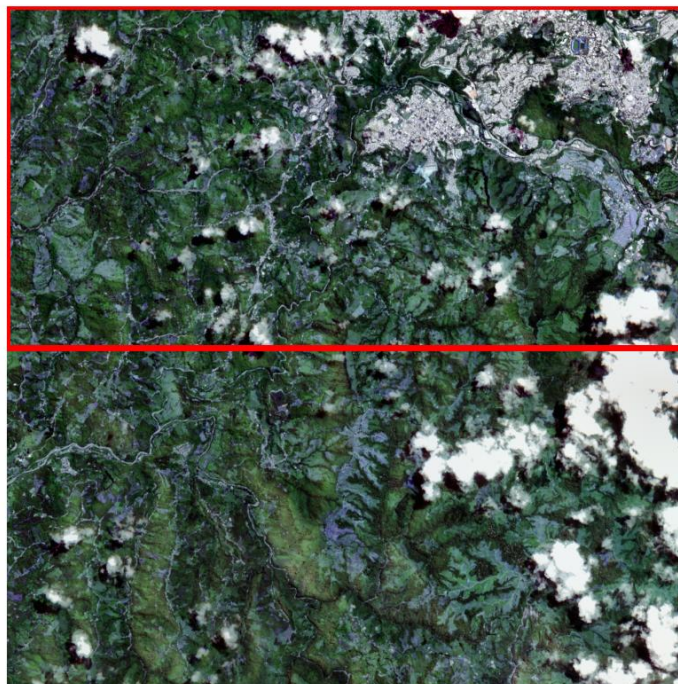

INTRODUCCIÓN Y DECISIONES TÉCNICAS INICIALES

La presente prueba técnica tiene como objetivo desarrollar un modelo de clasificación de coberturas del suelo a partir de imágenes satelitales Sentinel, utilizando la metodología Corine Land Cover (CLC) Nivel III. Para ello, se planteó una estrategia basada en técnicas de aprendizaje automático y análisis multiespectral, con el fin de identificar, etiquetar y clasificar con precisión las distintas coberturas presentes en el área de estudio.

En el marco de este proceso, se optó por delimitar una subárea específica del total de 14.253 hectáreas disponibles en la imagen satelital. Esta decisión se fundamenta en criterios técnicos relacionados con el tiempo disponible para el desarrollo, la calidad espectral de los datos y la viabilidad operativa del flujo de trabajo propuesto. En particular, se seleccionó la mitad superior del área de estudio, ya que presenta menor interferencia por nubosidad (Figura 1), lo que garantiza mejores condiciones para el análisis espectral y la generación de etiquetas precisas, esta subárea, de aproximadamente 7.126 hectáreas, ofrece además una suficiente diversidad de coberturas para realizar un entrenamiento representativo del modelo.

En la Figura 1 se presenta una visualización mejorada mediante un realce estratégico basado en el análisis de los histogramas individuales de cada banda espectral. Esta técnica permite optimizar la interpretación de las coberturas y sus características distintivas, facilitando la identificación de áreas clave y de esta forma identificando patrones importantes que luego son claves para estructurar el modelo de aprendizaje automático.

Figura 1. Visualización de área seleccionada en rojo



Adicionalmente, se implementó una estrategia de entrenamiento por clase, es decir, se entrenó el modelo utilizando un enfoque individual por tipo de cobertura. Esta decisión responde, en primera instancia, a los tiempos que implica cada proceso de entrenamiento y al interés de evidenciar mis capacidades en la elaboración y evaluación general de modelos. Asimismo, entrenar de forma separada para cada cobertura permite identificar con mayor claridad las características espaciales y espectrales propias de cada clase, lo que facilita una mejor interpretación de los resultados y mejora el desempeño del modelo. La proyección es repetir este proceso con todas las clases de coberturas identificadas en el nivel 3 del Corine Land Cover.

ORGANIZACIÓN DE DATOS Y ENTREGABLES

La metodología desarrollada combinó herramientas y entornos de trabajo de manera híbrida, incluyendo ArcGIS Pro para el análisis geográfico, ArcGIS Notebooks para la integración con Python, Google Earth Engine para la determinación de coberturas en la clasificación supervisada y no supervisada, y Google Colab para el desarrollo y ejecución de modelos con mayores capacidades computacionales.

La documentación y los resultados han sido organizados y consolidados dentro de una estructura de carpetas que se entrega comprimida en formato .zip y también está disponible en el repositorio de GitHub.

https://github.com/dilmillano/CLC_N3_IA_DianaMillan.git

Los entregables se encuentran organizados de la siguiente forma:

- data : Insumos utilizados y generados durante el procesamiento.
- doc : Este documento técnico con la descripción metodológica completa.
- output : Almacena los resultados del proceso, incluyendo el archivo shapefile con las coberturas clasificadas.
- script: Contiene los notebooks y scripts utilizados en el desarrollo del modelo.
- README.txt : Detalla el contenido y propósito de cada archivo entregado.

