

UCUENCA

Universidad de Cuenca

Facultad de Agronomía

Carrera de Ingeniería Agronómica

Análisis de cambios de cobertura del suelo mediante imágenes

Sentinel-2 y Landsat-8 en la subcuenca del río Burgay

Trabajo de titulación previo a la
obtención del título de Ingeniero
Agrónomo

Autores:

María Augusta Murillo Patiño

María Camila Obando Loja

Director:

Daniel Augusto Orellana Vintimilla

ORCID:  0000-0001-8945-2035

Cuenca, Ecuador

2024-06-06

Resumen

El análisis de las imágenes captadas por satélites han evidenciado su importancia en la identificación y seguimiento de las particularidades de la superficie terrestre. Estos estudios permiten detectar las alteraciones a lo largo del tiempo. Nuestra investigación se centró en analizar la evolución de la cobertura del suelo en la subcuenca del río Burgay, utilizando imágenes de los satélites Landsat-8 OLI y Copernicus S2 Sentinel-2 MSI, disponibles en la biblioteca del Servicio Geológico de Estados Unidos (USGS). La metodología incluyó la codificación en la plataforma Google Earth Engine y el uso del algoritmo clasificador supervisado Random Forest. Se produjeron mapas de cobertura del suelo para tres períodos de tiempo (2016, 2018 y 2020) independientemente para cada satélite. La precisión de las clasificaciones fue alta, alcanzando el 95,33% (2016), 96,75% (2018) y 98,62% (2020) para Landsat-8, y el 99,28% (2016), 93,65% (2018) y 98,63% (2020) para Sentinel-2. Los resultados resaltan la importancia de la resolución espacial en la identificación de coberturas, con una diferencia de cambio del 6,37% (2016-2020) para Landsat-8 (resolución: 30 m) y del 8,97% (2016-2020) para Sentinel-2 (resolución: 10 m). Estos resultados no sólo indican cambios y persistencias en las coberturas del suelo, también demuestran las ventajas y limitaciones de utilizar Google Earth Engine y la base de datos de archivos públicos en su plataforma, para rastrear y monitorear estas transiciones en el tiempo.

Palabras clave del autor: cambio cobertura, teledetección, Landsat-8, Sentinel-2, Google Earth Engine, Random Forest



El contenido de esta obra corresponde al derecho de expresión de los autores y no compromete el pensamiento institucional de la Universidad de Cuenca ni desata su responsabilidad frente a terceros. Los autores asumen la responsabilidad por la propiedad intelectual y los derechos de autor.

Repositorio Institucional: <https://dspace.ucuenca.edu.ec/>

Abstract

The analysis of images captured by satellites has demonstrated their importance in the identification and monitoring of the peculiarities of the earth's surface. These studies make it possible to detect alterations over time. Our research focused on analyzing the evolution of land cover in the Burgay River sub-basin, using images from Landsat-8 OLI and Copernicus S2 Sentinel-2 MSI satellites, available at the United States Geological Survey (USGS) library. The methodology included coding in the Google Earth Engine platform and use of the Random Forest supervised classifier algorithm. Land cover maps for three time periods (2016, 2018, and 2020) were produced independently for each satellite. The accuracy of the classifications was high, reaching 95.33% (2016), 96.75% (2018), and 98.62% (2020) for Landsat-8, and 99.28% (2016), 93.65% (2018) and 98.63% (2020) for Sentinel-2. The results highlight the importance of spatial resolution in land cover identification, with a difference in change of 6.37% (2016-2020) for Landsat-8 (resolution: 30 m) and 8.97% (2016-2020) for Sentinel-2 (resolution: 10 m). These results not only indicate land cover change and persistence, they also demonstrate the advantages and limitations of using Google Earth Engine and the public archive database in its platform to track and monitor these transitions over time.

Author Keywords: coverage change, remote sensing, Landsat-8, Sentinel-2, Google Earth Engine, Random Forest



The content of this work corresponds to the right of expression of the authors and does not compromise the institutional thinking of the University of Cuenca, nor does it release its responsibility before third parties. The authors assume responsibility for the intellectual property and copyrights.

Institutional Repository: <https://dspace.ucuenca.edu.ec/>

Índice de contenido

Resumen.....	2
1. Introducción.....	15
2. Objetivos.....	17
2.1. Objetivo general.....	17
2.2. Objetivos específicos (OE).....	17
3. Revisión bibliográfica.....	18
3.1. Teledetección.....	18
3.2. Cobertura del suelo.....	18
3.3. Cambios en la cobertura del suelo.....	19
3.4. Google Earth Engine.....	20
3.5. Interfaz de Google Earth Engine.....	20
3.6. Características de los satélites.....	21
3.6.1 Programa Landsat.....	21
3.6.2 Programa Sentinel.....	22
3.7. Preprocesamiento de imágenes en Google Earth Engine.....	23
3.8. Clasificación de imágenes.....	24
3.8.1. Clasificación supervisada.....	24
3.8.2. Clasificación no supervisada.....	24
3.9. Machine Learning.....	24
3.10. Random Forest (RF).....	25
3.11. Diagrama de Sankey.....	26
4. Metodología.....	27
4.1. Recursos.....	27
4.1.1. Programas.....	27
4.1.2. Datos.....	27
4.2. Zona de estudio.....	28
4.3. Metodología para el cumplimiento del objetivo específico 1.....	30
4.3.1 Datos utilizados.....	32

4.3.2 Preprocesamiento de Imágenes.....	34
4.3.3 Categorías de cobertura terrestre y datos de referencia (Datos de entrenamiento y validación).....	37
4.3.4 Clasificación de imágenes (Random Forest Classifier) y evaluaciones de precisión.....	42
4.3.5 Detección de cambio de cobertura del suelo.....	43
4.3.6 Validación.....	43
4.3.7 Matriz de Confusión.....	44
4.4. Metodología para el cumplimiento del objetivo específico 2.....	47
4.5. Metodología para el cumplimiento del objetivo específico 3.....	48
5. Resultados y discusión.....	49
5.1 Resultados del objetivo número 1.....	49
5.1.1 Identificación y cuantificación de Coberturas.....	49
5.1.2 Análisis de clasificaciones supervisadas en la cobertura del suelo.....	53
5.1.3 Dinámica de Cambio en las coberturas de suelo mediante diagramas de SANKEY.....	70
5.2 Resultados del objetivo número 2.....	76
5.2.1 Ventajas y desventajas de las imágenes de Landsat-8 y Sentinel-2.....	76
5.3 Resultados del objetivo número 3.....	79
6. Conclusiones.....	80
7. Recomendaciones.....	81
8. Referencias.....	82
9. Anexos.....	94

Índice de figuras

Figura 1. Interfaz de Google Earth Engine.....	21
Figura 2. Diagrama de bandas - Landsat-8.....	22
Figura 3. Diagrama de bandas - Sentinel-2.....	23
Figura 4. Zona de estudio correspondiente a la Subcuenca del río Burgay.....	29
Figura 5. Diagrama de Flujo.....	31
Figura 6. Nubosidad en las imágenes Landsat-8 y Sentinel-2.....	36
Figura 7. Fmask en imágenes Landsat-8 y Sentinel-2.....	37
Figura 8. Mapa de índice NDVI.....	39
Figura 9. Distribución de datos de referencia.....	40
Figura 10. Distribución de coberturas de suelo Landsat-8, en el transcurso de 2016, 2018, 2020.....	50
Figura 11. Distribución de coberturas de suelo Sentinel-2, en el transcurso de 2016, 2018 y 2020.....	52
Figura 12. Mapa de cobertura terrestre del área de la subcuenca del río Burgay, Landsat-8 en el año 2016.....	55
Figura 13. Mapa de cobertura terrestre del área de la subcuenca del río Burgay, Sentinel-2 en el año 2016.....	57
Figura 14. Mapa de cobertura terrestre del área de la subcuenca del río Burgay, Landsat-8 en el año 2018.....	60
Figura 15. Mapa de cobertura terrestre del área de la subcuenca del río Burgay, Sentinel-2 en el año 2018.....	63
Figura 16. Mapa de cobertura terrestre del área de la subcuenca del río Burgay, Landsat-8 en el año 2020.....	66
Figura 17. Mapa de cobertura terrestre del área de la subcuenca del río Burgay, Sentinel-2 en el año 2020.....	68
Figura 18. Diagrama de Sankey representando los cambios en la cobertura del suelo en los años 2016, 2018 y 2020 con Landsat-8.....	72
Figura 19. Diagrama de Sankey representando los cambios en la cobertura del suelo en los años 2016, 2018 y 2020 con Sentinel-2.....	73
Figura 20. Análisis optimizado de puntos; Subcuenca del río Burgay. Satélite Landsat-8....	74

Figura 21. Análisis optimizado de puntos; Subcuenca del río Burgay. Satélite Sentinel-2....	74
Figura 22. Portada principal, tutorial.....	79

Índice de tablas

Tabla 1. Datos utilizados.....	32
Tabla 2. Características de los satélites.....	34
Tabla 3. Interpretación Visual en RGB.....	38
Tabla 4. Categorías de cobertura de suelo en Google Earth Engine.....	41
Tabla 5. Principales componentes de la matriz de confusión.....	44
Tabla 6. Parámetros de precisión de clasificación.....	46
Tabla 7. Escala de valorización del índice Kappa.....	46
Tabla 8. Diferentes propiedades satelitales que presentan los sensores L-8 y S-2; seleccionadas para establecer ventajas y limitaciones entre los satélites.....	47
Tabla 9. Identificación y cuantificación de cobertura de suelo en la subcuenca del río Burgay, satélite Landsat-8 en los años 2016, 2018 y 2020 (ha, %)......	50
Tabla 10. Identificación y cuantificación de cobertura de suelo en la subcuenca del río Burgay, satélite Sentinel-2 en los años 2016, 2018 y 2020 (ha, %)......	51
Tabla 11. Matriz de confusión del sensor Landsat-8 en la subcuenca del río Burgay del año 2016.....	54
Tabla 12. Matriz de confusión del sensor Sentinel-2 en la subcuenca del río Burgay del año 2016.....	56
Tabla 13. Matriz de confusión del sensor Landsat-8 en la subcuenca del río Burgay del año 2018.....	58
Tabla 14. Matriz de confusión del sensor Sentinel-2 en la subcuenca del río Burgay del año 2018.....	61
Tabla 15. Matriz de confusión del sensor Landsat-8 en la subcuenca del río Burgay del año 2020.....	65
Tabla 16. Matriz de confusión del sensor Sentinel-2 en la subcuenca del río Burgay del año 2020.....	67
Tabla 17. Ganancias y pérdidas en las diferentes coberturas de suelo para los períodos 2016-2018, 2018-2020 para los satélites, Landsat-8 y Sentinel-2.....	71
Tabla 18. Resumen de comparación de los satélites Landsat-8 y Sentinel-2.....	78

Índice de anexos

Anexo A. Codificación Landsat 8 2016.....	94
Anexo B. Codificación Landsat 8 2018.....	99
Anexo C. Codificación Landsat 8 2020.....	104
Anexo D. Codificación Sentinel 2 2016.....	109
Anexo E. Codificación Sentinel 2 2018.....	114
Anexo F. Codificación Sentinel 2 2020.....	119
Anexo G. Tabla de valores Precisión general y Coeficiente Kappa - Intervalvalidation, PlanetLAB de Landsat 8 y Sentinel 2.....	124
Anexo H. Comparación de análisis de datos en QGIS vs GEE.....	125

Abreviatura y simbología

GEE: Google Earth Engine.

USGS: Servicio Geológico de los Estados Unidos.

NASA: Administración Nacional de Aeronáutica y el Espacio.

OLI: Imagen terrestre operativa.

TIRS: Sensor infrarrojo térmico

ESA: Agencia Espacial Europea.

SWIR: Short Wave Infrared (acrónimo con el que se designa a la parte del espectro electromagnético del infrarrojo de onda corta).

S2: Sentinel-2

L8: Landsat-8

SVM: Máquinas de Vectores de Soporte

PCA: Análisis de Componentes Principales

RF Random Forest

QGIS: Quantum GIS

IGM: Instituto Geográfico Militar Ecuatoriano

Agradecimientos

Quiero expresar mi más profundo agradecimiento a Dios y a la Virgen María por brindarme una nueva oportunidad de vida, por guiar mi camino en cada etapa de esta tesis y permitirme culminarla. Agradezco a mis padres, hermanas, y sobrinos, quienes han sido mi mayor fuente de apoyo y motivación, gracias por su amor incondicional, su paciencia y su constante aliento. A mi tutor le agradezco por su compromiso, orientación y sus palabras de aliento. Y a Camila, gracias por ser un pilar importante en mi vida, por todo tu apoyo, entrega y sobre todo comprensión.

María Augusta Murillo

Agradecimientos

En primer lugar, quiero agradecerle a Dios por haberme permitido llegar a este punto de mi vida con salud y culminar este trabajo. A mis padres y a mi hermana que siempre me brindaron su apoyo y amor incondicional, muchas gracias. Le agradezco también a mi tutor por su dedicación y extrema paciencia, sin su guía y correcciones precisas no hubiera logrado llegar a esta instancia tan anhelada. Un agradecimiento especial a mi amiga y compañera de tesis Magus, tú optimismo y amabilidad hizo el camino más ameno y divertido.

María Camila Obando.

Dedicatoria

Quiero dedicar esta tesis a mis padres amados, Nelly y Alfonso, para honrar todo el amor, sacrificio y esfuerzo que han hecho por mí. Su ejemplo de perseverancia, dedicación y trabajo arduo me ha inspirado a dar lo mejor de mí en cada aspecto de mi vida, incluyendo mis estudios académicos.

Dedico también esta tesis a mis hermanas Carolina y Paola, por todo su apoyo, amor y paciencia. Y finalmente a mis sobrinos Esteban y Emilia, por ser la alegría de mi vida y mi motivación para seguir adelante. Los amo con todo mi corazón.

María Augusta Murillo

Dedicatoria

Quiero dedicar este trabajo a mi mamá, María Ofelia. Mi súper heroína y ejemplo de vida, sin ti mami no lo hubiera logrado. Tu dedicación, amor y valores que me inculcaste han sido la base de mi vida y éxito académico. Sin tu apoyo incondicional y tus sabias palabras, esta tesis no hubiera sido posible. Gracias por ser mi guía, mi inspiración y mi refugio. Gracias mamá.

También quiero dedicar este trabajo a mi papá, Francisco; las charlas y risas compartidas me brindaron una perspectiva de vida más libre y sin miedo. Finalmente quiero dedicar este trabajo a mi hermana Laura por ser un ejemplo de superación y fortaleza de vida, tú me inspiras Lauris.

María Camila Obando.

1. Introducción

Aparentemente uno de los indicadores más evidentes que demuestran los cambios superficiales de la Tierra, especialmente en relación con el suelo, es la cobertura de suelo. Estudios recientes demuestran que la extensión y calidad de la cobertura de suelo está disminuyendo a escala local y global debido a las actividades antrópicas y el cambio climático (Dwire & Brown, 2017). Es decir, el cambio en la cobertura de suelo está teniendo un impacto negativo en diversos aspectos de la superficie terrestre, como los ecosistemas terrestres, la biodiversidad y el clima (Díaz & Gaspari, 2017).

Esto genera un impacto en los ecosistemas terrestres y acuáticos, que afecta a la capacidad de proveer servicios ambientales a la población y produce efectos directos como en el calentamiento global, la erosión del suelo, la pérdida de la biodiversidad, la seguridad alimentaria e incluso a la salud pública (Uribe, 2015). El surgimiento de herramientas de teledetección y sistemas de información geográfica, que proveen una visión gráfica espacialmente explícita de las características de la superficie del planeta, se han constituido en una fuente de información esencial al momento de la caracterización de la cobertura del suelo y del estudio del cambio a través del tiempo (Pastrana, 2021).

El empleo de imágenes satelitales ha ganado relevancia, especialmente debido a su capacidad para abarcar áreas extensas, incluso aquellas de acceso complicado (Gorelick et al., 2017). Además, la rapidez con la que se pueden obtener resultados preliminares de las áreas de interés y la disponibilidad de una variedad de satélites que ofrecen información en distintas resoluciones e intervalos de tiempo hacen que sea fundamental en la investigación científica (Orimoloye, 2018).

Los estudios referentes a la detección de cambios de cobertura de suelo en amplias zonas, tienden a ser realizados en temporalidades amplias, por lo que la información existente no es del todo constante, considerando que algunas regiones tienen una dinámica más activa que otras (AHADY, 2022). Tal es el caso de la subcuenca del río Burgay, perteneciente a la cuenca del río Paute, la cual es una importante arteria hidrográfica del sur del país y representa un eje de procesos ambientales, sociales y económicos (Pauta & Chang, 2016). En esta subcuenca se han desencadenado múltiples problemas ambientales por las actividades antrópicas y la mala gestión de recursos (Cardenas & Quesada, 2007).

Sería muy laborioso si procesamos los datos utilizando los métodos tradicionales, desde la búsqueda, el filtrado, la descarga y la creación de mosaicos hasta el preprocesamiento, como el enmascaramiento de nubes o la corrección atmosférica.

Estos datos además de requerir mucho esfuerzo y tiempo, también requieren una capacidad de almacenamiento significativa y acceso a computación de alta potencia. La plataforma de computación en la nube Google Earth Engine (GEE) se puede utilizar para procesar una gran cantidad de imágenes rápidamente, debido a que proporciona una forma de procesamiento concurrente (Phan, 2020). La plataforma no solo almacena grandes cantidades de datos de una variedad de sistemas satelitales de observación de la tierra, sino que también permite a los usuarios cargar sus propios datos para su procesamiento, lo cual brinda y nos brindó en el desarrollo de nuestro estudio la posibilidad de hacer un uso integral de los datos de detección remota. Además, también cuenta con la capacidad de computación en la nube más avanzada, diseñada para el procesamiento de big data (Zhang, 2022)

Por estas razones, el presente estudio utiliza la ventaja de la nueva tecnología geoespacial de GEE y el registro histórico de datos satelitales Landsat-8 y Sentinel-2 para investigar el cambio de cobertura del suelo dentro de la subcuenca del río Burgay y evaluar las ventajas y limitaciones de estos satélites en el análisis de las coberturas del suelo en la zona andina.

2. Objetivos

2.1. Objetivo general

Analizar los cambios de la cobertura del suelo en la subcuenca del río Burgay mediante el uso de imágenes Sentinel-2 y Landsat-8.

2.2. Objetivos específicos (OE)

OE 1: Identificar y cuantificar los principales tipos de cobertura del suelo y sus cambios en la subcuenca del Río Burgay mediante técnicas de teledetección.

OE 2: Comparar las ventajas y desventajas de las imágenes de Landsat-8 y Sentinel-2 para el estudio de la cobertura del suelo en la zona andina.

OE 3: Producir un tutorial para el análisis de cambio de cobertura del suelo en cuencas andinas usando imágenes satelitales.

3. Revisión bibliográfica

3.1. Teledetección

La teledetección es un método empleado para adquirir datos sobre la superficie terrestre y otros objetos desde una distancia, utilizando sensores remotos instalados en aviones o satélites (Pontius, 2004). Esta disciplina combina los principios de la física, la fotogrametría y la ciencia de los datos para adquirir, procesar e interpretar las imágenes y datos capturados por los sensores (Graciani et al., 2011). La teledetección permite obtener información sobre la cobertura del suelo, la vegetación, los recursos hídricos, la calidad del aire y otros aspectos ambientales, lo que resulta fundamental para el monitoreo y la gestión del medio ambiente (Veneros et al., 2020).

La teledetección se basa en la interacción entre la radiación electromagnética emitida o reflejada por la Tierra y los objetos en su superficie (Potavov, 2012). Los sensores remotos capturan esta radiación en diferentes rangos del espectro electromagnético, incluyendo la luz visible, el infrarrojo y el microondas. La información recopilada se utiliza para generar imágenes y datos que revelan características de la superficie terrestre y sus cambios a lo largo del tiempo (Arenas, 2016).

Las aplicaciones de la teledetección son amplias y abarcan áreas como la agricultura, la gestión de recursos naturales, la planificación urbana, la detección de desastres naturales y el estudio del cambio climático (Ponce, 2010). La disponibilidad de imágenes satelitales de alta resolución, así como el desarrollo de técnicas de procesamiento de imágenes y análisis espacial, han impulsado el avance de la teledetección y su uso en una variedad de campos científicos y aplicaciones prácticas (Rodríguez et al., 2005).

3.2. Cobertura del suelo

Las coberturas del suelo se refieren a la distribución y composición de los diferentes tipos de superficie terrestre en una determinada área geográfica. Estas coberturas representan las diferentes clases del suelo, como bosques, cultivos, cuerpos de agua, áreas urbanas, pastizales y otros tipos de vegetación (Zambrano, 2021).

El análisis de las coberturas de suelo es fundamental para comprender los patrones espaciales, los cambios temporales y las interacciones entre los diferentes componentes del ecosistema (Saldaña, 2019). Las coberturas de suelo se pueden identificar y clasificar mediante técnicas de teledetección, que utilizan imágenes satelitales y otros datos geoespaciales para obtener información sobre la composición y distribución de los distintos tipos de cobertura. Estos datos se analizan utilizando algoritmos de clasificación y análisis

espacial para generar mapas de cobertura terrestre. Estos mapas proporcionan una representación visual y cuantitativa de las diferentes coberturas presentes en un área y son utilizados en diversas aplicaciones, como la planificación del uso del suelo, la conservación ambiental y la gestión de recursos naturales (Poveda et al., 2022).

Las coberturas del suelo son dinámicas y están sujetas a cambios debido a factores naturales y humanos, como la deforestación, la urbanización, la agricultura intensiva y los eventos climáticos extremos (IPCC, 2019). El monitoreo de los cambios en las coberturas del suelo a lo largo del tiempo es esencial para evaluar los impactos ambientales, identificar tendencias y desarrollar estrategias de manejo sostenible. Además, el análisis de las coberturas del suelo puede ayudar a comprender los procesos ecológicos, evaluar la biodiversidad y predecir posibles escenarios futuros (García et al., 2022).

3.3. Cambios en la cobertura del suelo

Los cambios en la cobertura del suelo se refieren a las modificaciones que ocurren en la distribución y composición de las diferentes superficies terrestres a lo largo del tiempo. Estos cambios pueden ser causados por una variedad de factores, como la urbanización, la deforestación, la agricultura intensiva, los incendios forestales y los fenómenos naturales (Carvajal & Pabón, 2016). El análisis de los cambios en la cobertura del suelo es fundamental para comprender las transformaciones del paisaje y sus implicaciones ambientales, sociales y económicas (Montero & Viales, 2015).

Los cambios en la cobertura del suelo se pueden monitorear utilizando técnicas de teledetección y análisis espacial. Las imágenes satelitales y otros datos geoespaciales proporcionan información detallada sobre la distribución de las diferentes coberturas de suelo en diferentes momentos (Rosero, 2021). El análisis de series temporales de imágenes permite identificar y cuantificar los cambios en la cobertura del suelo, así como analizar las causas y los impactos asociados. Esta información es invaluable para la planificación del uso del suelo, la conservación de los recursos naturales y la evaluación de políticas y estrategias de manejo ambiental (Correa, 2018).

El monitoreo de los cambios en la cobertura del suelo es esencial para evaluar los impactos ambientales y socioeconómicos, así como para tomar decisiones informadas en relación con el desarrollo sostenible (IPCC, 2019). Permite identificar áreas de deforestación, degradación del suelo o expansión urbana, así como áreas de regeneración o conservación. Además, proporciona una base para evaluar la pérdida de hábitats, la fragmentación del paisaje y la calidad del agua y los suelos. El seguimiento continuo de los cambios en la

cobertura del suelo es fundamental para implementar estrategias de gestión efectivas y promover un uso sostenible de los recursos naturales (Camacho, 2017).

3.4. Google Earth Engine

Google Earth Engine es una plataforma en línea desarrollada por Google que permite el acceso y análisis de una gran cantidad de datos de teledetección y geoespaciales de manera rápida y eficiente. Esta plataforma integra una gran cantidad de imágenes satelitales, datos climáticos, información geográfica y otras fuentes de datos en una infraestructura de computación en la nube (GEE, 2023). Proporciona herramientas y algoritmos de procesamiento y análisis geoespacial que permiten a los usuarios realizar investigaciones, monitorear cambios ambientales, realizar análisis de cobertura terrestre (Gorelick et al., 2017).

3.5. Interfaz de Google Earth Engine

Google Earth Engine ofrece una interfaz de usuario que facilita la interacción de los usuarios con la plataforma para realizar análisis geoespaciales. A través de esta interfaz, los usuarios pueden acceder a diversas funciones y herramientas para trabajar con datos geoespaciales y ejecutar análisis personalizados (GEE, 2023).

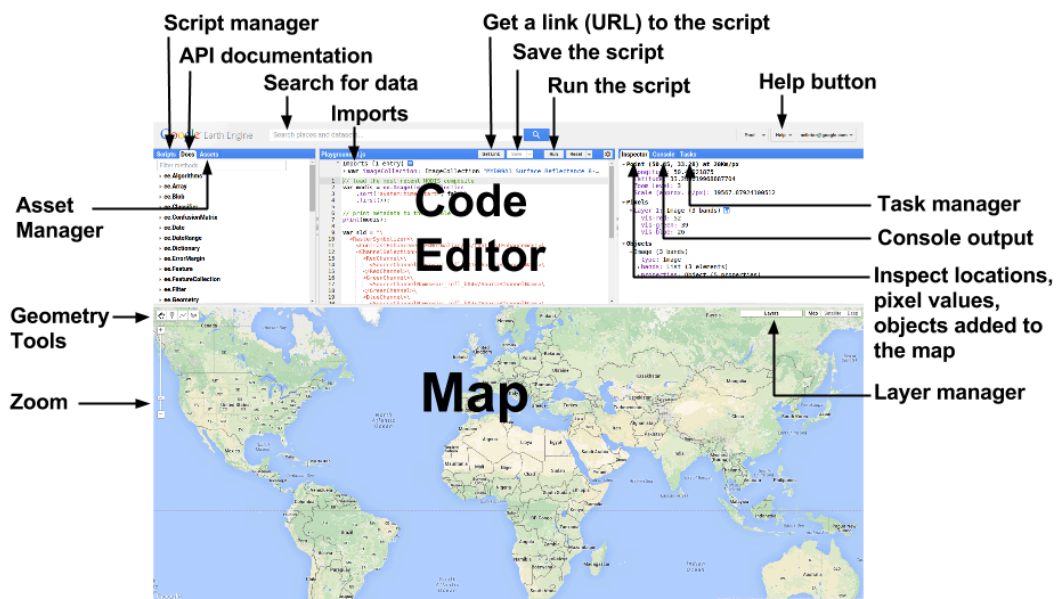
Los componentes clave de la interfaz de Google Earth Engine son:

1. Editor de Código: Proporciona un entorno de desarrollo integrado (IDE) donde los usuarios pueden escribir, probar y ejecutar código en JavaScript o Python para crear y modificar scripts de análisis geoespacial (GEE, 2023).
2. Explorador de Datos: Permite a los usuarios explorar el catálogo de datos disponibles en Google Earth Engine, seleccionar imágenes satelitales, conjuntos de datos climáticos, datos de elevación y otros conjuntos de datos geoespaciales para su análisis (GEE, 2023).
3. Map Viewer: Ofrece una interfaz de mapa interactiva que permite a los usuarios visualizar y analizar datos geoespaciales, aplicar operaciones de análisis y realizar consultas espaciales para obtener información detallada sobre una región específica (GEE, 2023).
4. Panel de Resultados: Muestra los resultados de los análisis y visualizaciones realizadas por los usuarios, generando gráficos, diagramas y mapas interactivos para comunicar los hallazgos de manera efectiva (GEE, 2023).

5. Bibliotecas y Herramientas: Google Earth Engine proporciona una variedad de bibliotecas y herramientas de análisis geoespacial que los usuarios pueden utilizar en sus proyectos, incluyendo funciones para el procesamiento de imágenes, el cálculo de índices y el análisis de series temporales (GEE, 2023).

Figura 1

Interfaz de Google Earth Engine



Fuente: USGS, 2023.

3.6. Características de los satélites

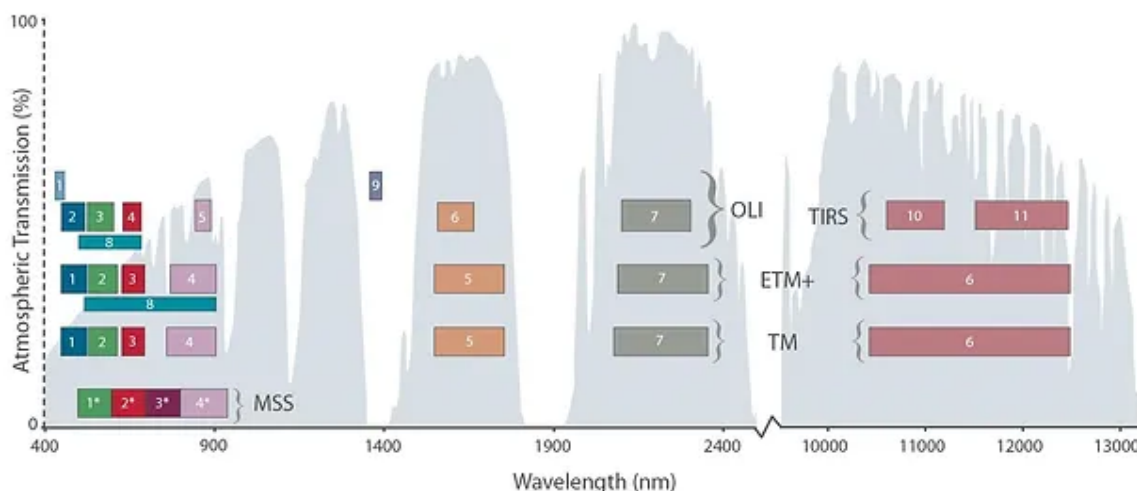
3.6.1 Programa Landsat

El programa Landsat, dirigido por la NASA y el Servicio Geológico de los Estados Unidos (USGS), consiste en una sucesión de misiones satelitales diseñadas para adquirir y suministrar datos de imágenes de la Tierra durante un extenso período de tiempo. Estas misiones han sido fundamentales para monitorear y comprender cambios en la superficie terrestre, como la cobertura vegetal, los recursos hídricos, la urbanización y otros aspectos ambientales. Desde su lanzamiento inicial en 1972 con Landsat-1, ha habido varias misiones sucesivas que han proporcionado imágenes continuas y de larga duración de la superficie terrestre (USGS, 2023). La misión ha incluido siete satélites, siendo el Landsat-8 el más reciente en ser lanzado al espacio en 2013, además de ser el más tecnológicamente avanzado dentro de esta misión. Cuenta con dos sensores OLI y TIRS. Las resoluciones de

dicho sensor son: radiométrica de 16 bits, espacial de 30 m, espectral nueve bandas, 5 de ellas en el visible y las cuatro restantes en el infrarrojo cercano y medio (Picone, 2017).

Figura 2

Diagrama de bandas - Landsat-8



Bandpass wavelengths for Landsat-8 OLI and TIRS sensor, compared to Landsat-7 ETM + sensor.

Note: atmospheric transmission values for this graphic were calculated using MODTRAN for a summertime mid- latitude hazy atmosphere (circa 5 km visibility).

Nota. Ancho de banda para los sensores OLI y TIRS en Landsat-8 y ETM+ en Landsat-7.

Fuente: USGS, 2013.

3.6.2 Programa Sentinel

El programa Sentinel, desarrollado por la Agencia Espacial Europea (ESA, 2023) en colaboración con la Comisión Europea, consiste en una serie de misiones satelitales cuyo objetivo principal es proporcionar datos de observación de la Tierra para una amplia gama de aplicaciones. Estas incluyen la monitorización ambiental, la gestión de recursos naturales, la vigilancia marítima y la respuesta a desastres. El programa Sentinel se integra en el marco más amplio del programa Copernicus. Dentro de la constelación Sentinel, que se enfoca en la observación de la tierra, se utilizan los satélites Sentinel 1, 2 y 3 en el ámbito terrestre (Copernicus, 2015).

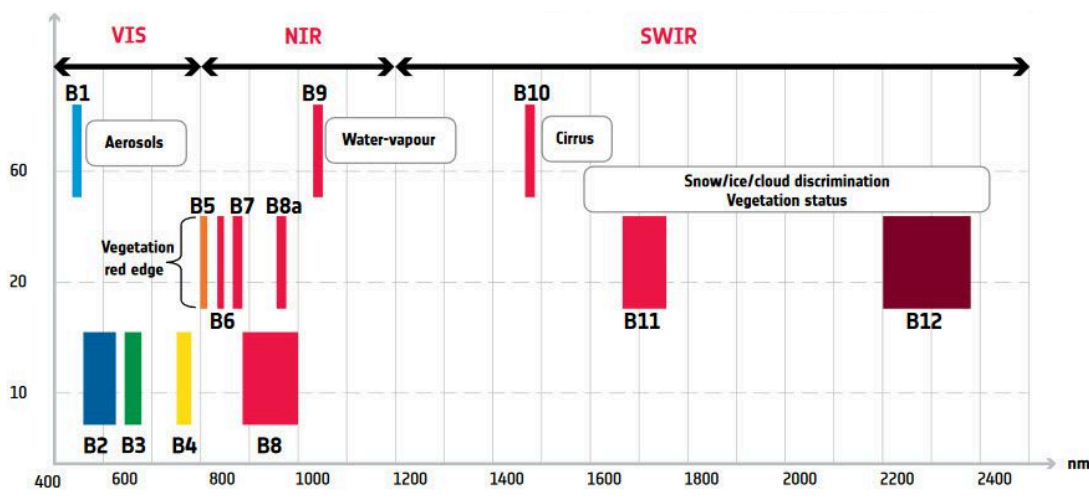
La misión Sentinel-2 está compuesta por una pareja de satélites multispectrales en órbita polar, diseñados para la vigilancia de la Tierra. Esta misión ofrece imágenes que se utilizan en diversas aplicaciones, como la monitorización de la vegetación, el suelo y el

agua, así como para el seguimiento de vías de navegación interior y áreas costeras. Además, Sentinel-2 puede proporcionar información útil para servicios de emergencia. La constelación consta de dos satélites idénticos, denominados Sentinel 2A y Sentinel 2B, que se encuentran en la misma órbita, pero separados por 180°, lo que garantiza una cobertura óptima de la Tierra (Picone, 2017).

Dispone de 13 bandas, entre las cuales cuatro (en azul, verde, rojo e infrarrojo cercano) tienen una resolución espacial de 10 metros, seis tienen una resolución de 20 metros, que incluyen bandas en red-edge y en el SWIR, mientras que las tres restantes tienen una resolución de 60 metros para la corrección atmosférica y la detección de nubes. El primer satélite de la serie S2 fue lanzado el 23 de junio de 2015. La Agencia Espacial Europea (ESA) ha proporcionado de forma gratuita las imágenes, así como una caja de herramientas para la visualización y el procesamiento de imágenes S2 (Delegido et al., 2016).

Figura 3

Diagrama de bandas - Sentinel-2



Fuente: ESA, 2023.

3.7. Preprocesamiento de imágenes en Google Earth Engine

El preprocesamiento de imágenes en Google Earth Engine se refiere a las operaciones y técnicas aplicadas a las imágenes antes de su análisis y procesamiento. Estas operaciones tienen como objetivo mejorar la calidad de las imágenes, corregir distorsiones, reducir el ruido y prepararlas para su posterior uso en análisis geoespaciales (Perilla & Mas, 2020).

