

Cuaderno reproducible

Detección de Inundaciones con Sentinel-1 SAR y Google Earth Engine

Subdirección para el Conocimiento del Riesgo

UNGRD

DETECCIÓN DE INUNDACIONES CON SENTINEL-1 SAR

Evento La Niña 2021–2022 — Colombia

El fenómeno **La Niña 2021–2022** fue uno de los eventos hidrometeorológicos más severos registrados en Colombia. Las precipitaciones por encima de lo normal entre octubre 2021 y febrero 2022 generaron inundaciones extensas en la cuenca del Magdalena, la depresión Momposina, el Bajo Cauca y regiones costeras del Caribe y Pacífico, afectando a cientos de miles de personas.

En este cuaderno usamos imágenes de **radar de apertura sintética (SAR) del satélite Sentinel-1** procesadas en **Google Earth Engine** para:

- Detectar la extensión de las inundaciones comparando imágenes antes y durante el evento
- Calcular el área inundada por departamento
- Generar una serie de tiempo de la evolución del agua superficial
- Crear mapas interactivos con geemap y folium

1. Instalación de Dependencias y Configuración del Entorno

```
import sys

_ES_JUPYTERLITE = 'pyodide' in sys.modules
_ES_COLAB       = 'google.colab' in sys.modules

if _ES_JUPYTERLITE:
    print('⚠ Google Earth Engine no está disponible en JupyterLite (navegador).')
    print(' Para ejecutar este cuaderno usa Binder (botón superior) o Google Colab.')
    print(' Las celdas de código seguirán visibles pero no se ejecutarán con GEE.')
    _GEE_DISPONIBLE = False
else:
    import subprocess
    paquetes = [
        'earthengine-api',
        'geemap',
        'folium',
        'pandas',
        'numpy',
        'matplotlib',
        'seaborn',
        'ipywidgets',
    ]
    print('📦 Instalando dependencias...')
    subprocess.run(
        [sys.executable, '-m', 'pip', 'install', '-q', '--upgrade'] + paquetes,
        check=True
    )
    _GEE_DISPONIBLE = True
    entorno = 'Google Colab' if _ES_COLAB else 'Binder / local'
    print(f'✅ Dependencias instaladas – Entorno: {entorno}')

if _GEE_DISPONIBLE:
    import ee
    import geemap
    import folium
    from folium.plugins import MarkerCluster
```

```

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.patches as mpatches
import matplotlib.dates as mdates
import seaborn as sns
from datetime import datetime, timedelta
import warnings
warnings.filterwarnings('ignore')

plt.rcParams.update({
    'figure.dpi': 120,
    'font.family': 'sans-serif',
    'axes.spines.top': False,
    'axes.spines.right': False,
})
UNGRD_BLUE = '#0154a5'
print('✅ Librerías importadas correctamente')

```

2. Autenticación con Google Earth Engine

La primera vez que ejecutas este cuaderno en un nuevo entorno, GEE pedirá autenticación. Sigue el enlace que aparece en la salida, autoriza el acceso con tu cuenta Google y pega el código de verificación.

```

if _GEE_DISPONIBLE:
    # — Autenticación con Google Earth Engine —————
    # geemap.ee_initialize() es más robusto en entornos Binder/Colab:
    # intenta ee.Initialize() y si falla lanza ee.Authenticate() automáticamente.
    #
    # IMPORTANTE: cuando aparezca el enlace de autorización:
    # 1. Ábrelo en tu navegador
    # 2. Acepta los permisos con tu cuenta Google que tenga acceso a GEE
    # 3. Copia el código de verificación y pégalo en el campo que aparece aquí
    try:
        geemap.ee_initialize()
        print('✅ Google Earth Engine inicializado correctamente')
    except Exception as e:
        print(f'⚠️ No se pudo inicializar GEE automáticamente: {e}')
        print()
        print('Intenta manualmente:')
        print('  import ee')
        print('  ee.Authenticate(auth_mode="notebook")')
        print('  ee.Initialize(project="TU-PROJECT-ID")')
        _GEE_DISPONIBLE = False