METODOLOGÍA

1. Segmentación espacial de la imagen satelital

Se realizó la segmentación sistemática de la imagen satelital en mosaicos de 120 por 120 píxeles, conservando la resolución y el sistema de referencia original. Mediante Google Colab, cada segmento fue exportado como un archivo independiente en formato GeoTIFF, utilizando un nombre secuencial que mantiene la trazabilidad del tile dentro del área de estudio. Junto con la segmentación, se generó un índice geográfico en formato shapefile y GeoJSON, que contiene la geometría y el identificador único de cada mosaico, permitiendo su visualización e integración en sistemas de información geográfica.

Como parte de este preprocesamiento, se generó una máscara binaria para identificar áreas con presencia de nubes, utilizando umbrales ajustables sobre los valores normalizados de las bandas RGB, lo cual permitió discriminar píxeles con alta reflectancia que es una característica típica de las nubes. Este insumo se utilizó para el shape etiquetado y también para filtrar los mosaicos con mayor presencia de nubes antes del entrenamiento, asegurando así una mayor calidad en la generación de etiquetas y en el desempeño posterior del modelo.

- *Script: 1_preprocesamiento.ipynb*
- *Output: tiles_index.shp ; tiles_index.geojson ; nube_mask.shp ; nube_mask_geojson*

2. Reducción del área de estudio y recomendación técnica

Durante la inspección preliminar de la imagen satelital, se identificó una cobertura nubosa significativa que afectaba aproximadamente al 9% del área de estudio, lo que generaba interferencias en la interpretación visual y en los procesos automáticos de clasificación. Para mitigar este problema, se decidió reducir el área de trabajo a la mitad, seleccionando los sectores con menor presencia de nubes y mayor homogeneidad.

Como recomendación técnica para futuros análisis, se sugiere complementar las imágenes Sentinel con composiciones multitemporales del mismo año. Es procedimiento se puede hacer mediante Google Earth Engine (GEE) el cual ofrece algoritmos directamente para generar composiciones libres de nubes mediante técnicas de enmascaramiento y combinación de imágenes. Eso esta documentado en “Sentinel-2 Cloud Masking with s2cloudless” directamente en el repositorio de Google como se muestra en el siguiente enlace:

<https://developers.google.com/earth-engine/tutorials/community/sentinel-2-s2cloudless>

- *Output: cropped_image_1_subset.tif*

3. Exploración de clases a partir de referencia Corine Land Cover III

Como parte del análisis exploratorio, se realizó un ejercicio que funcionó como un muestreo virtual, con el propósito de identificar la variabilidad ecológica presente en el área de estudio y entender cómo se distribuyen espacialmente las coberturas del suelo. Para ello, se superpuso una capa de referencia del Corine Land Cover nivel III, proveniente del IDEAM, lo que permitió detectar las clases existentes y orientar mejor el proceso de etiquetado. Utilizando la herramienta Intersect en ArcGIS Pro, se cruzaron los polígonos de cobertura con el índice espacial de mosaicos previamente generado. A partir del resultado, se calcularon las proporciones de cada clase dentro de cada mosaico (tile) mediante un análisis en ArcGIS Notebooks, considerando el área relativa ocupada por cada categoría (PCT_AREA). Esta proporción se empleó como criterio para clasificar los mosaicos en dos grupos: aquellos con una clase dominante ($\geq 30\%$ del área) y aquellos con representación baja pero relevante ($< 30\%$). Esta información se consolidó en un archivo Excel y fue utilizada para generar un shapefile con los mosaicos seleccionados para el etiquetado.

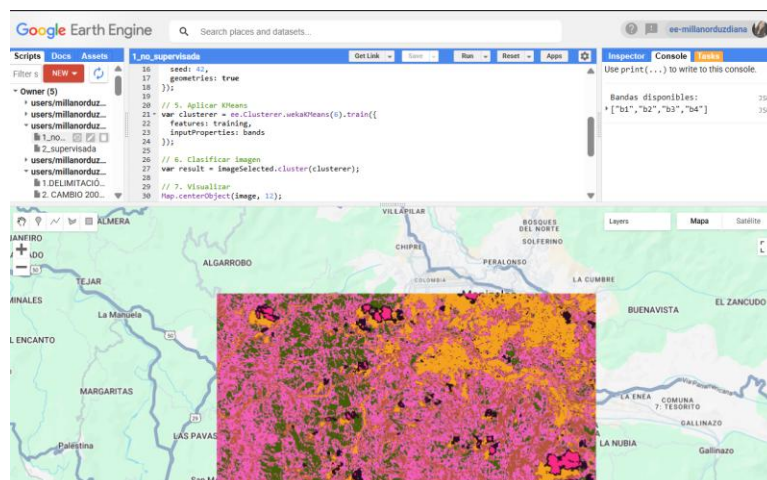
- Script: 2_seleccion_tiles_arcgis.ipynb
- Output: tiles_por_clase.xlsx

4. Generación de etiquetas mediante clasificación supervisada y no supervisada

Para construir un dataset de entrenamiento de manera eficiente se utilizaron dos metodologías complementarias para la generación de etiquetas: clasificación no supervisada y clasificación supervisada, ambas ejecutadas en Google Earth Engine (GEE).

