

INVESTIGACIÓN _X_ PROFUNDIZACIÓN ____

DISEÑO DE UN PROTOTIPO PARA CLASIFICACIÓN AUTOMÁTICA DE IMÁGENES SATELITALES MEDIANTE ALGORITMOS DE MACHINE LEARNING

Autor

Hector Patiño

Tutor

PhD. Alvaro Ortiz Dávila

abril de 2024

A mi madre y mi tía Marina.

AGRADECIMIENTOS

La realización de este documento se hizo en un periodo muy duro y complejo de mi vida, la realización de este hubiese sido imposible sin el ánimo, fuerza y coraje otorgado por grandes amistades y familiares.

Quiero agradecer a Jhon Angarita, Juan Torrado, Karen Merchan, Mayerly Florez y Carlos Lugo por acompañarme en estos duros momentos y por tenerme tanta paciencia haberlos conocido a todos ustedes ha sido increíble, a todos muchas gracias.

A Cristian Socha, Katherine Mikan y Andrés Pantoja por aguantarme, apoyarme y brindarme su cariño y amistad siempre y de manera incondicional, los amo.

A la Magister Maryury Montilla, quien en los peores momentos siempre estuvo para darme mucha fuerza y muchísimo apoyo técnico y moral. Al PhD Johan Avendaño, quien desde hace algunos años siempre está para aconsejarme y guiarme. Al especialista Edwin Rivera, por estar siempre cuando lo necesité apoyándome desde la distancia.

Muy especialmente a mi director el PhD Álvaro Ortiz por su ideas, aportes y exceso de paciencia.

A mi madre, mi tía Marina y mi tocayo quienes siempre han estado presentes en mi vida académica y me han brindado apoyo y amor incondicional. No sería quien soy si no fuera por ustedes, jamás podré retribuirles todo lo que han hecho por mí.

Finalmente, a Kanvitas, mi pitbull para quien no tengo palabras de agradecimiento de todo lo que hizo por mí. No alcanzaste a verme sustentar este documento, te extraño todos los días.

1. CONTENIDO

AGRADECIMIENTOS 3

LISTA DE ILUSTRACIONES 8

LISTA DE ANEXOS 10

LISTA DE TABLAS 11

LISTA DE ECUACIONES 12

Resumen 13

Palabras Clave 15

Introducción 16

2. PROBLEMA DE INVESTIGACIÓN 17

3. OBJETIVOS 18

3.1 Objetivo General 18

3.2 Objetivos Específicos 18

4. JUSTIFICACIÓN 19

4.1 Justificación Teórica 19

4.2 Justificación Metodológica 19

4.3 Justificación Práctica 19

5. MARCO DE REFERENCIA 20

5.1 Marco Teórico 20

5.1.1 Conceptos de percepción remota 20

5.1.2 Canales de preprocesamiento de datos 21

5.1.2.1 Ingeniería de características 21

5.1.2.1.1 Imputación de datos. 22

5.1.2.1.2 Creación de variables mediante índices espectrales 22
5.1.2.1.3 Reducción de dimensionalidad 24
5.1.2.1.3.1 Eliminación de variables correlacionadas 24
5.1.2.1.3.2 Aproximación y proyección uniforme de variedades (UMAP). 24
5.1.2.1.3.3 Análisis de componentes principales (PCA) 25
5.1.2.1.4 Escalar datos. 25
5.1.2.1.4.1 Escalador Robusto 25
5.1.2.1.4.2 Escalador por máximo y por mínimo. 26
5.1.2.1.5 Transformador por cuantiles. 27
5.1.3 Aprendizaje no supervisado 27
5.1.3.1 Agrupación de datos 28
5.1.3.1.1 K-means 28
5.1.3.1.2 DBSCAN 28
5.1.3.1.3 HDBSCAN 29
5.1.4 Aprendizaje Supervisado en imágenes de satélite 29
5.1.4.1 XGBoost 30
5.1.5 Métricas de validación. 31
5.1.5.1 Métricas de algoritmos no supervisados. 31
5.1.5.1.1 Métrica de siluetas 31
5.1.5.1.2 Métrica DBCV (Density Based Clustering Validation) 32
5.1.5.2 Métricas de algoritmos supervisados. 32
5.1.6 Validación cruzada 34
5.2 MARCO ESPACIAL 35
5.3 MARCO TEMPORAL 35
6. HIPÓTESIS 36

6.1 Pregunta de investigación 36
7. METODOLOGÍA 37
7.1 FASE I creación del conjunto de entrenamiento. 39
7.1.1 SELECCIÓN DE ZONAS DE ESTUDIO 40
7.1.2 ADQUISICIÓN DE DATOS 41
7.2 FASE II Aprendizaje no supervisado 42
7.2.1 Aumento de dimensionalidad 43
7.2.2 Modelo no supervisado y optimización de hiperparámetros. 43
7.2.2.1 UMAP y HDBSCAN 43
7.2.2.1.1 Canal de procesamiento de datos del modelo propuesto 44
7.2.2.1.1 Reducción de dimensionalidad y clustering 44
7.2.2.1.1.1 UMAP y HDBSCAN 44
7.2.2.2 PCA y K-Means 46
7.2.3 Interpretación visual de clústeres. 47
7.3 FASE III APRENDIZAJE SUPERVISADO 47
7.3.1 Entrenamiento del modelo en fase de experimentación. 49
7.3.2 Validación del modelo. 51
7.4 FASE IV DESARROLLO DEL PROTOTIPO 51
7.4.1 Desarrollo del prototipo. 52
7.4.1.1 Entrenamiento 52
7.4.1.2 Desarrollo de la API 53
7.4.1.3 Pruebas unitarias y de integración 54
7.4.1.4 Documentación 55
7.4.2 Repositorio nube, CI/CD y publicación 55
8. DESARROLLO METODOLÓGICO. 56

8.1 UMAP y HDBSCAN....60
8.1 PCA y K-MEANS....65
9. ResultadoS Y ANÁLISIS....76
9.1.1 Curva de entrenamiento....79
9.1.2 Evaluación en el conjunto de pruebas. 80
9.1.3 Interpretación del modelo. 82
10. CONCLUSIONES Y PRODUCTOS....88
10.1 Conclusiones....88
10.1.1 Sobre el desarrollo metodológico 88
10.1.1.1 Sobre el proceso no supervisado 88
10.1.1.2 Sobre el proceso supervisado 89
10.1.1.3 Sobre la automatización del proceso de clasificación....89
10.2 Aportes originales. 90
10.3 Trabajos derivados....90
10.3.1 Artículos. 90
10.3.2 Ponencias.....90
11. TRABAJO FUTURO....92
12. DISCUSIÓN Y RECOMENDACIONES....94
12.1 Comparando resultado frente a Dynamic World....94
12.2 Recomendaciones 94
13. ANEXOS....96
14. BIBLIOGRAFIA 128

LISTA DE ILUSTRACIONES

Ilustración 1 Flujo metodológico general del proceso de clasificación de imágenes de satélite. 30
Ilustración 2 Forma general de la validación cruzada por grupos. 35
Ilustración 3 Metodología general del proyecto por fases. 38
Ilustración 4 Fases desarrolladas en cada ambiente. 38
Ilustración 5 Proceso general de la fase I 40
Ilustración 6 Muestreo de cada una de las zonas de estudio en una banda 41
Ilustración 7 Ejemplo bandas por transformar. 42
Ilustración 8 Proceso general de la fase II. 43
Ilustración 9 Proceso General de la fase III 49
Ilustración 10 Proceso general de la fase IV 52
Ilustración 11 Idea central el entrenamiento del modelo en el prototipo 53
Ilustración 12 Diagrama de uso de la API utilizando STAC 53
Ilustración 13 Clase principal del prototipo 54
Ilustración 14 Distribución espacial de las zonas de estudio. 57
Ilustración 15 Cantidad de zonas por año. 58
Ilustración 16 Coordenadas paralelas umap - hdbscan imagen con ID 1. 61
Ilustración 17 Visualización de reducción de la dimensionalidad con umap. 64
Ilustración 18 Coordenadas paralelas PCA – k means imagen con ID 1. 65
Ilustración 19 Comportamiento de los mejores hiperparámetros en PCA y K means. 66
Ilustración 20 Concentración de clústeres por hiperparámetros 67
Ilustración 21 Índice de siluetas para los mejores hiperparámetros 67
Ilustración 22 Interpretación de coberturas de clústeres para la zona 19 69
Ilustración 23 Interpretación de coberturas de clústeres para la zona 23 70
Ilustración 24 Interpretación de coberturas de clústeres para la zona 4 71
Ilustración 25 Boxplot Exactitud balanceada entrenamiento vs validación. 73
Ilustración 26 Coordenadas paralelas y su impacto en la exactitud balanceada 74

Ilustración 27 Impacto de cada hiperparámetro en el conjunto de validación _____ 74
Ilustración 28 Comparación de max depth en entrenamiento y validación _____ 75
Ilustración 29 Curva de aprendizaje del modelo propuesto _____ 80
Ilustración 30 Importancia de características con valores SHAP e impacto por clase. _____ 82
Ilustración 31 Imagen resultado de la clasificación automática. _____ 84
Ilustración 32 Resultado de la clasificación en otra zona del planeta. _____ 87

LISTA DE ANEXOS

Anexo 1 Índices utilizados por grupo de extracción en el aprendizaje no supervisado 96
Anexo 2 Resultados de experimentos para las tres imágenes de exploración umap y hdbsan .97
Anexo 3 Índice de silueta por hiperparámetros para cada zona de estudio. 106
Anexo 4 Historial de optimización de hiperparámetros 122

LISTA DE TABLAS

Tabla 1 Datos crudos de un conjunto de entrenamiento. 21
Tabla 2 Datos imputados por mediana 22
Tabla 3 Índices espectrales por grupos de extracción de información. 23
Tabla 4 Aumento de dimensionalidad mediante índices. 23
Tabla 5 Disminución de dimensionalidad usando correlación. 24
Tabla 6 Datos transformados utilizando escalador robusto. 26
Tabla 7 Datos transformados usando escalador máximo y mínimo. 26
Tabla 8 Hiperparámetros de xgboost. 30
Tabla 9 Especificación técnicas de cada ambiente. 39
Tabla 10 Canal de procesamiento de datos para uso de UMAP y HDBSCAN. 44
Tabla 11 Valores del espacio de hiperparámetros para el proceso UMAP y HDBSCAN 45
Tabla 12 Valores del espacio de hiperparámetros en PCA y K-Means para cada zona. 47
Tabla 13 Flujo de datos en fase de experimentación. 49
Tabla 14 Valores del espacio de hiperparámetros a ser optimizados 51
Tabla 15 Información de cada imagen y cantidad de zonas. 58
Tabla 16 Resultados con UMAP y HDBSCAN vs PCA y K Means 62
Tabla 17 Cinco mejores experimentos en validación por grupos 72
Tabla 18 Matriz de confusión y reporte de clasificación en el conjunto de entrenamiento 78
Tabla 19 Matriz de confusión y reporte de clasificación en el conjunto de pruebas 81

LISTA DE ECUACIONES

Ecuación

(1)....25
(2)....26
(3)....32
(4)....32
(5)....32
(6)....33
(7)....33
(8)....33
(9)....33
(10)....34
(11)....34

RESUMEN

La identificación de coberturas de la tierra empleando imágenes de satélite y algoritmos supervisados está enmarcado en un proceso en el cual es necesaria la interacción de expertos que comprendan cómo se debe trabajar con este tipo de imágenes. Este proceso requiere algunos pasos que por naturaleza deben ser realizados manualmente, un ejemplo de esto es la identificación de las zonas de interés y su clase asignada. Sin embargo, todo el proceso es repetitivo, no solo para proyectos académicos, sino para entidades gubernamentales y privadas. De igual manera, es frecuente que los datos de entrenamiento sean descartados después de obtener los resultados sobre una imagen (o serie de imágenes), pero en este proceso se pierde un trabajo muy valioso pues se pierden los valores de los píxeles de las diferentes bandas y la clase asignada a dichos píxeles. Ahora bien, aunque en la actualidad existen diferentes herramientas y capacidad en la nube, el proceso de clasificación sigue siendo prácticamente el mismo al anteriormente mencionado. Adicionalmente, existen incluso nuevas herramientas que permiten obtener una clasificación automática de manera casi inmediata, aunque los datos de entrenamiento de esas herramientas nunca son conocidos por el usuario, convirtiendo estas herramientas en cajas negras.

En este documento se diseña e implementa un prototipo que mediante algoritmos de machine learning permite realizar clasificación automática de coberturas de la tierra para clases predefinidas; adicionalmente, se presenta el conjunto de entrenamiento, esto permitirá que, a futuro, dicho prototipo pueda ser versionado de manera que nuevos datos de entrenamiento, nuevos pasos de preprocesamiento, diferentes algoritmos o diferentes combinaciones de hiperparámetros puedan ser tenidas en cuenta en el momento del entrenamiento y evitar así que el rendimiento del modelo decrezca con el tiempo.

Para lograr este resultado se realiza una selección de zonas en el territorio colombiano y en tiempos diferentes (años 2021 y 2022) de manera que se logre obtener un conjunto de datos multitemporal y espacialmente distribuido, dichas zonas son escogidas teniendo en cuenta que tengan poca presencia de nubes (aunque se escogen zonas que las posean, para que el modelo aprenda a caracterizarlas correctamente, al igual que las sombras generadas). Posteriormente,

para cada imagen se entrenaron una serie de modelos no supervisados los cuales identifican diferentes clústeres presentes en cada imagen. Seguido a esto, se realiza una reclasificación de los clústeres previamente mencionados a clases predefinidas. Después, por cada imagen y con la finalidad de reducir un poco la cantidad de datos por temas asociados a hardware se realiza una muestra aleatoria estratificada para asegurar que todas las clases de todas las imágenes estén presentes en el conjunto de entrenamiento final. Luego se construye y entrena un modelo propuesto, los hiperparámetros del modelo son encontrados utilizando una validación por grupos, donde cada imagen es un grupo. La validación del modelo hiperparametrizado es realizada sobre imágenes del 2023 las cuales no fueron tenidas en cuenta en el proceso de modelamiento. Finalmente, obtenidos los hiperparámetros del modelo que maximizan la efectividad del modelo se desarrolla el prototipo, el cual es desplegado en un repositorio abierto para su consumo libre.

PALABRAS CLAVE

Aprendizaje de máquina, xgboost, imágenes de satélite, procesamiento de imágenes de sensores remotos.

INTRODUCCIÓN

La investigación se realiza como una forma de generar valor a los diferentes roles asociados en el proceso de clasificación de uso y cobertura de la tierra mediante imágenes de satélite, con la finalidad de que este proceso se realice de una manera más rápida. También se realiza como un punto de partida para la Universidad Distrital, para que las metodologías propuestas sirvan como base para el despliegue de modelos de aprendizaje de máquina o de aprendizaje profundo que tengan un impacto en temas de planeación, cartografía y temas relacionados con geomática los cuales permitan tomar decisiones más rápidas con menor reproceso y más económicas.

La investigación se centrará en flujos de trabajo del proceso de clasificación de imágenes de satélite, metodologías de aprendizaje de máquina y en cómo integrar estos flujos de trabajo en un prototipo que permita automatizar dicho flujo de trabajo, para ello se realizará un modelo que no solo será multitemporal, si no también tendrá en cuenta diferentes áreas del país, con la finalidad de obtener diferente información espectral, tanto en el tiempo como en el espacio. Para facilitar el proceso de adquisición y tratamiento de datos fuente se utilizan imágenes de satélite que puedan ser descargadas de manera sencilla por cualquier usuario. En este caso se utilizan imágenes del sensor Landsat 9.

2. PROBLEMA DE INVESTIGACIÓN

Conocer la clasificación del suelo es importante para estudios de cambio climático, monitoreo de servicios ecosistémicos, planeación del territorio, prevención de desastres, etc.(Feng et al., 2012). Esta clasificación en la mayoría de los casos se realiza mediante interpretación visual de imágenes de satélite, lo cual es costoso, dispendioso e impreciso (Suárez Londoño et al., 2017), esto debido a que es necesario contar con personal especializado para llevar a cabo esta tarea. En la actualidad existen dos grandes aplicaciones que permiten realizar clasificación automática o semiautomática, que pueden ser utilizadas por cualquier usuario del mundo. La primera de ellas es RECAP (Murray et al., 2018) la cual tiene una interfaz de usuario amigable, permite realizar clasificación de cobertura de manera semiautomática en la cual el usuario escoge los puntos de entrenamiento y sobre esta información ejecuta un modelo que utiliza un algoritmo de bosques aleatorios. La segunda herramienta es Dynamic World (Brown et al., 2022) el cual realiza una clasificación automática sobre imágenes Sentinel 2, este modelo tiene sus clases ya previamente definidas (al contrario de RECAP) e incluso tiene un módulo de análisis de cambios. Sin embargo, al momento de realizar esta investigación existen pocos modelos a los cuales se les haya implementado un prototipo para ser utilizados por diferentes usuarios (Suárez Londoño et al., 2017)(Trujillo Jiménez et al., 2021)(Wan et al., 2022) crean modelos y algunos comparten datos, sin embargo, ninguno tiene un prototipo que pueda ser utilizado. Todo esto y con el auge de Python en programación no es fácil encontrar librerías que permitan realizar este trabajo de manera automática, incluso grandes resultados como los obtenidos por (Naushad et al., 2021) en el cual obtienen una exactitud general del 99% usando el dataset propuesto por (Helber et al., 2019) no escalan debido a que quedan en notebooks de jupyter.

3. OBJETIVOS

3.1 OBJETIVO GENERAL

Diseñar un prototipo que utilice un algoritmo de aprendizaje de máquina el cual permita realizar clasificación automática de cobertura de la tierra

3.2 OBJETIVOS ESPECÍFICOS

- Crear un conjunto de entrenamiento con imágenes de satélite multitemporal y repartidas en el territorio colombiano.
- Identificar agrupaciones de datos que permitan establecer similitud entre los valores de píxel de las diferentes bandas utilizando algoritmos no supervisados.
- Construir un modelo supervisado utilizando el conjunto de entrenamiento creado el cual permita realizar clasificación automática de imágenes de satélite.
- Validar el modelo construido en imágenes de satélite del territorio colombiano cuyos datos no hayan sido utilizados en el proceso de entrenamiento.

4. JUSTIFICACIÓN

Se propone diseñar e implementar un sistema que permita realizar clasificación de imágenes de satélite de manera automática, conociendo previamente el identificador de la imagen y el área deseada.

4.1 JUSTIFICACIÓN TEÓRICA

El propósito teórico de la investigación es aportar al conocimiento actual sobre la implementación de sistemas de aprendizaje de máquina, en especial, en los temas geoespaciales, más concretamente en imágenes de satélite, aumentando mediante la elaboración de esta investigación un contexto teórico para el diseño de futuros sistemas.

4.2 JUSTIFICACIÓN METODOLÓGICA

Para lograr los objetivos de esta investigación es necesario recurrir a varias técnicas tanto estadísticas, como de aprendizaje de máquina y percepción remota. Inicialmente se propone establecer un conjunto de datos de entrenamiento, con ello, se pretende encontrar agrupaciones de datos que permitan establecer diferentes coberturas en varias imágenes. Posteriormente, se creará un modelo de aprendizaje de máquina el cual será usado para diseñar e implementar un sistema. De esta investigación, su metodología y los resultados pretende apoyar implementaciones en diferentes sensores, mejoramiento continuo del sistema.

4.3 JUSTIFICACIÓN PRÁCTICA

Realizar clasificación de cobertura de la tierra para sus diferentes usos es una labor ardua, desgastante y costosa, ofrecer a un usuario determinado el conjunto de entrenamiento y un sistema que le permita automatizar sus procesos puede permitir agilizar flujos de trabajo y liberar cuellos de botella en el proceso de clasificación de imágenes.

5. MARCO DE REFERENCIA

5.1 MARCO TEÓRICO

5.1.1 Conceptos de percepción remota.

Debido al hecho de que la principal fuente de información del proyecto proviene del espacio se hace necesario realizar una breve introducción a la percepción remota. “la percepción remota es la ciencia y el arte de obtener información sobre un objeto, área o fenómeno a través del análisis de datos adquiridos por un dispositivo que no está en contacto con el objeto, área o fenómenos de investigación. Al usar varios sensores, remotamente se está recolectando información que puede ser analizada para obtener información sobre el objeto, área o fenómeno bajo investigación.”(Lillesand et al., 2014). Las imágenes de satélite son un tipo clásico de percepción remota, pues para realizar la toma de una escena un satélite está orbitando el planeta a una distancia considerable.

En concordancia con Chuvieco (1995) existen 2 tipos de sensores que adquieren información de la cobertura terrestre desde el espacio, esto son los sensores pasivos y los sensores activos. Los sensores pasivos son aquellos que necesitan energía natural para realizar mediciones (como por ejemplo el sol), mientras que, los sensores activos son capaces de emitir la energía. Esta energía es una forma de radiación electromagnética. La principal fuente de información de esta investigación es la proveniente de sensores pasivos. Los sensores pasivos logran obtener la información desde longitudes de onda de 0.4 µm hasta aproximadamente 14 µm, es decir, van desde el espectro visible (rojo, verde y azul) hasta el infrarrojo térmico. Cada elemento en la superficie de la tierra al ser sometida a la luz solar refleja energía que es captada por el sensor. El cómo reacciona la energía a un contacto con un cuerpo puede cuantificarse en todo el espectro, por tanto, cada cuerpo tiene un comportamiento único, lo cual permite identificar qué tipo de cobertura es; a esto se le llama la curva de reflectividad o curva espectral (Chuvieco, 1995).

5.1.2 Canales de preprocesamiento de datos

Los canales de preprocesamiento de datos son pasos claves en el proceso de análisis de datos y uso de algoritmos de aprendizaje; es por esto por lo que el diseño de estos canales influye en la robustez de un algoritmo de aprendizaje de realizar mejores o peores estimaciones, sin embargo, no existe un solo canal de preprocesamiento de datos que funcione para todos los diferentes tipos de conjuntos de datos que existen (Giovanelli et al., 2022). Los canales de preprocesamiento de datos no son más que un conjunto de transformaciones a los datos crudos de un conjunto de datos; en imágenes de satélite una forma de ver una transformación puede ser vista al momento de añadir índices espectrales al conjunto de entrenamiento, pero existen desde luego una amplia cantidad de transformaciones que puede ser utilizadas. Estas transformaciones sobre los datos crudos son muy importantes debido a que algunos algoritmos son muy sensibles a la distribución de los datos, como lo son la mayoría de los algoritmos lineales; también existen algoritmos cuyo rendimiento es afectado si los datos de entrenamiento tienen escalas diferentes (como lo algoritmos basados en distancias)(Zebari et al., 2020). En esta investigación cada transformación realizada en un canal de preprocesamiento será conocida como ingeniería de características.

5.1.2.1 Ingeniería de características

Como se mencionó previamente cada transformación del canal de procesamiento tiene un objetivo claro sobre un conjunto de datos, estas transformaciones dependiendo de la naturaleza de estas deben permitir a la trasformación aprender ciertas particularidades del conjunto de datos de entrenamiento. Apelando a un ejemplo, supóngase que un conjunto de datos utilizado no tiene valores nulos, pero al momento de realizar la estimación por diferentes problemas, algunos registros o píxeles vienen nulos, esto podría hacer que un modelo completo deje de funcionar al no poder manejar de manera correcta este tipo de situaciones; en este caso específico un transformador debería aprender cómo solucionar los valores nulos. A continuación, se explicarán diferentes transformaciones de datos aplicados a los valores de la Tabla 1

Tabla 1 Datos crudos de un conjunto de entrenamiento.
Fuente: Elaboración propia.

Var 1	Var 2	Var 3
45	23	1

Var 1	Var 2	Var 3
52	72	78
63	NaN	NaN
98	96	8
12	NaN	0

5.1.2.1.1 Imputación de datos.

La imputación es el proceso en el cual se da manejo a los datos nulos o faltantes de un conjunto de datos. Con respecto a las diferentes técnicas que existen, estas pueden ser agrupadas en dos familias: la imputación univariada y la multivariada, en esta investigación solo se usarán técnicas univariadas debido a rapidez de procesamiento.

Las transformaciones univariadas más sencillas que se pueden aprender son la media y la mediana de cada variable, al tener estos valores es posible reemplazar los datos faltantes por dichos valores. La Tabla 2 muestra los resultados de una imputación que aprenda la mediana de los datos de la Tabla 1.

Tabla 2 Datos imputados por mediana

Var 1	Var 2	Var 3
45.00	23.00	1.00
52.00	72.00	78.00
63.00	67.25	18.80
98.00	96.00	8.00
12.00	67.25	0.00

5.1.2.1.2 Creación de variables mediante índices espectrales

La creación de variables de un conjunto de datos de entrenamiento permite generar nuevas variables y ser agregadas (si así es querido) al conjunto de datos original, esto permite extraer cierta información de los datos. En imágenes de satélite, este proceso es por lo general, realizado con índices espectrales. Dichos índices son operaciones matemáticas las cuales, combinadas de determinada manera, extraen información de una o varias coberturas de la tierra para la cual fueron diseñadas. La cantidad de índices espectrales es muy amplia y se puede dividir en varios grupos:

Tabla 3 Índices espectrales por grupos de extracción de información.
Fuente: Elaboración propia

Tipo de índice	**Índices de ejemplo**
Vegetación	NDVI(Rouse et al., 1973), AFRI1600(Karnieli et al., 2001), GOSAVI(Sripada et al., 2005), MSI(Hunt & Rock, 1989).
Zonas construidas	BLFEI(Bouhennache et al., 2019), NBAI(Waqar et al., 2012), NDBI(Zha et al., 2003), UI(Kawamura et al., 1996).
Agua	ANDWI(Rad et al., 2021), NDWI(McFeeters, 1996), WRI(Rokni et al., 2014), NDTI(Lacaux et al., 2007).
Nieve	S3(Saito & Yamazaki, 1999), SWI(Dixit et al., 2019), NDSI(Riggs et al., 1994), NBSIMS(Arreola-Esquivel et al., 2021).
Incendios	BAI(Del Pilar et al., 1998), MIRBI(Trigg & Flasse, 2001), NBRT1 y NBRT2 (Holden et al., 2005).
Suelos	BI(Rikimaru et al., 2002), BaI(Lin et al., 2005), DBSI(Rasul et al., 2018), NDSol(Deng et al., 2015)

Debido a que este proceso ya se encuentra automatizado no se listaran todos los índices. La Tabla 4 muestra la creación de índices artificiales (usados solo a manera de ejemplo) de los datos procedentes de la Tabla 2 y del cómo el índice 2 y el índice 3 tienen similitud en sus datos, pero las escalas son diferentes, esto se realizó a propósito para mostrar transformaciones adicionales.

Tabla 4 Aumento de dimensionalidad mediante índices.
Fuente: Elaboración propia

Var 1	**Var 2**	**Var 3**	**Índice 1**	**Índice 2**	**Índice 3**
45.00	23.00	1.00	22.00	22.00	0.22
52.00	72.00	78.00	-20.00	-6.00	-0.06
63.00	67.25	18.80	-4.25	48.45	0.48
98.00	96.00	8.00	2.00	88.00	0.88
12.00	67.25	0.00	-55.25	67.25	0.67

5.1.2.1.3 Reducción de dimensionalidad

En cualquier proceso de aprendizaje automático, es esencial considerar la dimensionalidad de los datos, ya que puede haber características que resulten poco relevantes o altamente correlacionadas con otras. La reducción de la dimensionalidad es un proceso fundamental en el aprendizaje automático, ya que contribuye a simplificar la complejidad de los modelos, lo que, a su vez, facilita la visualización y la interpretación de los datos. Además, esta reducción puede acelerar significativamente el proceso de entrenamiento de los modelos (Reddy et al., 2020). A continuación, se enumeran algunas técnicas consideradas en el proyecto.

5.1.2.1.3.1 Eliminación de variables correlacionadas

Una manera de reducción de dimensionalidad es la eliminación de variables correlacionadas. En este tipo de técnica se escoge un umbral a partir del cual las variables que superen dicha correlación son eliminadas del conjunto de datos. Tomando como ejemplo los resultados obtenidos en la Tabla 4 y un umbral de correlación del 0.99 se obtienen los resultados de la Tabla 5, como se mencionó de manera previa, los datos del índice 3 han sido eliminados al tener una correlación de 1.0 con el índice 2.

Tabla 5 Disminución de dimensionalidad usando correlación.

Var 1	Var 2	Var 3	Índice 1	Índice 2
45	23	1	22	22
52	72	78	-20	-6
63	67.25	18.8	-4.25	48.45
98	96	8	2	88
12	67.25	0	-55.25	67.25

5.1.2.1.3.2 Aproximación y proyección uniforme de variedades (UMAP).

La aproximación y proyección uniforme de variedades o UMAP (Uniform Manifold Approximation and Projection) es una técnica de aprendizaje no supervisado utilizado para reducir dimensionalidad y para visualizar datos en dimensiones menores a la original. Se basa en la teoría de variedades y utiliza la topología algebraica para encontrar una proyección de los datos en un espacio de menor dimensión. Utiliza la estructura de vecindad de los datos para construir una función de densidad que capture la estructura subyacente de los datos. Posterior a

esto, la función es utilizada para encontrar una proyección en una menor dimensión preservando la estructura topología y las relaciones entre los puntos de datos (McInnes et al., 2018).

5.1.2.1.3.3 Análisis de componentes principales (PCA)

El análisis de componentes principal o PCA (Principal Component Analysis) es una técnica estadística para identificar patrones y reducir la dimensionalidad de un conjunto de datos, conservando la mayor cantidad de información posible. Para ellos se transforman las variables originales en un nuevo conjunto de variables no correlacionadas llamadas componentes principales, en la cual cada una de las variables explica la varianza del conjunto original de manera ordenada. Esto se realiza mediante un proceso de diagonalización de la matriz de covarianza del conjunto de datos original, esta matriz representa la relación entre las variables originales y proporciona información sobre la varianza y la correlación entre ellas. Al representar la varianza de manera ordenada es fácil seleccionar componentes que sean más o menos importantes en el proceso de selección de características. Una característica importante de este método es que se asume linealidad entre las diferentes variables.(Holland, 2008) .

5.1.2.1.4 Escalar datos.

Las transformaciones de datos en los cuales se llevan los datos de todas las variables a una misma escala son también muy variadas y dependen mucho del análisis de los datos realizados, algunas técnicas son por ejemplo el uso de logaritmos si las variables tienen una distribución con colas largas y tienen muchos posibles valores en escalas diferentes. A lo largo de la investigación se usaron principalmente, cuatro maneras de escalar los datos.

5.1.2.1.4.1 Escalador Robusto

Está técnica se utiliza en el caso de presencia de datos atípicos en un conjunto de datos, dichos datos atípicos pueden ser de magnitudes muy grandes. Este se basa en el uso de del rango intercuartílico, para ellos el nuevo valor escalado tomará el valor original y le restará la media (por lo cual la remueve) y la divide sobre el rango intercuartílico.

	$x' = \frac{x - \tilde{x}}{IQR}$	(1)

Si se partiera de los datos de la Tabla 5 utilizando el escalador robusto los resultados serían los de la Tabla 6

Tabla 6 Datos transformados utilizando escalador robusto.

Var 1	Var 2	Var 3	Índice 1	Índice 2
-0.62	-5.04	-0.48	1.56	-0.86
-0.14	0.26	5.18	-0.36	-1.54
0.62	-0.26	0.83	0.36	-0.23
3.03	2.85	0.04	0.64	0.73
-2.90	-0.26	-0.55	-1.97	0.23

5.1.2.1.4.2 Escalador por máximo y por mínimo.

Es una transformación en la cual las variables de un dataset de datos son llevadas a un rango de valores escogidos por el usuario (generalmente entre 0 y 1) en el cual el mínimo valor de cada variable será 0 y el máximo será 1, para el resto de los valores se utiliza la siguiente formula:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (2)$$

Partiendo de los datos de la Tabla 6 los resultados después de aplicar esta transformación serían los de la Tabla 7, en ella se puede apreciar como, por ejemplo, para la variable 1 el valor que representaba un 3.03 previamente ahora representa un 1.00.

Tabla 7 Datos transformados usando escalador máximo y mínimo.

Var 1	Var 2	Var 3	Índice 1	Índice 2
0.38	0.00	0.01	1.00	0.30
0.47	0.67	1.00	0.46	0.00
0.59	0.61	0.24	0.66	0.58
1.00	1.00	0.10	0.74	1.00
0.00	0.61	0.00	0.00	0.78

5.1.2.1.5 Transformador por cuantiles.

Este tipo de transformación está basado en la función cuantiles, el objetivo principal es transformar los datos de manera que puedan tener una distribución normal estándar o una distribución uniforme. Es muy sensible a los datos atípicos por lo que un tratamiento previo es necesario. También puede distorsionar correlaciones lineales entre las diferentes variables. Para poder realizar este proceso se realizan los siguientes pasos:

- Se usa una estimación de la función de distribución acumulativa para mapear los valores originales a una distribución uniforme.
- Los resultados anteriores son mapeados a una distribución deseada usando la función de cuantiles que está asociada a dicha distribución.
- Los nuevos valores que estén por arriba o por debajo del rango ajustado quedarán en los límites de la distribución de salida

Este tipo de transformaciones y modelos asociados a aprendizaje profundo han tenido un impacto muy amplio en procesamiento natural de lenguaje y otras áreas(Vaswani et al., 2017) .