Algunas de las operaciones comunes de preprocesamiento en GEE incluyen el recorte de imágenes para seleccionar áreas de interés, el cambio de proyección y resolución de las imágenes, la mejora del contraste y la visualización de las imágenes, la eliminación de artefactos y el filtrado del ruido. Estas operaciones se realizan mediante el uso de las funciones y herramientas proporcionadas por Google Earth Engine, que permiten a los usuarios procesar y manipular imágenes satelitales y de otros tipos para su posterior análisis y extracción de información (Ji, 2020).

3.8. Clasificación de imágenes

La clasificación de imágenes es un proceso esencial en el análisis de datos de teledetección, que implica asignar etiquetas o categorías a los píxeles o regiones de una imagen en función de características específicas (Richards, 2006).

3.8.1. Clasificación supervisada

La clasificación supervisada implica utilizar muestras de entrenamiento etiquetadas, donde se conocen las categorías o clases a las que pertenecen los píxeles. Estas muestras se utilizan para entrenar un algoritmo de clasificación que luego se aplica a la imagen completa para asignar etiquetas a los píxeles no etiquetados. Algunos algoritmos comunes utilizados en la clasificación supervisada incluyen Árboles de Decisión, Máquinas de Vectores de Soporte (SVM) y Redes Neuronales Artificiales (Ruíz, 2017).

3.8.2. Clasificación no supervisada

La clasificación no supervisada, por otro lado, agrupa los píxeles en clases sin la necesidad de muestras de entrenamiento previamente etiquetadas. El algoritmo analiza las similitudes y diferencias entre los píxeles para agruparlos en clases distintas. Los algoritmos de agrupamiento, como el K-means y el Análisis de Componentes Principales (PCA), son comúnmente utilizados en la clasificación no supervisada (Ruíz, 2017).

3.9. Machine Learning

El Machine Learning es un área dentro del ámbito de la inteligencia artificial que se centra en desarrollar algoritmos y modelos que permiten a las computadoras aprender y tomar decisiones basadas en datos, sin requerir programación explícita para cada tarea específica. El Machine Learning utiliza algoritmos que aprenden de manera autónoma a medida que se les presenta más información y datos. Estos algoritmos son capaces de detectar patrones y relaciones en los datos, aprender de ellos y utilizar ese conocimiento

para hacer predicciones o tomar decisiones en situaciones similares en el futuro (IBM, 2022).

El Machine Learning se basa en el procesamiento de grandes volúmenes de datos, ya que cuanto más información se proporcione al algoritmo, mejor será su capacidad para aprender y realizar predicciones precisas. Los diferentes enfoques del Machine Learning incluyen el aprendizaje supervisado, donde se proporcionan datos de entrenamiento etiquetados para que el algoritmo aprenda a asociar características con etiquetas predefinidas, el aprendizaje no supervisado, donde el algoritmo busca patrones y estructuras ocultas en los datos sin etiquetar, y el aprendizaje por refuerzo, donde el algoritmo aprende a través de la interacción con un entorno y la obtención de recompensas o castigos (Simeone, 2018).

El Machine Learning tiene numerosas aplicaciones en diversos campos, como la medicina, la industria, el comercio, la seguridad, la predicción del clima, el procesamiento de imágenes y el procesamiento del lenguaje natural, entre otros. Se ha convertido en una herramienta poderosa para extraer conocimiento valioso de grandes conjuntos de datos y mejorar la toma de decisiones en diferentes industrias y sectores (Maisueche, 2018).

3.10. Random Forest (RF)

RF es un algoritmo de clasificación conjunto, que utiliza la agregación de arranque o empaquetado para generar un conjunto de clasificación (usando árboles de decisión múltiples) cada árbol entrena un cierto subconjunto de datos de entrenamiento completos (Waske & Braun, 2009).

Según Mahdianpari et al (2018) y Xia et al. (2018) las razones por las que RF consiguió un interés considerable en las últimas dos décadas son las siguientes:

1. Buen manejo de los valores atípicos y los conjuntos de datos más ruidosos;
2. Buen rendimiento con alto dimensional y multi-fuente conjuntos de datos;
3. Mayor precisión que otros clasificadores populares,
4. Aumenta la velocidad de procesamiento mediante la selección de variables importantes.

Otro factor que hizo su popularidad sobre otros algoritmos de aprendizaje automático, es su codificación simple, ya que solo se requiere optimizar dos parámetros: el número de árboles de clasificación deseados y el número de variables de predicción (ntree y mtry) utilizadas en cada nodo para hacer crecer el árbol, lo que facilita la aplicación (Rodríguez-Galiano et al., 2012).

3.11. Diagrama de Sankey

El diagrama de Sankey es una representación gráfica utilizada para visualizar flujos de energía, materiales o recursos en un sistema dado. Su estructura se basa en la disposición de nodos interconectados por flechas de ancho proporcional a la magnitud del flujo que representan. Estas flechas pueden apuntar en varias direcciones, lo que permite mostrar tanto flujos entrantes como salientes en un sistema. La principal utilidad del diagrama de Sankey radica en su capacidad para proporcionar una representación visual intuitiva de cómo se distribuyen los recursos o la energía a lo largo de un proceso o sistema. Al observar el diagrama, los usuarios pueden identificar fácilmente las principales fuentes de entrada, los puntos de consumo o salida, así como los flujos intermedios entre diferentes etapas del proceso (Hernandez, 2016).

4. Metodología

4.1. Recursos

4.1.1. Programas

1. Google Earth Engine

Usamos la plataforma Google Earth Engine ya que su funcionamiento está basado en la nube permitiendo el análisis y procesamiento de grandes volúmenes de datos geoespaciales de manera eficiente (Evans & Rodriguez, 2018).

2. QGIS

La plataforma QGIS fue la que utilizamos para la presentación de los mapas por su versatilidad y amplio uso en la comunidad de SIG debido a su capacidad de visualizar, analizar y editar datos geoespaciales, desde imágenes satelitales y mapas topográficos hasta datos vectoriales y raster (Ortega & Cantos, 2016).

4.1.2. Datos

- Conjunto de datos de límites de cuencas y subcuencas hidrográficas proporcionados por el Geoportal del instituto geográfico militar ecuatoriano (IGM, 2015).
- Datos de la plataforma Google Earth Engine proporcionados por el Servicio Geológico de EE.UU (USGS) como fuente de las imágenes satelitales Landsat-8 en los años 2016, 2018, 2020 (USGS, 2023)
- Datos de la plataforma Google Earth Engine proporcionados por Copernicus como fuente de las imágenes satelitales Sentinel-2 en los años 2016, 2018, 2020 (Copernicus, 2015).
- Base de datos de cobertura terrestre mundial proporcionado por el Servicio Geológico de EE.UU USGS (USGS, 2023).
- Capa adicional de índices de vegetación (NDVI, NDWI y NDBI) creada mediante codificación en la plataforma Google Earth Engine para la identificación de coberturas según índices.

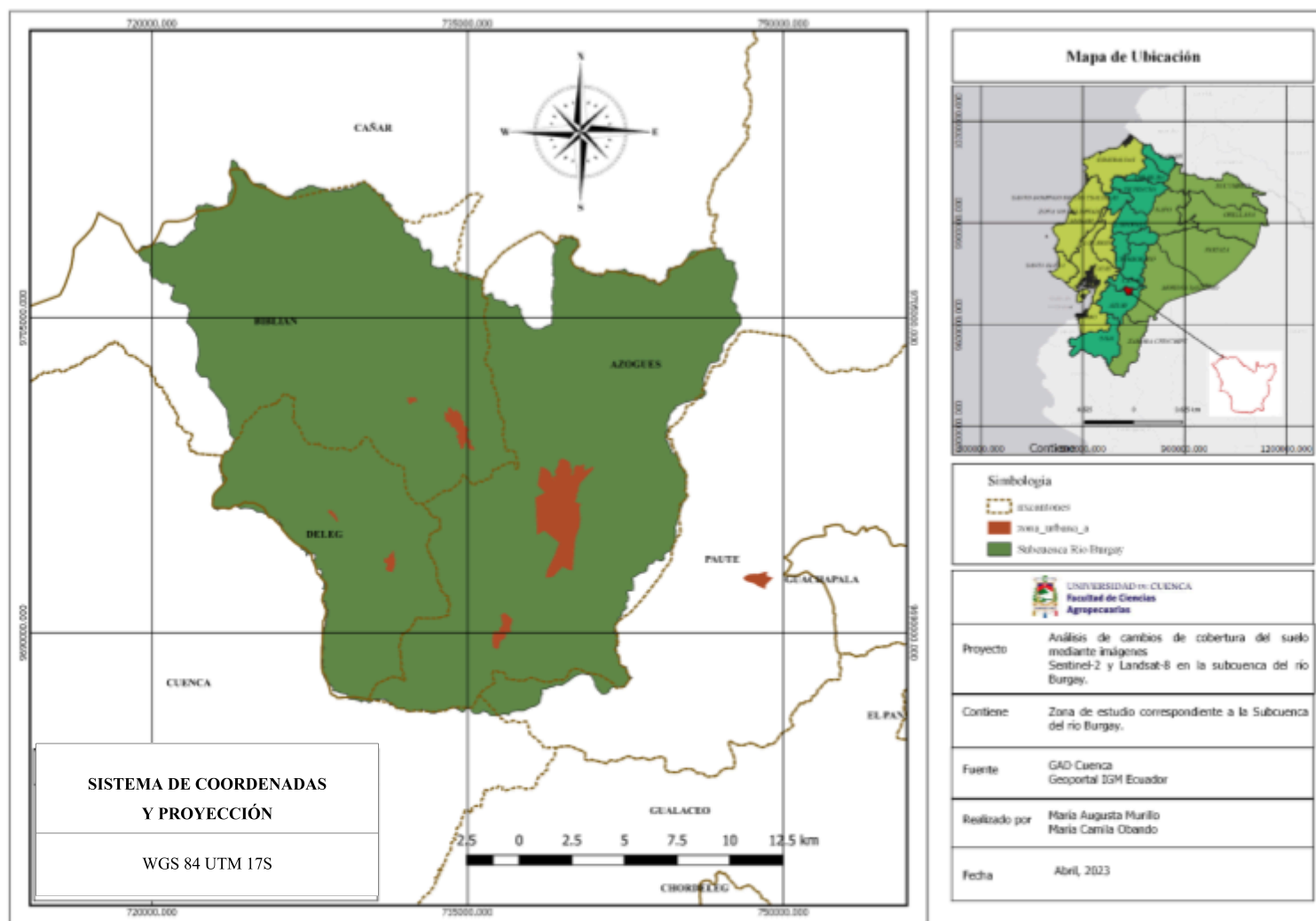
4.2. Zona de estudio

La zona de estudio corresponde a la Subcuenca del río Burgay, cuyas coordenadas son 2°49'43"S 78°52'48"O (*Figura 4*) localizada en la provincia del Cañar entre las ciudades Biblián y Azogues, la misma que forma parte de la Cuenca del río Paute. La subcuenca del río Burgay representa el 8,7% del área total de esta cuenca, con un área de 447,04 km² y una pendiente promedio de 15,4% (Calle & Chang, 2016). La subcuenca se caracteriza por tener un relieve montañoso, con un rango altitudinal que va desde los 2300 hasta los 4200 msnm.

Hay una variedad de coberturas presentes en la subcuenca del río Burgay. La vegetación predominante consiste en vegetación leñosa (23,9%), la cual se localiza principalmente en la zona media de la cuenca. En las áreas más altas de la cuenca se encuentran pastizales (19,5%) y vegetación de páramo (13,9%), mientras que en las áreas más bajas de la cuenca prevalece la presencia de eucaliptos/matorrales abiertos (17%) y matorrales intervenidos (17%) (Timbe & Timbe, 2012).

Figura 4

Zona de estudio correspondiente a la Subcuenca del río Burgay



4.3. Metodología para el cumplimiento del objetivo específico 1

Para la identificación y cuantificación de las coberturas más representativas en la subcuenca del río Burgay y su consiguiente detección de cambio lo aplicamos y evaluamos mediante el desarrollo de código en la plataforma Google Earth Engine utilizando el algoritmo clasificador supervisado Random Forest y series de tiempo de imágenes Landsat-8 y Sentinel-2 en presentación de “Composites” anuales posteriormente para cada año elegido. Estos Composites se refieren a imágenes o conjuntos de datos que se han generado combinando múltiples imágenes individuales en una sola imagen compuesta.

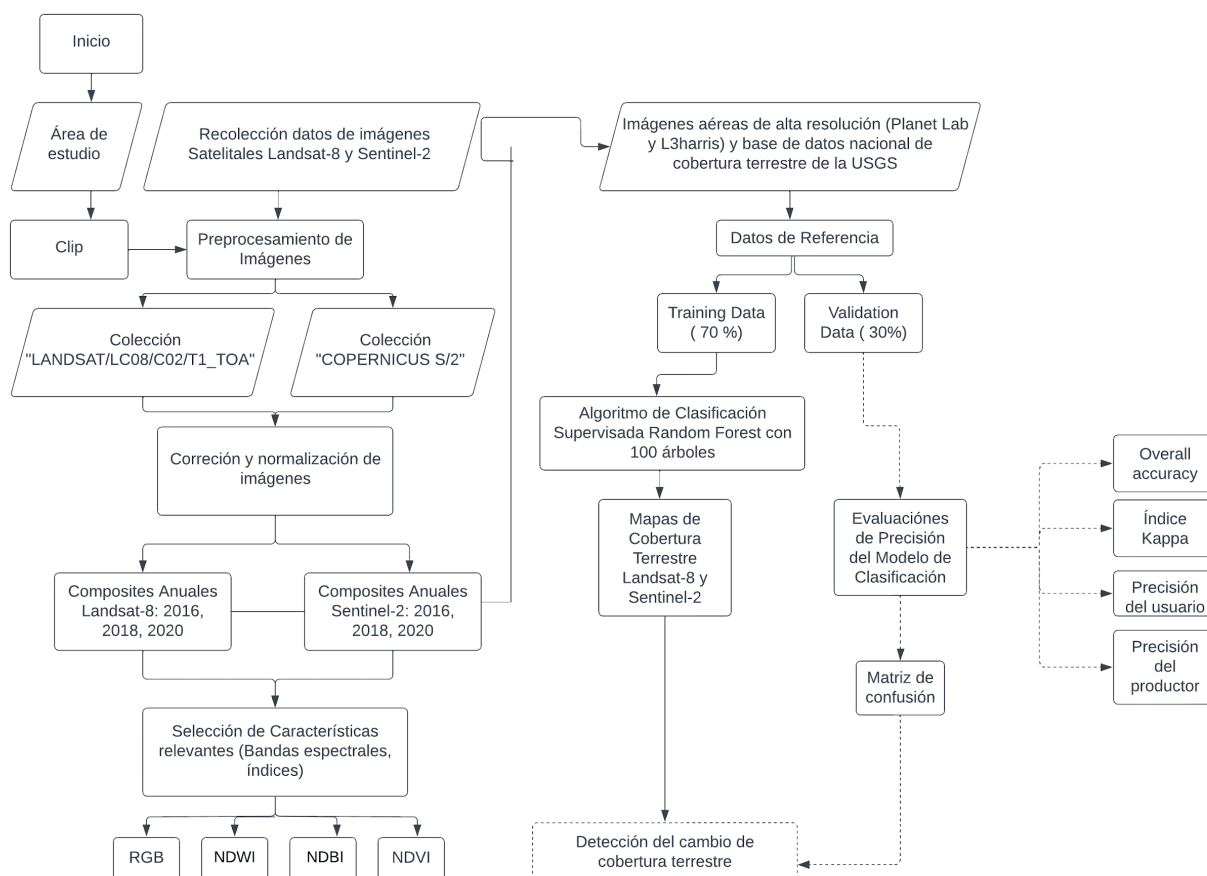
Utilizamos imágenes aéreas de alta resolución disponibles en GEE, como capa de referencia para entrenar y validar las clasificaciones. Hicimos comparaciones entre las imágenes clasificadas para cada año de estudio para encontrar lugares donde hubo detección de cambio y ausencia de cambio en la cobertura del suelo.

Cabe destacar que, todas las imágenes que utilizamos en este estudio las preprocesamos y normalizamos en la plataforma Google Earth Engine mediante el desarrollo de código para cada uno de los grupos de las imágenes Landsat-8 y Sentinel-2 que empleamos, fue indispensable Google Earth Engine en todo el proceso de tratamiento de imágenes; también utilizamos el conjunto de datos de referencia; todo esto previo a la clasificación de la cobertura del suelo.

El análisis que realizamos posteriormente identificó áreas modificadas a través del tiempo. Los procedimientos generales se resumen a continuación y de manera concisa en el diagrama de flujo ilustrado en la *Figura 5*. En esta figura, el preprocesamiento de imágenes, la categorización de la cobertura terrestre y la clasificación de imágenes se muestra como el paso 1, y la detección del cambio de cobertura de suelo se muestra como el paso 2.

Figura 5

Diagrama de Flujo

_____ *paso 1*----- *paso 2*

Nota. Diagrama de flujo del procesamiento de datos. Expone el procedimiento para el preprocesamiento de imágenes, categorización de coberturas, clasificación de imágenes como el paso 1 y la detección de cambio de cobertura como el paso 2.

4.3.1 Datos utilizados

4.3.1.1 Landsat-8 Surface Reflectance Tier 1 data (L8sr)

Para centrarnos de la mejor manera en la selección de imágenes en la clasificación de la cobertura terrestre, Google Earth Engine nos proporcionó acceso en línea a los datos de Landsat archivados como una colección del USGS (Huang et al., 2017). Luego procedimos a seleccionar en la biblioteca de la USGS la colección “*Landsat-8 Surface Reflectance Tier 1 data (L8sr)*” que mostró tener dentro de su disposición características como los datos de nivel 1, que representan la información cruda capturada por el sensor antes de aplicar correcciones atmosféricas u otros procesamientos específicos.

4.3.1.2 Copernicus Open Access Hub S2 (CS2)

Los datos para las imágenes del satélite Sentinel-2 los recolectamos del programa *Copernicus*, que cuenta hasta el momento con 7 misiones disponibles (ESA,2023).

Además, también utilizamos datos del geoportal del instituto geográfico militar ecuatoriano, para la descarga del conjunto de datos de límites de cuencas y subcuencas hidrográficas. Finalmente agregamos datos adicionales, como capas necesarias para la realización del estudio. Todos los datos utilizados se resumen en la *Tabla 1*, y las características de los satélites se resumen en la *Tabla 2*.

Tabla 1

Datos utilizados

Capa de Datos	Fuente	Resolución Espacial (m)	Fecha
Conjunto de datos de límites de cuencas y subcuencas hidrográficas.	Geoportal del instituto geográfico militar ecuatoriano	1:50000	2013
USGS Landsat-8 Reflectancia de superficie (SR), Collection 1 Tier 1 TOA Reflectancia	Datos de la plataforma Google Earth Engine proporcionado por el Servicio Geológico de EE.UU (USGS)	30	2016,2018,2020 Identificador de colección: LANDSAT/LCO8/CO1/T1_TOA¹

Sentinel-2 Copernicus S2	Datos de la plataforma Google Earth Engine proporcionados por Copernicus	10	2016,2018,2020	Identificador de colección: COPERNICUS S/2 ²
Base de datos de cobertura terrestre	Datos de la plataforma Google Earth Engine proporcionado por el Servicio Geológico de EE.UU (USGS)	30	2016,2018,2020	
NDVI, NDWI y NDBI	Capa adicional de índices creada por Autoras mediante codificación en la plataforma Google Earth Engine para la identificación de coberturas según índices.	30, 10	2016,2018,2020	
Imágenes aéreas de alta resolución	Datos de la plataforma Google Earth Engine proporcionado por el Servicio Geológico de EE.UU (USGS)	3	2016,2018,2020	

Nota. Fuentes de datos y descripciones utilizadas en el desarrollo de la clasificación supervisada de cobertura terrestre.

Tabla 2

Características de los satélites

Plataforma Satelital	Landsat-8	Sentinel-2
Sensor	OLI	MSI
Extensión de la escena	179 x 185 kilómetros	100x100 kilómetros
Resolución espacial	30 metros	10/ 20 metros
Repetición ciclo	16 días	5 días
Bandas utilizadas	Banda2(Azul:0,49-0,51 μ m) Banda3 (verde:0,53-0,59 μ m) Banda4(rojo:0,64-0,67 μ m) Banda5(NIR:0,85-0,88 μ m) Banda6(SWIR1:1,57-1,65 μ m) Banda7:(SWIR2:2,11-2,29 μ m)	Banda2(Azul:0,46-0,52 μ m) Banda3 (verde:0,54-0,58 μ m) Banda4(rojo:0,65-0,69 μ m) Banda8A(NIR:0,86-0,89 μ m) Banda11(SWIR1:1,57-1,66 μ m)) Banda12:(SWIR2:2,10-2,29 μ m)

Fuente: USGS, 2013; ESA, 2023.

4.3.2 Preprocesamiento de Imágenes

Las imágenes capturadas por los sensores Landsat-8 y Sentinel-2 pueden experimentar distorsiones debido a diversos factores como los propios sensores, los efectos solares, atmosféricos y topográficos. El proceso de preprocesamiento busca reducir al mínimo estos efectos según sea necesario para una aplicación específica (Young et al., 2017).

Todo el procesamiento de datos de imágenes de teledetección que realizamos fue empleando la tecnología de computación en la nube de la plataforma GEE, que almacena productos procesados de Landsat-8 Operational Land Imager (OLI) y Sentinel-2 A/B Multi-Spectral Instrument (MSI). En este estudio utilizamos la colección de escenas de imágenes “LANDSAT/LCO8/CO1/T1_TOA” para los años: 2016, 2018 y 2020 en el caso de Landsat-8 y la colección “COPERNICUS S/2” para las imágenes Sentinel-2. Las seleccionamos basándonos en la disponibilidad de imágenes aéreas de alta resolución para el área de estudio.

En sí el preprocesamiento que realizó Google Earth Engine en los conjuntos de imágenes que utilizamos fue minimizar todos los efectos que abarca la colección como las diferencias entre los sensores remotos (RS), la diferencia entre Landsat OLI, los sensores Landsat anteriores, el efecto de preprocesamiento (por ejemplo, corrección atmosférica) y el efecto de diferentes huellas (Roy & Zhang et al., 2016). La codificación en la nube realiza todo este procesamiento de normalización y corrección.

En consecuencia, el código seleccionó todas las escenas de reflectancia superficial corregidas de los satélites disponibles en la plataforma para los años 2016, 2018 y 2020 en las colecciones “LANDSAT/LCO8/CO1/T1_TOA” y “COPERNICUS S/2”

Nuestra área de estudio está cubierta por la composición de un grupo de escenas para cada año. Cabe mencionar que, en GEE, si los usuarios no seleccionamos específicamente el path y el row de las escenas, o las coordenadas de estas; automáticamente se seleccionará todas las imágenes que se cruzan con el límite del área de estudio. Es por eso que indicamos en la codificación que toda la búsqueda se realice específicamente dentro del recorte del área de estudio.

Cabe destacar también que para el gran problema de nubes y sombras que se suele encontrar en toda escena, utilizamos la función Fmask para reducir y casi en su totalidad anular este factor. Fmask es un algoritmo basado en objetos (función de máscara) el cuál lo aplicamos a las escenas de Sentinel-2 y Landsat-8 seleccionadas. Todo esto con el fin de crear observaciones claras de nubes e imágenes libres de sombras de nubes seleccionando el mejor pixel libre de nubes (Zhu & Woodcock, 2012)

Fmask

Como parte esencial de nuestra metodología, implementamos el algoritmo Fmask para el preprocesamiento de las imágenes provenientes de los satélites Landsat-8 y Sentinel-2. Fmask es una función disponible en Google Earth Engine que nos permite realizar una interpretación visual mejorada de nuestras imágenes. Esta etapa es fundamental debido a las diferencias significativas en la resolución espacial entre ambas plataformas. Mientras que Landsat-8 ofrece una resolución de 30 metros en varias bandas espectrales, Sentinel-2 proporciona una resolución más detallada de 10 metros para las bandas visibles e infrarrojas cercanas (Flood, 2017). A pesar de las capacidades avanzadas de estos satélites, seguimos enfrentando desafíos naturales como la presencia de nubes que pueden afectar la calidad y utilidad de las imágenes (Chuvieco, 2010).

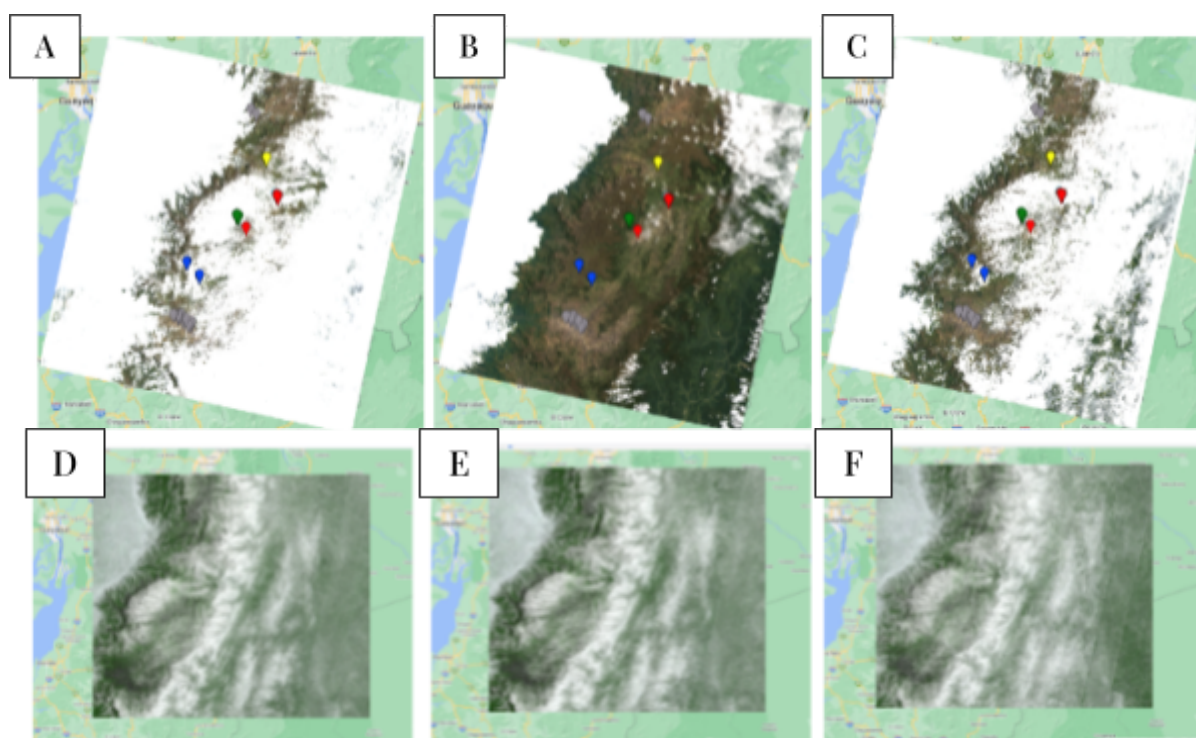
Chuvieco en el 2010, las describe como el obstáculo natural para una definición clara.

Por lo tanto, la codificación del algoritmo Fmask fue esencial para el enmascaramiento de las nubes en cada imagen, lo que evitó obstáculos significativos en la obtención de datos de entrenamiento. Las figuras proporcionadas ilustran claramente el impacto del algoritmo Fmask, mostrando una mejora notable en la claridad de las imágenes después de su aplicación. La *Figura 6* muestra el antes de Fmask en L-8 y S-2 y La *Figura 7* muestra el después de Fmask en L-8 y S-2.

Sin embargo, es importante tener en cuenta que, en algunas circunstancias, como en el caso de Landsat-8-2018 (*B -Figura 6*), la visualización puede ser aceptable sin la aplicación de Fmask, especialmente en áreas con una alta presencia de nubes que no afectan la delimitación del estudio.

Figura 6

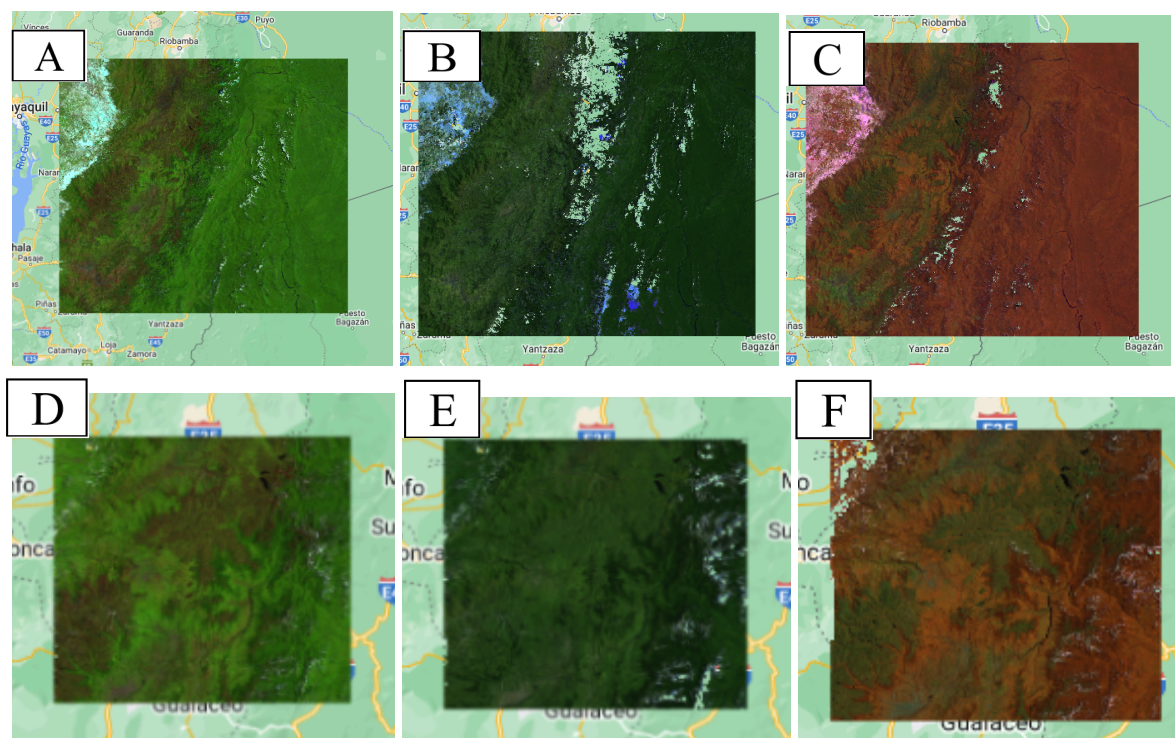
Nubosidad en las imágenes Landsat-8 y Sentinel-2



Nota. (A),(B),(C) Composites de imágenes sin algoritmo de enmascaramiento en los años 2016, 2018 y 2020 respectivamente del satélite Landsat-8 representados en color natural; (D),(E),(F) Composites de imágenes sin algoritmo de enmascaramiento en los años 2016, 2018 y 2020 respectivamente del satélite Sentinel-2, representados en color natural.

Figura 7

Fmask en imágenes Landsat-8 y Sentinel-2



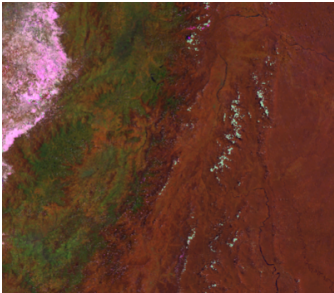
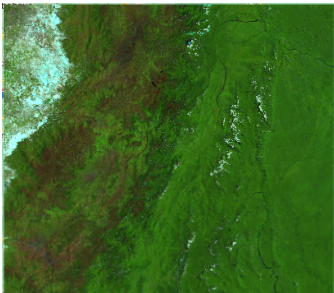
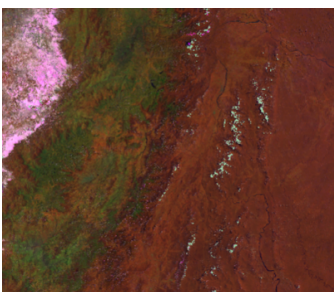
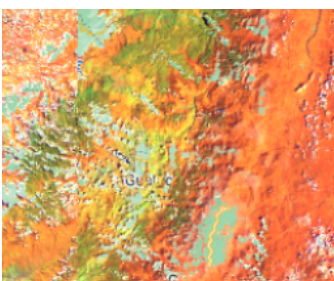
Nota. (A),(B),(C) Composites de imágenes con algoritmo de enmascaramiento en los años 2016, 2018 y 2020 respectivamente del satélite Landsat-8 representados en color natural; (D),(E),(F) Composites de imágenes con algoritmo de enmascaramiento en los años 2016, 2018 y 2020 respectivamente del satélite Sentinel-2, representados en color natural.

4.3.3 Categorías de cobertura terrestre y datos de referencia (Datos de entrenamiento y validación)

El paisaje heterogéneo hizo posible que identificáramos 5 clases de cobertura terrestre en el área de estudio. Todos los datos de entrenamiento y validación que recolectamos están basados en el manual de interpretación visual de imágenes de alta resolución de Google Earth. Este método es ampliamente aplicado y reportado constantemente en la literatura (De Sousa et.al, 2020). Adicional, para fortalecer la recolección de datos, utilizamos como guía el mapa de índices de diferencia normalizada NDVI con la clasificación de cobertura (Figura 8) y la interpretación visual de imágenes con composiciones de color RGB (Tabla 3).