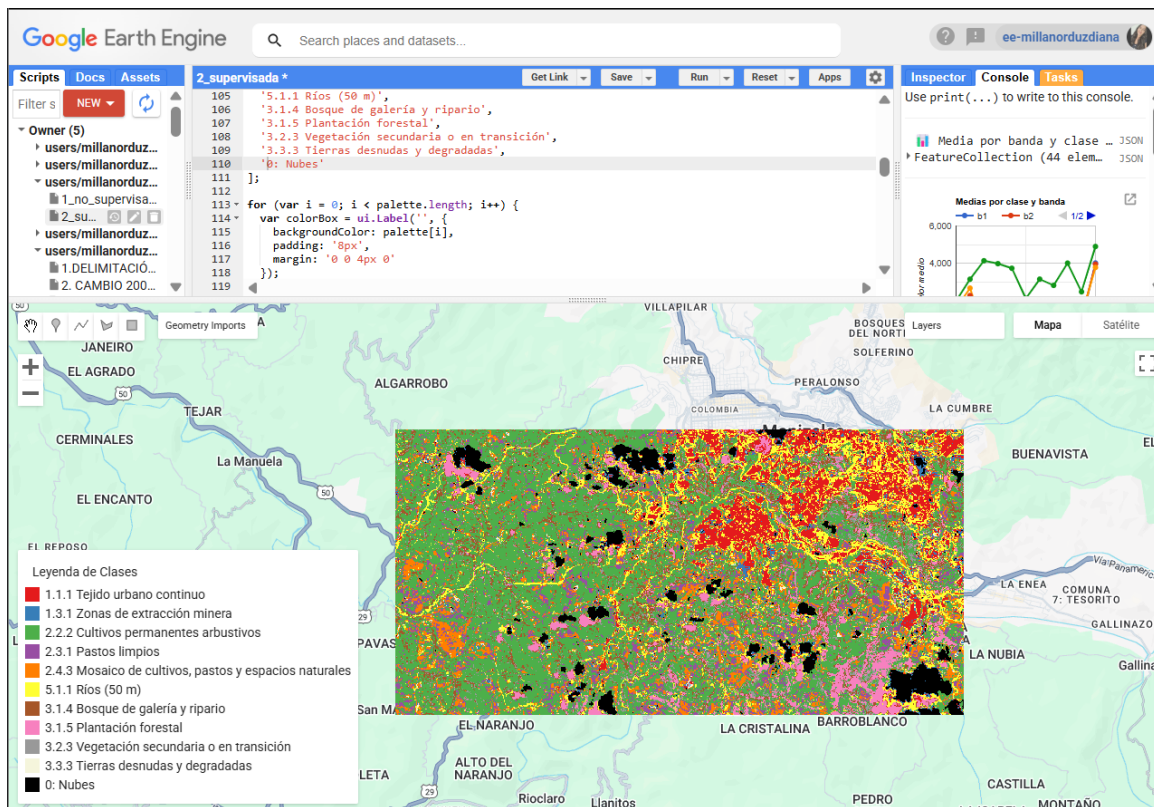
4.1 Clasificación no supervisada: se utilizó el algoritmo KMeans para agrupar los píxeles de la imagen en clusters, sin necesidad de muestras de entrenamiento manuales. El proceso comenzó con la selección de bandas espectrales, específicamente las bandas RGB y NIR. Posteriormente, se extrajeron 5000 píxeles de muestra con una escala espacial de 10 metros. Con estos datos, se entrenó un clasificador utilizando el algoritmo wekaKMeans(6), que generó 6 clusters para la imagen. El resultado de la clasificación fue exportado como una imagen en formato TIF, también con una resolución de 10 metros.