5.1.3 Aprendizaje no supervisado

El aprendizaje no supervisado es una técnica fundamental en el campo del aprendizaje automático que se utiliza para descubrir patrones intrínsecos, estructuras ocultas y relaciones no etiquetadas en un conjunto de datos. A través de algoritmos de aprendizaje no supervisado, se busca comprender en profundidad la información contenida en los datos, lo que puede incluir la identificación de grupos, la reducción de dimensionalidad y la detección de anomalías. Además, esta técnica es esencial para desentrañar tanto los patrones evidentes como el ruido latente en un conjunto de datos, lo que contribuye a una comprensión más completa de la información subyacente (Dridi, 2015). Es muy importante mencionar que no es necesario que en el conjunto de entrenamiento exista una variable a estimar, dado el caso que se tenga una variable a estimar es posible utilizar este tipo de aprendizaje como un paso de preprocesamiento de los datos, permitiendo en algunos casos, mejorar los resultados de un aprendizaje supervisado(Oh et al., 2018). El aprendizaje no supervisado tiene algunas ramas, en las cuales puede ser utilizado siendo estas:

- Agrupación de datos o Clustering.

- Reducción de dimensionalidad.
- Aprendizaje de manifolds.

5.1.3.1 Agrupación de datos

La agrupación de datos o Clustering es el proceso de reunir observaciones de datos en grupos homogéneos (Gormley et al., 2023). Existen diferentes técnicas de agrupación de datos, algunos tienen en cuenta la cantidad de grupos que se requieren como k-means, otros solo requieren una distancia de búsqueda como DBSCAN, algunos requieren solo la cantidad mínima de puntos para que un grupo de observaciones formen un grupo o clúster como lo es HDBSCAN. Existen desde luego muchas de estas técnicas y todas tiene ventajas y desventajas.

5.1.3.1.1 K-means

K-means es una técnica de Clustering que agrupa los puntos de un conjunto de datos en k clústeres donde k es un número predefinido de clústeres. El objetivo de K-means es minimizar la distancia entre los puntos dentro de cada clúster y maximizar la distancia entre los puntos de diferentes clústeres, para ellos se escoge con una selección aleatoria de k centroides. Luego los puntos se asignan al clúster cuyo centroide está más cercano. Una vez los puntos han sido asignados los centroides se recalculan como el promedio de todos los puntos en cada clúster, esto se repite hasta que los centroides no cambien significativamente de una iteración a otra (Banerjee et al., 2005).

K-means tiene algunos problemas asociados a su uso, el primero el que es importante conocer el número de clústeres previamente (aunque esto se puede conocer en la optimización de los hiperparámetros), también puede utilizarse el método del codo o el método de siluetas, aunque existen problemas asociados al método del codo (Vergara et al., 2020). Por otra parte, dado que utiliza las distancias para identificar los puntos de cada clúster, no tiene, por naturaleza, una buena manera de identificar clústeres no esféricos.

5.1.3.1.2 DBSCAN

DBSCAN (Density- Based Spatial Clustering of Applications with Noise) es un algoritmo utilizado para identificar clústeres en un conjunto de datos. A diferencia de K-means no requiere

conocer la cantidad de clústeres, ya que trabaja identificando regiones densas en el espacio dimensional de los datos, los puntos que se encuentran en áreas de baja densidad se consideran ruido (Schubert et al., 2017). Esto es posible ya que el algoritmo define un radio y un umbral mínimo de puntos. El radio define la distancia máxima que se considera para que un punto se considere dentro del mismo clúster que otro punto. El umbral de mínimo es la cantidad mínima que debe estar dentro del ese radio para que el punto sea considera parte del clúster. Una ventaja enorme de este algoritmo es que los grupos pueden tener cualquier forma. Una desventaja es que su rendimiento va en función de los parámetros escogidos (Ester et al., 1996) .

5.1.3.1.3 HDBSCAN

HDBSCAN (Hierarchical Density – Based Spatial Clustering of Applications with Noise) es una técnica de clustering basada en la densidad de los puntos del conjunto de datos. Es una extensión de DBSCAN, al igual que DBSCAN, permite identificar el número óptimo de clústers, pero a diferencia de él tiene un enfoque jerárquico, comienza agrupando los puntos cercanos entre sí en pequeños grupos y luego va uniendo estos grupos a otros, lo cual hace que cada vez vayan aumentando la cantidad de puntos del clúster. La técnica utiliza una medida de estabilidad para identificar clústeres más estables y evitar que existan clústeres ruidosos. Tiene una ventaja y es que permite identificar clústeres de diferentes formas y tamaños, lo cual es útil si estos no tienen una forma esférica (como en K-Means). Adicionalmente, permite identificar puntos ruidosos o atípicos los cuales no quedan agregados a ningún clúster(McInnes et al., 2017)

5.1.4 Aprendizaje Supervisado en imágenes de satélite

El objetivo del aprendizaje supervisado es construir modelos con base en la distribución de datos etiquetados en términos de variables predictoras(Kotsiantis, 2007). El propósito de dicho modelo es estimar las etiquetas de nuevos datos cuya etiqueta no sea conocida. Para lograr este objetivo en imágenes de satélite se realizan por lo general 5 pasos (D. et al., 2017) los cuales son la colección de imágenes, el preprocesamiento de imágenes, extracción de características, entrenamiento del modelo y evaluación del modelo (Ver Ilustración 1).

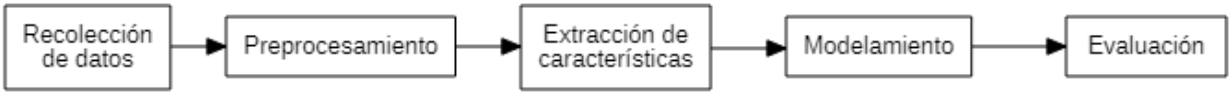


Ilustración 1 Flujo metodológico general del proceso de clasificación de imágenes de satélite.

Ahora bien, este flujo metodológico no es del todo lineal, existen pasos que por naturaleza y en búsqueda de mejorar las estimaciones del modelo deben ser iterativos. Un ejemplo de esto es que un conjunto de configuraciones o hiperparámetros del modelo puede, para un conjunto de entrenamiento definido, dar mejores o peores resultado, por lo cual, buscar estas configuraciones requiere esquemas de validación cruzada, la cual es una tarea iterativa. Adicionalmente, puede suceder que el modelo propuesto, no generalice bien sobre datos de prueba; por tanto, es necesario replantear el flujo incluso desde el paso de preprocesamiento.

5.1.4.1 XGBoost

Se utiliza cómo una técnica de regresión y clasificación. Es muy utilizado debido a su eficacia, capacidad de estimar con exactitud y por ganar competencias de machine learning (Nielsen, 2016). XGBoost funciona combinando varios árboles muy simples que terminan creando un modelo complejo. Cada modelo se entrena de manera secuencial para corregir errores del modelo previo. El objetivo es minimizar la función de pérdida que mide el error del modelo en la estimación de la variable objetivo. Además, tiene una capacidad muy buena para manejar grandes conjuntos de datos debido a que puede procesar en paralelo y distribuir en varios núcleos (Chen & Guestrin, 2016). En la API de Python este algoritmo tiene un conjunto amplio de hiperparámetros, pero es particularmente más sensible a cambios en unos cuantos.

Tabla 8 Hiperparámetros de xgboost.
Fuente: Elaboración propia con base en (Chen & Guestrin, 2016)

Hiperparámetro	**Argumento en Python**	**Explicación**
Cantidad de árboles	n_estimators	Cantidad de árboles a ser tenidos en cuenta
Profundidad del	max_depth	Profundidad de cada árbol

Hiperparámetro	Argumento en Python	Explicación
árbol		
Peso mínimo del hijo	min_child_weight	Número mínimo requerido en un nodo hoja.
Muestra	subsample	El porcentaje de filas que ser utilizarán en cada ronda del entrenamiento.
Gamma	gamma	Da un umbral que cada nodo debe sobrepasar antes de seguirse dividiendo

5.1.5 Métricas de validación.

Dependiendo un problema en cuestión existen varias métricas que buscan indicar la fiabilidad de un algoritmo para mostrar consistencia o estimar sobre nuevos datos, siendo lo primero útil para modelos no supervisados y lo segundo para modelos supervisados. En este proyecto se utilizaron varias métricas dependiendo del experimento que se estuviese desarrollando.

5.1.5.1 Métricas de algoritmos no supervisados.

Las diferentes métricas de algoritmos no supervisados buscan establecer qué tan consistentes son los resultados esperados, en problemas de agrupación de datos buscan interpretar qué tan cohesionados están los diferentes clústeres. Cada métrica tiene su rango de valores posibles y su propia interpretación a dichos valores.

5.1.5.1.1 Métrica de siluetas

Evalúa la cohesión y la separación de los diferentes clústeres encontrados en la ejecución de un modelo no supervisado. En este caso, se calcula la métrica de silueta para cada muestra, el cual mide que tan cerca está la muestra a su propi grupo comparado frente a los grupos vecinos. Un valor cercano a 1 es deseado, un valor cercano a 0 indica que hay superposición en los grupos y un valor de -1 indica que las muestras no están bien cohesionadas Para calcularlo se utiliza la formula (3) en la cual *a* es la media de la distancia entre todos los clústeres y *b* es la distancia

entre una muestra y el clúster más cercano del cual esa muestra no hace parte (Rousseeuw, 1987).

$$SC = \frac{(b-a)}{\max(b-a)} \quad (3)$$

5.1.5.1.2 Métrica DBCV (Density Based Clustering Validation)

Es utilizada en para evaluar la cohesión y la separación de los diferentes clústeres de un modelo no supervisado y es particularmente útil en algoritmos que pueden detectar clústeres de maneras no esféricas y adicionalmente tiene en cuenta los puntos de que identificados por el modelo como 'ruido'. El resultado es una suma ponderada de validación de cada clúster, O es la cantidad ruido, l es la cantidad de clusters encontrados por el modelo, Ci $V_c(C_i)$ es la calidad de consistencia de cada cluster, y Ci es la cantidad de registros de cada cluster. Es importante mencionar que esta métrica es algo lenta de calcular debido al cálculo de calidad de cada clúster (Moulavi et al., 2014).

$$DBCV(C) = \sum_{i=1}^{i=l} \frac{|C_i|}{|O|} V_c(C_i) \quad (4)$$

5.1.5.2 Métricas de algoritmos supervisados.

Tambien existen una amplia cantidad de métricas de validación de modelos supervisados y su utilidad depende exclusivamente del caso de estudio. Para el estudio se decidió escoger la métrica conocida como exactitud balanceada (Mosley, 2013). Está métrica se tomó dado que tiene en cuenta el balance entre clases; es decir, tiene en cuenta tanto la cantidad de ocurrencias de una clase en un conjunto de datos como el hecho de que todas las clases pueden tener diferentes ocurrencias. En contrapartida, existe una métrica conocida como la exactitud general; esta métrica no tiene en cuenta los balances de las clases, ni la cantidad de ocurrencias de esta; la manera de calcular la exactitud general está dada por:

$$EG = \frac{TP+TN}{TP+\ TN+FP+FN} \quad (5)$$

En la ecuación (5), TP hace referencia a los verdaderos positivos, TN a los verdaderos negativos, FP y FN a los falsos positivos y a los falsos negativos respectivamente. Para

ejemplificar porque no se utilizará la exactitud general obsérvese la matriz de confusión de la Matriz 1 en ella se puede apreciar como existe fuerte desbalance entre las diferentes clases A y B, el clasificador logra identificar muy bien la clase A, pero para la única instancia de la clase B realiza una mala estimación.

Matriz 1 Resultado de un clasificador

		Estimadas	
		Clase A	Clase B
Real	Clase A	99	0
	Clase B	1	0

Ahora bien, suponiendo que la clase B es la que se interesa detectar y sería TP, mientras que los valores correctamente identificados de la clase A serían los verdaderos negativos o TN. Cómo se puede observar a continuación dicho modelo tiene una exactitud general de 0.99 lo cual es completamente falso o erróneo ya que el modelo no logra estimar de manera correcta la clase B.

$$EG = \frac{0 + 99}{0 + 99 + 0 + 1} = 0.99 \qquad (6)$$

Adicionalmente, teniendo en cuenta que la sensibilidad de un modelo está dada por:

$$S = \frac{TP}{TP + FN} \qquad (7)$$

Y que la especificidad está determinada por:

$$E = \frac{TN}{TN + FP} \qquad (8)$$

Es posible definir la exactitud balanceada como:

$$EB = \frac{E + S}{2} \qquad (9)$$

Al reemplazar estos valores de la Matriz 1 se obtiene un 0.5 lo cual es un valor bastante bajo comparado frente al 0.99 que se estaba obteniendo inicialmente.

$$EB = \frac{0+1}{2} = 0.5 \qquad (10)$$

Sin embargo, esta fórmula está definida solo para procesos de clasificación binario, esto es, clase A vs clase B; para problemas de multiclasificación existe un ajuste el cual es que se tiene en cuenta solo la sensibilidad y se divide por el número de clases. En este caso Ei es la sensibilidad de dicha una clase determinada y n es la cantidad de clases que hay presentes en el conjunto de datos.

$$EB = \sum_{i=1}^{n} \frac{E_i}{n} \qquad (11)$$

5.1.6 Validación cruzada

La validación cruzada es una estrategia de validación valga la redundancia que busca identificar en un dataset de validación que tan exacto es un modelo teniendo en cuenta determinados hiperparámetros (Cawley & Talbot, 2010). Para ello, se divide el dataset de entrenamiento entre conjunto de entrenamiento y conjunto de validación; posteriormente, se entrena el algoritmo en conjunto de entrenamiento y se valida sobre el conjunto de validación, logrando así conocer si el modelo entrenado con determinados hiperparámetros sufre de sobreajuste o subajuste (Khaledian & Miller, 2020). Ahora bien, existe una viable de este tipo de validación la cual es llamada validación cruzada por grupos. En este tipo de validación existen sujetos que poseen una cantidad de clases que es posible que no estén en otros sujetos, al dividir por grupos se elimina el hecho de los datos de un grupo se filtren en el conjunto de validación y se obtenga una estimación muy optimista de la clasificación. Para entender lo anterior de manera más clara suponga los siguiente; se tienen tres imágenes de satélite, en la primera de las imágenes A están presentes las clases verde y naranja, en la segunda imagen B están presentes las clases naranja y rosa y en la tercera imagen C están presentes las clases rosa y naranja (ver Ilustración 2)

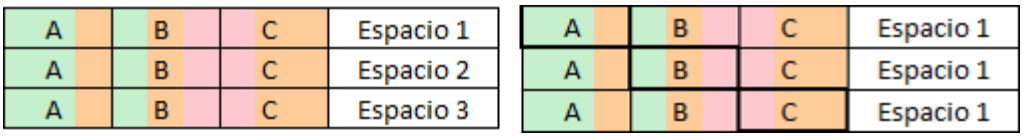


Ilustración 2 Forma general de la validación cruzada por grupos.

Fuente: Adaptado de (Raschka, 2018)

En la parte derecha se pueden ver los conjuntos de datos y los espacios de hiperparámetros definidos en los experimentos, en la parte izquierda se puede ver como se realiza para cada experimento una división de los datos, en este caso las imágenes que están señaladas en negrita hacen parte del conjunto de validación, mientras que las que no lo están hacen parte del conjunto de entrenamiento, tambien es posible agrupar un poco estos conjuntos con la finalidad de disminuir un poco la carga computacional, es decir se entrena sobre A y B y se valida sobre el conjunto C, de esta manera se reduce considerablemente el tiempo de optimización(Ruppert, 2004).

5.2 MARCO ESPACIAL

Se propone tomar todo el territorio colombiano, con la finalidad de obtener muestras variadas y representativas, como tal el único problema asociado para la toma de muestras es que se pueda observar la cobertura la de tierra al momento de la toma en una zona específica.

5.3 MARCO TEMPORAL

La investigación propone realizar tomar imágenes de muestra de los años 2021 y 2022, analizando las zonas en las cuales la presencia de nubes exista, pero no sea predominante, no se plantea tener restricciones temporales en cuanto a meses con mucha o poca precipitación, sino que se pueda observar la cobertura de la tierra. La validación de los datos se realizará para datos del año 2023.

6. HIPÓTESIS

Utilizando aprendizaje no supervisado cuyo objetivo sea identificar patrones en los valores de los píxeles de diferentes imágenes de satélite distribuidas en el territorio colombiano y su posterior reclasificación a etiquetas predefinidas, es posible, crear un conjunto de datos que pueda ser utilizado para un construir un modelo supervisado con el cual pueda diseñarse un prototipo cuya finalidad sea automatizar el proceso de clasificación de cobertura de la tierra mediante imágenes de satélite, ayudando así a analistas de datos espaciales, empresas privadas, entidades públicas en sus procesos.

6.1 PREGUNTA DE INVESTIGACIÓN

¿Cómo pueden emplearse técnicas de aprendizaje no supervisado para detectar patrones en los valores de píxeles de imágenes satelitales del territorio colombiano, con el propósito de reasignarlos en categorías predefinidas y así construir un conjunto de datos que facilite la creación de un modelo supervisado para la automatización de la clasificación de coberturas terrestres a partir de imágenes satelitales?

7. METODOLOGÍA

En este capítulo se hace una descripción de los datos y las áreas de estudio, el software utilizado y los métodos aplicados durante el desarrollo de la investigación. Para lograr los objetivos del proyecto se decidió realizar la clasificación de cobertura de la tierra teniendo en cuenta los valores de los píxeles de las diferentes bandas de las zonas de estudio seleccionadas, es decir, una clasificación espectral. El satélite escogido para realizar el estudio es Landsat9, un sensor que tiene poco más de año y medio en operación. En la Ilustración 3 se puede apreciar la metodología general del proyecto. En términos generales, el proyecto se dividió en cuatro grandes fases con las cuales se garantiza la realización de este. En resumen, la fase I fue la búsqueda de información, el muestreo de los datos, la descarga de las zonas de estudio; para esta fase se tuvo en cuenta que las zonas de estudio del año 2021 y 2022 fueran tenidas en cuenta para el conjunto de entrenamiento, en cambio, las imágenes del año 2023 fueron en su totalidad tenidas en cuenta como los datos de pruebas finales (en la ilustración hacen referencia a los procesos que tienen un asterisco), adicionalmente en esta fase se definen las coberturas presentes en las diferentes zonas de estudio. La fase II consistió en detectar clústeres de cada una de las zonas de estudio y su posterior reclasificación a cada una de las clases definidas en el estudio. La fase III, consistió en la creación de un muestreo aleatorio estratificado de todos los datos de las zonas de estudio con la finalidad de disminuir un poco la cantidad de recursos computacionales requeridos para el entrenamiento, en esta fase también se construye y entrena un modelo, la línea de color naranja de la validación final hace referencia a los datos del año 2023, los cuales pasaron por el mismo procesamiento que los datos de entrenamiento. La fase IV en el desarrollo del prototipo para ser utilizado por el usuario y su posterior despliegue.

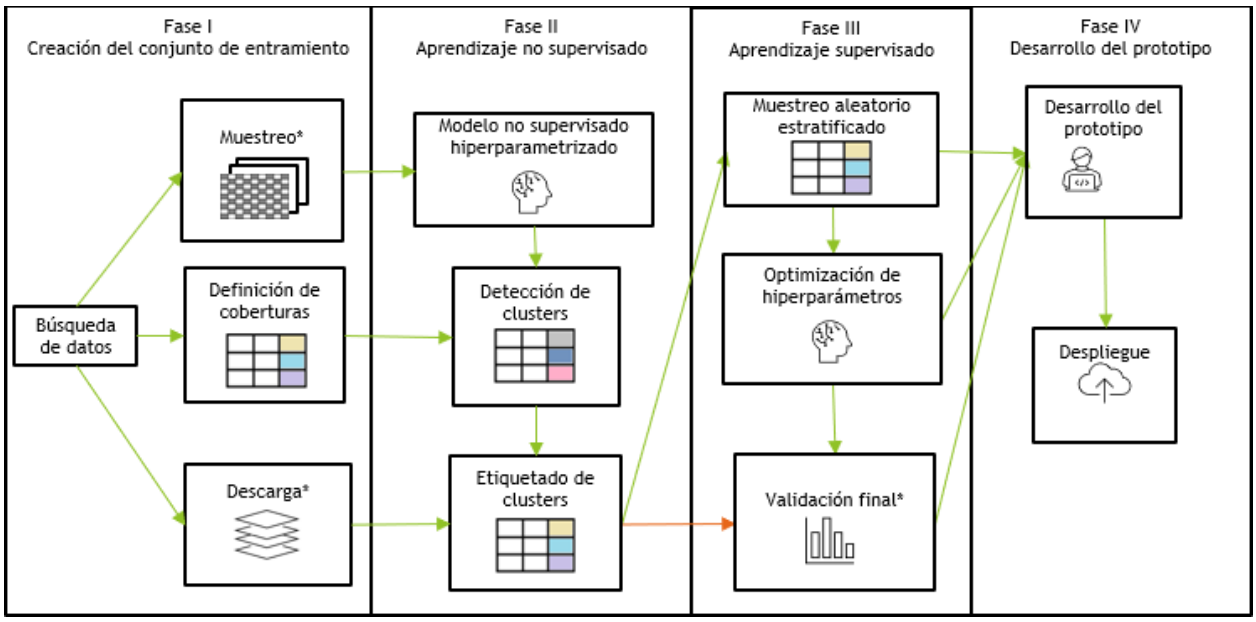


Ilustración 3 Metodología general del proyecto por fases.

Con la finalidad de evitar conflictos entre diferentes paquetes de Python, se decidió utilizar tres ambientes diferentes, uno para la búsqueda y descarga de información, otro ambiente para la experimentación (tanto no supervisada como supervisada) y las validaciones necesarias, finalmente otro ambiente para desarrollar el prototipo. En la Ilustración 4, se aprecian las cómo se utilizó cada ambiente para cada una de las fases y cómo están divididas.



Ilustración 4 Fases desarrolladas en cada ambiente.

La Tabla 9 muestra las especificaciones técnicas de cada uno de los ambientes. Con respecto al ambiente de la fase II y la fase III es importante mencionar que esta fase se realizó en la nube, se escogió la plataforma Kaggle para ejecutar los diferentes análisis y experimentaciones necesarias, esto debido a que la plataforma permite ejecutar procesos hasta un total de 12 horas sin necesidad de interacción de ningún tipo, adicionalmente, permite guardar versiones de cada una de las ejecuciones de los diferentes notebooks en los que se ejecute un proceso, eso sin

mencionar que también tiene acceso gratuito a GPUs las cuales fueron utilizadas para realizar más el proceso de experimentación de manera más ágil. Sobre la fase I y la fase IV, hay que mencionar que se utilizó un computador personal con las especificaciones detalladas en la Tabla 9. En este punto se quiere recalcar que debido a la forma en la que se realizó esta división de ambientes el modelo que se obtenga al final de la fase III debe volver a ser entrenado en la fase IV, esto con la finalidad de que el canal de procesamiento de datos y formas de llamado dentro del prototipo se mantenga de manera más simple.

Tabla 9 Especificación técnicas de cada ambiente.

Fase	**OS**	**Procesador**	**Memoria RAM**	**Memoria GPU**	**Versión Python**
Fase I - Descarga	Windows 10 con WSL y Linux subsystem	Intel core i7 (7Gen)	32 Gbs	4 Gbs	3.8
Fase II y Fase III – Experimentación	Linux – Nube - Kaggle	Intel Xeon	16 Gbs	16 Gbs	3.8[1]
Fase IV - Desarrollo	Windows 10	Inter core i7 (7Gen)	32 Gbs	4 Gbs	3.10

7.1 FASE I CREACIÓN DEL CONJUNTO DE ENTRENAMIENTO.

La fase de creación del conjunto de entrenamiento consistió en la búsqueda de las imágenes de satélite, su posterior adquisición y unas transformaciones necesarias sobre las zonas derivadas de las imágenes para que fuera más fácil trabajar con ellas en el actual ecosistema de análisis de datos de Python. La búsqueda de las imágenes se realizó utilizando la plataforma de Microsoft Planetary Computer (MPC)[2]. En dicha plataforma se pueden visualizar de manera fácil diferentes programas de imágenes de satélite. Una vez seleccionadas las imágenes se descargaban accediendo al SpatioTemporal Access Catalog (STAC)[3], el cual un es lenguaje para describir información geoespacial. En Python existen librerías que permiten acceder a los tipos de datos

[1] En junio del año 2023 kaggle actualizó la versión de Python a 3.10
[2] https://planetarycomputer.microsoft.com/explore?c=-72.7877%2C4.1797&z=7.30&v=2
[3] https://stacspec.org/en

con los que trabaja el STAC y también permiten trabajar con un tipo de dato llamado xarray (Hoyer & Hamman, 2017) con el cual se puede trabajar con arreglos multidimensionales de una forma sencilla y eficiente, además dicho paquete es el núcleo central de todo el pangeo (Eynard-Bontemps et al., 2019) que es una infraestructura de código abierto que busca realizar procesamiento en la nube con un alto rendimiento, esta no se utiliza en la investigación, pero es pertinente que sea mencionado, debido a la escalabilidad con la que se diseñó el prototipo. Los productos esperados de esta fase son todas las zonas de estudio y el muestreo establecido en formato dataframe. Teniendo en cuenta que Landsat9 fue lanzado el 27 de septiembre del 2021, se plantea realizar la búsqueda de imágenes para los años 2021, 2022 y 2023. El proceso general puede ver en la Ilustración 5.

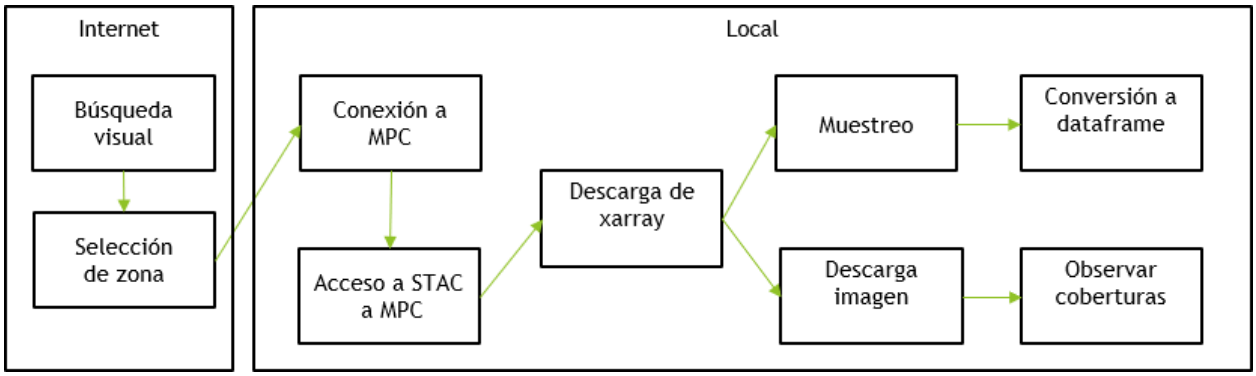


Ilustración 5 Proceso general de la fase I

7.1.1 SELECCIÓN DE ZONAS DE ESTUDIO

La investigación en un inicio planteó desarrollarse en todo el territorio colombiano. En términos generales, se realizó una inspección visual de imágenes del año 2021, 2022 y 2023 del satélite Landsat 9. Con la inspección se buscaba que existiera variedad de coberturas en las zonas escogidas, que tuvieran poca o nula presencia de nubosidad; sin embargo, no se descartaban imágenes con presencia de nubes, pues también se intentaba enseñarle al modelo que este tipo de cobertura[4] está presente en las diferentes imágenes. En esta inspección, se pudo evidenciar que en la región amazónica y la región de Orinoquia las coberturas tienden a cambiar muy poco y son bastante homogéneas, existen amplias regiones con una o dos coberturas siendo la parte

[4] Se aclara que las nubes no son una cobertura de la tierra, pero son un tipo de clase/objeto que está puede estar presente en una imagen de satélite.

boscosa predominante, esto podía llegar a ser complicado debido a que iba a existir una cobertura en general con muchas muestras (o valores de píxeles asociados a esta cobertura) lo provocaría un amplio desbalance entre la cantidad de muestra de una cobertura comparada con el resto de las coberturas, en esta zona también se evidenció nubosidad en los periodos buscados. Teniendo en cuenta lo mencionado anteriormente, esta zona fue excluida de la investigación.

7.1.2 ADQUISICIÓN DE DATOS

Una vez seleccionadas las zonas de estudio se procede a su descarga, en la Ilustración 5 se mostró que no solo se descargaba la imagen si no también una muestra de la imagen, dicha muestra se descargaba en formato tabular (como si fuera una tabla de base de datos) debido a que el ecosistema actual de Python permite realizar transformaciones sobre este tipo de dato con facilidad. Con respecto al muestreo, y con la finalidad de disminuir un poco el consumo de recursos informáticos se realizó un muestreo espacialmente distribuido, esto se realizó debido a que, al tomar valores aleatorios de los píxeles de una imagen, existía la posibilidad de que algunas coberturas presentes en dicha zona fueran excluidas del muestreo. De esta manera se garantizaba que existieran presencia de todas las clases en cada una de las zonas y una reducción del conjunto de entrenamiento. En la Ilustración 6 se puede apreciar cómo es el muestreo por cada una de las bandas, en color gris se encuentran los píxeles que son tenidos en cuenta en el muestro, el resto de los píxeles son omitidos, este proceso es el mismo para todas las bandas que conforman cada una de las zonas de estudio.

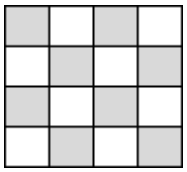


Ilustración 6 Muestreo de cada una de las zonas de estudio en una banda

Posteriormente al proceso de muestro se realiza una reconversión de la imagen de formato multidimensional a formato tabla, en la Ilustración 7 se muestra un ejemplo de 3 bandas espectrales, si importa cuales sean, los valores son una representación numérica de los niveles digitales de la imagen, al costado derecho se observa la representación en formato de tablas, la idea central es que se tomen los píxeles intercalados primero por filas y se va iterando entre todas las filas de la imagen (eje y).

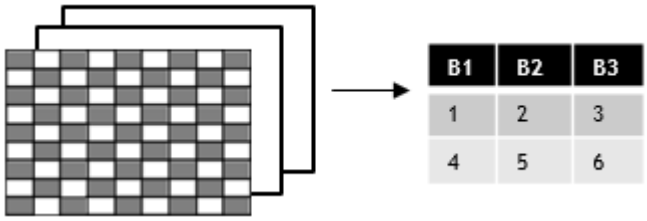


Ilustración 7 Ejemplo bandas por transformar.

7.2 FASE II APRENDIZAJE NO SUPERVISADO

El objetivo principal de esta fase es la identificación de coberturas presentes en cada una de las zonas seleccionadas. La principal característica de esta fase es que en dicha identificación de clústeres (coberturas) no debía existir interacción humana, sino únicamente al final del proceso. Para lograr esto se desarrolló un script que permitiera a la máquina encontrar los clústeres presentes de manera automática. Para ello, se partía de un supuesto importante: no se conocían a priori la cantidad de clústeres presentes en cada zona y era la máquina la que tenía que decidir la cantidad de estas, esto se determinó utilizando el resultado de la métrica de siluetas y DBCV, el primero siendo usado para K-means y el segundo para HDBSCAN, en un proceso de hiperparametrización en el cual se buscaba maximizar la métrica escogida. Una vez se obtenía la métrica más alta se generaba un archivo de Microsoft Excel con todas las coberturas del estudio de manera que fuera más fácil el proceso de interpretación, en paralelo se ejecutaba el modelo usando los datos de la imagen y se obtenía la imagen de los que representaba los clústeres de la imagen de entrada. Seguido a esto, mediante interpretación visual se hacía una reclasificación de cada uno de los clústeres identificados por el modelo no supervisado a las coberturas de la tierra del estudio.

Es pertinente recordar que el muestreo de la Fase I son los datos de entrada de esta fase. Los datos esperados de esta fase son los datos etiquetados con las coberturas de la tierra a ser identificadas de cada una de las zonas de estudio.

