de artefactos agresivo de ICA-AROMA, asimismo, ALFF y fALFF tuvieron mayor reproducibilidad con el método de eliminación de artefactos ICA-CNN e implementado con nuestro desarrollo RPBC y con la depuración de volúmenes.

La conectividad funcional mejoró los resultados con la depuración de volúmenes e ICA-AROMA no agresivo y en el caso de los grafos globales, la mejora se presentó con el método controversial de regresión global, seguido de ICA-CNN e ICA-AROMA casi en la misma proporción. Finalmente, las métricas de grafos por región tuvieron una notable mejora con la regresión global. Adicionalmente, los resultados concordaron en gran medida con los pocos estudios que se han reportado en reproducibilidad, donde

se ha podido mostrar que las métricas a escala local suelen ser más reproducibles que las métricas a meso escala o escala global [100], como es el caso de ReHo, ALFF y fALFF, las cuales efectivamente mostraron reproducibilidades por encima de moderado para la mayoría de sus regiones e incluso mostraron regiones con reproducibilidad casi perfecta. Caso contrario, la conectividad funcional y las métricas derivadas de grafos tuvieron muchas regiones con valores regulares o leves, algunas regiones con valores por encima de moderado y muy pocas regiones con reproducibilidad casi perfecta.

En este mismo sentido, el estudio permitió ver cuales regiones de interés presentaron un mejor comportamiento o presentaron una mayor reproducibilidad, por ejemplo, la métrica de ReHo conservó casi el mismo orden ICC entre las diferentes regiones, donde es posible ver que las regiones giro poscentral derecho y giro parahipocampal izquierdo fueron las de mayor ICC para el diferente número de vecindades. En el caso de ALFF y fALFF las regiones de núcleo caudado derecho, lóbulo VI de Vermis, núcleo caudado izquierdo y giro cingulado posterior izquierdo, se presentaron en mayor proporción entre las regiones de mayor reproducibilidad. La conectividad funcional para el esquema más reproducible mostró que las regiones de mayor valor ICC son mediodorsal lateral parvocelular izquierdo, posterolateral ventral izquierdo y lóbulo VII de vermis, sin embargo, estas regiones cambian cuando vemos los 6 pipelines más reproducibles, donde el común denominador sigue siendo el lóbulo VII de vermis, además del lóbulo VI de vermis y el giro orbital medial izquierdo. En las métricas de agrupamiento y grados, derivadas del modelo de grafo, se observó que las regiones más reproducibles fueron lóbulo VI de vermis, giro angular izquierdo, lateral ventral izquierdo y el tálamo (núcleo anteroventral).

Adicionalmente, para el caso de la red modo defecto, validada únicamente para la homogeneidad regional y la conectividad funcional, fue posible llegar a varias afirmaciones. Primero, la parcelación con el atlas msdl no afectó el comportamiento entre en número de vecinos, al igual que con el atlas AAL3. Además, los pipelines más reproducibles tuvieron por excelencia el algoritmo de TcompCor. Segundo, la conectividad funcional mostro que las regiones de la red modo defecto fueron las de mayor reproducibilidad, red que suele usada muy a menudo en el ámbito clínico. Finalmente, en el caso de esta última métrica, la mayor reproducibilidad se dio con los algoritmos de depuración y de eliminación de artefactos mediante ICA-CNN, lo cual muestra nuevamente el potencial de esta nueva técnica.

7.4. Conclusión

Hemos presentado un estudio bastante extenso del comportamiento de la reproducibilidad bajo diferentes esquemas de preprocesamiento para diferentes métricas de caracterización de la actividad neuronal sobre imágenes de resonancia magnética funcional en estado de reposo. El análisis permitió encontrar pipelines específicos para cada una de las métricas sujetas a estudio, además, esta sección proporcionó información relevante para establecer cuáles de las regiones anatómicas presentaron mejor reproducibilidad bajo estos mismos esquemas, los cuales a su vez se construyeron bajo los preprocesos más comunes y las técnicas de eliminación de artefactos más relevantes hasta el momento.

8. Conclusiones

Se logró estudiar varias arquitecturas de redes neuronales a fin de clasificar componentes independientes asociadas a ruido o actividad neuronal. Se abordaron desde varios perceptrones multicapa hasta redes neuronales convolucionales, las cuales fueron entrenadas con características de las series de tiempo o los mapas espaciales. Los perceptrones multicapa, permitieron establecer que la combinación de 5 características, derivadas de la serie de tiempo, alcanzaban un poder de clasificación superior al 97%. De igual forma, también se encontró una red neuronal convolucional con una precisión superior al 99%, usada sobre los mapas espaciales (volúmenes) generados por el análisis de componentes independientes. Esta última red, se usó transformando el volumen a una imagen con el nuevo método de reducción por patrones binarios consecutivos y, pese a la pérdida de información inherente al proceso, se logró superar la clasificación de los métodos ya existentes basados en ICA.

Se integraron diferentes herramientas de procesamiento y de cálculo de métricas de la actividad neuronal, a fin de establecer cuál era el efecto de los diferentes esquemas de procesamiento sobre las reproducibilidades de las métricas más relevantes en la exploración funcional en estado de reposo. La integración de las herramientas de procesamiento dio paso a un flujo de procesamiento fluido y continuo, lo cual permitió implementar 48 pipelines diferentes sobre los sujetos de estudio. Además, el desarrollo en el lenguaje Python, permitió incluir nuevas técnicas como la clasificación con redes neuronales convolucionales que no han sido integradas a las actuales herramientas de análisis de fMRIs.

Finalmente, se logró determinar que no existen un pipeline general para las métricas de caracterización de la actividad cerebral, sin embargo, fue posible determinar cuál es el mejor pipeline para cada una de más métricas. Por ejemplo, la homogeneidad regional mostró un mejor desempeño con ICA-AROMA. Las cuatro bandas de ALFF y fALFF mostraron mejor reproducibilidad frente a la nueva técnica basada en la CNN. Cabe resaltar, que los resultados se presentan como un hallazgo novedoso, debido a que los pocos estudios de reproducibilidad se enfocaron en pipelines estándar o se centraron en una única métrica.

El trabajo presenta resultados que hasta el momento no habían sido considerados e incluso se logró determinar cuáles procesos de eliminación de artefactos mejoran la reproducibilidad de las métricas de caracterización de la actividad cerebral. Sin embargo, el cerebro sigue siendo uno de los sistemas más complejos del universo y, por lo tanto, las patologías que lo afectan también tienen su grado de complejidad. Dependiendo de la patología, esta se puede manifestar con mayor detalle sobre una región u otra, así entonces, las regiones y métricas de mayor reproducibilidad podrían no corresponder con la condición patológica. Por lo tanto, partiendo de esta premisa, se podría estudiar la reproducibilidad de cada una de las regiones por separado buscando una mejor evaluación de las patologías.

Adicionalmente, el nuevo método de reducción de volúmenes se muestra como una técnica prometedora para la visualización de componentes independientes, lo cual nos da una nueva perspectiva de estudio, donde se podría validar con expertos a fin de determinar si esta técnica presenta ventajas o es de mayor utilidad que la visualización por cortes axiales. Por otro lado, el enfoque surgió como una necesidad

para reducir el tamaño de la red neuronal convolucional y eliminar la necesidad de usar una convolución en las tres dimensiones. En consecuencia, se esperaba pérdida de información, ya que se pasó de un volumen con valores de punto flotante a uno binario para generar la imagen de 8 bits. La reducción junto con la CNN no solo mostró ser de utilidad para clasificar las componentes, sino que mostró una precisión de clasificación superior a los métodos de clasificación de aprendizaje de máquina existentes. Por lo tanto, otro enfoque futuro podría ser estudiar este método para determinar cuáles de las características conservadas le están dando la precisión de clasificación a la red neuronal y, ya que esta está relacionada con las regiones anatómicas, sería fiable ver si este método revela alguna información adicional de la actividad neuronal contrastando los métodos ya existentes de exploración cerebral.

Cabe destacar que este estudio resalta la importancia de seleccionar el pipeline adecuado para cada métrica. Por lo tanto, la aplicabilidad de este se centra más allá de este enfoque, donde se espera que los resultados obtenidos sean de utilidad para estudios futuros con patologías específicas, como por ejemplo, el Alzheimer o Parkinson. Incluso, estos hallazgos abren la posibilidad de estudiar a detalle los biomarcadores existentes, o buscar nuevos, y hacer que estos tengan validez en el ámbito clínico.

Bibliografía

- [1] M. Redondo, M. Frutos Martín, and Á. Álvarez López, “La enfermedad de Alzheimer : características principales y cuidados,” Universidad de Valladolid, 2017.
- [2] M. Halavi and G. A. Ascoli, “Neuroinformatics,” *Encyclopedia of Neuroscience*. pp. 477–484, 2009.
- [3] P. M. Rossini *et al.*, “Clinical Neurophysiology Methods for analysis of brain connectivity : An IFCN-sponsored review,” *Clin. Neurophysiol.*, vol. 130, no. 10, pp. 1833–1858, 2019.
- [4] J. L. Armony, D. Trejo Martínez, and D. Hernández, “Resonancia Magnética Funcional (RMf): Principios y aplicaciones en Neuropsicología y Neurociencias Cognitivas,” *Rev. Neuropsicol. Latinoam.*, vol. 4, no. 2, pp. 36–50, 2012.
- [5] M. E. Raichle, “The Brain’s Default Mode Network,” *Annu. Rev. Neurosci.*, vol. 38, pp. 433–447, 2015.
- [6] W. Qian, C. E. Fischer, N. W. Churchill, S. Kumar, T. Rajji, and T. A. Schweizer, “Delusions in Alzheimer Disease are Associated With Decreased Default Mode Network Functional Connectivity,” *Am. J. Geriatr. Psychiatry*, vol. 27, no. 10, pp. 1060–1068, 2019.
- [7] S. Delli *et al.*, “NeuroImage : Clinical Somatic symptoms disorders in Parkinson ’ s disease are related to default mode and salience network dysfunction,” vol. 23, no. April, 2019.
- [8] S. Lang and G. Northoff, “Resting-State Functional Magnetic Resonance Imaging: Review of Neurosurgical Applications,” vol. 74, no. 5, 2014.
- [9] G. D. Pearlson, “Applications of Resting State Functional MR Imaging to Neuropsychiatric Diseases,” *Neuroimaging Clinics of North America*, vol. 27, no. 4. Elsevier Inc, pp. 709–723, 2017.
- [10] M. H. Lee, C. D. Smyser, and J. S. Shimony, “Resting state fMRI: A review of methods and clinical applications,” vol. 34, no. 10, pp. 1866–1872, 2014.
- [11] T. Labbé Atenas, E. Ciampi Díaz, J. P. Cruz Quiroga, S. Uribe Arancibia, and C. Cárcamo Rodríguez, “Resonancia magnética funcional: Principios básicos y aplicaciones en neurociencias,” *Radiología*, vol. 60, no. 5, pp. 368–377, 2018.
- [12] W. R. Shirer, H. Jiang, C. M. Price, B. Ng, and M. D. Greicius, “NeuroImage Optimization of rs-fMRI Pre-processing for Enhanced Signal-Noise Separation , Test-Retest Reliability , and Group Discrimination,” *Neuroimage*, vol. 117, pp. 67–79, 2015.
- [13] R. Zaki, A. Bulgiba, N. Nordin, and N. A. Ismail, “A Systematic Review of Statistical Methods Used to Test for Reliability of Medical Instruments Measuring Continuous Variables,” no. December 2009, 2010.
- [14] C. Caballero-Gaudes and R. C. Reynolds, “Methods for cleaning the BOLD fMRI signal,” *Neuroimage*, vol. 154, no. December 2016, pp. 128–149, 2017.
- [15] X. Zuo, B. B. Biswal, and R. A. Poldrack, “Editorial : Reliability and Reproducibility in Functional Connectomics,” vol. 13, no. February, pp. 1–4, 2019.
- [16] L. Griffanti *et al.*, “Hand classification of fMRI ICA noise components,” *Neuroimage*, vol. 154, no. December 2016, pp. 188–205, 2017.
- [17] J. I. Glaser, A. S. Benjamin, R. Farhoodi, and K. P. Kording, “Progress in Neurobiology The roles of supervised machine learning in systems neuroscience,” *Prog. Neurobiol.*, vol. 175, no. February, pp. 126–137, 2019.

- [18] M. Zitnik, F. Nguyen, B. Wang, J. Leskovec, A. Goldenberg, and M. M. Hoffman, “Machine learning for integrating data in biology and medicine : Principles , practice , and opportunities,” *Inf. Fusion*, vol. 50, no. September 2018, pp. 71–91, 2019.
- [19] M. Wilke and T. Baldeweg, “A multidimensional artefact-reduction approach to increase robustness of first-level fMRI analyses : Censoring vs . interpolating,” *J. Neurosci. Methods*, vol. 318, no. November 2018, pp. 56–68, 2019.
- [20] Y. Behzadi, K. Restom, J. Liau, and T. T. Liu, “A component based noise correction method (CompCor) for BOLD and perfusion based fMRI,” *Hum. Brain Mapp. J.*, vol. 37, no. 1, pp. 90–101, 2007.
- [21] S. Haykin, “Multilayer perceptrons,” in *Neural Networks and Learning Machines*, Third., USA: Prentice Hall, 2009, pp. 122–125.
- [22] D. Ravi *et al.*, “Deep Learning for Health Informatics,” *IEEE J. Biomed. Heal. Informatics*, vol. 21, no. 1, pp. 4–21, 2017.
- [23] S. Muhammad, A. Muhammad, M. Adnan, Q. Muhammad, A. Majdi, and M. K. Khan, “Medical Image Analysis using Convolutional Neural Networks A Review,” *J. Med. Syst.*, vol. 42, pp. 1–13, 2018.
- [24] J. Teuwen and N. Moriakova, “Convolutional neural networks,” in *Handbook of Medical Image Computing and Computer Assisted Intervention*, Academic P., <https://doi.org/10.1016/B978-0-12-816176-0.00025-9>, Ed. 2020, pp. 481–501.
- [25] E. Proal, M. Alvarez-Segura, M. de la Iglesia-Vayá, L. Martí-Bonmatí, and F. X. Castellanos, “[Functional cerebral activity in a state of rest: connectivity networks].,” *Rev. Neurol.*, vol. 52 Suppl 1, no. 0 1, pp. S3-10, 2011.
- [26] J. C. Reijneveld, S. C. Ponten, H. W. Berendse, and C. J. Stam, “The application of graph theoretical analysis to complex networks in the brain.,” *Clin. Neurophysiol. Off. J. Int. Fed. Clin. Neurophysiol.*, vol. 118, no. 11, pp. 2317–2331, Nov. 2007.
- [27] J. Wang, X. Zuo, and Y. He, “Graph-based network analysis of resting-state functional MRI,” *Front. Syst. Neurosci.*, vol. 4, p. 16, 2010.
- [28] P. López Larrubia, “Aplicaciones de la Resonancia Magnética Nuclear a la investigación biomédica,” *SEBBM Divulg. LA Cienc. AL ALCANCE LA MANO*, p. 2, 2009.
- [29] W. Thompson and J. Aston, “Resting-State fMRI,” in *Handbook of Neuroimaging Data Analysis*, USA: CRC Press, 2016, pp. 165–166.
- [30] J. D. Power, K. A. Barnes, A. Z. Snyder, B. L. Schlaggar, and S. E. Petersen, “NeuroImage Spurious but systematic correlations in functional connectivity MRI networks arise from subject motion,” *Neuroimage*, vol. 59, no. 3, pp. 2142–2154, 2012.
- [31] J. B. Andre *et al.*, “Toward Quantifying the Prevalence , Severity , and Cost Associated With Patient Motion During Clinical MR Examinations,” *J. Am. Coll. Radiol.*, vol. 12, no. 7, pp. 689–695, 2019.
- [32] K. Chen, A. Azeez, D. Y. Chen, and B. B. Biswal, “Resting-state Functional Connectivity: Signal Origins and Analytic Methos,” *Neuroimag Clin N Am*, pp. 15–23, 2020.
- [33] M. A. Lindquist, “The Statistical Analysis of fMRI Data,” vol. 23, no. 4, pp. 439–464, 2008.
- [34] D. S. Margulies *et al.*, “Resting developments : a review of fMRI post-processing methodologies for spontaneous brain activity,” pp. 289–307, 2010.
- [35] M. Khosla, K. Jamison, G. H. Ngo, A. Kuceyeski, and M. R. Sabuncu, “Machine

- learning in resting-state fMRI analysis,” *Magnetic Resonance Imaging*, vol. 64, no. December 2018. Elsevier, pp. 101–121, 2019.
- [36] C. F. Beckmann, M. Deluca, J. T. Devlin, S. M. Smith, J. R. Hospital, and O. Ox, “Investigations into resting-state connectivity using independent component analysis,” no. May, pp. 1001–1013, 2005.
- [37] R. H. R. Pruim, M. Mennes, D. Van Rooij, A. Llera, J. K. Buitelaar, and C. F. Beckmann, “NeuroImage ICA-AROMA : A robust ICA-based strategy for removing motion artifacts from fMRI data,” *Neuroimage*, vol. 112, pp. 267–277, 2015.
- [38] G. Salimi-khorshidi, G. Douaud, C. F. Beckmann, M. F. Glasser, L. Griffanti, and S. M. Smith, “NeuroImage Automatic denoising of functional MRI data : Combining independent component analysis and hierarchical fusion of classifiers,” *Neuroimage*, vol. 90, pp. 449–468, 2014.
- [39] Q. H. Zou *et al.*, “An improved approach to detection of amplitude of low-frequency fluctuation (ALFF) for resting-state fMRI: Fractional ALFF,” *J. Neurosci. Methods*, vol. 172, no. 1, pp. 137–141, 2008.
- [40] M. Wibral, J. T. Lizier, S. Vögler, V. Priesemann, and R. Galuske, “Local active information storage as a tool to understand distributed neural information processing,” vol. 8, no. January, pp. 1–11, 2014.
- [41] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, and F. E. Alsaadi, “Neurocomputing A survey of deep neural network architectures and their applications ☆,” *Neurocomputing*, vol. 234, no. November 2016, pp. 11–26, 2017.
- [42] K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, vol. 2015 Inter, pp. 1026–1034.
- [43] A. S. Lundervold and A. Lundervold, “An overview of deep learning in medical imaging focusing on,” *Zeitschrift für Medizinische Phys.*, vol. 29, no. 2, pp. 102–127, 2019.
- [44] M. Mostapha and M. Styner, “Role of deep learning in infant brain MRI analysis,” *Magn. Reson. Imaging*, vol. 64, no. January, pp. 171–189, 2019.
- [45] Z. Mao, Y. Su, G. Xu, X. Wang, and Y. Huang, “Spatio-temporal deep learning method for ADHD fMRI classification,” vol. 499, pp. 1–11, 2019.
- [46] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, “Neurocomputing Deep learning for visual understanding : A review,” vol. 187, pp. 27–48, 2016.
- [47] W. Hernandez and A. Mendez, “Application of Principal Component Analysis to Image Compression,” in *Statistics - Growing Data Sets and Growing Demand for Statistics*, Türkmen Gö., Open access peer-reviewed chapter, 2018.
- [48] M. Pietikäinen and G. Zhao, “Two decades of local binary patterns: A survey,” in *Advances in Independent Component Analysis and Learning Machines*, Academic P., 2015, pp. 175–210.
- [49] J. Qiu, Q. Zhang, T. Bi, G. Wu, D. Wei, and W. Yang, “Southwest University Longitudinal Imaging Multimodal (SLIM) Brain Data Repository: A Long-term Test-Retest Sample of Young Healthy Adults in Southwest China.” [Online]. Available: http://fcon_1000.projects.nitrc.org/indi/retro/southwestuni_qiu_index.html.
- [50] “HCP Young Adult - Connectome - Publications.” [Online]. Available: <https://www.humanconnectome.org/study/hcp-young-adult>. [Accessed: 17-Mar-

- 2020].
- [51] G. Salimi-khorshidi *et al.*, “Fix Hand-Training Datasets.” [Online]. Available: <https://www.fmrib.ox.ac.uk/datasets/FIX-training/>.
 - [52] P. McCarthy, “FSLeyes.” Zenodo, 2019.
 - [53] X. Zuo *et al.*, “The Oscillating Braing: Complex and Realiabe,” vol. 49, no. 2, pp. 1432–1445, 2011.
 - [54] D. J. Bouveresse and D. N. Rutledge, *Independent Components Analysis : Theory and Applications*, 1st ed., vol. 30. Elsevier B.V., 2016.
 - [55] A. Phinyomark, S. Thongpanja, and H. Hu, “The Usefulness of Mean and Median Frequencies in Electromyography Analysis,” no. October, 2012.
 - [56] Python, “The python network.” [Online]. Available: <https://www.python.org/downloads/>.
 - [57] S. Suzuki and K. Be, “Topological Structural Analysis of Digitized Binary Images by Border Following,” vol. 46, pp. 32–46, 1985.
 - [58] A. Schmitt, N. Upadhyay, J. Anthony, S. Rojas, and H. K. Str, “Modulation of Distinct Intrinsic Resting State Brain Networks by Acute Exercise Bouts of Differing Intensity,” vol. 5, pp. 39–55, 2019.
 - [59] D. Carone, R. Licenik, S. Suri, L. Gri, N. Filippini, and J. Kennedy, “NeuroImage : Clinical Impact of automated ICA-based denoising of fMRI data in acute stroke patients,” vol. 16, no. January, pp. 23–31, 2017.
 - [60] J. Bijsterbosch, M. S. Smith, and C. F. Beckmann, “Data preprocessing - Conventional preprocessing steps,” in *Introduction to Resting State fMRI Functional Connectivity*, M. Jenkinson and M. Chappell, Eds. United Kingdom: Oxford University Press, 2017, pp. 28–33.
 - [61] R. Sladky, K. J. Friston, J. Tröstl, R. Cunnington, E. Moser, and C. Windischberger, “Slice-timing effects and their correction in functional MRI,” *Neuroimage*, vol. 58, no. 2, pp. 588–594, 2011.
 - [62] J. D. Power, B. L. Schlaggar, and S. E. Petersen, “Recent progress and outstanding issues in motion correction in resting state fMRI,” *Neuroimage*, vol. 105, pp. 536–551, 2015.
 - [63] A. C. Evans, A. L. Janke, D. L. Collins, and S. Baillet, “Brain templates and atlases,” *Neuroimage*, vol. 62, no. 2, pp. 911–922, 2012.
 - [64] C. Ong and R. S. Hamilton, “Constructing Computational Pipelines,” *Encycl. Bioinforma. Comput. Biol.*, pp. 135–143, Jan. 2019.
 - [65] C.-G. Yan, X.-D. Wang, X.-N. Zuo, and Y.-F. Zang, “DPABI: Data Processing & Analysis for (Resting-State) Brain Imaging,” *Neuroinformatics*, vol. 14, no. 3, pp. 339–351, 2016.
 - [66] “International Neuroimaging Data-sharing Initiative.” [Online]. Available: <https://fcp-indi.github.io/>. [Accessed: 02-Jun-2020].
 - [67] S. M. S. M.W. Woolrich, S. Jbabdi, B. Patenaude, M. Chappell, S. Makni, T. Behrens, C. Beckmann, M. Jenkinson, “Bayesian analysis of neuroimaging data in FSL,” *Neuroimage*, vol. 45, no. 1, pp. 173–186, 2009.
 - [68] and P. M. M. S.M. Smith, M. Jenkinson, M.W. Woolrich, C.F. Beckmann, T.E.J. Behrens, H. Johansen-Berg, P.R. Bannister, M. De Luca, I. Drobnyak, D.E. Flitney, R. Niazy, J. Saunders, J. Vickers, Y. Zhang, N. De Stefano, J.M. Brady, “Advances in functional and structural MR image analysis and implementation as FSL,” *Neuroimage*, vol. 23, pp. 208–219, 2004.