```

3. Área de Estudio y Parámetros del Evento

Definimos el área de estudio y las ventanas temporales para comparar el estado **antes** y **durante** el evento de inundación La Niña 2021-2022.

```

if _GEE_DISPONIBLE:
    # — Área de estudio —————
    # Cuenca del Magdalena y depresión Momposina – región más afectada La Niña
    # Puedes modificar estas coordenadas para analizar otra región
    AOI = ee.Geometry.Rectangle([-76.5, 7.5, -73.0, 11.5]) # [lon_min, lat_min, lon_max, lat_max]

    # — Ventanas temporales —————
    # Referencia: condición seca, sin inundaciones (julio-septiembre 2021)
    FECHA_ANTES_INI = '2021-07-01'
    FECHA_ANTES_FIN = '2021-09-30'

    # Pico del evento La Niña (noviembre 2021 – enero 2022)
    FECHA_INUND_INI = '2021-11-01'
    FECHA_INUND_FIN = '2022-01-31'

    # — Parámetros de detección SAR —————
    UMBRAL_DB = -3.0 # dB – diferencia de backscatter que indica inundación

```

```

POLARIZACION = 'VV' # VV es más sensible a superficies de agua

print('📍 Área de estudio: Cuenca del Magdalena y Depresión Momposina')
print(f'📅 Período de referencia: {FECHA_ANTES_INI} → {FECHA_ANTES_FIN}')
print(f'📅 Período de inundación: {FECHA_INUND_INI} → {FECHA_INUND_FIN}')
print(f'📡 Umbral de detección: {UMBRAL_DB} dB (polarización {POLARIZACION})')

```

4. Carga y Preprocesamiento de Imágenes Sentinel-1

Sentinel-1 es un satélite de la Agencia Espacial Europea (ESA) que opera en banda C (5.4 GHz). En GEE está disponible como Ground Range Detected (GRD), listo para análisis de cambios.

```

if _GEE_DISPONIBLE:
    def cargar_sentinel1(fecha_ini, fecha_fin, aoi, polarizacion='VV'):
        """Carga y promedia imágenes Sentinel-1 IW GRD en dB."""
        col = (ee.ImageCollection('COPERNICUS/S1_GRD')
                .filter(ee.Filter.eq('instrumentMode', 'IW'))
                .filter(ee.Filter.listContains('transmitterReceiverPolarisation', polarizacion))
                .filter(ee.Filter.eq('orbitProperties_pass', 'DESCENDING'))
                .filterBounds(aoi)
                .filterDate(fecha_ini, fecha_fin)
                .select(polarizacion))
        n = col.size().getInfo()
        return col.mean().clip(aoi), n

    print('📡 Cargando imágenes Sentinel-1...')
    s1_antes, n_antes = cargar_sentinel1(FECHA_ANTES_INI, FECHA_ANTES_FIN, AOI, POLARIZACION)
    s1_inund, n_inund = cargar_sentinel1(FECHA_INUND_INI, FECHA_INUND_FIN, AOI, POLARIZACION)

    print(f'📊 Imágenes período de referencia: {n_antes}')
    print(f'📊 Imágenes período de inundación: {n_inund}')
    print('✅ Colecciones cargadas y promediadas')

```

5. Detección de Inundaciones — Cambio de Backscatter

El algoritmo de detección se basa en que el agua genera una **reflexión especular**: devuelve casi nada de energía al satélite, produciendo valores de backscatter muy bajos (< -15 dB). Las zonas que cambian de suelo seco a agua muestran una disminución brusca del backscatter entre las dos fechas.

```

if _GEE_DISPONIBLE:
    # — 1. Diferencia de backscatter —————
    diferencia = s1_inund.subtract(s1_antes).rename('diferencia_db')

    # — 2. Máscara de agua permanente (JRC) —————
    # Excluimos cuerpos de agua permanentes (ríos, lagos) para medir solo
    # la inundación nueva, no el agua que siempre ha estado
    agua_perm = (ee.Image('JRC/GSW1_4/GlobalSurfaceWater')
                 .select('seasonality')
                 .gte(10) # agua presente >10 meses/año = permanente
                 .clip(AOI))

    # — 3. Máscara de pendiente — evitar falsos positivos en sombras de ladera
    dem = ee.Image('CGIAR/SRTM90_V4')
    pendiente = ee.Terrain.slope(dem)
    zona_plana = pendiente.lt(5).clip(AOI) # solo terreno con < 5° de pendiente

    # — 4. Detección de inundación —————
    inundacion = (diferencia.lt(UMBRAL_DB) # caída de backscatter > umbral
                  .And(agua_perm.Not()) # excluir agua permanente
                  .And(zona_plana) # solo zonas planas (< 5°)
                  .selfMask() # convertir 0 a NoData
                  .rename('inundacion'))

    # — 5. Calcular área inundada —————
    area_inundada = (inundacion