Figura 2. Visualización de clasificación no supervisada



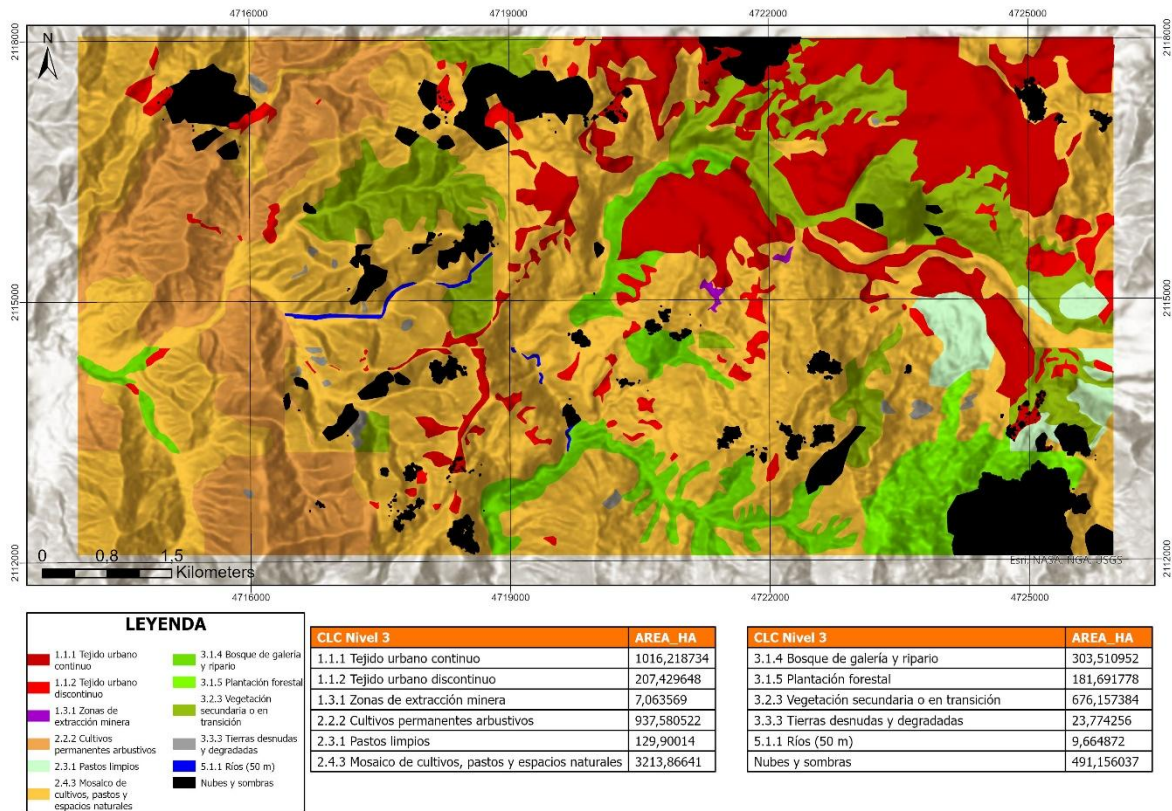
4.2 Clasificación supervisada: se desarrolló utilizando muestras de entrenamiento manuales, lo que permitió un mayor control sobre el resultado. Para ello, se definieron puntos de entrenamiento para 11 clases específicas, que incluyen áreas como zonas urbanas, cultivos y vegetación secundaria, entre otras. A partir de estos puntos de referencia, se entrenó un clasificador Random Forest con 50 árboles, utilizando las bandas espectrales seleccionadas para la clasificación. Este clasificador generó una imagen clasificada en 11 clases. Además, se exportó el resultado de la clasificación en formato TIF.

Figura 3. Visualización de clasificación supervisada



Finalmente, tras obtener la clasificación supervisada, se procedió a un refinamiento de las clases a través de digitalización manual en ArcGIS Pro. Este paso fue crucial para mejorar la precisión de las clases, ajustándolas más estrechamente a la realidad observada en el terreno y asegurando que las etiquetas fueran lo más representativas posible, el resultado visual se muestra en la siguiente Figura 4.

Figura 4. Visualización de coberturas corine land cover nivel 3



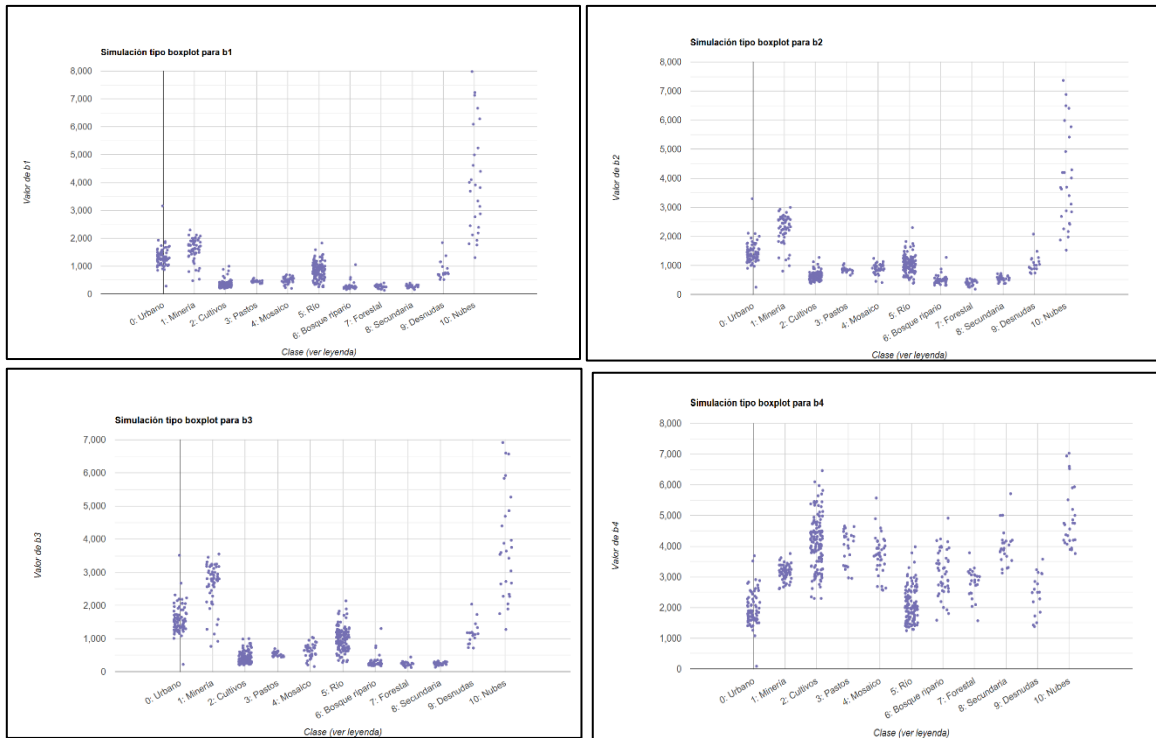
- Script: *3_supervised_classification.js* ; *4_unsupervised_classification.js* ;
- Output: *clasificacion_supervisada.tif* ; *coberturas_CLCN3_revisado.shp*