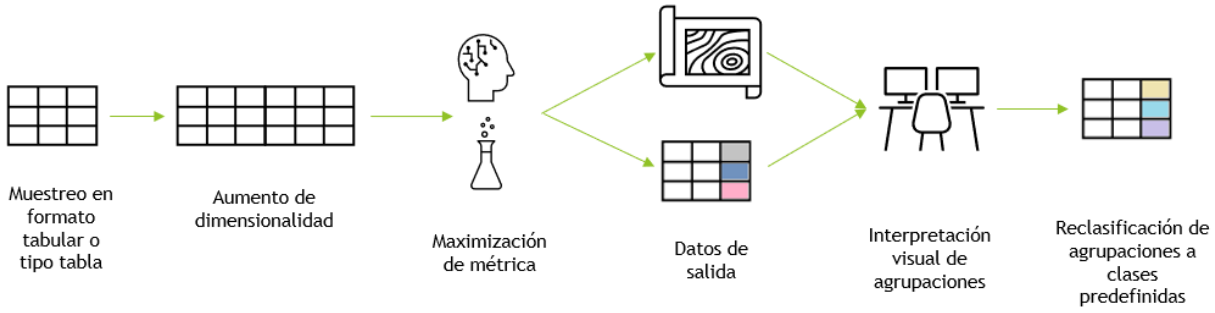


Ilustración 8 Proceso general de la fase II.

7.2.1 Aumento de dimensionalidad

El aumento de dimensionalidad se realizó mediante la utilización de índices espectrales que permitieran identificar diferentes coberturas sin importar la zona. Para esto se utilizó un paquete de Python llamado spyndex (Montero et al., 2023) el cual permite realizar este aumento de dimensionalidad de datos de imágenes de satélite de manera simple y eficiente. Los índices utilizados en esta parte del proceso pueden ser revisados en Anexo 1

7.2.2 Modelo no supervisado y optimización de hiperparámetros.

Este proceso consistió en la construcción de un canal de procesamiento de datos que permitiera obtener la cantidad de coberturas en cada una de las zonas de estudio sin conocer la cantidad de coberturas presentes. Con la finalidad de disminuir el coste computacional del proceso de Clustering se utilizó un proceso de reducción de la dimensionalidad ya que esto aceleraría la obtención de dichos clústeres. No obstante, el proceso canal de procesamiento y algoritmos utilizados en un primer caso no fueron satisfactorios, este proceso constó de utilizar UMAP como método de reducción de la dimensionalidad y HDBSCAN cómo método de agrupación. Por otro lado, con un proceso más simple como PCA y KMEANS, se obtuvieron resultados más idóneos para el análisis. En este proceso se explicarán los métodos utilizados por ambos canales de procesamiento.

7.2.2.1 UMAP y HDBSCAN

En primer lugar, se realizaron una serie de experimentos con UMAP y HDBSCAN, la finalidad de dichos experimentos era: analizar la rapidez del algoritmo y revisar de qué manera eran agrupados los píxeles de las imágenes. Para lograr esto se creó un canal de

preprocesamiento de datos y posteriormente se implementaron los algoritmos de reducción de dimensionalidad y de clustering.

7.2.2.1.1 Canal de procesamiento de datos del modelo propuesto

Un pequeño ejemplo de los datos de salida teniendo en cuenta valores de la Tabla 1 pueden ser visualizado en la Tabla 10. La idea central del canal de procesamiento es llevar los datos a una misma escala numérica, realizando tratamientos relativamente sencillos sobre los datos.

7.2.2.1.1 Reducción de dimensionalidad y clustering

En términos generales, se escogieron 6 hiperparámetros (Ver Tabla 10). Sin embargo, después de analizar la salida obtenida y adicionalmente, se optó por no utilizar esta forma de realizar los clústeres. La razón fue que a pesar de que HDBSCAN es un algoritmo que tiene ventajas para identificar clústeres de diferentes formas, las inspecciones visuales de los datos no eran nada satisfactorias, y eran, por decir poco, muy pobres. Ahora bien, también es posible que los diferentes espacios de hiperparámetros no fueran los mejores para este caso específico y más experimentos deban ser realizados. Adicionalmente, UMAP tiene un costo computacional alto y al tener que procesar varias zonas múltiples veces por la optimización de hiperparámetros esto se hacía poco eficiente. Los resultados de este proceso pueden ser vistos en

7.2.2.1.1.1 UMAP y HDBSCAN

Tabla 10 Canal de procesamiento de datos para uso de UMAP y HDBSCAN.

Paso	Explicación
Imputación de datos	Puede suceder que los datos provenientes del STAC tengan en algunas de las bandas información faltante, por tanto, se hace necesario realizar un tratamiento a estos datos, en este caso, todos los datos faltantes son llevados a la mediana de cada banda[5].

[5] Es importante aclarar que también pudo utilizarse la media, pero esto se realizó como proceso exploratorio de rendimiento del método de clustering. También es importante aclarar que existen muchas maneras diversas de hacer tratamiento de datos faltantes, existen técnicas multivariadas que pudieron haber sido utilizadas, pero de nuevo, esto solo se hizo de manera exploratoria.

Paso	Explicación
Creación de índices espectrales	Utilizando spyndex se utilizan un total de 150[6] índices espectrales. En el ejemplo se crean tres índices el índice 2 y el índice 3 tienen en este caso una correlación de 1.0
Eliminación de datos correlacionados	Con la finalidad de disminuir la dimensionalidad del dataset se utiliza un filtro para detectar aquellas variables que tienen una alta correlación entre ellas, el filtro es de 0.99. En este caso el índice 3 es eliminado
Escalar datos de manera robusta	Para evitar que datos atípicos de cada una de las variables puedan llegar a generar inconvenientes se utiliza un escalador robusto de datos. Se puede apreciar un cambio de los valores de los datos.
Escalar datos (min-max)	Debido a que los algoritmos de clustering en su mayoría requieren que las diferentes variables tengan similitud de escala se añade este paso.
Imputación de datos	Solo por evitar que existan posibles datos nulos como salida de algunos de los pasos de procesamiento se establece un imputador al final del canal de procesamiento. En este caso no hay cambios.

En la Tabla 11 se muestran el espacio de hiperparámetros seleccionado para los experimentos realizados en esta etapa para que sea más sencilla su interpretación también se ha añadidado el paso o el algoritmo en el cual ha sido utilizado cada uno de ellos y el nombre del argumento en Python.

Tabla 11 Valores del espacio de hiperparámetros para el proceso UMAP y HDBSCAN

Hiperparámetro	Paso	Argumento	Tipo de dato	Rango de valores
Número de componentes	UMAP	n_components	Entero positivo	Entre 8 y 15
Número de vecinos	UMAP	n_neighbors	Entero positivo	Entre 10 y 100
Tamaño mínimo del clúster	HDBSCAN	min_cluster_size	Entero positivo	Entre 100 y 1000

[6] Era la cantidad máxima del paquete spyndex para imágenes Landsat.

Hiperparámetro	Paso	Argumento	Tipo de dato	Rango de valores
Distancia mínima	HDBSCAN	min_dist	Continuo positivo	Entre 0.0 y 0.3
Épsilon	HDBSCAN	cluster_selection_epsilon	Continuo positivo	Entre 0.0 y 0.5

7.2.2.2 PCA y K-Means

Otro canal de procesamiento que se desarrolló fue utilizando PCA como método de disminución de la dimensionalidad y K-means como método de clustering. Si bien PCA cómo asume que las relaciones entre las variables originales (el conjunto a disminuir) son lineales, también es cierto que debido a que existen combinaciones matemáticas en la creación de los índices espectrales o features derivados, esta suposición puede llegar a generar algunos inconvenientes para la generación de los clústeres. Por otra parte, el método de k-means tiene dificultades para agrupar datos que son no tiene una forma esférica y debido a que no se quería tener mayor interacción humana esto supuso una gran complejidad, pues se partía de un supuesto que no era fácilmente verificable: los datos tenían en un espacio multidimensional un comportamiento esférico. Dado lo anterior, se entiende por qué inicialmente se propuso trabajar con UMAP y HDBSCAN. A pesar de las situaciones mencionadas, con este canal de procesamiento es muy importante mencionar que los resultados fueron más eficientes y mucho más satisfactorios al momento de realizar la inspección visual de los clústeres. Para poder hacer una comparación relativamente similar el canal de procesamiento de datos es el mismo que en UMAP y HDBSCAN (Ver Tabla 10) la diferencia radica en los algoritmos de reducción de dimensionalidad y clustering PCA y K-Means. Respecto al espacio de hiperparámetros escogidos es importante mencionar que en sí la cantidad de componentes del PCA no es importante, pues en teoría al utilizarse el puntaje de siluetas. Lo que se busca es la maximización del puntaje de siluetas. Caso contrario sucede con la cantidad de clústeres de K-Means, dadas las suposiciones que realiza el algoritmo es posible que un puntaje de siluetas muy alto se obtenga con muy pocos clústeres, esto es problemático por que puede que en algún momento se agrupen coberturas que son espectralmente diferentes (un ejemplo sería que píxeles de agua y coberturas vegetales

estuvieran en la misma clase). De nuevo, se esperaba que en este paso no existiera mayor intervención humana, por tanto, se decidió partir de una cantidad de 8 clústeres el cual es la cantidad de coberturas reales que se tienen en el estudio[7].

Tabla 12 Valores del espacio de hiperparámetros en PCA y K-Means para cada zona.

Hiperparámetro	Paso	Argumento	Tipo de dato	Rango de valores
Número de componentes	PCA	n_components	Entero positivo	Entre 4 y 16
Cantidad de clústeres	K-Means	n_clusters	Entero positivo	Entre 8 y 25

7.2.3 Interpretación visual de clústeres.

Este fue el proceso más largo en tiempo y probablemente la más importante, en esta fase, se revisaban los diferentes clústeres que estaban presentes en la zona y se le asignaba una de las clases predefinidas. Este procesamiento fue 100% manual y visual. Para ello se utilizó QGIS para realizar la inspección visual de las imágenes lo cual se hizo revisando diferentes combinaciones espectrales y archivo de Excel generado en la fase previa. Es importante mencionar que algunas de las zonas existían clusters que podían llegar a tener diferentes coberturas asociadas (o eso se observaba en la interpretación de las imágenes), para este tipo de píxeles se creó un 'cobertura'[8] a la que se le dio el nombre de 'ruido', esto puede deberse a que en el espacio multidimensional en el cual el algoritmo de k-means es entrenado es posible que estos datos estuvieran muy alejados de algunos de sus clústeres naturales pero más cerca de otros, o tener formas como lunas u óvalos.

7.3 FASE III APRENDIZAJE SUPERVISADO

La finalidad de esta fase es construir y entrenar un modelo que permita realizar la clasificación automática. Los datos de entrada de esta fase fueron todos los archivos de clústeres

[7] En realidad, en el estudio hay 10 posibles clases, pero en una imagen perfecta que tuviera todas las coberturas no habría ni nubes ni sus sombras, por esta razón se decide partir del 8.

[8] Mas que una cobertura realmente es una clase, una etiqueta asignada a esos píxeles.

los cuales fueron concatenados. Ahora bien, al tener todos los archivos de esta manera la cantidad de CPU requerida para el entrenamiento era muy elevada. Por esta razón, se decidió utilizar un muestreo de los píxeles de cada una de las zonas, dicho muestreo se hizo de manera aleatoria, pero se respetó la clase, teniendo en cuenta el estudio realizado por se establece un porcentaje de 10% por cada una de las clases presentes en cada una de las zonas. Paso seguido, se crea un modelo base, sin preprocesamiento, la finalidad es observa el comportamiento de un canal de procesamiento y un algoritmo de clasificación para el conjunto de datos creados. Posteriormente, se realizó una modificación de los hiperparámetros asociados al algoritmo de clasificación para identificar si el conjunto de datos era muy sensible a los hiperparámetros escogidos. Al identificarse que el algoritmo era sensible a los hiperparámetros asociados a los algoritmos se procedió a realizar crear un esquema de validación cruzada.

Sobre el esquema de validación cruzada diseñado es importante mencionar los criterios por los cuales se diseñó de la manera en que se hizo.

- No podía realizar una validación cruzada separando el conjunto de entrenamiento en conjunto de entrenamiento y conjunto de validación de manera aleatoria. La razón de esto es era posible que valores de píxeles de una zona A quedaran en el conjunto de entramiento y en el conjunto de validación lo cual, al ser similares, los resultados podrían llegar a ser demasiado buenos, pero que en realidad son sobreestimados, esto porque esos valores y valores similares ya son conocidos por el modelo.
- Dado lo anterior tampoco era posible que el esquema de validación cruzada se desarrollara dejando un píxel por fuera tipo LOO por que la cantidad de entrenamientos sería muy alta y valores similares ya serían conocidos por el modelo.

Dado lo anterior se decidió crear una validación por grupos y dejando un grupo afuera LOGO.

Este esquema de validación cruzada tiene las siguientes ventajas:

- Todas las coberturas de una zona completa son excluidas del entrenamiento el modelo. El modelo se entrena en las otras zonas, posteriormente se evalúa el comportamiento del modelo en esa zona, de esta manera se pueden tener resultados menos sesgados. En este caso una imagen está representada como la parte subrayada de negro en la Ilustración 2

Los resultados esperados de este proceso son:

- Los hiperparámetros del modelo propuesto que maximizan la exactitud balanceada con la finalidad de que sean utilizados en el prototipo.
- La matriz de confusión que indique el comportamiento del modelo sobre datos no vistos, en este caso todos los píxeles de las zonas del año 2023.

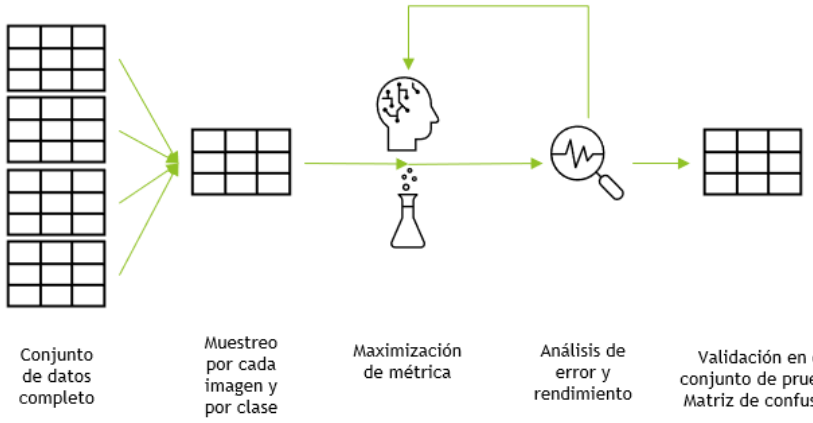


Ilustración 9 Proceso General de la fase III

7.3.1 Entrenamiento del modelo en fase de experimentación.

El modelo propuesto consta de un canal de procesamiento de datos y el algoritmo xgboost. Como datos de entrada se tienen un muestreo de cada una de las imágenes que fueron reclasificadas en la fase anterior, para este muestreo se crea el canal de procesamiento que se puede observar en la Tabla 13.

Tabla 13 Flujo de datos en fase de experimentación.

Paso	Explicación
Imputación de datos	Puede suceder que los datos provenientes del STAC tengan en algunas de las bandas información faltante, por tanto, se hace necesario realizar un tratamiento a estos datos, en este caso, todos los datos faltantes son llevados a la mediana de cada banda[9].

[9] Es importante aclarar que también pudo utilizarse la media, pero esto se realizó como proceso exploratorio de rendimiento del método de clustering. También es importante aclarar que existen muchas maneras diversas de hacer tratamiento de datos faltantes, existen técnicas multivariadas que pudieron haber sido utilizadas, pero de nuevo, esto solo se hizo de manera exploratoria.

Paso	Explicación
Creación de índices espectrales	Utilizando spyndex se utilizan un total de 150[10] índices espectrales. En el ejemplo se crean tres índices el índice 2 y el índice 3 tienen en este caso una correlación de 1.0
Escalar los datos de manera robusto	Para evitar que datos atípicos de cada una de las variables puedan llegar a generar inconvenientes se utiliza un escalador robusto de datos. Se puede apreciar un cambio de los valores de los datos.
Escalar datos (min-max)	Debido a que los algoritmos de clustering en su mayoría requieren que las diferentes variables tengan similitud de escala se añade este paso.
Transformar por cuantiles	Se propone utilizar este paso para que los datos tengan una distribución normal, la idea principal de este paso es facilitar la reducción de dimensionalidad por PCA, que si bien no requiere datos distribuidos de manera normal si puede[11] beneficiarse de este tipo de distribuciones debido a la dirección de la máxima varianza de los datos y si los datos están distribuidos de manera normal la varianza es un indicador confiable de la dispersión de los datos.

Una vez todos los datos han sido transformados se utiliza el algoritmo de xgboost. En esta fase de experimentación se tienen en cuenta diferentes parámetros que pueden ser tenidos en cuenta y que dependiendo de la exactitud de estimación pueden ir cambiando. Como tal en el proceso de optimización de hiperparámetros se toman los siguientes hiperparámetros:

- Del canal de procesamiento de datos solo se optimiza la cantidad de componentes del algoritmo PCA.
- Del algoritmo de clasificación se optimizan los siguiente hiperparámetros: la cantidad de estimadores, la profundidad de cada uno, el peso mínimo cada hijo, la submuestra y gamma.

Debido a que se escogió una validación cruzada por grupos el proceso de entrenamiento va a ser muy costoso computacionalmente, luego, para minimizar el costo se reduce el número de

[10] Era la cantidad máxima del paquete spyndex para imágenes Landsat.

[11] Lo ideal para verificar esta aseveración es entrenar un modelo sin este paso y comparar el resultado, pero como este no es el objetivo del proyecto ha decidido mantenerse.

iteraciones que se van a realizar. El tipo del hiperparámetro, el canal de ejecución, el nombre del hiperparámetro en Python, el tipo de dato y el rango escogido se muestra en la Tabla 14.

Tabla 14 Valores del espacio de hiperparámetros a ser optimizados

Hiperparámetro	Paso	Argumento	Tipo de dato	Rango de valores
Componentes PCA	PCA	n_components	Entero positivo	Entre 2 y 8
Profundidad de los árboles	xgboost	max_depth	Entero positivo	Entre 3 y 20
Iteraciones	xgboost	num_boost_round	Entero positivo	Entre 11 y 100
Peso mínimo del hijo	xgboost	min_child_weight	Entero positivo	Entre 1 y 6
Submuestra	xgboost	subsample	Continuo positivo	Entre 0 y 1
Gamma	xgboost	gamma	Continuo positivo	Entre 0.1 y 0.2

7.3.2 Validación del modelo.

Como se mencionó previamente el dataset de validación final del modelo son todas las zonas del año 2023. Este proceso consistió en la creación de la matriz de confusión de todos los píxeles del modelo 2021 y 2022 contra los del año 2023.

7.4 FASE IV DESARROLLO DEL PROTOTIPO

En esta fase se construye el prototipo propuesto para ello se utiliza el modelo desarrollado en la fase anterior. Se crea una documentación técnica que establece cómo fue creado el modelo y una documentación de uso mediante la una API creada para el mismo. Por otra parte, y para garantizar el correcto funcionamiento del modelo se crean una serie de pruebas unitarias y pruebas de integración que garanticen el correcto funcionamiento de este.



Ilustración 10 Proceso general de la fase IV

Los productos esperados de esta fase son:

- Prototipo desarrollado y desplegado en un repositorio público.

7.4.1 Desarrollo del prototipo.

El lenguaje utilizado para desarrollar el prototipo fue Python y la versión utilizada fue la 3.10. La idea principal detrás del diseño del prototipo era la siguiente:

- Se debía poder entrenar el modelo dentro del mismo.
- El prototipo debía tener un conjunto de pruebas para verificar su funcionamiento.
- Debía tener una API para ser utilizada por un usuario.
- La documentación de la API y del entrenamiento debe ser automatizada lo más posible.

Esta parte del proyecto se dividió en dos pasos, el primero el entrenamiento del modelo con los datos obtenidos del muestreo en la fase III y los hiperparámetros encontrados en el proceso de optimización. El segundo fue el desarrollo de la API que permite a un usuario utilizar el modelo entrenado. La última parte hace referencia a el despliegue y flujo CI/CD del prototipo.

7.4.1.1 Entrenamiento.

Los hiperparámetros del canal de procesamiento y del algoritmo de clasificación son los datos de entrada de esta parte del proyecto. Existía un interés propio por permitir al modelo ir evolucionando y ser mejorado en versiones futuras, esto puede realizarse por cambios en el canal de procesamiento, cambios de algoritmo de clasificación o cambios de datos de entrenamiento, por esta razón, es posible entrenar el modelo dentro del mismo prototipo. Para lograr esto y minimizar la cantidad de mantenimiento requerido sobre el prototipo a medida que se va desarrollando el código necesario para el entrenamiento se crea un conjunto de pruebas unitarias y de integración sobre los diferentes pasos del modelo. Adicionalmente, se dejan parametrizadas ciertas funcionalidades para que la documentación relacionada al entrenamiento del modelo

pueda ser realizada de manera automática. La Ilustración 11 muestra como el entrenamiento del prototipo va ligado a las pruebas automáticas, a la matriz, la documentación y a los hiperparámetros escogidos.

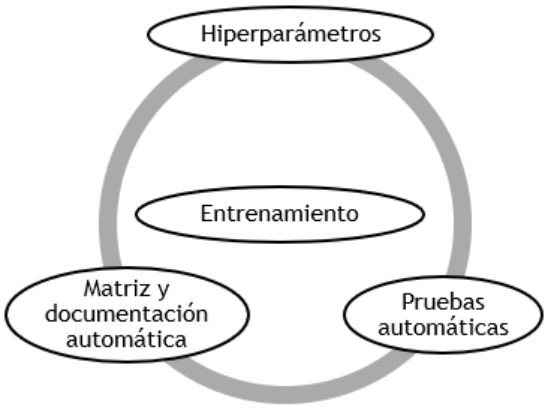


Ilustración 11 Idea central el entrenamiento del modelo en el prototipo

7.4.1.2 Desarrollo de la API

La API fue diseñada de forma tal que el uso del modelo y sus diferentes particularidades esté oculta para el usuario final. La idea central es poder tener una representación de una imagen de Landsat 9. Dicha representación puede ser el identificador de la imagen, las coordenadas de la esquina superior izquierda, la cantidad de pixeles desde la esquina en el eje x y en el eje y (en este caso se toma el mismo para ambas) o un objeto pandas dataframe trabajado y preprocesado por el usuario, lo cual da un poco más de libertad en el flujo de trabajo. La Ilustración 12 muestra el primer caso de representación, como se puede apreciar para el usuario toda la lógica inmersa en el uso del modelo está oculta, cómo tal el usuario solo recibe la clasificación de salida de la imagen solicitada.

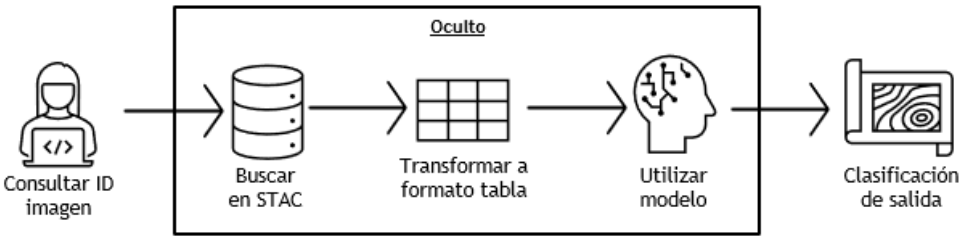


Ilustración 12 Diagrama de uso de la API utilizando STAC

La Ilustración 13 muestra la única clase que será utilizada por el usuario, dicha clase es Landsat9LCClassifier (LC son las siglas Land Cover) esta clase tiene 2 métodos asociados el primero de ellos es make_prediction, el cual recibe un dataframe en el cual esté trabajando el usuario en ese momento (el prototipo realiza las validaciones necesarias para verificar que el dataframe cumpla con las condiciones necesarias). El segundo método es el que utiliza la Ilustración 12 y quizás el mejor método de la clase pues permite acceder a STAC y realizar procesamientos sobre la imagen y obtener directamente un xarray que es un mejor tipo de dato para análisis para imágenes.

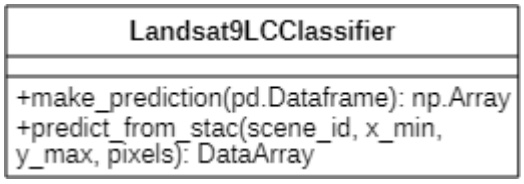


Ilustración 13 Clase principal del prototipo

7.4.1.3 Pruebas unitarias y de integración

El marco de trabajo para realizar las pruebas unitarias fue pytest. El conjunto de pruebas no es realmente grande pero revisa procesos claves en el desarrollo de la ejecución y utilización de la API, al crearlas la idea centrar fue verificar las entradas de la API, al igual que sus salidas y ciertas partes intermedias que son fundamentales, la verificación de tipos de datos asociados a estas partes intermedias no se realiza ya que se busca en términos generales el correcto funcionamiento de la API y algunas partes intermedias pueden optimizarse mucho más a como están diseñadas en este momento. En términos generales el sistema de pruebas valida que:

- Finalmente se valida que no exista la etiqueta Ruido en el dataset de entrenamiento.
- Los datos de entrada de clase diseñada sean los correctos.
- La cantidad de registros de las predicciones se la misma que la cantidad de datos de entrada.
- Si los datos de entrada no son los correctos se le dé al usuario un mensaje claro.
- Si se hace uso de la API que llama los datos desde el STAC se valida que los datos obtenidos sean los correctos y estén acorde a los valores ingresados por el usuario.

- Se valida que las predicciones realizadas utilizando STAC sean las correctas (en cantidad, en dimensiones y en tipo de dato de salida).
- Se valida que las predicciones realizadas utilizando de un dato tipo pandas creado por el usuario sean las correctas (en cantidad, en dimensiones y en tipo de dato de salida).

7.4.1.4 Documentación

Para realizar la documentación de la API se utilizó mkdocs como forma de generar automáticamente una interfaz web que permita al usuario saber cómo se debe utilizar el paquete. La ventaja de realizarlo de esta manera es que la documentación se realiza dependiendo de los diferentes cambios que vaya teniendo el prototipo en todo su ciclo de vida. Dicha documentación web sale explícitamente de la documentación del formato docstring de cada clase y cada método realizado en el prototipo.

7.4.2 Repositorio nube, CI/CD y publicación

Debido a algunas de las ventajas que tiene la plataforma GitHub, se decide alojar el prototipo en dicha plataforma, de esta forma es más fácil mantener versionado el código del prototipo y el modelo. Adicionalmente, GitHub cuenta con un flujo que permite crear integración y despliegue continuó de manera que las mismas políticas del prototipo creadas en las pruebas siempre sean cumplidas antes de unir el código de un usuario al repositorio, esto permite aumentar la confiabilidad del usuario sobre los procesos que el prototipo desarrolla. Una vez que todas las pruebas son superadas el paquete se carga a una plataforma paquetes de Python, en este caso la plataforma escogida es pypi.

8. DESARROLLO METODOLÓGICO.

En este capítulo se presentan el desarrollo metodológico de la investigación y los productos intermedios obtenidos realizando un análisis descriptivo y cuantitativo de los mismos.

En total, se seleccionaron 38 imágenes, sin embargo, como las imágenes cubren un área bastante amplia se podían seleccionar de una imagen diferentes zonas, por tanto, al de final estas 38 imágenes de satélite se obtienen 58 zonas. La Ilustración 14 muestra imágenes de las diferentes zonas de estudio. En la Tabla 15 se pueden revisar tanto el identificador asignado de la imagen como la cantidad de zonas escogidas en cada una de las imágenes y las coordenadas de la esquina superior izquierda y el sistema de referencia. Respecto a las zonas de estudio también es importante mencionar que se escogieron zonas que fueran cuadradas misma cantidad de píxeles a los largo y a lo ancho el cual es de 800 píxeles, esto se hizo por conveniencia para el trabajo pensando en los recursos computacionales que podían llegar a consumirse en cada uno de los procesos, de hecho, es importante mencionar que se escogieron zonas que no estuvieran cerca a los bordes de cada imagen satelital, esto debido a que las transformaciones de dimensiones de las imágenes en forma (banda, pixeles x, pixeles y) a formato tabular[12] podía ser un poco más engorroso en esas zonas debido a la cantidad de datos faltantes[13].

[12] Ver **Error! Reference source not found.** y Tabla 1 para una explicación detallada.

[13] Por lo general, las imágenes de satélite son rectangulares o cuadradas, pero la cobertura que muestran se observa en forma de rombo, sin embargo, las zonas que no tienen información se muestran como datos vacíos o faltantes en los diferentes softwares.

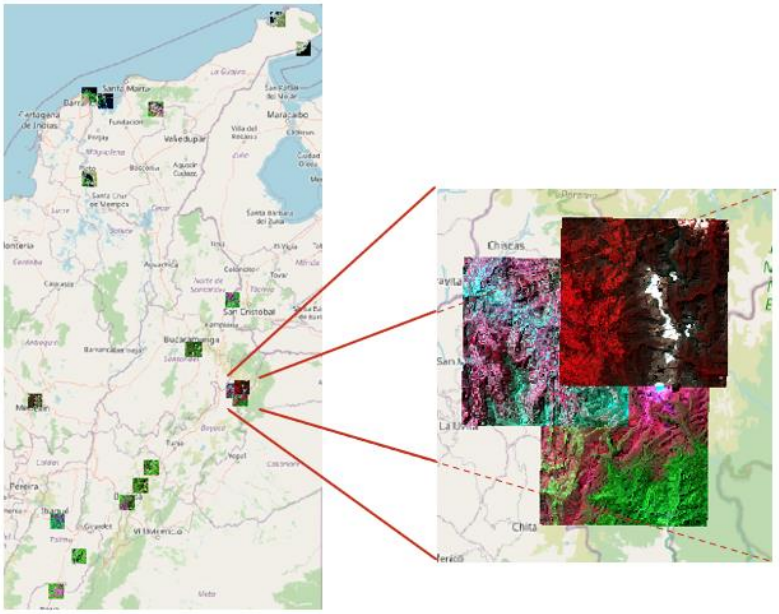


Ilustración 14 Distribución espacial de las zonas de estudio.

Ahora bien, es importante recordar que no solo se quería realizar un muestreo espacialmente distribuido, sino que también se quería tienen una temporalidad asociado a dicho muestreo, por tanto, existen zonas sobrepuestas, observando la Ilustración 14 se puede apreciar como sobre el parque nevado El Cocuy se sobreponen 3 imágenes, esta sobreposición no solo es porque de una imagen de satélite se pueden escoger múltiples zonas, sino que también pueden zonas de diferentes fechas, esto permite obtener información de cómo varían los valores de los píxeles para una determinada cobertura y permite al modelo desarrollado conocer valores nuevos (en caso de que existan cambios) de píxeles para una misma cobertura en una misma zona. La Ilustración 15 muestra la cantidad de zonas recolectadas por año, en total se obtuvieron 11 imágenes para el año 2021, 26 imágenes para el año 2026 y 21 imágenes para el año 2023. Ahora bien, teniendo en cuenta que cada zona es compuesta de una imagen de 800 * 800 píxeles y que son 58 zonas se obtiene un total de 37’120.000 de píxeles en cada banda, pero teniendo en cuenta que son 8 bandas espectrales se tiene un total de 128’000.000 de píxeles en el conjunto de datos de muestreo. En todas las zonas hay un total de 296’960.000 de píxeles.



Ilustración 15 Cantidad de zonas por año.

Una vez revisadas las imágenes se procede a realizar la definición de coberturas presentes en ellas. Las coberturas definidas fueron:

1. Cuerpos de agua.
2. Bosques.
3. Hielo / Nieve
4. Nubes y sus sombras asociadas
5. Zonas de cultivo
6. Áreas abiertas con poca vegetación
7. Áreas abiertas con vegetación.
8. Areas artificiales.
9. Zonas Húmedas.

Tabla 15 Información de cada imagen y cantidad de zonas.