- [69] M. Jenkinson, C. F. Beckmann, T. E. Behrens, M. W. Woolrich, and S. M. Smith, “FSL,” *Neuroimage*, vol. 62, no. 782–90, 2012.
- [70] J. Wang, X. Wang, M. Xia, X. Liao, A. Evans, and Y. He, “GRETNA: A graph theoretical network analysis toolbox for imaging connectomics,” *Front. Hum. Neurosci.*, vol. 9, no. JUNE, pp. 1–16, 2015.
- [71] “CCS Lab.” [Online]. Available: <https://ccs-lab.github.io/>. [Accessed: 02-Jun-2020].
- [72] “CONN toolbox.” [Online]. Available: <https://web.conn-toolbox.org/>. [Accessed: 02-Jun-2020].
- [73] “SPM Software - Statistical Parametric Mapping.” [Online]. Available: <https://www.fil.ion.ucl.ac.uk/spm/software/>. [Accessed: 02-Jun-2020].
- [74] R. W. Cox, “AFNI: Software for analysis and visualization of functional magnetic resonance neuroimages,” *Comput. Biomed. Res.*, vol. 29, no. 3, pp. 162–173, 1996.
- [75] “FreeSurfer.” [Online]. Available: <https://surfer.nmr.mgh.harvard.edu/>. [Accessed: 02-Jun-2020].
- [76] “Nilearn: Machine learning for NeuroImaging in Python — Machine learning for NeuroImaging.” [Online]. Available: <https://nilearn.github.io/index.html#>. [Accessed: 02-Jun-2020].
- [77] M. Brett *et al.*, “nipy/nibabel: 3.0.1,” Jan. 2020.
- [78] K. Gorgolewski *et al.*, “Nipype: A flexible, lightweight and extensible neuroimaging data processing framework in Python,” *Front. Neuroinform.*, vol. 5, Aug. 2011.
- [79] K. J. Gorgolewski *et al.*, “BIDS apps: Improving ease of use, accessibility, and reproducibility of neuroimaging data analysis methods,” *PLoS Comput. Biol.*, vol. 13, no. 3, Mar. 2017.
- [80] D. N. Greve and B. Fischl, “Accurate and robust brain image alignment using boundary-based registration,” *Neuroimage*, vol. 48, no. 1, pp. 63–72, 2009.
- [81] Y. Zang, T. Jiang, Y. Lu, Y. He, and L. Tian, “Regional homogeneity approach to fMRI data analysis,” *Neuroimage*, vol. 22, no. 1, pp. 394–400, May 2004.
- [82] P. A. Taylor and Z. S. Saad, “FATCAT: (An Efficient) functional and tractographic connectivity analysis toolbox,” *Brain Connect.*, vol. 3, no. 5, pp. 523–535, 2013.
- [83] E. T. Rolls, C. C. Huang, C. P. Lin, J. Feng, and M. Joliot, “Automated anatomical labelling atlas 3,” *Neuroimage*, vol. 206, no. August 2019, p. 116189, 2020.
- [84] J. Saramäki, M. Kivelä, J. P. Onnela, K. Kaski, and J. Kertész, “Generalizations of the clustering coefficient to weighted complex networks,” *Phys. Rev. E - Stat. Nonlinear, Soft Matter Phys.*, vol. 75, no. 2, pp. 2–5, 2007.
- [85] M. Kaiser, “Mean clustering coefficients: The role of isolated nodes and leafs on clustering measures for small-world networks,” *New J. Phys.*, vol. 10, 2008.
- [86] Y. A. N. Chao-Gan, “Data Processing Assistant for Resting-State fMRI (DPARSF),” 2014.
- [87] J. Yang, S. Gohel, and B. Vachha, “Current methods and new directions in resting state fMRI,” *Clin. Imaging*, vol. 65, no. November 2019, pp. 47–53, 2020.
- [88] K. A. Smitha *et al.*, “Resting state fMRI: A review on methods in resting state connectivity analysis and resting state networks,” *Neuroradiol. J.*, vol. 30, no. 4, pp. 305–317, Aug. 2017.
- [89] T. K. Koo and M. Y. Li, “A Guideline of Selecting and Reporting Intraclass Correlation Coefficients for Reliability Research,” *J. Chiropr. Med.*, vol. 15, no. 2, pp. 155–163, 2016.
- [90] J. H. Fröhner, V. Teckentrup, M. N. Smolka, and N. B. Kroemer, “Addressing the

- reliability fallacy in fMRI: Similar group effects may arise from unreliable individual effects,” *Neuroimage*, vol. 195, no. March, pp. 174–189, 2019.
- [91] P. E. Shrout and J. L. Fleiss, “Intraclass correlations: uses in assessing rater reliability.1. Shrout PE, Fleiss JL: Intraclass correlations: uses in assessing rater reliability. *Psychol Bull* 1979, 86:420–8.,” *Psychol. Bull.*, vol. 86, no. 2, pp. 420–8, 1979.
- [92] D. Liljequist, B. Elfving, and K. S. Roaldsen, *Intraclass correlation – A discussion and demonstration of basic features*, vol. 14, no. 7. 2019.
- [93] K. O. McGraw and S. P. Wong, “Forming Inferences about Some Intraclass Correlation Coefficients,” *Psychol. Methods*, vol. 1, no. 1, pp. 30–46, 1996.
- [94] M. Termenon, A. Jaillard, C. Delon-Martin, and S. Achard, “Reliability of graph analysis of resting state fMRI using test-retest dataset from the Human Connectome Project,” *Neuroimage*, vol. 142, pp. 172–187, 2016.
- [95] Joaquín Amat Rodrigo, “Resampling: test de permutación, simulación de Monte Carlo y Bootstrapping,” 2016. [Online]. Available: https://www.cienciadedatos.net/documentos/23_resampling_test_permutacion_simulacion_de_monte_carlo_bootstrapping.
- [96] Sergios Theodoridis, “Monte Carlo Methods,” in *Machine Learning*, Second., Academic Press, 2020, pp. 731–769.
- [97] J. R. Landis and G. G. Koch, “The Measurement of Observer Agreement for Categorical Data,” *Biometrics*, vol. 33, no. 1, pp. 159–174, Jun. 1977.
- [98] G. Varoquaux and R. C. Craddock, “Learning and comparing functional connectomes across subjects,” *Neuroimage*, vol. 80, pp. 405–415, 2013.
- [99] G. Varoquaux, A. Gramfort, F. Pedregosa, V. Michel, and B. Thirion, “Multi-subject dictionary learning to segment an atlas of brain spontaneous activity,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 6801 LNCS, pp. 562–573, 2011.
- [100] B. Chen, T. Xu, C. Zhou, L. Wang, N. Yang, and Z. Wang, “Individual Variability and Test-Retest Reliability Revealed by Ten Repeated Resting-State Brain Scans over One Month,” pp. 1–21, 2015.
- [101] M. E. J. Newman, “Mixing patterns in networks,” *Phys. Rev. E*, vol. 67, no. 2, p. 26126, Feb. 2003.
- [102] J. Foster, D. Foster, P. Grassberger, and M. Paczuski, “Edge Direction and the Structure of Networks,” *Proc. Natl. Acad. Sci. U. S. A.*, vol. 107, pp. 10815–10820, Jun. 2010.
- [103] W. H. Wang and Y. S. Xue, “Extremal Estrada indices of the weighted trees with fixed total weight sum,” *Appl. Math. Comput.*, vol. 354, pp. 32–41, 2019.
- [104] L. W. Beineke, O. R. Oellermann, and R. E. Pippert, “The average connectivity of a graph,” *Discrete Math.*, vol. 252, no. 1–3, pp. 31–45, 2002.
- [105] Phillip Bonacich, “Power and Centrality: A Family of Measures,” *Am. J. Sociol.*, vol. 92, no. 5, pp. 1170–1182, 2012.
- [106] M. E. J. Newman, “Measures and metrics,” in *Networks: An Introduction*, USA: Oxford University Press, 2010, p. 169.
- [107] K. I. Goh, B. Kahng, and D. Kim, “Universal Behavior of Load Distribution in Scale-Free Networks,” *Phys. Rev. Lett.*, vol. 87, no. 27, pp. 278701–278701–4, 2001.
- [108] T. Pagerank, C. Ranking, and B. Order, “1 Introduction and Motivation 2 A Ranking for Every Page on the Web,” pp. 1–17, 1998.

- [109] L. Amy N and M. Carl D, “A Survey of Eigenvector Methods for Web Information Retrieval,” *SIAM Rev.*, vol. 47, no. 1, pp. 135–161, 2005.
- [110] L. C. Freeman, “Centrality in social networks conceptual clarification,” *Soc. Networks*, vol. 1, no. 3, pp. 215–239, 1978.
- [111] U. Brandes, “A faster algorithm for betweenness centrality,” *J. Math. Sociol.*, vol. 25, no. 2, pp. 163–177, Jun. 2001.
- [112] U. Brandes, “On variants of shortest-path betweenness centrality and their generic computation,” *Soc. Networks*, vol. 30, no. 2, pp. 136–145, 2008.
- [113] U. Brandes and C. Pich, “Centrality estimation in large networks,” *Int. J. Bifurc. Chaos*, vol. 17, no. 7, pp. 2303–2318, 2007.
- [114] L. C. Freeman, “A Set of Measures of Centrality Based on Betweenness,” *Sociometry*, vol. 40, no. 1. p. 35, 1977.
- [115] F. S. Roberts and J. H. Spencer, “A characterization of clique graphs,” *J. Comb. Theory, Ser. B*, vol. 10, no. 2, pp. 102–108, 1971.
- [116] J. M. Kleinberg, “Authoritative sources in a hyperlinked environment,” *J. ACM*, vol. 46, no. 5, pp. 604–632, 1999.
- [117] V. Batagelj and M. Zaversnik, “An $O(m)$ Algorithm for Cores Decomposition of Networks,” pp. 1–10, 2003.
- [118] P. G. Lind, M. C. González, and H. J. Herrmann, “Cycles and clustering in bipartite networks,” *Phys. Rev. E*, vol. 72, no. 5, p. 56127, Nov. 2005.
- [119] A. Kosowski and K. Manuszewski, “Classical coloring of graphs,” pp. 1–19, 2004.
- [120] J. S. K. Maciej M. Sysło, Narsingh Deo, “Discrete optimization algorithms : with Pascal programs,” *Mineola, N.Y. Dover Publ.*, 2006.
- [121] M. E. J. Newman, “Scientific collaboration networks. II. Shortest paths, weighted networks, and centrality,” *Phys. Rev. E*, vol. 64, no. 1, p. 16132, Jun. 2001.

Leonel Mera Jiménez

Nombre y Firma del Estudiante

C.C. 1086360521

John Fredy Ochoa Gómez

Nombre y Firma del Director

C.C. 71279364

9. Anexos

A 1. Métricas adicionales derivadas de la teoría de grafos

En neurociencias existen muchas métricas y enfoques para caracterizar la actividad cerebral. Uno de ellos es la teoría de grafos, la cual se basa en modelar la actividad neuronal como un grafo, en donde los nodos son las regiones anatómicas del cerebro y los vértices la fuerza de la conexión o conectividad funcional. En este sentido, aparte de las métricas propias para la actividad cerebral, también se pueden obtener muchas métricas propias de la teoría de grafos, aplicadas al modelo del cerebro como un modelo de grafo ponderado.

En este documento ya vimos algunas métricas derivadas de la teoría de grafos, sin embargo, en esta sección se presentan 17 métricas adicionales con su respectiva definición y con los resultados obtenidos con los mismos 17 sujetos de estudio.

Las métricas calculadas son las siguientes:

- La variabilidad del grado del grafo o a lo que es igual a el grado de asortatividad del grafo se estimó con el coeficiente de correlación de pearson [101], [102] mediante la función 'degree_pearson_correlation_coefficient'.
- Se calculó el índice Estrada IE mediante la función 'estrada_index' para el grafo ponderado G definido como [103]:

$$IE(G) = \sum_{i=1}^n e^{\lambda_i} \quad (20)$$

Donde λ_i son los valores propios de cada uno de los nodos.

- Se calculó la transitividad del grafo con la función 'transitivity' el cual se define como el número relativo de triángulos en el grafo, es decir, la relación entre el número de bucles cerrados de longitud tres y el número total presente en la red, llamado ocasionalmente como coeficiente de agrupamiento global.
- Con la función 'average_node_connectivity' se calculó la conectividad media por nodo \bar{k} definido como el promedio, sobre todos los pares de nodos, del número máximo de caminos disjuntos internos que conectan estos nodos el cual se expresa matemáticamente como:

$$\bar{k}(G) = \frac{\sum_{u,v} k_g(u,v)}{\binom{n}{2}} \quad (21)$$

Donde u y v son todos los posibles nodos, k_g la conectividad del par u,v siendo esta el número mínimo de nodos necesarios para desconectar dichos pares y n el número de nodos existentes en el grafo [104].

- Con la función 'eigenvector centrality_numpy' se calculó la centralidad de vectores propios, partiendo de la centralidad relativa x del nodo u en función

de la centralidad de cada nodo t de los M vecinos próximos, siguiendo la siguiente relación [105]:

$$x_u = \frac{1}{\lambda} \sum_{t \in M} x_t = \frac{1}{\lambda} \sum_{t \in G} a_{v,t} x_t \quad (22)$$

Donde $a_{u,v}$ es el peso de la matriz de adyacencia del par de nodos u, t , expresión de la cual se deriva la relación de los vectores propios (ecuación (23)), necesaria para calcular la centralidad de vectores propios.

$$Ax = \lambda x \quad (23)$$

Donde A es la matriz de adyacencia del grafo con el valor propio λ y este es el valor más grande asociado con el vector propio de la matriz de adyacencia, valor que se toma del teorema de Perron-Frobenius [106].

- La centralidad de Katz x_i se calculó con la función 'katz_centrality_numpy' partiendo de la centralidad de un nodo en función de la centralidad de sus vecinos próximos, es decir, se calcula la influencia relativa de un nodo dentro de una red midiendo el número de vecinos próximos y el número de todos los nodos en la red que se conectan al nodo i a través de estos vecinos próximos [107].

$$x_i = \alpha \sum_j A_{ij} x_j + \beta \quad (24)$$

Donde A nuevamente representa la matriz de adyacencia, α un factor de atenuación con valor menor al inverso del máximo valor propio λ_{max} y β una constante que controla la centralidad inicial.

- Se calculó el rango de página con el uso de la función 'pagerank_numpy', calculando la clasificación de los nodos en el grafo según la estructura de los enlaces del nodo en consideración [108], [109].
- Con la función 'closeness_centrality', se calculó la centralidad de proximidad C del nodo u como el inverso de la suma de las distancias de camino más cortas desde el nodo de interés con todos los nodos contenidos en el grafo No ponderado, expresado como sigue [110]:

$$C(u) = \frac{n-1}{\sum_{v=1}^{n-1} d(v,u)} \quad (25)$$

Siendo $d(u,v)$ la distancia más corta entre el par de nodos u, v .

- Con el uso de la función 'betweenness_centrality', se encontró la centralidad de intermediación de cada nodo como la suma de las fracciones de los caminos más cortos de todos los posibles pares de nodos que pasan a través de mismo [111]–[114].
- Con la función 'average_neighbor_degree' se encontró el grado promedio de cada nodo i mediante la expresión matemática (26).

$$k_{nm,i}^w = \frac{1}{S_i} \sum_{j \in N(i)} w_{ij} k_j \quad (26)$$

donde $N(i)$ son el conjunto de vecinos del nodo i -ésimo y k_j el grado correspondiente a cada nodo, w_{ij} el peso asociado entre el nodo i y el nodo vecino j .

- En teoría de grafos se tiene que un clique es un subconjunto C del grafo G tal que:

$$C \subseteq V(G) \text{ y } u, v \in C \wedge u \neq v \Rightarrow u \wedge v \in E(G) \quad (27)$$

Es decir, cada vértice de un clique C hace parte de los vértices del grafo G y cada par de vértices $u - v$ distintos son adyacentes al grafo G . Por lo tanto, con la función 'number_of_cliques' se encontró el número máximo de cliques asociados a cada nodo y con la función 'node_clique_number' se calculó el tamaño del clique más grande por nodo [115].

- Con la función 'hits_numpy' derivada del algoritmo *Hypertext Induced Topic Selection*, se estimaron el valor de autoridad en función de los enlaces entrantes y el valor de concentrador del nodo en función de los enlaces salientes [109], [116].
- El número de núcleo de un nodo es el número más grande de un núcleo- k que contienen a ese nodo, siendo el núcleo- k un subgrafo máximo derivado del grafo principal en la que todos los vértices tienen un grado de al menos k , en otras palabras, el subgrafo máximo es el resultado de eliminar de manera repetida todos los vértices de grado menor a k , y si existe un núcleo- k no vacío, entonces se dice que el nodo tienen un número de núcleo k [117]. El valor del número del núcleo se calculó con la función 'core_number'.
- Con la función 'square_clustering' se estimó la fracción de posibles cuadrados relacionados con cada nodo a partir de la siguiente expresión.

$$C_4(v) = \frac{\sum_{u=1}^{k_v} \sum_{w=u+1}^{k_v} q_v(u, w)}{\sum_{u=1}^{k_v} \sum_{w=u+1}^{k_v} [a_v(u, w) + q_v(u, w)]} \quad (28)$$

Donde $q_v(u, v)$ es el número de vecinos en común de u y donde w no es un agrupamiento cuadrado. Además,

$$a_v(u, v) = (k_u - (1 - q_v(u, w) + \theta_{uv})) (k_w - (1 + q_v(u, w) + \theta_{uw})) \quad (29)$$

Donde para cada par de nodos θ vale 1 si estos están conectados y cero en caso contrario [118].

- Mediante la función 'triangles' se calculó el número de triángulos que se forman para cada nodo con ese nodo como vértice de todos los posibles triángulos.
- La coloración de un vértice es un etiquetado especial que se le da a los vértices de un grafo, cuya etiqueta no puede tener el mismo valor para dos pares de vértices adyacentes, por lo tanto, con la función 'greedy_color' se encontró el 'color' de cada vértice con la menor cantidad de colores posibles sobre el grafo, implementado la estrategia de coloración codiciosa [119], [120].

B 4. ICC sobre las métricas de ALFF y fALFF en todos los pipelines

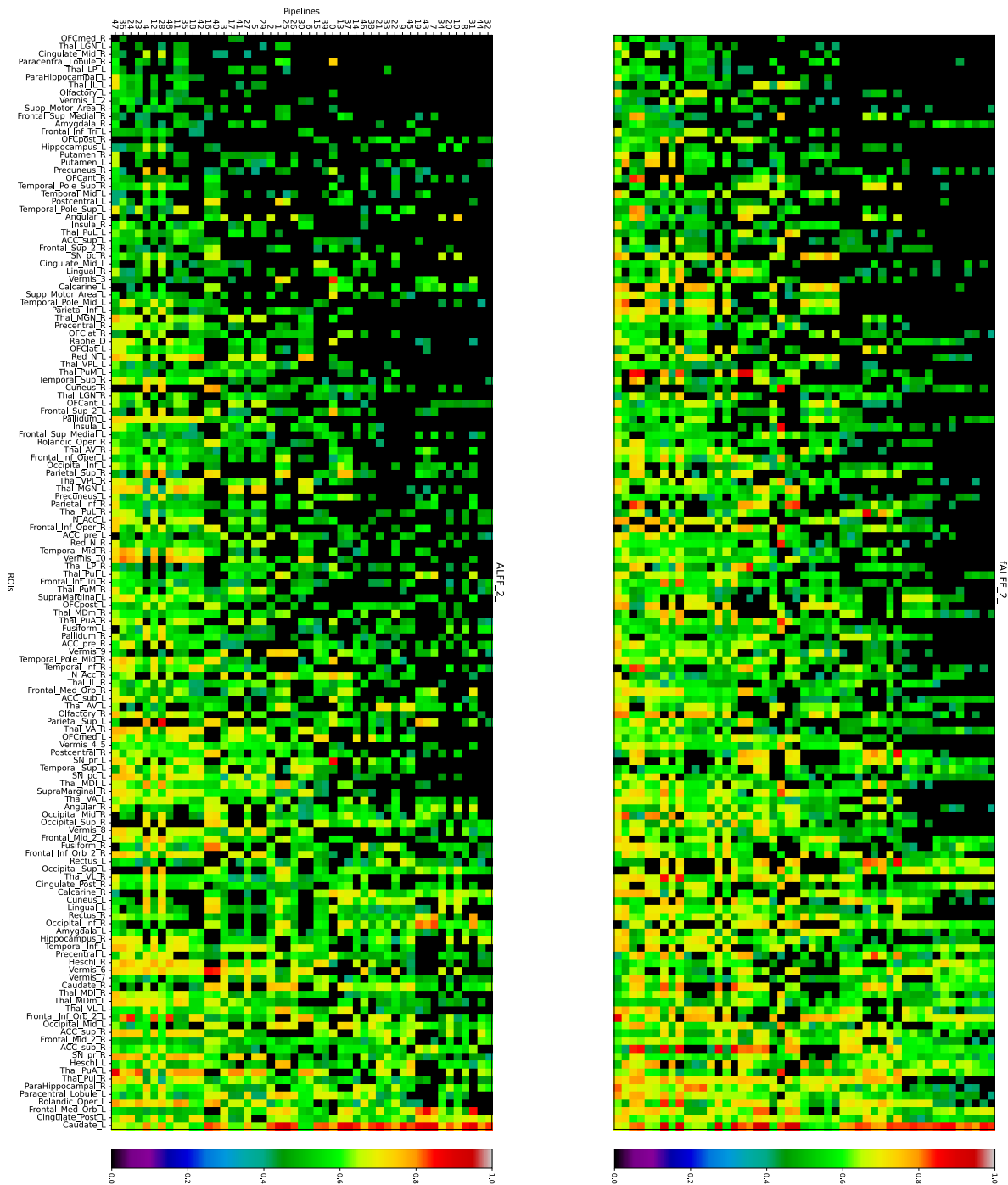


Figura 50 ICC sobre las métricas de ALFF y fALFF en banda 2 en las diferentes regiones para todos los pipelines.

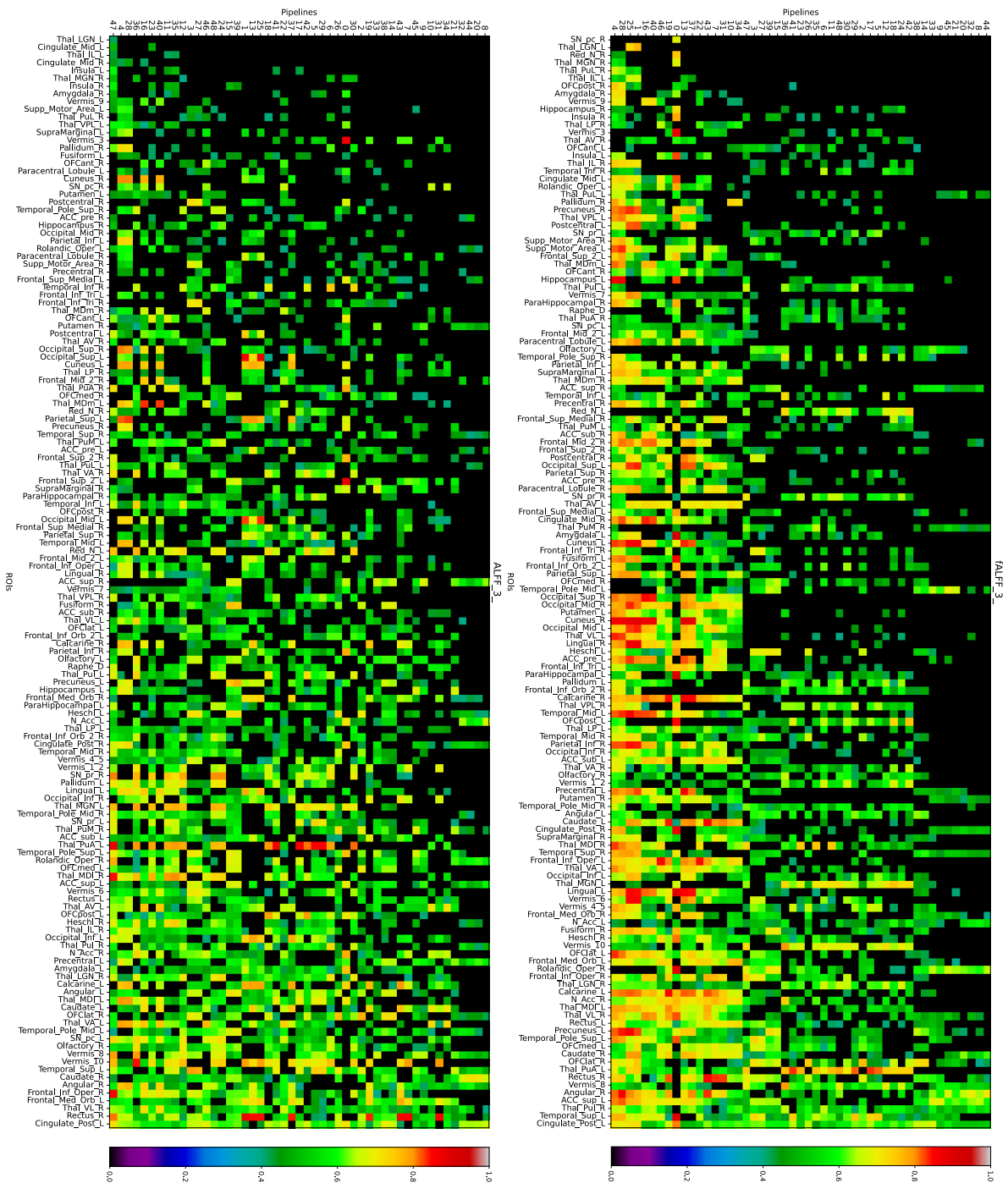


Figura 51 ICC sobre las métricas de ALFF y fALFF en banda 3 en las diferentes regiones para todos los pipelines.