5. Análisis de bandas espectrales y selección de variables significativas

Se trabajó con las bandas espectrales RGB y NIR, y durante el preprocesamiento se identificó que la normalización era un paso clave. Los valores originales de Sentinel presentan amplias diferencias de escala entre bandas y entre clases, lo que puede afectar negativamente el desempeño del modelo.

La Figura 5 muestra la distribución de valores para las bandas b1 a b4 antes de normalizar. Se observa una gran dispersión, con algunas bandas que superan los 7000 y otras que apenas alcanzan los 500. Esta variabilidad puede inducir sesgos en los algoritmos de clasificación, razón por la cual se optó por normalizar los datos antes del entrenamiento.

Figura 5. Evaluación de variables significativas



- Script: *5_significant_variables.js*

6. Preparación del dataset: imágenes y etiquetas en formato compatible

Una vez seleccionados los tiles y definida la fuente de etiquetado, se generaron imágenes individuales por tile, acompañadas de sus respectivas etiquetas en formato GeoJSON. Este formato fue seleccionado por su compatibilidad con herramientas SIG como ArcGIS y QGIS, las cuales ofrecen funcionalidades avanzadas para la digitalización y edición de polígonos de manera precisa y eficiente.

Además, el uso de GeoJSON permite posteriormente adaptar las etiquetas a distintos entornos de entrenamiento en notebooks de Jupyter, ya sea para detección, segmentación o clasificación. Dependiendo del requerimiento del modelo, las etiquetas pueden convertirse fácilmente a formatos como .txt, .jpg o máscaras raster, lo cual garantiza una alta flexibilidad del dataset construido.

- Script: *6_exportar_labels_images.ipynb*
- Output: Carpeta de “dataset” con “images” en formato .tif y “labels” en formato .geojson

7. Selección de clases para entrenamiento inicial

Para el entrenamiento inicial se seleccionaron dos clases representativas, con el objetivo de optimizar los tiempos de cómputo y responder a la disponibilidad limitada de datos etiquetados. Entrenar el modelo con la totalidad de las clases implicaría un mayor costo computacional y requeriría una curación más exhaustiva de las muestras, lo cual podría ralentizar el proceso.

Al trabajar con un número reducido de clases, fue posible evaluar el desempeño del modelo en condiciones controladas, permitiendo validar la estrategia metodológica sin comprometer los recursos disponibles.

Este procedimiento facilita la futura replicabilidad del ejercicio para las demás categorías de nivel 3 del Corine Land Cover, y también demuestra mis habilidades en la formulación, ajuste y entrenamiento progresivo de modelos supervisados, priorizando tanto la eficiencia como la calidad del resultado final.

8. Entrenamiento del modelo y consideraciones técnicas

Para abordar la tarea de segmentación de imágenes, se optó por utilizar la arquitectura U-Net debido a su eficacia comprobada en este tipo de problemas. U-Net es una red neuronal convolucional diseñada específicamente para la segmentación semántica, lo cual según [Ultralytics](#) se destaca por su estructura en forma de "U" que facilita la localización precisa y la segmentación de objetos en imágenes, incluso cuando se dispone de un número limitado de datos de entrenamiento.

8.1 Estructuración y Preparación de Datos: Dado que U-Net es una arquitectura orientada a la segmentación semántica, se estructuraron los datos de entrada y sus correspondientes máscaras de segmentación para adaptarse a este modelo. Cada imagen de entrada fue emparejada con su respectiva máscara, donde cada píxel de la máscara indica la clase correspondiente del píxel en la imagen de entrada. Esta preparación es esencial para que el modelo aprenda a predecir la clase de cada píxel en las imágenes durante el proceso de entrenamiento.