ID	imagen	Zonas	año	path	row	coord_x	coord_y	SRE
1	LC09_L2SP_007052_20211207_02_T1	1	2021	7	52	182774	1366356	EPSG:32619
2	LC09_L2SP_007052_20211223_02_T1	1	2021	7	52	182774	1366356	EPSG:32619
3	LC09_L2SP_007052_20220124_02_T1	2	2022	7	52	182774	1366356	EPSG:32619
4	LC09_L2SP_007052_20220124_02_T1		2022	7	52	227384	1315842	EPSG:32619
5	LC09_L2SP_007052_20220209_02_T1	1	2022	7	52	182774	1366356	EPSG:32619
6	LC09_L2SP_007052_20220414_02_T1	1	2022	7	52	227384	1315842	EPSG:32619
7	LC09_L2SP_007052_20220516_02_T1	1	2022	7	52	227384	1315842	EPSG:32619
8	LC09_L2SP_007052_20220601_02_T1	2	2022	7	52	182774	1366356	EPSG:32619
9	LC09_L2SP_007052_20220601_02_T1		2022	7	52	227384	1315842	EPSG:32619

ID	imagen	Zonas	año	path	row	coord_x	coord_y	SRE
10	LC09_L2SP_007052_20220703_02_T1	2	2022	7	52	227384	1315842	EPSG:32619
11	LC09_L2SP_007052_20220703_02_T1		2022	7	52	182774	1366356	EPSG:32619
12	LC09_L2SP_007055_20211207_02_T1	2	2021	7	55	702000	793569	EPSG:32618
13	LC09_L2SP_007055_20211207_02_T1		2021	7	55	784562	729161	EPSG:32618
14	LC09_L2SP_007055_20220124_02_T1	1	2022	7	55	784562	729161	EPSG:32618
15	LC09_L2SP_007055_20220225_02_T1	1	2022	7	55	784562	729161	EPSG:32618
16	LC09_L2SP_007055_20221226_02_T1	3	2022	7	55	702000	793569	EPSG:32618
17	LC09_L2SP_007055_20221226_02_T1		2022	7	55	766363	880176	EPSG:32618
18	LC09_L2SP_007055_20221226_02_T1		2022	7	55	784562	729161	EPSG:32618
19	LC09_L2SP_007055_20230127_02_T1	4	2023	7	55	698427	793569	EPSG:32618
20	LC09_L2SP_007055_20230127_02_T1		2023	7	55	702000	793569	EPSG:32618
21	LC09_L2SP_007055_20230127_02_T1		2023	7	55	766363	880176	EPSG:32618
22	LC09_L2SP_007055_20230127_02_T1		2023	7	55	784562	729161	EPSG:32618
23	LC09_L2SP_007056_20230127_02_T1	2	2023	7	56	770376	723173	EPSG:32618
24	LC09_L2SP_007056_20230127_02_T1		2023	7	56	781866	708924	EPSG:32618
25	LC09_L2SP_008053_20211118_02_T1	1	2021	8	53	629049	1206997	EPSG:32618
26	LC09_L2SP_008053_20211214_02_T1	1	2021	8	53	629049	1206997	EPSG:32618
27	LC09_L2SP_008053_20211230_02_T1	1	2021	8	53	629049	1206997	EPSG:32618
28	LC09_L2SP_008053_20220115_02_T1	1	2022	8	53	629049	1206997	EPSG:32618
29	LC09_L2SP_008053_20220131_02_T1	1	2022	8	53	629049	1206997	EPSG:32618
30	LC09_L2SP_008053_20221217_02_T1	1	2022	8	53	629049	1206997	EPSG:32618
31	LC09_L2SP_008053_20230102_02_T1	1	2023	8	53	629049	1206997	EPSG:32618
32	LC09_L2SP_008053_20230203_02_T1	1	2023	8	53	629049	1206997	EPSG:32618
33	LC09_L2SP_008053_20230219_02_T1	1	2023	8	53	629049	1206997	EPSG:32618
34	LC09_L2SP_008056_20221201_02_T1	1	2022	8	56	630345	589144	EPSG:32618
35	LC09_L2SP_008057_20211214_02_T1	2	2021	8	57	589042	528101	EPSG:32618
36	LC09_L2SP_008057_20211214_02_T1		2021	8	57	612043	558180	EPSG:32618
37	LC09_L2SP_008057_20220131_02_T1	2	2022	8	57	612043	558180	EPSG:32618
38	LC09_L2SP_008057_20220131_02_T1		2022	8	57	589042	528101	EPSG:32618
39	LC09_L2SP_008057_20220811_02_T1	1	2022	8	57	469415	494891	EPSG:32618
40	LC09_L2SP_008057_20230203_02_T1	3	2023	8	57	506774	434582	EPSG:32618
41	LC09_L2SP_008057_20230203_02_T1		2023	8	57	589042	528101	EPSG:32618
42	LC09_L2SP_008057_20230203_02_T1		2023	8	57	612043	558180	EPSG:32618
43	LC09_L2SP_008057_20230219_02_T1	2	2023	8	57	612043	558180	EPSG:32618
44	LC09_L2SP_008057_20230219_02_T1		2023	8	57	589042	528101	EPSG:32618
45	LC09_L2SP_008058_20230203_02_T1	1	2023	8	58	468391	374256	EPSG:32618
46	LC09_L2SP_009052_20211205_02_T1	2	2021	9	52	514456	1229380	EPSG:32618
47	LC09_L2SP_009052_20211205_02_T1		2021	9	52	542269	1218807	EPSG:32618
48	LC09_L2SP_009052_20220106_02_T1	2	2022	9	52	514456	1229380	EPSG:32618

ID	imagen	Zonas	año	path	row	coord_x	coord_y	SRE
49	LC09_L2SP_009052_20220106_02_T1		2022	9	52	542269	1218807	EPSG:32618
50	LC09_L2SP_009052_20220122_02_T1	2	2022	9	52	514456	1229380	EPSG:32618
51	LC09_L2SP_009052_20220122_02_T1		2022	9	52	542269	1218807	EPSG:32618
52	LC09_L2SP_009052_20230210_02_T1	2	2023	9	52	514456	1229380	EPSG:32618
53	LC09_L2SP_009052_20230210_02_T1		2023	9	52	542269	1218807	EPSG:32618
54	LC09_L2SP_009052_20230226_02_T1	2	2023	9	52	542269	1218807	EPSG:32618
55	LC09_L2SP_009052_20230226_02_T1		2023	9	52	514456	1229380	EPSG:32618
56	LC09_L2SP_009053_20230210_02_T1	1	2023	9	53	516952	1084640	EPSG:32618
57	LC09_L2SP_009053_20230226_02_T1	1	2023	9	53	516952	1084640	EPSG:32618
58	LC09_L2SP_009056_20220311_02_T1	1	2022	9	56	429016	701537	EPSG:32618

8.1 UMAP Y HDBSCAN.

El proceso de umap y hdbscan tuvo varios inconvenientes, tanto en términos computacionales como en el resultado del clustering. En la Tabla 16 se muestran los resultados previos del uso del flujo de procesamiento umap y hdbscan comparados frente los de PCA y k means, en esta sección se hablará únicamente de la primera columna, en la parte izquierda se muestra el resultado del procesamiento y en el centro se muestra la imagen en rgb754, los colores de la representación de los clústeres no son importantes, excepto el color negro, que se dejó de esta manera ya que es el ruido detectado por hdbscan, como se puede evidenciar en las dos primeras imágenes este tipo de modelos no lograba diferenciar diferentes coberturas que son claramente visibles en las dos primeras imágenes centrales, como la zona con poca vegetación, existen zonas donde el agua es identificada como ruido, inclusive existen varios clústeres a los que a posteriori se les hubiese asignado dicha clase; en la tercera imagen la presencia de ruido es excesiva, prácticamente toda las zonas con poca vegetación son identificadas como ruido. Estos resultados provienen de una hiperparametrización de cada una de las imágenes y los hiperparámetros propuestos en la Tabla 11.

Tomando por ejemplo diferentes hiperparámetros de la primera imagen de la Tabla 16, y su diferente comportamiento se obtienen los resultados de la Ilustración 16. Esta ilustración que es una gráfica de coordenadas paralelas, muestra cómo va cambiando un valor objetivo (en este caso el DBCV) en función de los hiperparámetros escogidos en el momento de la optimización; cada barra del eje x muestra un hiperparámetro los valores en el eje y muestran el valor del

hiperparámetro y las líneas azules muestran las combinaciones, como se puede apreciar, en este caso se tienen resultado cercanos a 1 lo cual es lo indicado por la métrica DBCV, sin embargo, los resultados visualmente distan mucho de ser óptimos. Este error es muy complejo de depurar y existe una multitud de factores que pueden tener este efecto, incluso puede que el problema no sea como tal HDBSCAN si no el algoritmo UMAP). Los resultados detallados para las tres imágenes pueden verse en el Anexo 2.

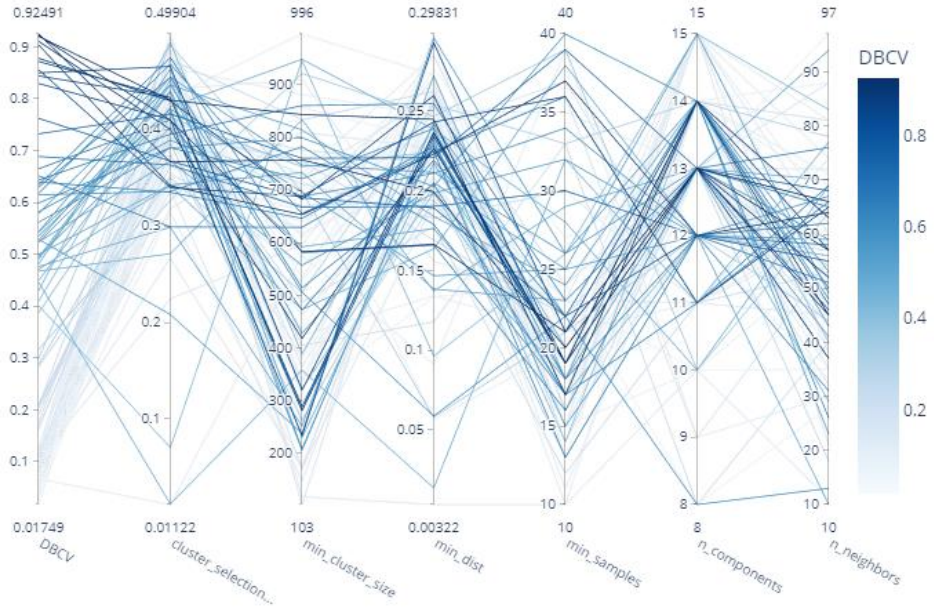
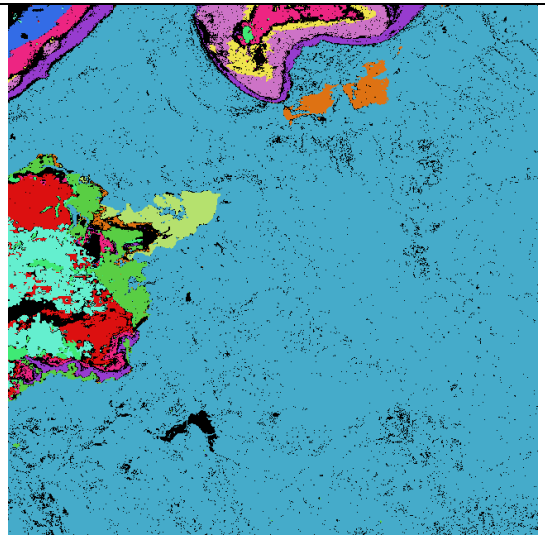
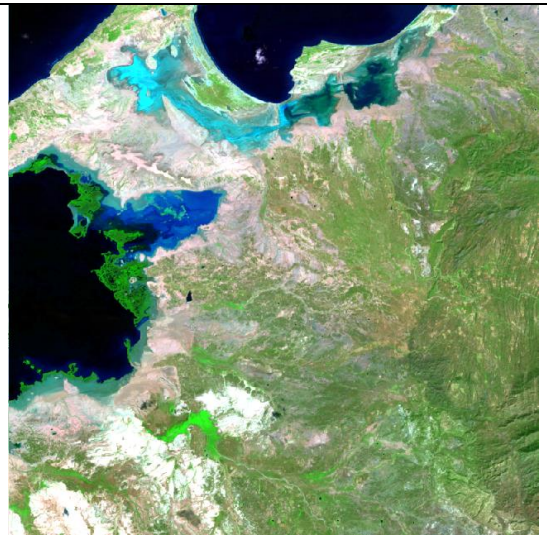
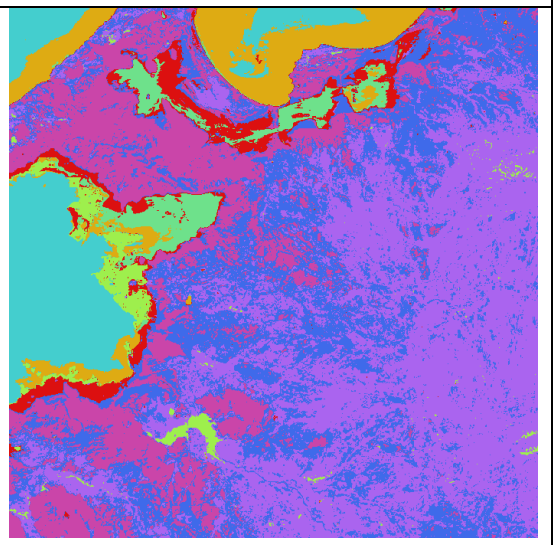
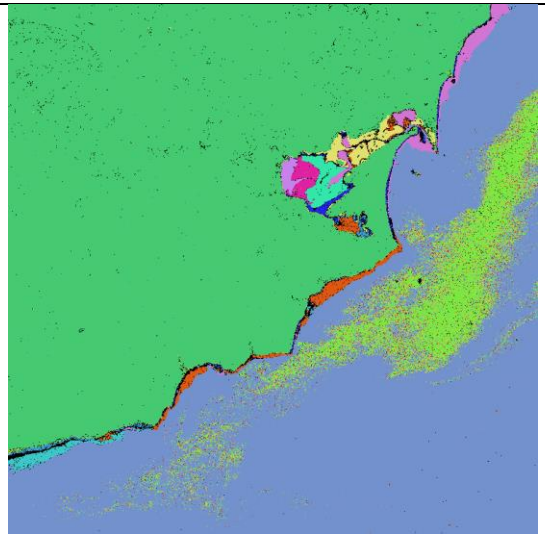
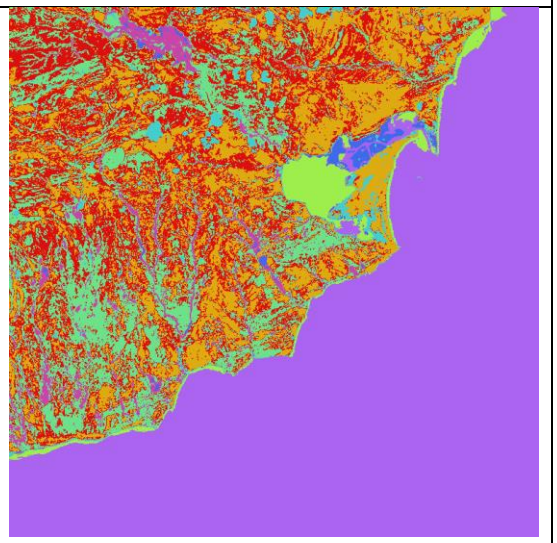
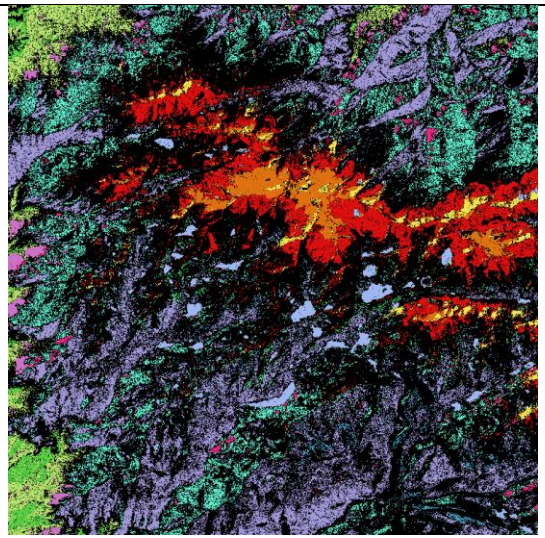
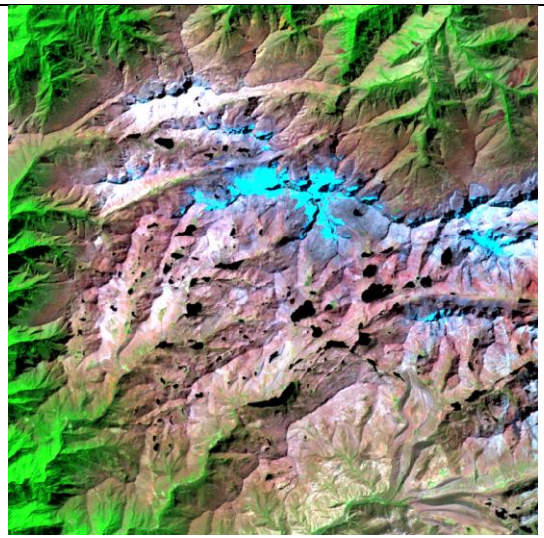
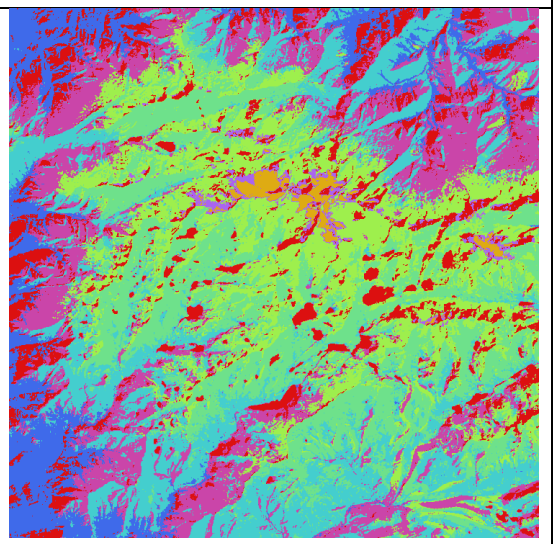


Ilustración 16 Coordenadas paralelas umap - hdbscan imagen con ID 1.

Respecto a lo dicho previamente, en el sentido de que puede que el error no esté asociado directamente al proceso de clustering sino al proceso de reducción de dimensionalidad, se realiza un análisis de una reducción de dimensionalidad para una imagen al azar del dataset de entrenamiento para visualizar cómo es el comportamiento del algoritmo.

Tabla 16 Resultados con UMAP y HDBSCAN vs PCA y K Means

ID	Imagen UMAP-HDBSCAN	Imagen RGB754	Imagen PCA - Kmeans
LC09_L2SP_007052_20211207_02_T1			
LC09_L2SP_007052_20220124_02_T1			

ID	Imagen UMAP-HDBSCAN	Imagen RGB754	Imagen PCA - Kmeans
LC09_L2SP_008053_20230203_02_T1			

En la Ilustración 17 se muestra la representación de una imagen del dataset reducida a dos dimensiones utilizando umap, cada punto representa un píxel, en el eje x está la representación de la primera dimensión y en el eje y la representación de la segunda, al costado izquierdo en representación RGB 543 y al costado derecho en RGB 754. Como puede apreciarse la representación en dos dimensiones no es del todo clara y existe mezcla de píxeles, quizás las agrupaciones más estables o consistentes son las de agua (en tonos azules en ambos costados), mientras que la vegetación y zonas de poca vegetación no logran diferenciarse con claridad, de hecho, se puede apreciar como existe una zona que corta en el centro de las imágenes los píxeles de vegetación (de color rojo al costado derecho y de color verde a costado izquierdo). Esto puede dar una indicación clara de que el problema mostrado en este tipo de modelado está presente como tal en umap y no tanto en hdbscan y es, además, una razón válida para no seguir experimentando en esta investigación con este tipo de modelado.

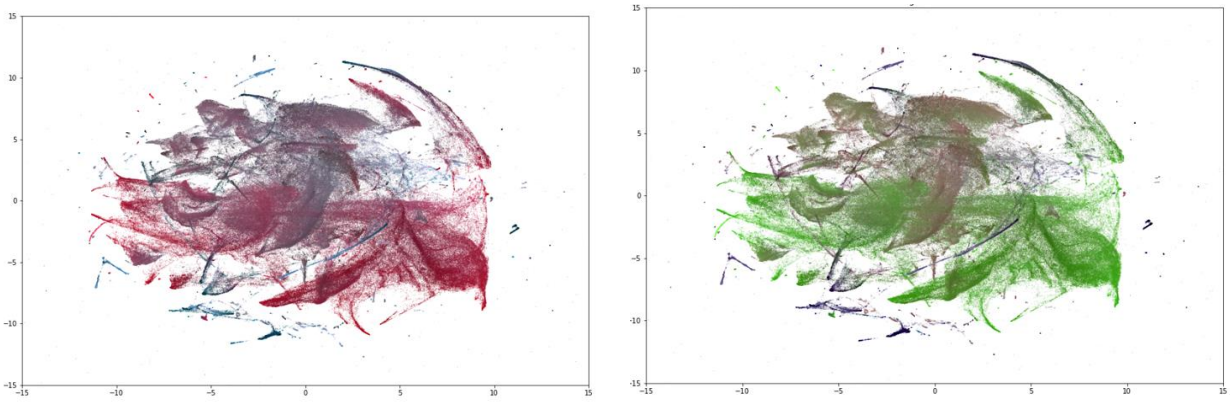


Ilustración 17 Visualización de reducción de la dimensionalidad con umap.

Ahora bien, puede decirse que la representación en dos dimensiones es quizás, no lo suficiente para el modelado y que entra en contraposición con los datos mostrados en la Ilustración 16, lo cual es cierto, pues la dimensionalidad mínima representada allí es de 8 componentes / dimensiones, sin embargo, este análisis se realizó para poder visualizar de manera simple un resultado real del algoritmo umap, lo cual no es posible en más de tres dimensiones.

8.1 PCA Y K-MEANS

El proceso de PCA y K-means tuvo un componente importante asociado a su ejecución y fue la rapidez con la que se ejecutaba la reducción de la dimensionalidad y proceso de clustering, proceso que requería casi un 50% del tiempo total de la metodología anterior.

Como se puede apreciar en la Tabla 16 este tipo de modelo identifica mejor las coberturas presentes en la imagen y es un poco más estable, se puede visualizar fácilmente algunos cambios de coberturas en todas las imágenes presentes. En la zona de nevados también se logra evidenciar cómo logra agrupar de manera eficiente el nevado, nótese además que comparando los resultados frente a umap y hdbscan todas las zonas detectadas como ruido pueden ser aparentemente[14] asignadas a clases de suelo con poca vegetación. Ahora bien, para hablar de los resultados de esta imagen se puede tomar de referencia los datos resultantes del proceso de optimización de hiperparámetros en la Ilustración 18, nótese cómo el algoritmo tiene tomar valores pequeños tanto para la cantidad de componentes como para la cantidad de clusters, es posible que al disminuir aún más estos valores se obtenga valores más altos del índice de siluetas, pero, cómo bien se observó en el caso de umap y hdbscan no necesariamente los resultados sea buenos, pues al disminuir la cantidad de clusters, disminuirá también la posibilidad de distinguir entre diferentes tipos de coberturas, no lo cual no es lo ideal.

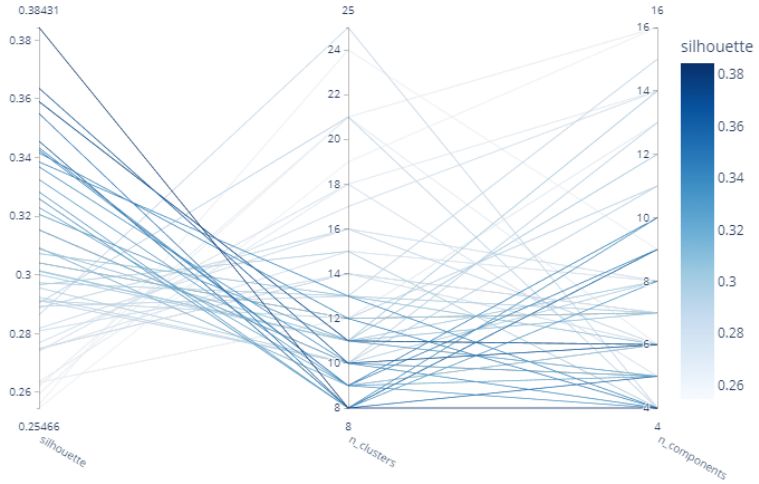


Ilustración 18 Coordenadas paralelas PCA – k means imagen con ID 1.

[14] Desde luego, para cada zona toca hacer la interpretación independiente.

Ahora bien, estos son solo los valores de una de las zonas de estudio, los resultados completos de este proceso pueden ser visualizados en el Anexo 3[15]. Adicionalmente, al realizar un análisis del comportamiento de las variables del proceso de hiperparametrización se observa que la cantidad de clústeres predominante para los mejores conjuntos de hiperparámetros es 8 en la cual se puede observar un repunte de la cantidad de zonas que utilizan esta cantidad de agrupaciones, por otra parte, se observa como existe una tendencia a mantener relativamente baja la cantidad de componentes principales, lo cual puede ser divisado en la Ilustración 19.

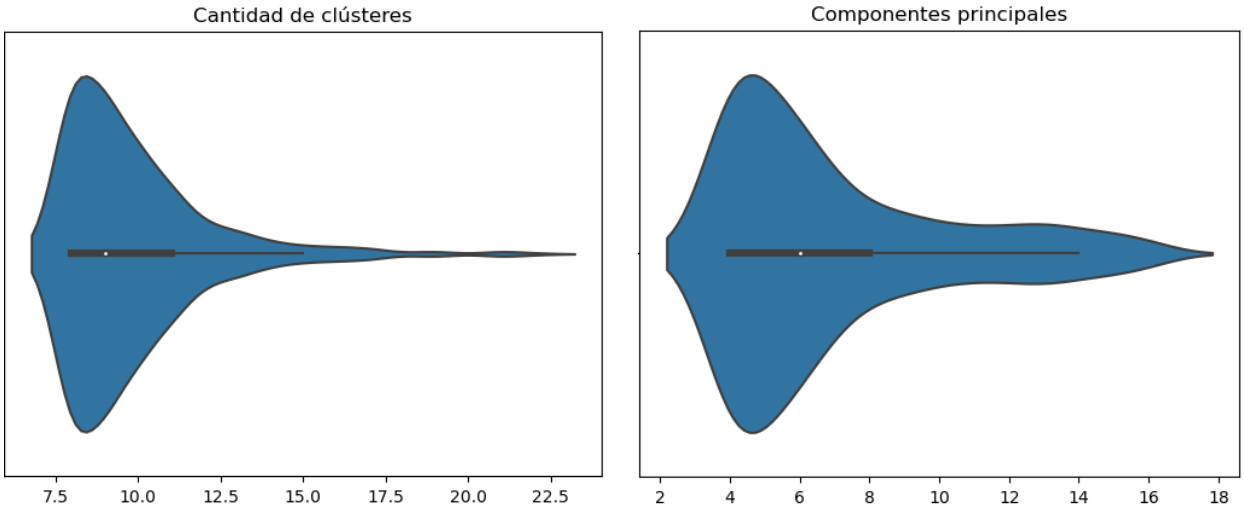


Ilustración 19 Comportamiento de los mejores hiperparámetros en PCA y K means.

Para poder constatar la forma en la cual se concentran los diferentes hiperparámetros por las zonas es necesario revisar la Ilustración 20, en esta gráfica queda claro que este tipo de modelos tiende a mantener bajos tanto la cantidad de componentes como la cantidad de clústeres que maximicen el índice de silueta. Esto puede ser inicialmente poco benéfico pues puede indicar que con pocos clústeres se tiene una muy buena identificación de estos lo cual puede ser dependido del caso falso, especialmente grave sería el caso en el cual un único clúster fuera identificado en una zona de estudio.

[15] Con la finalidad de disminuir espacio solo se tuvieron en cuenta solo los diez mejores resultados por cada zona.

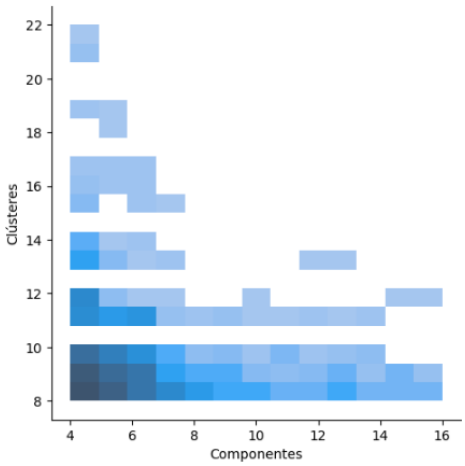


Ilustración 20 Concentración de clústeres por hiperparámetros

Vale destacar el comportamiento del índice de siluetas obtenido, se puede apreciar que la media está ubicada en 0.35 y la media en 0.32, se tienen valores que sobrepasan el un valor de 0.5 lo cual es un indicador confiable para las zonas para las cuales ese haya sido su puntuaje. En la Ilustración 21 se puede apreciar la distribución de los mejores puntajes de silueta para todas las imágenes.

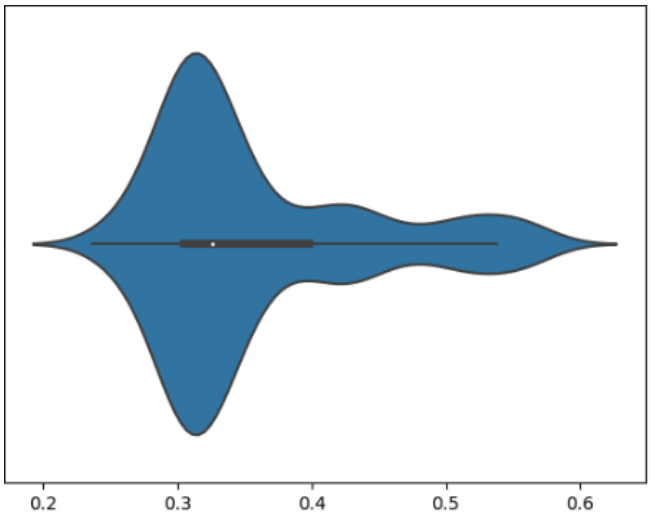
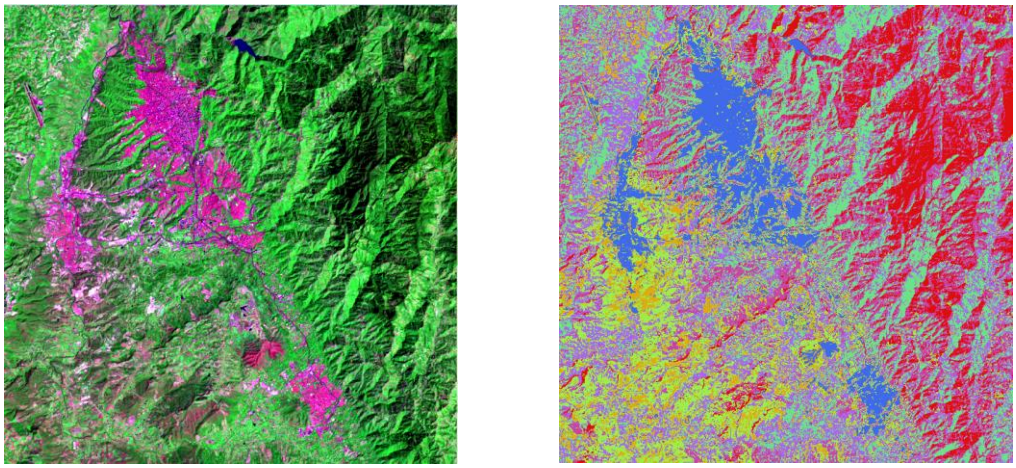


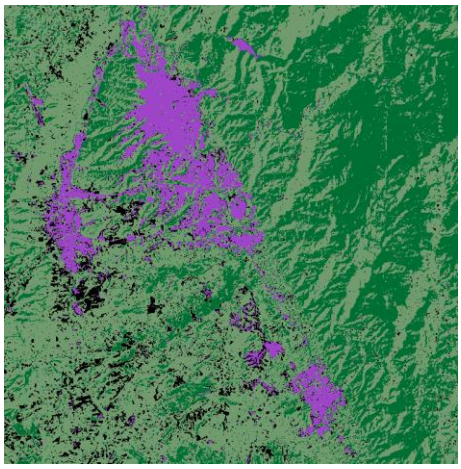
Ilustración 21 Índice de siluetas para los mejores hiperparámetros

Posterior a este proceso debe realizarse la asignación de los clústeres a las coberturas definidas, es importante recalcar que solo se mencionarán algunos ejemplos con la idea de que el lector logre identificar algunas situaciones que podían presentarse en cada una de las zonas y del manejo que se le daba a cada situación. Para la zona identificada con el id 19 de la Tabla 15 la cual se encuentra ubicada en la ciudad de Bucaramanga (Ilustración 22), para esta zona la

cantidad de clusters fue de 8, se puede apreciar como existe un cuerpo agua el cual es confundido como cobertura de área artificial (cluster de color azul), si bien este cluster podría ser identificado como ruido, dejar perder esa cantidad de píxeles de área artificial también podría ser perjudicial, es por eso que en la interpretación de las coberturas se asigna este cuerpo de agua como una zona de artificial (añadiendo ruido al dataset), este resultado sin lugar a dudas tendrá un impacto directo en el entrenamiento del modelo y los resultados del mismo. Existen también algunas mezclas de coberturas especialmente para zonas que en la inspección visual son suelo con poca/nula vegetación, con algunos píxeles que son de suelo con vegetación/pastizales, sin embargo, la cantidad de píxeles asociados a este tipo de ruido son pocos, y por eso se asigna la cobertura de suelo con poca vegetación de color verde claro en la imagen inferior de la. Existen algunos píxeles que fueron clasificados como ruido y que no serán tenidos en cuenta en el dataset, debido a que la cantidad de píxeles identificados como mezcla de coberturas es amplia, por tanto, se decide no asignarle ninguna cobertura. El puntaje de silueta para esta zona es de 0.27. En esta sección las imágenes con subtítulo a hacen referencia a la imagen en RGB 754 el cual permite visualizar en tonalidades verdes la vegetación, el suelo desnudo y las zonas artificiales en tonalidades moradas y los cuerpos de agua en tonalidades azules. La imagen con subtitulo b hace referencia a los clústeres encontrados por el modelo de la imagen, los colores de estas zonas son aleatorios pues no representan aún una clase. Finalmente, las imágenes de subtitulo c representan la asignación a clases de la investigación; en tonalidades de verde oscuro se pueden observar las zonas boscosas, de verde claro las zonas de áreas abiertas con vegetación, de color morado las zonas artificiales, de color naranja los suelos desnudos, de negro las sombras de nubes (también incluye el ruido, aunque la coloración es solo para la representación) y de color azul los diferentes cuerpos de agua.