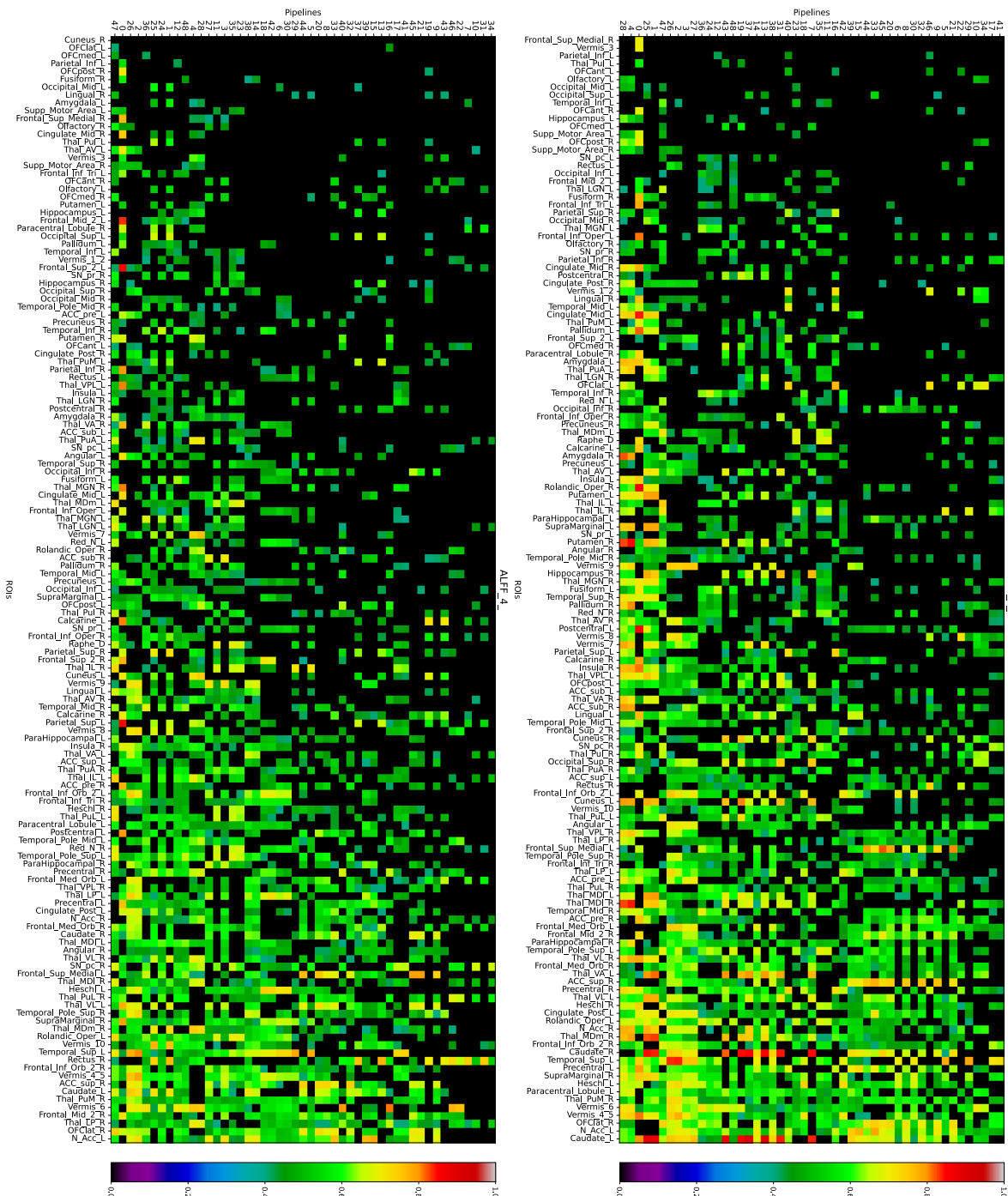


Figura 52 ICC sobre las métricas de ALFF y fALFF en banda 4 en las diferentes regiones para todos los pipelines.

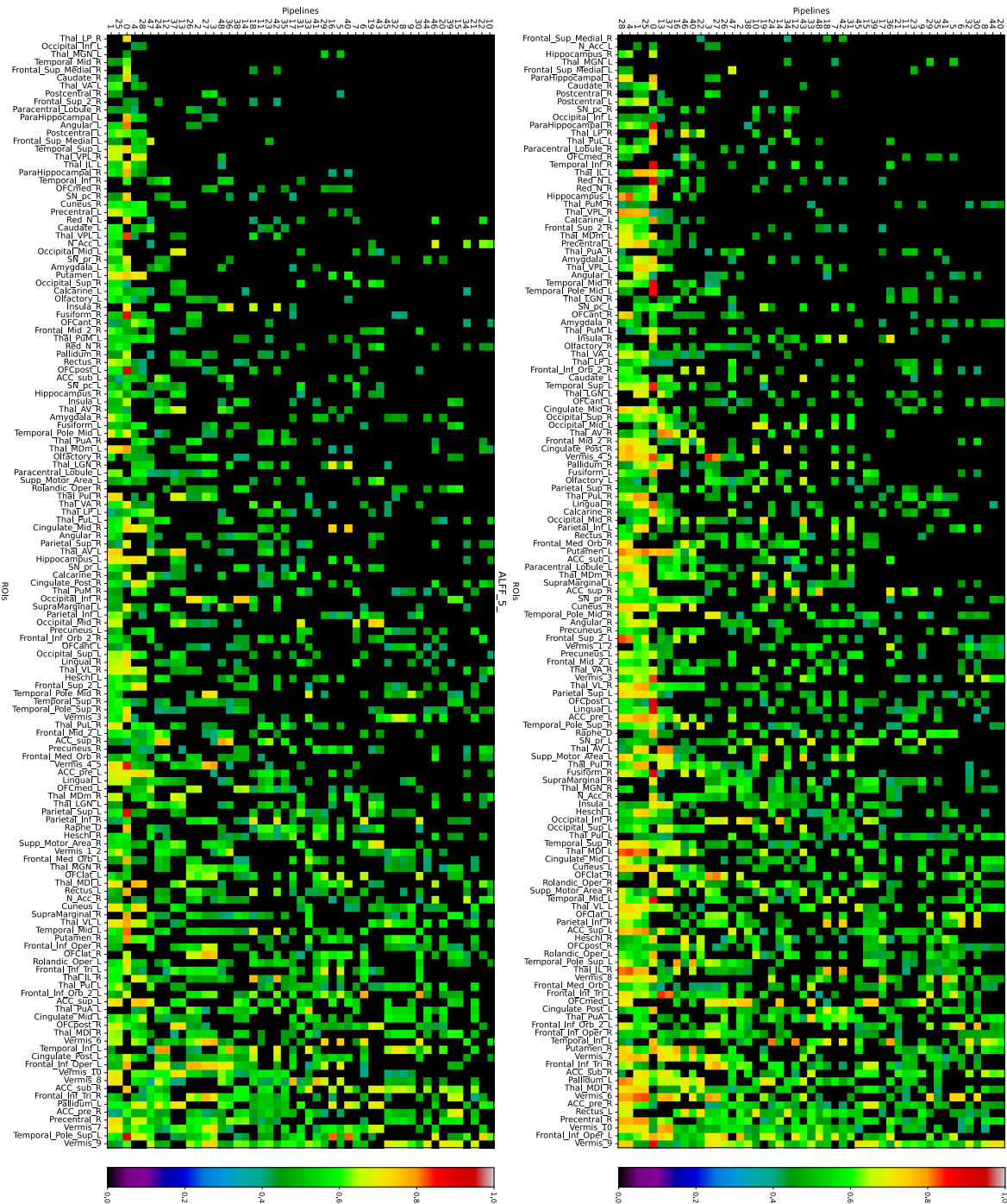


Figura 53 ICC sobre las métricas de ALFF y fALFF en banda 5 en las diferentes regiones para todos los pipelines.

B 5. ICC sobre la métrica de conectividad funcional en todos los pipelines

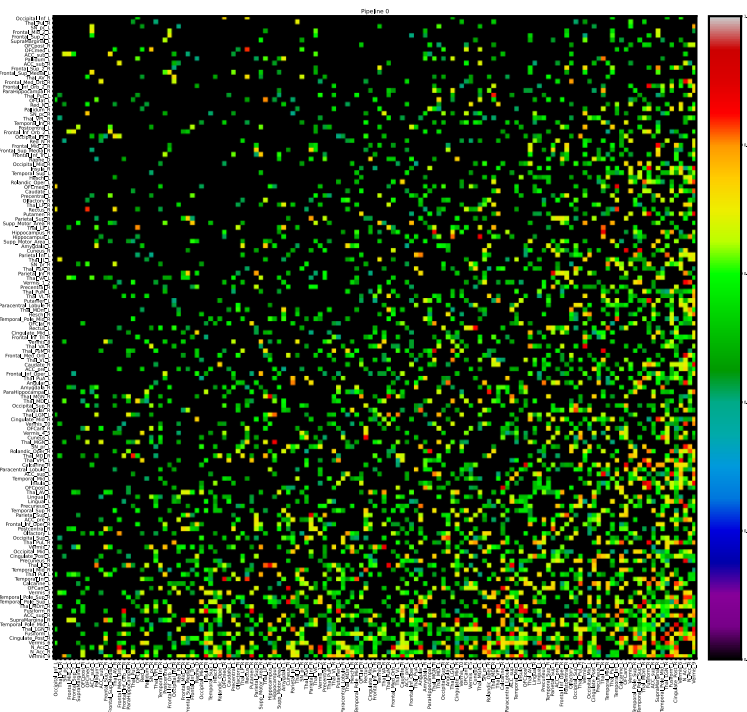


Figura 54 ICC sobre la métrica de conectividad funcional para el pipeline 0.

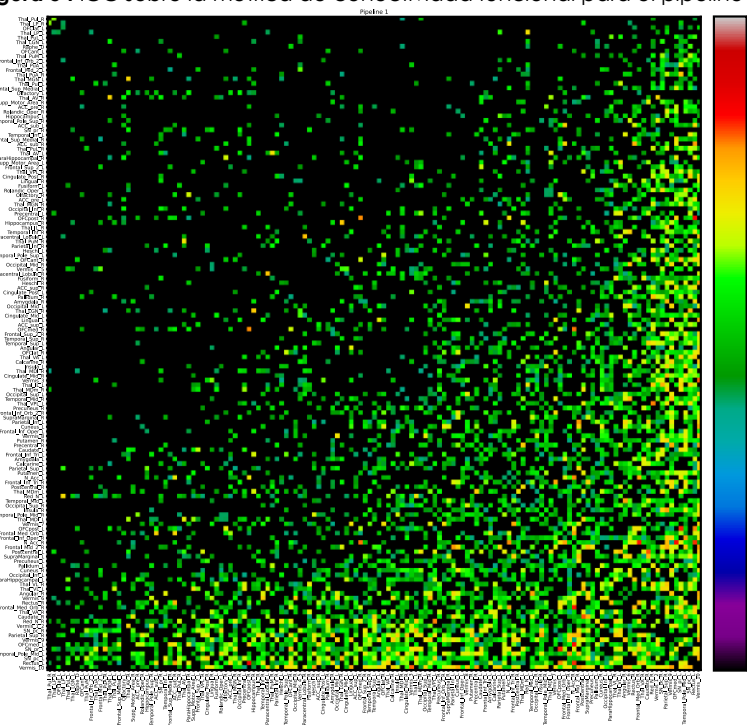


Figura 55 ICC sobre la métrica de conectividad funcional para el pipeline 1.