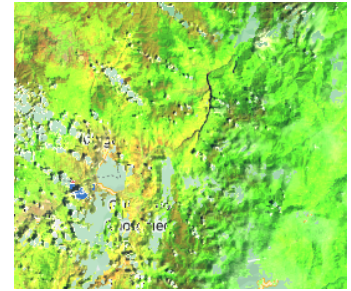
Tabla 3

Interpretación Visual en RGB

Satélite-Año	Capa de Datos	Combinación de bandas (R-G-B)	Imagen Satelital Composite Anual
Landsat-8 2016	"LANDSAT/LCO8/CO1/T1_TOA"	Análisis de Vegetación (6-5-4)	
Landsat-8 2018	"LANDSAT/LCO8/CO1/T1_TOA"	Falso Color para detección de zonas Urbanas (7-6-4)	
Landsat-8 2020	"LANDSAT/LCO8/CO1/T1_TOA"	Análisis de Vegetación (6-5-4)	
Sentinel-2 2016	"COPERNICUS S/2"	Análisis de Vegetación (8A-11-2)	

Sentinel-2
2018 “COPERNICUS S/2”

Usos Agrícolas
(11-8A-2)



Sentinel-2
2020 “COPERNICUS S/2”

Falso Color para
detección de
zonas Urbanas
(12-11-4)

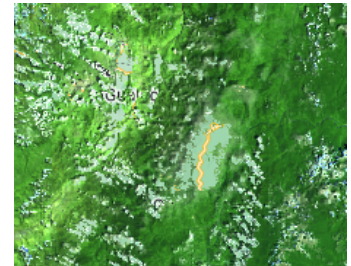
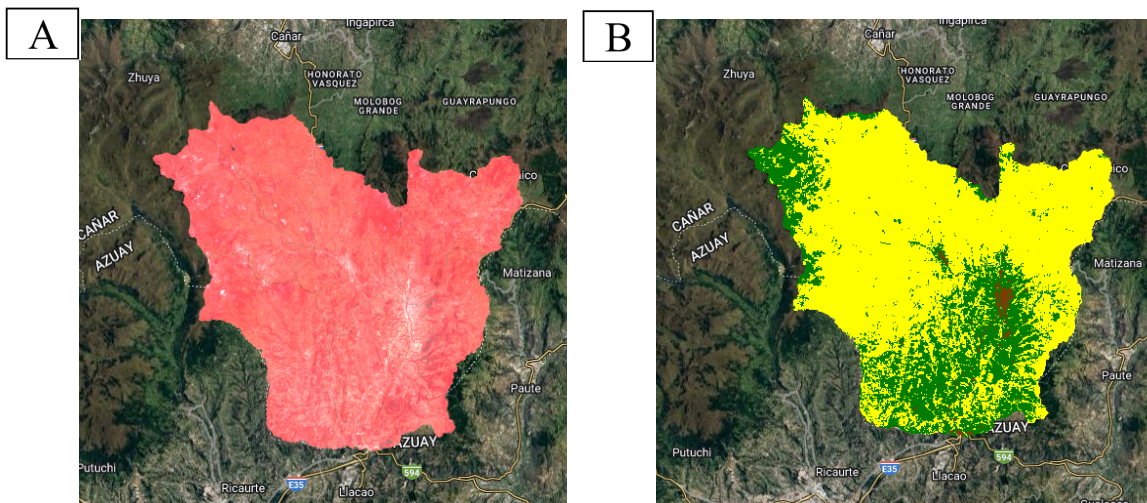


Figura 8

Mapa de índice NDVI

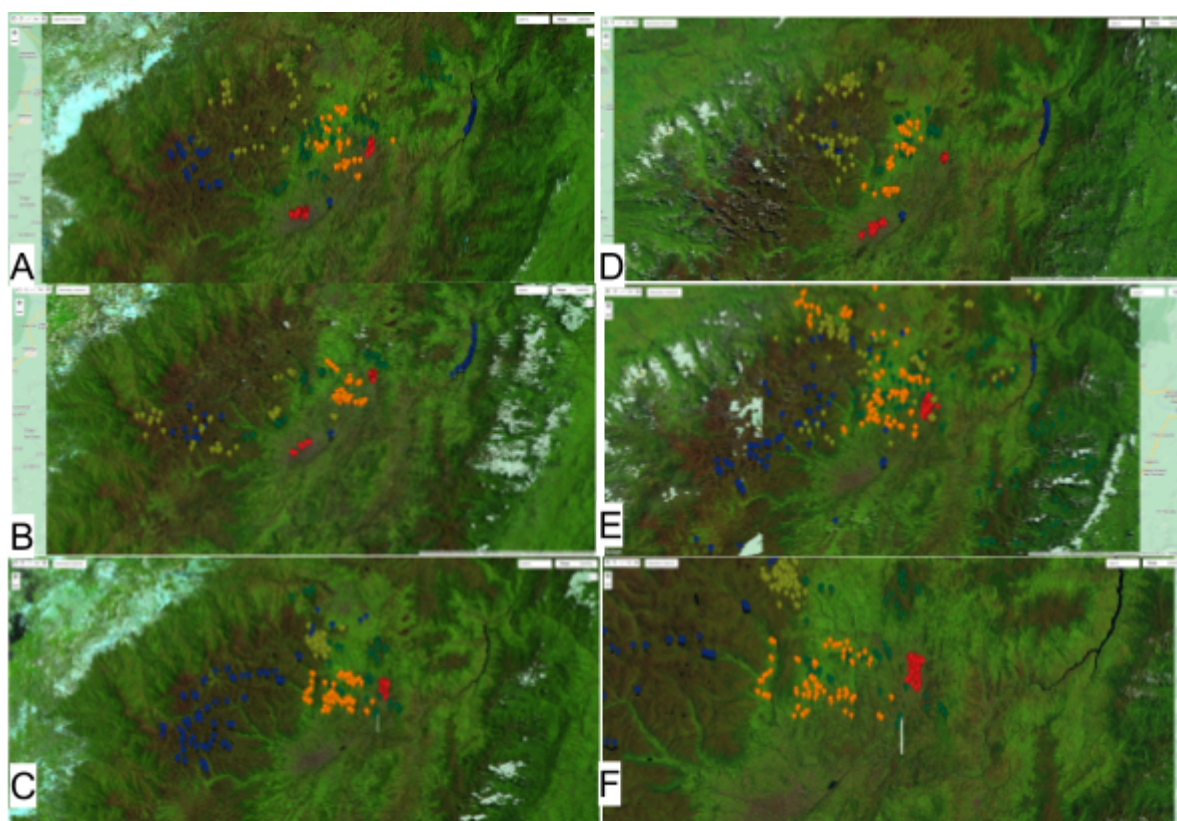


Nota. (A) Cálculo de mediana NDVI para suelo desnudo. (B) Cálculo de índices:NDVI , NDWI y NDBI.

Delineamos 500 puntos de referencia para los años 2016, 2018 y 2020 tanto para Landsat-8, como para Sentinel-2; todo esto lo hicimos respectivamente en función de las características previamente mencionadas para la recolección de datos, con una consideración de no menos de 100 puntos por categoría de cobertura de suelo; la distribución de los datos de referencia se muestra en la *Figura 9* (Congalton, 1988). En conclusión, creamos un conjunto de datos de entrenamiento que consistió en un 70% de observaciones seleccionadas al azar; el 30% restante de las observaciones lo utilizamos para el conjunto de datos de validación para garantizar que no hubiera superposición entre ellas (Rodríguez-Galiano et al., 2012). El conjunto de datos de entrenamiento lo aplicamos para mejorar el algoritmo clasificador supervisado Random Forest, mientras que el conjunto de datos de validación lo usamos en la evaluación de la precisión de los mapas de clasificación de cobertura terrestre producidos.

Figura 9

Distribución de datos de referencia



Nota. (A),(B),(C) Puntos de referencia del año 2016, 2018 y 2020 respectivamente del satélite Landsat-8; (D),(E),(F) Puntos de referencia año 2016, 2018 y 2020 respectivamente del satélite Sentinel-2, representados en color natural.

Tabla 4

Categorías de cobertura de suelo en Google Earth Engine

<i>Clase</i>	<i>Nombre de la Clase (Cobertura)</i>
0	<i>Cuerpos de agua.</i>
1	<i>Suelos descubiertos, centros poblados.</i>
2	<i>Bosque</i>
3	<i>Pastos y Cultivos.</i>
4	<i>Páramos</i>

Se clasificó a “Pastos y Cultivos” en una sola categoría ya que en algunas condiciones y etapas de crecimiento, los pastos y los cultivos pueden tener características espectrales similares en las bandas utilizadas por los sensores remotos como Landsat-8 y Sentinel-2. Esto puede dificultar su distinción en imágenes de teledetección, especialmente en áreas donde ambos tipos de vegetación están presentes de manera intercalada.

Además, los conjuntos de datos de teledetección disponibles tienen una resolución espacial limitada, lo que significa que cada píxel en la imagen representa un área relativamente grande en la superficie terrestre. En áreas donde los cultivos y los pastos están entremezclados, la resolución espacial puede no ser suficiente para distinguir entre estas clases con precisión.

Los algoritmos de clasificación automática pueden tener dificultades para discriminar entre pastos y cultivos debido a la similitud en sus firmas espectrales. En algunos casos, clasificar pastos y cultivos como una sola clase puede simplificar el proceso de clasificación y reducir la complejidad del análisis.

Ya que nuestro estudio tiene como objetivo evaluar el cambio de cobertura de suelo en general, agrupar pastos y cultivos en una sola clase logró ser suficiente para proporcionar información sobre la distribución y la extensión de las áreas agrícolas dentro de la Subcuenca.

4.3.4 Clasificación de imágenes (Random Forest Classifier) y evaluaciones de precisión

Random Forest es un algoritmo que se compone de múltiples árboles de decisión, los cuales trabajan en conjunto, tal como sugiere su nombre. Se ha observado que este enfoque puede ser menos afectado por el ruido y ofrecer una mayor eficiencia en comparación con otros clasificadores no paramétricos ampliamente utilizados (Pelletier et al., 2016). Actualmente es uno de los algoritmos más aplicados para la clasificación de la cobertura terrestre utilizando datos de teledetección (Rodríguez-Galiano et al., 2012). Por estas razones y por las mencionadas en el capítulo de revisión bibliográfica lo elegimos como algoritmo clasificador en nuestro estudio.

Basándonos en las recomendaciones de estudios previos y ensayos previos de nuestros datos; codificamos mediante el lenguaje JavaScript en Google Earth Engine 100 árboles en el algoritmo Random Forest ($n_{tree}=100$), mientras que m_{try} se estableció con el valor predeterminado (raíz cuadrada del número total de características).

La evaluación de precisión en la clasificación supervisada es un proceso para determinar la exactitud de los resultados obtenidos por un algoritmo de clasificación en comparación con las clases de referencia conocidas (Zurqani et al., 2018). La etapa final del análisis de datos de teledetección implica la realización de evaluaciones de precisión utilizando diferentes métricas cuantitativas. Se han seleccionado medidas como la precisión general, el coeficiente Kappa, así como la precisión del productor y del usuario para determinar la precisión global de la clasificación de la cobertura terrestre en cada uno de los años correspondientes a las imágenes de Landsat-8 y Sentinel-2 (2016, 2018, 2020).

Estas son técnicas muy útiles y efectivas para determinar una vez terminada la clasificación, que tan efectivas fuimos como usuarias y si la clasificación cumplió con un correcto desarrollo (Tsutsumida & Comber, 2015). Continuando con los análisis de evaluación de nuestro modelo de clasificación, fue estimada la matriz de confusión para cada una de las clasificaciones la cual contiene las muestras de píxeles clasificados y el recuento total de clasificaciones incorrectas para cada clase. Los mapas de clasificación de cobertura terrestre que producimos los validamos mediante el conjunto de datos de validación usados en el código, los cuales fueron $\frac{1}{3}$ de los datos de referencia. Esta medida fue seleccionada en base a estudios anteriores (Congalton, 1991).

4.3.5 Detección de cambio de cobertura del suelo

Las imágenes de clasificación de cobertura terrestre de las diferentes fechas, y satélites Landsat-8 (2016, 2018, 2020) y Sentinel-2 (2016, 2018, 2020) las destinamos para calcular el área de las cinco diferentes clases de cobertura terrestre y observar los cambios que se produjeron en ese momento. Estas áreas de cobertura las expresamos usando gráficos de barras para las imágenes en cada año, una tabla que expone su extensión en hectáreas y porcentajes, y resume estadísticamente los cambios de las coberturas a lo largo del tiempo.

Finalmente realizamos diagramas de Sankey para visualizar la dinámica de la cobertura terrestre en el área de la subcuenca del río Burgay, el cual mostró los resultados de la clasificación y los cambios y persistencias de coberturas en tres intervalos de tiempo 2016-2018-2020 para Landsat-8 y Sentinel-2.

Adicionalmente también realizamos mapas con mediciones precisas de la dinámica de la cobertura terrestre que comparan estos cambios: Landsat-8 (2016-2020) , Sentinel-2 (2016-2020).

4.3.6 Validación

Es indispensable en la producción de información mediante la clasificación supervisada validar nuestros datos. La validación es crucial para asegurarse de que el modelo es capaz de generalizar bien a datos no vistos y para entender su rendimiento en diferentes aspectos, en otras palabras, mide el rendimiento de un modelo de aprendizaje supervisado (Tsutsumida & Comber,2015).

Suelen haber opciones para realizar este proceso dependiendo el tipo de estudio con imágenes que se esté realizando. En el caso de un estudio en tiempo real con series de imágenes multitemporales, lo ideal sería un estudio in-situ de las áreas que se estén analizando.

En el cambio de cobertura con la media de mosaicos anuales, caso presente de nuestro estudio; una inter validación entre los evaluadores del algoritmo RF con sus respectivas matrices de confusión y la utilización de la validación con imágenes satelitales de alta resolución, son ampliamente halladas en la literatura (Xu et al., 2020).

Los resultados de nuestros datos de clasificación con Landsat-8 y Sentinel-2 fueron validados de las dos maneras: En la plataforma PlanetLAB en los diferentes periodos de tiempo (2016, 2018, 2020) para el análisis visual de los resultados; y mediante matrices de confusión (exactitud, precisión, sensibilidad, especificidad) con métricas de precisión

(precisión global “Overall Accuracy”, precisión del productor, precisión del usuario y coeficientes kappa).

4.3.7 Matriz de Confusión

La matriz de confusión se emplea frecuentemente como un recurso para evaluar la precisión de una clasificación supervisada.

La matriz de confusión es una representación tabular empleada para evaluar cómo se desempeña un algoritmo de clasificación, contrastando sus predicciones con las clases reales de las muestras. La matriz está compuesta por cuatro componentes principales: Verdadero Positivo (VP), Falso Positivo (FP), Verdadero Negativo (VN) y Falso Negativo (FN) (*Tabla 5*) (Bonilla & Taipei, 2021).

Además de estos componentes, en la matriz de confusión también se pueden calcular dos métricas adicionales: la Comisión y la Omisión. La Comisión representa la proporción de muestras que fueron incorrectamente clasificadas como pertenecientes a una clase específica, mientras que la Omisión indica la proporción de muestras reales de una clase que fueron incorrectamente clasificadas como no pertenecientes a esa clase. Estas métricas proporcionan una visión más completa de cómo el algoritmo de clasificación está funcionando en términos de sus predicciones y sus resultados reales (Pons, 2015).

Tabla 5

Principales componentes de la matriz de confusión

	Clase positiva	Clase negativa
	(Predicción)	(Predicción)
Clase real	Verdadero Positivo (VP)	Falso Positivo (FP)
Clase real	Falso Negativo (FN)	Verdadero Negativo (VN)

Fuente: De Ullibbarri & Pita, 1999.

Las métricas derivadas de la matriz de confusión comprenden la exactitud, la precisión, la sensibilidad y la especificidad.

La exactitud, denominada “Accuracy” en inglés, describe cuán cercano está el resultado de una medición al valor verdadero. En términos estadísticos, la exactitud está ligada al sesgo de una estimación. Se representa como la proporción de resultados verdaderos (tanto verdaderos positivos (VP) como verdaderos negativos (VN)) dividido entre el número total de casos examinados (verdaderos positivos, falsos positivos, verdaderos negativos, falsos negativos) (De Ullibarri & Pita, 1999).

La precisión, mencionada anteriormente y empleada en el objetivo 1, denota la consistencia de validación en una clasificación, focalizándose en la dispersión del conjunto de valores obtenidos mediante mediciones repetidas de una magnitud. Una menor dispersión indica una mayor precisión. Se calcula como la proporción de verdaderos positivos dividida entre todos los resultados positivos, incluyendo tanto verdaderos positivos como falsos positivos (De Ullibarri & Pita, 1999).

La sensibilidad y la especificidad representan dos medidas que reflejan la capacidad de nuestro estimador para distinguir entre los casos positivos y negativos. La sensibilidad se representa como la fracción de verdaderos positivos, mientras que la especificidad, es la fracción de verdaderos negativos (De Ullibarri & Pita, 1999).

La Sensibilidad, o Tasa de Verdaderos Positivos (True Positive Rate) también conocida como TP, indica la proporción de casos positivos correctamente identificados por el algoritmo. Por otro lado, la Especificidad, o Tasa de Verdaderos Negativos (True Negative Rate) también conocida como TN, se refiere a los casos negativos clasificados correctamente por el algoritmo. Estas métricas revelan qué tan eficientemente puede el modelo detectar cada clase (De Ullibarri & Pita, 1999).

El coeficiente kappa (*Tabla 7*) es una medida estadística utilizada para evaluar la concordancia entre dos o más observadores o métodos al clasificar datos categóricos. Su valor varía entre -1 y 1, y se considera que un valor de kappa mayor a 0,8 indica un acuerdo muy bueno, mientras que un valor menor a 0,4 indica un acuerdo pobre (De Ullibarri & Pita, 1999).

Tabla 6

Parámetros de precisión de clasificación

Parámetro	Fórmula
Precisión global o porcentaje de acuerdo	$P = \sum_{i=1}^m X_{ii} / N$ (1)
Precisión del productor	$Pp = X_{ii} / X_{i\Sigma}$ (2)
Precisión del usuario	$Pu = X_{ii} / X_{\Sigma i}$ (3)
Coefficiente Kappa	$k = \frac{N \sum_{i=1}^m X_{ii} - \sum_{i=1}^m X_{i\Sigma} X_{\Sigma i}}{N^2 - \sum_{i=1}^m X_{i\Sigma} X_{\Sigma i}}$ (4)

Fuente: Ariza, 2017, p.130.

Donde:

m = número total de clases

N = número total de píxeles en las m clases de referencia

X_{ii} = elementos de la diagonal de la matriz de confusión

$X_{\Sigma i}$ = suma de los píxeles de la clase i de referencia

$X_{i\Sigma}$ = suma de los píxeles clasificados como la clase i

Tabla 7

Escala de valorización del índice Kappa

Valor Índice Kappa	Grado de acuerdo
<0.0	Sin acuerdo
0.0-0.20	Insignificante
0.21-0.40	Discreto
0.41-0.60	Moderado
0.61-0.80	Sustancial
0.81-1.00	Casi perfecto

Fuente: Landis y Koch, 1977.

4.4. Metodología para el cumplimiento del objetivo específico 2

Para el desarrollo de este objetivo fueron utilizadas métricas de evaluación de la clasificación supervisada, debido a que estas características inherentes satelitales difieren de una imagen a otra y de esta manera establecer el análisis del porqué las imágenes Sentinel-2 y Landsat-8 presentaron potencialidades y limitaciones para analizar un mismo lugar. Entre las propiedades satelitales que utilizamos para determinar estas ventajas y desventajas están: La resolución espacial, número de bandas espectrales, frecuencia de captura, revisitación rápida, longitudes de onda, costo y las aplicaciones (*Tabla 8*).

Tabla 8

Diferentes propiedades satelitales que presentan los sensores L-8 y S-2; seleccionadas para establecer ventajas y limitaciones entre los satélites

Características de sensores satelitales Landsat-8 y Sentinel-2

Resolución espacial	Hace referencia a la capacidad del sensor para distinguir detalles en la superficie terrestre (Jensen & John R. 2007).
Número de bandas espectrales	Indica la cantidad de rangos espectrales que el sensor pueda capturar (Campbell et al; 2013).
Frecuencia de captura	Se refiere a la regularidad con la que el sensor adquiere nuevas imágenes de una ubicación en la Tierra (Xu et al; 2012).
Revisitación rápida	Indica la capacidad del sensor para volver a capturar una misma área en un corto periodo de tiempo (Schwarz & Roy, 2019).
Longitudes de onda	Son los rangos específicos del espectro electromagnético que el sensor puede detectar (Avogadro & Padró, 2019).
Costo	Indica el precio asociado con la adquisición de datos del sensor (Foody & Giles, 2010).
Aplicaciones	Se refiere a los diferentes usos y aplicaciones que se pueden realizar utilizando los datos del sensor (Gumma et al; 2011).

Nota. Estas propiedades satelitales fueron consideradas como las más relevantes que presentan los sensores L-8 y S-2.

4.5. Metodología para el cumplimiento del objetivo específico 3

En el desarrollo de la metodología para lograr el tercer objetivo, que consiste en crear un tutorial de teledetección para el análisis de cobertura, hemos organizado el tutorial de manera dinámica en 8 partes. Cada capítulo está estructurado con procedimientos detallados y específicos, acompañados de capturas de pantalla numeradas que ilustran paso a paso lo que se debe hacer para llevar a cabo una clasificación de cobertura en Google Earth Engine utilizando los lenguajes de programación JavaScript y Python. Este enfoque permite que el lector utilice el tutorial como un documento independiente o en conjunto con la tesis, utilizando la misma codificación principal que hemos empleado en nuestro proyecto.

5. Resultados y discusión

Los resultados se presentan divididos en tres partes de acuerdo a nuestros objetivos planteados. La primera parte corresponde a la identificación y cuantificación de las coberturas más representativas en la subcuenca y su respectiva dinámica de cambio. En la segunda parte, diferentes características en la clasificación supervisada utilizada por los sensores, los presentamos con el fin de definir pruebas y representar de manera comparativa estas ventajas y desventajas o en otras palabras estas potencialidades y limitaciones que poseen las imágenes Landsat-8 y Sentinel-2 para el estudio de la cobertura del suelo. Y por último, la tercera parte es el tutorial producido para el análisis de cambio de cobertura del suelo en cuencas andinas usando imágenes satelitales.

5.1 Resultados del objetivo número 1

En esta sección los resultados responden al primer objetivo: Identificación y cuantificación de las coberturas más representativas en la subcuenca y su respectiva dinámica de cambio; iniciando por un recuento cuantitativo de cada cobertura del suelo y su distribución en la subcuenca para cada satélite en los respectivos años 2016, 2018 y 2020, seguido de una dinámica de cambio representada por diagramas Sankey y Land Use Change.

5.1.1 Identificación y cuantificación de Coberturas

Al aplicar el clasificador Random Forest a los datos de los sensores Landsat-8 y Sentinel-2 en Google Earth Engine utilizando sus bandas de reflectancia de superficie, se identificaron las siguientes coberturas:

- 1) Cuerpos de agua
- 2) Suelos descubiertos y centros poblados
- 3) Bosque
- 4) Pastos y Cultivos
- 5) Páramos.

Iniciando con Landsat-8, las estimaciones fueron generadas en base a la extensión de cobertura de suelo clasificada *Tabla 9*.

Adicionalmente se representa la distribución de estas coberturas identificadas en diagramas de barras (*Figura 10*), que expresa por grupo la extensión en cada fecha temporal; al verlas se puede apreciar que las coberturas que mayor extensión ocuparon fueron pastos y cultivos en el 2018, seguido de páramos en el 2020, bosques en el 2016, suelos y centros poblados en el 2018 conjuntamente con los cuerpos de agua.

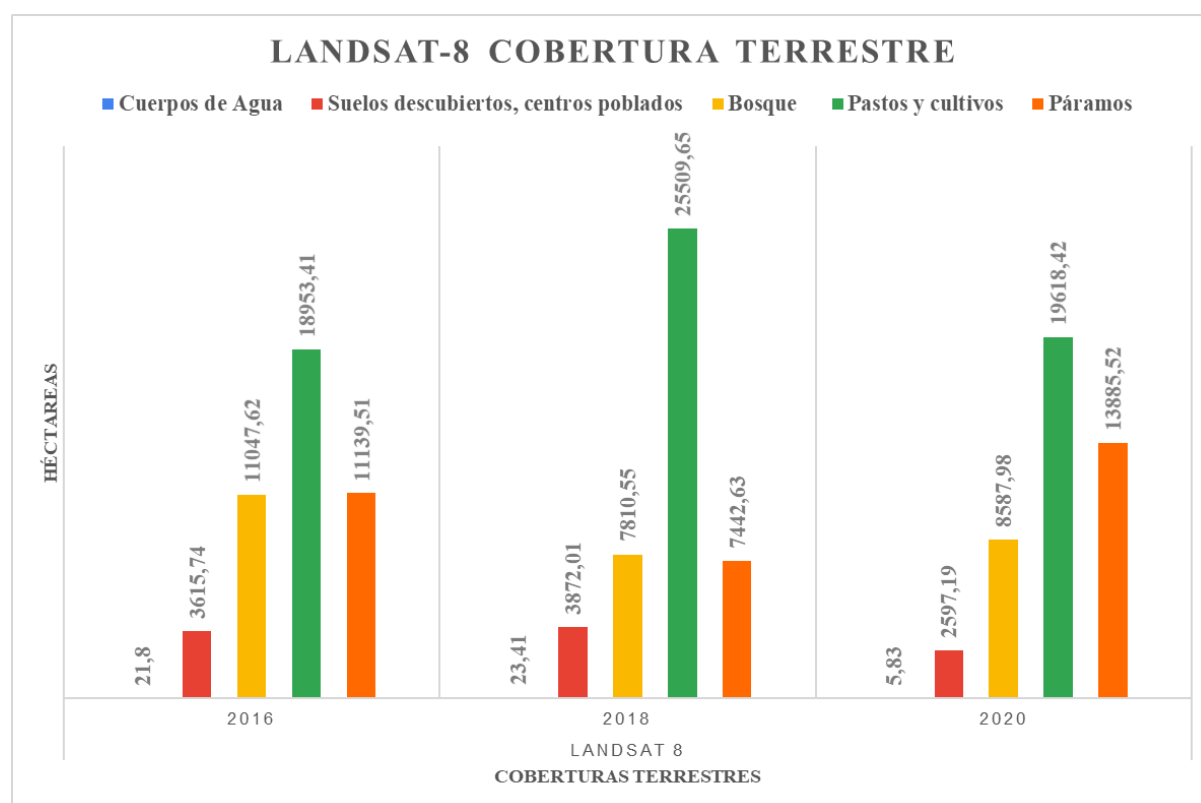
Tabla 9

Identificación y cuantificación de cobertura del suelo en la subcuenca del río Burgay, satélite Landsat-8 en los años 2016, 2018 y 2020 (ha, %)

Coberturas	2016	%	2018	%	2020	%
Cuerpos de Agua	21,8	0,04	23,41	0,05	5,83	0,02
Suelos descubiertos, centros poblados	3615,74	8,07	3872,01	8,67	2597,19	5,81
Bosque	11047,62	24,37	7810,55	17,48	8587,98	19,21
Pastos y cultivos	18953,41	42,32	25509,65	57,12	19618,42	43,89
Páramos	11139,51	24,87	7442,63	16,66	13885,52	31,06

Figura 10

Distribución de coberturas de suelo Landsat-8, en el transcurso de 2016, 2018, 2020



En el caso de Sentinel-2 las estimaciones fueron generadas con el mismo método; La distribución de estas coberturas identificadas se encuentran en el diagramas de barras (*Tabla 10*), que expresa por grupo la extensión en cada fecha temporal; al verlas se puede apreciar que las coberturas que mayor extensión ocupan fueron pastos y cultivos en el 2016, 2020 y 2018 en ese orden, seguido de páramos en el 2016, bosques en el 2020, suelos y centros poblados en el 2018 y cuerpos de agua en el 2016 (*Figura 11*).

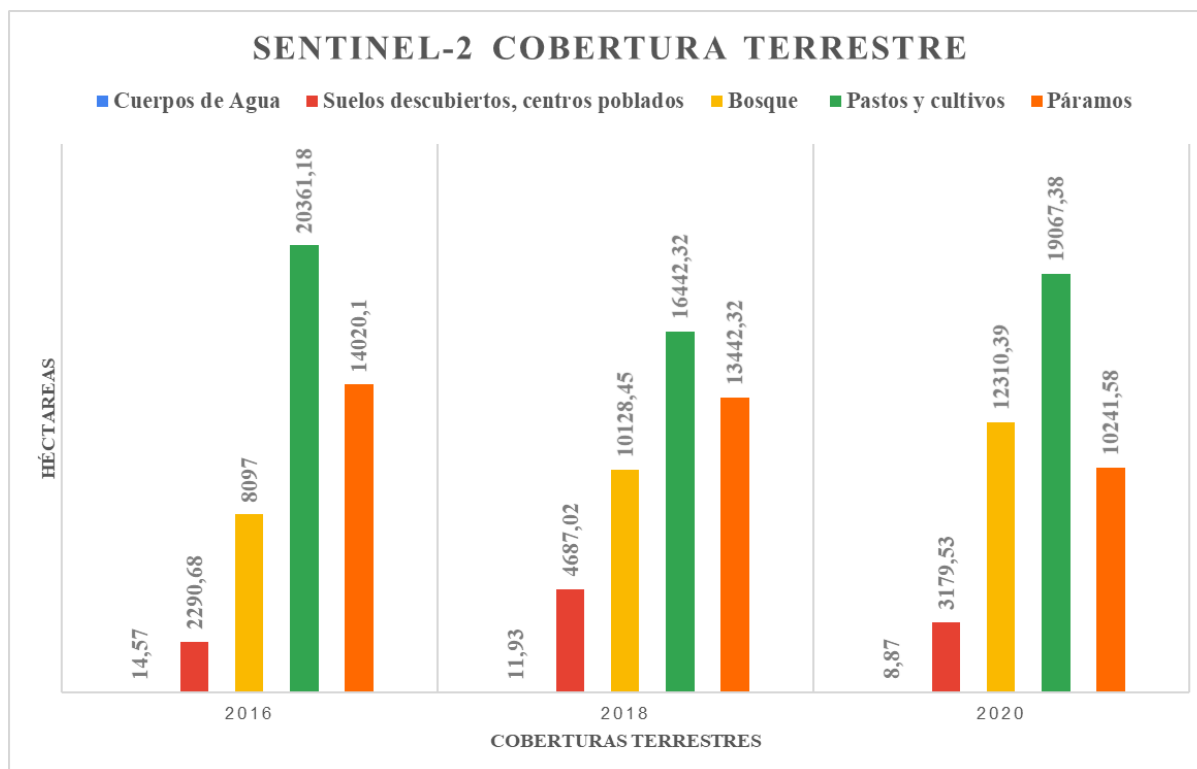
Tabla 10

Identificación y cuantificación de cobertura del suelo en la subcuenca del río Burgay, satélite Sentinel-2 en los años 2016, 2018 y 2020 (ha, %)

Coberturas	2016	%	2018	%	2020	%
Cuerpos de Agua	14,57	0,03	11,93	0,02	8,87	0,02
Suelos descubiertos, centros poblados	2290,68	5,11	4687,02	10,47	3179,53	7,09
Bosque	8097	18,08	10128,45	22,64	12310,39	27,47
Pastos y cultivos	20361,18	45,46	16442,32	36,75	19067,38	42,55
Páramos	14020,1	31,3	13442,32	30,09	10241,58	22,85

Figura 11

Distribución de coberturas de suelo Sentinel-2, en el transcurso de 2016, 2018 y 2020



Landsat-8 y Sentinel-2 lograron resultados favorables para la identificación de coberturas terrestres, más no iguales, estas variaciones entre sensores se evidenciaron tras la clasificación. Siendo en los tres periodos de tiempo la identificación de superficie de cobertura superior en Sentinel-2 en comparación con Landsat-8, con la única excepción del año 2018 en “Pastos y Cultivos” única cobertura analizada por el satélite Landsat-8 que logró superioridad sobre Sentinel-2.

Xu Feng en el 2020 menciona que la identificación de las coberturas con el satélite Sentinel-2 en su estudio, es más eficiente debido a la capacidad de Sentinel-2 para discriminar sutilezas en la cobertura del suelo, especialmente en áreas agrícolas y boscosas, permitiendo un análisis más detallado y preciso de los cambios. Según Jara y Delegido (2019) su frecuencia de revisita mejorada contribuye a capturar eventos rápidos, como cambios en cultivos, esta característica nos facilitó la observación de los páramos. Esto concuerda con nuestros resultados debido a que la identificación de coberturas con este sensor fue más rápida y sencilla en estas coberturas, la razón principal; una combinación de características que en conjunto hicieron posible que nuestro análisis fuera

más simple. No obstante, los resultados con el sensor Landsat-8 también se mostraron factibles según nuestros resultados presentados.

5.1.2 Análisis de clasificaciones supervisadas en la cobertura del suelo

En esta sección, reportamos los resultados de las clasificaciones supervisadas de la cobertura de suelo, con su respectiva validación y análisis de precisión, en tres momentos temporales: 2016, 2018 y 2020. Lo cual corresponde al objetivo 1.

Comparación de la imagen Landsat-8 versus la imagen Sentinel-2 del año 2016

Los resultados de la clasificación de la imagen Landsat-8 vs la imagen Sentinel-2 de la Subcuenca del río Burgay en el año 2016 evidencian una confiabilidad del 91% en adelante. Esta confiabilidad se obtuvo al analizar las pruebas de precisión en nuestras clasificaciones. Según Landis y Koch (1977) que exponen la interpretación del coeficiente kappa adaptado a esta investigación; determinamos que los resultados de las dos clasificaciones L-8 y S-2 presentan un grado de acuerdo Kappa casi perfecto.

La *Tabla 11* muestra el valor del coeficiente Kappa para Landsat-8 de 0,94 en comparación con Sentinel-2 (*Tabla 12*) que obtuvo el valor de 0,99. Y una precisión general (Overall Accuracy) de 0,95% L-8 y de 0,99% S-2. Todo esto expuesto en nuestra Matriz de confusión *Tabla 11* en donde las filas indican la fiabilidad de los píxeles de entrenamiento digitalizados por nosotras las usuarias, y las columnas muestran el acierto de la clasificación de los píxeles por clases (Saldaña, 2019).

En las filas de la *Tabla 11* (Landsat-8, 2016) la clase cuerpos de agua obtuvo 33 píxeles de entrenamiento digitalizados, obteniendo el 100% en la exactitud del usuario y del productor. De igual manera los suelos descubiertos conjuntamente con los centros poblados obtuvieron 100% en la exactitud del usuario y del productor en los 34 píxeles de entrenamiento digitalizados.

Sin embargo, estas altas tasas de precisión no se replicaron en todas las clases. Por ejemplo, la clase Bosque experimentó una confusión con los píxeles de páramo, lo que resultó en una caída de la exactitud de clasificación al 95,65%. Asimismo, la clase Pastos y cultivos mostró una confusión con los píxeles de bosque, obteniendo una exactitud del 92,59%. Por último, la clase páramo fue la que más errores experimentó en su clasificación, con píxeles confundidos principalmente con los de la clase Bosque, disminuyendo su exactitud al 87,88%.

Estos resultados obtenidos en nuestro estudio, sugieren similitud con los de Pabón (2022) en el cuál al analizar 5 diferentes tipos de coberturas en la subcuenca del río Tabacay (bosque, páramo, tierra agropecuaria, vegetación arbustiva y zona antrópica), presentaron discrepancia de píxeles para la identificación de páramo y bosque. La *Figura 12*, presenta el mapa resultante de la cobertura terrestre, proporciona una visualización clara de la clasificación supervisada y la distribución de las clases a lo largo de 2016 para Landsat-8, sirviendo como un recurso valioso para la interpretación de resultados y la identificación de áreas de mejora.