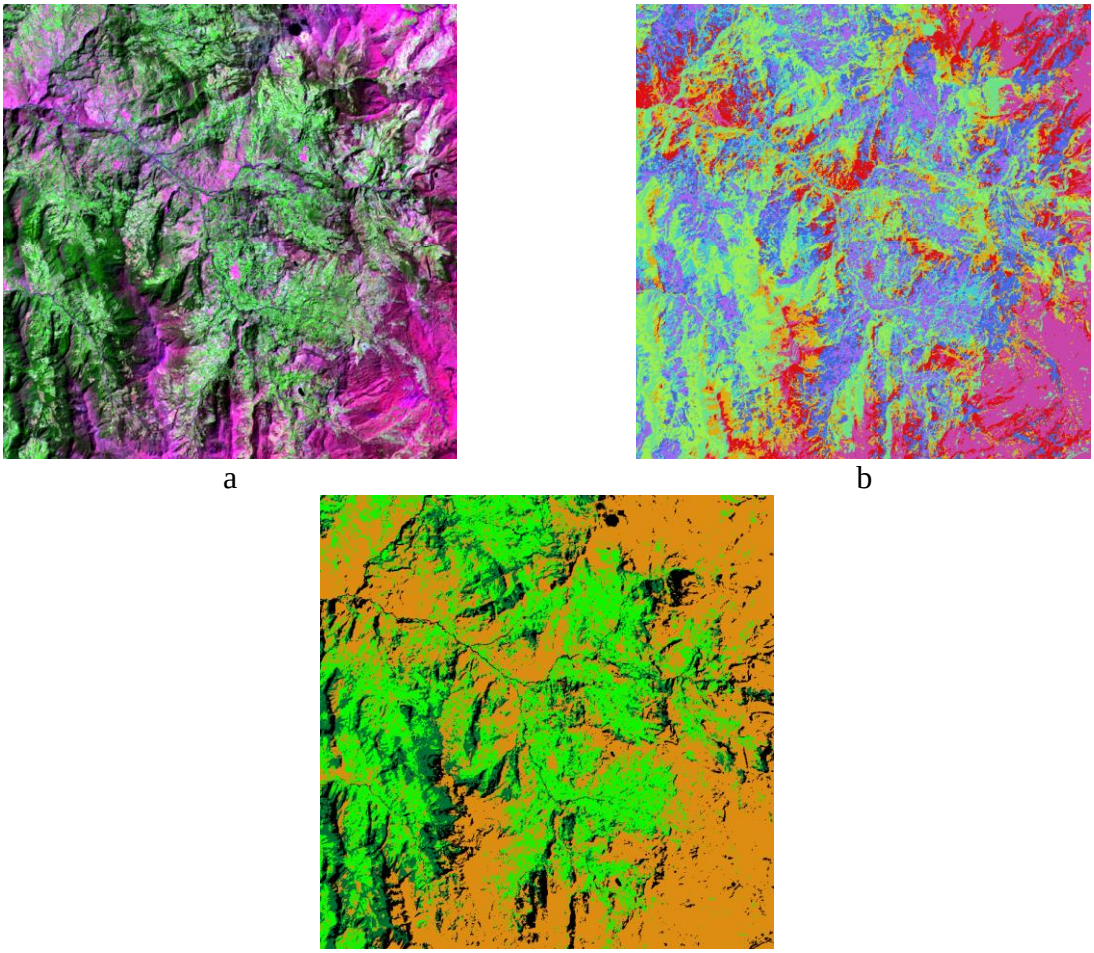
a b



c

Ilustración 22 Interpretación de coberturas de clústeres para la zona 19

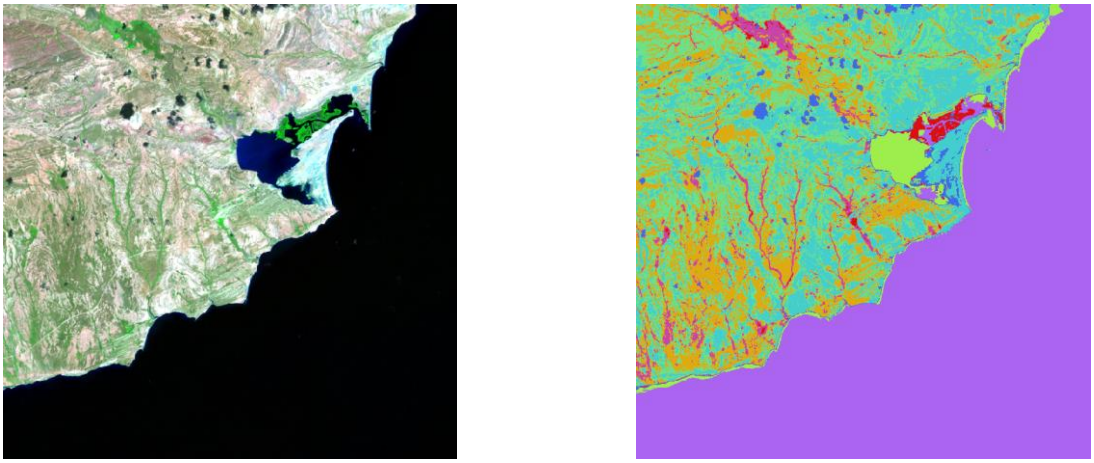
La segunda zona con el índice más es la identificada con el id 23 de la Tabla 15 esta zona está ubicada en el municipio de Guacamayas en el departamento de Boyacá. Para esta imagen la cantidad de clústeres identificada como óptima fue de 8, en este caso existen zonas con poca vegetación o suelo desnudo, zonas boscosas que fueron bien identificadas, existieron algunas mezclas de píxeles de zonas de bosque con territorios agrícolas, pero la cantidad de mezclas en la inspección visual no fue determinante como para asignar ruido, sin embargo, si existieron para el cluster identificado como 0 mucha mezcla de cobertura y por esta razón se le asigno la clase de Ruido. En términos generales, a pesar de su bajo índice de silueta el cual es de 0.28 los resultados no son realmente malos, lo anterior puede constatarse al observa la Ilustración 23.



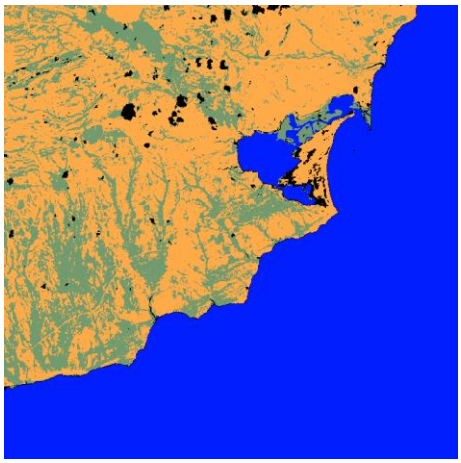
a b c

Ilustración 23 Interpretación de coberturas de clústeres para la zona 23

Ahora bien, si se toma el mejor índice de silueta el cual fue la identificado con el id 4 de la Tabla 15 con un puntaje de 0.58. La zona está ubicada en el departamento Guajira y cubre la zona la laguna de cocineta. Es importante notar que esta zona fue utilizada para analizar los resultados de umap y hdbscan en la Tabla 16, como se puede apreciar en esta zona existe nubosidad, sin embargo el modelo no logra diferenciarla de manera eficiente (lo que parecen ser la nubes en la imagen de los clústeres en realidad son las sombras generadas del mismo), sin embargo si se aprecia la zona cercana a la laguna de cocineta mezcla algunas de estas coberturas con otros tipos (agua y suelo), por tanto a todos estos píxeles se les asigna la clase de Ruido, sobre el agua de mar se puede decir que logra una buena estimación incluso el borde de la costa y la laguna de cocineta, respecto a la vegetación se puede apreciar que logra diferenciar el suelo con poca vegetación y los suelos con vegetación, incluso los manglares.



a b



c

Ilustración 24 Interpretación de coberturas de clústeres para la zona 4

Este tipo de interpretaciones se realizaron para todas las imágenes, combinando y cambiando diferentes canales para poder ser lo más precisos posibles en la identificación de las coberturas de cada una de las zonas, pero como puede intuirse cada zona tenía sus particularidades y dificultades propias.

Para lograr construir el modelo una vez asignadas las clases a todos los clústeres de todas las zonas y con la finalidad de disminuir la cantidad de recursos de máquina se realiza un muestreo del 10%[16] de cada uno de los clústeres de cada una de las zonas, se realiza sobre los clústeres y no sobre las coberturas para poder tener diferentes comportamientos de una misma cobertura. Debido al hecho de que el esquema de validación utilizado en el proceso de optimización de

[16] Al momento de realizar este documento se intentó realizar con todo el dataset, al igual que con 50%, 25% y 20%, pero en todos los casos la máquina quedaba sin recursos y terminaba el proceso.

hiperparámetros fue por grupos se toma un valor de 5 divisiones por grupos para cada conjunto de hiperparámetros y la cantidad de iteraciones escogidas fue de 100, esto quiere decir que en total se entrenan 500 modelos diferentes en el conjunto de entrenamiento. A continuación, se muestran los resultados para los cinco mejores experimentos realizados, en ellos se detallan las diferentes métricas los cuales permitirán mostrar ciertos comportamientos de todos los modelos desarrollados. El mejor resultado fueron los hiperparámetros del experimento número 98.

Tabla 17 Cinco mejores experimentos en validación por grupos

	98	**95**	**99**	**27**	**79**
exactitud balanceada validación	0.749103	0.748562	0.748084	0.747983	0.747544
exactitud balanceada entrenamiento	0.870682	0.865065	0.883589	0.87297	0.866764
gamma	0.112657	0.103297	0.11998	0.131344	0.111068
max_depth	5	4	8	6	4
min_child_weight	6	6	6	6	3
n_estimators	39	39	16	46	34
subsample	0.014997	0.01749	0.020334	0.011265	0.188737
f1 entrenamiento	0.814006	0.803615	0.83733	0.82323	0.800048
f1 validación	0.735478	0.731635	0.734912	0.736612	0.729184
precision entrenamiento	0.83782	0.830163	0.854905	0.84368	0.828649
precision validación	0.773106	0.772971	0.770093	0.773259	0.772945
recall entrenamiento	0.810489	0.799901	0.834704	0.82001	0.796108
recall validación	0.723652	0.71855	0.724547	0.725569	0.714851
exactitud entrenamiento	0.810489	0.799901	0.834704	0.82001	0.796108
exactitud validación	0.723652	0.71855	0.724547	0.725569	0.714851

En la Tabla 17 se muestran diferentes métricas y el comportamiento tanto en el conjunto de entrenamiento como en el conjuntos de validación de las cinco mejores exactitudes balanceadas sobre el conjunto de validación, como se puede observar la exactitud balanceada en el conjunto de entrenamiento en todos los casos es almenos 12 puntos porcentuales mayor al conjunto de

validación, esto puede indicar que existen ciertos problemas de generalización dentro del modelo ese modelo particular, sin embargo, al ser número consistente en todos los experimentos y al estar siempre en los mismos rangos no se evidencia que el modelo esté sobre ajustando sobre los datos de entrenamiento, adicional a esto, como ya se mostró en apartados anteriores, existían píxeles que tenían valores de otras clases, es posible que el modelo interprete estos píxeles en las clases correctas, pero al ya tener la clase definida los valores se afectan. Llama la atención que la profundidad de los árboles no es mucha en estos experimentos al menos. Sin embargo, al tomar el resultado de todos los experimentos se puede apreciar que el modelo si tiene problemas para generalizar lo cual queda en evidencia al observar la Ilustración 25, en esta ilustración se puede observar como el valor más alto de exactitud balanceada de validación está varios puntos porcentuales (3.5 puntos exactamente) de la exactitud balanceada más baja de entrenamiento.

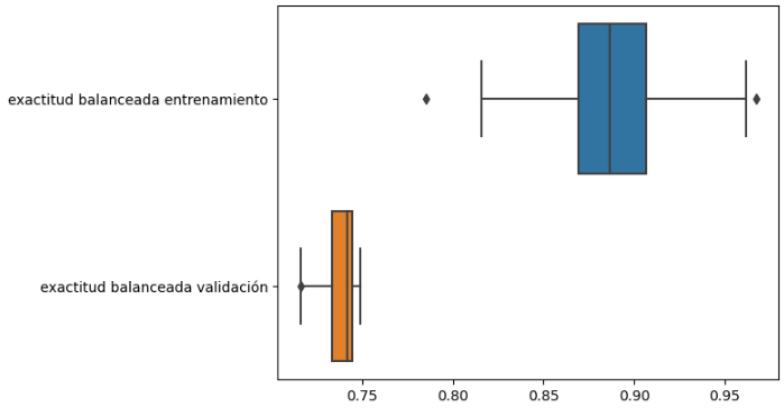


Ilustración 25 Boxplot Exactitud balanceada entrenamiento vs validación.

Por otra parte, al analizar la Ilustración 26 se puede observar como en el proceso de optimización si se tendió a disminuir la cantidad de la profundidad de los árboles, al igual que mantener bajos los valores de gamma, por otra parte, no se observa que exista un impacto directo por el valor de min child weight en el modelo ni tampoco de la cantidad de estimadores, pues los valores que toma tienen a estar bastante distribuidos.

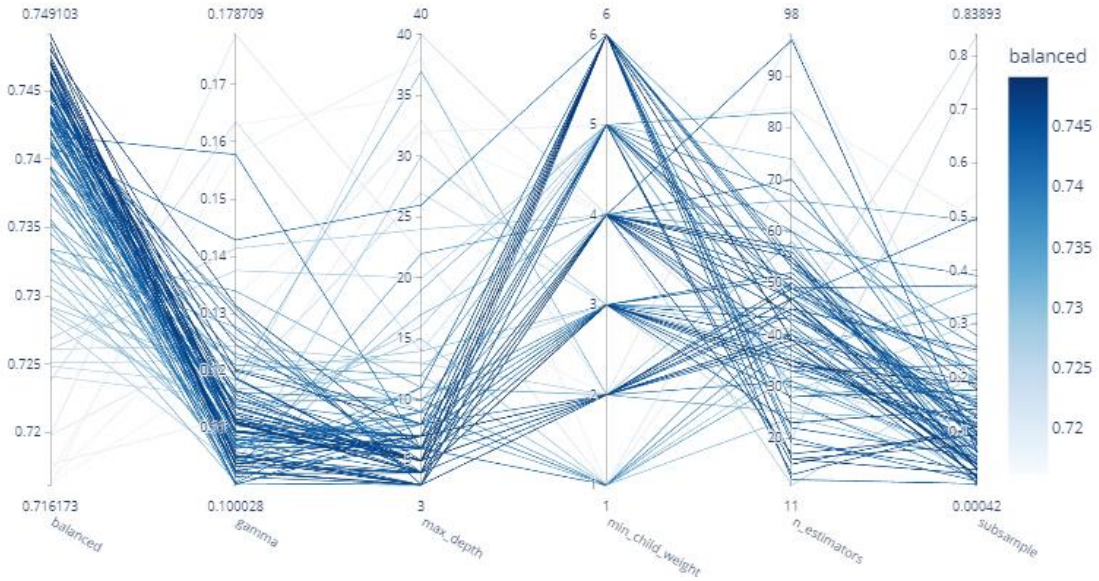


Ilustración 26 Coordenadas paralelas y su impacto en la exactitud balanceada

Con la finalidad de evidenciar un poco más el impacto de cada uno de los hiperparámetros en la exactitud balanceada en el conjunto de validación se realiza la Ilustración 27 en la cual cómo se analizó previamente se evidencia un impacto directo de los valores y como los aumentos o disminuciones en el valor de este hiperparámetro influyen en el resultado.

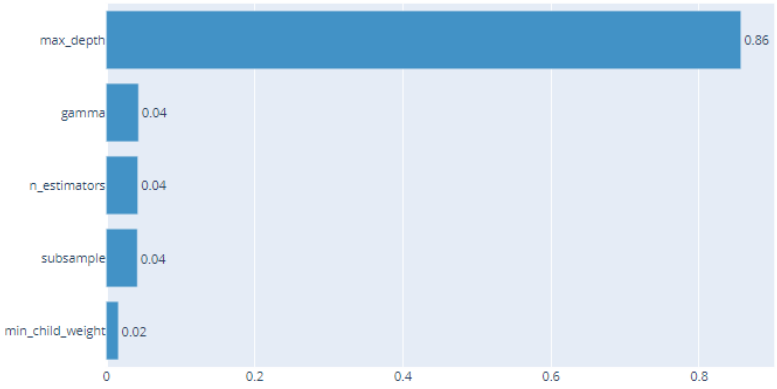


Ilustración 27 Impacto de cada hiperparámetro en el conjunto de validación

Por último y para ahondar un poco más en la importancia de hiperparámetro max depth se realiza la Ilustración 28, esta gráfica aporta algo muy importante al análisis que se realiza en la optimización de hiperparámetros, nótese como a medida que aumenta la profundidad de los árboles en el conjunto de entrenamiento la exactitud balanceada va en aumento, sin embargo exactamente lo opuesto ocurre en el conjunto de validación, donde cada vez existen valores menores de la exactitud balanceada. Esto significa que este proceso si está evitando la

sobreestimación, lo cual es bastante importante y aumenta la confiabilidad del modelo. El historial completo de validación puede ser visto en el Anexo 4

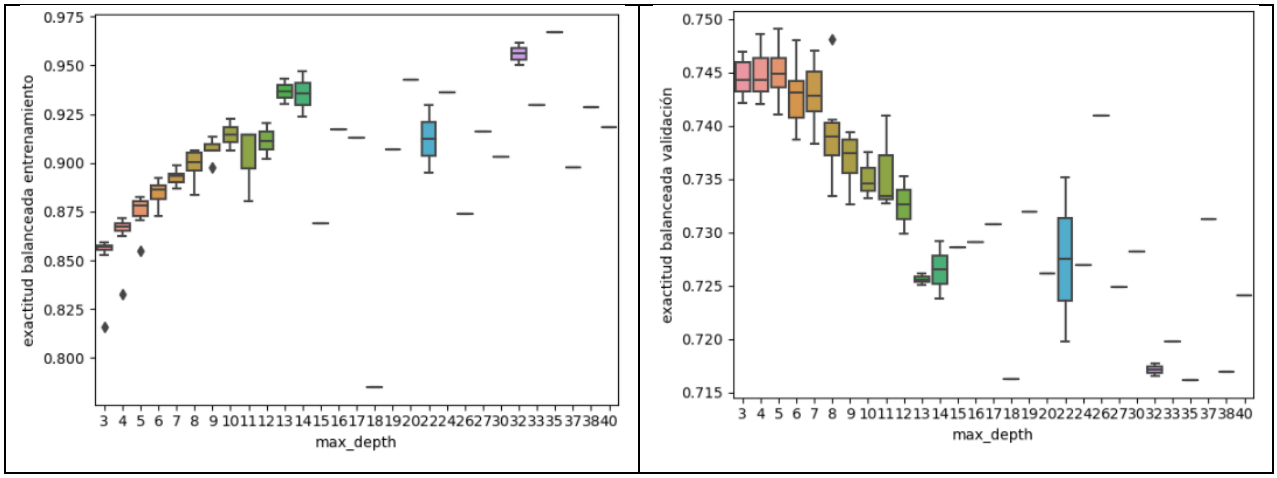


Ilustración 28 Comparación de max depth en entrenamiento y validación

9. RESULTADOS Y ANÁLISIS

Para validar el modelo, es necesario conocer cómo se comporta en el conjunto de entrenamiento y en el conjunto de pruebas. Utilizando los hiperparámetros obtenidos previamente se reentrena el modelo en todo el conjunto de entrenamiento y se analizan los resultados comparando predicciones dadas del conjunto real vs el conjunto de entrenamiento. Los resultados de esta operación pueden ser visualizados en la Tabla 18, en esta tabla se puede observar algunas cosas que vale la pena considerar. En las clases de Agua, Hielo y nubes el modelo hace un trabajo muy bueno logrando distinguir entre valores de píxeles que corresponden a esas clases y valores que no. Por otra parte, tiene problemas para distinguir Bosques y áreas seminaturales y otras coberturas, puede verse como píxeles de todas las clases tienden en mayor o menor medida a ser clasificadas como Bosques y áreas seminaturales, este problema es especialmente alto paras las áreas abiertas con vegetación. Respecto a la clase sombra el modelo realiza un trabajo muy bueno asignando los píxeles reales a su clase, pero, tiende a clasificar valores de píxeles de otras clases como sombra, lo cual muestra el bajo recall del mismo y su impacto directo en el f1-score. Para territorios agrícolas, ocurre algo similar si bien el modelo hace un buen trabajo con los píxeles de su propia clase, al ser predichos como esta clase en un 91%, pero desafortunadamente tiende a confundir los píxeles de otras clases con territorios agrícolas, y solo acierta el 68.2% de las veces. Para las áreas abiertas con vegetación existe el problema contrario, el modelo tiende a asignar clases de esta cobertura a otras coberturas y solo tiene éxito en el 76.3% de las veces, aunque también asigna píxeles de otras coberturas a esta, tiene a no hacerlo el 90% de las veces. Para las áreas de abiertas con poca vegetación se puede ver como tiene a asignar píxeles de esta cobertura a otras coberturas, esto sucede para un 11.5% de los píxeles, por otra parte, no es tan marcada la tendencia de que otras coberturas queden asignadas a esta, excepto por las areas abiertas con vegetación. Para las zonas artificiales se tiene una buena asignación de píxeles dentro de esta clase, y el modelo no tiende a asignar valores de estos píxeles a otras clases de una manera marcada, teniendo una buena estimación el 94% de las veces, pero desafortunadamente si tiende a asignar otras coberturas a esta clase, especialmente delicado las zonas de vegetación, por esta razón se observa una baja precisión del 76.6%.

Finalmente, para las zonas húmedas el modelo realiza un excelente trabajo asignando los píxeles de esta cobertura, pero tiene a confundir algunos píxeles de vegetación y asignarles esta clase en un 14% de las veces.

Una vez analizados estos resultados se observa con las clases de vegetación tienen problemas, tiene a confundir alguna de estas coberturas, especialmente delicado el caso de los territorios agrícolas en los cuales asigna una gran cantidad de píxeles de otras coberturas a esta. El resultado de la exactitud balanceada para el conjunto de entrenamiento es de 0.93, no es un valor particularmente bajo dado que los territorios agrícolas están presentes en solo el 4% de los datos, esta es la clase con el menor recall, es seguido por las sombras, pero estos registros solo están presentes en el 0.9% de los datos. Ahora bien, aunque para esta investigación la exactitud general ni kappa son relevantes, se menciona que la exactitud general es del 0.88 y kappa es de 0.85.

Tabla 18 Matriz de confusión y reporte de clasificación en el conjunto de entrenamiento

Cobertura asignada \ Estimados	Agua	Bosques y Areas Seminaturales	Hielo	Nubes	Sombra	Territorios Agrícolas	Área abierta con vegetación	Área abierta poca vegetación	Áreas Artificiales	Áreas Húmedas	recall	F1. score	Cantidad pixeles	Porcentaje
Agua	432504	191	8	187	615	18	518	110	161	2285	0.991	0.994	436597	0.170
Bosques y Areas Seminaturales	15	220253	0	521	1383	12282	8262	883	436	6	0.903	0.803	244041	0.095
Hielo	0	0	17485	0	0	0	0	0	0	0	1.000	0.980	17485	0.007
Nubes	0	30	19	76159	6	4	92	42	213	1	0.995	0.939	76566	0.030
Sombra	0	20	0	0	23890	1	7	0	44	0	0.997	0.810	23962	0.009
Territorios Agrícolas	0	2716	0	148	6	107137	1350	127	5164	1	0.918	0.775	116649	0.045
Área abierta con vegetación	643	73223	0	5295	6214	32229	586971	32933	23336	3499	0.768	0.831	764343	0.298
Área abierta poca vegetación	247	7691	685	2753	2454	4224	48829	614811	10830	2246	0.885	0.914	694770	0.271
Áreas Artificiales	3	151	0	586	399	3952	2047	1197	131607	123	0.940	0.844	140065	0.055
Áreas Húmedas	5	0	0	0	61	0	14	2	8	49926	0.998	0.924	50016	0.020
Precision	0.998	0.724	0.961	0.889	0.682	0.670	0.906	0.946	0.766	0.860				
Exactitud balanceada = 0.9393														

9.1.1 Curva de entrenamiento

Antes de analizar el resultado en el conjunto de pruebas final (año 2023) es necesario revisar el comportamiento del modelo en una curva de aprendizaje, esto permitirá analizar si existen problemas de sobreajuste o subajuste en el modelo. En este caso se utilizó un muestreo aleatorio estratificado de los datos para el conjunto de entrenamiento completo. En la Ilustración 29 se muestra el resultado de la exactitud balanceada para 5 divisiones de datos, cada punto representa la media de exactitud balanceada sobre una porción de datos del conjunto de entrenamiento, en color azul se muestran los resultados del conjunto de entrenamiento y en color naranja los resultados del conjunto de validación, las áreas sombreadas hacen referencia a la desviación estándar de las divisiones de los conjuntos. En el eje y se puede observar la cantidad de datos entrenados. Teniendo en cuenta estos resultados de la, se puede observar que el modelo posee ciertos problemas de generalización en el conjunto de validación, si bien al aumentar la cantidad de datos ingresados este inconveniente va disminuyendo la brecha sigue siendo amplia, lo cual tiene un impacto directo en la generalización. Por lo que si antes teníamos una exactitud balanceada de 0.93, se puede esperar un valor menor o que duramente pueda pasar el 0.75 (teniendo en cuenta el resultado de la última exactitud balanceada en la gráfica)

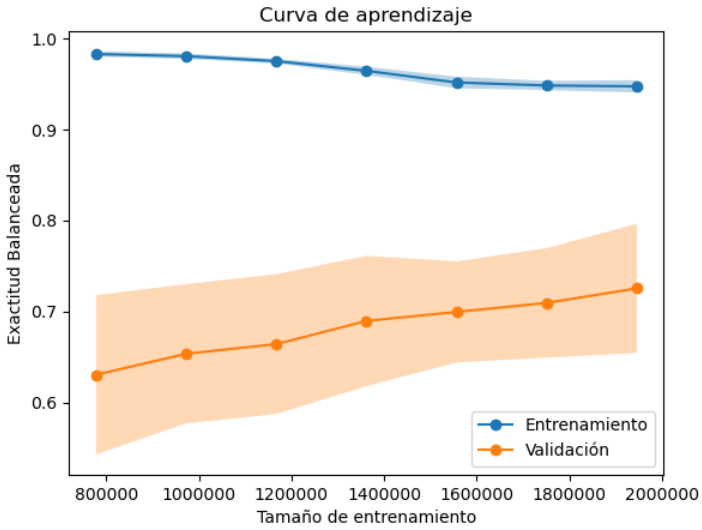


Ilustración 29 Curva de aprendizaje del modelo propuesto

9.1.2 Evaluación en el conjunto de pruebas.

Al igual cuando se validaron los datos con el conjunto de entrenamiento, esta vez se volverá a utilizar la matriz de confusión para analizar los valores de los píxeles. Como se mencionó previamente se esperaba que el valor de la exactitud balanceada estuviera alrededor de 0.75 y los resultados confirman la sospecha con un 0.753 lo cual puede ser visto en la Tabla 19. En este caso, en la matriz de confusión del año 2023 (que es lo mismo que la matriz de confusión del conjunto de pruebas) se puede observar que existen clases particularmente confusas para el modelo. Para iniciar, el modelo tiene problemas para identificar la clase de Bosques y áreas seminaturales, asignando valores de esta cobertura a otras coberturas y asignando valores de otras coberturas a esta, sin embargo lo mismo pasa para las áreas abiertas con vegetación, áreas abiertas con poca vegetación, pero es muchísimo más grave para territorios agrícolas con un 0.49 de f1-score lo cual sucede dada su baja precisión, el modelo parece estar asignando muchas más píxeles de sombra a otras coberturas, esto ya sucedía al revisar la matriz de confusión del conjunto de entrenamiento, pero en el conjunto de pruebas el efecto es mayor. Especialmente bueno es el trabajo realizado por el modelo para identificar coberturas de hielo, tendiendo a confundirse muy poco y confundir muy pocos píxeles de otras clases, lo mismo podría decirse del agua. Con las nubes hace un trabajo decente, aunque tiene a clasificar algunas coberturas como nubes. Para este conjunto el resultado de la exactitud general es de 0.7 y kappa es de 0.63.

Tabla 19 Matriz de confusión y reporte de clasificación en el conjunto de pruebas

		Estimados													
		Agua	**Bosques y Areas Seminaturales**	**Hielo**	**Nubes**	**Sombra**	**Territorios Agrícolas**	**Área abierta con vegetación**	**Área abierta poca vegetación**	**Áreas Artificiales**	**Áreas Húmedas**	**recall**	**F1score**	Cantidad pixeles	Porcentaje
Cobertura asignada	**Agua**	122405	26	1	108	682	2	613	114	140	2146	0.970	0.967	126237	0.140
	Bosques y Areas Seminaturales	62	128447	0	1015	7103	8039	34145	222	325	51	0.716	0.719	179409	0.199
	Hielo	7	0	3063	4	0	0	0	72	0	0	0.974	0.975	3146	0.003
	Nubes	0	69	0	12912	121	75	1099	18	26	9	0.901	0.826	14329	0.016
	Sombra	51	196	0	146	6941	11	249	15	486	7	0.857	0.503	8102	0.009
	Territorios Agrícolas	0	4329	0	6	0	25334	4580	236	0	1	0.735	0.497	34486	0.038
	Área abierta con vegetación	411	38744	0	117	2236	17979	164499	13120	3389	8827	0.660	0.628	249322	0.277
	Área abierta poca vegetación	52	6141	68	415	1428	10391	57085	114634	15398	223	0.557	0.682	205835	0.229
	Áreas Artificiales	86	42	3	2094	366	5540	5218	2091	43149	149	0.735	0.709	58738	0.065
	Áreas Húmedas	3731	0	0	105	605	1	6789	16	66	8494	0.429	0.428	19807	0.022
	Precision	0.965	0.722	0.977	0.763	0.356	0.376	0.600	0.878	0.685	0.427				
		Exactitud Balanceada 0.753													

9.1.3 Interpretación del modelo.

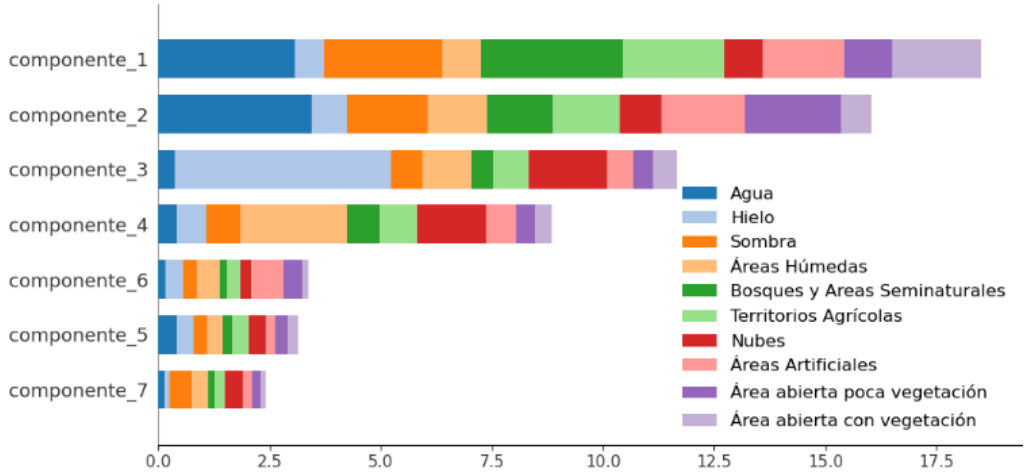


Ilustración 30 Importancia de características con valores SHAP e impacto por clase.