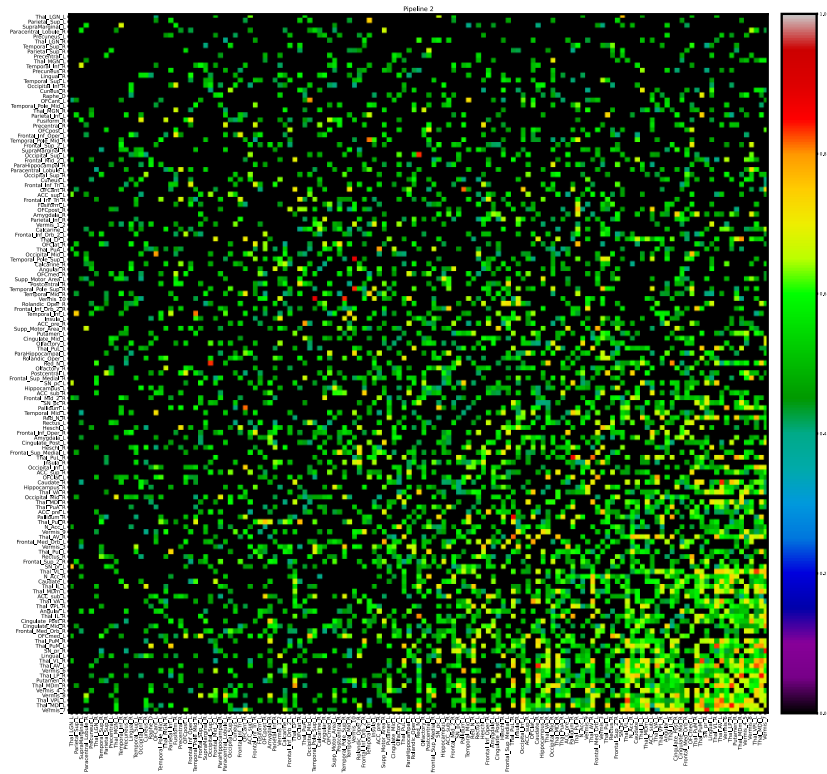


Figura 56 ICC sobre la métrica de conectividad funcional para el pipeline 2

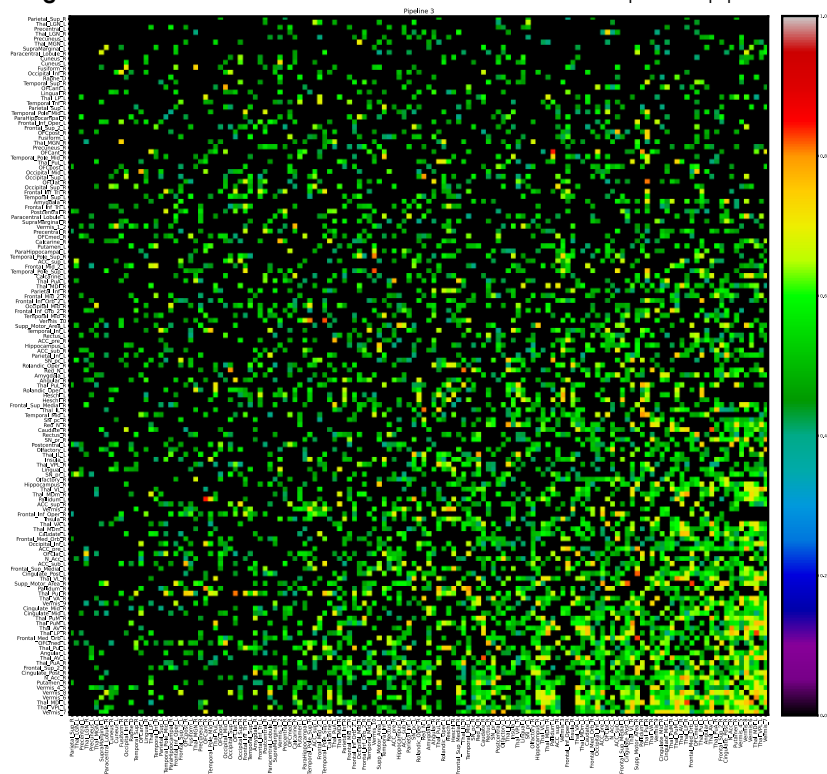


Figura 57 ICC sobre la métrica de conectividad funcional para el pipeline 3

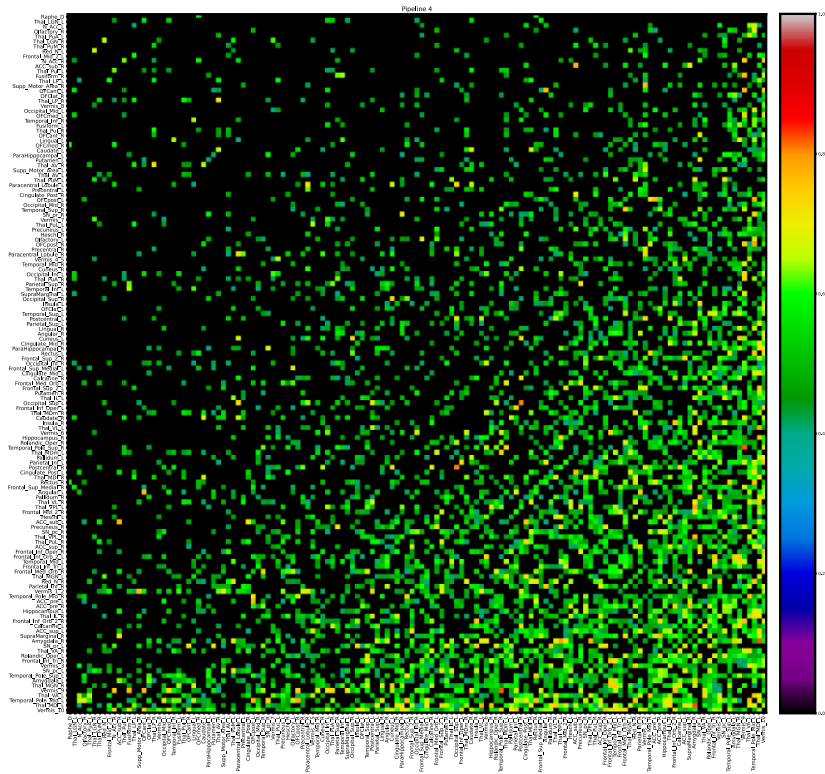


Figura 58 ICC sobre la métrica de conectividad funcional para el pipeline 4

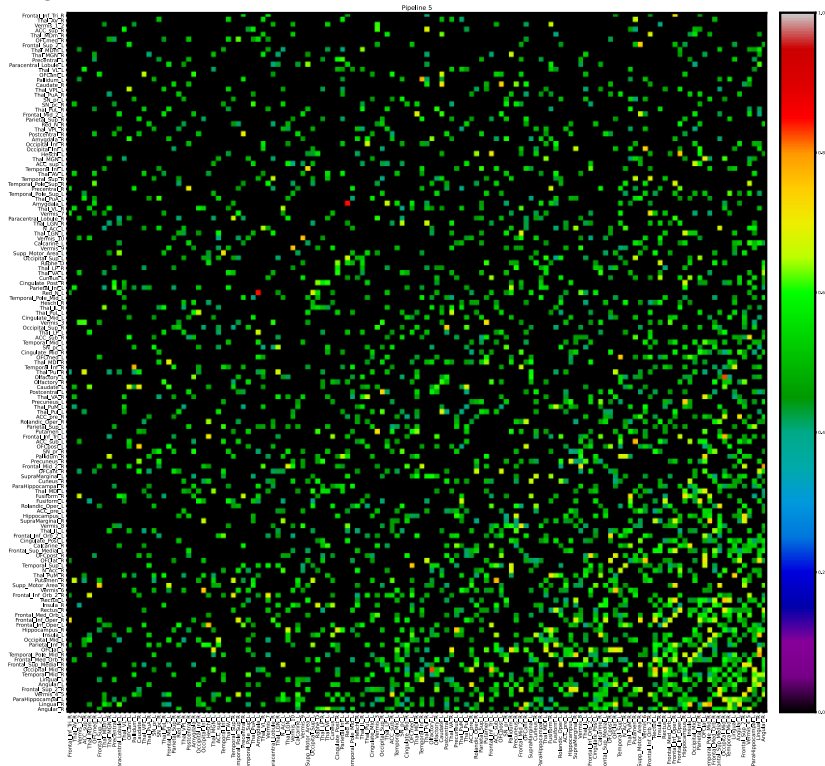


Figura 59 ICC sobre la métrica de conectividad funcional para el pipeline 5

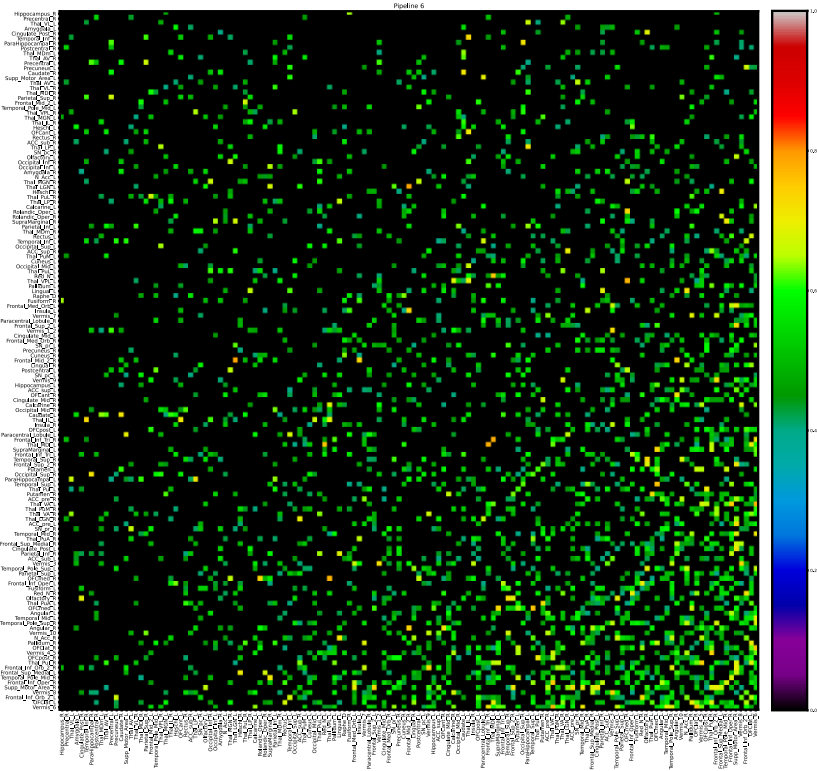


Figura 60 ICC sobre la métrica de conectividad funcional para el pipeline 6

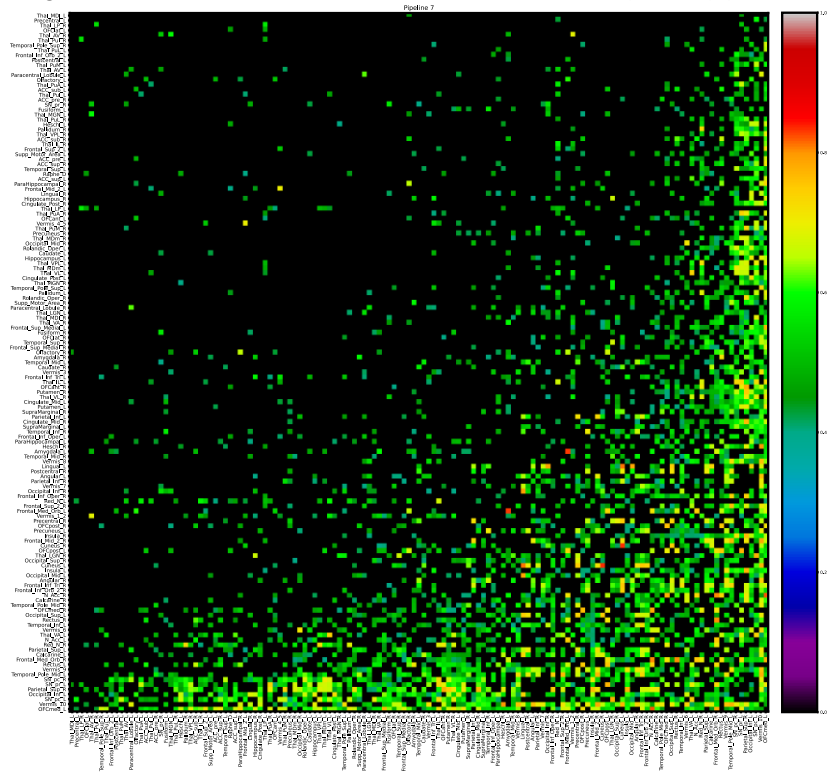


Figura 61 ICC sobre la métrica de conectividad funcional para el pipeline 7

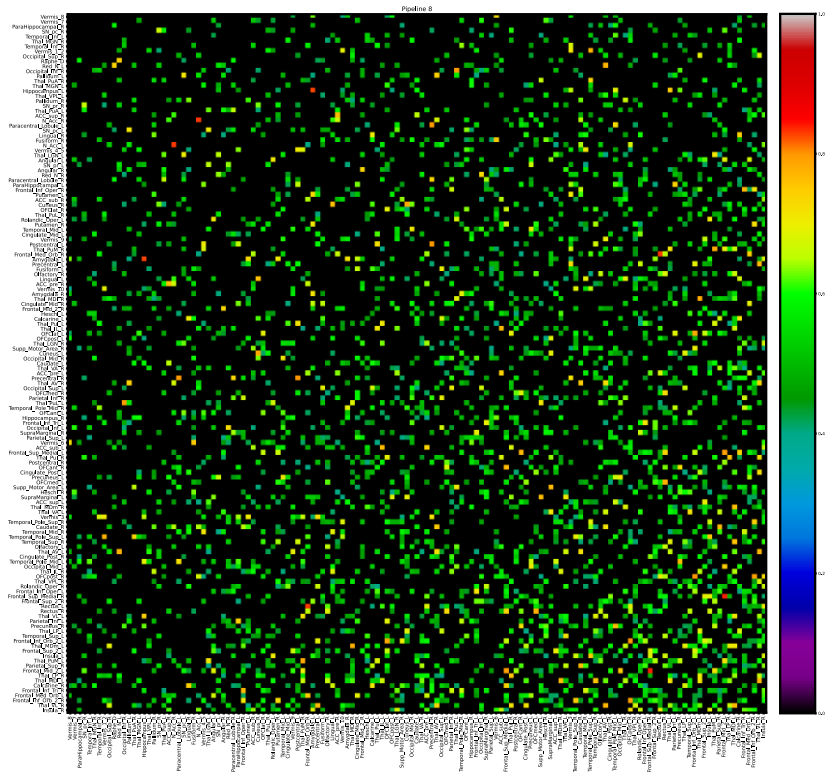


Figura 62 ICC sobre la métrica de conectividad funcional para el pipeline 8

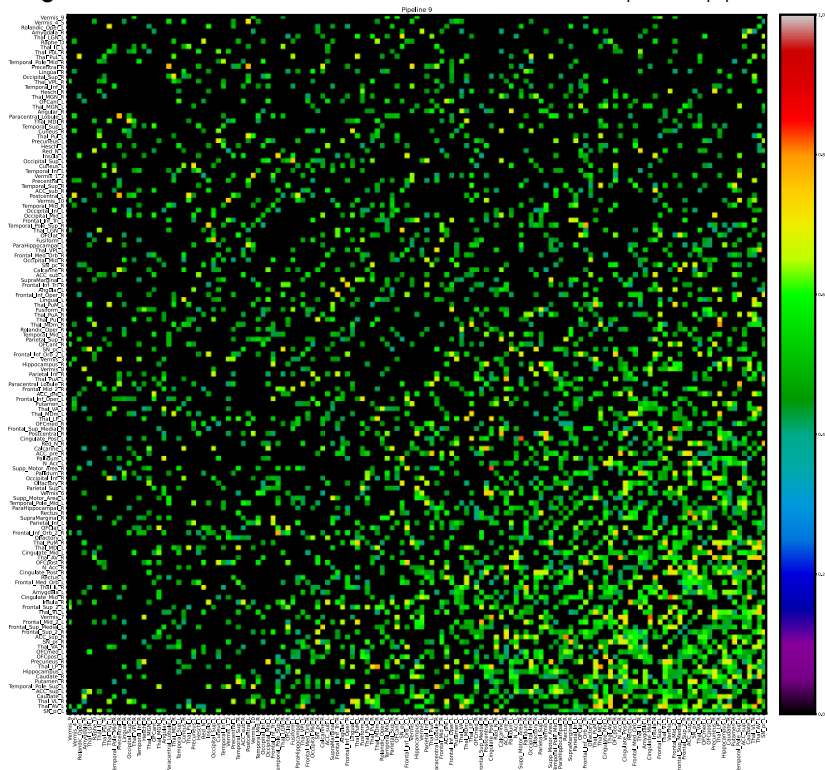


Figura 63 ICC sobre la métrica de conectividad funcional para el pipeline 9

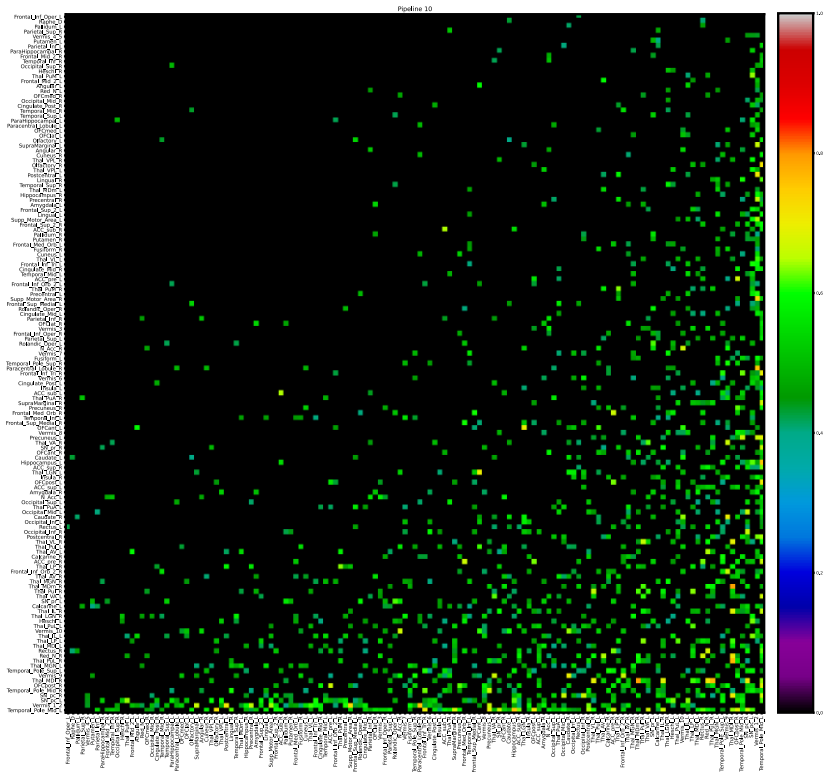


Figura 64 ICC sobre la métrica de conectividad funcional para el pipeline 10

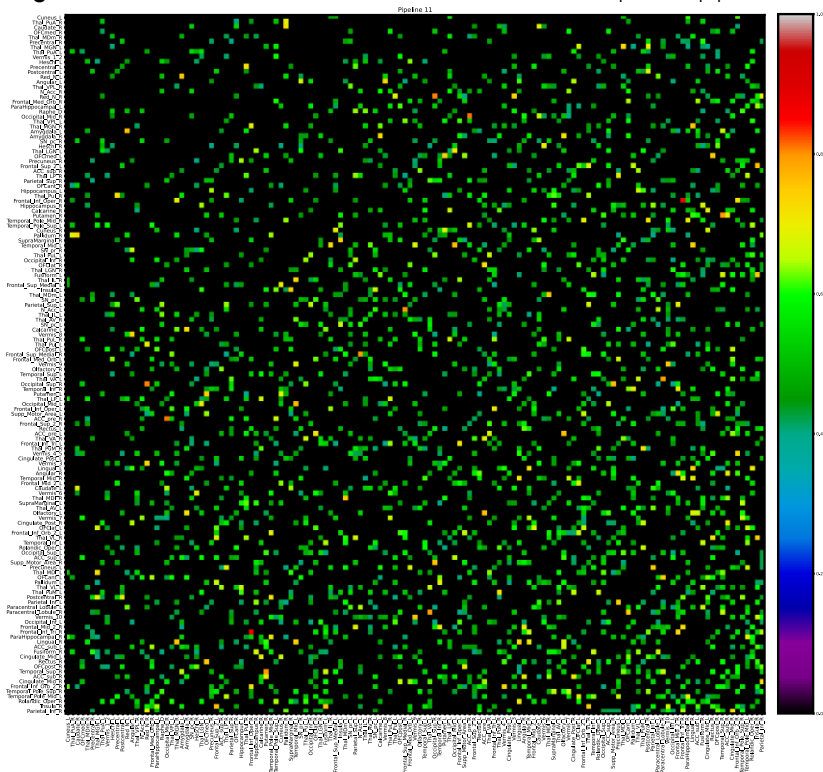


Figura 65 ICC sobre la métrica de conectividad funcional para el pipeline 11

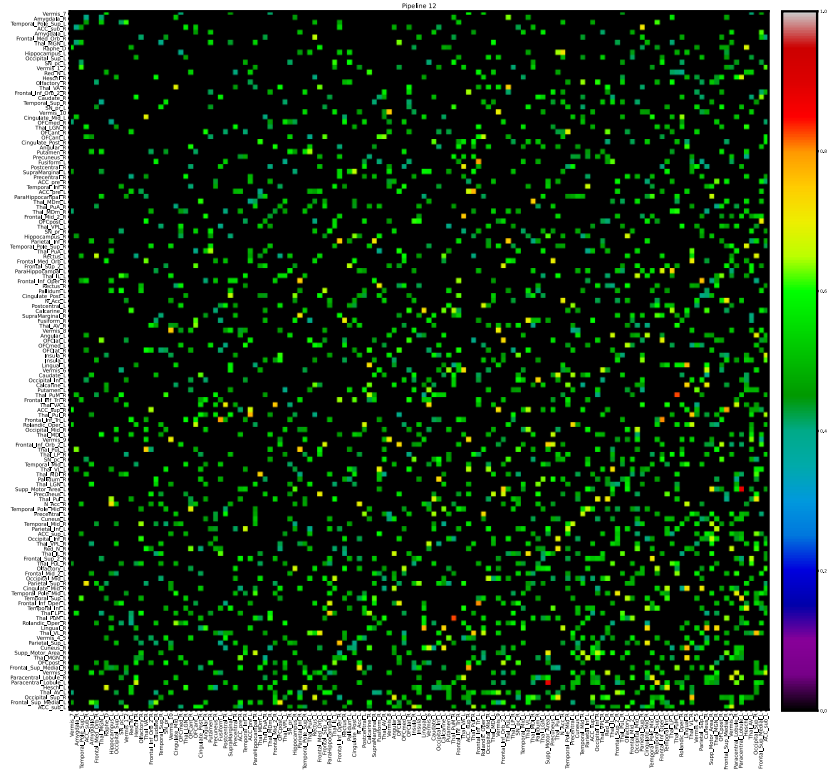


Figura 66 ICC sobre la métrica de conectividad funcional para el pipeline 12

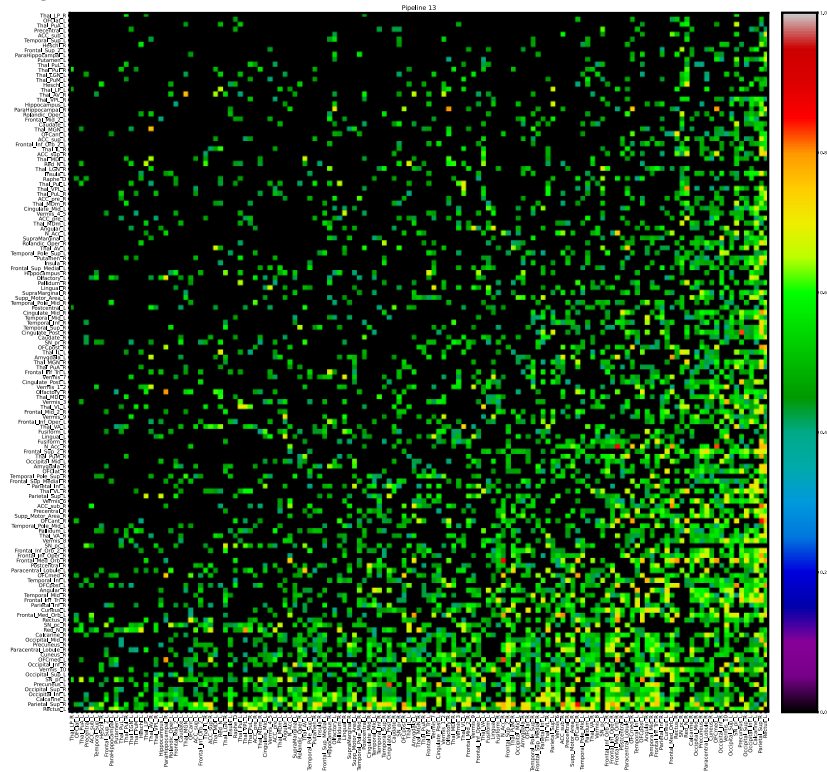


Figura 67 ICC sobre la métrica de conectividad funcional para el pipeline 13

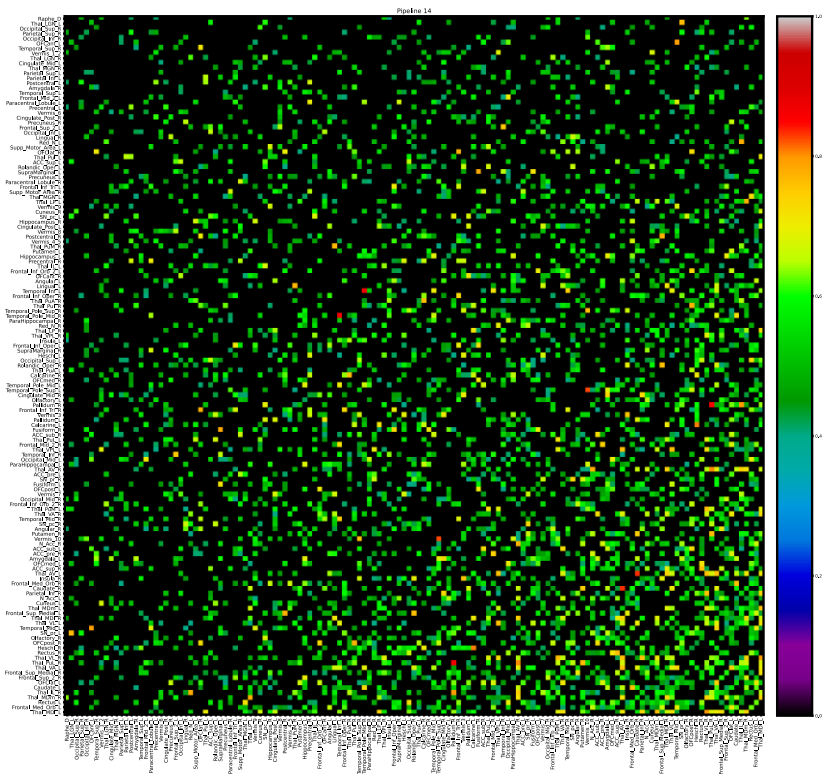


Figura 68 ICC sobre la métrica de conectividad funcional para el pipeline 14

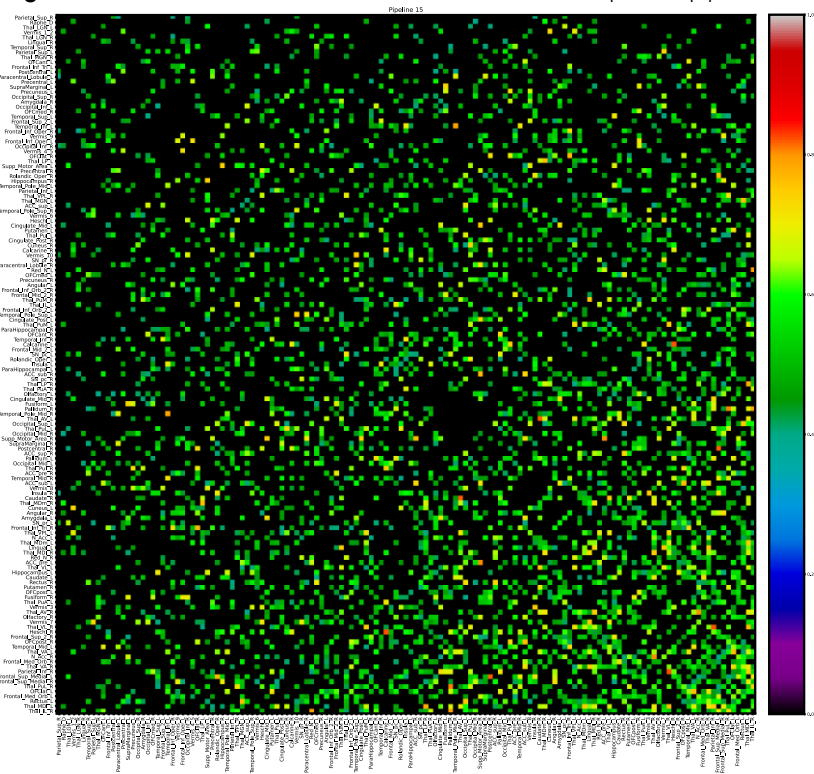


Figura 69 ICC sobre la métrica de conectividad funcional para el pipeline 15