8.2 Aumento de Datos: Para enriquecer el conjunto de datos y mejorar la generalización del modelo, se implementaron técnicas de aumento de datos. Estas técnicas incluyeron transformaciones como rotaciones, desplazamientos, escalados y cambios de brillo y contraste, aplicadas aleatoriamente a las imágenes y sus máscaras correspondientes. Este enfoque permitió al modelo ser más robusto y menos propenso al sobreajuste.

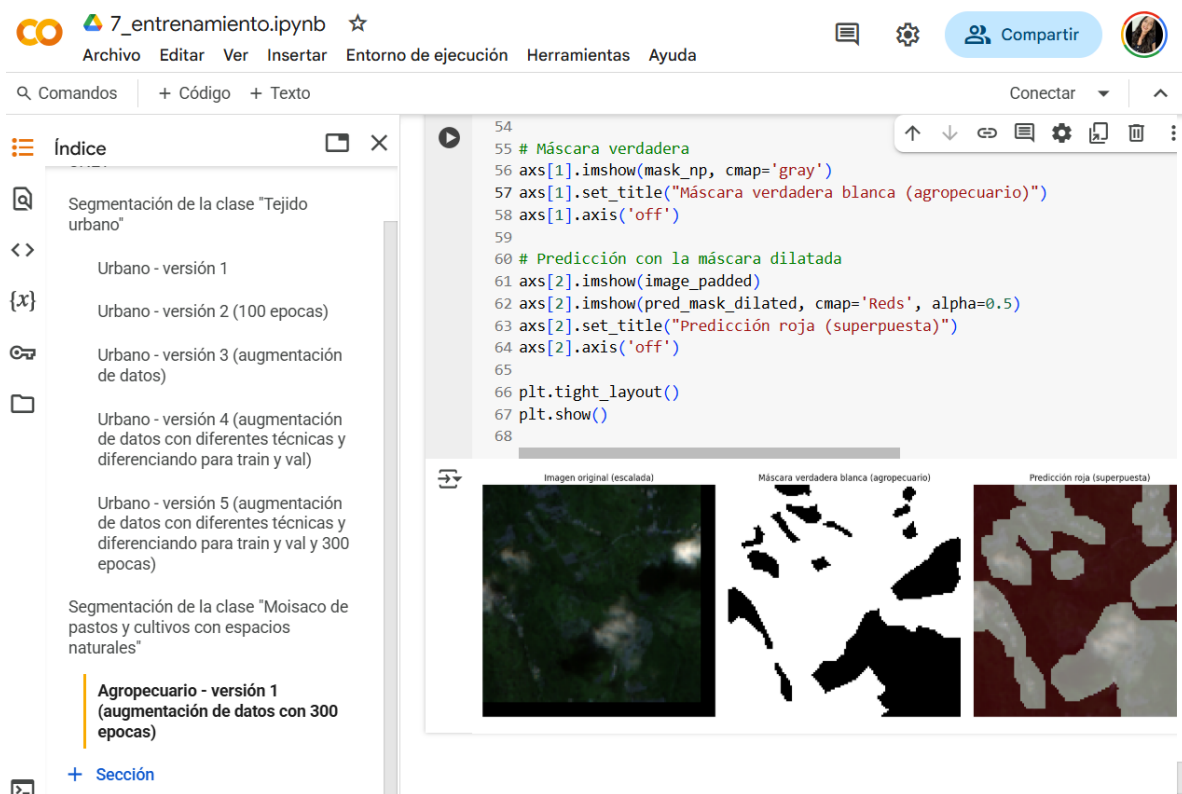
8.3 Proceso de Entrenamiento y Evaluación: El entrenamiento de los modelos se llevó a cabo utilizando Google Colab, aprovechando su capacidad de procesamiento gratuito. Se estableció un total de 300 épocas para el entrenamiento, con el objetivo de permitir que el modelo convergiera adecuadamente. Durante este proceso, se monitorizó:

- La función de pérdida en el conjunto de validación (validation loss) para evaluar el desempeño del modelo y ajustar los hiperparámetros según fuera necesario.

- Se emplearon métricas específicas para segmentación, como el Índice de Solapamiento de Jaccard (IoU, por sus siglas en inglés). El IoU mide la intersección entre la predicción del modelo y la verdad terreno, dividida por la unión de ambas. Esta métrica es fundamental para evaluar la precisión de la segmentación, especialmente en escenarios donde las clases son desbalanceadas.
- Se generaron matrices de confusión para cada clase, permitiendo analizar detalladamente el rendimiento del modelo en términos de verdaderos positivos, falsos positivos, verdaderos negativos y falsos negativos. Estas matrices proporcionaron información valiosa sobre las fortalezas y debilidades del modelo en la clasificación de cada categoría.
- Evaluación directa sobre imágenes para una inspección visual de las segmentaciones realizadas por el modelo, facilitando la identificación de áreas donde el modelo podría requerir ajustes adicionales o más datos de entrenamiento.