Antes de explicar la Ilustración 30 es muy importante mencionar que el análisis de importancia de características del modelo se realizó con la salida del flujo de procesamiento de datos, en este caso la salida de PCA, la razón de esto es que este análisis es muy costoso computacionalmente y analizar todas las bandas de entrada adicionando los índices haría este proceso muy complejo. Ahora bien, en dicha gráfica se puede apreciar un comportamiento particular para el componente 5 pues es menos importante que el componente 6 para el modelo, identificar la razón de esto dado como funciona xgboost es extremadamente complejo y se sale por completo del alcance de esta investigación, sin embargo este hallazgo no deja de ser curioso pues al aumentar la cantidad de componentes existe una disminución en la información de cada uno, esto se puede apreciar de igual manera en la diferencia de magnitud del componente 4 al componente 6, donde se puede apreciar que la disminución es de casi una tercera parte. Por otra parte, respecto al impacto directo que tienen los valores de cada componente sobre cada clase es importante mencionar que, en términos generales las clases tienen una distribución similar por cada uno de los componentes, sin embargo, el impacto de los valores para las clases Hielo y Áreas húmedas es mayor en el componente 3 y componente 4 respectivamente; esto significa que valores o bajos de cada uno de los componentes están mayormente asociados con estas clases. Las nubes tienen a tener una afectación mayor según los valores del componente 3 y el componente 4 que del componente 1 y el componente 2.

Una vez embebido el modelo en un prototipo se pueden realizarse clasificaciones automáticas. Suponiendo que se tienen tanto el identificador de una imagen landsat9 y las coordenadas x, y superior izquierda de la zona de interés, se puede utilizar le método predict_from_stac de la clase Landsat9LCClassifier, el resultado de dicha operación será un xarray que contendrá toda la información espacial de la imagen, y la clasificación resultante. Para este ejemplo es importante mencionar que la escena mencionada no hace parte del conjunto de entrenamiento y los valores de los píxeles no han sido previamente vistos por el modelo. El tiempo de ejecución es de 39 segundos.

```
import rioxarray as rio
from landsat9_lc import Landsat9LCClassifier
from landsat9_lc.plotting.classification_plot import classification_plot
import matplotlib.pyplot as plt
import matplotlib
clf = Landsat9LCClassifier()
bogota_array = clf.predict_from_stac(
    scene_id='LC09_L2SP_008057_20230219_02_T1',
    x_min=590010,
    y_max=521139,
    pixels=800,
)
classification_plot(bogota_array, 'Bogotá Classification')
```

En la Ilustración 31 se puede ver el resultado de la clasificación, los colores son asignados directamente por el prototipo[17], se puede ver como las zonas de cobertura de agua de la represa de san Rafael son mostradas como agua, al igual que las zonas del parque Simón Bolívar, las zonas de construcciones también se muestran de manera correcta en la imagen, en la imagen hay presentes nubes y sombras las cuales están presentes también en la clasificación(color gris y gris oscuro)

[17] Nuevos colores pueden ser asignados en próximas versiones del prototipo

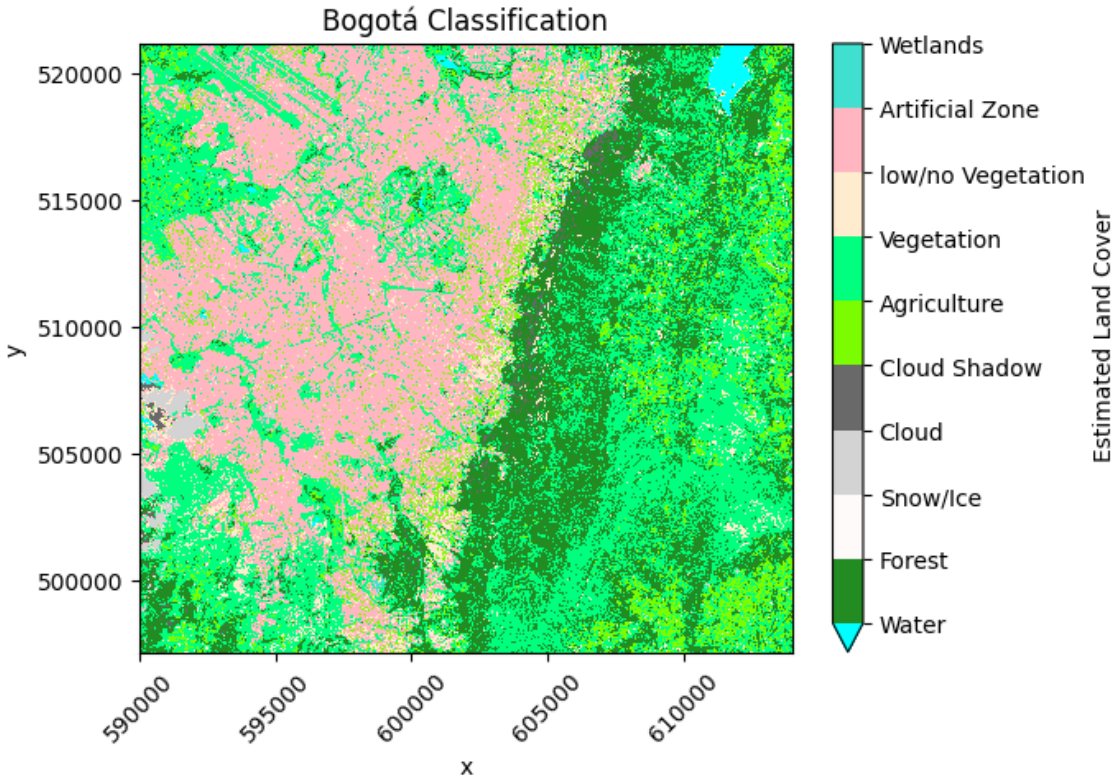
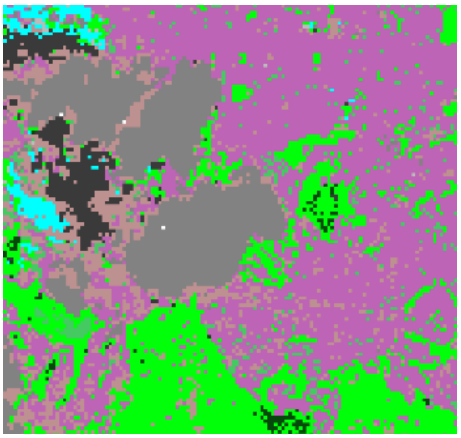


Ilustración 31 Imagen resultado de la clasificación automática.

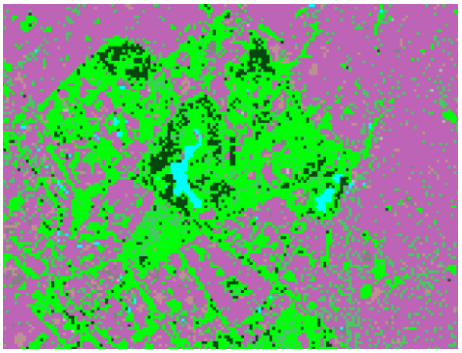
A continuación, se realiza un análisis de los resultados realizando una ampliación sobre diferentes zonas de la imagen usada en el ejemplo, para ello se comparan los resultados y la imagen en RGB 754.

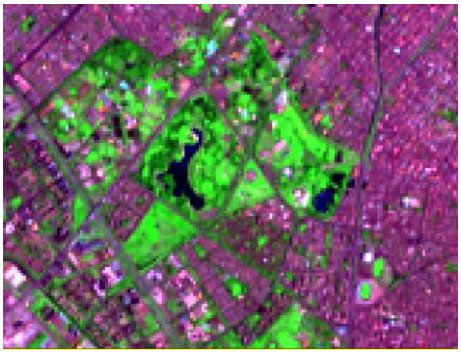
Las nubes son correctamente asignadas a la clase de nubes, y la mayoría de los píxeles asociados a las sombras, se puede observar como las zonas construidas están clasificadas como tal, en la parte inferior se observan algunos píxeles de vegetación de tonalidad más oscura que el prototipo identifica como bosque, sin embargo, se observa que no todos los píxeles de zonas de poca vegetación son asignados correctamente.



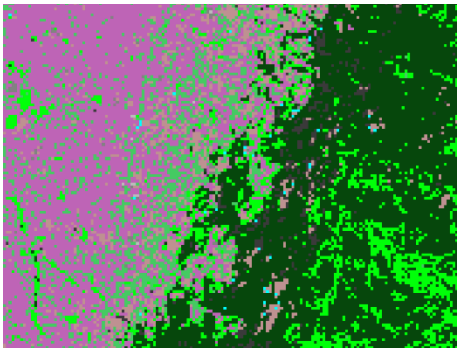


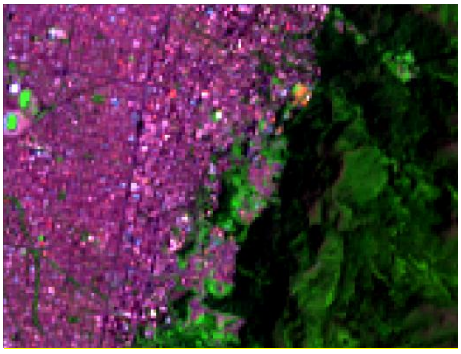
Para la zona del parque Simón Bolívar se observa como el cuerpo de agua del mismo está correcta mente identificado a la vez que la mayor de zonas de área construida, en esta zona no se observan tantas discrepancias entre el valor verdadero y el resultado ofrecido por el prototipo





Respecto a las zonas más ruidosas de la clasificación sobre los cerros orientales se puede ver que existen algunos valores que se confunden con zonas de poca vegetación contra las áreas construidas, las zonas de bosques parecen ser correctamente identificadas, sin embargo, se muestran píxeles de zonas de agricultura lo cual no es cierto.





Adicionalmente, el prototipo permite también realizar clasificaciones sobre cual zona del planeta, para dar un ejemplo se mostrarán los resultados obtenidos en una zona de parís. Debido a que la cantidad de píxeles tenidas en cuenta es menor a la cantidad del ejemplo anterior, la duración es de solo 11.1 segundos

```
import rioxarray as rio
from landsat9_lc import Landsat9LCClassifier
from landsat9_lc.plotting.classification_plot import classification_plot
import matplotlib.pyplot as plt
import matplotlib
clf = Landsat9LCClassifier()
paris_xarray = clf.predict_from_stac(
    scene_id='LC09_L2SP_199026_20230418_02_T1',
    x_min=452927,
    y_max=5411105,
    pixels=200,
)
```

Como se visualiza en la Ilustración 32 es posible clasificar otras zonas, pero, las implicaciones son importantes, en este ejemplo se puede observar que en efecto el agua está bien clasificada, pero las sombras de las nubes tienen a confundirse con píxeles de agua, la zona tiene mezcla de coberturas pero tiene a ser homogénea con representación de zonas artificiales, sin embargo el modelo tiene problemas al lograr identificar correctamente estas zonas y las clasifica como zonas con poca vegetación, esto resultado son esperados debido a los resultados de la matriz de confusión, y los resultados analizados en las salidas de los modelos no supervisados.

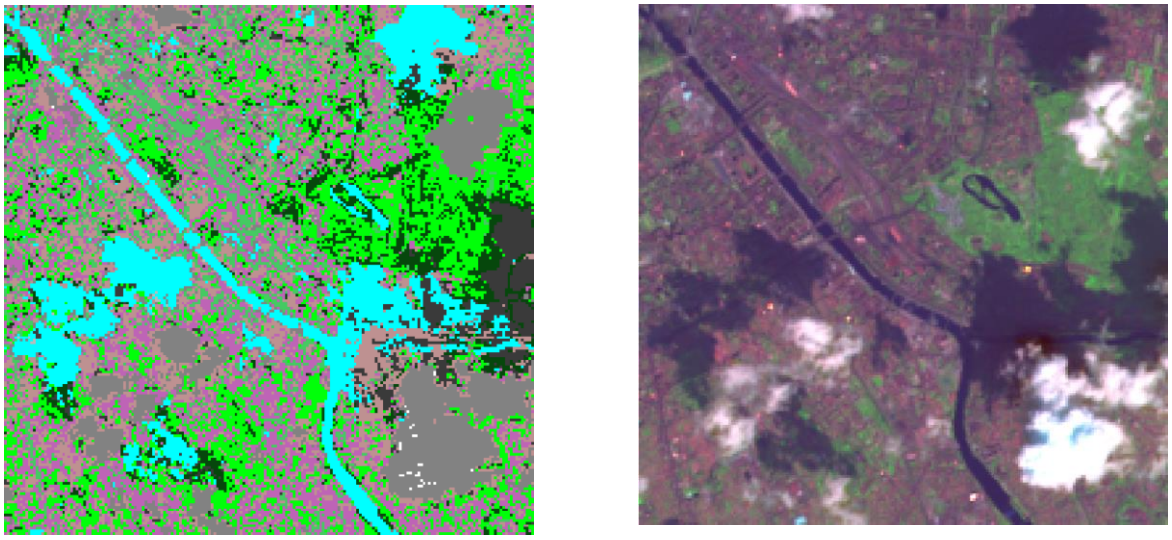


Ilustración 32 Resultado de la clasificación en otra zona del planeta.

10. CONCLUSIONES Y PRODUCTOS

10.1 CONCLUSIONES

10.1.1 Sobre el desarrollo metodológico

- La implementación de la metodología fue exitosa ya que se logró desarrollar un prototipo que permitiera realizar clasificaciones automáticas utilizando primero aprendizaje no supervisado y posteriormente realizando reasignación de los clústeres a las coberturas definidas, de igual manera se logra entrenar y validar un modelo sobre el cual se desarrolla un prototipo para realizar la clasificación automática de coberturas de la tierra, no solo en el territorio colombiano.
- Se crea un modelo consistente, en diferentes regiones y para diferentes fechas el cual, tiene en cuenta diferentes combinaciones espectrales y de esta manera logra identificar patrones, aunque tiene algunos inconvenientes en ciertas clases.
- Al crear el conjunto de entrenamiento de la manera que se hizo permitió utilizar herramientas que se utilizan en la actualidad para procesos de machine learning, lo cual permite que el mantenimiento y las mejoras del prototipo sean más sencillas.

10.1.1.1 Sobre el proceso no supervisado

- La selección de píxeles de muestras en las imágenes de satélite se llevó a cabo de manera que se eliminaron sesgos y se consideraron todas las posibles clases presentes en cada zona. Este enfoque no solo disminuye la cantidad de datos necesarios para el proceso de aprendizaje, sino que también mejora la eficiencia computacional sin afectar la integridad de las coberturas.
- Si bien los resultados obtenidos de la presente investigación se basan en su gran medida del proceso no supervisado, a pesar de las limitaciones que tienen tanto PCA como K-means, se logran obtener resultados que son relativamente estables, sin embargo, después de revisar a detalle se observa como en este tipo de modelos se

tiende a tener poca cantidad de componentes y clústeres lo cual dependiendo del estudio puede no ser deseado.

10.1.1.2 Sobre el proceso supervisado

- El proceso de muestreo en este proceso al ser aleatorio y estratificado permitió tener consistencia a lo largo de todas las imágenes utilizadas, teniendo en cuenta los valores de los niveles digitales al momento de la toma y su reclasificación, al mismo tiempo que disminuye la cantidad de información necesaria para el procesamiento.
- La manera en la cual se optimizaron los hiperparámetros fue computacionalmente muy costosa, ya que requería de varias horas de entrenamiento para un solo conjunto de hiperparámetros, si bien esto permite realizar una validación más detallada su uso tuvo un impacto directo en la velocidad del desarrollo del prototipo.
- Como se evidenció al utilizar una imagen y realizarse la prueba sobre la misma existe un sesgo muy grande en los resultados del modelo, es decir, se pueden obtener resultados muy buenos, pero son solo aplicables a esa imagen particular, por lo que ese mismo modelo, aplicado en otra imagen, podría no estimar de manera correcta.
- El modelo no logra generalizar bien las coberturas vegetales, tiende a estimar de manera imprecisa territorios agrícolas con poca vegetación, áreas abiertas con vegetación y bosques áreas seminaturales. Con las áreas húmedas tiene fuertes problemas pues tiende a confundirlas con áreas abiertas con vegetación. Con nubes, hielo y cuerpos de agua tiene buenos resultados, siendo particularmente alto para los dos últimos tipos de coberturas. Finalmente, los resultados sobre las sombras de las nubes son pobres.

10.1.1.3 Sobre la automatización del proceso de clasificación

- El prototipo desarrollado es simple de utilizar y logra identificar coberturas de manera automática conociendo la imagen de referencia necesaria, por tanto, ya no se hace necesario realizar muestreo de clases, realizar análisis de los datos y la preparación de

estos ya que este se encarga de todo el proceso. Esto permite ahorrar tiempo y esfuerzos por parte de analistas.

- El prototipo desarrollado permite la mejora continua, de esta manera pueden realizarse mejoras constantes al modelo, las cuales pueden ser medidas en: tiempos de ejecución, recursos consumidos, exactitud de la clasificación. También puede realizarse mejoras en el dataset de entrenamiento, mejorando las clases, reduciendo ruido, incrementando el tamaño del conjunto de datos de entrenamiento.

10.2 APORTES ORIGINALES.

Este trabajo contribuye al campo de clasificación de suelo de la tierra, específicamente para el programa Landsat 9 de la siguiente manera:

- Se entrega un dataset de clústeres a los cuales se les pueden asignar diferentes clases, o ser comparados con otros modelos no supervisados y analizar de manera comparativa los resultados.
- Como producto, se entrega un dataset listo para su uso de manera que puedan realizarse diferentes análisis, o desarrollarse metodologías diferentes a la planteada que permitan obtener diferentes resultados.
- Se entrega un prototipo el cual permite realizar la clasificación automática de cobertura de la tierra.

10.3 TRABAJOS DERIVADOS

10.3.1 Artículos.

Patiño Rivera, H., & Ortiz Dávila, Á. (2024). Diseño de un prototipo para clasificación automática de imágenes satelitales mediante algoritmos de machine learning. Revista Ibérica de Sistemas e Tecnologias de Informação.

10.3.2 Ponencias.

Patiño Rivera, H., & Ortiz Dávila, Á. (s. f.). Diseño de un prototipo para clasificación automática de imágenes satelitales mediante algoritmos de machine learning. En *Revista Ibérica*

de Sistemas e Tecnologias de Informação. ICITS 7th International Conference on Information Technology & System, Temuco, Chile. https://icits.me/index.php

11. TRABAJO FUTURO

Existen algunas consideraciones y sugerencias que serían muy útiles para mejorar el comportamiento del prototipo y del modelo. La primera sería teniendo un enfoque en los datos y la segunda un enfoque en el modelo. Teniendo un enfoque basado en los datos se podría:

- A cada zona que se vaya a utilizar se le pueden crear nuevas variables que añadan información de los píxeles vecinos a diferentes escalas y en diferentes agrupaciones, por ejemplos crear kernel de 3 x 3, 5 x 5 o 7 x 7 y almacenando el valor mínimo, máximo o promedio, eso le da información extra a cada uno de los píxeles y además logra un efecto visual de suavizado de la imagen.
- Realizar una nueva validación de las coberturas asignadas en la interpretación visual. En el transcurso de la investigación se realizó dos veces la asignación de coberturas a las clases del estudio, sin embargo, si se pudo evidenciar que en algunas ocasiones fue particularmente dificil debido a los resultados obtenidos del proceso no supervisado. Adicionalmente, para esta tarea se pueden asignar diferentes personas para identificar si existe un consenso en las coberturas propuestas por cada uno y de esta manera tener mayor seguridad en los datos de salida.
- Mejorar los resultados del agrupamiento en términos del modelo, a pesar de que esta sugerencia podría parecer más enfocada al modelo, la verdad es que es más bien como una mezcla, pero se decide dejar en esta sección por la importancia que tiene la salida de este proceso en el proceso supervisado. A pesar de los resultados de UMAP y HDBSCAN en el análisis exploratorio realizado en esta investigación, es posible que se logren obtener mejores resultados con diferentes procesamientos de datos y diferentes hiperparámetros, aunque como se observó tener una métrica de validación alta no lo es todo, llevar a cabo este proceso puede llegar a beneficiar la asignación de clústeres a las coberturas. En el proceso no clasificado también puede trabajarse con autoencoders una rama de aprendizaje profundo que sirve para realizar reducción de la

dimensionalidad, conservando una gran parte de la información de los datos. Y es sin dudas una de las mejoras que se pueden incorporar a este prototipo.

En contrapartida, tambien es posible mejorar el modelo teniendo un enfoque basado únicamente en el modelo para esto se sugiere:

- Inicialmente se puede cambiar el algoritmo de clasificación, se pueden utilizar algoritmos de boosting como catboost o lightgbm, también puede utilizarse Random forest, o la última versión de máquinas de soporte vectorial de la librería cuml, que están optimizadas y que se ejecuta de manera muy rápida. No se sugiere ninguna en particular ya que esto cambia dependiendo del canal de procesamiento de cada modelo propuesto, un canal de procesamiento puede ser muy bueno para un algoritmo, pero no para el otro.
- Se pueden añadir más experimentos al canal propuesto y seguir buscando un conjunto de hiperparámetros que ajuste mejor, sin embargo, esta idea se desaconseja con fuerza, ya que es posible que la ganancia sea mínima.
- Una mejora muy importante puede llegar a ser utilizar un marco de trabajo que permita realizar modelamiento en paralelo y de manera distribuida, se sugiere revisar la librería dask para ese fin. Esto permitirá tener una mayor cantidad de puntos de muestreo de entrenamiento, lo cual puede llegar a beneficiar mucho al modelo propuesto.

Por último, hay que recalcar que el prototipo fue diseñado para evolucionar por eso se sugiere enmarcar el prototipo en un flujo de MLOPS el cual permita tener control total de las versiones de:

- Los conjuntos de entrenamiento.
- Los modelos propuestos.
- Las diferentes métricas asociadas a cada ejecución de los modelos.
- Control de la optimización de hiperparámetros

12. DISCUSIÓN Y RECOMENDACIONES

12.1 COMPARANDO RESULTADO FRENTE A DYNAMIC WORLD

Si bien no es un objetivo de esta investigación comparar los resultados con Dynamic world si es posible realizar algunas comparaciones con esta aplicación, la cual es al momento de realizar esta investigación la más actualizada. Antes de realizar esta comparación es importante recordar que los algoritmos, los recursos de máquina y la cantidad de personas involucradas en el desarrollo de Dynamic world son muy diferentes a las de esta investigación.

Para iniciar, el equipo de Dynamic world, realizó varias matrices de confusión dependiendo de la concordancia entre diferentes expertos al momento de realizar la interpretación visual. Para ese caso la exactitud balanceada fue de 83.07%, el cual es mayor a la desarrollada en esta investigación, sin embargo, como bien se menciona este es en el caso en que existiera concordancia total, lo cual existió en 27’841.623 píxeles. Por otra parte, cuando existía diferencia entre alguno de ellos la exactitud balanceada fue de 67.70%, teniendo el peor comportamiento en lo que ellos etiquetaron como césped, que tiene su similitud con la clase de área abierta con vegetación, en ese caso para ellos hubo dificultad al distinguir entre árboles (Bosques), suelo desnudo (Áreas con poca vegetación), y zonas de cultivos (Zonas Agrícolas), la cantidad de píxeles para este caso fue de 78’916.422. Como se puede apreciar, si bien el modelo desarrollado aún tiene mucho por mejorar, no es un modelo que deba ser desechado sino debe ingresar a un ciclo de mejora continua.

12.2 RECOMENDACIONES

- Etiquetar datos que provengan de un aprendizaje no supervisado es una tarea compleja, dispendiosa y sobre todo muy delicada, una mala asignación de clases puede impactar negativamente tanto el entrenamiento como la validación, y por ende los resultados obtenidos. Por otra parte, esta no parece ser una tarea indicada para una sola persona, sino

que exista un consenso entre diferentes expertos por clúster/clase podría beneficiar aún más este proceso.

- Para avanzar en el desarrollo del modelo, es esencial priorizar la mejora de la clasificación de las coberturas vegetales. Estas tienden a entremezclarse, lo que dificulta la capacidad del modelo para estimar con precisión las coberturas. Al concentrarnos en la mejora de estas coberturas, es posible que también se logre una mejora en la clasificación de otras coberturas de manera simultánea.

13. ANEXOS

Anexo 1 Índices utilizados por grupo de extracción en el aprendizaje no supervisado

Grupo	Índices
Vegetación	AFRI1600, AFRI2100, ARVI, ATSAVI, BCC, BNDVI, BWDRVI, CIG, CVI, DVI, EVI, EVI2, ExG, ExGR, ExR, FCVI, GARI, GBNDVI, GCC, GDVI, GEMI, GLI, GNDVI, GOSAVI, GRNDVI, GRVI, GSAVI, GVMI, IAVI, IKAW, IPVI, MCARI1, MCARI2, MGRVI, MNDVI, MNLI, MRBVI, MSAVI, MSI, MSR, MTVI1, MTVI2, NDDI, NDII, NDMI, NDPI, NDVI, NDYI, NGRDI, NIRv, NLI, NMDI, NRFIg, NRFIr, NormG, NormNIR, NormR, OCVI, OSAVI, RCC, RDVI, RGBVI, RGRI, RI, SARVI, SAVI, SAVI2, SEVI, SIPI, SR, TDVI, TGI, TSAVI, TVI, TriVI, VARI, VIG, WDRVI, WDVI
Agua	ANDWI, AWEInsh, AWEIsh, LSWI, MBWI, MLSWI26, MLSWI27, MNDWI, MuWIR, NDPonI, NDTI, NDVIMNDWI, NDWI, NDWIns, NWI, SWM, WI1, WI2, WRI
Zonas quemadas	BAI, BAIM, CSI, CSIT, MIRBI, NBR, NBR2, NBRSWIR, NBRT1, NBRT2, NBRT3, NDSWIR, NDVIT, NSTv1, NSTv2, SAVIT, VI6T
Hielo / Nieve	NBSIMS, NDSI, NDSII, NDSInw, S3, SWI
Áreas construidas	BLFEI, BRBA, DBI, EBBI, IBI, NBAI, NBUI, NDBI, NDISIb, NDISIg, NDISIndwi, NDISIr, PISI, UI, VIBI, VgNIRBI, VrNIRBI
Suelo desnudo	BI, BaI, DBSI, EMBI, MBI, NBLI, NDBaI, NDSoI, NSDS, NSDSI1, NSDSI2, NSDSI3

Anexo 2 Resultados de experimentos para las tres imágenes de exploración umap y hdbsan

ID	value	cluster selection epsilon	min cluster size	min dist	min samples	n components	n neighbors
1	8.98.E-01	0.269	803	0.219	36	10	44
1	8.94.E-01	0.251	931	0.222	39	10	56
1	8.92.E-01	0.154	954	0.278	40	10	61
1	8.90.E-01	0.181	965	0.261	40	11	66
1	8.88.E-01	0.130	716	0.241	22	10	66
1	8.86.E-01	0.364	745	0.224	38	12	42
1	8.72.E-01	0.348	792	0.270	40	11	57
1	8.67.E-01	0.161	737	0.222	22	10	77
1	8.61.E-01	0.030	169	0.243	22	10	48
1	8.59.E-01	0.273	622	0.204	23	10	93
1	8.57.E-01	0.158	809	0.220	21	10	77
1	8.56.E-01	0.343	774	0.214	36	12	43
1	8.56.E-01	0.173	409	0.167	29	10	84
1	8.52.E-01	0.164	873	0.221	21	10	76
1	8.48.E-01	0.281	777	0.183	34	12	30
1	8.34.E-01	0.268	758	0.195	34	12	34
1	8.31.E-01	0.247	741	0.252	19	10	63
1	8.21.E-01	0.169	533	0.174	27	10	74
1	8.20.E-01	0.359	558	0.235	24	10	69
1	8.19.E-01	0.111	258	0.199	30	11	100
1	7.88.E-01	0.277	862	0.182	38	11	93
1	7.79.E-01	0.286	304	0.189	25	12	37
1	7.72.E-01	0.497	513	0.044	20	10	74
1	7.30.E-01	0.423	178	0.225	39	10	59
1	6.54.E-01	0.242	952	0.269	40	11	56
1	5.49.E-01	0.254	994	0.159	14	12	34
1	5.44.E-01	0.318	730	0.282	38	12	41
1	3.70.E-01	0.189	434	0.219	24	9	85
1	3.63.E-01	0.256	229	0.217	22	9	47
1	3.57.E-01	0.401	835	0.225	39	12	47
1	3.53.E-01	0.198	611	0.230	40	10	40
1	3.51.E-01	0.283	843	0.237	26	9	98
1	3.41.E-01	0.342	805	0.272	40	11	58
1	3.28.E-01	0.003	697	0.178	28	15	42
1	3.06.E-01	0.212	193	0.227	27	11	69

ID	value	cluster selection epsilon	min cluster size	min dist	min samples	n components	n neighbors
1	2.99.E-01	0.363	677	0.214	37	13	53
1	2.97.E-01	0.232	298	0.169	31	10	87
1	2.67.E-01	0.170	429	0.177	29	10	78
1	2.64.E-01	0.304	582	0.183	22	9	94
1	2.44.E-01	0.346	325	0.087	13	8	84
1	2.16.E-01	0.075	362	0.205	23	12	71
1	2.15.E-01	0.148	101	0.193	34	11	64
1	2.10.E-01	0.169	494	0.184	19	9	65
1	2.07.E-01	0.131	406	0.098	10	10	89
1	2.04.E-01	0.316	668	0.242	31	9	36
1	1.95.E-01	0.267	156	0.128	11	8	95
1	1.78.E-01	0.262	861	0.265	39	9	60
1	1.74.E-01	0.226	896	0.280	38	10	42
1	1.68.E-01	0.202	943	0.245	24	11	72
1	1.67.E-01	0.305	635	0.237	36	11	50
1	1.64.E-01	0.203	346	0.045	40	8	11
1	1.58.E-01	0.244	780	0.159	18	13	91
1	1.52.E-01	0.230	916	0.224	39	10	56
1	1.51.E-01	0.331	784	0.211	37	11	15
1	1.47.E-01	0.349	696	0.252	21	11	38
1	1.45.E-01	0.177	650	0.206	12	10	67
1	1.43.E-01	0.077	474	0.216	38	9	45
1	1.30.E-01	0.299	721	0.258	18	9	45
1	1.26.E-01	0.262	837	0.261	16	12	38
1	1.07.E-01	0.415	152	0.167	33	8	31
1	9.34.E-02	0.163	641	0.213	26	10	58
1	8.97.E-02	0.184	965	0.262	40	11	68
1	8.74.E-02	0.146	690	0.238	22	10	57
1	8.12.E-02	0.235	634	0.205	23	11	91
1	7.58.E-02	0.342	659	0.249	35	10	54
1	7.46.E-02	0.321	919	0.264	32	14	26
1	6.89.E-02	0.278	824	0.196	33	12	51
1	6.87.E-02	0.010	538	0.298	23	15	18
1	5.84.E-02	0.115	365	0.230	38	10	84
1	5.39.E-02	0.215	805	0.217	23	10	43
1	5.33.E-02	0.211	923	0.276	39	11	62
1	3.74.E-02	0.331	470	0.190	29	8	96

ID	value	cluster selection epsilon	min cluster size	min dist	min samples	n components	n neighbors
1	3.55.E-02	0.268	745	0.199	34	12	33
1	3.53.E-02	0.377	884	0.258	21	14	49
1	3.45.E-02	0.388	609	0.209	37	10	46
1	3.29.E-02	0.351	719	0.228	15	11	52
1	2.94.E-02	0.144	975	0.254	38	10	65
1	2.87.E-02	0.371	108	0.248	36	11	39
1	2.53.E-02	0.219	592	0.142	20	10	75
1	2.52.E-02	0.154	941	0.230	20	9	68
1	2.46.E-02	0.248	878	0.172	39	13	31
1	2.05.E-02	0.233	518	0.234	36	10	49
1	1.69.E-02	0.031	379	0.268	38	11	28
1	1.69.E-02	0.195	233	0.243	32	9	81
1	1.19.E-02	0.394	850	0.207	33	13	81
1	1.14.E-02	0.092	962	0.296	40	11	49
1	1.12.E-02	0.327	722	0.292	16	11	43
1	1.07.E-02	0.446	769	0.223	35	9	61
1	1.06.E-02	0.499	569	0.001	20	10	73
1	1.05.E-02	0.293	819	0.202	33	12	89
1	8.28.E-03	0.292	793	0.237	40	12	20
1	7.59.E-03	0.308	924	0.243	35	11	10
1	7.06.E-03	0.427	228	0.278	17	8	56
1	6.03.E-03	0.352	991	0.246	36	10	53
1	5.47.E-03	0.265	756	0.204	37	12	23
1	4.81.E-03	0.067	691	0.149	27	13	27
1	3.92.E-03	0.127	892	0.254	25	13	64
1	2.78.E-03	0.180	842	0.259	25	10	81
1	2.75.E-04	0.336	1000	0.227	30	12	55
1	2.39.E-04	0.187	906	0.288	37	10	51
4	9.08.E-01	0.377	170	0.072	12	12	61
4	8.79.E-01	0.359	166	0.043	39	9	80
4	8.69.E-01	0.241	190	0.065	27	14	49
4	8.63.E-01	0.396	356	0.092	12	11	49
4	8.61.E-01	0.332	162	0.088	38	12	32
4	8.54.E-01	0.375	317	0.036	39	9	68
4	8.51.E-01	0.257	126	0.193	39	10	59
4	8.37.E-01	0.386	153	0.117	12	12	68
4	8.32.E-01	0.310	262	0.019	28	9	66