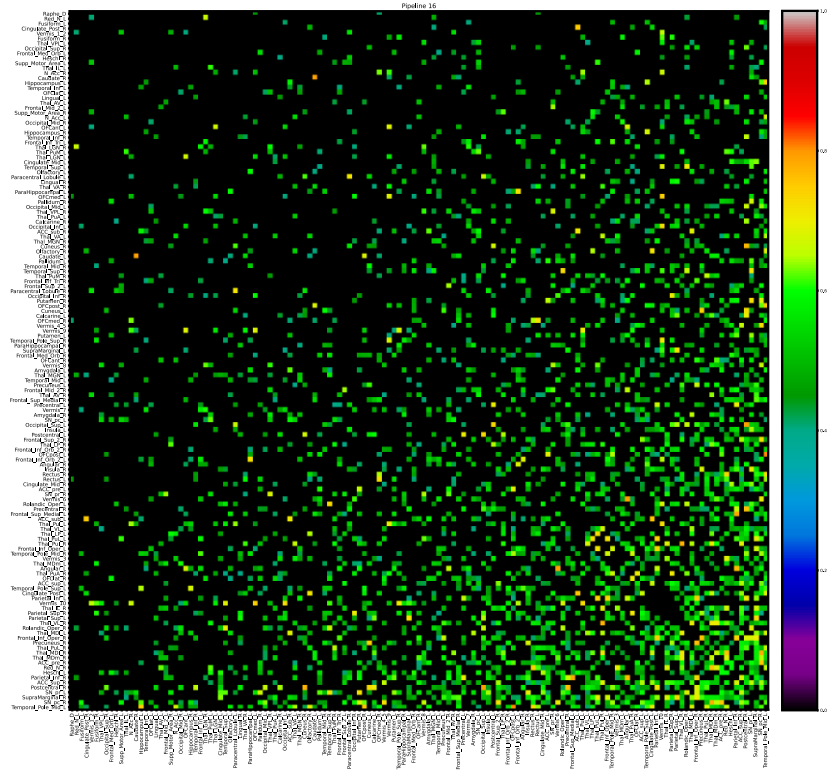


Figura 70 ICC sobre la métrica de conectividad funcional para el pipeline 16

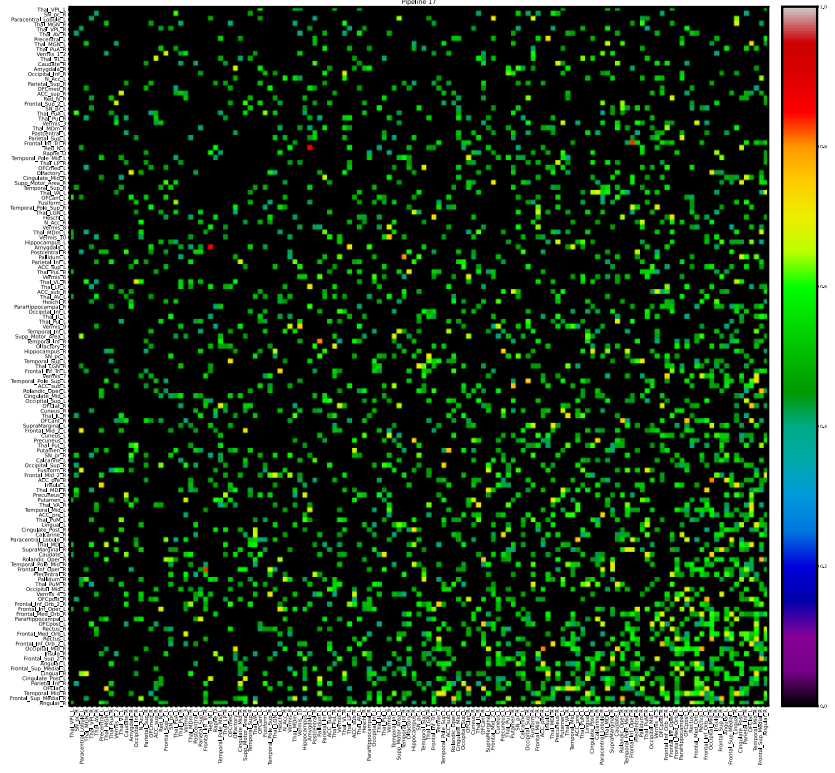


Figura 71 ICC sobre la métrica de conectividad funcional para el pipeline 17

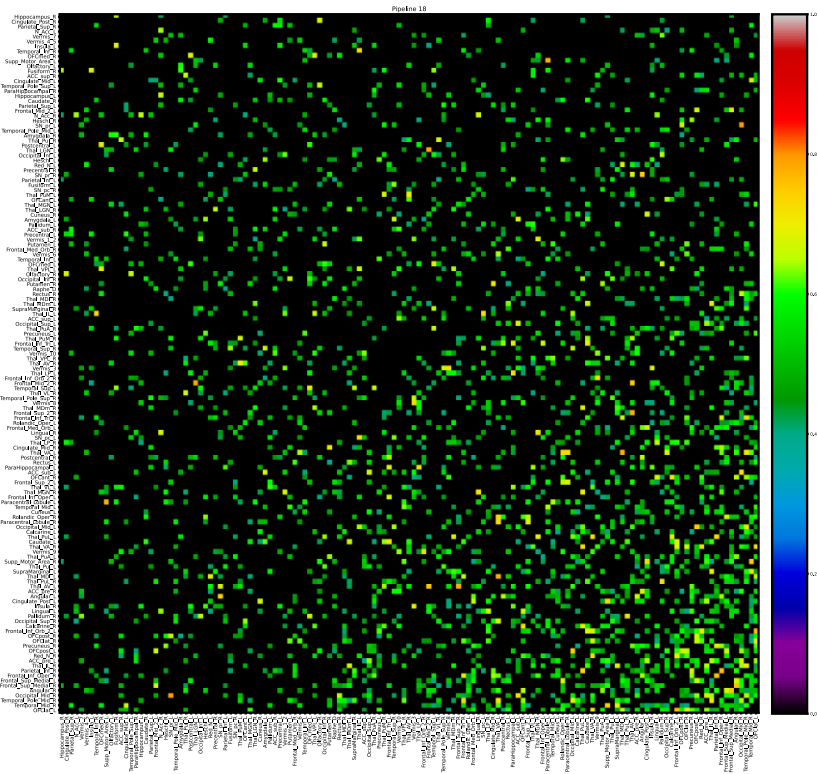


Figura 72 ICC sobre la métrica de conectividad funcional para el pipeline 18

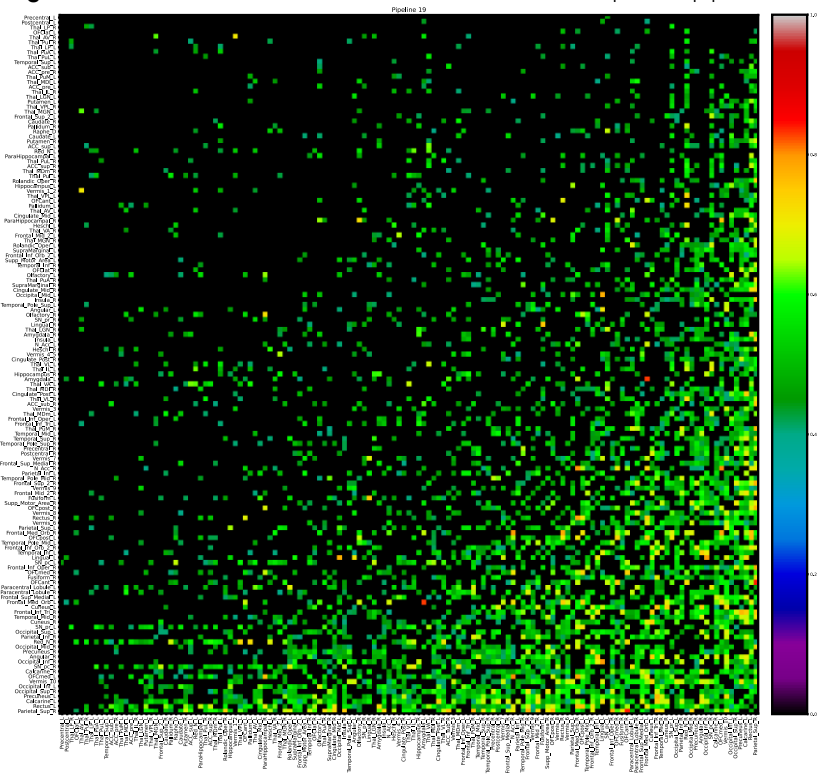


Figura 73 ICC sobre la métrica de conectividad funcional para el pipeline 19

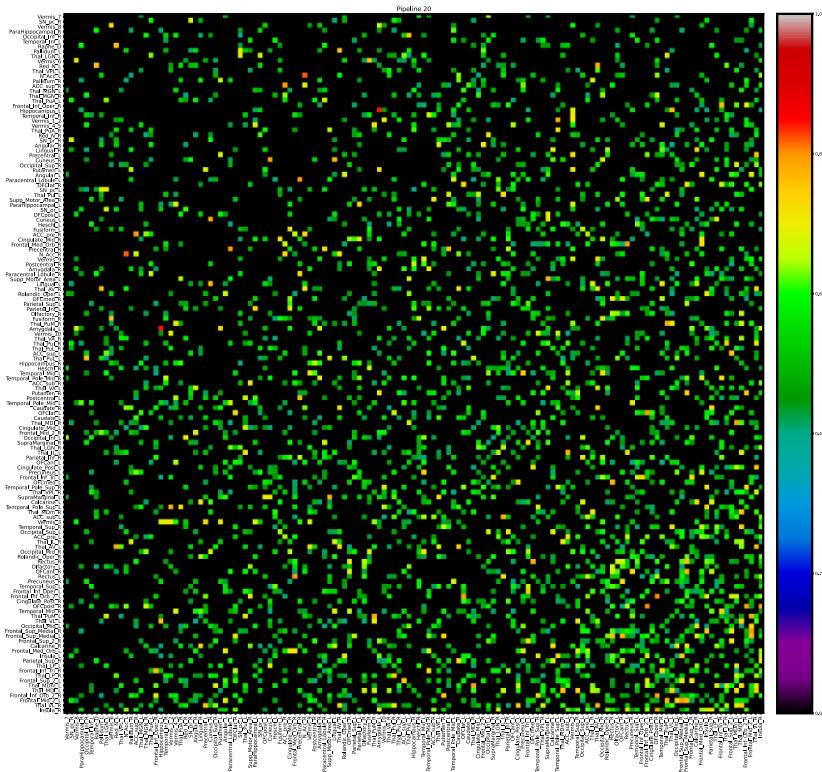


Figura 74 ICC sobre la métrica de conectividad funcional para el pipeline 20

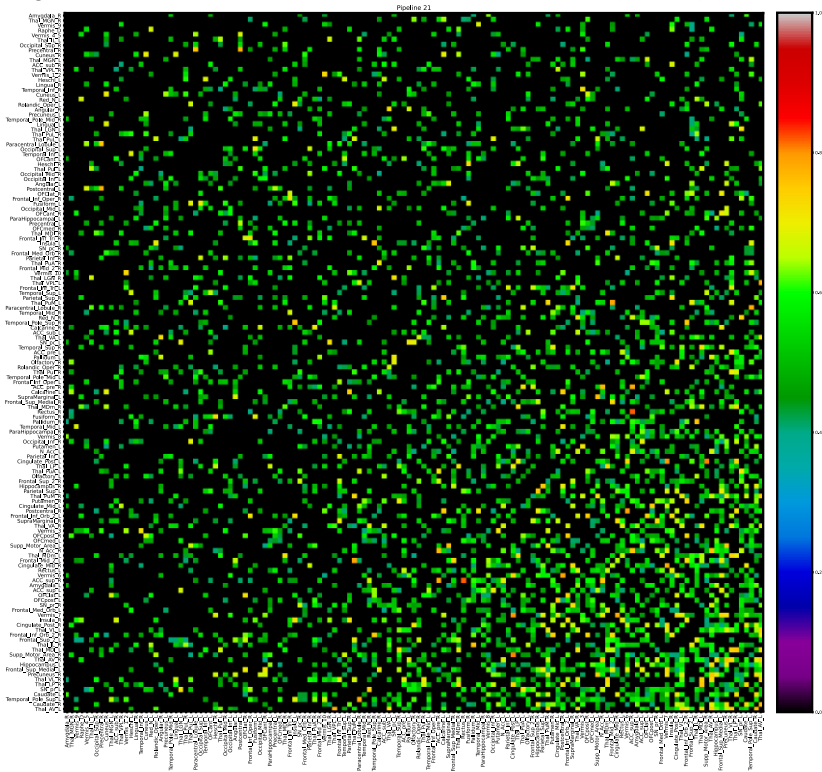


Figura 75 ICC sobre la métrica de conectividad funcional para el pipeline 21

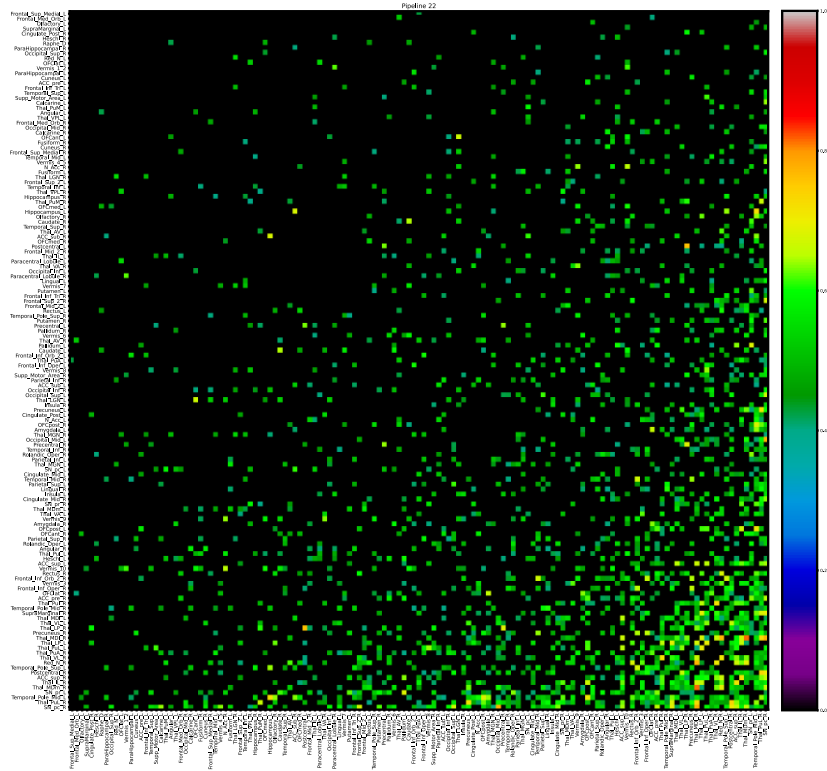


Figura 76 ICC sobre la métrica de conectividad funcional para el pipeline 22

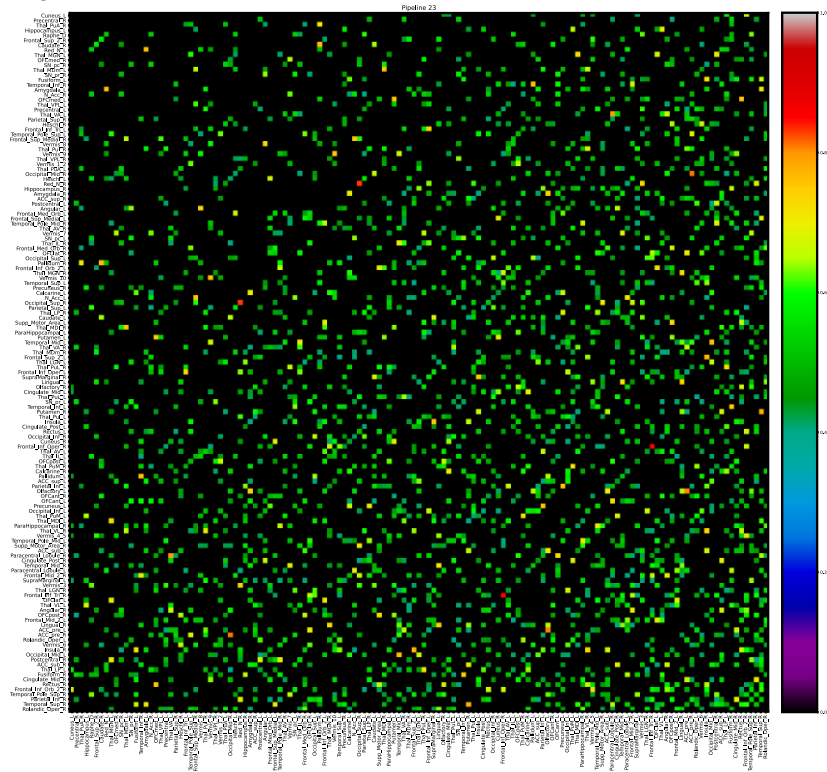


Figura 77 ICC sobre la métrica de conectividad funcional para el pipeline 23

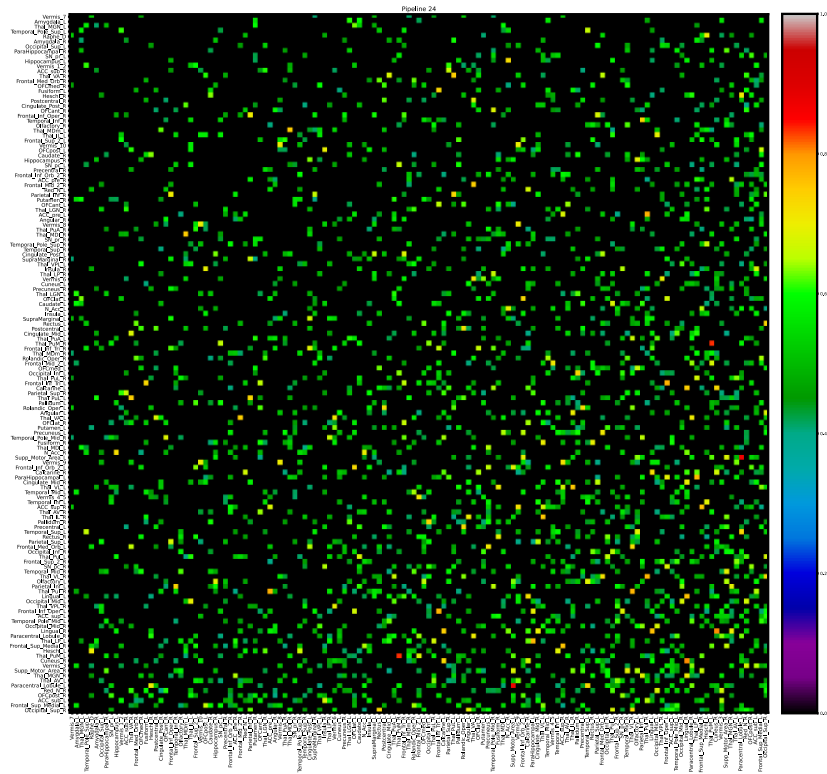


Figura 78 ICC sobre la métrica de conectividad funcional para el pipeline 24

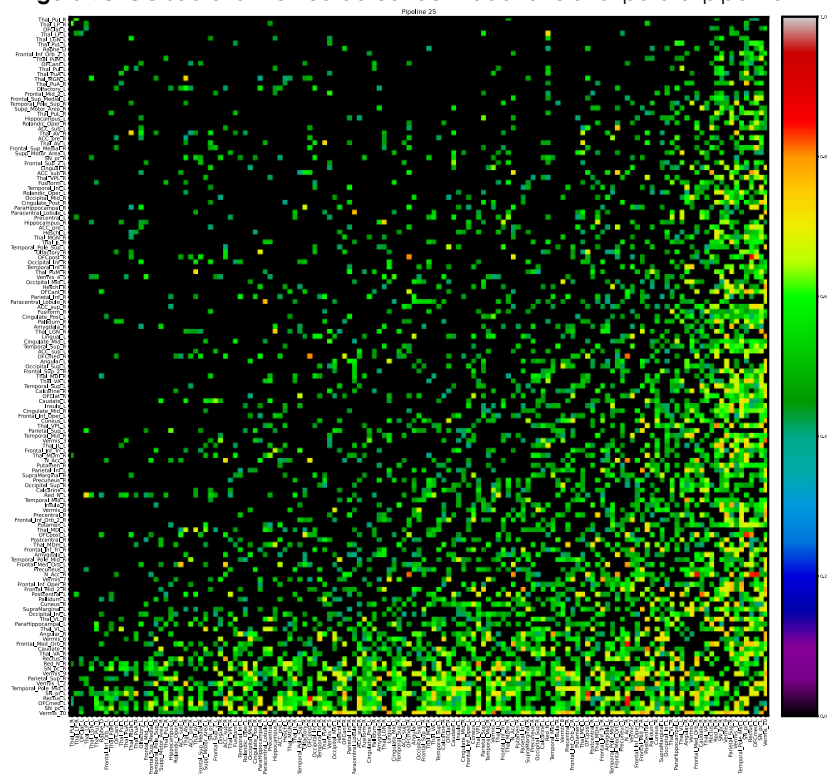


Figura 79 ICC sobre la métrica de conectividad funcional para el pipeline 25

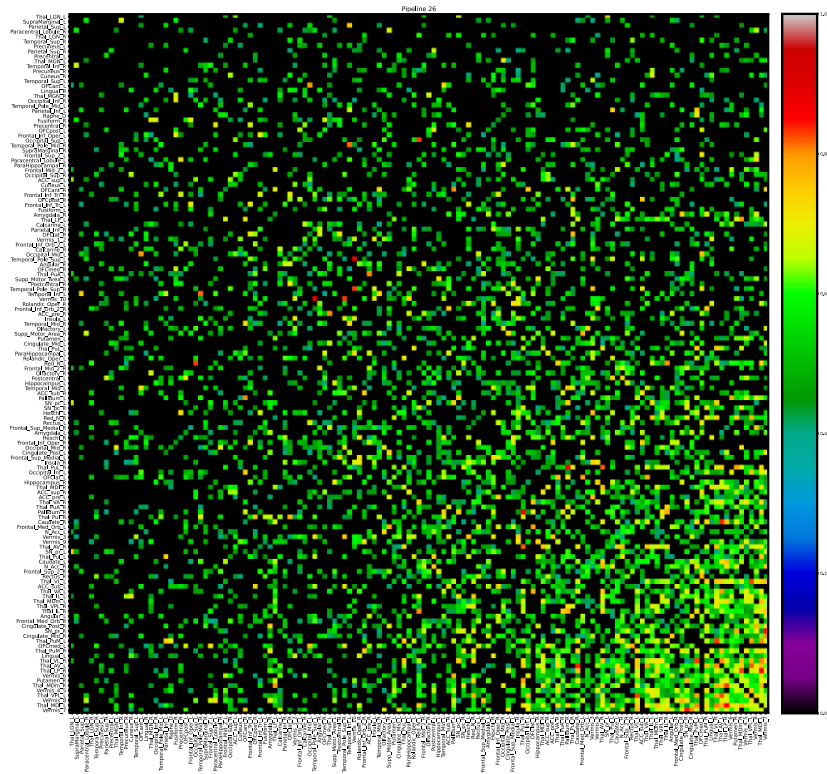


Figura 80 ICC sobre la métrica de conectividad funcional para el pipeline 26

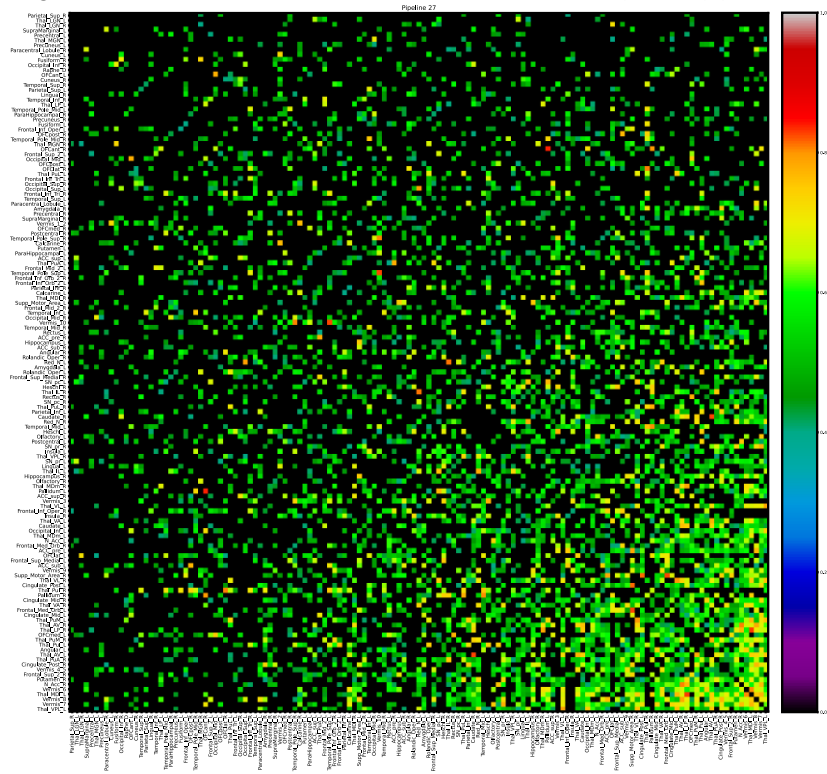


Figura 81 ICC sobre la métrica de conectividad funcional para el pipeline 27

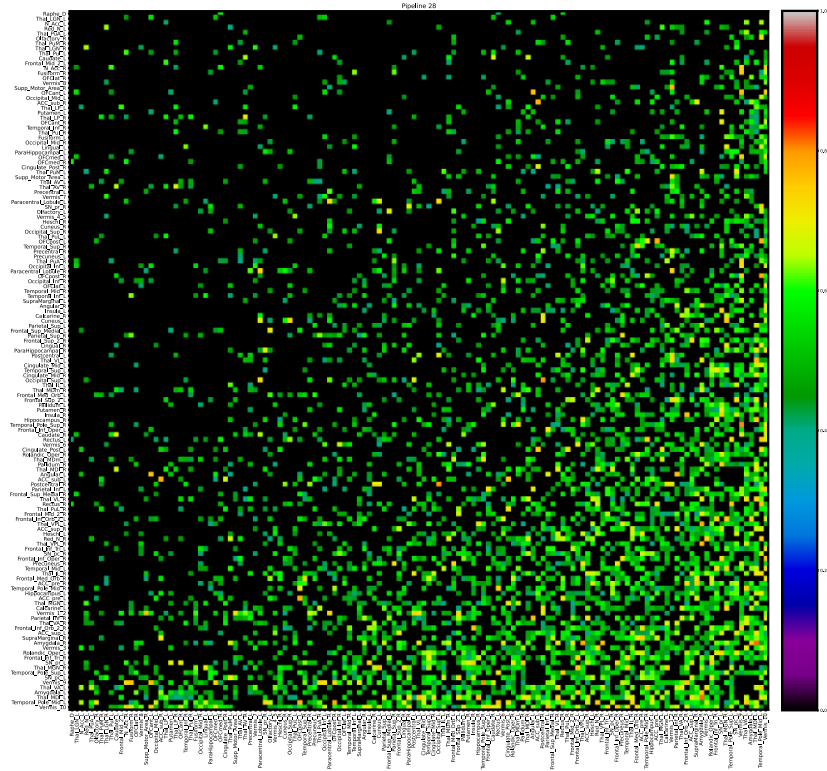


Figura 82 ICC sobre la métrica de conectividad funcional para el pipeline 28

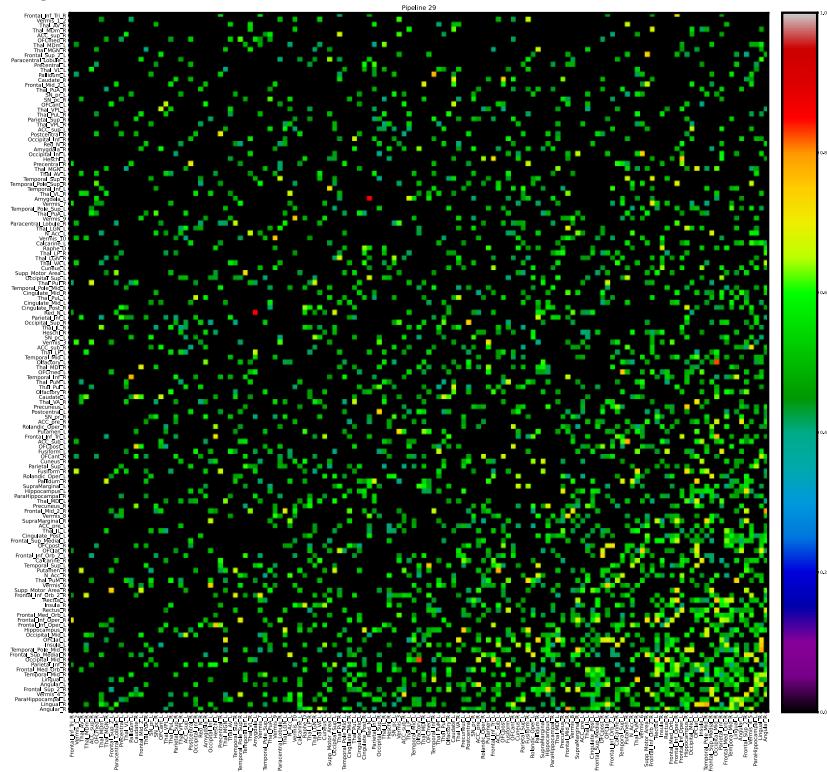