Es importante resaltar, que en el notebook de entrenamiento se realizaron varias pruebas para el caso de la clasificación de tejido urbano y menos para el mosaico debido a que se aplicaron las mismas configuraciones que se vio que funcionaron en el de tejido urbano, adecuando ciertas especificaciones para mejorar la clasificación de la clase de mosaicos. Estas diferentes pruebas y especificaciones se pueden evidenciar en cada título en la sección “Índice” como muestra la siguiente figura

Figura 6. Visualización de la estructuración de entrenamiento



- Script: 7_entrenamiento.ipynb

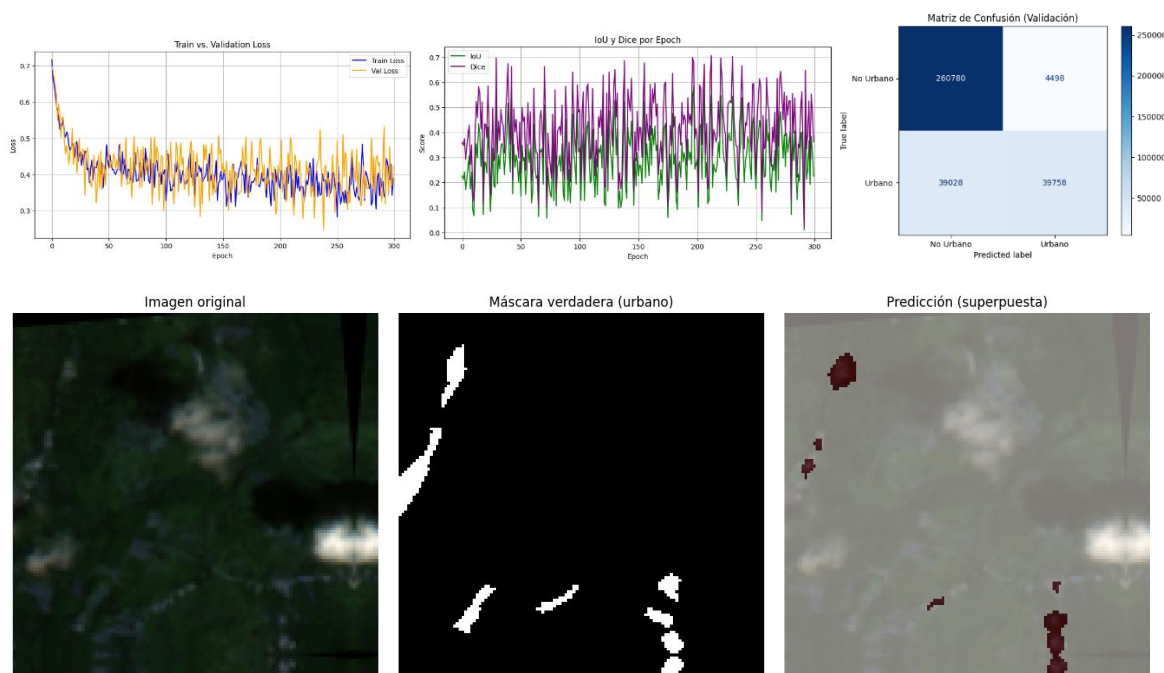
9. Evaluación de resultados

9.1 Evaluación de la clase urbana:

El modelo mostró una rápida convergencia inicial con una caída pronunciada en la pérdida (loss), estabilizándose luego entre 0.35 y 0.45 en entrenamiento y validación. La proximidad entre train_loss y val_loss sugiere que el modelo generaliza adecuadamente, sin evidencias de sobreajuste, a pesar de las oscilaciones propias del tamaño reducido del dataset.

Las métricas de desempeño refuerzan este comportamiento: tanto el IoU (0.2 a 0.45, con picos cercanos a 0.5) como el Dice Score (frecuentemente por encima de 0.6 y superando 0.7 en algunos casos) indican que el modelo ha logrado identificar correctamente las áreas urbanas, aunque con cierta variabilidad atribuida a la heterogeneidad de las clases.

Esta conclusión se ve respaldada por la precisión urbana del 89.8% observada en la matriz de confusión, que resalta la capacidad del modelo para identificar zonas urbanas, incluso siendo una clase minoritaria. Sin embargo, la exactitud urbana del 50.5% señala que aún hay ejemplos no detectados, lo que refleja un potencial de mejora en la sensibilidad del modelo. Finalmente, la visualización de los resultados indicó que el modelo fue capaz de identificar correctamente las zonas urbanas más destacadas, con una alineación precisa en las áreas principales. No obstante, algunos detalles urbanos más pequeños fueron omitidos, lo cual es esperado debido a la resolución de las imágenes y las limitaciones del modelo en cuanto a la cantidad de los datos etiquetados que fueron mitigados con la generación de nuevas imágenes pero no evita que se requieran muchos más datos para que el modelo aprenda realmente. De esta forma se evidencia que, aunque el modelo es efectivo para la tarea, existe un margen de mejora en la detección de detalles más pequeños y en el ajuste fino de la sensibilidad del modelo.