ID	value	cluster selection epsilon	min cluster size	min dist	min samples	n components	n neighbors
4	8.29.E-01	0.378	297	0.044	38	9	70
4	8.25.E-01	0.338	137	0.226	38	11	41
4	8.18.E-01	0.371	150	0.037	38	9	64
4	8.13.E-01	0.402	290	0.068	40	9	69
4	8.05.E-01	0.327	200	0.074	36	10	95
4	8.03.E-01	0.423	311	0.135	13	11	48
4	8.03.E-01	0.061	871	0.115	20	15	88
4	7.93.E-01	0.489	534	0.109	10	10	99
4	7.87.E-01	0.244	317	0.126	10	9	59
4	7.86.E-01	0.369	621	0.117	13	9	51
4	7.85.E-01	0.377	304	0.039	39	9	70
4	7.79.E-01	0.304	898	0.073	33	12	31
4	7.74.E-01	0.309	226	0.005	27	9	64
4	7.74.E-01	0.435	171	0.051	11	13	59
4	7.73.E-01	0.341	101	0.095	15	13	68
4	7.73.E-01	0.394	358	0.099	12	11	63
4	7.72.E-01	0.360	378	0.033	15	11	81
4	7.68.E-01	0.173	346	0.182	40	8	27
4	7.52.E-01	0.378	161	0.082	37	13	62
4	7.40.E-01	0.378	315	0.045	38	9	70
4	7.27.E-01	0.197	165	0.082	27	8	44
4	7.19.E-01	0.383	135	0.103	37	13	80
4	7.08.E-01	0.358	121	0.064	26	12	52
4	7.02.E-01	0.255	328	0.001	35	8	79
4	6.97.E-01	0.388	209	0.077	40	12	77
4	6.94.E-01	0.308	100	0.106	31	9	73
4	6.86.E-01	0.289	199	0.212	37	11	43
4	6.57.E-01	0.332	745	0.103	25	11	73
4	6.46.E-01	0.348	179	0.017	18	14	51
4	6.44.E-01	0.280	270	0.024	24	8	37
4	6.40.E-01	0.404	221	0.094	35	11	57
4	6.33.E-01	0.423	177	0.012	39	10	74
4	6.26.E-01	0.391	765	0.065	29	9	54
4	6.14.E-01	0.234	421	0.070	37	9	55
4	6.07.E-01	0.245	180	0.089	34	12	17
4	6.04.E-01	0.227	123	0.069	14	13	61
4	5.96.E-01	0.294	287	0.163	25	10	51

ID	value	cluster selection epsilon	min cluster size	min dist	min samples	n components	n neighbors
4	5.85.E-01	0.328	254	0.050	24	9	49
4	5.78.E-01	0.403	981	0.141	14	11	48
4	5.77.E-01	0.350	185	0.079	17	10	48
4	5.66.E-01	0.016	122	0.004	30	13	24
4	5.62.E-01	0.282	288	0.026	32	10	66
4	5.55.E-01	0.369	402	0.011	39	9	67
4	5.51.E-01	0.275	292	0.053	28	10	43
4	5.36.E-01	0.499	244	0.172	19	10	46
4	5.33.E-01	0.259	101	0.086	21	11	53
4	5.15.E-01	0.332	415	0.085	10	11	55
4	4.98.E-01	0.378	154	0.021	37	9	73
4	4.87.E-01	0.265	376	0.064	16	14	46
4	4.84.E-01	0.342	138	0.219	38	11	41
4	4.41.E-01	0.272	230	0.100	39	11	68
4	4.38.E-01	0.320	193	0.092	11	15	57
4	4.19.E-01	0.282	477	0.003	28	10	58
4	4.18.E-01	0.172	211	0.200	34	12	37
4	4.13.E-01	0.161	198	0.296	34	9	56
4	4.00.E-01	0.274	389	0.206	21	10	62
4	3.96.E-01	0.410	119	0.045	13	9	48
4	3.95.E-01	0.345	245	0.055	36	8	66
4	3.91.E-01	0.331	577	0.160	25	12	35
4	3.91.E-01	0.410	904	0.054	29	13	63
4	3.88.E-01	0.268	383	0.147	16	11	40
4	3.87.E-01	0.295	261	0.060	26	10	65
4	3.66.E-01	0.260	516	0.074	30	12	17
4	3.64.E-01	0.352	178	0.033	23	9	83
4	3.62.E-01	0.329	355	0.046	35	11	72
4	3.61.E-01	0.437	192	0.127	32	14	67
4	3.54.E-01	0.388	272	0.060	26	14	60
4	3.52.E-01	0.313	565	0.259	34	10	77
4	3.35.E-01	0.458	333	0.059	23	15	31
4	3.34.E-01	0.364	668	0.033	18	12	70
4	3.12.E-01	0.415	246	0.090	20	12	53
4	3.10.E-01	0.400	225	0.039	36	9	45
4	3.01.E-01	0.231	328	0.049	13	15	100
4	2.95.E-01	0.363	449	0.080	39	10	88

ID	value	cluster selection epsilon	min cluster size	min dist	min samples	n components	n neighbors
4	2.88.E-01	0.319	147	0.113	16	13	76
4	2.73.E-01	0.355	229	0.031	33	10	84
4	2.70.E-01	0.312	271	0.029	28	10	61
4	2.49.E-01	0.349	307	0.109	27	12	31
4	2.49.E-01	0.119	815	0.173	40	8	13
4	1.84.E-01	0.356	165	0.055	40	8	79
4	1.82.E-01	0.120	250	0.197	30	13	39
4	1.79.E-01	0.210	110	0.266	40	8	73
4	1.61.E-01	0.369	150	0.070	12	12	60
4	1.48.E-01	0.244	269	0.083	32	14	85
4	1.04.E-01	0.298	246	0.063	11	15	62
4	1.01.E-01	0.422	133	0.017	22	8	75
4	7.90.E-02	0.319	208	0.040	14	12	71
4	7.63.E-02	0.339	350	0.028	38	12	64
4	7.08.E-02	0.387	216	0.050	12	12	91
4	4.65.E-02	0.141	159	0.045	23	11	10
4	4.35.E-02	0.215	457	0.139	37	14	26
32	9.04.E-01	0.198	910	0.295	12	14	89
32	8.54.E-01	0.262	376	0.272	33	14	88
32	8.51.E-01	0.052	422	0.236	13	11	100
32	7.43.E-01	0.227	544	0.261	32	15	93
32	6.89.E-01	0.240	312	0.274	37	15	93
32	5.07.E-01	0.195	324	0.281	29	13	78
32	4.69.E-01	0.076	977	0.275	32	13	28
32	4.07.E-01	0.258	340	0.267	30	15	97
32	3.77.E-01	0.144	238	0.262	32	14	19
32	3.64.E-01	0.198	943	0.268	11	14	100
32	3.25.E-01	0.293	325	0.289	34	15	94
32	2.96.E-01	0.124	100	0.209	12	15	16
32	2.54.E-01	0.277	383	0.267	32	15	87
32	2.33.E-01	0.221	194	0.279	21	14	29
32	2.12.E-01	0.201	923	0.277	10	13	73
32	1.77.E-01	0.299	317	0.292	36	15	95
32	1.61.E-01	0.233	350	0.259	33	15	93
32	1.43.E-01	0.277	389	0.251	33	15	82
32	1.36.E-01	0.201	929	0.283	10	13	76
32	1.35.E-01	0.156	585	0.237	22	13	14

ID	value	cluster selection epsilon	min cluster size	min dist	min samples	n components	n neighbors
32	1.24.E-01	0.234	911	0.273	11	12	93
32	7.44.E-02	0.276	896	0.300	13	14	84
32	7.40.E-02	0.265	421	0.266	31	15	88
32	6.74.E-02	0.313	360	0.238	34	14	98
32	6.66.E-02	0.160	102	0.229	38	14	13
32	6.11.E-02	0.101	439	0.242	18	13	30
32	6.05.E-02	0.183	644	0.232	37	15	95
32	5.22.E-02	0.128	518	0.225	14	11	94
32	4.10.E-02	0.339	654	0.136	15	9	91
32	3.93.E-02	0.029	374	0.243	28	12	70
32	2.78.E-02	0.056	334	0.252	34	15	98
32	2.57.E-02	0.246	990	0.280	10	11	73
32	2.35.E-02	0.078	395	0.279	30	15	88
32	2.23.E-02	0.192	240	0.213	17	13	28
32	2.12.E-02	0.106	542	0.234	30	12	54
32	1.75.E-02	0.310	459	0.255	28	14	86
32	1.44.E-02	0.189	877	0.298	13	14	99
32	1.44.E-02	0.099	460	0.244	17	13	31
32	1.44.E-02	0.171	760	0.287	11	12	54
32	1.43.E-02	0.457	183	0.286	35	10	49
32	1.32.E-02	0.230	622	0.270	32	11	94
32	1.22.E-02	0.073	986	0.276	32	14	26
32	1.09.E-02	0.034	177	0.172	29	14	66
32	1.06.E-02	0.492	142	0.094	34	10	80
32	8.93.E-03	0.101	851	0.204	31	11	100
32	8.73.E-03	0.043	216	0.279	12	10	85
32	7.18.E-03	0.177	292	0.259	40	14	18
32	7.05.E-03	0.131	333	0.290	15	9	14
32	6.45.E-03	0.263	564	0.282	34	15	100
32	5.67.E-03	0.134	649	0.300	36	9	44
32	5.54.E-03	0.390	223	0.075	37	11	25
32	4.90.E-03	0.060	116	0.257	10	15	24
32	4.63.E-03	0.193	217	0.239	15	14	76
32	4.46.E-03	0.090	954	0.258	14	13	69
32	4.43.E-03	0.285	293	0.260	14	13	81
32	3.90.E-03	0.217	496	0.294	20	15	26
32	2.99.E-03	0.046	279	0.247	23	15	22

ID	value	cluster selection epsilon	min cluster size	min dist	min samples	n components	n neighbors
32	2.60.E-03	0.116	144	0.253	11	14	61
32	2.44.E-03	0.214	930	0.272	21	13	96
32	2.25.E-03	0.245	478	0.294	37	14	91
32	1.82.E-03	0.220	262	0.276	31	14	96
32	1.55.E-03	0.124	125	0.218	10	8	39
32	1.01.E-03	0.068	450	0.252	11	15	11
32	9.02.E-04	0.054	811	0.183	22	11	20
32	4.74.E-04	0.092	164	0.267	12	15	16
32	2.31.E-04	0.009	596	0.221	16	12	38
32	2.28.E-04	0.143	711	0.233	27	14	17
32	1.84.E-04	0.084	342	0.281	26	12	44
32	1.81.E-04	0.141	802	0.189	18	10	15
32	1.71.E-04	0.228	310	0.273	39	15	97
32	1.53.E-04	0.251	378	0.019	23	14	57
32	1.07.E-04	0.212	303	0.262	35	15	97
32	9.85.E-05	0.118	239	0.205	13	15	40
32	8.36.E-05	0.267	514	0.290	38	15	92
32	7.15.E-05	0.250	360	0.270	33	14	43
32	6.49.E-05	0.210	693	0.242	31	15	99
32	6.34.E-05	0.256	405	0.290	35	11	83
32	6.03.E-05	0.149	594	0.230	25	15	35
32	5.64.E-05	0.024	951	0.256	16	14	91
32	4.86.E-05	0.221	922	0.295	12	8	90
32	3.75.E-05	0.168	722	0.269	24	13	34
32	3.74.E-05	0.136	969	0.271	15	15	96
32	3.68.E-05	0.256	269	0.284	29	15	93
32	3.40.E-05	0.243	432	0.245	36	14	89
32	3.11.E-05	0.003	414	0.194	19	14	65
32	3.08.E-05	0.160	197	0.276	35	14	19
32	2.86.E-05	0.144	251	0.262	32	14	85
32	2.68.E-05	0.178	742	0.215	26	11	49
32	2.57.E-05	0.182	1000	0.287	12	15	89
32	2.43.E-05	0.206	805	0.224	16	14	97
32	2.23.E-05	0.485	186	0.181	28	8	65
32	9.66.E-06	0.239	788	0.267	11	15	94
32	6.88.E-06	0.076	497	0.168	14	12	10
32	6.55.E-06	0.292	541	0.300	30	15	91

ID	value	cluster selection epsilon	min cluster size	min dist	min samples	n components	n neighbors
32	4.90.E-06	0.150	593	0.263	21	12	10
32	4.72.E-06	0.076	856	0.296	33	13	32
32	4.51.E-06	0.124	673	0.249	19	14	22
32	2.57.E-06	0.066	843	0.263	14	10	86
32	2.61.E-07	0.216	887	0.247	13	13	100
32	3.21.E-08	0.165	953	0.277	10	13	80

Anexo 3 Índice de silueta por hiperparámetros para cada zona de estudio.

ID	n_clusters	n_components	secs	value
1	8	4	13.849	0.384
1	10	6	14.075	0.364
1	11	6	14.162	0.359
1	8	5	14.26	0.355
1	8	9	14.01	0.346
1	9	9	13.872	0.343
1	9	10	13.875	0.342
1	12	4	14.558	0.342
1	13	4	13.976	0.339
1	10	4	13.816	0.337
2	10	4	14.097	0.327
2	13	4	13.849	0.327
2	9	4	13.878	0.326
2	12	4	13.988	0.326
2	11	4	13.872	0.323
2	13	5	14.01	0.308
2	11	5	14.181	0.306
2	10	5	14.017	0.306
2	9	5	13.875	0.305
2	12	5	14.126	0.304
3	8	13	13.849	0.338
3	8	14	13.792	0.337
3	8	15	14.09	0.337
3	8	16	13.972	0.337
3	9	13	13.796	0.33
3	9	12	13.968	0.33
3	9	14	14.558	0.33
3	9	16	13.852	0.329
3	9	15	13.872	0.329
3	9	10	13.851	0.328
4	9	4	13.944	0.582
4	8	4	13.849	0.576
4	9	5	13.933	0.563
4	12	4	13.988	0.56

ID	n_clusters	n_components	secs	value
4	8	6	13.878	0.557
4	8	5	14.01	0.556
4	14	4	14.253	0.553
4	11	4	14.162	0.551
4	9	8	13.919	0.543
4	8	7	13.852	0.542
5	9	5	13.85	0.404
5	8	5	13.849	0.391
5	10	6	13.878	0.378
5	11	7	14.118	0.372
5	8	6	14.085	0.368
5	9	4	14.126	0.367
5	9	11	14.386	0.363
5	10	10	14.136	0.36
5	11	5	13.996	0.353
5	9	6	14.042	0.353
6	8	4	13.826	0.537
6	8	5	14.017	0.517
6	9	5	13.852	0.482
6	12	4	13.849	0.481
6	10	14	13.82	0.477
6	10	11	14.075	0.477
6	8	14	14.558	0.476
6	8	15	13.996	0.475
6	11	6	13.995	0.469
6	12	6	14.162	0.468
7	9	6	13.796	0.419
7	8	8	14.085	0.412
7	9	7	13.988	0.409
7	9	5	14.174	0.403
7	8	9	13.898	0.401
7	8	10	13.984	0.399
7	8	11	14.01	0.398
7	8	12	13.933	0.397
7	10	13	13.875	0.396
7	10	7	13.939	0.382
8	8	4	14.257	0.37

ID	n_clusters	n_components	secs	value
8	9	6	14.126	0.353
8	10	8	14.253	0.346
8	10	13	14.28	0.346
8	10	12	14.042	0.345
8	10	14	13.849	0.344
8	9	15	14.054	0.343
8	8	6	13.851	0.341
8	9	7	14.26	0.337
8	9	8	14.112	0.336
9	9	4	14.044	0.558
9	8	4	13.792	0.558
9	10	4	14.207	0.557
9	8	6	13.902	0.555
9	11	4	13.849	0.542
9	10	5	13.972	0.541
9	8	5	14.112	0.541
9	11	5	14.136	0.541
9	12	4	14.01	0.538
9	10	6	13.996	0.533
9	10	14	14.085	0.382
9	11	12	13.826	0.38
9	11	13	14.136	0.38
9	9	14	13.972	0.38
9	9	15	13.933	0.38
9	9	16	14.233	0.38
9	11	14	13.898	0.38
9	9	12	14.162	0.369
9	9	13	13.968	0.368
9	12	16	14.126	0.364
10	8	4	13.939	0.452
10	8	9	14.085	0.423
10	8	10	14.042	0.422
10	8	14	13.849	0.422
10	8	11	13.82	0.421
10	8	13	13.792	0.42
10	8	12	14.243	0.42
10	8	15	14.097	0.419

ID	n_clusters	n_components	secs	value
10	9	6	13.933	0.416
10	9	8	14.181	0.408
12	8	4	13.902	0.347
12	8	5	13.939	0.336
12	9	6	13.823	0.335
12	8	6	13.878	0.335
12	8	7	13.968	0.33
12	8	8	14.084	0.329
12	11	11	14.253	0.328
12	9	4	14.01	0.322
12	8	13	14.085	0.32
12	10	5	14.042	0.312
13	9	4	14.054	0.352
13	10	4	14.075	0.351
13	8	4	13.823	0.341
13	11	4	13.902	0.32
13	8	5	14.042	0.319
13	9	5	14.28	0.319
13	12	4	14.097	0.318
13	9	6	13.816	0.314
13	9	12	14.162	0.314
13	9	11	14.118	0.314
14	8	6	14.277	0.326
14	9	5	13.796	0.319
14	8	7	14.116	0.319
14	8	8	13.898	0.318
14	8	13	13.944	0.311
14	8	14	14.054	0.311
14	8	15	14.044	0.311
14	8	16	14.01	0.31
14	9	13	13.933	0.308
14	9	15	14.558	0.307
15	8	9	14.044	0.305
15	8	8	13.849	0.305
15	9	10	13.875	0.302
15	8	6	14.26	0.299
15	10	7	13.988	0.294

ID	n_clusters	n_components	secs	value
15	10	9	13.872	0.293
15	9	9	13.968	0.292
15	11	12	14.09	0.292
15	11	6	14.233	0.291
15	9	7	14.243	0.291
16	8	4	14.116	0.338
16	8	5	13.996	0.33
16	9	5	14.28	0.329
16	11	6	13.872	0.327
16	9	6	13.878	0.327
16	10	6	14.143	0.326
16	8	6	14.09	0.326
16	8	7	14.01	0.324
16	8	8	13.82	0.322
16	10	11	14.075	0.321
17	8	6	13.933	0.317
17	8	7	14.085	0.311
17	8	9	14.01	0.309
17	8	10	14.075	0.308
17	9	13	14.558	0.305
17	9	10	14.054	0.304
17	10	5	14.116	0.304
17	9	15	14.243	0.301
17	9	4	13.996	0.299
17	9	5	14.017	0.297
18	8	4	13.902	0.35
18	9	4	13.872	0.349
18	9	6	14.207	0.322
18	9	7	13.933	0.321
18	10	4	13.875	0.321
18	8	5	14.075	0.317
18	8	6	13.82	0.312
18	8	8	13.984	0.31
18	8	14	14.558	0.307
18	13	4	13.849	0.306
19	8	12	14.01	0.28
19	8	11	13.944	0.28

ID	n_clusters	n_components	secs	value
19	8	13	14.085	0.28
19	8	14	13.984	0.28
19	8	15	13.85	0.28
19	8	16	13.878	0.28
19	9	13	13.826	0.269
19	9	8	13.852	0.268
19	9	7	14.118	0.265
19	11	9	13.939	0.263
20	10	4	14.28	0.298
20	8	4	13.823	0.294
20	8	10	14.112	0.291
20	10	5	13.816	0.29
20	9	7	13.872	0.289
20	9	4	13.826	0.288
20	8	6	13.82	0.283
20	11	5	14.007	0.283
20	8	7	13.939	0.282
20	9	13	13.851	0.28
21	9	6	14.136	0.348
21	9	7	14.162	0.346
21	8	6	14.054	0.345
21	11	5	13.851	0.343
21	10	7	13.939	0.34
21	8	8	14.085	0.34
21	9	5	13.972	0.339
21	9	9	13.919	0.338
21	10	5	14.084	0.336
21	9	10	13.826	0.336
22	8	4	14.181	0.356
22	8	5	14.136	0.339
22	8	6	13.816	0.333
22	9	6	13.972	0.33
22	8	9	14.253	0.327
22	9	8	13.851	0.326
22	8	13	14.01	0.324
22	9	4	14.233	0.316
22	10	4	14.277	0.313

ID	n_clusters	n_components	secs	value
22	8	10	13.944	0.312
23	8	4	14.01	0.289
23	9	4	13.972	0.286
23	8	5	13.849	0.283
23	8	6	14.253	0.283
23	10	4	14.084	0.28
23	9	5	14.28	0.28
23	9	7	14.386	0.274
23	8	13	14.136	0.273
23	9	12	13.851	0.273
23	8	11	13.984	0.273
24	9	4	13.976	0.343
24	9	5	13.875	0.335
24	10	5	14.129	0.332
24	12	4	13.826	0.332
24	10	4	13.995	0.331
24	8	4	14.143	0.331
24	8	5	13.792	0.328
24	11	5	13.872	0.324
24	9	6	14.26	0.323
24	13	4	14.09	0.321
25	9	4	13.849	0.331
25	8	4	13.792	0.328
25	10	4	13.816	0.316
25	9	5	13.878	0.314
25	13	4	14.558	0.313
25	10	5	13.984	0.312
25	12	4	13.82	0.312
25	9	6	14.042	0.312
25	10	6	14.129	0.31
25	11	4	14.386	0.31
26	8	4	13.849	0.312
26	14	4	13.85	0.311
26	13	4	13.902	0.309
26	12	4	13.82	0.309
26	15	4	13.933	0.305
26	8	6	14.097	0.305

ID	n_clusters	n_components	secs	value
26	9	4	13.852	0.303
26	10	4	14.116	0.302
26	11	4	14.075	0.301
26	9	9	14.233	0.3
27	8	4	14.181	0.321
27	9	4	13.997	0.317
27	9	6	14.054	0.316
27	10	7	14.129	0.313
27	9	9	14.126	0.312
27	10	11	14.162	0.311
27	8	5	13.996	0.309
27	10	9	13.872	0.309
27	8	7	13.875	0.307
27	11	6	14.075	0.307
28	13	4	13.939	0.32
28	14	4	14.01	0.315
28	10	4	13.944	0.314
28	12	4	14.136	0.31
28	11	4	13.851	0.306
28	16	4	13.988	0.305
28	12	5	14.112	0.302
28	9	7	14.162	0.3
28	8	7	14.075	0.3
28	15	4	13.792	0.298
29	9	4	13.851	0.321
29	11	4	14.136	0.318
29	8	4	14.075	0.317
29	13	4	14.129	0.314
29	12	4	13.878	0.313
29	10	4	13.976	0.307
29	8	5	13.872	0.305
29	14	4	14.253	0.305
29	8	7	13.968	0.303
29	9	9	13.902	0.303
30	10	4	14.277	0.312
30	10	6	13.796	0.298
30	10	5	14.017	0.297

ID	n_clusters	n_components	secs	value
30	11	6	14.174	0.295
30	11	14	14.054	0.294
30	10	13	14.044	0.292
30	9	9	14.116	0.292
30	16	4	14.143	0.29
30	12	15	13.792	0.288
30	9	5	13.823	0.287
31	8	4	14.01	0.327
31	9	4	13.878	0.319
31	8	5	13.944	0.312
31	8	9	13.939	0.304
31	9	8	13.851	0.303
31	13	6	14.075	0.298
31	10	5	14.112	0.297
31	10	6	13.875	0.296
31	11	6	14.09	0.296
31	12	7	14.253	0.295
32	8	4	14.162	0.321
32	11	4	13.872	0.318
32	13	4	14.207	0.316
32	8	5	14.075	0.313
32	14	4	13.919	0.312
32	9	4	13.968	0.312
32	10	4	14.136	0.307
32	12	4	14.118	0.305
32	10	5	14.042	0.302
32	15	4	13.849	0.301
33	8	5	13.826	0.312
33	9	4	13.872	0.31
33	10	4	13.849	0.307
33	11	4	13.875	0.306
33	11	5	13.919	0.305
33	8	8	13.849	0.305
33	10	5	13.816	0.304
33	10	6	14.277	0.303
33	9	6	14.007	0.301
33	9	5	14.017	0.3

ID	n_clusters	n_components	secs	value
34	8	5	13.878	0.347
34	9	6	13.972	0.347
34	9	5	14.084	0.345
34	8	7	13.852	0.337
34	8	4	14.257	0.329
34	9	4	14.01	0.328
34	10	5	14.042	0.324
34	11	6	14.207	0.317
34	10	6	14.129	0.312
34	9	9	13.851	0.309
35	8	4	14.26	0.359
35	9	4	14.017	0.357
35	8	5	13.972	0.351
35	10	6	13.878	0.346
35	9	6	13.796	0.344
35	10	7	14.09	0.338
35	9	7	13.82	0.336
35	8	7	13.849	0.331
35	8	16	14.085	0.327
35	11	6	14.243	0.327
36	8	4	13.849	0.347
36	9	5	13.849	0.341
36	8	5	13.878	0.339
36	10	5	14.044	0.336
36	8	6	14.042	0.334
36	9	4	14.007	0.332
36	8	7	13.939	0.33
36	9	6	13.972	0.324
36	12	4	13.902	0.324
36	8	8	13.984	0.324
38	8	4	14.558	0.338
38	8	11	14.174	0.322
38	10	4	13.944	0.315
38	9	4	14.054	0.311
38	9	5	13.82	0.306
38	10	5	14.129	0.303
38	10	7	14.042	0.291

ID	n_clusters	n_components	secs	value
38	11	6	13.984	0.286
38	8	6	13.976	0.285
38	10	6	14.26	0.283
38	10	4	14.233	0.315
38	8	4	13.82	0.315
38	9	4	13.933	0.314
38	8	7	13.849	0.314
38	11	4	13.878	0.31
38	8	5	14.118	0.306
38	10	5	13.816	0.305
38	9	5	14.26	0.305
38	11	5	13.872	0.304
38	12	4	14.007	0.302
39	8	11	13.792	0.294
39	10	11	13.933	0.294
39	9	15	13.851	0.294
39	9	16	13.872	0.294
39	9	13	13.988	0.294
39	9	14	14.054	0.294
39	10	7	14.044	0.282
39	8	7	14.01	0.278
39	11	9	14.558	0.277
39	8	8	13.878	0.277
40	8	5	13.849	0.331
40	9	5	13.996	0.319
40	8	6	13.875	0.315
40	8	7	13.82	0.311
40	8	8	14.253	0.309
40	11	5	14.085	0.306
40	10	4	14.181	0.306
40	9	6	13.852	0.305
40	10	5	14.26	0.301
40	11	4	13.792	0.301
41	9	6	13.898	0.317
41	9	8	13.878	0.309
41	8	7	14.054	0.305
41	10	7	13.82	0.298

ID	n_clusters	n_components	secs	value
41	11	6	13.984	0.298
41	8	6	14.118	0.295
41	11	10	13.976	0.292
41	10	9	14.084	0.292
41	11	7	14.143	0.291
41	13	12	14.129	0.29
42	9	4	13.968	0.312
42	8	5	13.85	0.31
42	8	4	13.852	0.305
42	11	4	14.162	0.301
42	9	5	13.878	0.297
42	8	7	14.126	0.297
42	10	5	13.82	0.29
42	17	4	14.044	0.29
42	9	6	14.09	0.289
42	11	5	14.112	0.287
44	8	15	13.826	0.329
44	8	4	14.01	0.327
44	9	4	13.872	0.314
44	8	5	13.878	0.312
44	10	4	13.976	0.312
44	10	5	14.042	0.309
44	9	5	13.996	0.306
44	10	6	14.174	0.302
44	9	6	14.558	0.298
44	11	6	14.181	0.295
44	8	4	14.243	0.35
44	9	4	13.851	0.346
44	8	5	13.872	0.338
44	11	4	13.875	0.338
44	11	5	14.09	0.337
44	10	4	13.919	0.333
44	8	6	14.075	0.331
44	12	4	13.826	0.327
44	11	6	14.007	0.325
44	15	4	14.085	0.321
45	8	4	13.849	0.29

ID	n_clusters	n_components	secs	value
45	8	5	14.112	0.282
45	8	6	13.902	0.273
45	10	4	14.042	0.273
45	9	5	13.933	0.269
45	11	4	14.257	0.269
45	9	4	14.01	0.265
45	10	5	14.28	0.26
45	12	4	14.162	0.254
45	9	6	14.207	0.247
46	9	4	13.875	0.52
46	8	4	14.01	0.52
46	10	5	13.872	0.507
46	9	5	13.878	0.507
46	8	5	14.233	0.507
46	9	6	14.007	0.5
46	10	4	14.112	0.499
46	10	7	14.042	0.491
46	8	11	14.116	0.489
46	8	9	14.129	0.483
47	16	5	14.054	0.433
47	19	4	14.181	0.421
47	22	4	13.826	0.419
47	17	6	14.558	0.419
47	14	6	13.792	0.417
47	12	5	13.944	0.415
47	21	4	14.253	0.415
47	13	4	13.849	0.415
47	13	5	13.996	0.414
47	18	5	13.878	0.412
48	9	11	14.042	0.485
48	9	4	14.054	0.474
48	8	13	13.902	0.472
48	10	8	13.851	0.449
48	8	12	13.792	0.448
48	8	10	13.988	0.445
48	9	13	13.933	0.443
48	10	6	14.118	0.44

ID	n_clusters	n_components	secs	value
48	10	12	14.207	0.438
48	8	8	13.997	0.436
49	8	4	14.09	0.555
49	11	5	14.136	0.549
49	13	4	14.084	0.547
49	11	6	13.988	0.543
49	16	4	13.816	0.54
49	13	5	14.01	0.539
49	14	4	13.984	0.535
49	11	8	13.968	0.532
49	15	6	13.878	0.528
49	17	5	13.972	0.524
50	8	5	14.01	0.454
50	8	14	14.174	0.453
50	8	16	13.988	0.453
50	10	6	13.997	0.448
50	9	6	13.872	0.448
50	8	6	13.939	0.448
50	8	4	13.902	0.445
50	21	4	14.162	0.445
50	11	7	13.972	0.441
50	19	5	14.253	0.439
51	17	5	13.816	0.52
51	16	5	14.084	0.519
51	14	5	14.162	0.514
51	11	6	13.933	0.514
51	14	6	14.207	0.513
51	13	5	13.995	0.511
51	10	6	14.126	0.51
51	15	6	13.988	0.509
51	16	6	14.26	0.506
51	19	4	14.01	0.505
52	8	4	14.118	0.441
52	13	4	13.851	0.44
52	10	4	14.129	0.437
52	12	4	13.85	0.437
52	8	7	13.988	0.432

ID	n_clusters	n_components	secs	value
52	11	4	13.939	0.431
52	9	8	14.233	0.429
52	9	9	13.849	0.428
52	14	4	14.181	0.427
52	8	10	13.792	0.426
53	10	8	13.849	0.525
53	10	9	14.558	0.525
53	10	11	14.09	0.523
53	13	7	14.174	0.522
53	16	6	14.097	0.517
53	17	6	14.007	0.515
53	15	7	14.112	0.514
53	12	10	13.902	0.514
53	13	13	14.26	0.513
53	11	9	14.042	0.513
55	8	6	14.162	0.426
55	8	5	13.872	0.416
55	9	5	13.851	0.411
55	17	4	14.257	0.404
55	12	5	13.996	0.404
55	8	7	13.968	0.404
55	10	6	14.207	0.403
55	11	6	13.944	0.403
55	9	8	14.277	0.403
55	9	7	14.085	0.402
55	11	4	14.243	0.571
55	10	4	14.181	0.569
55	9	4	13.852	0.568
55	11	5	13.878	0.566
55	13	5	13.872	0.563
55	11	6	13.816	0.561
55	13	7	14.136	0.561
55	12	4	13.849	0.559
55	11	8	14.143	0.559
55	10	8	14.129	0.559
56	8	9	14.042	0.347
56	8	10	14.075	0.346

ID	n_clusters	n_components	secs	value
56	9	4	13.997	0.34
56	10	7	13.976	0.337
56	9	12	13.796	0.336
56	9	11	13.851	0.332
56	8	8	13.972	0.324
56	8	12	14.01	0.319
56	8	13	14.162	0.319
56	11	4	13.984	0.307
57	8	4	14.01	0.331
57	8	6	13.902	0.326
57	10	4	14.075	0.31
57	9	4	14.126	0.305
57	9	7	13.988	0.304
57	8	9	14.007	0.304
57	8	10	14.129	0.302
57	9	5	13.933	0.302
57	21	4	13.792	0.302
57	10	6	14.112	0.301
58	8	10	13.997	0.381
58	8	13	14.085	0.379
58	8	5	13.976	0.371
58	9	5	14.28	0.361
58	10	6	14.253	0.352
58	9	6	14.126	0.351
58	11	5	14.042	0.351
58	9	7	13.872	0.349
58	8	16	13.849	0.344
58	8	12	13.849	0.341