Figura 83 ICC sobre la métrica de conectividad funcional para el pipeline 29

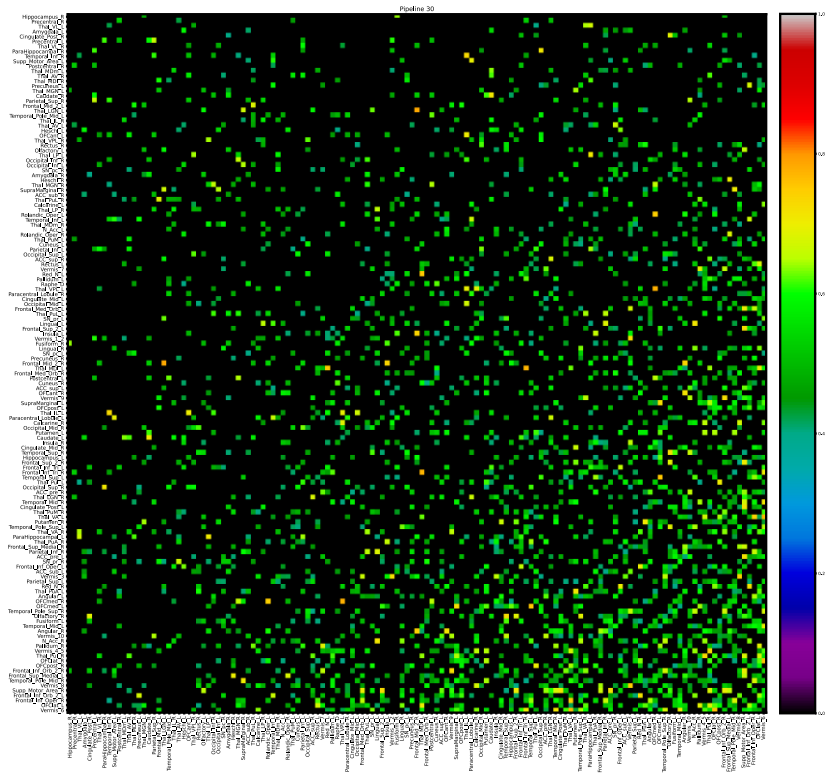


Figura 84 ICC sobre la métrica de conectividad funcional para el pipeline 30

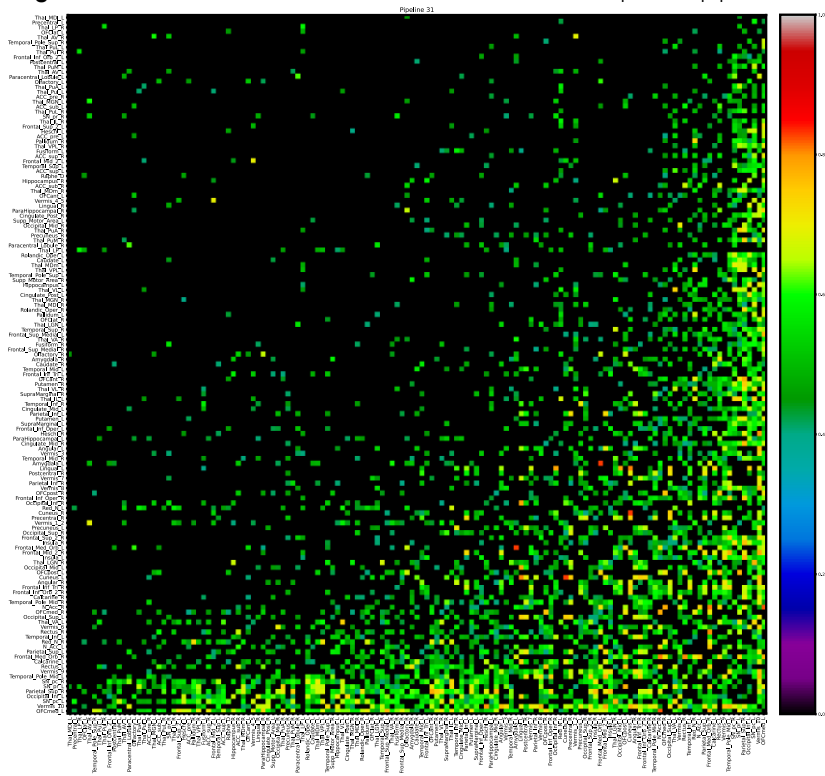


Figura 85 ICC sobre la métrica de conectividad funcional para el pipeline 31

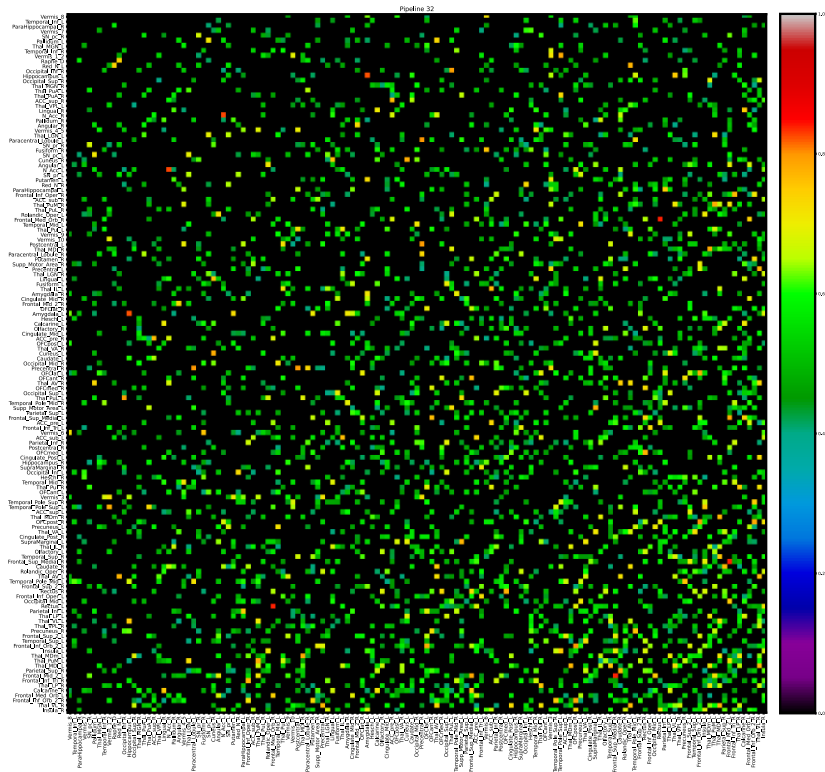


Figura 86 ICC sobre la métrica de conectividad funcional para el pipeline 32

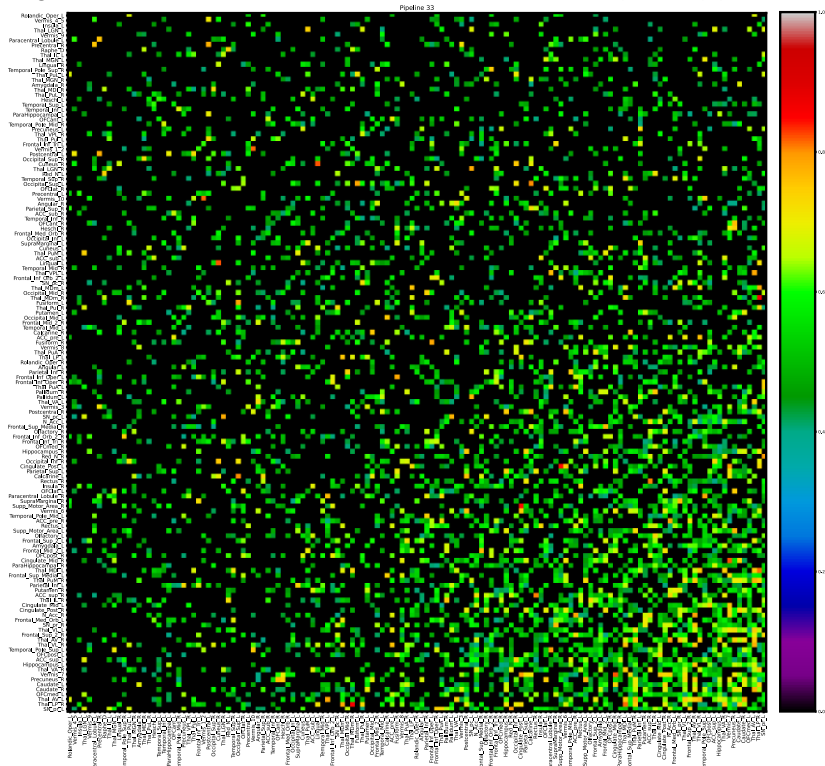


Figura 87 ICC sobre la métrica de conectividad funcional para el pipeline 33

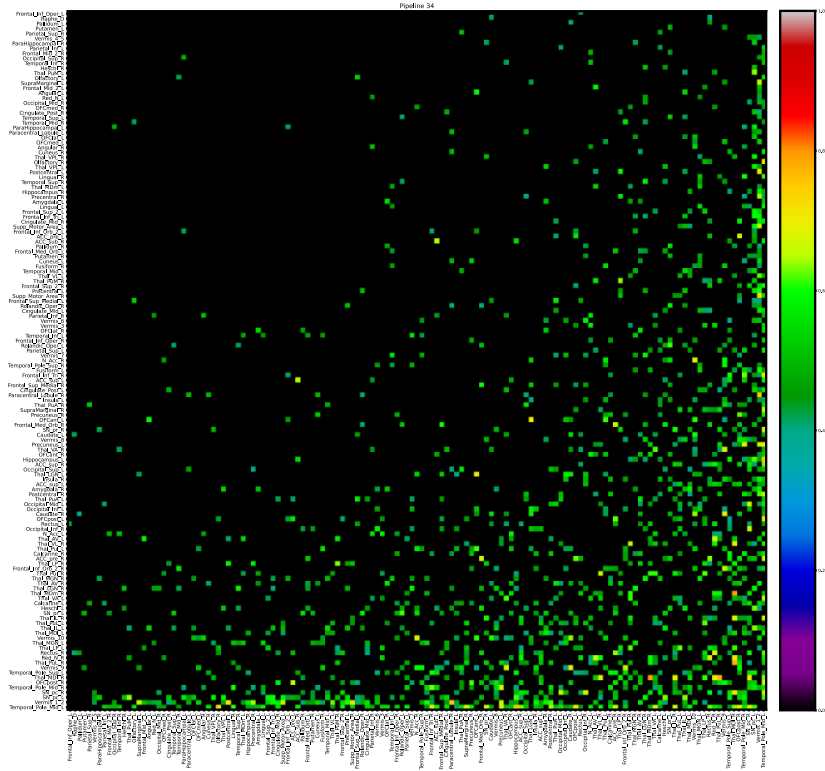


Figura 88 ICC sobre la métrica de conectividad funcional para el pipeline 34

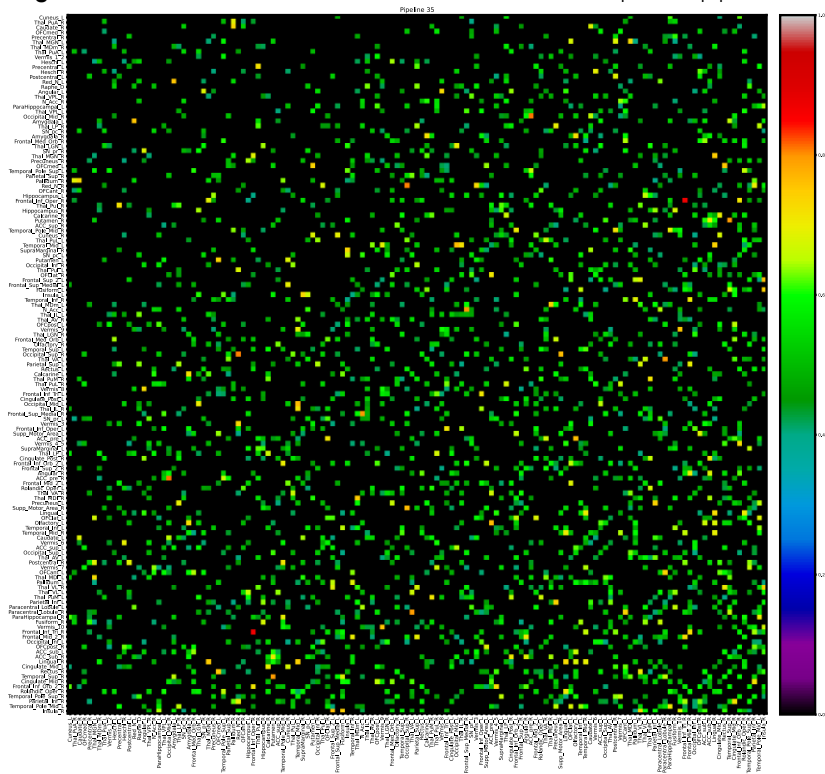


Figura 89 ICC sobre la métrica de conectividad funcional para el pipeline 35

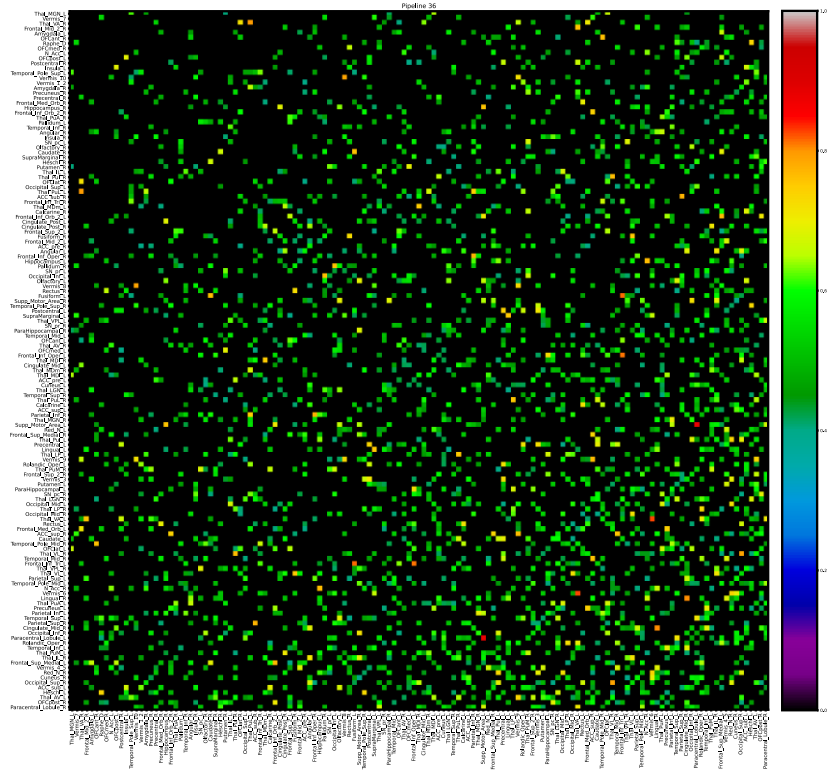


Figura 90 ICC sobre la métrica de conectividad funcional para el pipeline 36

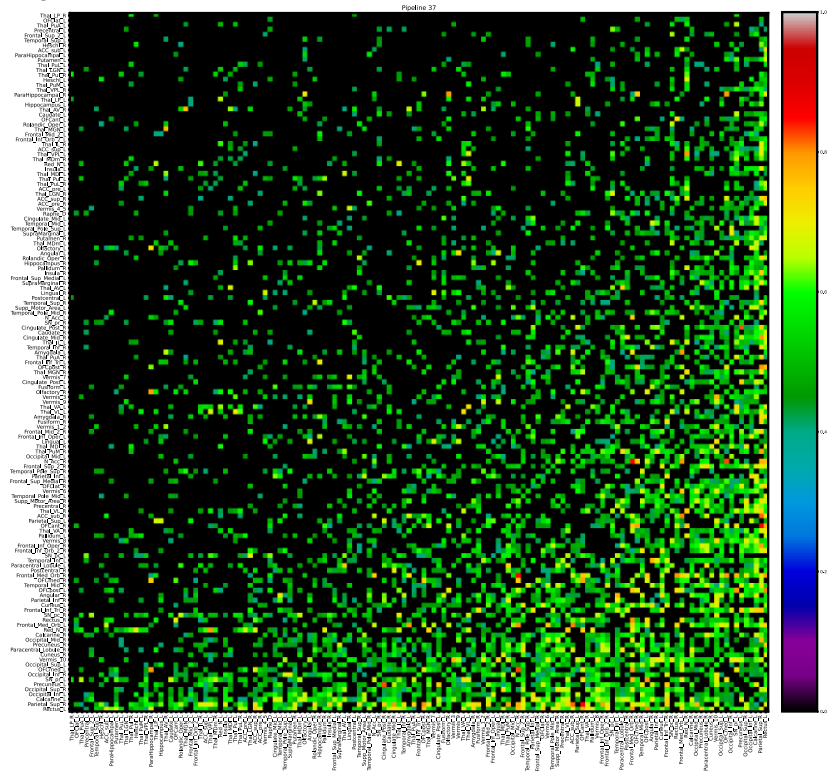


Figura 91 ICC sobre la métrica de conectividad funcional para el pipeline 37

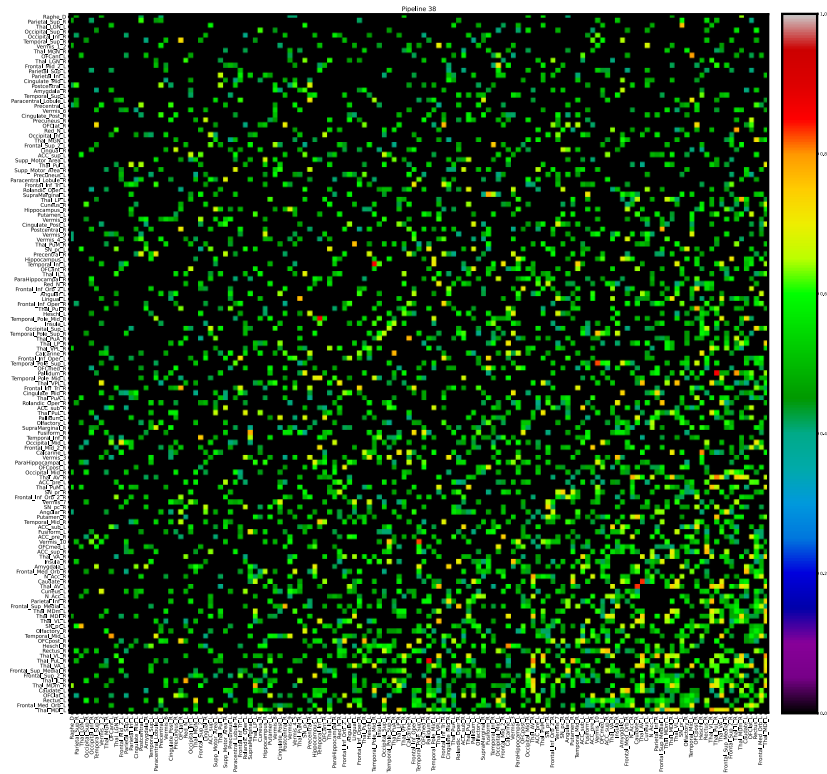


Figura 92 ICC sobre la métrica de conectividad funcional para el pipeline 38

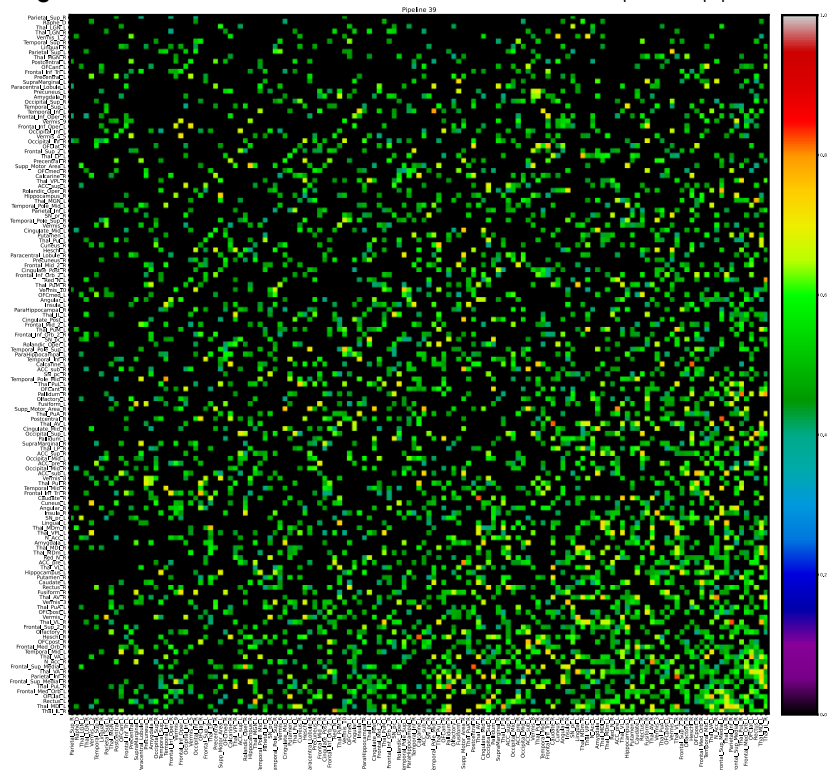


Figura 93 ICC sobre la métrica de conectividad funcional para el pipeline 39

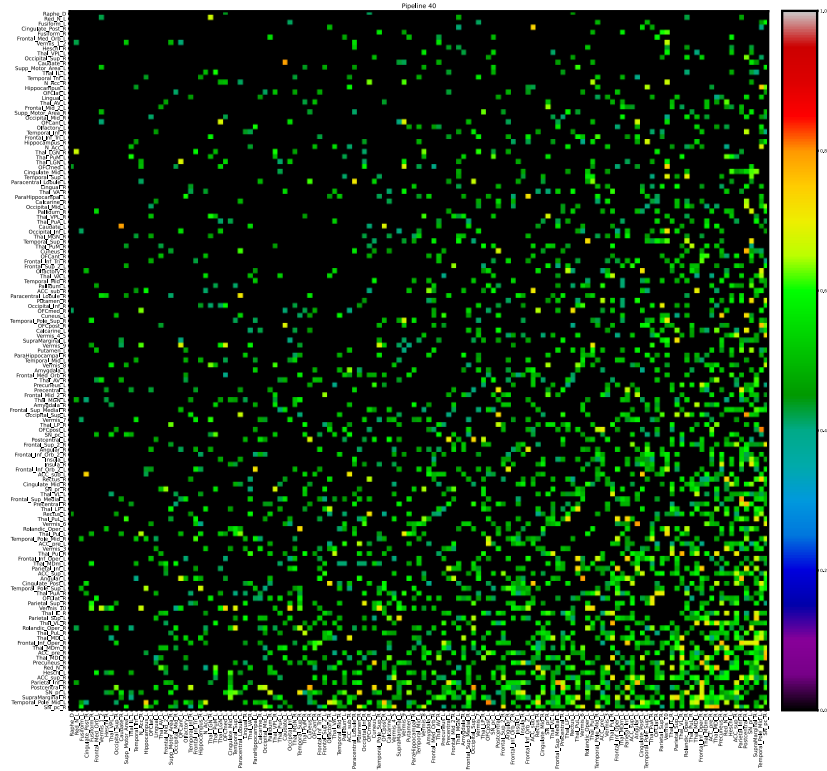


Figura 94 ICC sobre la métrica de conectividad funcional para el pipeline 40

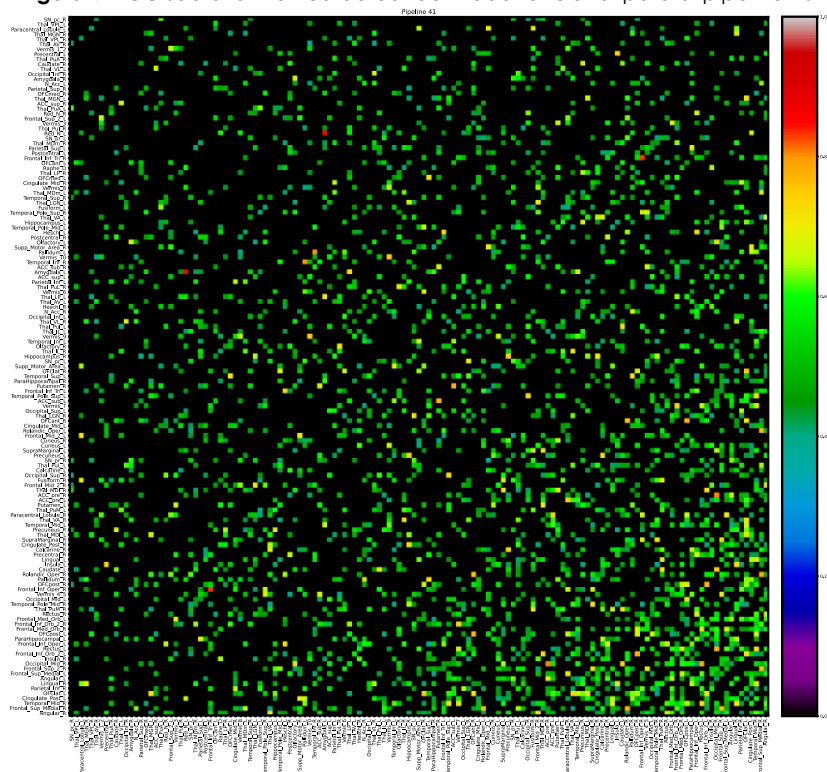


Figura 95 ICC sobre la métrica de conectividad funcional para el pipeline 41

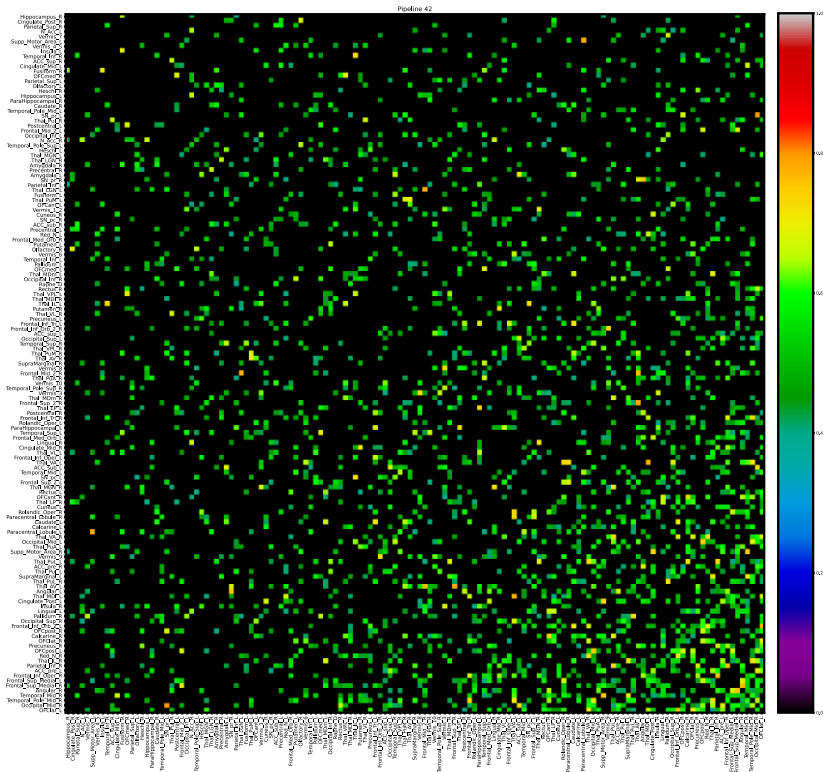


Figura 96 ICC sobre la métrica de conectividad funcional para el pipeline 42

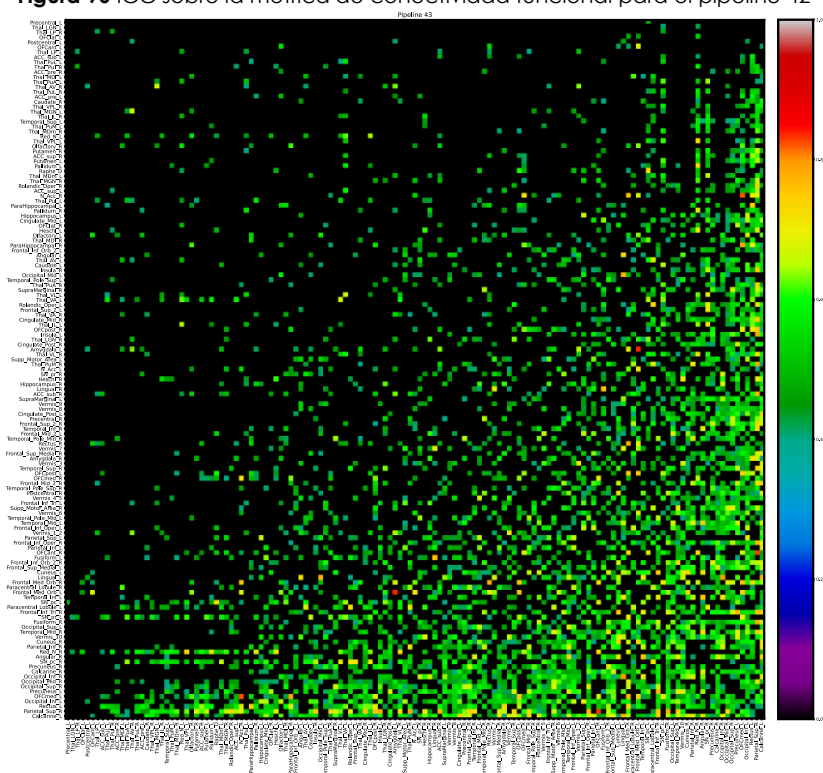