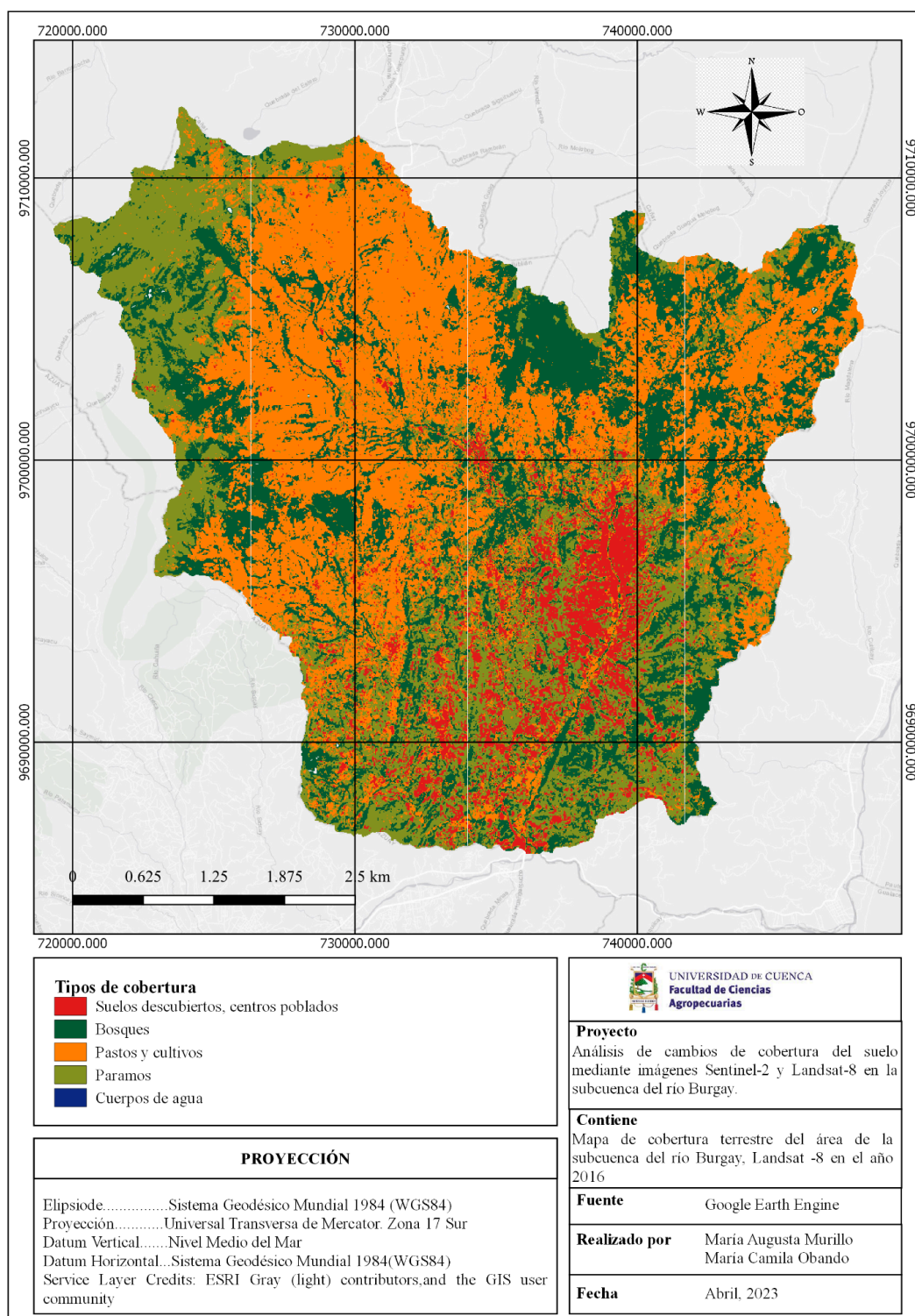
Tabla 11

Matriz de confusión del sensor Landsat-8 en la subcuenca del río Burgay del año 2016

Satélite Landsat 8		Matriz de confusión							Exactitud del usuario (%)
Año 2016		Cuerpos de agua.	Suelos descubiertos, centros poblados.	Bosque y Vegetación leñosa Nativa.	Pastos y Cultivos	Páramos	Total	Comisión	
Clase Actual	Cuerpos de agua.	33	0	0	0	0	33	0	100
	Suelos descubiertos, centros poblados	0	34	0	0	0	34	0	100,00
	Bosque	0	0	22	0	1	23	1	95,65
	Pastos y Cultivos.	0	0	2	25	0	27	2	92,59
	Páramos.	0	0	4	0	29	33	4	87,88
	Total	33	34	28	25	30	150		
Omisión		0	0	6	0	1			
Exactitud del productor (%)		100	100,00	78,57	100,00	96,67			
Índices globales									
Overall Acuraccy				0.953					
Kappa				0.941					

Figura 12

Mapa de cobertura terrestre del área de la subcuenca del río Burgay, Landsat-8 en el año 2016



Sentinel-2 presentó un resultado del coeficiente kappa casi perfecto 0,99 de confiabilidad según la interpretación de Landis y Koch (1977). Sus errores fueron casi mínimos, es decir la confusión entre sus píxeles entre una clase y otra fueron inferiores a los clasificados con Landsat-8. En las filas de la *Tabla 12*, cuerpos de agua expone 25 píxeles de entrenamiento digitalizados, obteniendo el 100% de fiabilidad del usuario y el 100% del productor, de igual manera resultó para las clases de bosques, pastos y cultivos y páramos. El error en esta clasificación (Sentinel-2, 2016) se encontró en la clasificación de los píxeles de suelos descubiertos, centros poblados, en esta clase los píxeles de esta categoría fueron confundidos con los píxeles de páramos, bajando su exactitud del usuario a un 96,7%. La *Figura 13* presenta el mapa resultante de la cobertura terrestre, esto es nuestra clasificación supervisada y la distribución de las clases a lo largo del año 2016 para el satélite Sentinel-2.

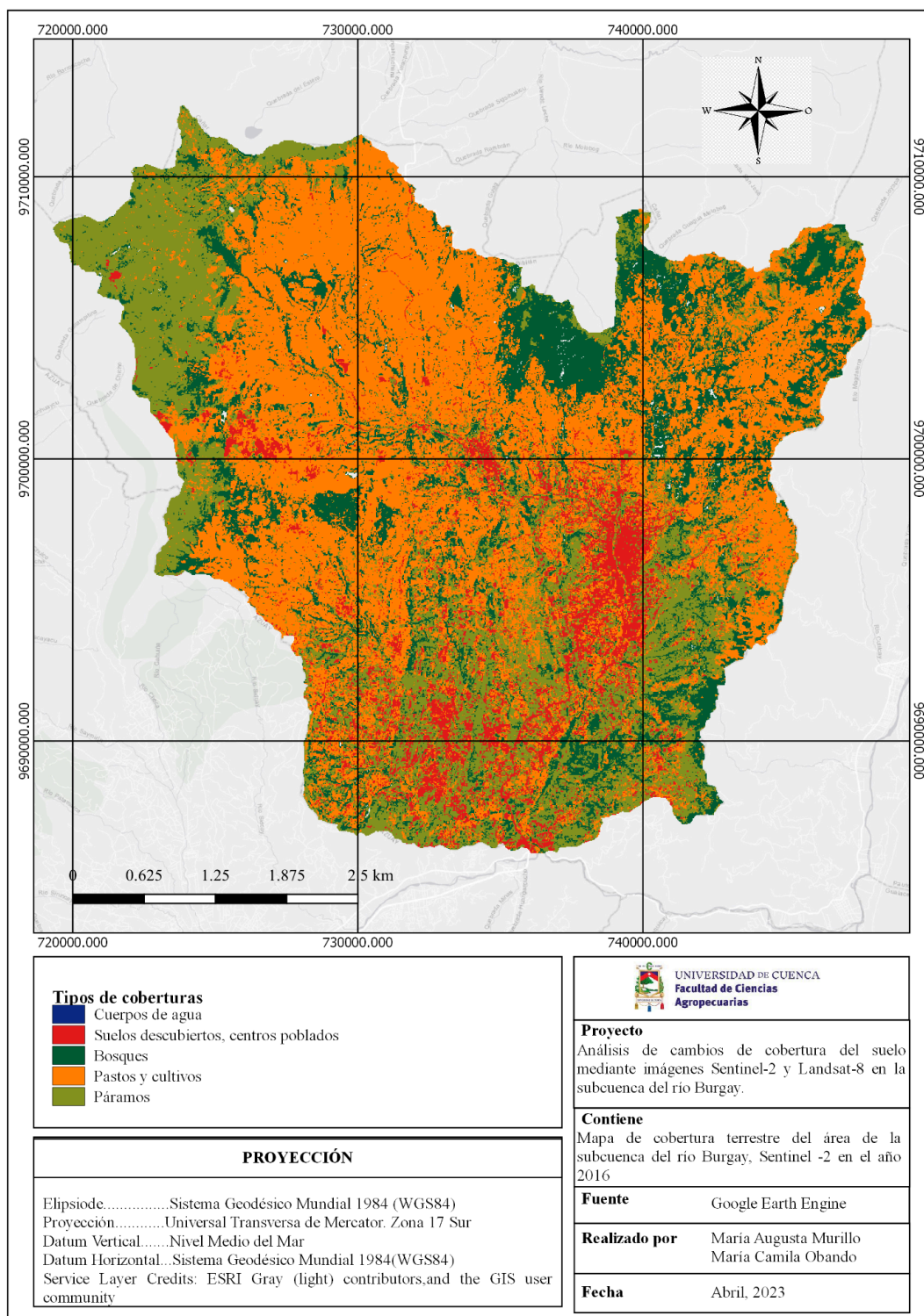
Tabla 12

Matriz de confusión del sensor Sentinel-2 en la subcuenca del río Burgay del año 2016

Satélite Sentinel 2		Matriz de confusión							
Año 2016		Cuerpos de agua.	Suelos descubiertos, centros poblados.	Bosque y Vegetación leñosa Nativa.	Pastos y Cultivos	Páramos	Total	Comisión	Exactitud del usuario (%)
Clase Actual	Cuerpos de agua.	25	0	0	0	0	25	0	100
	Suelos descubiertos, centros poblados	0	29	0	0	1	30	1	96,7
	Bosque	0	0	31	0	0	31	0	100,00
	Pastos y Cultivos.	0	0	0	24	0	24	0	100,00
	Páramos.	0	0	0	0	30	30	0	100,00
	Total	25	29	31	24	31	140		
Omisión		0	0	0	0	1			
Exactitud del productor (%)		100,00	100,00	100,00	100,00	96,77			
Índices globales									
Overall Acuraccy				0.992					
Kappa				0.991					

Figura 13

Mapa de cobertura terrestre del área de la subcuenca del río Burgay, Sentinel-2 en el año 2016



Comparación de la imagen Landsat-8 versus la imagen Sentinel-2 del año 2018

En el año 2018 tanto los sensores Landsat-8 como Sentinel-2 presentaron una confiabilidad del 92% en adelante; Interpretación de casi perfecto (Landis y Koch 1977). Landsat-8, 2018 con una precisión general de 0,96 (Overall Accuracy) y un índice Kappa de 0,95, manifestó en sus filas (Tabla 13) 29 píxeles en la clase cuerpos de agua y 24 píxeles en la clase suelos descubiertos y centros poblados; obtuvieron el 100% de fiabilidad del usuario. En cuanto a las clases que presentaron errores en su clasificación páramos presentó el mayor valor, confundió parte de sus píxeles con los de la cobertura de pastos y cultivos, por lo cual obtuvo una exactitud del 92,59.

Tabla 13

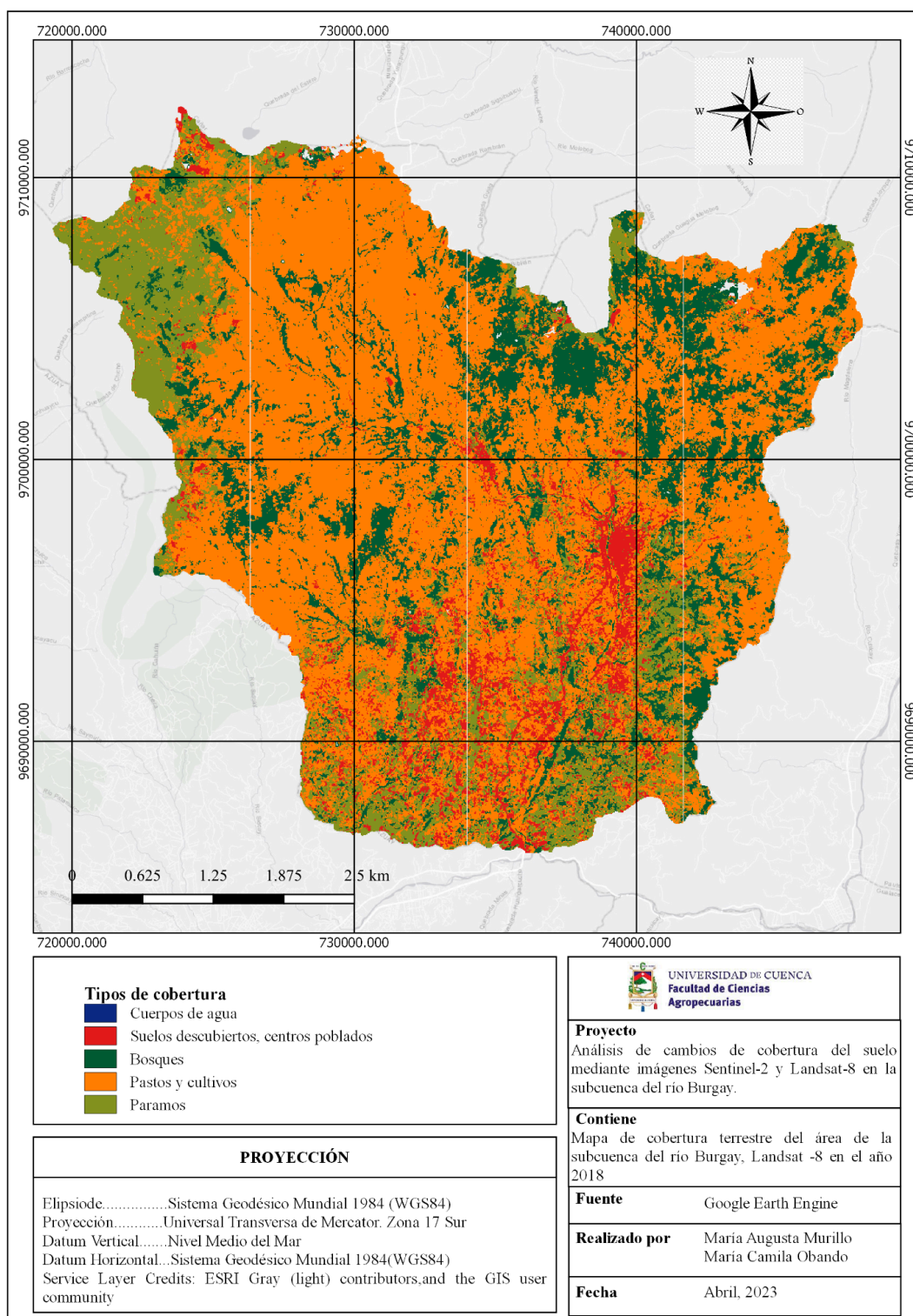
Matriz de confusión del sensor Landsat-8 en la subcuenca del río Burgay del año 2018

Satélite Landsat 8		Matriz de confusión							
Año 2018		Cuerpos de agua.	Suelos descubiertos, centros poblados.	Bosque y Vegetación leñosa Nativa.	Pastos y Cultivos	Páramos	Total	Comisión	Exactitud del usuario (%)
Clase Actual	Cuerpos de agua.	29	0	0	0	0	29	0	100
	Suelos descubiertos, centros poblados	0	24	0	0	0	24	0	100,00
	Bosque	0	0	39	1	0	40	1	97,50
	Pastos y Cultivos.	0	1	1	32	0	34	2	94,12
	Páramos.	0	0	0	2	25	27	2	92,59
	Total	29	25	40	35	25	154		
Omisión		0	1	1	3	0			
Exactitud del productor (%)		100	96,00	97,50	91,43	100,00			
Índices globales									
Overall Accuracy				0.967					
Kappa				0.959					

La clase pastos y cultivos tuvo píxeles clasificados en bosques y suelos descubiertos, obteniendo una exactitud del 94,12% y finalmente los bosques confundieron píxeles de su clasificación con los de pastos y cultivos exhibiendo una exactitud del 97,50%. El mapa de cobertura terrestre (*Figura 14*) muestra el resultado de nuestra clasificación supervisada y la distribución de las clases a lo largo del año 2018.

Figura 14

Mapa de cobertura terrestre del área de la subcuenca del río Burgay, Landsat-8 en el año 2018



El rendimiento casi perfecto del coeficiente kappa de 0,93 en 2018, presentado por Sentinel-2, es un testimonio de la eficacia de este satélite para la clasificación precisa de coberturas terrestres, respaldado por la interpretación de Landis y Koch (1977). No obstante, la distribución de errores entre las cinco clases resalta la complejidad inherente a la tarea de clasificación. Cada categoría de cobertura experimentó la clasificación errónea de píxeles con otras coberturas subrayando la importancia de reconocer las limitaciones y desafíos específicos asociados con el uso de Sentinel-2 en ese período. Las filas detalladas en la *Tabla 14* ofrecen una visión más específica de la distribución de errores en la clasificación.

Tabla 14

Matriz de confusión del sensor Sentinel-2 en la subcuenca del río Burgay del año 2018

Satélite Sentinel 2		Matriz de confusión							
Año 2018		Cuerpos de agua.	Suelos descubiertos, centros poblados.	Bosque y Vegetación leñosa Nativa.	Pastos y Cultivos	Páramos	Total	Comisión	Exactitud del usuario (%)
Clase Actual	Cuerpos de agua.	16	0	1	0	0	17	1	94,12
	Suelos descubiertos, centros poblados	0	27	0	0	1	28	1	96,43
	Bosque	0	0	22	1	0	23	1	95,65
	Pastos y Cultivos.	0	0	0	29	1	30	1	96,67
	Páramos.	0	4	0	0	24	28	4	85,71
Total		16	31	23	30	26	126		
Omisión		0	4	1	1	2			
Exactitud del productor (%)		100,00	87,10	95,65	96,67	92,31			
Índices globales									
Overall Acuraccy				0.936					
Kappa				0.919					

La clase agua, con 16 píxeles de entrenamiento completamente digitalizados, exhibió una confusión con la categoría de bosques, alcanzando una fiabilidad del usuario del 94% y una precisión del productor del 100%. Los suelos descubiertos y centros poblados presentaron píxeles dispersos en la cobertura de páramos, con una exactitud del usuario del 96,43% y una exactitud del productor del 87,10%. Asimismo, los bosques se confundieron con las coberturas de pastos y cultivos, demostrando una exactitud del usuario y del productor del 95,65%. Por otro lado, los pastos y cultivos se confundieron con los páramos, con una exactitud del usuario y del productor del 96,67%.

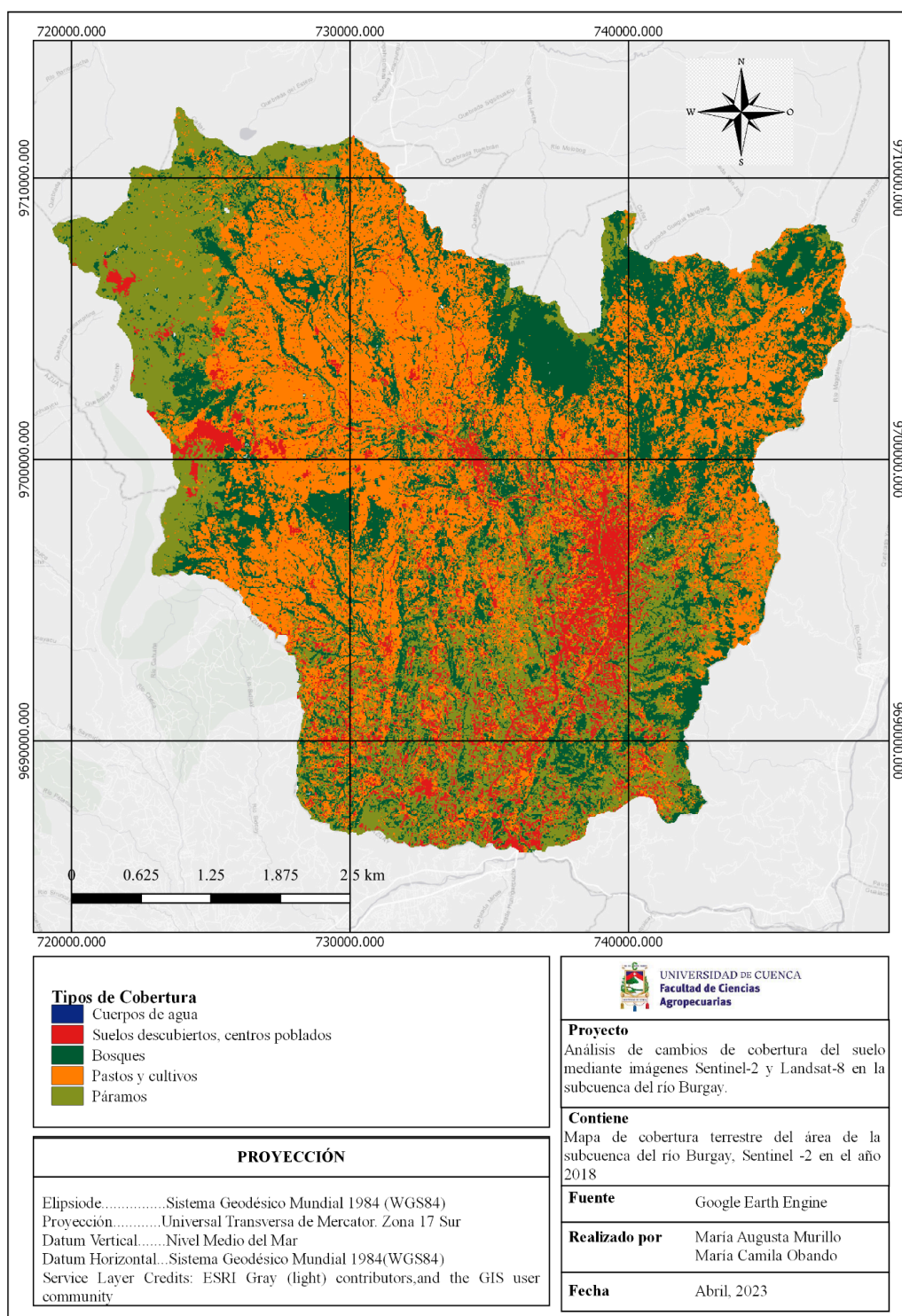
El mayor error en la clasificación de Sentinel-2 en 2018 se evidenció en la categoría de páramos, donde los píxeles fueron confundidos con suelos descubiertos y centros poblados, disminuyendo la exactitud del usuario a un 87,71%, y la exactitud del productor a un 92,31%, lo que sugiere posibles desafíos específicos en la diferenciación de estas clases particulares (Kelly-Fair et al; 2022).

Estos hallazgos subrayan la necesidad de considerar no sólo las métricas globales, sino también las peculiaridades en la clasificación de categorías específicas.

La *Figura 15*, presenta el mapa resultante de la cobertura terrestre, proporciona una visualización clara de la clasificación supervisada y la distribución de las clases a lo largo de 2018 para Sentinel-2.

Figura 15

Mapa de cobertura terrestre del área de la subcuenca del río Burgay, Sentinel-2 en el año 2018



Comparación de la imagen Landsat-8 versus la imagen Sentinel-2 del año 2020

Los resultados de clasificación de cobertura terrestre para los satélites Landsat-8 y Sentinel-2 en el año 2020 muestran una alta precisión general de clasificación cercana al 98,6%, lo que según los criterios de Landis y Koch (1977) se considera una interpretación casi perfecta.

Para Landsat-8 (*Tabla 15*), se observa una alta precisión general (Overall Accuracy) del 98,62% y un coeficiente "Kappa" de 0,9826. La clase de "Cuerpos de agua" tiene una exactitud del 94% con una comisión del 2% lo que indica que hubo una pequeña cantidad de falsos positivos. La clase "Bosque y Vegetación leñosa Nativa" presentó una exactitud del 92,31%. A pesar de la alta precisión de Landsat-8, este sensor tiene limitaciones significativas en términos de resolución espacial y frecuencia de revisita. Con una resolución espacial de 30 metros y un tiempo de revisita de 16 días, Landsat-8 no proporciona la capacidad necesaria para detectar cambios rápidos en los cuerpos de agua o eventos cortos pero significativos, como las proliferaciones de algas (Toming et al., 2016). Esta limitación en la resolución espacial puede resultar en la confusión de píxeles entre cuerpos de agua y bosques, especialmente en áreas donde ambos tipos de cobertura están presentes en proximidad. La baja resolución puede hacer que sea difícil distinguir los bordes de los cuerpos de agua, lo que puede llevar a errores en la clasificación de la cobertura terrestre (Sandoval et al., 2020).

La precisión del 100% en la clasificación de clases de cobertura terrestre, como "Suelos descubiertos, centros poblados", "Pastos y Cultivos", y "Páramos", se atribuye a varios factores interrelacionados respaldados por la literatura científica. Según autores como Gaitán et al., 2018, la combinación de la resolución espacial relativamente alta de Landsat-8 y la calidad radiométrica y geométrica mejorada de sus imágenes contribuye significativamente a la precisión en la clasificación de cobertura terrestre.

La *Figura 16*, presenta el mapa resultante de la cobertura terrestre, proporciona una visualización clara de la clasificación supervisada y la distribución de las clases a lo largo de 2020 para Landsat-8.

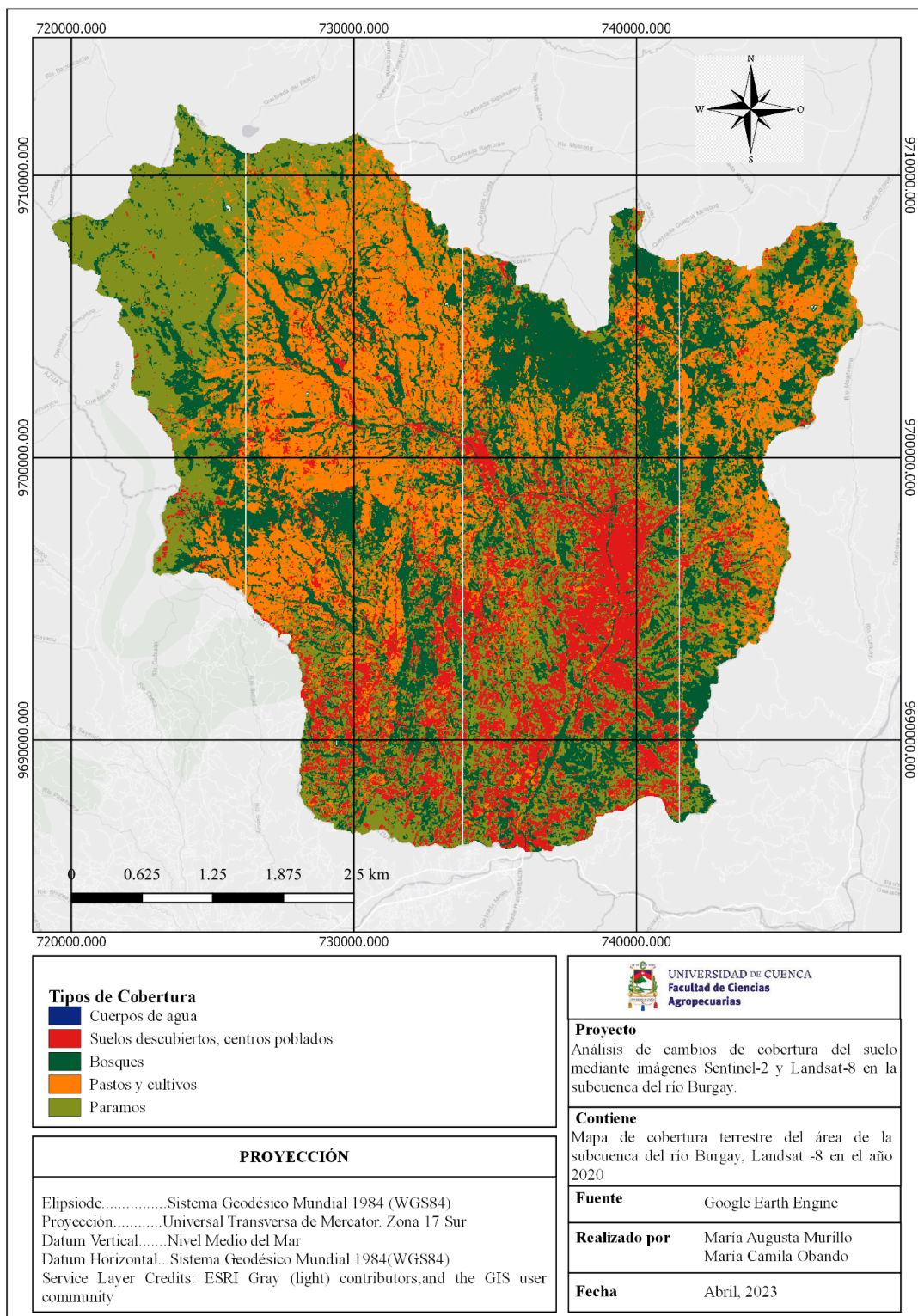
Tabla 15

Matriz de confusión del sensor Landsat-8 en la subcuenca del río Burgay del año 2020

Satélite Landsat 8		Matriz de confusión							Exactitud del usuario (%)
Año 2020		Cuerpos de agua.	Suelos descubiertos, centros poblados.	Bosque y Vegetación leñosa Nativa.	Pastos y Cultivos	Páramos	Total	Comisión	
Clase Actual	Cuerpos de agua.	34	0	2	0	0	36	2	94
	Suelos descubiertos, centros poblados	0	30	0	0	0	30	0	100,00
	Bosque	0	0	24	0	0	24	0	100,00
	Pastos y Cultivos.	0	0	0	33	0	33	0	100,00
	Páramos.	0	0	0	0	22	22	0	100,00
	Total	34	30	26	33	22	145		
Omisión		0	0	2	0	0			
Exactitud del productor (%)		100	100,00	92,31	100,00	100,00			
Índices globales									
Overall Acuraccy				0.986					
Kappa				0.982					

Figura 16

Mapa de cobertura terrestre del área de la subcuenca del río Burgay, Landsat-8 en el año 2020



Basado en la matriz de confusión proporcionada para Sentinel 2 en el año 2020, se puede observar que todas las clases, excepto "Pastos y Cultivos", muestran una precisión del 100% (Tabla 16).

La clase "Pastos y Cultivos" presenta una precisión ligeramente menor, con una exactitud del 93,75%. Los pastos y cultivos pueden variar en términos de su estado de crecimiento, tipo de cultivo, prácticas agrícolas y manejo del suelo, lo que puede generar variabilidad espectral dentro de esta clase y dificultar su distinción clara de otras clases, como los bosques. La Figura 17 presenta el mapa resultante de la cobertura terrestre, esto es nuestra clasificación supervisada y la distribución de las clases a lo largo del año 2020 para el satélite Sentinel-2.

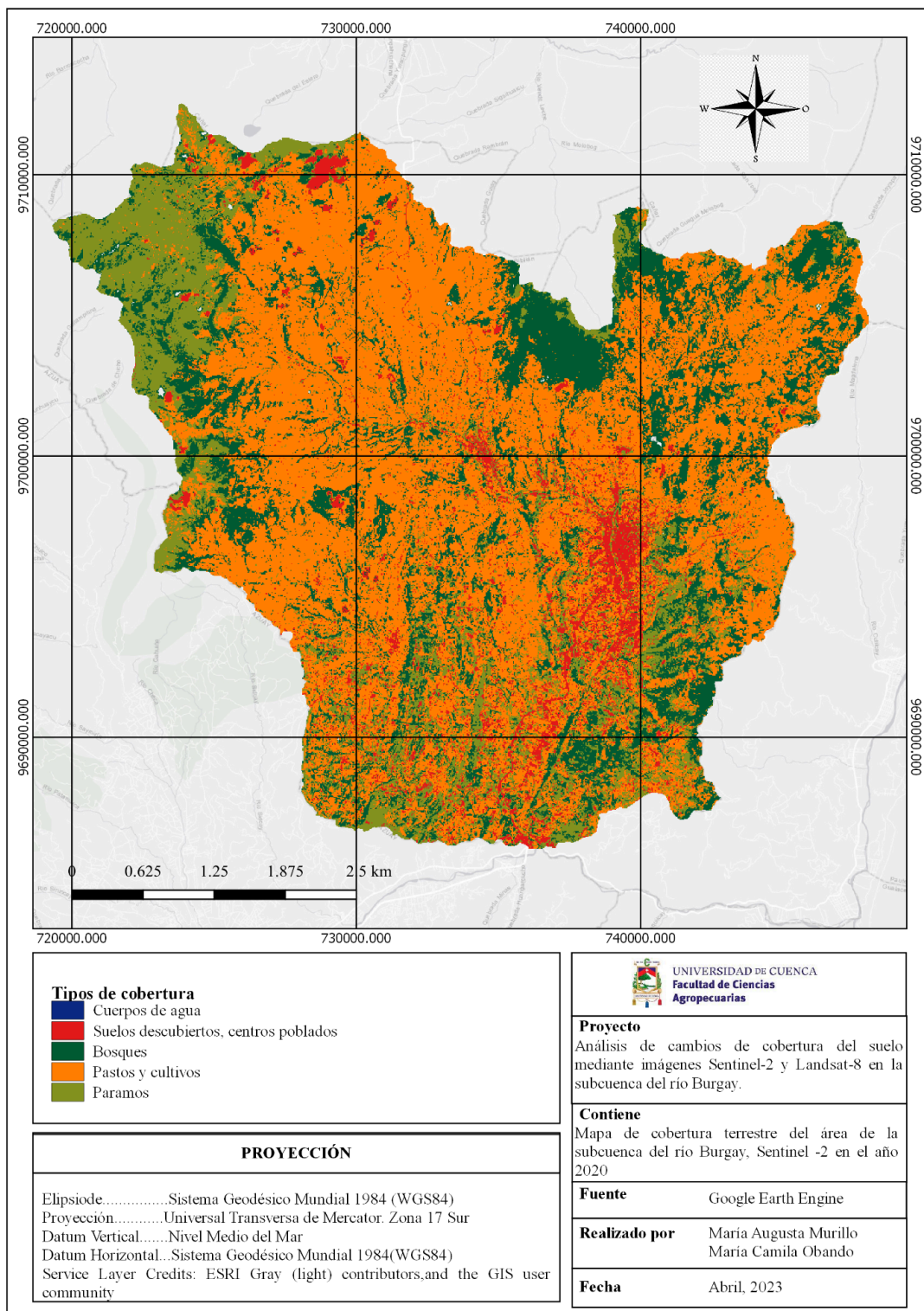
Tabla 16

Matriz de confusión del sensor Sentinel-2 en la subcuenca del río Burgay del año 2020

Satélite Sentinel 2		Matriz de confusión							
Año 2020		Cuerpos de agua.	Suelos descubiertos, centros poblados.	Bosque y Vegetación leñosa Nativa.	Pastos y Cultivos	Páramos	Total	Comisión	Exactitud del usuario (%)
Clase Actual	Cuerpos de agua.	31	0	0	0	0	31	0	100,00
	Suelos descubiertos, centros poblados	0	26	0	0	0	26	0	100,00
	Bosque	0	0	27	0	0	27	0	100,00
	Pastos y Cultivos.	0	0	2	30	0	32	2	93,75
	Páramos.	0	0	0	0	30	30	0	100,00
Total		31	26	29	30	30	146		
Omisión		0	0	2	0	0			
Exactitud del productor (%)		100,00	100,00	93,10	100,00	100,00			
Índices globales									
Overall Acuraccy				0.986					
Kappa				0.982					

Figura 17

Mapa de cobertura terrestre del área de la subcuenca del río Burgay, Sentinel-2 en el año 2020



Estos resultados sugieren que tanto Landsat-8 como Sentinel-2 pueden proporcionar información precisa y confiable sobre la cobertura terrestre en un período de tiempo determinado, lo que los convierte en herramientas valiosas para la investigación científica, la gestión ambiental y la toma de decisiones en la planificación del territorio.