Anexo 4 Historial de optimización de hiperparámetros

ID	Orden	exactitud balanceada validación	exactitud balanceada entrenamiento	gamma	Max depth	Min child weight	N estimators	subsample	f1 entrenamiento	f1 validación	precision entrenamiento	precision validación	recall entrenamiento	recall validación	exactitud entrenamiento	exactitud validación
98	**1**	0.75	0.87	0.11	5.00	6.00	39.00	0.01	0.81	0.74	0.84	0.77	0.81	0.72	0.81	0.72
95	**2**	0.75	0.87	0.10	4.00	6.00	39.00	0.02	0.80	0.73	0.83	0.77	0.80	0.72	0.80	0.72
99	**3**	0.75	0.88	0.12	8.00	6.00	16.00	0.02	0.84	0.73	0.85	0.77	0.83	0.72	0.83	0.72
27	**4**	0.75	0.87	0.13	6.00	6.00	46.00	0.01	0.82	0.74	0.84	0.77	0.82	0.73	0.82	0.73
79	**5**	0.75	0.87	0.11	4.00	3.00	34.00	0.19	0.80	0.73	0.83	0.77	0.80	0.71	0.80	0.71
81	**6**	0.75	0.87	0.11	4.00	3.00	35.00	0.08	0.80	0.73	0.83	0.77	0.80	0.72	0.80	0.72
91	**7**	0.75	0.87	0.11	5.00	4.00	17.00	0.09	0.81	0.73	0.84	0.77	0.81	0.72	0.81	0.72
49	**8**	0.75	0.89	0.10	7.00	2.00	39.00	0.04	0.84	0.73	0.86	0.77	0.83	0.72	0.83	0.72
93	**9**	0.75	0.87	0.11	5.00	5.00	15.00	0.11	0.81	0.73	0.83	0.77	0.80	0.72	0.80	0.72
64	**10**	0.75	0.86	0.10	3.00	2.00	47.00	0.04	0.79	0.73	0.82	0.77	0.79	0.72	0.79	0.72
78	**11**	0.75	0.87	0.11	4.00	4.00	34.00	0.11	0.80	0.73	0.83	0.77	0.80	0.72	0.80	0.72
65	**12**	0.75	0.85	0.11	3.00	4.00	39.00	0.04	0.79	0.73	0.82	0.77	0.78	0.72	0.78	0.72
67	**13**	0.75	0.86	0.10	3.00	4.00	55.00	0.03	0.79	0.73	0.82	0.77	0.78	0.72	0.78	0.72
85	**14**	0.75	0.88	0.12	5.00	3.00	35.00	0.15	0.82	0.73	0.84	0.77	0.81	0.72	0.81	0.72
37	**15**	0.75	0.89	0.11	6.00	4.00	44.00	0.11	0.83	0.73	0.85	0.77	0.82	0.72	0.82	0.72
89	**16**	0.75	0.86	0.12	4.00	3.00	23.00	0.14	0.80	0.73	0.83	0.77	0.79	0.71	0.79	0.71
68	**17**	0.75	0.86	0.10	3.00	4.00	56.00	0.03	0.79	0.73	0.82	0.77	0.79	0.72	0.79	0.72
63	**18**	0.75	0.86	0.11	3.00	4.00	47.00	0.05	0.79	0.73	0.82	0.77	0.79	0.72	0.79	0.72
44	**19**	0.75	0.88	0.11	5.00	2.00	49.00	0.12	0.81	0.73	0.84	0.77	0.81	0.72	0.81	0.72

ID	Orden	exactitud balanceada validación	exactitud balanceada entrenamiento	gamma	Max depth	Min child weight	N estimators	subsample	f1 entrenamiento	f1 validación	precision entrenamiento	precision validación	recall entrenamiento	recall validación	exactitud entrenamiento	exactitud validación
2				1	00	00	0									
4	**20**			0.1	7.	2.	40.0									
4		0.75	0.89	0	00	00	0	0.03	0.84	0.74	0.86	0.77	0.83	0.72	0.83	0.72
9	**21**			0.1	6.	6.	16.0									
6		0.75	0.88	1	00	00	0	0.07	0.82	0.73	0.84	0.77	0.82	0.72	0.82	0.72
8	**22**			0.1	4.	3.	39.0									
4		0.75	0.87	2	00	00	0	0.19	0.80	0.73	0.83	0.77	0.80	0.71	0.80	0.71
3	**23**			0.1	5.	2.	45.0									
8		0.74	0.88	1	00	00	0	0.25	0.82	0.73	0.84	0.77	0.81	0.72	0.81	0.72
7	**24**			0.1	3.	4.	97.0									
1		0.74	0.86	1	00	00	0	0.06	0.79	0.73	0.82	0.77	0.79	0.72	0.79	0.72
9	**25**			0.1	7.	5.	13.0									
4		0.74	0.89	1	00	00	0	0.11	0.83	0.73	0.85	0.77	0.83	0.72	0.83	0.72
6	**26**			0.1	5.	4.	51.0									
1		0.74	0.88	1	00	00	0	0.08	0.81	0.73	0.84	0.77	0.81	0.72	0.81	0.72
6	**27**			0.1	5.	3.	51.0									
2		0.74	0.88	1	00	00	0	0.09	0.82	0.73	0.84	0.77	0.82	0.72	0.82	0.72
7	**28**			0.1	4.	4.	57.0									
3		0.74	0.87	0	00	00	0	0.08	0.80	0.73	0.83	0.77	0.80	0.72	0.80	0.72
8	**29**			0.1	4.	3.	28.0									
0		0.74	0.87	3	00	00	0	0.18	0.80	0.73	0.83	0.77	0.79	0.71	0.79	0.71
3	**30**			0.1	5.	4.	49.0									
3		0.74	0.88	2	00	00	0	0.37	0.82	0.73	0.84	0.77	0.82	0.72	0.82	0.72
4	**31**			0.1	4.	2.	50.0									
1		0.74	0.87	1	00	00	0	0.11	0.81	0.73	0.83	0.77	0.80	0.72	0.80	0.72
2	**32**			0.1	4.	6.	54.0									
1		0.74	0.87	2	00	00	0	0.18	0.81	0.73	0.83	0.77	0.80	0.72	0.80	0.72
1	**33**			0.1	6.	5.	46.0									
		0.74	0.88	1	00	00	0	0.04	0.83	0.73	0.85	0.77	0.82	0.72	0.82	0.72
6	**34**			0.1	3.	4.	37.0									
6		0.74	0.85	1	00	00	0	0.06	0.78	0.73	0.82	0.77	0.78	0.71	0.78	0.71
2	**35**			0.1	3.	5.	49.0									
8		0.74	0.86	3	00	00	0	0.23	0.79	0.73	0.82	0.77	0.78	0.71	0.78	0.71
5	**36**			0.1	7.	3.	43.0									
9		0.74	0.89	1	00	00	0	0.15	0.84	0.73	0.86	0.77	0.83	0.72	0.83	0.72
1	**37**			0.1	6.	6.	19.0									
0		0.74	0.88	0	00	00	0	0.03	0.82	0.73	0.84	0.77	0.82	0.72	0.82	0.72
7	**38**			0.1	6.	5.	42.0									
7		0.74	0.88	2	00	00	0	0.03	0.83	0.73	0.85	0.77	0.82	0.72	0.82	0.72

ID	Orden	exactitud balanceada validación	exactitud balanceada entrenamiento	gamma	Max depth	Min child weight	N estimators	subsample	f1 entrenamiento	f1 validación	precision entrenamiento	precision validación	recall entrenamiento	recall validación	exactitud entrenamiento	exactitud validación
69	**39**	0.74	0.82	0.10	3.00	4.00	55.00	0.00	0.79	0.74	0.81	0.77	0.78	0.72	0.78	0.72
17	**40**	0.74	0.87	0.12	4.00	6.00	53.00	0.20	0.80	0.73	0.83	0.77	0.80	0.72	0.80	0.72
11	**41**	0.74	0.83	0.10	4.00	6.00	12.00	0.00	0.78	0.73	0.81	0.77	0.78	0.72	0.78	0.72
97	**42**	0.74	0.86	0.10	4.00	6.00	22.00	0.05	0.80	0.73	0.83	0.77	0.79	0.71	0.79	0.71
86	**43**	0.74	0.88	0.11	5.00	3.00	35.00	0.16	0.82	0.73	0.84	0.77	0.81	0.72	0.81	0.72
31	**44**	0.74	0.86	0.13	3.00	5.00	48.00	0.21	0.79	0.73	0.82	0.77	0.79	0.71	0.79	0.71
74	**45**	0.74	0.89	0.11	6.00	4.00	70.00	0.09	0.83	0.73	0.85	0.77	0.82	0.72	0.82	0.72
75	**46**	0.74	0.86	0.10	3.00	5.00	54.00	0.06	0.79	0.73	0.82	0.77	0.79	0.71	0.79	0.71
22	**47**	0.74	0.86	0.11	3.00	6.00	53.00	0.10	0.79	0.73	0.82	0.77	0.79	0.71	0.79	0.71
24	**48**	0.74	0.89	0.11	7.00	6.00	55.00	0.08	0.84	0.73	0.86	0.77	0.83	0.72	0.83	0.72
72	**49**	0.74	0.87	0.11	4.00	4.00	47.00	0.03	0.81	0.73	0.83	0.77	0.80	0.72	0.80	0.72
0	**50**	0.74	0.89	0.16	6.00	1.00	47.00	0.50	0.83	0.73	0.85	0.77	0.82	0.72	0.82	0.72
54	**51**	0.74	0.89	0.10	7.00	3.00	52.00	0.09	0.84	0.73	0.86	0.77	0.84	0.72	0.84	0.72
52	**52**	0.74	0.88	0.11	5.00	2.00	44.00	0.06	0.82	0.73	0.84	0.77	0.81	0.72	0.81	0.72
82	**53**	0.74	0.89	0.11	6.00	3.00	33.00	0.11	0.83	0.73	0.85	0.77	0.82	0.72	0.82	0.72
56	**54**	0.74	0.85	0.11	5.00	2.00	38.00	0.00	0.82	0.74	0.83	0.78	0.81	0.73	0.81	0.73
12	**55**	0.74	0.88	0.10	11.00	6.00	27.00	0.01	0.85	0.74	0.86	0.77	0.84	0.73	0.84	0.73
20	**56**	0.74	0.87	0.14	26.00	6.00	20.00	0.01	0.85	0.74	0.86	0.77	0.85	0.73	0.85	0.73
8	**57**	0.74	0.90	0.1	8.	3.	30.0	0.12	0.85	0.73	0.86	0.77	0.84	0.72	0.84	0.72

ID	Orden	exactitud balanceada validación	exactitud balanceada entrenamiento	gamma	Max depth	Min child weight	N estimators	subsample	f1 entrenamiento	f1 validación	precision entrenamiento	precision validación	recall entrenamiento	recall validación	exactitud entrenamiento	exactitud validación
3				2	00	00	0									
8	**58**			0.1	7.	3.	33.0									
8		0.74	0.90	1	00	00	0	0.22	0.84	0.73	0.86	0.77	0.84	0.72	0.84	0.72
3	**59**			0.1	6.	5.	60.0									
2		0.74	0.89	3	00	00	0	0.26	0.83	0.73	0.85	0.77	0.83	0.72	0.83	0.72
4	**60**			0.1	6.	2.	44.0									
3		0.74	0.89	1	00	00	0	0.13	0.83	0.73	0.85	0.77	0.83	0.72	0.83	0.72
4	**61**			0.1	9.	1.	41.0									
6		0.74	0.90	1	00	00	0	0.05	0.85	0.73	0.87	0.77	0.85	0.72	0.85	0.72
4	**62**			0.1	8.	2.	31.0									
5		0.74	0.90	0	00	00	0	0.05	0.85	0.73	0.86	0.77	0.84	0.72	0.84	0.72
2	**63**			0.1	9.	5.	67.0									
3		0.74	0.91	2	00	00	0	0.20	0.86	0.73	0.87	0.76	0.86	0.72	0.86	0.72
7	**64**			0.1	8.	4.	47.0									
6		0.74	0.90	1	00	00	0	0.15	0.85	0.73	0.86	0.77	0.85	0.72	0.85	0.72
3	**65**			0.1	6.	4.	57.0									
4		0.74	0.89	1	00	00	0	0.39	0.83	0.73	0.85	0.77	0.83	0.72	0.83	0.72
5	**66**			0.1	7.	2.	40.0									
1		0.74	0.90	0	00	00	0	0.13	0.84	0.73	0.86	0.77	0.83	0.72	0.83	0.72
8	**67**				10											
7				0.1	.0	3.	36.0									
		0.74	0.91	2	0	00	0	0.07	0.86	0.73	0.88	0.77	0.86	0.72	0.86	0.72
5	**68**			0.1	9.	3.	45.0									
7		0.74	0.91	0	00	00	0	0.11	0.86	0.73	0.87	0.77	0.85	0.72	0.85	0.72
1	**69**			0.1	8.	6.	63.0									
4		0.74	0.91	1	00	00	0	0.28	0.85	0.73	0.87	0.77	0.85	0.72	0.85	0.72
9	**70**			0.1	9.	3.	28.0									
0		0.74	0.91	1	00	00	0	0.20	0.86	0.73	0.87	0.76	0.85	0.72	0.85	0.72
2	**71**				12											
5				0.1	.0	5.	83.0									
		0.74	0.92	2	0	00	0	0.18	0.88	0.73	0.89	0.76	0.87	0.72	0.87	0.72
7	**72**				22											
0				0.1	.0	4.	60.0									
		0.74	0.90	0	0	00	0	0.03	0.87	0.74	0.88	0.77	0.87	0.73	0.87	0.73
5	**73**				10											
5				0.1	.0	1.	32.0									
		0.73	0.91	1	0	00	0	0.17	0.87	0.73	0.88	0.76	0.86	0.72	0.86	0.72
2	**74**			0.1	11	6.	48.0									
9		0.73	0.91	3	.0	00	0	0.15	0.87	0.73	0.88	0.76	0.87	0.72	0.87	0.72

ID	Orden	exactitud balanceada validación	exactitud balanceada entrenamiento	gamma	Max depth	Min child weight	N estimators	subsample	f1 entrenamiento	f1 validación	precision entrenamiento	precision validación	recall entrenamiento	recall validación	exactitud entrenamiento	exactitud validación
39	**75**	0.73	0.91	0.11	8.00	1.00	380	0.32	0.85	0.72	0.87	0.76	0.85	0.71	0.85	0.71
35	**76**	0.73	0.92	0.12	10.00	4.00	660	0.49	0.87	0.72	0.88	0.76	0.87	0.71	0.87	0.71
47	**77**	0.73	0.91	0.10	11.00	2.00	250	0.12	0.87	0.73	0.88	0.76	0.87	0.72	0.87	0.72
18	**78**	0.73	0.91	0.13	9.00	3.00	350	0.37	0.86	0.73	0.87	0.76	0.86	0.71	0.86	0.71
15	**79**	0.73	0.91	0.10	19.00	5.00	300	0.05	0.88	0.73	0.88	0.76	0.87	0.72	0.87	0.72
53	**80**	0.73	0.90	0.12	37.00	2.00	580	0.03	0.88	0.74	0.89	0.76	0.88	0.73	0.88	0.73
26	**81**	0.73	0.91	0.11	17.00	4.00	350	0.07	0.88	0.73	0.89	0.76	0.88	0.72	0.88	0.72
58	**82**	0.73	0.90	0.12	12.00	1.00	240	0.04	0.87	0.73	0.88	0.76	0.87	0.72	0.87	0.72
36	**83**	0.73	0.92	0.12	14.00	3.00	410	0.16	0.88	0.73	0.89	0.76	0.88	0.72	0.88	0.72
19	**84**	0.73	0.92	0.11	16.00	5.00	740	0.09	0.88	0.73	0.89	0.76	0.88	0.72	0.88	0.72
48	**85**	0.73	0.87	0.11	15.00	1.00	340	0.01	0.86	0.74	0.86	0.77	0.86	0.73	0.86	0.73
50	**86**	0.73	0.90	0.10	30.00	3.00	260	0.04	0.88	0.73	0.89	0.76	0.88	0.73	0.88	0.73
16	**87**	0.73	0.94	0.14	24.00	4.00	110	0.21	0.90	0.73	0.91	0.76	0.90	0.72	0.90	0.72
4	**88**	0.73	0.93	0.11	13	2.	450	0.25	0.89	0.73	0.90	0.76	0.89	0.72	0.89	0.72

ID	Orden	exactitud balanceada validación	exactitud balanceada entrenamiento	gamma	Max depth	Min child weight	N estimators	subsample	f1 entrenamiento	f1 validación	precision entrenamiento	precision validación	recall entrenamiento	recall validación	exactitud entrenamiento	exactitud validación
0				2	.00	00	0									
13	**89**	0.73	0.94	0.14	20.00	5.00	29.00	0.30	0.90	0.73	0.91	0.76	0.90	0.72	0.90	0.72
3	**90**	0.73	0.94	0.12	13.00	3.00	20.00	0.79	0.89	0.73	0.90	0.76	0.89	0.72	0.89	0.72
30	**91**	0.72	0.92	0.12	27.00	3.00	37.00	0.08	0.89	0.73	0.90	0.76	0.89	0.73	0.89	0.73
92	**92**	0.72	0.92	0.12	40.00	4.00	31.00	0.09	0.89	0.73	0.90	0.76	0.89	0.72	0.89	0.72
9	**93**	0.72	0.95	0.16	14.00	5.00	42.00	0.84	0.90	0.72	0.91	0.76	0.90	0.72	0.90	0.72
8	**94**	0.72	0.93	0.13	33.00	2.00	98.00	0.14	0.90	0.73	0.91	0.76	0.90	0.72	0.90	0.72
7	**95**	0.72	0.93	0.18	22.00	1.00	73.00	0.14	0.90	0.73	0.91	0.76	0.90	0.73	0.90	0.73
2	**96**	0.72	0.96	0.11	32.00	5.00	84.00	0.49	0.92	0.73	0.93	0.76	0.92	0.72	0.92	0.72
6	**97**	0.72	0.93	0.16	38.00	1.00	61.00	0.12	0.91	0.73	0.91	0.76	0.91	0.73	0.91	0.73
5	**98**	0.72	0.95	0.13	32.00	4.00	44.00	0.31	0.92	0.73	0.92	0.76	0.92	0.72	0.92	0.72
60	**99**	0.72	0.79	0.12	18.00	2.00	63.00	0.00	0.80	0.74	0.82	0.77	0.80	0.73	0.80	0.73
4	**100**	0.72	0.97	0.16	35.00	4.00	41.00	0.54	0.93	0.72	0.94	0.75	0.93	0.72	0.93	0.72

14. BIBLIOGRAFIA

Arreola-Esquivel, M., Toxqui-Quitl, C., Delgadillo-Herrera, M., Padilla-Vivanco, A., Ortega-Mendoza, G., & Carbone, A. (2021). Non-binary snow index for multi-component surfaces. *Remote Sensing*, *13*(14). https://doi.org/10.3390/rs13142777

Banerjee, A., Merugu, S., Dhillon, I. S., & Ghosh, J. (2005). Clustering with Bregman divergences. *Journal of Machine Learning Research*, *6*.

Bouhennache, R., Bouden, T., Taleb-Ahmed, A., & Cheddad, A. (2019). A new spectral index for the extraction of built-up land features from Landsat 8 satellite imagery. *Geocarto International*, *34*(14). https://doi.org/10.1080/10106049.2018.1497094

Brown, C. F., Brumby, S. P., Guzder-Williams, B., Birch, T., Hyde, S. B., Mazzariello, J., Czerwinski, W., Pasquarella, V. J., Haertel, R., Ilyushchenko, S., Schwehr, K., Weisse, M., Stolle, F., Hanson, C., Guinan, O., Moore, R., & Tait, A. M. (2022). Dynamic World, Near real-time global 10 m land use land cover mapping. *Scientific Data*, *9*(1). https://doi.org/10.1038/s41597-022-01307-4

Cawley, G. C., & Talbot, N. L. C. (2010). On over-fitting in model selection and subsequent selection bias in performance evaluation. *Journal of Machine Learning Research*, *11*.

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, *13-17-August-2016*. https://doi.org/10.1145/2939672.2939785

Chuvieco, E. (1995). *Fundamentos de Teledetección Espacial* (2nd ed.). Ediciones RIALP, S.A.

D., S., Deepa, P., & K., V. (2017). Remote Sensing Satellite Image Processing Techniques for Image Classification: A Comprehensive Survey. *International Journal of Computer Applications*, *161*(11). https://doi.org/10.5120/ijca2017913306

Del Pilar, M., Laabal, M., & Sallnaro, E. C. (1998). Cartografia e inventorie de incendios forestales en la Peninsula Ibérica a partir de imâgenes NOAA AVHRR. *Universidad de Alcalá de Henares, Serie Geográfica 7*.

Deng, Y., Wu, C., Li, M., & Chen, R. (2015). RNDSI: A ratio normalized difference soil index for remote sensing of urban/suburban environments. *International Journal of Applied Earth Observation and Geoinformation*, *39*. https://doi.org/10.1016/j.jag.2015.02.010

Dixit, A., Goswami, A., & Jain, S. (2019). Development and evaluation of a new "Snow Water Index (SWI)" for accurate snow cover delineation. *Remote Sensing*, *11*(23). https://doi.org/10.3390/rs11232774

Dridi, S. (2015). Unsupervised Learning- A Systematic Literature Review. *A Comparative Analysis on Linear Regression and Support Vector Regression*, *2015*.

Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*.

Eynard-Bontemps, G., Abernathey, R., Hamman, J., Ponte, A., & Rat, W. (2019). THE PANGEO BIG DATA ECOSYSTEM AND ITS USE AT CNES. *Proceedings of 2019 Big Data from Space (BiDS'19)*, *February*.

Feng, M., Huang, C., Sexton, J. O., Channan, S., Narasimhan, R., & Townshend, J. R. (2012). An approach for quickly labeling land cover types for multiple epochs at globally selected locations. *International Geoscience and Remote Sensing Symposium (IGARSS)*. https://doi.org/10.1109/IGARSS.2012.6352674

Giovanelli, J., Bilalli, B., & Abelló, A. (2022). Data pre-processing pipeline generation for AutoETL. *Information Systems*, *108*. https://doi.org/10.1016/j.is.2021.101957

Gormley, I. C., Murphy, T. B., & Raftery, A. E. (2023). Model-Based Clustering. In *Annual Review of Statistics and Its Application* (Vol. 10). https://doi.org/10.1146/annurev-statistics-033121-115326

Helber, P., Bischke, B., Dengel, A., & Borth, D. (2019). Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *12*(7). https://doi.org/10.1109/JSTARS.2019.2918242

Holden, Z. A., Smith, A. M. S., Morgan, P., Rollins, M. G., & Gessler, P. E. (2005). Evaluation of novel thermally enhanced spectral indices for mapping fire perimeters and comparisons with fire atlas data. *International Journal of Remote Sensing*, *26*(21). https://doi.org/10.1080/01431160500239008

Holland, S. M. (2008). Principal Component Analysis (PCA). *Department of Geology, University of Georgia*.

Hoyer, S., & Hamman, J. (2017). xarray: N-D labeled Arrays and Datasets in Python. *Journal of Open Research Software*, *5*(1). https://doi.org/10.5334/jors.148

Hunt, E. R., & Rock, B. N. (1989). Detection of changes in leaf water content using Near- and Middle-Infrared reflectances. *Remote Sensing of Environment*, *30*(1). https://doi.org/10.1016/0034-4257(89)90046-1

Karnieli, A., Kaufman, Y. J., Remer, L., & Wald, A. (2001). AFRI - Aerosol free vegetation index. *Remote Sensing of Environment*, *77*(1). https://doi.org/10.1016/S0034-4257(01)00190-0

Kawamura, M., Jayamanna, S., & Tsujiko, Y. (1996). Relation between social and environmental conditions in Colombo Sri Lanka and the urban index estimated by satellite remote sensing data. *International Archives of Photogrammetry and Remote Sensing*, *XXXI(B7)*.

Khaledian, Y., & Miller, B. A. (2020). Selecting appropriate machine learning methods for digital soil mapping. *Applied Mathematical Modelling*, *81*. https://doi.org/10.1016/j.apm.2019.12.016

Kotsiantis, S. B. (2007). Supervised machine learning: A review of classification techniques. In *Informatica (Ljubljana)* (Vol. 31, Issue 3).

Lacaux, J. P., Tourre, Y. M., Vignolles, C., Ndione, J. A., & Lafaye, M. (2007). Classification of ponds from high-spatial resolution remote sensing: Application to Rift Valley Fever epidemics in Senegal. *Remote Sensing of Environment*, *106*(1). https://doi.org/10.1016/j.rse.2006.07.012

Lillesand, T., Kiefer, R., & Chipman, J. (2014). *Remote sensing and image interpretation.* John Wiley & Sons.

Lin, H., Wang, J., Liu, S., Qu, Y., & Wan, H. (2005). Studies on urban areas extraction from Landsat TM images. *International Geoscience and Remote Sensing Symposium (IGARSS)*, *6*. https://doi.org/10.1109/IGARSS.2005.1525743

McFeeters, S. K. (1996). The use of the Normalized Difference Water Index (NDWI) in the delineation of open water features. *International Journal of Remote Sensing*, *17*(7). https://doi.org/10.1080/01431169608948714

McInnes, L., Healy, J., & Astels, S. (2017). hdbscan: Hierarchical density based clustering. *The Journal of Open Source Software*, *2*(11). https://doi.org/10.21105/joss.00205

McInnes, L., Healy, J., Saul, N., & Großberger, L. (2018). UMAP: Uniform Manifold Approximation and Projection. *Journal of Open Source Software*, *3*(29). https://doi.org/10.21105/joss.00861

Montero, D., Aybar, C., Mahecha, M. D., Martinuzzi, F., Söchting, M., & Wieneke, S. (2023). A standardized catalogue of spectral indices to advance the use of remote sensing in Earth system research. *Scientific Data*, *10*(1), 197. https://doi.org/10.1038/s41597-023-02096-0

Mosley, L. S. D. (2013). A balanced approach to the multi-class imbalance problem. In *ProQuest Dissertations and Theses*.

Moulavi, D., Jaskowiak, P. A., Campello, R. J. G. B., Zimek, A., & Sander, J. (2014). Density-based clustering validation. *SIAM International Conference on Data Mining 2014, SDM 2014*, *2*. https://doi.org/10.1137/1.9781611973440.96

Murray, N. J., Keith, D. A., Simpson, D., Wilshire, J. H., & Lucas, R. M. (2018). Remap: An online remote sensing application for land cover classification and monitoring. *Methods in Ecology and Evolution*, *9*(9). https://doi.org/10.1111/2041-210X.13043

Naushad, R., Kaur, T., & Ghaderpour, E. (2021). Deep transfer learning for land use and land cover classification: A comparative study. *Sensors*, *21*(23). https://doi.org/10.3390/s21238083

Nielsen, D. (2016). Tree Boosting With XGBoost Why Does XGBoost Win "Every" Machine Learning Competition? *Tree Boosting With XGBoost - Why Does XGBoost Win "Every" Machine Learning Competition?*, *December*.

Oh, H., Jung, J. H., Jeon, B. C., & Youn, B. D. (2018). Scalable and Unsupervised Feature Engineering Using Vibration-Imaging and Deep Learning for Rotor System Diagnosis. *IEEE Transactions on Industrial Electronics*, *65*(4). https://doi.org/10.1109/TIE.2017.2752151

Rad, A. M., Kreitler, J., & Sadegh, M. (2021). Augmented Normalized Difference Water Index for improved surface water monitoring. *Environmental Modelling and Software*, *140*. https://doi.org/10.1016/j.envsoft.2021.105030

Raschka, S. (2018). Model evaluation , model selection , and algorithm selection in machine learning Performance Estimation : Generalization Performance Vs . Model Selection. *ArXiv*, *January*.

Rasul, A., Balzter, H., Ibrahim, G. R. F., Hameed, H. M., Wheeler, J., Adamu, B., Ibrahim, S., & Najmaddin, P. M. (2018). Applying built-up and bare-soil indices from Landsat 8 to cities in dry climates. *Land*, *7*(3). https://doi.org/10.3390/land7030081

Reddy, G. T., Reddy, M. P. K., Lakshmanna, K., Kaluri, R., Rajput, D. S., Srivastava, G., & Baker, T. (2020). Analysis of Dimensionality Reduction Techniques on Big Data. *IEEE Access*, *8*. https://doi.org/10.1109/ACCESS.2020.2980942

Riggs, G. A., Hall, D. K., & Salomonson, V. V. (1994). Snow index for the Landsat Thematic Mapper and moderate resolution imaging spectroradiometer. *International Geoscience and Remote Sensing Symposium (IGARSS)*, *4*. https://doi.org/10.1109/igarss.1994.399618

Rikimaru, A., Roy, P. S., & Miyatake, S. (2002). Tropical forest cover density mapping. *Tropical Ecology*, *43*(1).

Rokni, K., Ahmad, A., Selamat, A., & Hazini, S. (2014). Water feature extraction and change detection using multitemporal landsat imagery. *Remote Sensing*, *6*(5). https://doi.org/10.3390/rs6054173

Rouse, J. W., Hass, R. H., Schell, J. A., & Deering, D. W. (1973). Monitoring vegetation systems in the Great Plains with ERTS. *Nasa ERTS Symposium*.

Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, *20*(C). https://doi.org/10.1016/0377-0427(87)90125-7

Ruppert, D. (2004). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. *Journal of the American Statistical Association*, *99*(466). https://doi.org/10.1198/jasa.2004.s339

Saito, A., & Yamazaki, T. (1999). Characteristics of Spectral Reflectance for Vegetation Ground Surfaces with Snow-cover; Vegetation Indices and Snow Indices. *Journal of Japan Society of Hydrology and Water Resources*, *12*(1). https://doi.org/10.3178/jjshwr.12.28

Schubert, E., Sander, J., Ester, M., Kriegel, H. P., & Xu, X. (2017). DBSCAN revisited, revisited: Why and how you should (still) use DBSCAN. *ACM Transactions on Database Systems*, *42*(3). https://doi.org/10.1145/3068335

Sripada, R. P., Heiniger, R. W., White, J. G., & Weisz, R. (2005). Aerial color infrared photography for determining late-season nitrogen requirements in corn. *Agronomy Journal*, *97*(5). https://doi.org/10.2134/agronj2004.0314

Suárez Londoño, A. S., Jiménez López, A. F., Castro Franco, M., & Cruz Roa, A. A. (2017). Clasificación y mapeo automático de coberturas del suelo en imágenes satelitales utilizando Redes Neuronales Convolucionales. *Orinoquia*, *21*(1 Sup). https://doi.org/10.22579/20112629.432

Trigg, S., & Flasse, S. (2001). An evaluation of different bi-spectral spaces for discriminating burned shrub-savannah. *International Journal of Remote Sensing*, *22*(13). https://doi.org/10.1080/01431160110053185

Trujillo Jiménez, M., Liberoff, A. L., Pessacg, N., Pacheco, C., & Flaherty, S. (2021). Metodología de clasificación automática de uso y cobertura de suelo. *XIII Congreso de AgroInformática*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, *2017-December*.

Vergara, V. M., Salman, M., Abrol, A., Espinoza, F. A., & Calhoun, V. D. (2020). Determining the number of states in dynamic functional connectivity using cluster validity indexes. *Journal of Neuroscience Methods*, *337*. https://doi.org/10.1016/j.jneumeth.2020.108651

Wan, L., Li, S., Chen, Y., He, Z., & Shi, Y. (2022). Application of Deep Learning in Land Use Classification for Soil Erosion Using Remote Sensing. *Frontiers in Earth Science*, *10*. https://doi.org/10.3389/feart.2022.849531

Waqar, M. M., Mirza, J. F., Mumtaz, R., & Hussain, E. (2012). Development of New Indices for Extraction of Built-Up Area & Bare Soil. *Open Access Scientific Reports*, *1*(1).

Zebari, R., Abdulazeez, A., Zeebaree, D., Zebari, D., & Saeed, J. (2020). A Comprehensive Review of Dimensionality Reduction Techniques for Feature Selection and Feature Extraction. *Journal of Applied Science and Technology Trends*, *1*(2). https://doi.org/10.38094/jastt1224

Zha, Y., Gao, J., & Ni, S. (2003). Use of normalized difference built-up index in automatically mapping urban areas from TM imagery. *International Journal of Remote Sensing*, *24*(3). https://doi.org/10.1080/01431160304987