Figura 97 ICC sobre la métrica de conectividad funcional para el pipeline 43

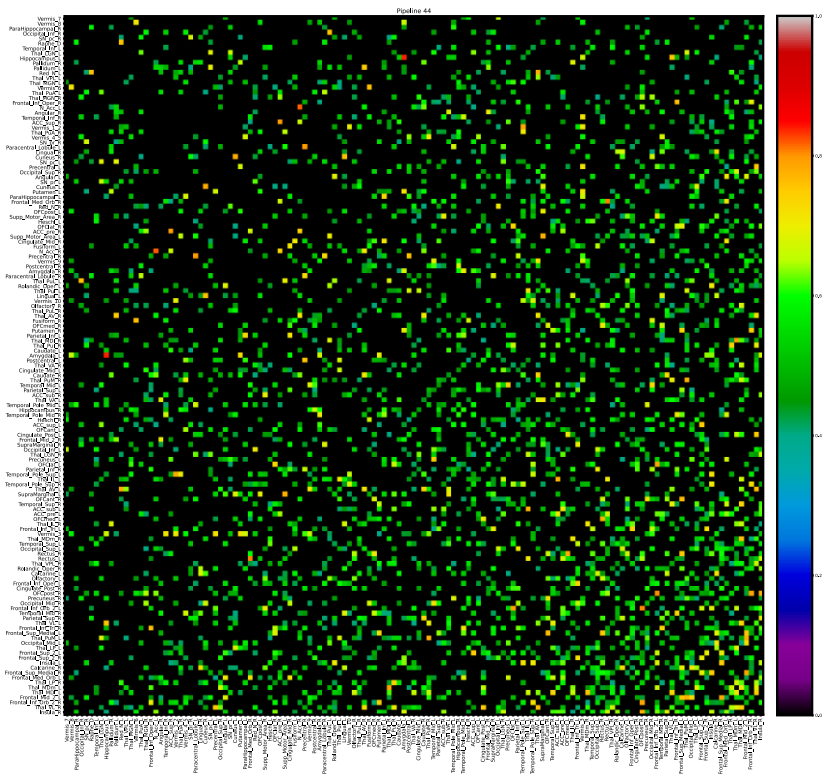


Figura 98 ICC sobre la métrica de conectividad funcional para el pipeline 44

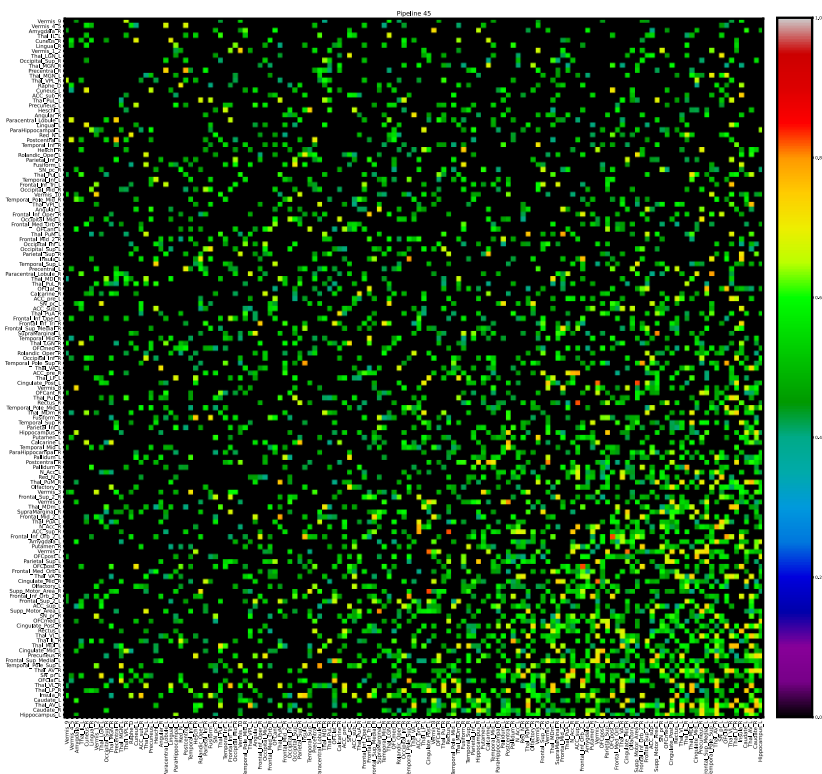


Figura 99 ICC sobre la métrica de conectividad funcional para el pipeline 45

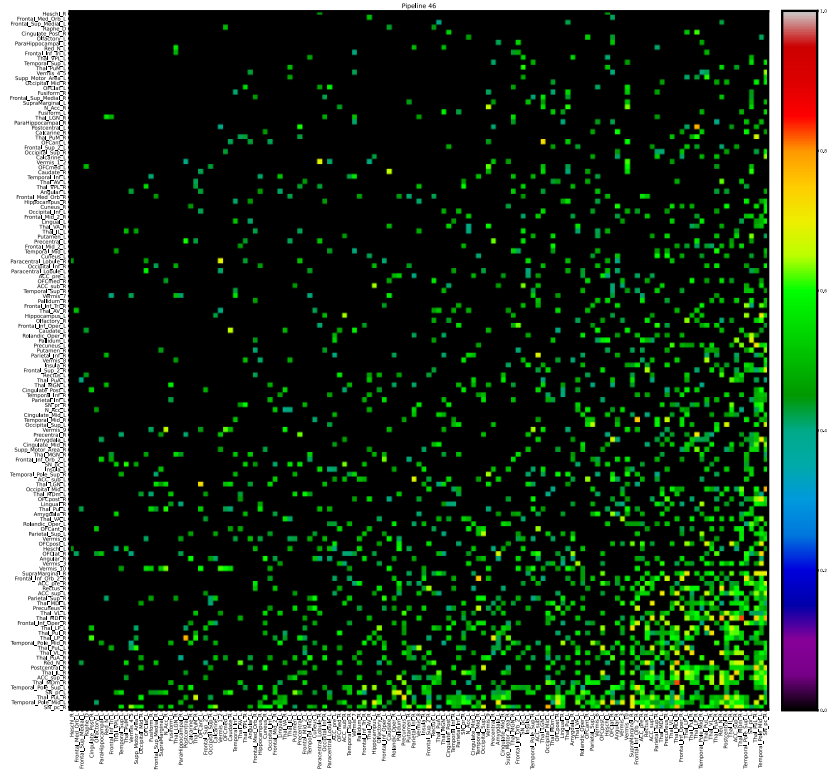


Figura 100 ICC sobre la métrica de conectividad funcional para el pipeline 46

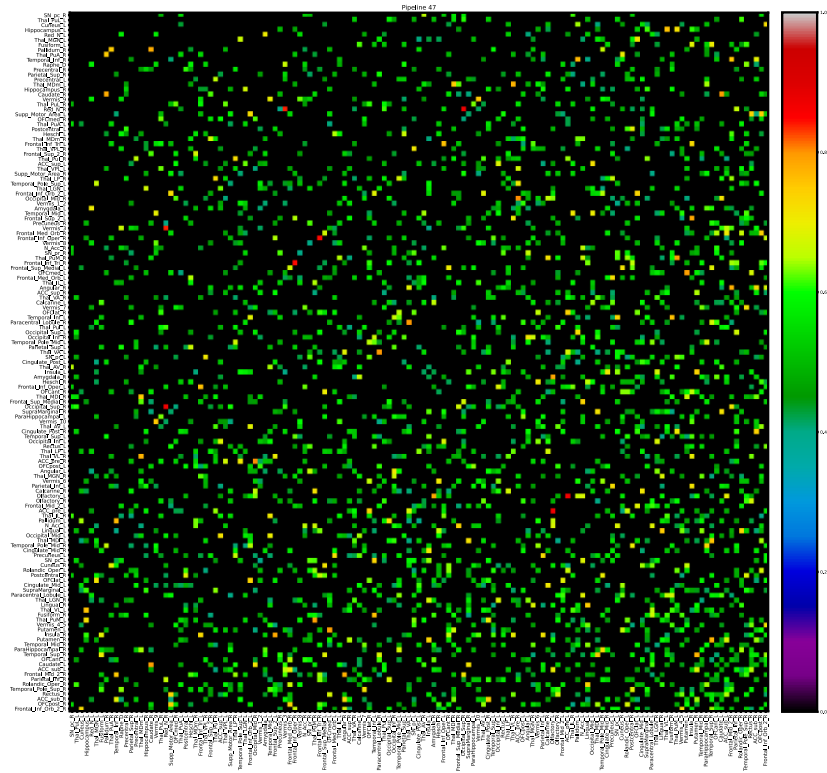


Figura 101 ICC sobre la métrica de conectividad funcional para el pipeline 47

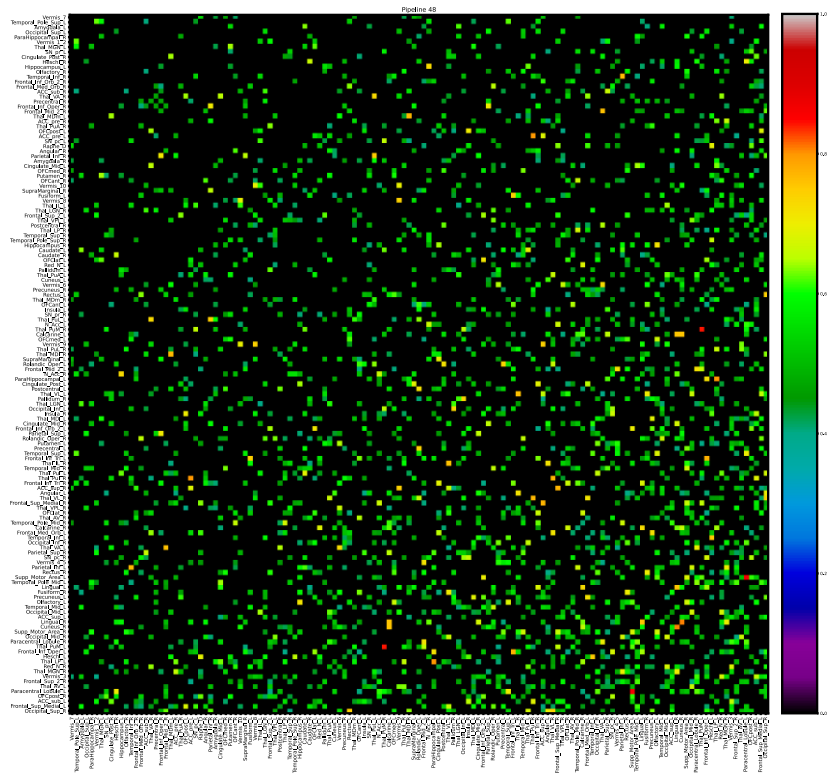


Figura 102 ICC sobre la métrica de conectividad funcional para el pipeline 48

B 6. ICC sobre las métricas de grafos globales en todos los pipelines

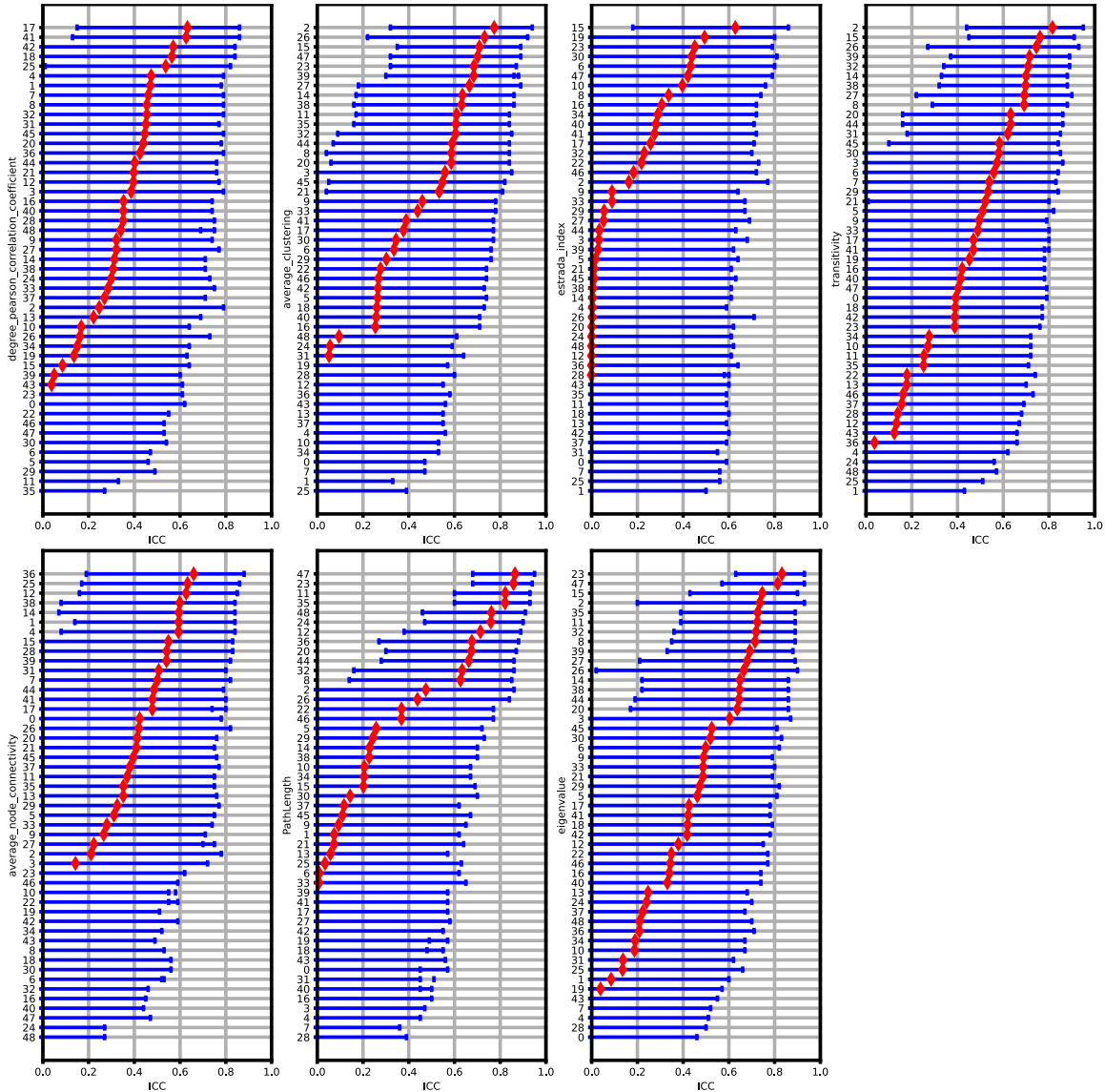


Figura 103 ICC sobre las métricas de: grado de asortatividad con el coeficiente de correlación de Pearson, agrupamiento promedio, índice de Estrada, transividad, conectividad promedio por nodo, camino característico y valor propio más grande de la matriz de adyacencia para todos los pelines.

B 7. ICC sobre las métricas de grafos por región en todos los pipelines

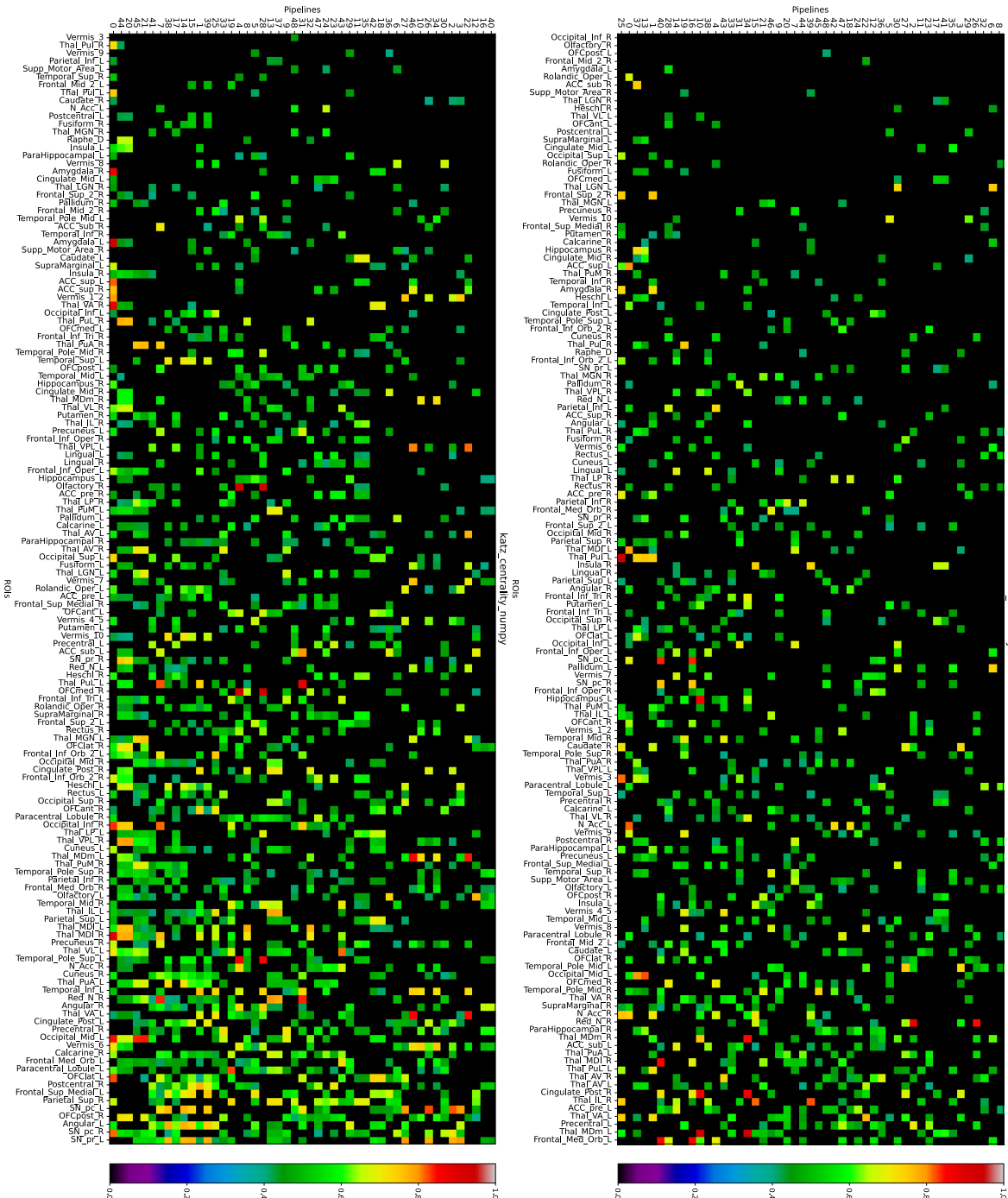


Figura 104 ICC sobre las métricas de centralidad de Katz e intermediación en las diferentes regiones para todos los pipelines.

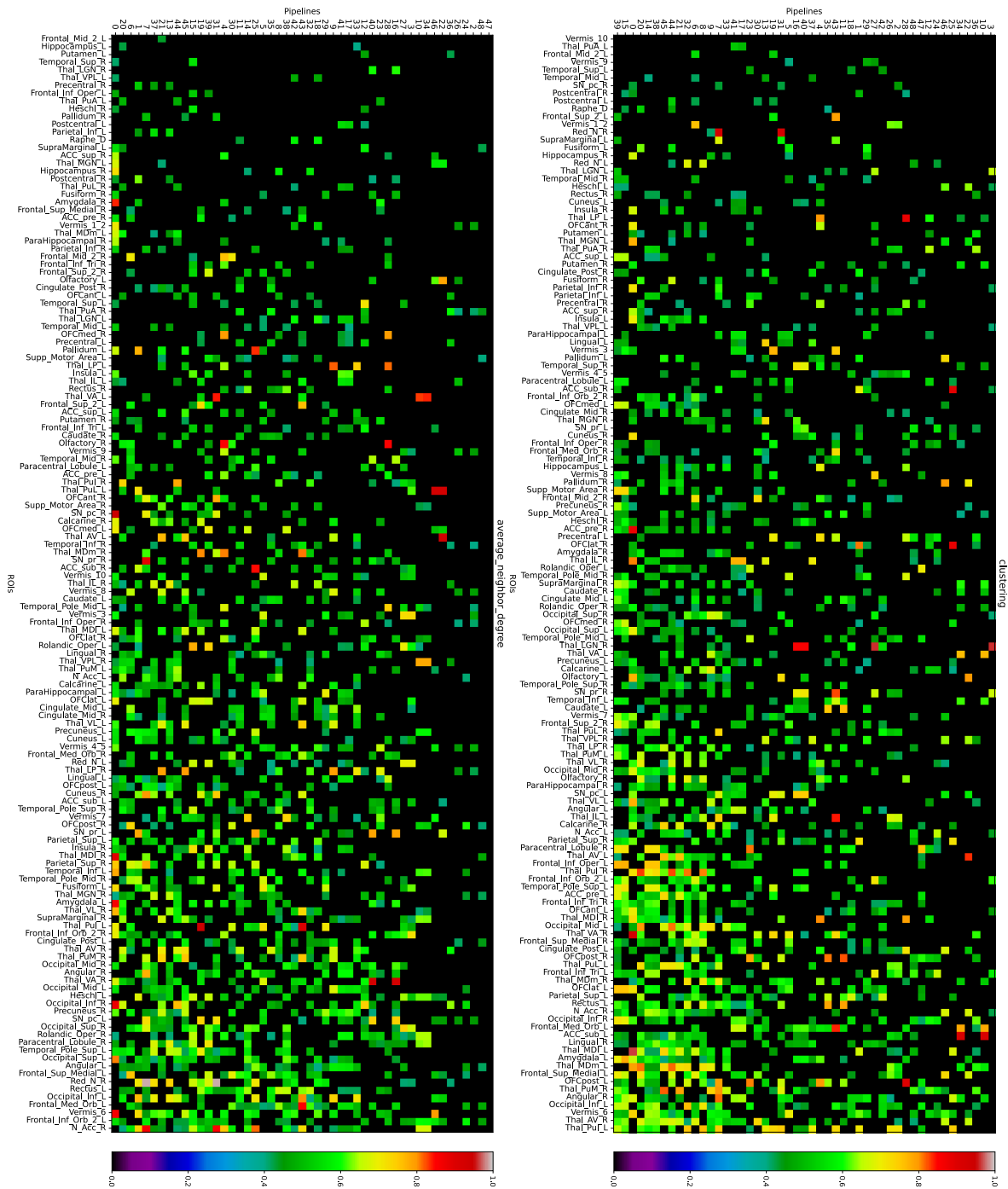


Figura 105 ICC sobre las métricas de grado promedio de los vecinos y agrupamiento en las diferentes regiones para todos los pipelines.

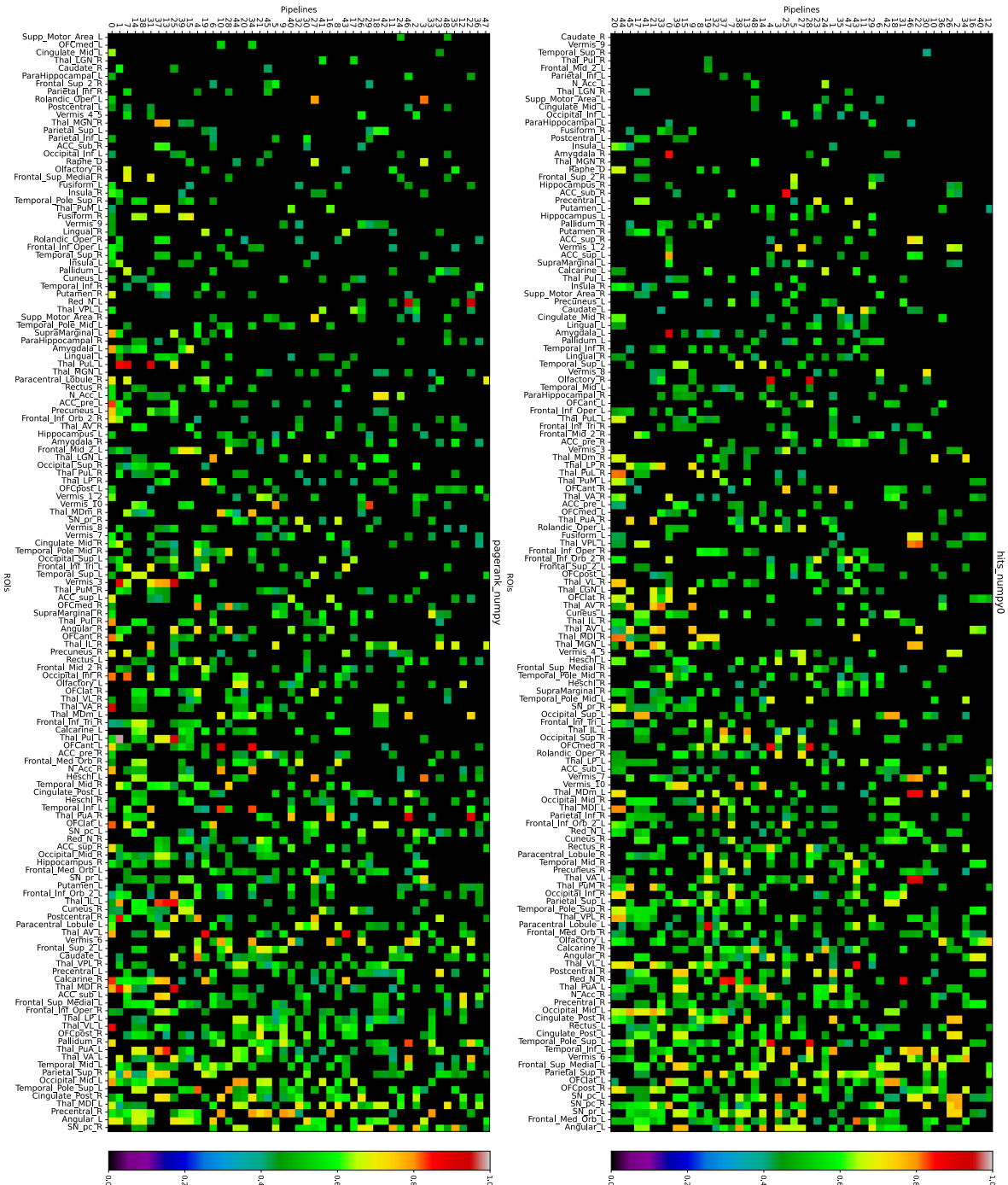


Figura 106 ICC por las métricas de grado de página e hits de autoridad en las diferentes regiones para todos los pipelines.