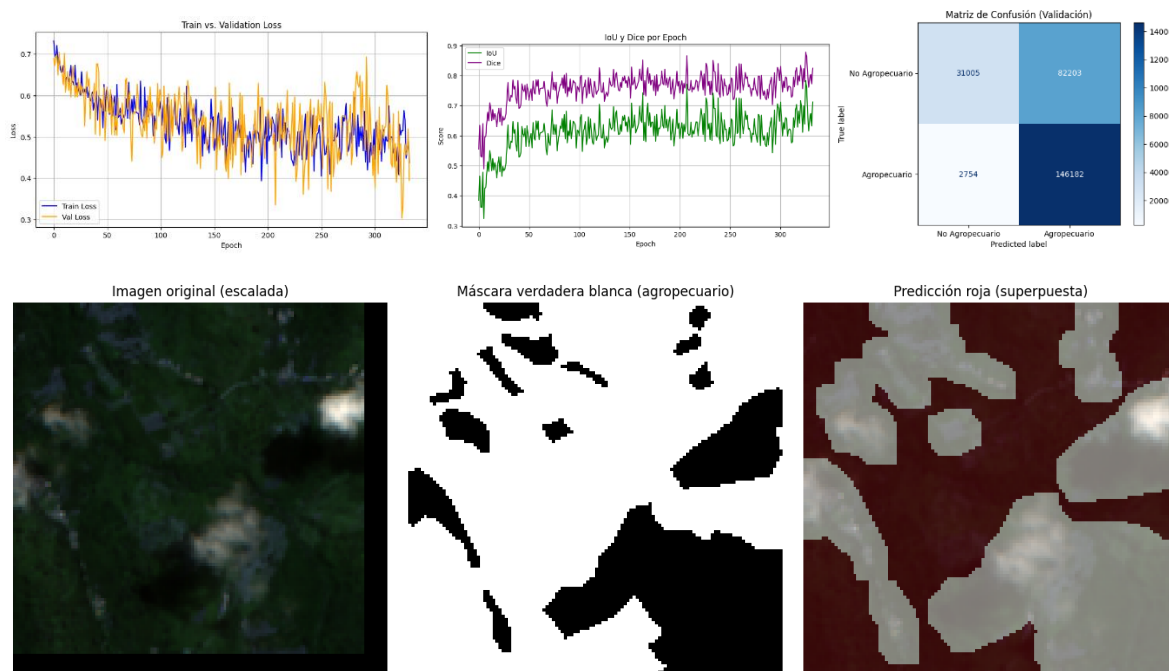


9.2 Evaluación de la clase agrícola:

El modelo mostró un comportamiento mixto al segmentar la clase agropecuario. Aunque la pérdida de entrenamiento y validación disminuyó al principio, se observó una mayor variabilidad en la pérdida de validación, lo que sugiere que el modelo tuvo más dificultades para generalizar en comparación con la clase urbano. Esto es reflejado en las métricas de IoU y Dice, donde el IoU se mantuvo entre 0.3 y 0.5, y el Dice alcanzó picos entre 0.6 y 0.8, indicando que el modelo capturó patrones agropecuarios, pero no con la misma precisión que con la clase urbana.

En términos de precisión, la clase agropecuaria obtuvo una tasa del 50.5%, significativamente más baja que la precisión para la clase urbana. Esto muestra que el modelo tuvo más dificultades para identificar correctamente las zonas agropecuarias, con un número mayor de falsos positivos en las predicciones.

Visualmente, aunque el modelo detectó varias áreas agropecuarias, las predicciones de estas zonas fueron menos precisas en comparación con las zonas urbanas. Algunas áreas pequeñas y fragmentadas de agropecuario fueron omitidas, lo que refleja las dificultades del modelo para segmentar correctamente este tipo de coberturas, especialmente en paisajes más complejos o menos definidos.



RECOMENDACIONES Y PRÓXIMOS PASOS

Los resultados obtenidos permiten validar la metodología empleada y sientan las bases para ampliar el entrenamiento del modelo. Como siguiente paso, se recomienda replicar este proceso sobre una extensión territorial más amplia, utilizando imágenes satelitales libres de nubosidad y un conjunto de clases representativas más equilibrado y completo, en concordancia con la clasificación del Corine Land Cover Nivel 3. Este enfoque permitirá entrenar un modelo generalizable, capaz de clasificar las coberturas identificadas en la región de Manizales (área en la que fue enviada la imagen).

Asimismo, se reafirma que la elaboración de etiquetas es uno de los pasos más críticos del proceso, ya que define la calidad del entrenamiento. Por ello, la combinación de clasificación supervisada y no supervisada resultó clave para optimizar la generación de etiquetas de manera eficiente.

A esto se suma la importancia del análisis realizado en el paso 3, donde se identificaron las coberturas presentes mediante la referencia Corine. Este ejercicio funcionó como una especie de pre-muestreo virtual, que permitió comprender la variabilidad ecológica del área, cómo se distribuyen las clases y cuáles requieren mayor priorización al momento de etiquetar, especialmente aquellas menos representadas territorialmente. Estos insumos son fundamentales para estructurar un modelo más preciso y escalable.