Los resultados de precisión obtenidos para los años 2016, 2018 y 2020 revelan diferencias significativas entre Landsat-8 y Sentinel-2 en términos de clasificación de la cobertura del suelo en la subcuenca del río Burgay. Los porcentajes más altos de precisión general obtenidos con Sentinel-2 en comparación con Landsat-8 en 2016 y 2020 podrían atribuirse a su mayor resolución espacial y capacidad para detectar cambios sutiles en la cobertura del suelo. Esto se alinea con los hallazgos del autor Picone (2017), quien destaca la ventaja de Sentinel-2 para la monitorización de cambios en el uso del suelo y la vegetación debido a su alta resolución temporal y espectral. En particular, los resultados de evaluación de precisión general muestran un rendimiento superior de Sentinel-2 en todos los años estudiados, con puntajes más altos que los obtenidos con Landsat-8. Por ejemplo, en el año 2016, Sentinel-2 alcanzó una precisión general del 99,28%, mientras que Landsat-8 obtuvo un puntaje ligeramente inferior del 95,33%. Este patrón se repite en los años 2018 y 2020, donde Sentinel-2 continúa mostrando una mayor precisión en la clasificación de la cobertura del suelo en comparación con Landsat-8.

Sin embargo, es importante tener en cuenta que, en el año 2018, Landsat-8 mostró una precisión general ligeramente superior a la de Sentinel-2, aunque la diferencia fue mínima. Este hallazgo podría atribuirse a factores como la presencia de nubes o la calidad de las imágenes adquiridas durante ese año, como mencionado por autores como Aldaz et al. (2023), quienes analizaron el impacto de la interferencia atmosférica en las imágenes de Sentinel-2.

5.1.3 Dinámica de Cambio en las coberturas de suelo mediante diagramas de SANKEY

El diagrama de sankey presentado visualiza la extensión de la categoría de cobertura terrestre en el área de la subcuenca del río Burgay en cinco años, y la persistencia y el cambio que se produjo de 2016 a 2018, de 2018 a 2020 y de 2016 a 2020. Landsat-8 (*Figura 18*) y Sentinel-2 (*Figura 19*).

Nuestro análisis cronológico detallado revela dinámicas significativas en las coberturas de suelo, resaltando tanto las ganancias como las pérdidas en diversas categorías (*Tabla 17*). Los resultados obtenidos indican un incremento del 3% en áreas de pastos y cultivos entre 2016 y 2020 para Landsat-8 y un 7% para Sentinel-2; como afirma Vaca (2019), en su estudio “Metodologías para la evaluación de la influencia de actividades agropecuarias en la calidad del agua”, indica que en la cuenca alta, la vegetación natural presenta relictos de bosques con mínima intervención; sin embargo, en las cuencas media y baja, la cobertura vegetal natural ha sido mayormente destruida para dar paso a la agricultura y pastizales.

En el 2018 Landsat-8, muchas áreas que experimentaron cambios fueron predominantemente zonas agrícolas o forestales (*Figura 18*). En los datos del 2020-Landsat-8, hay una pérdida aparente de tierras agrícolas; sin embargo, muchas áreas nuevas están clasificadas como bosques.

En contraste, los bosques experimentaron una disminución del 22% entre 2016 y 2020 para Landsat-8 y una disminución del 27% para Sentinel-2 (*Figura 19*). Estos hallazgos encuentran respaldo en estudios de zonificación económica ecológica y análisis históricos realizados por Bravo (2019), que documentan la reducción de bosques primarios y secundarios durante aproximadamente cinco años debido a la deforestación y actividades antropogénicas.

En términos de cobertura de páramo, suelos descubiertos y centros poblados, estos mantuvieron un porcentaje constante a lo largo de todo el periodo de estudio. En lo que respecta a los cuerpos de agua, no se observaron diferencias significativas en los valores entre los dos satélites, ya que ambos muestran una tendencia similar. Sin embargo, durante el proceso de clasificación, resultó más sencillo identificar los cuerpos de agua utilizando Sentinel-2, principalmente debido a la mayor cantidad y calidad de las bandas espectrales disponibles en este satélite.

Tabla 17

Ganancias y pérdidas en las diferentes coberturas del suelo para los periodos 2016-2018, 2018-2020 para los satélites, Landsat-8 y Sentinel-2

Cobertura	Landsat-8 (2016 - 2018)	Landsat-8 (2018 - 2020)	Ganancia/ Pérdida Landsat-8	Sentinel-2 (2016 - 2018)	Sentinel-2 (2018 - 2020)	Ganancia/ Pérdida Sentinel-2
Pastos y Cultivos	67%	70%	+3%	79%	86%	+7%
Bosque	64%	42%	-22%	30%	57%	+27%
Suelo Descubierto y Centros Poblados	51%	66%	+15%	40%	65%	+25%
Páramos	48%	65%	+17%	54%	60%	+6%
Cuerpos de Agua	2%	50%	+48%	-	1%	+1%

El conjunto de líneas en el diagrama de Sankey representa la persistencia y transición entre los mapas de cobertura de suelo de un tiempo inicial y uno posterior, conectando dos categorías y representando las transiciones o persistencias en ambos tiempos. El grosor de cada línea es proporcional al tamaño de área de tierra que experimenta la persistencia o transición correspondiente.

En adición a esto, fue pertinente la generación de mapas que marcaran con puntualidad esta dinámica de cambio generadas por el análisis de Landsat-8 (*Figura 20*) y Sentinel-2 (*Figura 21*)

Estos son un análisis optimizado de puntos de las áreas de las coberturas que probablemente experimentaron cambios en la subcuenca del río Burgay. Se pueden observar 3 tipos de cambios generados en Google Earth Engine función Land Use Change, y codificación Python (*Anexo 4 y 5*) entre el lapsus del 2016 a 2020.

5.1.4 Dinámica de Cambios en las Coberturas Sankey

Figura 18

Diagrama de Sankey representando los cambios en la cobertura del suelo en los años 2016, 2018 y 2020 con Landsat-8

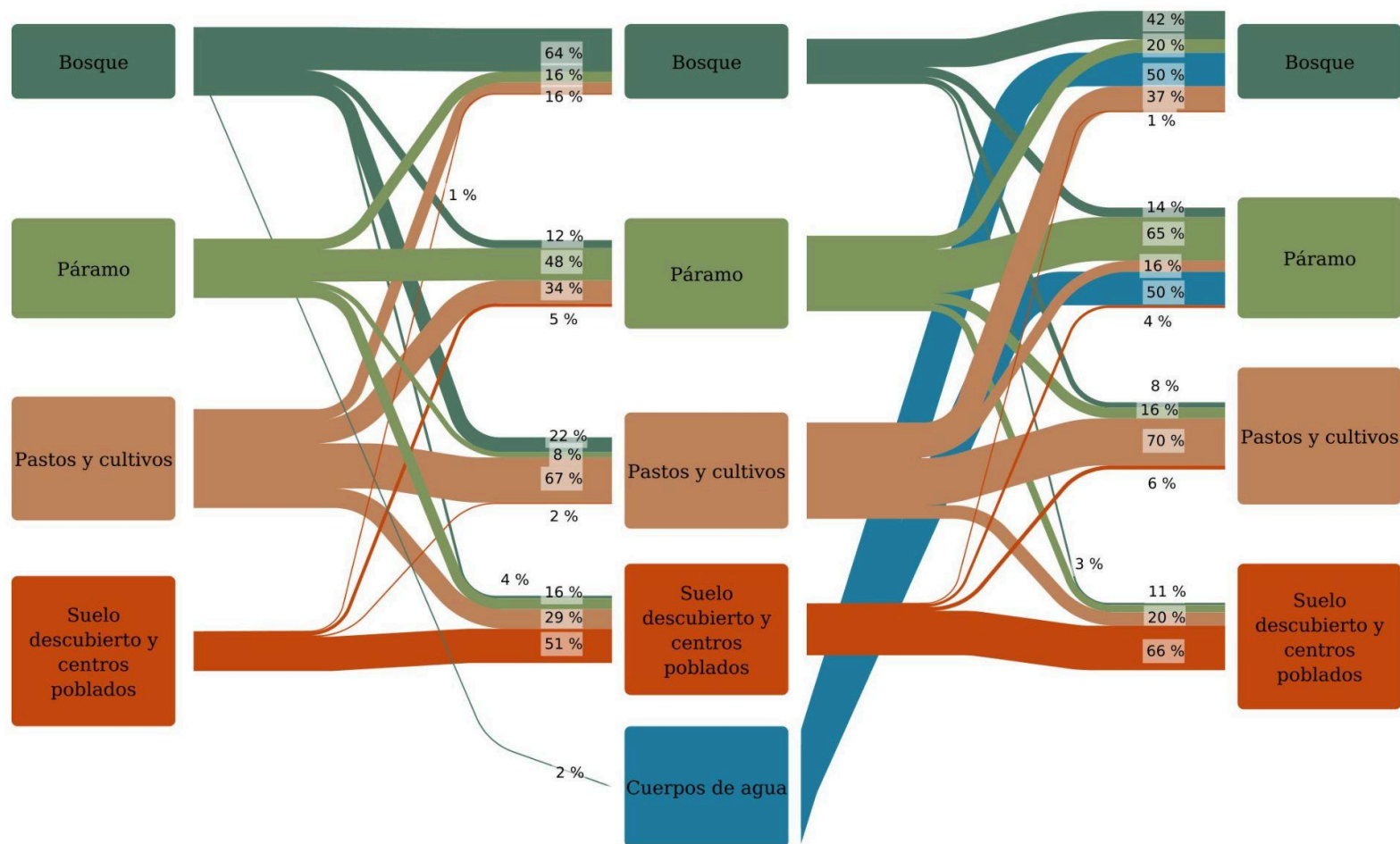


Figura 19

Diagrama de Sankey representando los cambios en la cobertura del suelo en los años 2016, 2018 y 2020 con Sentinel-2

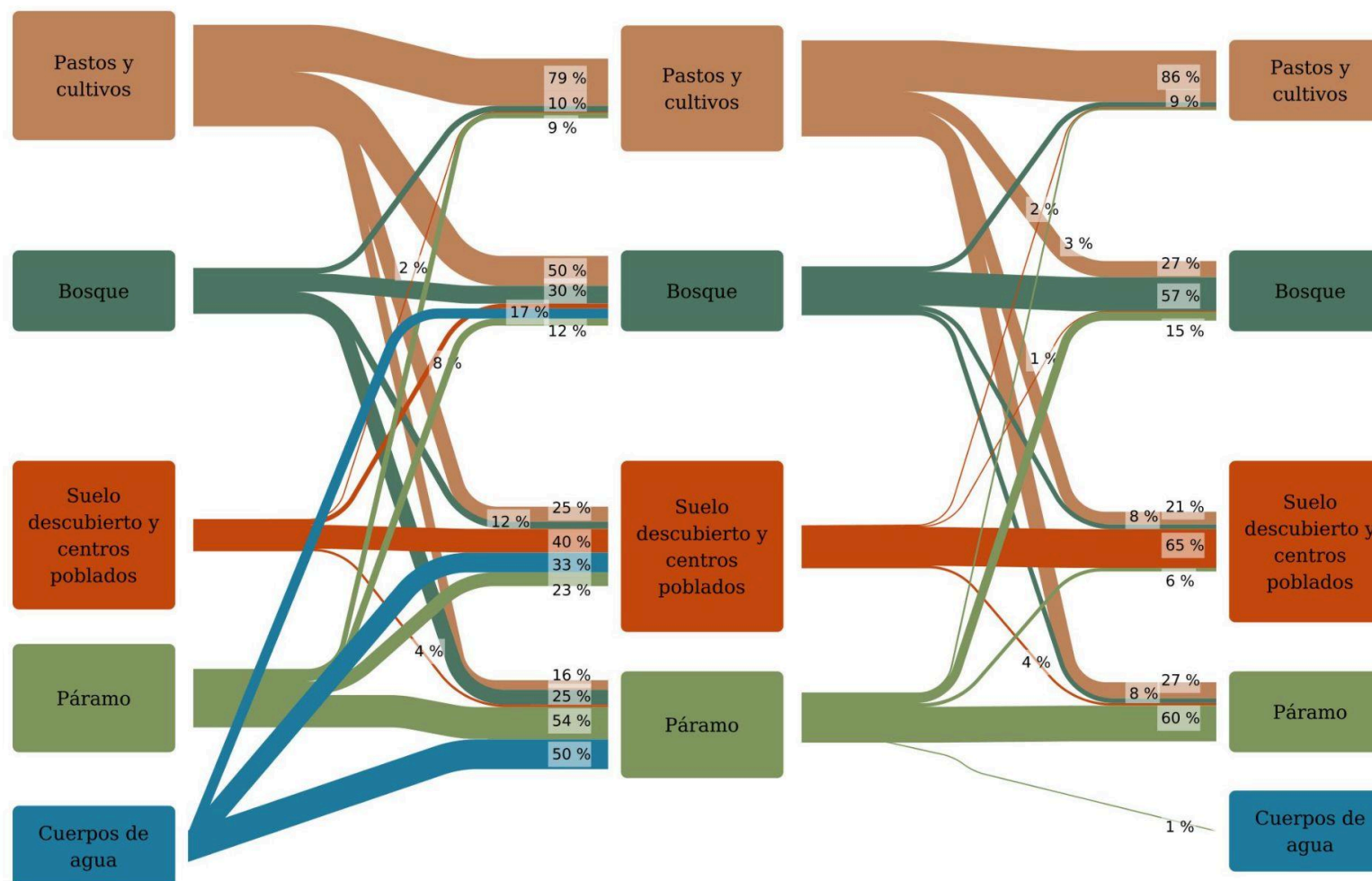
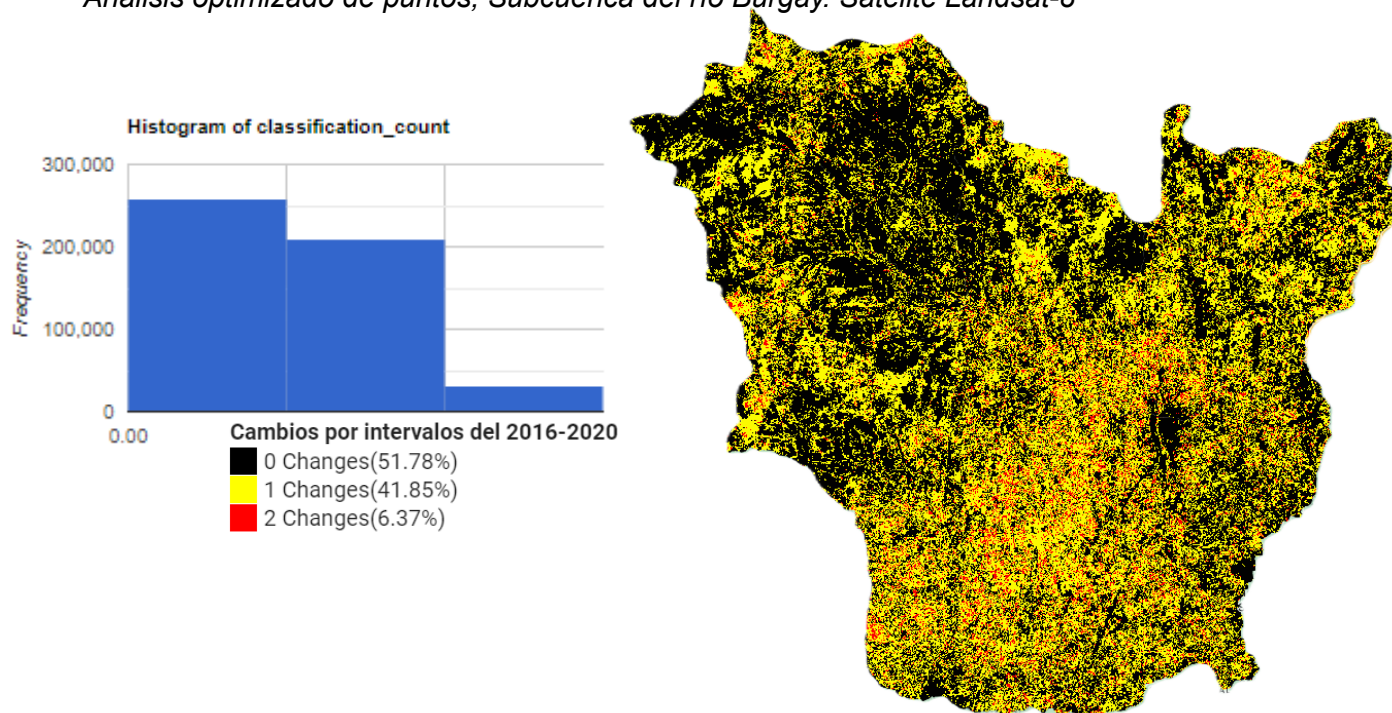


Figura 20

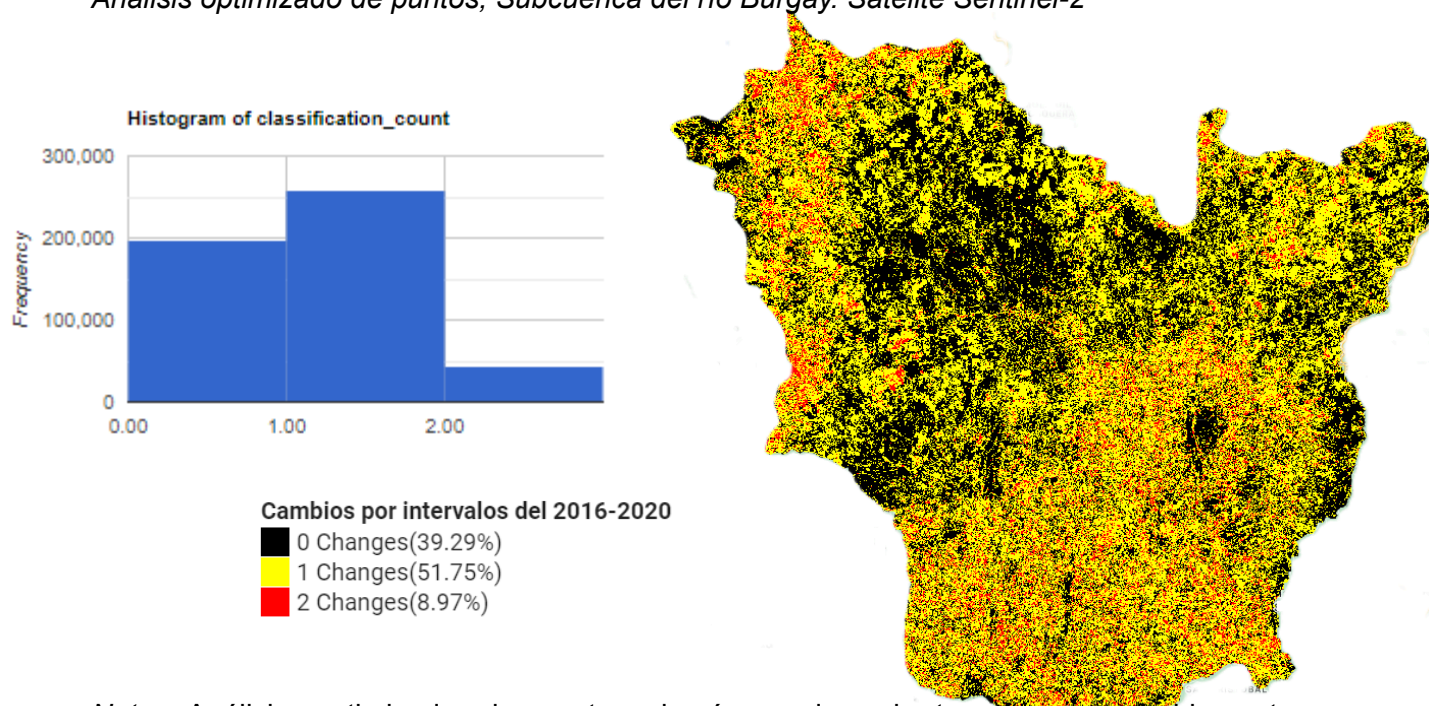
Análisis optimizado de puntos; Subcuenca del río Burgay. Satélite Landsat-8



Nota. Análisis optimizado de puntos de áreas de coberturas que probablemente experimentaron cambios en la Subcuenca del río Burgay. Satélite Landsat-8.

Figura 21

Análisis optimizado de puntos; Subcuenca del río Burgay. Satélite Sentinel-2



Nota. Análisis optimizado de puntos de áreas de coberturas que probablemente experimentaron cambios en la Subcuenca del río Burgay. Satélite Sentinel-2.

Validación PlanetLab

La aplicación de modelos de clasificación a imágenes de alta resolución provenientes de plataformas como PlanetLab ha sido un tema de interés creciente en la teledetección. Un estudio reciente, liderado por Csillik et al. (2019), exploró la eficacia de un modelo de clasificación basado en aprendizaje profundo para discriminar características específicas en imágenes de PlanetLab. Sin embargo, este análisis no estuvo exento de desafíos notables.

La validación de nuestro modelo se llevó a cabo mediante un conjunto de datos de referencia que incorporaba anotaciones precisas de cobertura terrestre. No obstante, se observó que la clasificación de áreas boscosas y cuerpos de agua experimentó cierta ambigüedad, destacando la necesidad de abordar específicamente la confusión entre estas clases durante futuras mejoras del modelo.

Adicionalmente, hicimos hincapié en comparar la precisión global (OA) y el coeficiente kappa entre nuestras intervalidaciones y la validación en PlanetLAB. Estás evidentemente mostraron que a pesar que nuestro modelo principal había mostrado tener un excelente rendimiento, al validarlas con las imágenes los porcentajes cayeron. Landsat-8 (2016) de 95,33% a 92,51%; Landsat-8 (2018) de 96,75% a 94,64% Landsat-8 (2020) de 98,62% a 95,4%. Los resultados en Sentinel-2 también mostraron descender: Sentinel-2 (2016) de 99,28% a 97,38%; Sentinel-2 (2018) de 93,65% a 91,42% Sentinel-2 (2020) de 98,63% a 94,16% todo esto respecto a la precisión global (OA). En cuanto al coeficiente kappa los valores también descendieron Landsat-8 (2016) de 94,14% a 92,1%; Landsat-8 (2018) de 95,9% a 93,92% Landsat-8 (2020) de 98,26% a 95,21%. Sentinel-2 (2016) de 99,09% a 96,7%; Sentinel-2 (2018) de 91,98% a 90,71% Sentinel-2 (2020) de 98,28% a 92,67%.

Como afirma Tian et al. (2023) , los paisajes heterogéneos son desafíos para cualquier algoritmo de fusión de datos; razón por la cual la validación con imágenes satelitales de alta resolución es de los métodos más utilizados para la verificación de datos, presentado nuestros resultados similitud con su estudio de reflectancia adaptativa espacial y temporal en el que validaron sus resultados de la clasificación con imágenes satelitales de alta resolución adquiridas en diferentes fechas de forma independiente. Kelly-Fair et al. (2022), en su estudio de la cobertura de árboles de caucho en los suelos de Indonesia, valida sus datos mediante una reclasificación e imágenes de alta resolución obtenidas de la plataforma NASA Worldview, sus resultados de validación fueron óptimos en la escala del 87% al 93% de precisión general y coeficiente Kappa.

5.2 Resultados del objetivo número 2

En esta sección reportamos las características que presentan cada satélite: Landsat-8 y Sentinel-2, las cuales potenciaron o limitaron la calidad de análisis, al estudiar la cobertura del suelo en la Subcuenca del río Burgay. Lo cual corresponde al objetivo 2.

5.2.1 Ventajas y desventajas de las imágenes de Landsat-8 y Sentinel-2

Precisión: Landsat-8 tiene una ventaja en precisión gracias a su experiencia probada en la corrección geométrica y radiométrica de imágenes. Sin embargo, puede experimentar problemas en áreas con nubosidad persistente, lo que puede afectar la calidad de las imágenes. Por otro lado, Sentinel-2 ofrece una mayor precisión debido a tecnologías avanzadas y correcciones atmosféricas integradas. No obstante, también puede tener dificultades en áreas con pendientes pronunciadas, lo que podría comprometer la precisión de los datos en terrenos accidentados.

Resolución espacial: Landsat-8 ofrece una resolución espacial de 30 metros, lo que permite detectar detalles finos en la superficie terrestre. Esta resolución es adecuada para una amplia gama de aplicaciones, aunque puede no ser suficiente para aquellas que requieran una resolución muy alta. Por otro lado, Sentinel-2 proporciona una resolución espacial de 10 metros en bandas visibles e infrarrojas cercanas, lo que permite una mayor discriminación de características terrestres. Sin embargo, esta resolución también puede ser limitada para aplicaciones que necesiten detalles extremadamente finos.

Resolución espectral: Landsat-8 cuenta con una amplia gama de bandas espectrales que facilitan la caracterización detallada de diferentes tipos de coberturas terrestres. Sin embargo, la resolución espectral podría no ser suficiente para algunas aplicaciones avanzadas de teledetección. Por otro lado, Sentinel-2 ofrece una mayor resolución espectral con 13 bandas, incluyendo bandas específicas para la detección de la vegetación y la calidad del agua. A pesar de esta ventaja, podría generar problemas de redundancia espectral en ciertos casos.

Resolución temporal: Landsat-8 ofrece una frecuencia de revisita de 16 días, lo que permite un monitoreo regular de cambios en la superficie terrestre. Sin embargo, esta frecuencia puede no ser suficiente para aplicaciones que requieran un monitoreo más frecuente. Por otro lado, Sentinel-2 tiene una frecuencia de revisita de 5 días, lo que permite un monitoreo más frecuente y una captura rápida de cambios. Aunque esto proporciona una

mayor cantidad de datos para análisis, también puede generar una carga de trabajo más pesada en términos de procesamiento.

Calibración y corrección: Landsat-8 requiere calibración y corrección posterior para garantizar la precisión de los datos. Por otro lado, Sentinel-2 realiza correcciones automáticas en línea, lo que reduce la necesidad de procesamiento adicional y proporciona datos más listos para usar.

Usos: Landsat-8 es ideal para una variedad de aplicaciones, como monitoreo ambiental, agricultura y gestión de recursos naturales. Por otro lado, Sentinel-2 es adecuado para aplicaciones de agricultura de precisión, detección de cambios en la cubierta terrestre y seguimiento de la calidad del agua.

Costos: En cuanto a los costos, tanto Landsat-8 como Sentinel-2 ofrecen datos gratuitos, lo que reduce significativamente los costos de adquisición para los usuarios. Sin embargo, ambas plataformas pueden implicar gastos adicionales relacionados con el procesamiento y análisis de las imágenes.

Tabla 18

Resumen de comparación de los satélites Landsat-8 y Sentinel-2

Aspecto	Landsat 8	Sentinel 2
Precisión	Experiencia probada en corrección geométrica y radiométrica, pero problemas en áreas con nubosidad persistente	Mayor precisión gracias a tecnologías avanzadas y correcciones atmosféricas, pero dificultades en áreas con pendientes pronunciadas
Resolución espacial	30 metros	10 metros en bandas visibles e infrarrojas cercanas
Resolución espectral	Amplia gama de bandas espectrales	13 bandas, incluyendo bandas específicas para la detección de la vegetación y la calidad del agua
Resolución temporal	Frecuencia de revisita de 16 días	Frecuencia de revisita de 5 días
Calibración y corrección	Requiere calibración y corrección posterior	Realiza correcciones automáticas en línea
Usos	Monitoreo ambiental, agricultura, gestión de recursos naturales	Agricultura de precisión, detección de cambios en la cubierta terrestre, seguimiento de la calidad del agua
Costos	Datos gratuitos, posibles costos adicionales de procesamiento y análisis	Datos gratuitos, posibles costos adicionales de procesamiento y análisis

Las imágenes satelitales Landsat-8 y Sentinel-2 son herramientas valiosas para el estudio de la cobertura del suelo en las zonas andinas, cada una con sus propias ventajas y desventajas. Landsat-8 destaca por su extensa historia de datos, brindando la posibilidad de realizar análisis a largo plazo en la región andina. Además, algunas de sus bandas están optimizadas para penetrar la atmósfera, siendo beneficiosas para estudios en áreas montañosas. Sin embargo, su resolución espacial de 30 metros puede limitar la capacidad para capturar detalles finos en terrenos accidentados, y su frecuencia de captura más baja puede ser una desventaja para el monitoreo de cambios rápidos.

Por otro lado, Sentinel-2 ofrece una alta resolución espacial de 10 metros y una visita rápida, lo que facilitó la identificación de detalles finos y el monitoreo dinámico entre las coberturas. Esto está muy presente entre sus potencialidades para hacer seguimientos en zonas andinas, seguimiento de la calidad del agua y agricultura de precisión.

Aunque tiene una serie temporal más corta y puede implicar costos asociados con el procesamiento, la combinación de ambas plataformas puede proporcionar una visión más completa de la cobertura del suelo en esta compleja región montañosa.

5.3 Resultados del objetivo número 3

Los resultados que presentamos para este objetivo, fue el desarrollo de un tutorial para el análisis de cambio de cobertura del suelo en cuencas andinas usando imágenes satelitales. El cual está explícita y detalladamente redactado desde la creación de una cuenta en Google Earth Engine, la carga de la subcuenca, la búsqueda y selección de imágenes hasta una clasificación supervisada que se expone paso a paso la toma de datos de entrenamiento. También se especifica el código utilizado en cada punto, utilizando el lenguaje de programación JavaScript; muchos de estos desarrollados por Google Earth Engine y otros desarrollados por nosotras las autoras. Con pautas de cómo utilizarlo y los posibles errores que se puedan encontrar y cómo evitarlos para una correcta aplicación de la codificación. Por último se presentan los resultados y los mapas del cambio de cobertura del suelo desarrollados con Python. Este tutorial se encuentra anexo a la tesis.

Figura 22

Portada principal, tutorial



6. Conclusiones

El uso de Google Earth Engine resultó ser la herramienta principal y esencial para el análisis de la cobertura del suelo en esta tesis. Su capacidad para procesar y manejar grandes volúmenes de datos satelitales de manera eficiente permitió la generación de resultados robustos y altamente precisos. La plataforma no solo facilitó la integración de datos de Landsat-8 y Sentinel-2, sino que también optimizó el rendimiento del algoritmo Random Forest, mejorando significativamente la calidad y fiabilidad de los mapas de cobertura del suelo producidos. Las características avanzadas de Google Earth Engine, como su potente capacidad de procesamiento en la nube y su amplia accesibilidad a datos geoespaciales, fueron fundamentales para alcanzar los objetivos de esta investigación, demostrando su valor incomparable en estudios de cambio de cobertura del suelo.

El diagrama de Sankey ha sido también una herramienta fundamental en este análisis, proporcionando una representación visual intuitiva y detallada de las transiciones entre diferentes categorías de cobertura de suelo a lo largo del tiempo. Este tipo de diagrama logró identificar fácilmente los flujos de cambio y la persistencia de cada categoría, facilitando la interpretación de datos complejos. En particular, se destacó la utilidad del diagrama para mostrar el incremento en áreas de pastos y cultivos, así como la disminución de áreas forestales, ofreciendo una visión clara de cómo las actividades humanas están impactando el paisaje de la subcuenca del río Burgay.

Finalmente podemos concluir que el análisis de los cambios de cobertura del suelo utilizando imágenes de Landsat-8 y Sentinel-2 se revela como un recurso crucial para comprender las dinámicas ambientales. Esto se traduce en una herramienta fundamental para futuros estudios que aborden temas como gestión del agua, planificación urbana sostenible, conservación forestal, agricultura y preservación de ecosistemas frágiles como los páramos en la Subcuenca del Río Burgay.

7. Recomendaciones

El uso de Random Forest como algoritmo clasificador de cobertura, es de los algoritmos más utilizados para el análisis de las coberturas del suelo, pero tienen una baja tasa de predicción. Para continuar con un análisis más exhaustivo y efectuar predicciones en trabajos futuros; sobre cómo podría cambiar la cobertura al cabo de una década o más, se recomienda el uso de algoritmos combinados en la plataforma GEE y el uso del modelo integrado de Markov, a partir del análisis de tabulación cruzada; Metodología propuesta por Al-sharif y Pradhan (2014).

Se recomienda tener siempre en cuenta la importancia de examinar detenidamente las clases individuales al evaluar la precisión de la clasificación, ya que las tasas de error pueden variar significativamente entre ellas, lo que tiene implicaciones importantes para la interpretación y aplicación de los resultados en estudios posteriores.

Las nuevas tecnologías lanzadas en noviembre del 2023 son realmente un avance en el campo de la teledetección, pueden lograr una identificación de cobertura a 3 metros del suelo. En la validación, para la descarga de imágenes satelitales de alta resolución se recomienda el uso de la plataforma Planet Lab a 3 metros; el cual, por su gran efectividad y claridad sólo están disponibles bajo pedido, sumando un peso económico al trabajo, pero de igual manera también lo añade de manera investigativa, se recomienda su uso.

El 27 de septiembre del 2021 fue lanzada la nueva misión de Landsat, número 9. Aunque mantiene muchas características similares a la de su predecesor Landsat-8; las actualizaciones de este permitirán a futuro una toma de datos más detallada. Se recomienda el uso de la nueva versión y estudios comparativos entre las misiones. Esta última recomendación la realizamos basada en que uno de los principales propósitos de esta tesis fue siempre mantener un estilo comparativo entre Landsat-8 y Sentinel-2 (las últimas versiones de cada satélite al realizar la investigación) de manera que el análisis fuera lo más actual posible hasta la presente fecha.