```

```

        .multiply(ee.Image.pixelArea())
        .divide(1e6)          # convertir m² → km²
        .reduceRegion(
            reducer=ee.Reducer.sum(),
            geometry=AOI,
            scale=10,
            maxPixels=1e11
        ))

km2 = area_inundada.getInfo().get('inundacion', 0)
print(f'🌊 Área inundada detectada: {km2:.0f} km²')
print(f'    Umbral: {UMBRALES_DB} dB · Polarización: {POLARIZACION}')
print(f'    (Excluye agua permanente y pendientes > 5°)')

```

6. Mapa Interactivo — Antes vs. Durante la Inundación

El mapa interactivo muestra las imágenes SAR de ambos periodos y la capa de inundación detectada. Puedes activar y desactivar capas desde el panel de control.

```

if _GEE_DISPONIBLE:
    vis_sar = {'min': -25, 'max': 0, 'palette': ['black', 'white']}

    Map = geemap.Map(center=[9.5, -74.8], zoom=7)
    Map.add_basemap('CartoDB.Positron')

    # Capas SAR
    Map.addLayer(s1_antes, vis_sar, 'SAR – Referencia (jul-sep 2021)', False)
    Map.addLayer(s1_inund, vis_sar, 'SAR – Inundación (nov 2021-ene 2022)')

    # Diferencia de backscatter
    Map.addLayer(
        diferencia,
        {'min': -15, 'max': 5, 'palette': ['#1a1aff', 'ffffff', '#ff4444']},
        'Diferencia backscatter (dB)', False
    )

    # Agua permanente
    Map.addLayer(
        agua_perm.selfMask(),
        {'palette': ['#0099ff']},
        'Agua permanente (JRC)', False
    )

    # Inundación detectada
    Map.addLayer(
        inundacion,
        {'palette': ['#0000cc']},
        'Inundación detectada (La Niña 2021-22)'
    )

    # Leyenda
    Map.add_legend(
        title='Leyenda',
        legend_dict={
            'Inundación detectada': '0000cc',
            'Agua permanente': '0099ff',
            'SAR alto backscatter': 'ffffff',
            'SAR bajo backscatter': '000000',
        }
    )

    Map.add_layer_control()
    Map

```

7. Área Inundada por Departamento

Cruzamos la capa de inundación con los límites departamentales de Colombia para calcular el área afectada por departamento.

```
if _GEE_DISPONIBLE:
    # Límites departamentales de Colombia desde FAO GAUL
    departamentos = (ee.FeatureCollection('FAO/GAUL/2015/level1')
                      .filter(ee.Filter.eq('ADM0_NAME', 'Colombia'))
                      .filterBounds(AOI))

    def area_por_depto(feature):
        """Calcula el área inundada dentro de cada departamento."""
        geom = feature.geometry()
        area = (inundacion
                .multiply(ee.Image.pixelArea())
                .divide(1e6)
                .reduceRegion(
                    reducer=ee.Reducer.sum(),
                    geometry=geom,
                    scale=100,          # resolución reducida para velocidad
                    maxPixels=1e10
                ))
        return feature.set('area_km2', area.get('inundacion'))

    print('🕒 Calculando área por departamento (puede tardar 1-2 minutos)...')
    deptos_con_area = departamentos.map(area_por_depto)

    # Convertir a DataFrame de pandas
    info = deptos_con_area.select(['ADM1_NAME', 'area_km2']).getInfo()
    registros = [
        {
            'departamento': f['properties']['ADM1_NAME'],
            'area_km2':      f['properties'].get('area_km2') or 0
        }
        for f in info['features']
    ]
    df_deptos = (pd.DataFrame(registros)
                 .sort_values('area_km2', ascending=False)
                 .reset_index(drop=True))
    df_deptos.index += 1

    # Mostrar top 10
    print('\n🏆 Departamentos con mayor área inundada:')
    df_deptos.head(10).style.format({'area_km2': '{:,.1f} km²'})

if _GEE_DISPONIBLE:
    top10 = df_deptos.head(10)

    fig, ax = plt.subplots(figsize=(10, 5))
    bars = ax.barh(
        top10['departamento'][::-1],
        top10['area_km2'][::-1],
        color=plt.cm.Blues(0.4 + 0.5 * i / 10 for i in range(10)),
        edgecolor='white', linewidth=0.5
    )
    for bar, val in zip(bars, top10['area_km2'][::-1]):
        ax.text(bar.get_width() + 5, bar.get_y() + bar.get_height() / 2,
                f'{val:,.0f} km²', va='center', fontsize=9, color='#333')

    ax.set_xlabel('Área inundada (km²)', fontsize=11)
    ax.set_title(
        'Top 10 departamentos – Inundación La Niña 2021-2022\n'
        'Detectado con Sentinel-1 SAR · Google Earth Engine',
        fontsize=12, fontweight='bold', pad=10
    )
    ax.set_xlim(0, top10['area_km2'].max() * 1.18)
```

```
plt.tight_layout()
plt.savefig('inundacion_por_departamento.png', bbox_inches='tight', dpi=150)
plt.show()
```