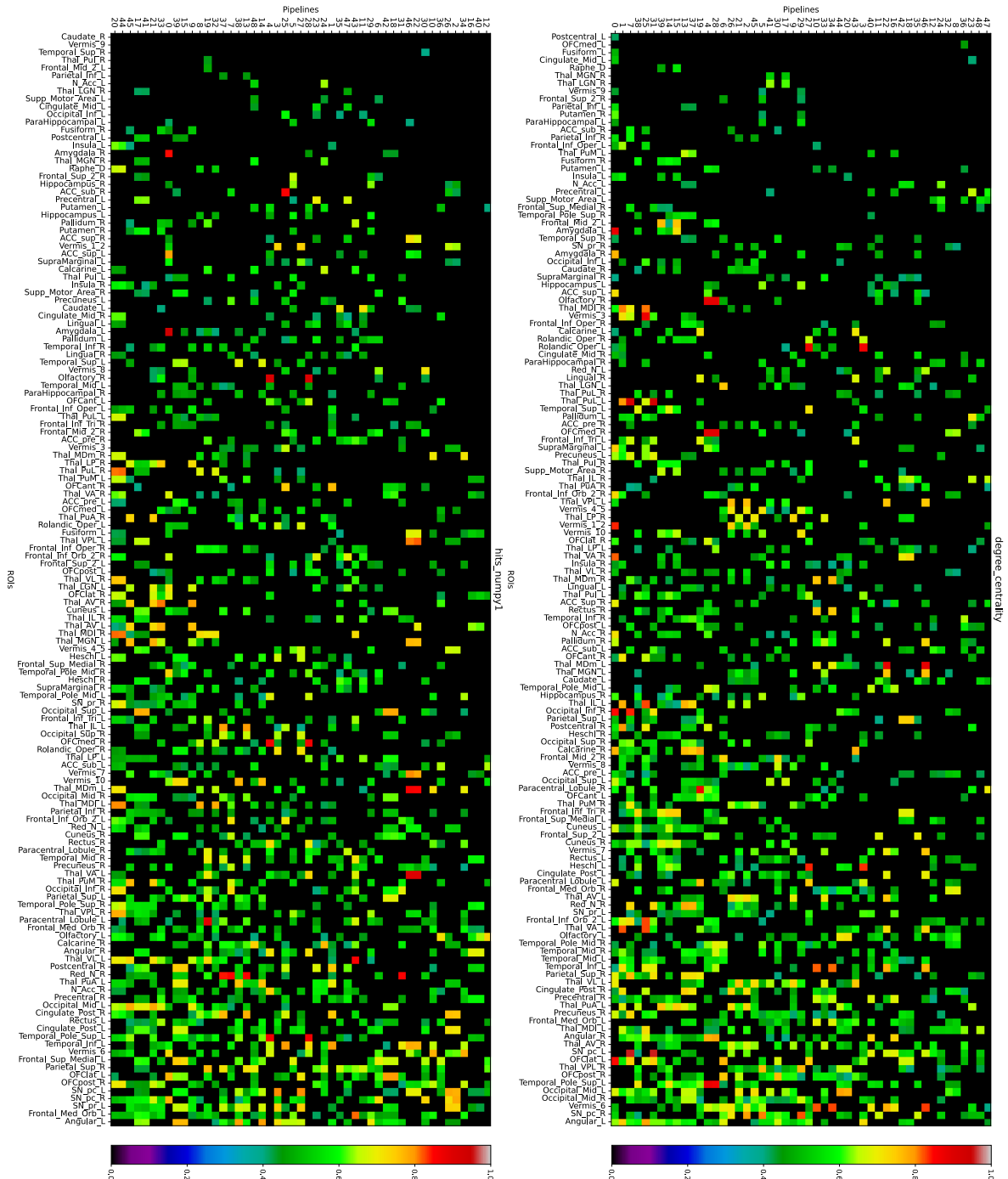


Figura 107 ICC sobre las métricas de hits de concentrador y grado de centralidad en las diferentes regiones para todos los pipelines.

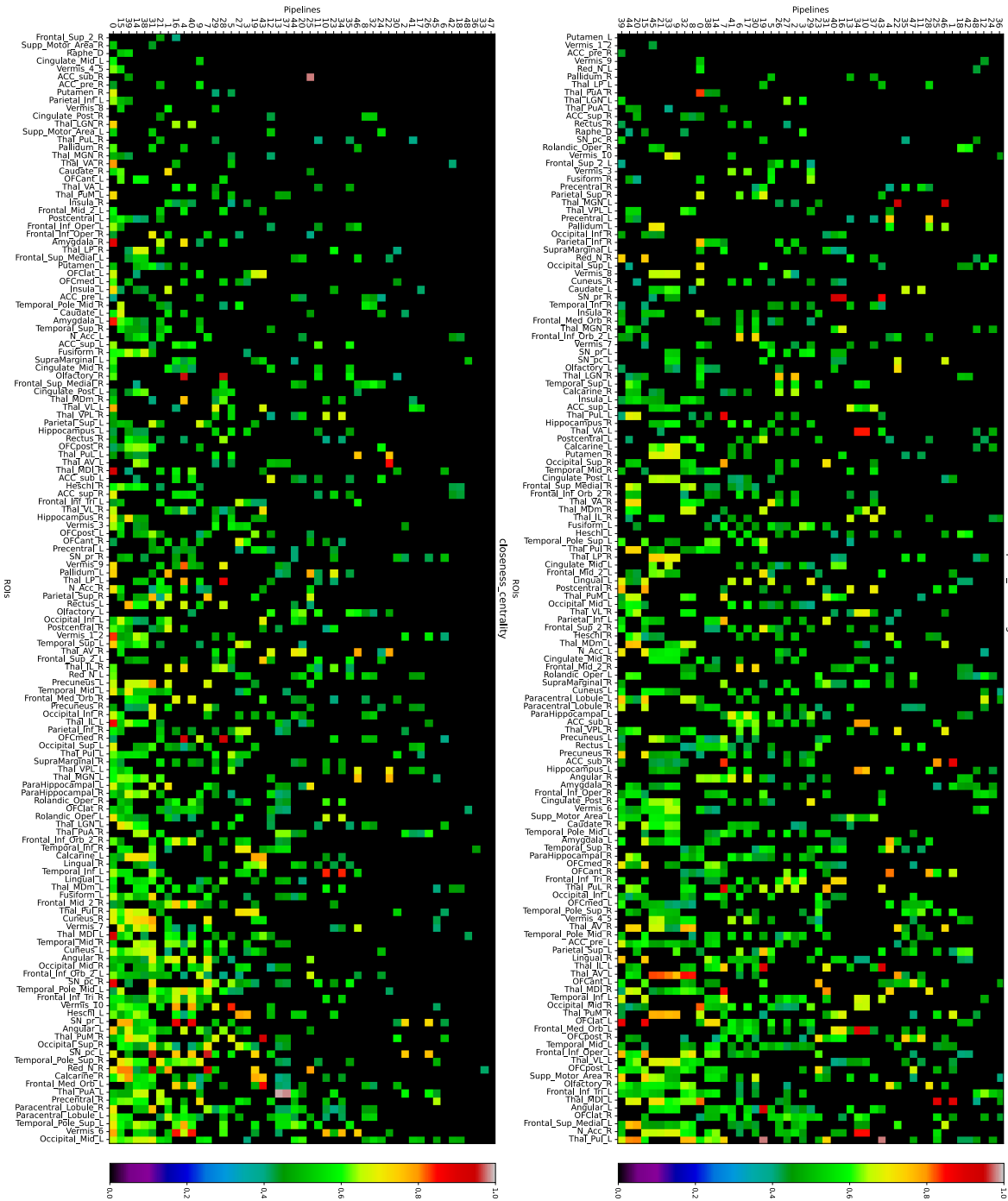


Figura 108 ICC sobre las métricas de centralidad de proximidad y de fracción de posibles cuadrados en las diferentes regiones para todos los pipelines.

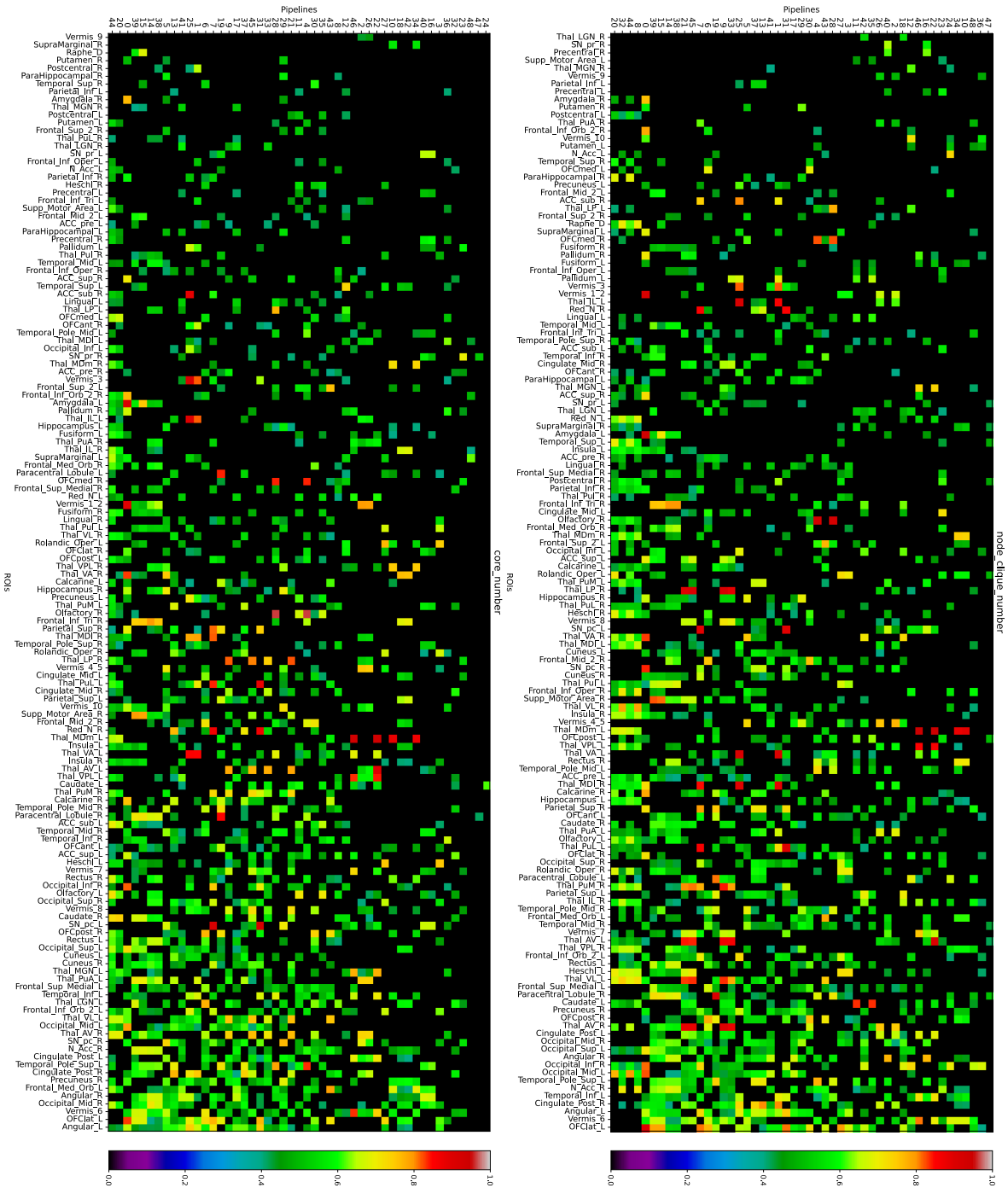


Figura 109 ICC sobre las métricas de número del núcleo y número máximo de cliques en las diferentes regiones para todos los pipelines.

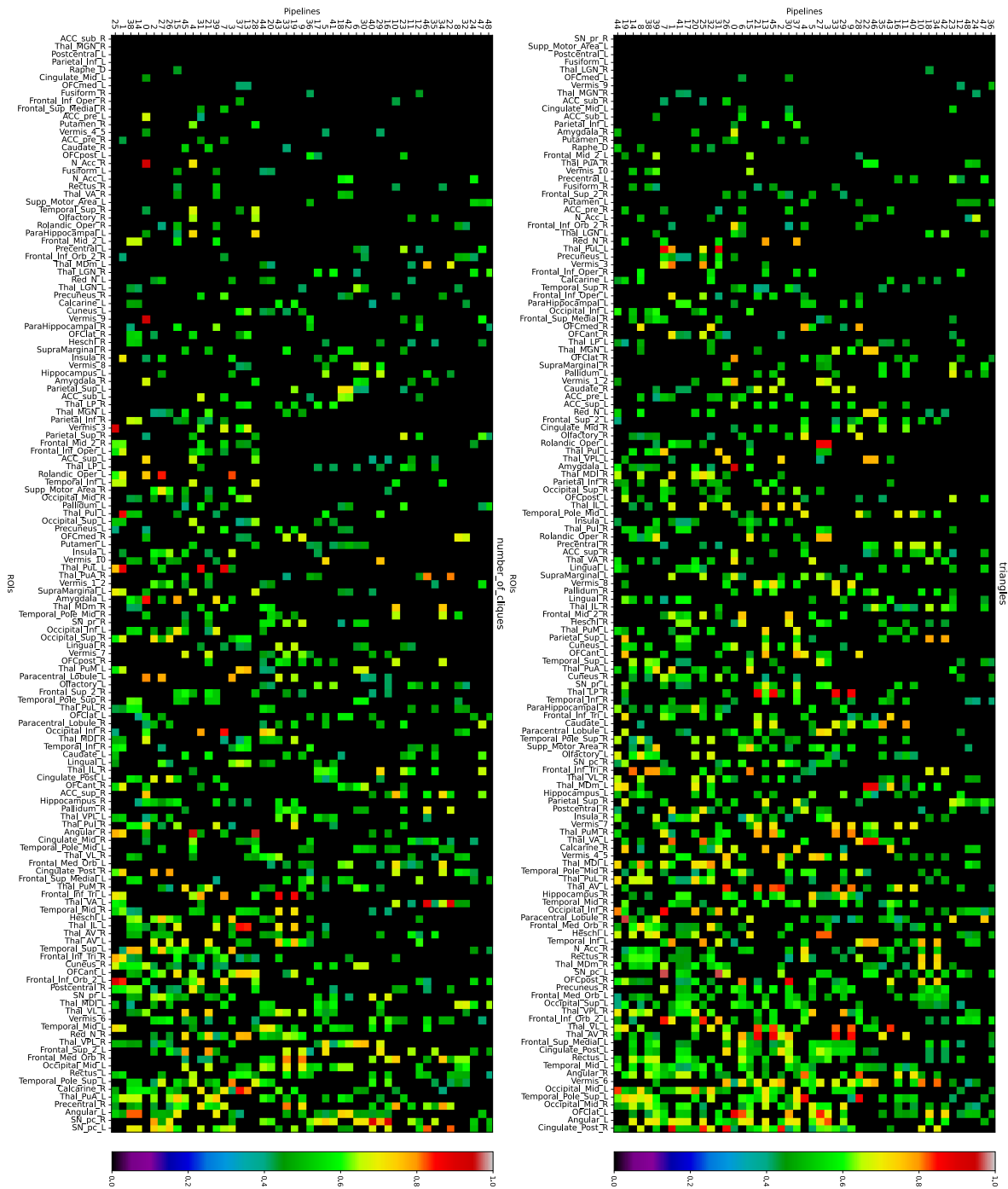


Figura 110 ICC sobre las métricas de clique de mayor tamaño y número de triángulos en las diferentes regiones para todos los pipelines.

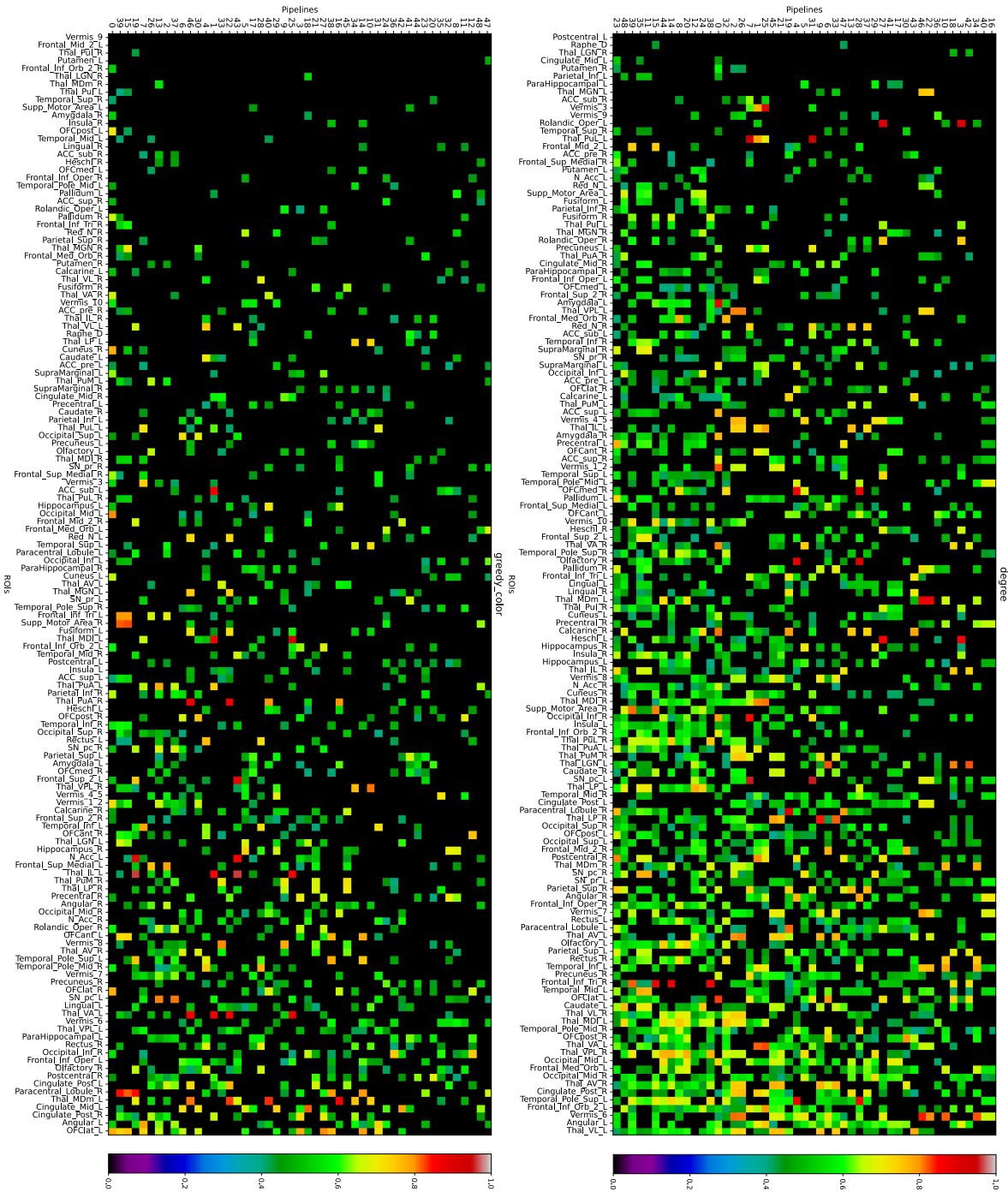


Figura 111 ICC sobre las métricas de color y grado en las diferentes regiones para todos los pipelines.

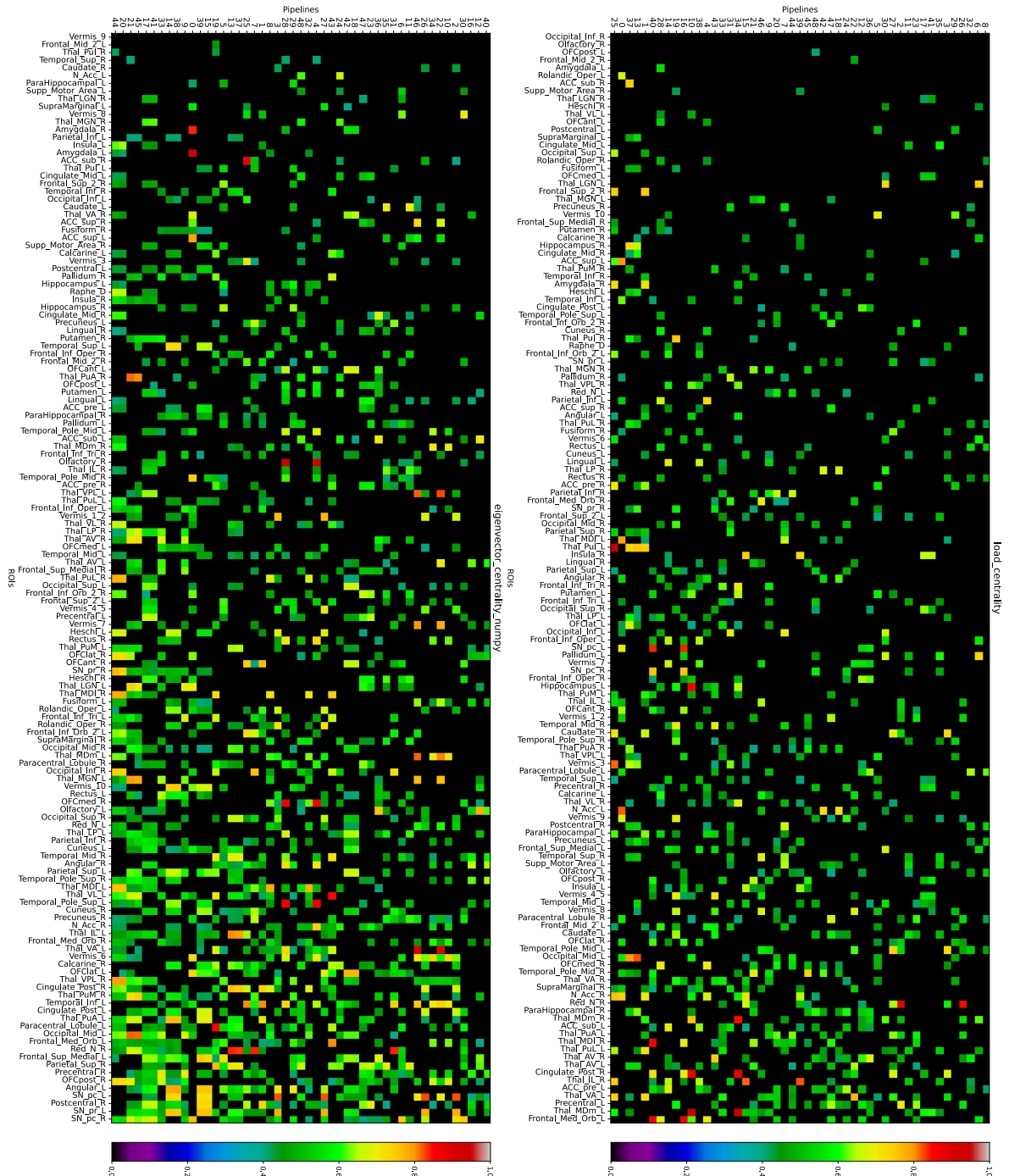


Figura 112 ICC sobre las métricas centralidad de vector propio y centralidad de carga en las diferentes regiones para todos los pipelines.