Referencias

- AHADY, A. B., & KAPLAN, G. (2022). Classification comparison of Landsat-8 and Sentinel-2 data in Google Earth Engine, study case of the city of Kabul. *International Journal of Engineering and Geosciences*, 7(1), 24-31. <https://doi.org/10.26833/ijeg.860077>
- Aldaz, B., Alessanderks y Ramírez Patiño, E., & Ángel, M. (s/f). Análisis multitemporal para la ubicación de actividades ilícitas en la frontera norte del Ecuador provincias de Carchi y Esmeraldas, mediante el uso de imágenes satelitales Sentinel-2 como apoyo a las operaciones militares. [Trabajo de titulación, Universidad de las Fuerzas Armadas]. <https://repositorio.espe.edu.ec/bitstream/21000/37154/3/T-ESPE-058251.pdf>
- Arenas, R. (2016). Aplicación de la teledetección en la exploración geominera y de recursos naturales. [Tesis de maestría, Universidad de Oviedo, Escuela de Ingenieros en minas, energía y materiales]. <https://digibuo.uniovi.es/dspace/bitstream/handle/10651/38502/?sequence=3>
- Avogadro, E. G., & Padró, J. C. (2019). Comparación de métodos de clasificación aplicados a imágenes Sentinel-2 y Landsat-8, para la diferenciación de plantaciones forestales en entre ríos, argentina. *Geofocus: Revista Internacional de Ciencia y Tecnología de la Información Geográfica*, (24), p 117-139. ISSN: 1578-5157. <http://dx.doi.org/10.21138/GF.652>
- Bonilla Vilatuña, D. M., & Taípe Quispe, J. M. (2021). Análisis multitemporal de la cobertura vegetal a partir de la imagen satelital Sentinel-2 y Landsat-8 mediante el uso de algoritmos de clasificación, en la Reserva Ecológica El Ángel y Reserva Ecológica Arenillas. [Trabajo de titulación, Universidad Politécnica Salesiana, Carrera Ingeniería Ambiental]. <https://dspace.ups.edu.ec/handle/123456789/20884>
- Bravo, A. M. (2019). Propuesta de manejo integral para la microcuenca hidrográfica del río Burgay bajo, provincia del Cañar. [Trabajo de titulación, Universidad Politécnica Salesiana, Carrera Ingeniería Ambiental]. <https://dspace.ups.edu.ec/handle/123456789/17934>

- Calle, G., Chang, J. (2016). Índices de calidad del agua de fuentes superficiales y aspectos toxicológicos, evaluación del Río Burgay. *Maskana*, 5, 165–176. <https://publicaciones.ucuenca.edu.ec/ojs/index.php/maskana/article/view/564>
- Camacho, J. (2017). Cambios de cobertura/uso del suelo en una porción de la Zona de Transición Mexicana de Montaña. *Madera bosques* vol.23 no.3 Xalapa sep. 2017, 93.
- Campbell, J. R., Reid, J. S., Westphal, D. L., Zhang, J., Tackett, J. L., Chew, B. N., Welton, E. J., Shimizu, A., Sugimoto, N., Aoki, K., & Winker, D. M. (2013). Characterizing the vertical profile of aerosol particle extinction and linear depolarization over Southeast Asia and the Maritime Continent: The 2007–2009 view from CALIOP. *Atmospheric Research*, 122, 520–543. <https://doi.org/10.1016/j.atmosres.2012.05.007>
- Cardenas, C., Quesada, F. (2007). Análisis del estado de conservación de cuatro microcuencas hidrográficas de la ciudad de Azogues, provincia del Cañar. [Trabajo de titulación, Universidad del Azuay, Facultad de Ciencia y Tecnología]. <https://dspace.uazuay.edu.ec/handle/datos/75>
- Carvajal, A., Pabón, J.(2016). Transformación de la superficie terrestre por la actividad humana y su relación con el cambio climático. *Sociedade & natureza*, 28(2), 185–198. <https://doi.org/10.1590/1982-451320160201>
- Chuvieco, E. (2010) *Teledetección Ambiental. La observación de la tierra desde el espacio*. Editorial Planeta, S. A. Barcelona.
- Csillik, O., Kumar, P., Mascaro, J., O'Shea, T., & Asner, G. P. (2019). Monitoring tropical forest carbon stocks and emissions using Planet satellite data. *Scientific reports*, 9(1),1–12. 17831. <https://doi.org/10.1038/s41598-019-54386-6>
- Congalton, R. G. (1988b), "A comparison of sampling scheme use in generating error matrices for assessing the accuracy of maps generated from remotely sensed data", *Photogrammetric Engineering and Remote Sensing*, 54(5):593-600.
- Congalton, RG (1991) Una revisión de la evaluación de la precisión de las clasificaciones de los datos de detección remota. *Teledetección del Medio Ambiente*, 37, 35-46. [http://dx.doi.org/10.1016/0034-4257\(91\)90048-B](http://dx.doi.org/10.1016/0034-4257(91)90048-B)

- Copernicus. (2015). Copernicus: La mirada de Europa sobre la Tierra. Bélgica. Comisión Europea. <https://www.copernicus.eu/es>
- Correa, J. (2018). Análisis multitemporal de coberturas y usos del suelo: Transformaciones ambientales a través del tiempo en Armenia, Quindío. [Trabajo de titulación, Universidad Tecnológica de Pereira, Facultad de Ciencias Ambientales]
<https://repositorio.utp.edu.co/items/9a93b8cf-c993-4649-b751-3ce93ee802d7/full>
- De Sousa, C.; Fatoyinbo, L.; Neigh, C.; Boucka, F.; Angoue, V.; Larsen, T. (2020). Cloud-computing and machine learning in support of country-level land cover and ecosystem extent mapping in Liberia and Gabon. PLoS ONE, 15, e0227438. <https://doi.org/10.1371/journal.pone.0227438>
- De Ullibarri, L., Pita, S. (1999). Medidas de concordancia: el índice de Kappa. Cad Aten Primaria, 6, 169-171. https://d1wqtxts1xzle7.cloudfront.net/33328322/Indice_de_kappa-libre.pdf?1395943383=&response-content-disposition=inline%3B+filename%3DInvestigacion_Medidas_de_concordancia_el.pdf&Expires=1717523635&Signature=X02y3ldXT5~J2CyNJrhiSNthwUEIL9lufO2fvikJuPbhXBnuQa8dLZkvRCw11ivUCa64gCGfzeVh6JjTxhCHnW~luk9ygOccv0zMVyp1rOE4IN6To9OAHgjp~wOxyl42dRgFTcx3ko6~Pvi2obfTJgek23TCJDDatNy1xKkJDCmtNZ-FEQcavdrfStDrNPQUxwamnqzg5Dr72T9Bv~9Mpqjpg44ePBozdWNeNPZ6bZ8-hUxEksmN-1ULXyneGIY2aCWjQy7cyAKDIBCB2Dd8SfMvK3UV2sx-iS3j2mSmPI0WidVM1MjxleW98KF0M1ZofkLTCqsm0xq6Oz9uyravAA__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA
- Delegido, J., Tenjo, C., Ruiz, A., Pereira, M., Pasqualotto, N., Gibaja, G., . . . Moreno, J. (2016). Aplicaciones de Sentinel-2 a estudios de vegetación y calidad de aguas continentales. Luján: Actas XVII Simposio Internacional SELPER 2016. https://www.researchgate.net/profile/Javier-Sanchis-Munoz/publication/311572126_Aplicaciones_de_Sentinel-2_a_estudios_de_vegetacion_y_calidad_de_aguas_continentalas/links/5ae0ddfa458515c60f65f14c/Aplicaciones-de-Sentinel-2-a-estudios-de-vegetacion-y-calidad-de-aguas-continentalas.pdf
- Díaz, A., Gaspari, F. (2017). Cambio de cobertura y uso del suelo en la zona ribereña en cuencas subtropicales del noroeste argentino. Quebracho - Revista de Ciencias Forestales, vol. 25, núm. 1-2, 2017, Enero-Diciembre, pp. 28-39.

http://www.scielo.org.ar/scielo.php?pid=S1851-30262017000100028&script=sci_arttext

Dwire, K., Brown, S. (2017). Climate Change and Special Habitats in the Blue Mountains: Riparian Areas, Wetlands, and Groundwater-Dependent Ecosystems. Climate change vulnerability and adaptation in the Blue Mountains. Gen. Tech. Rep. PNW-GTR-939. Portland, OR: US Department of Agriculture, Forest Service, Pacific Northwest Research Station. 251-323.https://www.researchgate.net/profile/Ka-Dwire/publication/322869852_Chapter_7_Climate_Change_and_Special_Habitats_in_the_Blue_Mountains_Riparian_Areas_Wetlands_and_Groundwater-Dependent_Ecosystems/links/5a73773b458515512077c8dd/Chapter-7-Climate-Change-and-Special-Habitats-in-the-Blue-Mountains-Riparian-Areas-Wetlands-and-Groundwater-Dependent-Ecosystems.pdf

ESA (The European Space Agency). (2023). <https://www.esa.int/>

Evans, J. S., & Rodriguez, A. B. (2018). Applications of Google Earth Engine in the Earth Sciences: A Review. *Remote Sensing*, 10(10), 1602.

Foody, G. M. (2010). Assessing the accuracy of land cover change with imperfect ground reference data. *Remote Sensing of Environment*, 114(10), 2271-2285.<https://doi.org/10.1016/j.rse.2010.05.003>

Flood, N. (2017). Comparing Sentinel-2A and Landsat 7 and 8 Using Surface Reflectance over Australia. *Remote Sensing*, 9(7), 659.<https://doi.org/10.3390/rs9070659>

Gaitán Rojas, D. J., & López Calle, M. I. (2018). Análisis multitemporal de la especie vegetal invasora Retamo espinoso (*Ulex europaeus*) en el embalse la regadera, zona rural de la localidad de Usme, a partir de Imágenes satelitales Sentinel 2 y Landsat 8 mediante el uso de algoritmos de clasificación.[Trabajo final de grado, Universidad Distrital Francisco José de Caldas, Ingeniería Catastral y Geodesia]. <https://repository.udistrital.edu.co/handle/11349/13799>

García, D., Camacho, M., François, J., Paegelow, M.(2022). Mapeo, Modelado y Validación de la Cobertura del Uso del Suelo. Un fondo.En: García-Álvarez, D., Camacho Olmedo, MT, Paegelow, M., Mas, JF (eds) Land Use Cover Datasets

and Validation Tools. Springer, Cham.
https://doi.org/10.1007/978-3-030-90998-7_2

Gumma, M. K., Nelson, A., Thenkabail, P. S., & Singh, A. N. (2011). Mapping rice areas of South Asia using MODIS multitemporal data. *Journal of applied remote sensing*, 5(1), 053547-053547. <https://doi.org/10.1117/1.3619838>

GEE (Google Earth Engine). (2023). User Guide. Recuperado de <https://developers.google.com/earth-engine/guides>

Graciani, S., Del Braco, P., Stangaferro, J.(2011). Principios Físicos de la Teledetección. Universidad Nacional del Litoral.
<http://www.bibliotecacpa.org.ar/greenstone/collect/facagr/index/assoc/HASH671d.dir/doc.pdf>

Gorelick, N., Hancher, M., Dixon, M., Ilyushchenko, S., Thau, D., & Moore, R. (2017). Google Earth Engine: análisis geoespacial a escala planetaria para todos. *Remote Sensing of Environment*, 202, 18-27. <https://doi.org/10.1016/j.rse.2017.06.031>

Hernández Cruz, G. (2016). Análisis tecnoeconómico de una turbina de gas híbrida (solar-fósil). [Trabajo de fin de grado, Universidad Politecnica de Madrid, Especialidad Técnicas energéticas]. https://oa.upm.es/42500/1/TFG_GUILLERMO_HERNANDEZ_CRUZ.pdf

Huang, H., Chen, Y., Clinton N., Wang, J., Wang, X., Liu, C., Gong, P., Yang, P., Bai, Y., Zheng, Y., Zhu, Z. (2017). Mapping major land cover dynamics in Beijing using all Landsat images in Google Earth Engine, *Remote Sensing of Environment*, Volume 202, Pages 166-176,ISSN 0034-4257. <https://doi.org/10.1016/j.rse.2017.02.021>

IBM. (2022). ¿Qué es el Machine Learning?. [IBM.com. https://www.ibm.com/es-es/topics/machine-learning](https://www.ibm.com/es-es/topics/machine-learning)

IGM. (2015). Instituto Geográfico Militar Ecuatoriano. Datos de cuencas y subcuencas geográficas. Geoportal de descargas. <http://www.geograficomilitar.gob.ec/>

IPCC. (2019). Climate Change and Land: An IPCC Special Report on climate change, desertification, land degradation, sustainable land management, food security,

and greenhouse gas fluxes in terrestrial ecosystems. Intergovernmental Panel on Climate Change. <https://philpapers.org/rec/SHUCCA-2>

Jara, C., Delegido, J., Ayala, J., Lozano, P., Armas, A., Flores, V. (2019). Estudio de bofedales en los Andes ecuatorianos a través de la comparación de imágenes Landsat-8 y Sentinel-2. *Revista de teledetección*, (53), 45-57. <https://riunet.upv.es/handle/10251/122783>

Jensen, J. R., & Im, J. (2007). Remote sensing change detection in urban environments. *Geo-spatial Technologies in Urban Environments: Policy, Practice, and Pixels*, 7-31. https://link.springer.com/chapter/10.1007/978-3-540-69417-5_2

Ji, J. (2020). Desarrollo de una Herramienta de Integración de datos de imágenes de satélite en Google Earth Engine. [Trabajo de grado, Universidad Politécnica de Madrid, Ingeniería Informática]. <https://oa.upm.es/63348/>

Kelly-Fair, M., Gopal, S., Koch, M., Pancasakti Kusumaningrum, H., Helmi, M., Khairunnisa, D., & Kaufman, L. (2022). Analysis of land use and land cover changes through the lens of SDGs in Semarang, Indonesia. *Sustainability*, 14(13), 7592. <https://www.mdpi.com/2071-1050/14/13/7592>

Landis, J. R., & Koch, G. G. (1977). An application of hierarchical kappa-type statistics in the assessment of majority agreement among multiple observers. *Biometrics*, 363-374. <https://www.jstor.org/stable/2529786>

Mahdianpari, M.; Salehi, B.; Mohammadimanesh, F.; Motagh, M. (2017). Random forest wetland classification using ALOS-2 L-band, RADARSAT-2 C-band and TerraSAR-X imagery. *ISPRS J. Photogramm. Remote Sens.* 130, 13–31. <https://www.sciencedirect.com/science/article/pii/S0924271617300850>

Maisueche, A. (2018). Utilización del Machine Learning en la Industria 4.0. Universidad de Valladolid.[Trabajos Fin de Máster, Universidad de Valladolid, Escuela de Ingenierías Industriales]. <https://uvadoc.uva.es/handle/10324/37908>

Montero, A., Viales, R.,(2015).La teoría del cambio en el paisaje a partir del cambio del uso de la tierra y la cobertura del suelo (enfoque lucc). Su utilidad para la historia ambiental. *Reflexiones*, 94(2), 25-33. <https://dialnet.unirioja.es/servlet/articulo?codigo=5585652>

- Orimoloye, I.R., Kalumba, A., Mazinyo, S.P., Nel, W. 2018. Geospatial analysis of wetland dynamics: wetland depletion and biodiversity conservation of Isimangaliso Wetland, South Africa. *Journal of King Saud University - Science*. (in press). <https://doi.org/10.1016/j.jksus.2018.03.004>
- Ortega, D., Cantos, J. (2016). SIG con software libre: Quantum GIS, la opción más conocida. *Boletín de la Asociación de Geógrafos Españoles*, (71), 447-463. Recuperado de <https://dialnet.unirioja.es/servlet/articulo?codigo=5627793>
- Pabón Salazar, E. D. (2022). Análisis de los procesos de cambio de uso y cobertura del suelo en la microcuenca del río Tabacay [Tesis de Maestría, Universidad Andina Simón Bolívar, Sede Ecuador]. <https://repositorio.uasb.edu.ec/handle/10644/8687>
- Pastrana, O. (2021). Análisis de cambio de uso de suelo mediante percepción remota en el municipio de Valle de Santiago. 2021, marzo 11, de Centro de Investigación en geografía y geomática Ing. Jorge I. Tamayo A.C. CentroGeo Recuperado de <https://centrogeo.repositorioinstitucional.mx/jspui/bitstream/1012/41/1/21-2011-Tesis-Pineda%20Pastrana%2C%20Oliva-Maestra%20en%20Geom%C3%A1tica.pdf>
- Pauta Calle, G., & Chang Gómez, J. (2016). Índices de calidad del agua de fuentes superficiales y aspectos toxicológicos, evaluación del Río Burgay. *Maskana*, 5, 165–176. Retrieved from <https://publicaciones.ucuenca.edu.ec/ojs/index.php/maskana/article/view/564>
- Pelletier, C., Valero, S., Inglada, J., Champion, N., & Dedieu, G. (2016). Assessing the robustness of Random Forests to map land cover with high resolution satellite image time series over large areas. *Remote Sensing of Environment*, 187, 156-168. <https://www.sciencedirect.com/science/article/pii/S0034425716303820>
- Perilla, G., Mas, J. (2020). Google Earth Engine (GEE): una poderosa herramienta que vincula el potencial de los datos masivos y la eficacia del procesamiento en la nube. versión On-line ISSN 2448-7279 versión impresa ISSN 0188-4611. https://www.scielo.org.mx/scielo.php?pid=S0188-46112020000100111&script=sci_arttext

- Phan, T., Kuch, V., Lehnert, L. (2020). Land Cover Classification using Google Earth Engine and Random Forest Classifier—The Role of Image Composition. *Remote Sens.* 12, 2411. <https://doi.org/10.3390/rs12152411>
- Picone, N. (2017). Comparación de imágenes satelitales Sentinel 2 y Landsat 8 en el estudio de áreas urbanas. In Congreso Nacional de Geografía de Universidades Nacionales, At Resistencia, Chaco, Argentina. https://www.researchgate.net/profile/Natasha-Picone/publication/320705706_COMPARACION_DE_IMAGENES_SATELITALES_SENTINEL_2_Y_LANDSAT_8_EN_EL_ESTUDIO_DE_AREAS_URBANAS/links/59f658e80f7e9b553ebd29c1/COMPARACION-DE-IMAGENES-SATELITALES-Sentinel-2-Y-Landsat-8-EN-EL-ESTUDIO-DE-AREAS-URBANAS.pdf
- Poveda, Y., Bermudez, M., Gil, P.(2022)Evaluación de métodos de clasificación supervisada para la estimación de cambios espacio-temporales de cobertura en los páramos de Merchán y Telecom, Cordillera Oriental de Colombia. *Revista Boletín de Geología*, 44(2), 51–72. <https://doi.org/10.18273/revbol.v44n2-2022002>
- Ponce, B. d. (2010). Detección y análisis del cambio de uso del suelo en la zona centro - norte de la provincia de Capitán Prat, XI Región período 1984-2003. [Trabajo de titulación, Universidad Austral de Chile, Facultad de Ciencias Forestales]. <http://cybertesis.uach.cl/tesis/uach/2010/fifp792d/doc/fifp792d.pdf>
- Pontius Jr, R. G., Shusas, E. & McEachern, M. (2004). Detecting important categorical land changes while accounting for persistence. *Agriculture, Ecosystems Environment* 101(2):251-268. <https://www.sciencedirect.com/science/article/pii/S016788090300327X>
- Potapov et al. (2012) .Quantifying forest cover loss in Democratic Republic of the Congo, 2000–2010, with Landsat ETM+ data. <https://www.sciencedirect.com/science/article/pii/S0034425712000430>
- Richards, J. (2006). Análisis de imágenes digitales de teledetección. The Australian National University Canberra.
- Roy, DP; Kovalskyy, V.; Zhang, Hong Kong; Vermote, EF; Yan, L.; Kumar, SS; Egorov, A. (2016). Caracterización de la longitud de onda reflectante de Landsat-7 a

Landsat-8 y la continuidad del índice de vegetación de diferencia normalizada. Sensores Remotos Entorno. , 185 , 57–70. <https://www.sciencedirect.com/science/article/pii/S0034425715302455>

Rosero, E. (2021). Estudio multitemporal de uso del suelo y cobertura vegetal de la subcuenca del río Ambi, provincia Imbabura, Ecuador. [Tesis de maestría, Universidad Técnica del Norte, Gestión Sustentable de Recursos Naturales]. <https://repositorio.utn.edu.ec/handle/123456789/10902>

Rodriguez, O., Arredondo, H.(2005). Manual para el manejo y procesamiento de imágenes satelitales obtenidas del sensor remoto Modis de la Nasa, aplicado en estudios de ingeniería civil. [Trabajo de titulación, Pontificia Universidad Javeriana, Facultad de Ingeniería Civil]. <https://repository.javeriana.edu.co/bitstream/handle/10554/7050/tesis123.pdf?sequence=3>

Rodriguez-Galiano, V., Ghimireb, B. Roganb, J., Chica-Olmo, M., Rigol-Samchez, J. (2012). Una evaluación de la eficacia de un clasificador forestal aleatorio para la clasificación de la cobertura terrestre. Revista ISPRS de fotogrametría y teledetección. Volumen 67, enero de 2012 , páginas 93-104. <https://www.scirp.org/reference/referencespapers?referenceid=2865907>

Ruíz, A. (2017). Detección de objetos urbanos en imágenes satelitales, fuente de actualización de Sistemas de Información Geográfica. [Trabajo de grado, Universidad Politécnica de Madrid, Ingeniería Informática]. https://oa.upm.es/47197/1/TFG_ALVARO_RUIZ_MARONO.pdf

Saldaña Saldaña, V. A. (2019). Análisis de los cambios de cobertura y uso del suelo con imágenes satelitales del distrito de Chancay años 2001 y 2018.[Trabajo de grado, Universidad Nacional de Cajamarca, Escuela de Agronomía]. <https://repositorio.unc.edu.pe/bitstream/handle/20.500.14074/2990/AN%c3%81LISIS%20DE%20LOS%20CAMBIOS%20DE%20COBERTURA%20Y%20USO%20DEL%20SUELO%20CON%20IM%c3%81GENES%20SATELITALES%20DEL%20DISTRITO%20DE%20CHA.pdf?sequence=1&isAllowed=y>

Sandoval, S., Escobar-Flores, J. G., & Sánchez-Ortíz, E. (2020). Inventario de cuerpos de agua de la Sierra Madre Occidental (México) usando SIG y percepción remota. Investigaciones geográficas, (102).

https://www.scielo.org.mx/scielo.php?pid=S0188-46112020000200111&script=sci_arttext

Schwarz, C. J., & Roy, R. (2019, March). The systematics of Mantodea revisited: an updated classification incorporating multiple data sources (Insecta: Dictyoptera). In *Annales de la Société entomologique de France (NS)* (Vol. 55, No. 2, pp. 101-196). Taylor & Francis. https://www.researchgate.net/publication/332032683_The_systematics_of_Mantodea_revisited_an_updated_classification_incorporating_multiple_data_sources_Insecta_Dictyoptera

Simeone, O. (2018). A Very Brief Introduction to Machine Learning With Applications to Communication Systems. in *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 4, pp. 648-664, Dec. 2018, doi: 10.1109/TCCN.2018.2881442. keywords: {Machine learning;Mathematical model;Machine learning algorithms;Unsupervised learning;Training;Task analysis;Decoding;Machine learning;supervised learning;unsupervised learning;communication networks;wireless communications},

Tian, S., Guo, H., Xu, W., Zhu, X., Wang, B., Zeng, Q., ... & Huang, J. J. (2023). Remote sensing retrieval of inland water quality parameters using Sentinel-2 and multiple machine learning algorithms. *Environmental Science and Pollution Research*, 30(7), 18617-18630. <https://link.springer.com/article/10.1007/s11356-022-23431-9>

Timbe, L., Timbe, E. (2012). Mapeo del peligro de inundación en ríos de montaña, caso de estudio del río Burgay. *Maskana*, 3(1), 87-96. <https://doi.org/10.18537/mskn.03.01.0>

Toming, K., Kutser, T., Laas, A., Sepp, M., Paavel, B., & Nõges, T. (2016). First experiences in mapping lake water quality parameters with Sentinel-2 MSI imagery. *Remote Sensing*, 8(8), 640. <https://www.mdpi.com/2072-4292/8/8/640>

Tsutsumida, N., & Comber, A. J. (2015). Measures of spatio-temporal accuracy for time series land cover data. *International Journal of Applied Earth Observation and Geoinformation*, 41, 46-55. <https://www.sciencedirect.com/science/article/pii/S0303243415000975>

- Uribe, E. (2015). El cambio climático y sus efectos en la biodiversidad en América Latina. Euroclima.
<https://bibliotecadigital.ciren.cl/items/d969ecac-e7df-439a-8afb-6ce9cd3f5b93>
- USGS.(2013). Department of the Interior. U.S. Geological Survey. Recuperado de:
<https://www.un-spider.org/sites/default/files/LDCM-L8.R1.pdf>
- USGS.(2023). Department of the Interior. U.S. Geological Survey. Recuperado de:
<https://www.un-spider.org/sites/default/files/LDCM-L8.R1.pdf>
- Vaca, M. (2019). Metodologías para la evaluación de la influencia de actividades agropecuarias en la calidad del agua. [Tesis de maestría, Pontificia Universidad Católica del Ecuador].
<https://repositorio.puce.edu.ec/server/api/core/bitstreams/15481fa2-8398-44ee-896d-8bff35d443f8/content>
- Veneros, J., García, L., Morales, E., Gomez, V., Torres, M., Lopez, F. (2020). Aplicación de sensores remotos para el análisis de cobertura vegetal y cuerpos de agua. Chile: Idesia vol.38 no.4 Arica dic. 2020.
https://www.scielo.cl/scielo.php?pid=S0718-34292020000400099&script=sci_arttext
- Waske, B., Braun, M. (2019). Classifier ensembles for land cover mapping using multitemporal SAR imagery. Volume 64, Issue 5, September 2009, Pages 450-457. <https://www.sciencedirect.com/science/article/pii/S0924271609000070>
- Xia, J.; Falco, N.; Benediktsson, J.A.; Du, P.; Chanussot, J. (2018). Hyperspectral Image Classification With Rotation Random Forest Via KPCA. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens., 10, 1601–1609.
<https://ieeexplore.ieee.org/abstract/document/7805181>
- Xu, F., Li, Z., Zhang, S., Huang, N., Quan, Z., Zhang, W., ... & Prishchepov, A. V. (2020). Mapping winter wheat with combinations of temporally aggregated Sentinel-2 and Landsat-8 data in Shandong Province, China. Remote Sensing, 12(12), 2065. <https://www.mdpi.com/2072-4292/12/12/2065>
- Xu, Y., Yu, H., Zhong, J., Lin, T., & Chen, S. (2012). Real-time image capturing and processing of seam and pool during robotic welding process. Industrial Robot: An International Journal, 39(5), 513-523.

<https://www.emerald.com/insight/content/doi/10.1108/01439911211249805/full/html>

Young, N. E., Anderson, R. S., Chignell, S. M., Vorster, A. G., Lawrence, R., & Evangelista, P. H. (2017). A survival guide to Landsat preprocessing. *Ecology*, 98(4), 920-932.
<https://esajournals.onlinelibrary.wiley.com/doi/full/10.1002/ecy.1730>

Zambrano, E. (2021). Análisis espacio temporal de la cobertura vegetal y uso del suelo, y fragmentación del paisaje en el cantón Piñas, provincia de El Oro.[Tesis de grado, Universidad Nacional De Loja, Ingeniería Forestal].
https://dspace.unl.edu.ec/jspui/bitstream/123456789/24279/1/EdinsonGabrie_%20ZambranoFeijoo.pdf

Zhang, J., Yang, G., Yang, L., Li, Z., Gao, M., Yu, C., ... & Hu, H. (2022). Dynamic monitoring of environmental quality in the Loess Plateau from 2000 to 2020 using the Google Earth Engine Platform and the Remote Sensing Ecological index. *Remote Sensing*, 14(20), 5094.
<https://www.mdpi.com/2072-4292/14/20/5094>

Zhu, Z., & Woodcock, C. E. (2012). Object-based cloud and cloud shadow detection in Landsat imagery. *Remote sensing of environment*, 118, 83-94.
<https://www.sciencedirect.com/science/article/pii/S0034425711003853>

Zurqani, H. A., Post, C. J., Mikhailova, E. A., Schlautman, M. A., & Sharp, J. L. (2018). Geospatial analysis of land use change in the Savannah River Basin using Google Earth Engine. *International journal of applied earth observation and geoinformation*, 69, 175-185.
<https://www.sciencedirect.com/science/article/pii/S0303243417302957>

Anexos

Anexo A

Codificación Landsat-8 2016

```

1 var bounds=ee.FeatureCollection(["users/MAGUS/SUBCUENCA_BURGAY_Q"]);
2
3 // This example demonstrates the use of the Landsat 8 QA band to mask clouds.
4
5 // Function to mask clouds using the quality band of Landsat 8.
6 var maskLS = function(image) {
7   var qa = image.select("BQA");
8   /// Check that the cloud bit is off.
9   /// See https://www.usgs.gov/land-resources/nli/landsat/landsat-collection-1-level-1-quality-assessment-band
10  var mask = qa.bitwiseAnd(1 << 4).eq(0);
11  return image.updateMask(mask);
12 }
13
14 // Map the function over one year of Landsat 8 TOA data and take the median.
15 var composite = ee.ImageCollection("LANDSAT/LC08/C01/T1_TOA")
16   .filterDate('2016-01-01', '2016-12-31')
17   .filterBounds(bounds)
18   .map(maskLS)
19   .median();
20
21 // Get total number of pixels
22 var areaTotal = composite.reduceRegion({
23   reducer: ee.Reducer.count(),
24   geometry: bounds,
25   scale: 30, // Adjust the scale according to your needs
26   maxPixels: 1e13 // Adjust as needed
27 });
28
29 // Extract the total number of pixels from the property
30 var totalPixels = ee.Number(areaTotal.get("86")).getInfo(); // Adjust the band according to your needs
31
32 print("Número total de píxeles:", totalPixels);
33
34
35 // Display the results in a cloudy place.
36 Map.setCenter(-78.8666, -2.7263, 12);
37 Map.addLayer(composite.clip( bounds), {bands: ['86', '85', '84'], max: 0.8}, "Imagen de agricultura");
38
39 var muestras = Agua.merge(Bosque).merge(Cultivo).merge(Urbano).merge(Paramo);
40 var entrenamiento = composite.sampleRegions(muestras, ["Clase"], 30);
41
42
43 var randomcolumn = entrenamiento.randomColumn();
44
45 var trainingsample = randomcolumn.filter(ee.Filter.gte("random", 0.3));
46 var validationssample = randomcolumn.filter(ee.Filter.lte("random", 0.3));
47
48
49 print(entrenamiento.limit(1))
50 print(entrenamiento.size())
51
52
53 var bandas = ['82', '83', '84', '85', '86', '87'];
54 var clasificacion = ee.Classifier.smileRandomForest(10).train(entrenamiento, "Clase", bandas);
55 var imagen = composite.classify(clasificacion);
56
57
58
59 var landcoverPalette = [
60   "80C2C8", // Agua (0)
61   "8e3181c", // Urbano (1)
62   "8005e32", // Bosque (2)
63   "FF8000", // Cultivo (3)
64   "8829121", // Paramo (4)
65
66 ];
67
68
69 Map.addLayer(imagen.clip( bounds), {palette:landcoverPalette, min:0 , max: 4}, "classification");
70

```

```

71
72 //Crear conjuntos de entrenamiento y validación:
73
74
75 var randomcolumn = entrenamiento.randomColumn();
76 var trainingsample = randomcolumn.filter(ee.Filter.gte("random", 0.3));
77 var validationssample = randomcolumn.filter(ee.Filter.lte("random", 0.3));
78 //Entrenar el clasificador y realizar la clasificación:
79
80
81 var bandas = ['B2', 'B3', 'B4', 'B5', 'B6', 'B7'];
82 var clasificación = ee.Classifier.smileRandomForest(10).train(trainingsample, "Clase", bandas);
83 var imagen = composite.classify(clasificación);
84 //Visualizar la clasificación:
85
86
87 Map.addLayer(imagen.clip(bounds), { palette: landcoverPalette, min: 0, max: 4 }, "classification");
88 //Realizar la clasificación en el conjunto de validación:
89
90
91 var classifiedValidation = validationssample.classify(clasificación);
92 //Obtener la matriz de confusión:
93
94
95 var errorMatrix = classifiedValidation.errorMatrix({
96   actual: "Clase",
97   predicted: "classification"
98 });
99
100 print(errorMatrix, "Matriz de Confusión");
101 //Calcular la precisión global y las precisiones del usuario y del productor:
102
103
104 var overallAccuracy = errorMatrix.accuracy();
105 print("Precisión Global:", overallAccuracy);
106
107 var userAccuracy = errorMatrix.consumersAccuracy();
108 print("Precisión del Usuario:", userAccuracy);
109
110 var producerAccuracy = errorMatrix.producersAccuracy();
111 print("Precisión del Productor:", producerAccuracy);
112
113
114 // Calcular el coeficiente Kappa
115 var kappa = errorMatrix.kappa();
116 print("Coeficiente Kappa:", kappa);
117
118
119 // Crear una imagen con la verdad básica y la clasificación en cada pixel
120 var classifiedImage = imagen.select('classification');
121 var truthImage = validationssample.reduceToImage(["Clase"], ee.Reducer.first());
122
123 // Concatenar las imágenes de verdad básica y clasificación
124 var combinedImage = ee.Image.cat([truthImage, classifiedImage]);
125
126 // Reducir la imagen a través de las regiones de interés (ROI)
127 var confusionMatrix = combinedImage.reduceRegion({
128   reducer: ee.Reducer.fixedHistogram(0, 4, 5), // 0-4 son las clases posibles, ajusta según sea necesario
129   geometry: bounds,
130   scale: 30, // Ajustar según sea necesario
131   maxPixels: 1e13 // Ajustar según sea necesario
132 });
133
134 // Extraer la matriz de confusión
135 var matrix = confusionMatrix.get('classification');
136
137 // Mostrar la matriz de confusión
138 print("Matriz de Confusión", matrix);
139
140
141 var Burgay= imagen.clip(bounds);

```

```

142
143 Map.addLayer(Burgay,{palette:landcoverPalette, min:0 , max: 4}, "Burgay_2016");
144
145 var shapeBurgay = Burgay.reduceToVectors(
146 {geometry:bounds,
147 //sentinel a 10 o landsat a 30
148 scale:30 ,
149 geometryType: 'polygon',
150 maxPixels:10e13,
151 labelProperty: 'clase',
152 eightConnected:true,
153 });
154
155
156 var claseAgua=shapeBurgay.filter(ee.Filter.eq("clase",0));
157 var claseUrbano=shapeBurgay.filter(ee.Filter.eq("clase",1));
158 var claseBosque=shapeBurgay.filter(ee.Filter.eq("clase",2));
159 var claseCultivo=shapeBurgay.filter(ee.Filter.eq("clase",3));
160 var claseParamo=shapeBurgay.filter(ee.Filter.eq("clase",4));
161
162 var areaAgua=claseAgua.geometry().area().divide(1e4).getInfo();
163 var areaUrbano= claseUrbano.geometry().area().divide(1e4).getInfo();
164 var areaBosque=claseBosque.geometry().area().divide(1e4).getInfo();
165 var areaCultivo=claseCultivo.geometry().area().divide(1e4).getInfo();
166 var areaParamo=claseParamo.geometry().area().divide(1e4).getInfo();
167 var suma=areaAgua+areaUrbano+areaBosque+areaCultivo+areaParamo;
168
169 print("clasificación supervisada")
170 print("clase Agua",areaAgua);
171 print("clase Urbano",areaUrbano);
172 print("clase Bosque",areaBosque);
173 print("clase Cultivo",areaCultivo);
174 print("clase Paramo", areaParamo);
175 print("suma",suma);
176 print("area Interes",bounds.geometry().area().divide(1e4));
177
178 var porcentajeAgua=(ee.Number(areaAgua).divide(ee.Number(suma)).multiply(100)).getInfo();
179 var porcentajeUrbano=(ee.Number(areaUrbano).divide(ee.Number(suma)).multiply(100)).getInfo();
180
181 var porcentajeBosque=(ee.Number(areaBosque).divide(ee.Number(suma)).multiply(100)).getInfo();
182 var porcentajeCultivo=(ee.Number(areaCultivo).divide(ee.Number(suma)).multiply(100)).getInfo();
183 var porcentajeParamo=(ee.Number(areaParamo).divide(ee.Number(suma)).multiply(100)).getInfo();
184
185 print("porcentaje agua",porcentajeAgua);
186 print("porcentaje urbano", porcentajeUrbano);
187 print("porcentaje bosque",porcentajeBosque);
188 print("porcentaje cultivo",porcentajeCultivo);
189 print("porcentaje Paramo", porcentajeParamo);
190
191 var SumaPorcentaje= porcentajeAgua+porcentajeUrbano+ porcentajeBosque+ porcentajeCultivo+porcentajeParamo;
192
193 print("SumaPorcentaje:",SumaPorcentaje);
194
195
196
197 Export.image.toDrive({
198 image:imagen.clip(bounds),
199 description:"Burgay_2016",
200 scale:30,
201 region:table
202 });
203
204