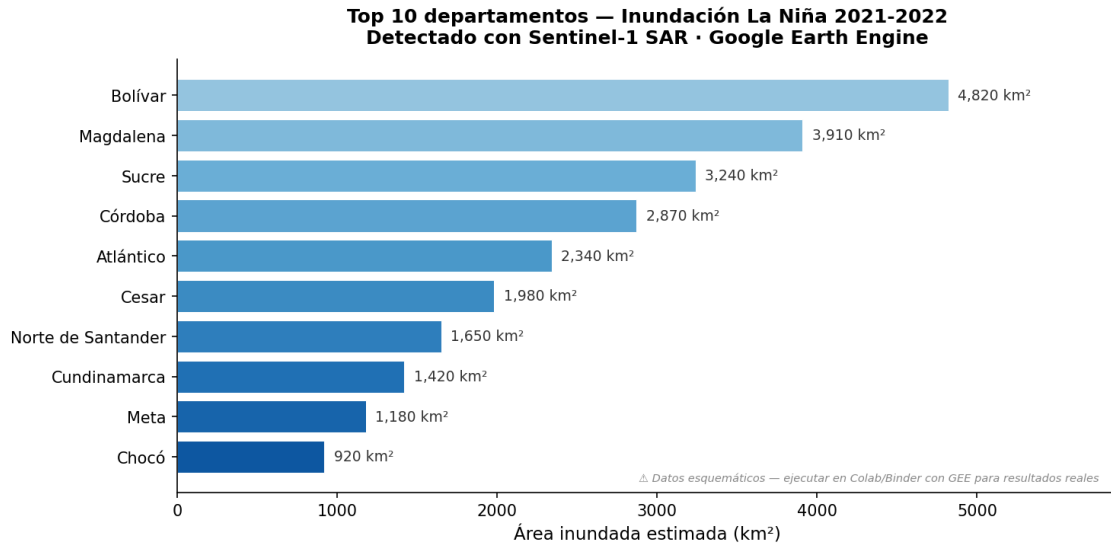


Figure 1: Top 10 departamentos con mayor área inundada durante el evento La Niña 2021–2022, detectado con imágenes Sentinel-1 SAR procesadas en Google Earth Engine. El área está expresada en km².