```



```

1  var areaInteres=ee.FeatureCollection("users/MAGUS/SUBCUENCA_BURGAY_0");
2
3
4  Map.addLayer(areaInteres);
5  //Eliminar nubes y sombras
6  var cloud_masks = require('users/fitoprincipe/geotools:cloud_masks');
7  // "shadow"
8  var options = ["cloud"];
9  var landsatTOAcloudmask = cloud_masks.landsatTOA (options);
10
11  var l5coll = ee.ImageCollection("LANDSAT/LT05/C01/T1_TOA")
12    .filterBounds(areaInteres)
13    ;
14
15  var l8coll = ee.ImageCollection('LANDSAT/LC08/C01/T1_TOA')
16    .filterBounds(areaInteres)
17  .map(function(image){
18    return image.rename(['B0', 'B1', 'B2', 'B3', 'B4', 'B5', 'B6', 'B7', 'B8', 'B9', 'B10', 'BQA']);
19  });
20
21
22  var landsatCollection = ee.ImageCollection(l5coll.merge(l8coll));
23
24  //más seca con pocas lluvias desde junio a septiembre 2016-06-01", "2016-09-30
25  //2016-10-01", "2016-12-30
26  //
27  var collection1=landsatCollection
28    .filterDate("2016-01-01", "2016-12-30")
29    .filterMetadata("CLOUD_COVER", "less_than",100)
30    .map(landsatTOAcloudmask)
31  .map( function (imagen){
32
33    // CALCULO DE INDICES
34    var ndvi = imagen.normalizedDifference(['B4',"B3"]).rename("NDVI");
35    var ndwi = imagen.normalizedDifference(['B2',"B4"]).rename("NDWI");
36    var ndbi = imagen.normalizedDifference(['B5',"B4"]).rename("NDBI");
37    imagen= imagen.addBands([ndbi,ndvi,ndwi]);
38
39    return imagen.clip(areaInteres);
40  });
41
42
43  var mediana=collection1.median();
44
45
46  var medianaNDVI= mediana.select('NDVI');
47  var medianaNDWI= mediana.select('NDWI');
48  var medianaNDBI= mediana.select('NDBI');
49
50
51
52  var agua= medianaNDWI.gte(0);
53  var vegetacion = medianaNDVI.gte(0.2).and( medianaNDVI.lte(0.43));
54  var vegetacionVigorosa=medianaNDVI.gt(0.43);
55  var construccion=medianaNDBI.gt(0.05);
56  //var sueloDesnudo = medianaNDVI.lte(0.2).and(construccion.not());
57  var sueloDesnudo = medianaNDVI.lte(0.2);
58
59
60
61  //Mascaras
62  var mascara_sueloDesnudo= sueloDesnudo
63    .updateMask(sueloDesnudo);
64  var mascara_agua=agua
65    .updateMask(agua);
66  var mascara_vegetacion=vegetacion
67    .updateMask(vegetacion);
68
69  var mascara_vegetacionVigorosa= vegetacionVigorosa
70    .updateMask( vegetacionVigorosa);
71  var mascara_construccion= construccion
72    .updateMask( construccion);
73

```

María Augusta Murillo Patiño - María Camila Obando Loja

Anexo B

Codificación Landsat-8 2018

```

1  var bounds=ee.FeatureCollection("users/MAGUS/SUBCUENCA_BURGAY_0");
2
3  // This example demonstrates the use of the Landsat 8 QA band to mask clouds.
4
5  // Function to mask clouds using the quality band of Landsat 8.
6  var maskL8 = function(image) {
7    var qa = image.select('BQA');
8    // Check that the cloud bit is off.
9    // See https://www.usgs.gov/land-resources/nli/landsat/landsat-collection-1-level-1-quality-assessment-band
10   var mask = qa.bitwiseAnd(1 << 4).eq(0);
11   return image.updateMask(mask);
12 }
13
14 // Map the function over one year of Landsat 8 TOA data and take the median.
15 var composite = ee.ImageCollection("LANDSAT/LC08/C01/T1_TOA")
16   .filterDate('2018-01-01', '2018-12-31')
17   .filterBounds(bounds)
18   .map(maskL8)
19   .median();
20
21 // Get total number of pixels
22 var areaTotal = composite.reduceRegion({
23   reducer: ee.Reducer.count(),
24   geometry: bounds,
25   scale: 30, // Adjust the scale according to your needs
26   maxPixels: 1e13 // Adjust as needed
27 });
28
29 // Extract the total number of pixels from the property
30 var totalPíxeles = ee.Number(areaTotal.get("B6")).getInfo(); // Adjust the band according to your needs
31
32 print("Número total de píxeles:", totalPíxeles);
33
34
35 // Display the results in a cloudy place.
36 Map.setCenter(-78.8666, -2.7263, 12);
37 Map.addLayer(composite.clip(bounds), {bands: ['B6', 'B5', 'B4'], max: 0.8}, "Imagen de agricultura");
38
39
40 var muestras = Agua.merge(Bosque).merge(Cultivo).merge(Urbano).merge(Paramo);
41 var entrenamiento = composite.sampleRegions(muestras, ["Clase"], 30);
42
43
44 var randomcolumn = entrenamiento.randomColumn();
45
46 var trainingsample = randomcolumn.filter(ee.Filter.gte("random", 0.3));
47 var validationsample = randomcolumn.filter(ee.Filter.lte("random", 0.3));
48
49
50 print(entrenamiento.limit(1))
51 print(entrenamiento.size())
52
53 var bandas = ['B2', 'B3', 'B4', 'B5', 'B6', 'B7'];
54 var clasificacion = ee.Classifier.smileRandomForest(10).train(entrenamiento, "Clase", bandas);
55 var imagen = composite.classify(clasificacion);
56
57
58
59 var landcoverPalette = [
60   "#0C2C84", // Agua (0)
61   "#e31a1c", // Urbano (1)
62   "#005a32", // Bosque (2)
63   "#ff8000", // Cultivo (3)
64   "#829121", // Paramo (4)
65
66
67
68 ];

```

```

69
70 Map.addLayer(imagen.clip( bounds),{palette:landcoverPalette, min:0 , max: 4}, "classification");
71
72 //Crear conjuntos de entrenamiento y validación:
73
74
75 var randomcolumn = entrenamiento.randomColumn();
76 var trainingsample = randomcolumn.filter(ee.Filter.gte("random", 0.3));
77 var validationsample = randomcolumn.filter(ee.Filter.lte("random", 0.3));
78 //Entrenar el clasificador y realizar la clasificación:
79
80
81 var bandas = ['B2', 'B3', 'B4', 'B5', 'B6', 'B7'];
82 var clasificacion = ee.Classifier.smileRandomForest(10).train(trainingsample, "Clase", bandas);
83 var imagen = composite.classify(clasificacion);
84 //Visualizar la clasificación:
85
86
87 Map.addLayer(imagen.clip(bounds), { palette: landcoverPalette, min: 0, max: 4 }, "classification");
88 //Realizar la clasificación en el conjunto de validación:
89
90
91 var classifiedValidation = validationsample.classify(clasificacion);
92 //Obtener la matriz de confusión:
93
94
95 var errorMatrix = classifiedValidation.errorMatrix({
96   actual: "Clase",
97   predicted: "classification"
98 });
99
100 print(errorMatrix, "Matriz de Confusión");
101 //Calcular la precisión global y las precisiones del usuario y del productor:
102
103
104 var overallAccuracy = errorMatrix.accuracy();
105 print("Precisión Global:", overallAccuracy);
106
107 var userAccuracy = errorMatrix.consumersAccuracy();
108 print("Precisión del Usuario:", userAccuracy);
109
110 var producerAccuracy = errorMatrix.producersAccuracy();
111 print("Precisión del Productor:", producerAccuracy);
112
113
114 // Calcular el coeficiente Kappa
115 var kappa = errorMatrix.kappa();
116 print("Coeficiente Kappa:", kappa);
117
118
119 // Crear una imagen con la verdad básica y la clasificación en cada píxel
120 var classifiedImage = imagen.select('classification');
121 var truthImage = validationsample.reduceToImage(["Clase"], ee.Reducer.first());
122
123 // Concatenar las imágenes de verdad básica y clasificación
124 var combinedImage = ee.Image.cat([truthImage, classifiedImage]);
125
126 // Reducir la imagen a través de las regiones de interés (ROI)
127 var confusionMatrix = combinedImage.reduceRegion({
128   reducer: ee.Reducer.fixedHistogram(0, 4, 5), // 0-4 son las clases posibles, ajusta según sea necesario
129   geometry: bounds,
130   scale: 30, // Ajustar según sea necesario
131   maxPixels: 1e13 // Ajustar según sea necesario
132 });
133
134 // Extraer la matriz de confusión
135 var matrix = confusionMatrix.get('classification');
136
137 // Mostrar la matriz de confusión
138 print("Matriz de Confusión", matrix);
139

```

```

140
141 var Burgay= imagen.clip(bounds);
142
143 Map.addLayer(Burgay,{palette:landcoverPalette, min:0 , max: 4}, "Burgay_2018");
144
145 var shapeBurgay = Burgay.reduceToVectors(
146 * {geometry:bounds,
147   //sentinel a 10 o landsat a 30
148   scale:30 ,
149   geometryType: 'polygon',
150   maxPixels:10e13,
151   labelProperty: 'clase',
152   eightConnected:true,
153   });
154
155
156 var claseAgua=shapeBurgay.filter(ee.Filter.eq("clase",0));
157 var claseUrbano=shapeBurgay.filter(ee.Filter.eq("clase",1));
158 var claseBosque=shapeBurgay.filter(ee.Filter.eq("clase",2));
159 var claseCultivo=shapeBurgay.filter(ee.Filter.eq("clase",3));
160 var claseParamo=shapeBurgay.filter(ee.Filter.eq("clase",4));
161
162 var areaAgua=claseAgua.geometry().area().divide(1e4).getInfo();
163 var areaUrbano= claseUrbano.geometry().area().divide(1e4).getInfo();
164 var areaBosque=claseBosque.geometry().area().divide(1e4).getInfo();
165 var areaCultivo=claseCultivo.geometry().area().divide(1e4).getInfo();
166 var areaParamo=claseParamo.geometry().area().divide(1e4).getInfo();
167 var suma=areaAgua+areaUrbano+areaBosque+areaCultivo+areaParamo;
168
169 print("clasificación supervisada")
170 print("clase Agua",areaAgua);
171 print("clase Urbano",areaUrbano);
172 print("clase Bosque",areaBosque);
173 print("clase Cultivo",areaCultivo);
174 print("clase Paramo", areaParamo);
175 print("suma",suma);
176 print("area Interes",bounds.geometry().area().divide(1e4));
177
178 var porcentajeAgua=(ee.Number(areaAgua).divide(ee.Number(suma)).multiply(100)).getInfo();
179 var porcentajeUrbano=(ee.Number(areaUrbano).divide(ee.Number(suma)).multiply(100)).getInfo();
180
181 var porcentajeBosque=(ee.Number(areaBosque).divide(ee.Number(suma)).multiply(100)).getInfo();
182 var porcentajeCultivo=(ee.Number(areaCultivo).divide(ee.Number(suma)).multiply(100)).getInfo();
183 var porcentajeParamo=(ee.Number(areaParamo).divide(ee.Number(suma)).multiply(100)).getInfo();
184
185 print("porcentaje agua",porcentajeAgua);
186 print("porcentaje urbano", porcentajeUrbano);
187 print("porcentaje bosque",porcentajeBosque);
188 print("porcentaje cultivo",porcentajeCultivo);
189 print("porcentaje Paramo", porcentajeParamo);
190
191 var SumaPorcentaje= porcentajeAgua+porcentajeUrbano+ porcentajeBosque+ porcentajeCultivo+porcentajeParamo;
192
193 print("SumaPorcentaje:",SumaPorcentaje);
194
195
196
197 * Export.image.toDrive({
198   image:imagen.clip(bounds),
199   description:"Burgay_2018",
200   scale:30,
201   region:table
202 });

```



```

1  var areaInteres=ee.FeatureCollection("users/MAGUS/SUBCUENCA_BURGAY_0");
2
3
4  Map.addLayer(areaInteres);
5  //Eliminar nubes y sombras
6  var cloud_masks = require('users/fitoprincipe/geetools:cloud_masks');
7  // "shadow"
8  var options = ["cloud"];
9  var landsatTOAcloudmask = cloud_masks.landsatTOA (options);
10
11  var L5coll = ee.ImageCollection("LANDSAT/LT05/C01/T1_TOA")
12    .filterBounds(areaInteres)
13    ;
14
15  var L8coll = ee.ImageCollection('LANDSAT/LC08/C01/T1_TOA')
16    .filterBounds(areaInteres)
17    .map(function(image){
18      return image.rename(['B0', 'B1', 'B2', 'B3', 'B4', 'B5', 'B6', 'B7', 'B8', 'B9', 'B10', 'BQA']);
19    });
20
21
22  var landsatCollection = ee.ImageCollection(L5coll.merge(L8coll));
23
24  //más seca con pocas lluvias desde junio a septiembre 2018-06-01", "2018-09-30
25  //2018-10-01", "2018-12-30
26  //
27  var collection1=landsatCollection
28    .filterDate("2018-01-01", "2018-12-30")
29    .filterMetadata("CLOUD_COVER", "less_than",100)
30    .map(landsatTOAcloudmask)
31    .map( function (imagen){
32
33      // CALCULO DE INDICES
34      var ndvi = imagen.normalizedDifference(['B4',"B3"]).rename("NDVI");
35      var ndwi = imagen.normalizedDifference(['B2',"B4"]).rename("NDWI");
36      var ndbi = imagen.normalizedDifference(['B5',"B4"]).rename("NDBI");
37      imagen= imagen.addBands([ndbi,ndvi,ndwi]);
38
39      return imagen.clip(areaInteres);
40    });
41
42
43  var mediana=collection1.median();
44
45
46  var medianaNDVI= mediana.select('NDVI');
47  var medianaNDWI= mediana.select('NDWI');
48  var medianaNDBI= mediana.select('NDBI');
49
50
51
52  var agua= medianaNDWI.gte(0);
53  var vegetacion = medianaNDVI.gte(0.2).and( medianaNDVI.lte(0.43));
54  var vegetacionVigorosa=medianaNDVI.gt(0.43);
55  var construccion=medianaNDBI.gt(0.05);
56  //var sueloDesnudo = medianaNDVI.lte(0.2).and(construccion.not());
57  var sueloDesnudo = medianaNDVI.lte(0.2);
58
59
60
61  //Mascaras
62  var mascara_sueloDesnudo= sueloDesnudo
63    .updateMask(sueloDesnudo);
64  var mascara_agua=agua
65    .updateMask(agua);
66  var mascara_vegetacion=vegetacion
67    .updateMask(vegetacion);
68
69  var mascara_vegetacionVigorosa= vegetacionVigorosa
70    .updateMask( vegetacionVigorosa);
71  var mascara_construccion= construccion
72    .updateMask( construccion);
73

```

María Augusta Murillo Patiño - María Camila Obando Loja

Anexo C

Codificación Landsat-8 2020

```

1  var bounds=ee.FeatureCollection("users/MAGUS/SUBCUENCA_BURGAY_0");
2
3  // This example demonstrates the use of the Landsat 8 QA band to mask clouds.
4
5  // Function to mask clouds using the quality band of Landsat 8.
6  var maskL8 = function(image) {
7    var qa = image.select('BQA');
8    /// Check that the cloud bit is off.
9    // See https://www.usgs.gov/land-resources/nli/landsat/landsat-collection-1-level-1-quality-assessment-band
10   var mask = qa.bitwiseAnd(1 << 4).eq(0);
11   return image.updateMask(mask);
12 }
13
14 // Map the function over one year of Landsat 8 TOA data and take the median.
15 var composite = ee.ImageCollection('LANDSAT/LC08/C01/T1_TOA')
16   .filterDate('2020-01-01', '2020-12-31')
17   .filterBounds(bounds)
18   .map(maskL8)
19   .median();
20
21 // Get total number of pixels
22 var areaTotal = composite.reduceRegion({
23   reducer: ee.Reducer.count(),
24   geometry: bounds,
25   scale: 30, // Adjust the scale according to your needs
26   maxPixels: 1e13 // Adjust as needed
27 });
28
29 // Extract the total number of pixels from the property
30 var totalPíxeles = ee.Number(areaTotal.get('B6')).getInfo(); // Adjust the band according to your needs
31
32 print("Número total de píxeles:", totalPíxeles);
33
34
35
36 // Display the results in a cloudy place.
37 Map.setCenter(-78.8666, -2.7263, 12);
38 Map.addLayer(composite.clip( bounds), {bands: ['B6', 'B5', 'B4'], max: 0.8}, "Imagen de agricultura");
39
40 var muestras = Agua.merge(Bosque).merge(Cultivo).merge(Urbano).merge(Paramo);
41 var entrenamiento = composite.sampleRegions(muestras, ["Clase"], 30);
42
43
44 var randomcolumn = entrenamiento.randomColumn();
45
46 var trainingsample = randomcolumn.filter(ee.Filter.gte("random", 0.3));
47 var validationssample = randomcolumn.filter(ee.Filter.lte("random", 0.3));
48
49
50 print(entrenamiento.limit(1))
51 print(entrenamiento.size())
52
53 var bandas = ['B2', 'B3', 'B4', 'B5', 'B6', 'B7'];
54 var clasificacion = ee.Classifier.smileRandomForest(10).train(entrenamiento, "Clase", bandas);
55 var imagen = composite.classify(clasificacion);
56
57
58
59 var landcoverPalette = [
60   "#000000", // Agua (0)
61   "#000000", // Urbano (1)
62   "#000000", // Bosque (2)
63   "#000000", // Cultivo (3)
64   "#000000", // Paramo (4)
65
66
67 ];
68
69
70 Map.addLayer(imagen.clip( bounds), {palette: landcoverPalette, min: 0, max: 4}, "classification");
71
72 //Crear conjuntos de entrenamiento y validación:
73
74
75 var randomcolumn = entrenamiento.randomColumn();
76 var trainingsample = randomcolumn.filter(ee.Filter.gte("random", 0.3));
77 var validationssample = randomcolumn.filter(ee.Filter.lte("random", 0.3));
78 //Entrenar el clasificador y realizar la clasificación:
79

```

```

80
81 var bandas = ['B2', 'B3', 'B4', 'B5', 'B6', 'B7'];
82 var clasificacion = ee.Classifier.smileRandomForest(10).train(trainingsample, "Clase", bandas);
83 var imagen = composite.classify(clasificacion);
84 //Visualizar la clasificación:
85
86
87 Map.addLayer(imagen.clip(bounds), { palette: landcoverPalette, min: 0, max: 4 }, "classification");
88 //Realizar la clasificación en el conjunto de validación:
89
90
91 var classifiedValidation = validationssample.classify(clasificacion);
92 //Obtener la matriz de confusión:
93
94
95 var errorMatrix = classifiedValidation.errorMatrix({
96   actual: "Clase",
97   predicted: "classification"
98 });
99
100 print(errorMatrix, "Matriz de Confusión");
101 //Calcular la precisión global y las precisiones del usuario y del productor:
102
103
104 var overallAccuracy = errorMatrix.accuracy();
105 print("Precisión Global:", overallAccuracy);
106
107 var userAccuracy = errorMatrix.consumersAccuracy();
108 print("Precisión del Usuario:", userAccuracy);
109
110 var producerAccuracy = errorMatrix.producersAccuracy();
111 print("Precisión del Productor:", producerAccuracy);
112
113
114 // Calcular el coeficiente Kappa
115 var kappa = errorMatrix.kappa();
116 print("Coeficiente Kappa:", kappa);
117
118
119 // Crear una imagen con la verdad básica y la clasificación en cada píxel
120 var classifiedImage = imagen.select('classification');
121 var truthImage = validationssample.reduceToImage(["Clase"], ee.Reducer.first());
122
123 // Concatenar las imágenes de verdad básica y clasificación
124 var combinedImage = ee.Image.cat([truthImage, classifiedImage]);
125
126 // Reducir la imagen a través de las regiones de interés (ROI)
127 var confusionMatrix = combinedImage.reduceRegion({
128   reducer: ee.Reducer.fixedHistogram(0, 4, 5), // 0-4 son las clases posibles, ajusta según sea necesario
129   geometry: bounds,
130   scale: 30, // Ajustar según sea necesario
131   maxPixels: 1e13 // Ajustar según sea necesario
132 });
133
134 // Extraer la matriz de confusión
135 var matrix = confusionMatrix.get('classification');
136
137 // Mostrar la matriz de confusión
138 print("Matriz de Confusión", matrix);
139
140
141 var Burgay = imagen.clip(bounds);
142
143 Map.addLayer(Burgay, {palette: landcoverPalette, min: 0, max: 4}, "Burgay_2020");
144
145 var shapeBurgay = Burgay.reduceToVectors(
146   {
147     geometry: bounds,
148     //sentinel a 10 o landsat a 30
149     scale: 30,
150     geometryType: 'polygon',
151     maxPixels: 10e13,
152     labelProperty: 'clase',
153     eightConnected: true,
154   });

```

```

154
155
156 var claseAgua=shapeBurgay.filter(ee.Filter.eq("clase",0));
157 var claseUrbano=shapeBurgay.filter(ee.Filter.eq("clase",1));
158 var claseBosque=shapeBurgay.filter(ee.Filter.eq("clase",2));
159 var claseCultivo=shapeBurgay.filter(ee.Filter.eq("clase",3));
160 var claseParamo=shapeBurgay.filter(ee.Filter.eq("clase",4));
161
162 var areaAgua=claseAgua.geometry().area(1).divide(1e4).getInfo();
163 var areaUrbano=claseUrbano.geometry().area(1).divide(1e4).getInfo();
164 var areaBosque=claseBosque.geometry().area(1).divide(1e4).getInfo();
165 var areaCultivo=claseCultivo.geometry().area(1).divide(1e4).getInfo();
166 var areaParamo=claseParamo.geometry().area(1).divide(1e4).getInfo();
167 var suma=areaAgua+areaUrbano+areaBosque+areaCultivo+areaParamo;
168
169 print("clasificación supervisada")
170 print("clase Agua",areaAgua);
171 print("clase Urbano",areaUrbano);
172 print("clase Bosque",areaBosque);
173 print("clase Cultivo",areaCultivo);
174 print("clase Paramo", areaParamo);
175 print("suma",suma);
176 print("area Interes",bounds.geometry().area().divide(1e4));
177
178 var porcentajeAgua=(ee.Number(areaAgua).divide(ee.Number(suma)).multiply(100)).getInfo();
179 var porcentajeUrbano=(ee.Number(areaUrbano).divide(ee.Number(suma)).multiply(100)).getInfo();
180
181 var porcentajeBosque=(ee.Number(areaBosque).divide(ee.Number(suma)).multiply(100)).getInfo();
182 var porcentajeCultivo=(ee.Number(areaCultivo).divide(ee.Number(suma)).multiply(100)).getInfo();
183 var porcentajeParamo=(ee.Number(areaParamo).divide(ee.Number(suma)).multiply(100)).getInfo();
184
185 print("porcentaje agua",porcentajeAgua);
186 print("porcentaje urbano", porcentajeUrbano);
187 print("porcentaje bosque",porcentajeBosque);
188 print("porcentaje cultivo",porcentajeCultivo);
189 print("porcentaje Paramo", porcentajeParamo);
190
191 var SumaPorcentaje= porcentajeAgua+porcentajeUrbano+ porcentajeBosque+ porcentajeCultivo+porcentajeParamo;
192
193 print("SumaPorcentaje:",SumaPorcentaje);
194
195
196
197 ~ Export.image.toDrive({
198   image:imagen.clip(bounds),
199   description:"Burgay_2020",
200   scale:30,
201   region:table
202 });
203
204

```

```

1  var areaInteres=ee.FeatureCollection("users/MAGUS/SUBCUENCA_BURGAY_0");
2
3
4  Map.addLayer(areaInteres);
5  //Eliminar nubes y sombras
6  var cloud_masks = require('users/fitoprincipe/geetools:cloud_masks');
7  // "shadow"
8  var options = ["cloud"];
9  var landsatTOAcloudmask = cloud_masks.landsatTOA (options);
10
11  var l5coll = ee.ImageCollection("LANDSAT/LT05/C01/T1_TOA")
12    .filterBounds(areaInteres)
13    ;
14
15  var l8coll = ee.ImageCollection('LANDSAT/LC08/C01/T1_TOA')
16    .filterBounds(areaInteres)
17    .map(function(image){
18      return image.rename(['B0', 'B1', 'B2', 'B3', 'B4', 'B5', 'B6', 'B7', 'B8', 'B9', 'B10', 'BQA']);
19    });
20
21
22  var landsatCollection = ee.ImageCollection(l5coll.merge(l8coll));
23
24  //más seca con pocas lluvias desde junio a septiembre 2020-06-01", "2020-09-30
25  //2020-10-01", "2020-12-30
26  //
27  var collection1=landsatCollection
28    .filterDate("2020-01-01", "2020-12-30")
29    .filterMetadata("CLOUD_COVER", "less_than",100)
30    .map(landsatTOAcloudmask)
31    .map(function (imagen){
32
33      // CALCULO DE INDICES
34      var ndvi = imagen.normalizedDifference(['B4','B3']).rename("NDVI");
35      var ndwi = imagen.normalizedDifference(['B2','B4']).rename("NDWI");
36      var ndbi = imagen.normalizedDifference(['B5','B4']).rename("NDBI");
37      imagen= imagen.addBands([ndbi,ndvi,ndwi]);
38
39      return imagen.clip(areaInteres);
40    });
41  });
42
43  var mediana=collection1.median();
44
45
46  var medianaNDVI= mediana.select('NDVI');
47  var medianaNDWI= mediana.select('NDWI');
48  var medianaNDBI= mediana.select('NDBI');
49
50
51
52  var agua= medianaNDWI.gte(0);
53  var vegetacion = medianaNDVI.gte(0.2).and( medianaNDVI.lte(0.43));
54  var vegetacionVigorosa=medianaNDVI.gt(0.43);
55  var construccion=medianaNDBI.gt(0.05);
56  //var sueloDesnudo = medianaNDVI.lte(0.2).and(construccion.not());
57  var sueloDesnudo = medianaNDVI.lte(0.2);
58
59
60
61  //Mascaras
62  var mascara_sueloDesnudo= sueloDesnudo
63    .updateMask(sueloDesnudo);
64  var mascara_agua=agua
65    .updateMask(agua);
66  var mascara_vegetacion=vegetacion
67    .updateMask(vegetacion);
68
69  var mascara_vegetacionVigorosa= vegetacionVigorosa
70    .updateMask( vegetacionVigorosa);
71  var mascara_construccion= construccion
72    .updateMask( construccion);
73

```

```

74 | | | |
75 | var mosaic = ee.ImageCollection
76 | ([mascara_sueloDesnudo.visualize({palette: ['804000']})),
77 |  mascara_agua.visualize({palette: ['blue']})),
78 |  mascara_vegetacion.visualize({palette: ['green']})),
79 |  mascara_vegetacionVigorosa.visualize({palette: ['yellow']})),
80 |  //mascara_construccion.visualize({palette: ['red']})),
81 |  ]).mosaic();
82 |
83 |
84 | var area = ee.Image.pixelArea().divide(1e4);
85 | Map.addLayer(mediana,{bands:['B4','B3','B2'],max:0.5},'Mediana',true);
86 | //Map.addLayer(slope,{},'slope',false);
87 | Map.addLayer(mosaic,{},'Mosaico');
88 | var mascara_sueloDesnudo_area = mascara_sueloDesnudo.multiply(area).select([0],['Suelo desnudo']);
89 | var mascara_agua_area = mascara_agua.multiply(area).select([0],['Agua']);
90 | var mascara_vegetacion_area = mascara_vegetacion.multiply(area).select([0],['Vegetacion']);
91 | var mascara_vegetacionVigorosa_area = mascara_vegetacionVigorosa.multiply(area).select([0],['vegetacion Vigorosa']);
92 |
93 |
94 | var area_image = mascara_sueloDesnudo_area.addBands(mascara_agua_area )
95 |                                     .addBands( mascara_vegetacion_area)
96 |                                     .addBands(mascara_vegetacionVigorosa_area );
97 | var areas = area_image.reduceRegion({
98 |   reducer:ee.Reducer.sum(),
99 |   geometry: areaInteres,
100 |   scale: 30,
101 |   maxPixels:1e13
102 | });
103 | print('hectareas',areas);
104 | var areaAgua=areas.get("Agua").getInfo();
105 | var areaSuelo=areas.get("Suelo desnudo").getInfo();
106 | var areaVegetacion=areas.get("Vegetacion").getInfo();
107 | var areaVegetacionVigorosa=areas.get("vegetacion Vigorosa").getInfo();
108 | var suma=areaAgua+ areaSuelo +areaVegetacion+areaVegetacionVigorosa;
109 | print("suma",suma);
110 | print("area Interes",areaInteres.geometry().area().divide(1e4));
111 | print("porcentaje agua",ee.Number(areaAgua).divide(ee.Number(suma)).multiply(100))
112 | print("porcentaje suelo",ee.Number(areaSuelo).divide(ee.Number(suma)).multiply(100))
113 | print("porcentaje vegetacion",ee.Number(areaVegetacion).divide(ee.Number(suma)).multiply(100))
114 | print("porcentaje vegetacion vigorosa",ee.Number( areaVegetacionVigorosa).divide(ee.Number(suma)).multiply(100))
115 |
116 | Map.centerObject(areaInteres,8)
117 |

```


Anexo D

Codificación Sentinel-2 2016

```

1 var bounds = ee.FeatureCollection("users/MAGUS/SUBCUENCA_BURGAY_O");
2
3 // Function to mask clouds using the Sentinel-2 QA band.
4 function maskS2clouds(image) {
5   var qa = image.select('QA60');
6
7   // Bits 10 and 11 are clouds and cirrus, respectively.
8   var cloudBitMask = 1 << 10;
9   var cirrusBitMask = 1 << 11;
10
11   // Both flags should be set to zero, indicating clear conditions.
12   var mask = qa.bitwiseAnd(cloudBitMask).eq(0).and(
13     qa.bitwiseAnd(cirrusBitMask).eq(0));
14
15   // Return the masked and scaled data, without the QA bands.
16   return image.updateMask(mask).divide(10000)
17     .select(['B2', 'B3', 'B4', 'B5', 'B6', 'B7', 'B8', 'B8A', 'B11', 'B12'])
18     .copyProperties(image, ['system:time_start']);
19 }
20
21 // Map the function over one year of data and take the median.
22 // Load Sentinel-2 TOA reflectance data.
23 var composite = ee.ImageCollection('COPERNICUS/S2')
24   .filterBounds(bounds)
25   .filterDate('2016-01-01', '2016-01-31')
26   // Pre-filter to get less cloudy granules.
27   .filter(ee.Filter.lt('CLOUDY_PIXEL_PERCENTAGE', 20))
28   .map(maskS2clouds)
29   .median();
30
31 // Get total number of pixels
32 var areaTotal = composite.reduceRegion({
33   reducer: ee.Reducer.count(),
34   geometry: bounds,
35   scale: 10, // Adjust the scale according to your needs
36   maxPixels: 1e13 // Adjust as needed
37 });
38
39 // Extract the total number of pixels from the property
40 var totalPíxeles = ee.Number(areaTotal.get("B2")).getInfo(); // Adjust the band according to your needs
41
42 print("Número total de píxeles:", totalPíxeles);
43
44 // Display the results in a cloudy place.
45 Map.setCenter(-78.8666, -2.7263, 12);
46 Map.addLayer(composite.clip(bounds), {bands: ['B4', 'B3', 'B2'], max: 0.8}, "Imagen de agricultura");
47
48 var muestras = Agua.merge(Bosque).merge(Cultivo).merge(Urbano).merge(Paramo);
49 var entrenamiento = composite.sampleRegions(muestras, ["Clase"], 30);
50
51 var randomcolumn = entrenamiento.randomColumn();
52
53 var trainingsample = randomcolumn.filter(ee.Filter.gte("random", 0.3));
54 var validationssample = randomcolumn.filter(ee.Filter.lte("random", 0.3));
55
56 print(entrenamiento.limit(1));
57 print(entrenamiento.size());
58
59 var bandas = ['B2', 'B3', 'B4', 'B5', 'B6', 'B7', 'B8', 'B8A', 'B11', 'B12'];
60 var clasificacion = ee.Classifier.smileRandomForest(10).train(trainingsample, "Clase", bandas);
61 var imagen = composite.classify(clasificacion);
62
63 var landcoverPalette = [
64   "#0C2C84", // Agua (0)
65   "#e31a1c", // Urbano (1)
66   "#005a32", // Bosque (2)
67   "#ff0000", // Cultivo (3)
68   "#829121" // Paramo (4)
69 ];
70
71 Map.addLayer(imagen.clip(bounds), {palette: landcoverPalette, min: 0, max: 4}, "classification");

```

```

72
73 // Crear conjuntos de entrenamiento y validación:
74 var randomcolumn = entrenamiento.randomColumn();
75 var trainingsample = randomcolumn.filter(ee.Filter.gte("random", 0.3));
76 var validationssample = randomcolumn.filter(ee.Filter.lte("random", 0.3));
77
78 // Entrenar el clasificador y realizar la clasificación:
79 var bandas = ['B2', 'B3', 'B4', 'B5', 'B6', 'B7', 'B8', 'B8A', 'B11', 'B12'];
80 var clasificacion = ee.Classifier.smileRandomForest(10).train(trainingsample, "Clase", bandas);
81 var imagen = composite.classify(clasificacion);
82
83 // Visualizar la clasificación:
84 Map.addLayer(imagen.clip(bounds), {palette: landcoverPalette, min: 0, max: 4}, "classification");
85
86 // Realizar la clasificación en el conjunto de validación:
87 var classifiedValidation = validationssample.classify(clasificacion);
88
89 // Obtener la matriz de confusión:
90 var errorMatrix = classifiedValidation.errorMatrix({
91   actual: "Clase",
92   predicted: "classification"
93 });
94
95 print(errorMatrix, "Matriz de Confusión");
96
97 // Calcular la precisión global y las precisiones del usuario y del productor:
98 var overallAccuracy = errorMatrix.accuracy();
99 print("Precisión Global:", overallAccuracy);
100
101 var userAccuracy = errorMatrix.consumersAccuracy();
102 print("Precisión del Usuario:", userAccuracy);
103
104 var producerAccuracy = errorMatrix.producersAccuracy();
105 print("Precisión del Productor:", producerAccuracy);
106
107 // Calcular el coeficiente Kappa
108 var kappa = errorMatrix.kappa();
109 print("Coeficiente Kappa:", kappa);
110
111 // Crear una imagen con la verdad básica y la clasificación en cada píxel
112 var classifiedImage = imagen.select('classification');
113 var truthImage = validationssample.reduceToImage(["Clase"], ee.Reducer.first());
114
115 // Concatenar las imágenes de verdad básica y clasificación
116 var combinedImage = ee.Image.cat([truthImage, classifiedImage]);
117
118 // Reducir la imagen a través de las regiones de interés (ROI)
119 var confusionMatrix = combinedImage.reduceRegion({
120   reducer: ee.Reducer.fixedHistogram(0, 4, 5), // 0-4 son las clases posibles, ajusta según sea necesario
121   geometry: bounds,
122   scale: 10, // Ajustar según sea necesario
123   maxPixels: 1e13 // Ajustar según sea necesario
124 });
125
126 // Extraer la matriz de confusión
127 var matrix = confusionMatrix.get('classification');
128
129 // Mostrar la matriz de confusión
130 print("Matriz de Confusión", matrix);
131
132 var Burgay = imagen.clip(bounds);
133
134 Map.addLayer(Burgay, {palette: landcoverPalette, min: 0, max: 4}, "Burgay_2016");
135
136

```



```

137 var shapeBurgay = Burgay.reduceToVectors(
138 * {geometry:bounds,
139 //sentinel a 10 o landsat a 30
140 scale:10 ,
141 geometryType: 'polygon',
142 maxPixels:10e13,
143 labelProperty: 'clase',
144 eightConnected:true,
145 });
146
147
148 var claseAgua=shapeBurgay.filter(ee.Filter.eq("clase",0));
149 var claseUrbano=shapeBurgay.filter(ee.Filter.eq("clase",1));
150 var claseBosque=shapeBurgay.filter(ee.Filter.eq("clase",2));
151 var claseCultivo=shapeBurgay.filter(ee.Filter.eq("clase",3));
152 var claseParamo=shapeBurgay.filter(ee.Filter.eq("clase",4));
153
154 var areaAgua=claseAgua.geometry().area().divide(1e4).getInfo();
155 var areaUrbano= claseUrbano.geometry().area().divide(1e4).getInfo();
156 var areaBosque=claseBosque.geometry().area().divide(1e4).getInfo();
157 var areaCultivo=claseCultivo.geometry().area().divide(1e4).getInfo();
158 var areaParamo=claseParamo.geometry().area().divide(1e4).getInfo();
159 var suma=areaAgua+areaUrbano+areaBosque+areaCultivo+areaParamo;
160
161 print("clasificación supervisada")
162 print("clase Agua",areaAgua);
163 print("clase Urbano",areaUrbano);
164 print("clase Bosque",areaBosque);
165 print("clase Cultivo",areaCultivo);
166 print("clase Paramo", areaParamo);
167 print("suma",suma);
168 print("area Interes",bounds.geometry().area().divide(1e4));
169
170 var porcentajeAgua=(ee.Number(areaAgua).divide(ee.Number(suma)).multiply(100)).getInfo();
171 var porcentajeUrbano=(ee.Number(areaUrbano).divide(ee.Number(suma)).multiply(100)).getInfo();
172
173 var porcentajeBosque=(ee.Number(areaBosque).divide(ee.Number(suma)).multiply(100)).getInfo();
174 var porcentajeCultivo=(ee.Number(areaCultivo).divide(ee.Number(suma)).multiply(100)).getInfo();
175 var porcentajeParamo=(ee.Number(areaParamo).divide(ee.Number(suma)).multiply(100)).getInfo();
176
177 print("porcentaje agua",porcentajeAgua);
178 print("porcentaje urbano", porcentajeUrbano);
179 print("porcentaje bosque",porcentajeBosque);
180 print("porcentaje cultivo",porcentajeCultivo);
181 print("porcentaje Paramo", porcentajeParamo);
182
183 var SumaPorcentaje= porcentajeAgua+porcentajeUrbano+ porcentajeBosque+ porcentajeCultivo+porcentajeParamo;
184
185 print("SumaPorcentaje:",SumaPorcentaje);
186
187
188 * Export.image.toDrive({
189 image:imagen.clip(bounds),
190 description:"Burgay_2016",
191 scale:30,
192 region:geometry
193 });
194

```

```

1 var areaInteres=ee.FeatureCollection("users/MAGUS/SUBCUENCA_BURGAY_0");
2
3 var SentinelVizParam=function(imagenL,bandsL,nombre,shown){
4
5     Map.addLayer(imagenL,{min:0,max: 2000,gamma:1,opacity:1,bands:bandsL},nombre,shown);
6
7 };
8
9 Map.addLayer(areaInteres);
10 //Eliminar nubes y sombras
11 var cloud_masks = require('users/fitoprincipe/geetools:cloud_masks');
12 var options = ["cloud"];
13 var SentinelEnmascarar = cloud_masks.sentinel2 ([['cirrus','opaque']]);
14
15 var sentinel = ee.ImageCollection("COPERNICUS/S2")
16 .filterBounds(areaInteres);
17
18 //2016-10-01", "2016-12-30
19 //2016-01-01", "2016-04-30"
20 //quema 2016-06-01", "2016-09-30
21 var collection1=sentinel
22 .filterDate("2016-01-01", "2016-12-30")
23 .filterMetadata("CLOUDY_PIXEL_PERCENTAGE","less_than", 100)
24 .map(SentinelEnmascarar)
25 .map( function (imagen){
26
27     // CALCULO DE INDICES
28     var ndvi = imagen.normalizedDifference(["B8","B4"]).rename("NDVI");
29     var ndwi = imagen.normalizedDifference(["B3","B8"]).rename("NDWI");
30     var ndbi = imagen.normalizedDifference(["B11","B8"]).rename("NDBI");
31     imagen= imagen.addBands([ndbi,ndvi,ndwi]);
32
33     return imagen.clip(areaInteres);
34
35 });
36
37 var mediana=collection1.median();
38
39 var medianaNDVI= mediana.select('NDVI');
40 var medianaNDWI= mediana.select('NDWI');
41 var medianaNDBI= mediana.select('NDBI');
42
43
44 var agua= medianaNDWI.gte(0);
45 var vegetacion = medianaNDVI.gte(0.2).and( medianaNDVI.lte(0.43));
46 var vegetacionVigorosa=medianaNDVI.gt(0.43);
47 var construccion=medianaNDBI.gt(0.05);
48 //var sueloDesnudo = medianaNDVI.lte(0.2).and(construccion.not());
49 var sueloDesnudo = medianaNDVI.lte(0.2);
50
51
52
53 //Mascaras
54 var mascara_sueloDesnudo= sueloDesnudo
55     .updateMask(sueloDesnudo);
56 var mascara_agua=agua
57     .updateMask(agua);
58 var mascara_vegetacion=vegetacion
59     .updateMask(vegetacion);
60
61 var mascara_vegetacionVigorosa= vegetacionVigorosa
62     .updateMask( vegetacionVigorosa);
63 var mascara_construccion= construccion
64     .updateMask( construccion);
65
66
67 var mosaic = ee.ImageCollection
68 ([mascara_sueloDesnudo.visualize({palette: ['804000']})),
69  mascara_agua.visualize({palette: ['blue']}),
70  mascara_vegetacion.visualize({palette: ['green']}),
71  mascara_vegetacionVigorosa.visualize({palette: ['yellow']}),
72  //mascara_construccion.visualize({palette: ['red']}),
73 ]).mosaic();
74

```

```

75
76 var area = ee.Image.pixelArea().divide(1e4);
77 //B12 B8A B4
78 //B8A, 'B3', 'B2'
79 SentinelVizParam(mediana,['B8A','B3','B2'],'Mediana',true);
80
81 //Map.addLayer(slope,{},'slope',false);
82 Map.addLayer(mosaic,{},'Mosaico');
83 var mascara_sueloDesnudo_area = mascara_sueloDesnudo.multiply(area).select([0],['Suelo desnudo']);
84 var mascara_agua_area = mascara_agua.multiply(area).select([0],['Agua']);
85 var mascara_vegetacion_area = mascara_vegetacion.multiply(area).select([0],['Vegetacion']);
86 var mascara_vegetacionVigorosa_area = mascara_vegetacionVigorosa.multiply(area).select([0],['vegetacion Vigorosa']);
87
88
89 var area_image = mascara_sueloDesnudo_area.addBands(mascara_agua_area )
90 |>.addBands( mascara_vegetacion_area)
91 |>.addBands(mascara_vegetacionVigorosa_area );
92 * var areas = area_image.reduceRegion({
93   reducer:ee.Reducer.sum(),
94   geometry: areaInteres,
95   scale: 10,
96   maxPixels:1e13
97 });
98 print('hectareas',areas);
99 var areaAgua=areas.get("Agua").getInfo();
100 var areaSuelo=areas.get("Suelo desnudo").getInfo();
101 var areaVegetacion=areas.get("Vegetacion").getInfo();
102 var areaVegetacionVigorosa=areas.get("vegetacion Vigorosa").getInfo();
103 var suma=areaAgua+ areaSuelo +areaVegetacion+areaVegetacionVigorosa;
104 print("suma",suma);
105 print("area Interes",areaInteres.geometry().area().divide(1e4));
106 print("porcentaje agua",ee.Number(areaAgua).divide(ee.Number(suma)).multiply(100))
107 print("porcentaje suelo",ee.Number(areaSuelo).divide(ee.Number(suma)).multiply(100))
108 print("porcentaje vegetacion",ee.Number(areaVegetacion).divide(ee.Number(suma)).multiply(100))
109 print("porcentaje vegetacion vigorosa",ee.Number( areaVegetacionVigorosa).divide(ee.Number(suma)).multiply(100))
110
111
112 Map.centerObject(areaInteres,8);
113
114

```

Anexo E

Codificación Sentinel-2 2018

```

1 var bounds = ee.FeatureCollection("users/MAGUS/SUBCUENCA_BURGAY_Q");
2
3 // Function to mask clouds using the Sentinel-2 QA band.
4 function maskS2clouds(image) {
5   var qa = image.select("QA60");
6
7   // Bits 10 and 11 are clouds and cirrus, respectively.
8   var cloudBitMask = 1 << 10;
9   var cirrusBitMask = 1 << 11;
10
11   // Both flags should be set to zero, indicating clear conditions.
12   var mask = qa.bitwiseAnd(cloudBitMask).eq(0).and(
13     qa.bitwiseAnd(cirrusBitMask).eq(0));
14
15   // Return the masked and scaled data, without the QA bands.
16   return image.updateMask(mask).divide(10000)
17     .select(['B2', 'B3', 'B4', 'B5', 'B6', 'B7', 'B8', 'B8A', 'B11', 'B12'])
18     .copyProperties(image, ["system:time_start"]);
19 }
20
21 // Map the function over one year of data and take the median.
22 // Load Sentinel-2 TOA reflectance data.
23 var composite = ee.ImageCollection("COPERNICUS/S2")
24   .filterBounds(bounds)
25   .filterDate('2018-01-01', '2018-12-31')
26   // Pre-filter to get less cloudy granules.
27   .filter(ee.Filter.lt('CLOUDY_PIXEL_PERCENTAGE', 20))
28   .map(maskS2clouds)
29   .median();
30
31 // Get total number of pixels
32 var areaTotal = composite.reduceRegion({
33   reducer: ee.Reducer.count(),
34   geometry: bounds,
35   scale: 10, // Adjust the scale according to your needs
36   maxPixels: 1e13 // Adjust as needed
37 });
38
39 // Extract the total number of pixels from the property
40 var totalPixels = ee.Number(areaTotal.get("B2")).getInfo(); // Adjust the band according to your needs
41
42 print("Número total de píxeles:", totalPixels);
43
44 // Display the results in a cloudy place.
45 Map.setCenter(-78.8666, -2.7263, 12);
46 Map.addLayer(composite.clip(bounds), {bands: ['B4', 'B3', 'B2'], max: 0.8}, "Imagen de agricultura");
47
48 var muestras = Agua.merge(Bosque).merge(Cultivo).merge(Urbano).merge(Paramo);
49 var entrenamiento = composite.sampleRegions(muestras, ["Clase"], 30);
50
51 var randomcolumn = entrenamiento.randomColumn();
52
53 var trainingsample = randomcolumn.filter(ee.Filter.gte("random", 0.3));
54 var validationssample = randomcolumn.filter(ee.Filter.lte("random", 0.3));
55
56 print(entrenamiento.limit(1));
57 print(entrenamiento.size());
58
59 var bandas = ['B2', 'B3', 'B4', 'B5', 'B6', 'B7', 'B8', 'B8A', 'B11', 'B12'];
60 var clasificacion = ee.Classifier.smileRandomForest(10).train(trainingsample, "Clase", bandas);
61 var imagen = composite.classify(clasificacion);
62
63 var landcoverPalette = [
64   "#000000", // Agua (0)
65   "#333333", // Urbano (1)
66   "#005a32", // Bosque (2)
67   "#ff8000", // Cultivo (3)
68   "#829121" // Paramo (4)
69 ];

```

```

71 Map.addLayer(imagen.clip(bounds), {palette: landcoverPalette, min: 0, max: 4}, "classification");
72
73 // Crear conjuntos de entrenamiento y validación:
74 var randomcolumn = entrenamiento.randomColumn();
75 var trainingsample = randomcolumn.filter(ee.Filter.gte("random", 0.3));
76 var validationssample = randomcolumn.filter(ee.Filter.lte("random", 0.3));
77
78 // Entrenar el clasificador y realizar la clasificación:
79 var bandas = ['B2', 'B3', 'B4', 'B5', 'B6', 'B7', 'B8', 'B8A', 'B11', 'B12'];
80 var clasificacion = ee.Classifier.smileRandomForest(10).train(trainingsample, "Clase", bandas);
81 var imagen = composite.classify(clasificacion);
82
83 // Visualizar la clasificación:
84 Map.addLayer(imagen.clip(bounds), {palette: landcoverPalette, min: 0, max: 4}, "classification");
85
86 // Realizar la clasificación en el conjunto de validación:
87 var classifiedValidation = validationssample.classify(clasificacion);
88
89 // Obtener la matriz de confusión:
90 var errorMatrix = classifiedValidation.errorMatrix({
91   actual: "Clase",
92   predicted: "classification"
93 });
94
95 print(errorMatrix, "Matriz de Confusión");
96
97 // Calcular la precisión global y las precisiones del usuario y del productor:
98 var overallAccuracy = errorMatrix.accuracy();
99 print("Precisión Global:", overallAccuracy);
100
101 var userAccuracy = errorMatrix.consumersAccuracy();
102 print("Precisión del Usuario:", userAccuracy);
103
104 var producerAccuracy = errorMatrix.producersAccuracy();
105 print("Precisión del Productor:", producerAccuracy);
106
107 // Calcular el coeficiente Kappa
108 var kappa = errorMatrix.kappa();
109 print("Coeficiente Kappa:", kappa);
110
111 // Crear una imagen con la verdad básica y la clasificación en cada píxel
112 var classifiedImage = imagen.select('classification');
113 var truthImage = validationssample.reduceToImage(["Clase"], ee.Reducer.first());
114
115 // Concatenar las imágenes de verdad básica y clasificación
116 var combinedImage = ee.Image.cat([truthImage, classifiedImage]);
117
118 // Reducir la imagen a través de las regiones de interés (ROI)
119 var confusionMatrix = combinedImage.reduceRegion({
120   reducer: ee.Reducer.fixedHistogram(0, 4, 5), // 0-4 son las clases posibles, ajusta según sea necesario
121   geometry: bounds,
122   scale: 10, // Ajustar según sea necesario
123   maxPixels: 1e13 // Ajustar según sea necesario
124 });
125
126 // Extraer la matriz de confusión
127 var matrix = confusionMatrix.get('classification');
128
129 // Mostrar la matriz de confusión
130 print("Matriz de Confusión", matrix);
131
132 var Burgay = imagen.clip(bounds);
133
134 Map.addLayer(Burgay, {palette: landcoverPalette, min: 0, max: 4}, "Burgay_2018");
135

```



```

136
137 var shapeBurgay = Burgay.reduceToVectors(
138 + {geometry:bounds,
139 //sentinel a 10 o landsat a 30
140 scale:10 ,
141 geometryType: 'polygon',
142 maxPixels:10e13,
143 labelProperty: 'clase',
144 eightConnected:true,
145 });
146
147
148 var claseAgua=shapeBurgay.filter(ee.Filter.eq("clase",0));
149 var claseUrbano=shapeBurgay.filter(ee.Filter.eq("clase",1));
150 var claseBosque=shapeBurgay.filter(ee.Filter.eq("clase",2));
151 var claseCultivo=shapeBurgay.filter(ee.Filter.eq("clase",3));
152 var claseParamo=shapeBurgay.filter(ee.Filter.eq("clase",4));
153
154 var areaAgua=claseAgua.geometry().area().divide(1e4).getInfo();
155 var areaUrbano= claseUrbano.geometry().area().divide(1e4).getInfo();
156 var areaBosque=claseBosque.geometry().area().divide(1e4).getInfo();
157 var areaCultivo=claseCultivo.geometry().area().divide(1e4).getInfo();
158 var areaParamo=claseParamo.geometry().area().divide(1e4).getInfo();
159 var suma=areaAgua+areaUrbano+areaBosque+areaCultivo+areaParamo;
160
161 print("clasificación supervisada")
162 print("clase Agua",areaAgua);
163 print("clase Urbano",areaUrbano);
164 print("clase Bosque",areaBosque);
165 print("clase Cultivo",areaCultivo);
166 print("clase Paramo", areaParamo);
167 print("suma",suma);
168 print("area Interes",bounds.geometry().area().divide(1e4));
169
170 var porcentajeAgua=(ee.Number(areaAgua).divide(ee.Number(suma)).multiply(100)).getInfo();
171 var porcentajeUrbano=(ee.Number(areaUrbano).divide(ee.Number(suma)).multiply(100)).getInfo();
172
173 var porcentajeBosque=(ee.Number(areaBosque).divide(ee.Number(suma)).multiply(100)).getInfo();
174 var porcentajeCultivo=(ee.Number(areaCultivo).divide(ee.Number(suma)).multiply(100)).getInfo();
175 var porcentajeParamo=(ee.Number(areaParamo).divide(ee.Number(suma)).multiply(100)).getInfo();
176
177 print("porcentaje agua",porcentajeAgua);
178 print("porcentaje urbano", porcentajeUrbano);
179 print("porcentaje bosque",porcentajeBosque);
180 print("porcentaje cultivo",porcentajeCultivo);
181 print("porcentaje Paramo", porcentajeParamo);
182
183 var SumaPorcentaje= porcentajeAgua+porcentajeUrbano+ porcentajeBosque+ porcentajeCultivo+porcentajeParamo;
184
185 print("SumaPorcentaje:",SumaPorcentaje);
186
187
188 + Export.image.toDrive({
189 image:imagen.clip(bounds),
190 description:"Burgay_2018",
191 scale:30,
192 region:geometry
193 });
194
195

```

```

1  var areaInteres=ee.FeatureCollection("users/MAGUS/SUBCUENCA_BURGAY_0");
2
3  var SentinelVizParam=function(imagenL,bandsL,nombre,shown){
4
5      Map.addLayer(imagenL,{min:0,max: 2000,gamma:1,opacity:1,bands:bandsL},nombre,shown);
6
7  };
8
9  Map.addLayer(areaInteres);
10 //Eliminar nubes y sombras
11 var cloud_masks = require('users/fitoprincipe/geetools:cloud_masks');
12 var options = ["cloud"];
13 var SentinelEnmascarar = cloud_masks.sentinel2 ([ 'cirrus','opaque' ]);
14
15 var sentinel = ee.ImageCollection("COPERNICUS/S2")
16   .filterBounds(areaInteres);
17
18 //2020-10-01", "2020-12-30
19 //2020-01-01", "2020-04-30"
20 //quema 2020-06-01", "2020-09-30
21 var collection1=sentinel
22   .filterDate("2020-01-01", "2020-12-30")
23   .filterMetadata("CLOUDY_PIXEL_PERCENTAGE","less_than", 100)
24   .map(SentinelEnmascarar)
25   .map( function (imagen){
26
27       // CALCULO DE INDICES
28       var ndvi = imagen.normalizedDifference(["B8","B4"]).rename("NDVI");
29       var ndwi = imagen.normalizedDifference(["B3","B8"]).rename("NDWI");
30       var ndbi = imagen.normalizedDifference(["B11","B8"]).rename("NDBI");
31       imagen= imagen.addBands([ndbi,ndvi,ndwi]);
32
33       return imagen.clip(areaInteres);
34   });
35
36 var mediana=collection1.median();
37
38 var medianaNDVI= mediana.select("NDVI");
39 var medianaNDWI= mediana.select("NDWI");
40 var medianaNDBI= mediana.select("NDBI");
41
42
43 var agua= medianaNDWI.gte(0);
44 var vegetacion = medianaNDVI.gte(0.2).and( medianaNDVI.lte(0.43));
45 var vegetacionVigorosa=medianaNDVI.gt(0.43);
46 var construccion=medianaNDBI.gt(0.05);
47 //var sueloDesnudo = medianaNDVI.lte(0.2).and(construccion.not());
48 var sueloDesnudo = medianaNDVI.lte(0.2);
49
50
51
52
53 //Mascaras
54 var mascara_sueloDesnudo= sueloDesnudo
55   .updateMask(sueloDesnudo);
56 var mascara_agua=agua
57   .updateMask(agua);
58 var mascara_vegetacion=vegetacion
59   .updateMask(vegetacion);
60
61 var mascara_vegetacionVigorosa= vegetacionVigorosa
62   .updateMask( vegetacionVigorosa);
63 var mascara_construccion= construccion
64   .updateMask( construccion);
65

```



```

66 | | | |
67 var mosaic = ee.ImageCollection
68 ([mascara_sueloDesnudo.visualize({palette: ['804000']})),
69 mascara_agua.visualize({palette: ['blue']})),
70 mascara_vegetacion.visualize({palette: ['green']})),
71 mascara_vegetacionVigorosa.visualize({palette: ['yellow']})),
72 //mascara_construccion.visualize({palette: ['red']})),
73 ]).mosaic();
74
75
76 var area = ee.Image.pixelArea().divide(1e4);
77 //B12 B8A B4
78 //'B8A','B3','B2'
79 SentinelVizParam(mediana,['B8A','B3','B2'],'Mediana',true);
80
81 //Map.addLayer(slope,{},'slope',false);
82 Map.addLayer(mosaic,{},'Mosaico');
83 var mascara_sueloDesnudo_area = mascara_sueloDesnudo.multiply(area).select([0],['Suelo desnudo']);
84 var mascara_agua_area = mascara_agua.multiply(area).select([0],['Agua']);
85 var mascara_vegetacion_area = mascara_vegetacion.multiply(area).select([0],['Vegetacion']);
86 var mascara_vegetacionVigorosa_area = mascara_vegetacionVigorosa.multiply(area).select([0],['vegetacion Vigorosa']);
87
88
89 var area_image = mascara_sueloDesnudo_area.addBands(mascara_agua_area )
90 .addBands( mascara_vegetacion_area)
91 .addBands(mascara_vegetacionVigorosa_area );
92 var areas = area_image.reduceRegion({
93   reducer:ee.Reducer.sum(),
94   geometry: areaInteres,
95   scale: 10,
96   maxPixels:1e13
97 });
98 print('hectareas',areas);
99 var areaAgua=areas.get("Agua").getInfo();
100 var areaSuelo=areas.get("Suelo desnudo").getInfo();
101 var areaVegetacion=areas.get("Vegetacion").getInfo();
102 var areaVegetacionVigorosa=areas.get("vegetacion Vigorosa").getInfo();
103 var suma=areaAgua+ areaSuelo +areaVegetacion+areaVegetacionVigorosa;
104 print("suma",suma);
105 print("area Interes",areaInteres.geometry().area().divide(1e4));
106 print("porcentaje agua",ee.Number(areaAgua).divide(ee.Number(suma)).multiply(100))
107 print("porcentaje suelo",ee.Number(areaSuelo).divide(ee.Number(suma)).multiply(100))
108 print("porcentaje vegetacion",ee.Number(areaVegetacion).divide(ee.Number(suma)).multiply(100))
109 print("porcentaje vegetacion vigorosa",ee.Number( areaVegetacionVigorosa).divide(ee.Number(suma)).multiply(100))
110
111
112 Map.centerObject(areaInteres,8);
113
114

```

Anexo F

Codificación Sentinel-2 2020

```

1 var bounds = ee.FeatureCollection("users/MAGUS/SUBCUENCA_BURGAY_0");
2
3 // Function to mask clouds using the Sentinel-2 QA band.
4 function maskS2clouds(image) {
5   var qa = image.select('QA60');
6
7   // Bits 10 and 11 are clouds and cirrus, respectively.
8   var cloudBitMask = 1 << 10;
9   var cirrusBitMask = 1 << 11;
10
11   // Both flags should be set to zero, indicating clear conditions.
12   var mask = qa.bitwiseAnd(cloudBitMask).eq(0).and(
13     qa.bitwiseAnd(cirrusBitMask).eq(0));
14
15   // Return the masked and scaled data, without the QA bands.
16   return image.updateMask(mask).divide(10000)
17     .select(['B2', 'B3', 'B4', 'B5', 'B6', 'B7', 'B8', 'B8A', 'B11', 'B12'])
18     .copyProperties(image, ['system:time_start']);
19 }
20
21 // Map the function over one year of data and take the median.
22 // Load Sentinel-2 TOA reflectance data.
23 var composite = ee.ImageCollection("COPERNICUS/S2")
24   .filterBounds(bounds)
25   .filterDate('2020-01-01', '2020-12-31')
26   // Pre-filter to get less cloudy granules.
27   .filter(ee.Filter.lt('CLOUDY_PIXEL_PERCENTAGE', 20))
28   .map(maskS2clouds)
29   .median();
30
31 // Get total number of pixels
32 var areaTotal = composite.reduceRegion({
33   reducer: ee.Reducer.count(),
34   geometry: bounds,
35   scale: 10, // Adjust the scale according to your needs
36   maxPixels: 1e13 // Adjust as needed
37 });
38
39 // Extract the total number of pixels from the property
40 var totalPíxeles = ee.Number(areaTotal.get("B2")).getInfo(); // Adjust the band according to your needs
41
42 print("Número total de píxeles:", totalPíxeles);
43
44 // Display the results in a cloudy place.
45 Map.setCenter(-78.8666, -2.7263, 12);
46 Map.addLayer(composite.clip(bounds), {bands: ['B4', 'B3', 'B2'], max: 0.8}, "Imagen de agricultura");
47
48 var muestras = Agua.merge(Bosque).merge(Cultivo).merge(Urbano).merge(Paramo);
49 var entrenamiento = composite.sampleRegions(muestras, ["Clase"], 30);
50
51 var randomcolumn = entrenamiento.randomColumn();
52
53 var trainingsample = randomcolumn.filter(ee.Filter.gte("random", 0.3));
54 var validationssample = randomcolumn.filter(ee.Filter.lte("random", 0.3));
55
56 print(entrenamiento.limit(1));
57 print(entrenamiento.size());
58
59 var bandas = ['B2', 'B3', 'B4', 'B5', 'B6', 'B7', 'B8', 'B8A', 'B11', 'B12'];
60 var clasificacion = ee.Classifier.smileRandomForest(10).train(trainingsample, "Clase", bandas);
61 var imagen = composite.classify(clasificacion);
62
63 var landcoverPalette = [
64   "#0C2C84", // Agua (0)
65   "#e31a1c", // Urbano (1)
66   "#005a32", // Bosque (2)
67   "#FF8000", // Cultivo (3)
68   "#829121" // Paramo (4)
69 ];

```

```

70
71 Map.addLayer(imagen.clip(bounds), {palette: landcoverPalette, min: 0, max: 4}, "classification");
72
73 // Crear conjuntos de entrenamiento y validación:
74 var randomcolumn = entrenamiento.randomColumn();
75 var trainingsample = randomcolumn.filter(ee.Filter.gte("random", 0.3));
76 var validationssample = randomcolumn.filter(ee.Filter.lte("random", 0.3));
77
78 // Entrenar el clasificador y realizar la clasificación:
79 var bandas = ['B2', 'B3', 'B4', 'B5', 'B6', 'B7', 'B8', 'B8A', 'B11', 'B12'];
80 var clasificacion = ee.Classifier.smileRandomForest(10).train(trainingsample, "Clase", bandas);
81 var imagen = composite.classify(clasificacion);
82
83 // Visualizar la clasificación:
84 Map.addLayer(imagen.clip(bounds), {palette: landcoverPalette, min: 0, max: 4}, "classification");
85
86 // Realizar la clasificación en el conjunto de validación:
87 var classifiedValidation = validationssample.classify(clasificacion);
88
89 // Obtener la matriz de confusión:
90 var errorMatrix = classifiedValidation.errorMatrix({
91   actual: "Clase",
92   predicted: "classification"
93 });
94
95 print(errorMatrix, "Matriz de Confusión");
96
97 // Calcular la precisión global y las precisiones del usuario y del productor:
98 var overallAccuracy = errorMatrix.accuracy();
99 print("Precisión Global:", overallAccuracy);
100
101 var userAccuracy = errorMatrix.consumersAccuracy();
102 print("Precisión del Usuario:", userAccuracy);
103
104 var producerAccuracy = errorMatrix.producersAccuracy();
105 print("Precisión del Productor:", producerAccuracy);
106
107 // Calcular el coeficiente Kappa
108 var kappa = errorMatrix.kappa();
109 print("Coeficiente Kappa:", kappa);
110
111 // Crear una imagen con la verdad básica y la clasificación en cada píxel
112 var classifiedImage = imagen.select("classification");
113 var truthImage = validationssample.reduceToImage(["Clase"], ee.Reducer.first());
114
115 // Concatenar las imágenes de verdad básica y clasificación
116 var combinedImage = ee.Image.cat([truthImage, classifiedImage]);
117
118 // Reducir la imagen a través de las regiones de interés (ROI)
119 var confusionMatrix = combinedImage.reduceRegion({
120   reducer: ee.Reducer.fixedHistogram(0, 4, 5), // 0-4 son las clases posibles, ajusta según sea necesario
121   geometry: bounds,
122   scale: 10, // Ajustar según sea necesario
123   maxPixels: 1e13 // Ajustar según sea necesario
124 });
125
126 // Extraer la matriz de confusión
127 var matrix = confusionMatrix.get("classification");
128
129 // Mostrar la matriz de confusión
130 print("Matriz de Confusión", matrix);
131
132 var Burgay = imagen.clip(bounds);
133
134 Map.addLayer(Burgay, {palette: landcoverPalette, min: 0, max: 4}, "Burgay_2020");
135

```

```

136
137 var shapeBurgay = Burgay.reduceToVectors(
138   {geometry:bounds,
139     //sentinel a 10 o landsat a 30
140     scale:10 ,
141     geometryType: 'polygon',
142     maxPixels:10e13,
143     labelProperty: 'clase',
144     eightConnected:true,
145   });
146
147
148 var claseAgua=shapeBurgay.filter(ee.Filter.eq("clase",0));
149 var claseUrbano=shapeBurgay.filter(ee.Filter.eq("clase",1));
150 var claseBosque=shapeBurgay.filter(ee.Filter.eq("clase",2));
151 var claseCultivo=shapeBurgay.filter(ee.Filter.eq("clase",3));
152 var claseParamo=shapeBurgay.filter(ee.Filter.eq("clase",4));
153
154 var areaAgua=claseAgua.geometry().area().divide(1e4).getInfo();
155 var areaUrbano= claseUrbano.geometry().area().divide(1e4).getInfo();
156 var areaBosque=claseBosque.geometry().area().divide(1e4).getInfo();
157 var areaCultivo=claseCultivo.geometry().area().divide(1e4).getInfo();
158 var areaParamo=claseParamo.geometry().area().divide(1e4).getInfo();
159 var suma=areaAgua+areaUrbano+areaBosque+areaCultivo+areaParamo;
160
161 print("clasificación supervisada")
162 print("clase Agua",areaAgua);
163 print("clase Urbano",areaUrbano);
164 print("clase Bosque",areaBosque);
165 print("clase Cultivo",areaCultivo);
166 print("clase Paramo", areaParamo);
167 print("suma",suma);
168 print("area Interes", bounds.geometry().area().divide(1e4));
169
170 var porcentajeAgua=(ee.Number(areaAgua).divide(ee.Number(suma)).multiply(100)).getInfo();
171 var porcentajeUrbano=(ee.Number(areaUrbano).divide(ee.Number(suma)).multiply(100)).getInfo();
172
173 var porcentajeBosque=(ee.Number(areaBosque).divide(ee.Number(suma)).multiply(100)).getInfo();
174 var porcentajeCultivo=(ee.Number(areaCultivo).divide(ee.Number(suma)).multiply(100)).getInfo();
175 var porcentajeParamo=(ee.Number(areaParamo).divide(ee.Number(suma)).multiply(100)).getInfo();
176
177 print("porcentaje agua",porcentajeAgua);
178 print("porcentaje urbano", porcentajeUrbano);
179 print("porcentaje bosque",porcentajeBosque);
180 print("porcentaje cultivo",porcentajeCultivo);
181 print("porcentaje Paramo", porcentajeParamo);
182
183 var SumaPorcentaje= porcentajeAgua+porcentajeUrbano+ porcentajeBosque+ porcentajeCultivo+porcentajeParamo;
184
185 print("SumaPorcentaje:",SumaPorcentaje);
186
187
188 Export.image.toDrive({
189   image:imagen.clip(bounds),
190   description:"Burgay_2020",
191   scale:30,
192   region:geometry
193 });
194
195

```

```

1  var areaInteres=ee.FeatureCollection("users/MAGUS/SUBCUENCA_BURGAY_0");
2
3  var SentinelVizParam=function(imagenL,bandsL,nombre,shown){
4
5      Map.addLayer(imagenL,{min:0,max: 2000,gamma:1,opacity:1,bands:bandsL},nombre,shown);
6
7  };
8
9  Map.addLayer(areaInteres);
10 //Eliminar nubes y sombras
11 var cloud_masks = require('users/fitoprincipe/geetools:cloud_masks');
12 var options = ["cloud"];
13 var SentinelEnmascarar = cloud_masks.sentinel2 ([ 'cirrus','opaque' ]);
14
15 var sentinel = ee.ImageCollection("COPERNICUS/S2")
16   .filterBounds(areaInteres);
17
18 //2020-10-01", "2020-12-30
19 //2020-01-01", "2020-04-30"
20 //quema 2020-06-01", "2020-09-30
21 var collection1=sentinel
22   .filterDate("2020-01-01", "2020-12-30")
23   .filterMetadata("CLOUDY_PIXEL_PERCENTAGE","less_than", 100)
24   .map(SentinelEnmascarar)
25   .map( function (imagen){
26
27       // CALCULO DE INDICES
28       var ndvi = imagen.normalizedDifference(["B8","B4"]).rename("NDVI");
29       var ndwi = imagen.normalizedDifference(["B3","B8"]).rename("NDWI");
30       var ndbi = imagen.normalizedDifference(["B11","B8"]).rename("NDBI");
31       imagen= imagen.addBands([ndbi,ndvi,ndwi]);
32
33       return imagen.clip(areaInteres);
34   });
35
36 var mediana=collection1.median();
37
38 var medianaNDVI= mediana.select('NDVI');
39 var medianaNDWI= mediana.select('NDWI');
40 var medianaNDBI= mediana.select('NDBI');
41
42
43
44 var agua= medianaNDWI.gte(0);
45 var vegetacion = medianaNDVI.gte(0.2).and( medianaNDVI.lte(0.43));
46 var vegetacionVigorosa=medianaNDVI.gt(0.43);
47 var construccion=medianaNDBI.gt(0.05);
48 //var sueloDesnudo = medianaNDVI.lte(0.2).and(construccion.not());
49 var sueloDesnudo = medianaNDVI.lte(0.2);
50
51
52
53 //Mascaras
54 var mascara_sueloDesnudo= sueloDesnudo
55   .updateMask(sueloDesnudo);
56 var mascara_agua=agua
57   .updateMask(agua);
58 var mascara_vegetacion=vegetacion
59   .updateMask(vegetacion);
60
61 var mascara_vegetacionVigorosa= vegetacionVigorosa
62   .updateMask( vegetacionVigorosa);
63 var mascara_construccion= construccion
64   .updateMask( construccion);
65

```


María Augusta Murillo Patiño - María Camila Obando Loja

Anexo G

Tabla de valores Precisión general y Coeficiente Kappa - Intervalidation, PlanetLAB de Landsat-8 y Sentinel-2

	Precisión general	Coeficiente Kappa	Precisión general - PL	Coeficiente Kappa - PL
Landsat 8				
2016	95,33	94,14	92,51	92,1
Landsat 8				
2018	96,75	95,9	94,64	93,92
Landsat 8				
2020	98,62	98,26	95,4	95,21
Sentinel 2				
2016	99,28	99,1	97,38	96,7
Sentinel 2				
2018	93,65	91,98	91,42	90,71
Sentinel 2				
2020	98,63	98,28	94,16	92,67

Anexo H

Comparación de análisis de datos en QGIS vs GEE