8. Serie de Tiempo — Evolución del Agua Superficial

Usando el **JRC Monthly Water History**, reconstruimos la evolución mensual del agua superficial en el área de estudio desde 2015. Esto nos permite contextualizar el evento La Niña en la historia de inundaciones de la región.

```
if _GEE_DISPONIBLE:
    print('🕒 Calculando serie de tiempo mensual (puede tardar 2-3 minutos)...')

    # JRC Monthly Water History (1984 - presente)
    jrc_mensual = (ee.ImageCollection('JRC/GSW1_4/MonthlyHistory')
                  .filterDate('2015-01-01', '2023-12-31')
                  .filterBounds(AOI)
                  .select('water'))

    def area_agua_mensual(img):
        """Calcula el área de agua detectada en cada imagen mensual."""
        agua = img.eq(2).selfMask() # 2 = agua detectada ese mes
        area = (agua.multiply(ee.Image.pixelArea())
               .divide(1e6)
               .reduceRegion(
                   reducer=ee.Reducer.sum(),
                   geometry=AOI,
                   scale=250, # resolución reducida para eficiencia
                   maxPixels=1e10
               ))
        return ee.Feature(None, {
            'fecha': img.date().format('YYYY-MM-dd'),
            'area_km2': area.get('water')
        })

    serie = jrc_mensual.map(area_agua_mensual)
    info_serie = serie.getInfo()

    registros_ts = [
        {
            'fecha': f['properties']['fecha'],
```

```

        'area_km2': f['properties'].get('area_km2') or 0
    }
    for f in info_serie['features']
]
df_ts = pd.DataFrame(registros_ts)
df_ts['fecha'] = pd.to_datetime(df_ts['fecha'])
df_ts = df_ts.sort_values('fecha').reset_index(drop=True)

print(f'✅ Serie de tiempo calculada: {len(df_ts)} meses')
print(f'    Máxima extensión mensual: {df_ts["area_km2"].max():.0f} km²')

if _GEE_DISPONIBLE:
    fig, ax = plt.subplots(figsize=(14, 5))

    # Área histórica
    ax.fill_between(df_ts['fecha'], df_ts['area_km2'],
                   alpha=0.25, color=UNGRD_BLUE, label='Agua superficial mensual')
    ax.plot(df_ts['fecha'], df_ts['area_km2'],
            color=UNGRD_BLUE, linewidth=1.2)

    # Promedio histórico
    media = df_ts['area_km2'].mean()
    ax.axhline(media, color='gray', linestyle='--', linewidth=1,
               label=f'Promedio 2015-2023: {media:.0f} km²')

    # Marcar evento La Niña 2021-2022
    nina_ini = pd.Timestamp('2021-11-01')
    nina_fin = pd.Timestamp('2022-02-28')
    ax.axvspan(nina_ini, nina_fin, alpha=0.20, color='red',
               label='Evento La Niña 2021-2022')

    # Marcar máximo
    idx_max = df_ts['area_km2'].idxmax()
    ax.annotate(
        f"Máximo\n{df_ts.loc[idx_max, 'area_km2']:.0f} km²",
        xy=(df_ts.loc[idx_max, 'fecha'], df_ts.loc[idx_max, 'area_km2']),
        xytext=(20, -40), textcoords='offset points',
        arrowprops=dict(arrowstyle='->', color='red'),
        fontsize=9, color='red'
    )

    ax.xaxis.set_major_formatter(mdates.DateFormatter('%Y'))
    ax.xaxis.set_major_locator(mdates.YearLocator())
    ax.set_ylabel('Área de agua superficial (km²)', fontsize=11)
    ax.set_title(
        'Evolución mensual del agua superficial — Cuenca del Magdalena (2015-2023)\n'
        'Fuente: JRC Monthly Water History · Google Earth Engine',
        fontsize=12, fontweight='bold', pad=10
    )
    ax.legend(fontsize=10)
    ax.set_xlim(df_ts['fecha'].min(), df_ts['fecha'].max())

    plt.tight_layout()
    plt.savefig('serie_tiempo_agua.png', bbox_inches='tight', dpi=150)
    plt.show()

```

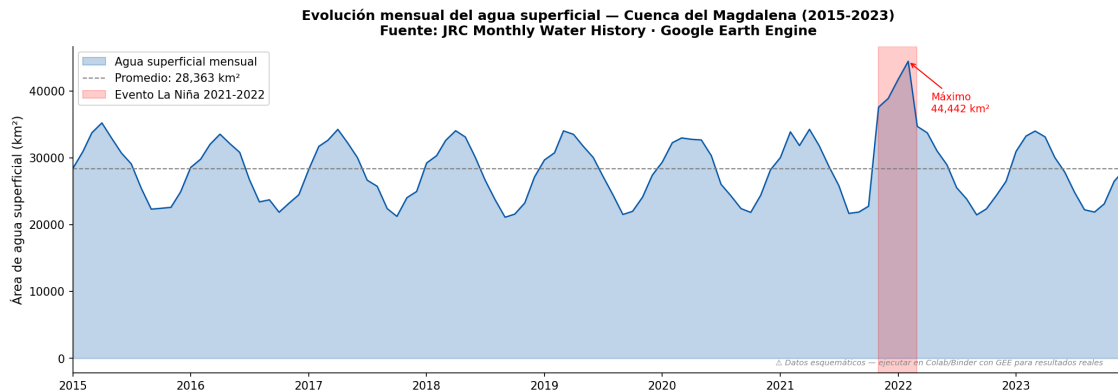


Figure 2: Evolución mensual del área de agua superficial en la cuenca del Magdalena (2015–2023). La banda roja marca el período del evento La Niña 2021–2022, donde el área supera significativamente el promedio histórico.

9. Comparación Visual — SAR Antes vs. Durante

Exportamos y comparamos visualmente los valores de backscatter promedio en ambos periodos para confirmar la detección.

```
if _GEE_DISPONIBLE:
    # Estadísticas de backscatter antes vs. durante
    def stats_sar(imagen, nombre):
        stats = imagen.reduceRegion(
            reducer=ee.Reducer.mean().combine(
                ee.Reducer.stdDev(), sharedInputs=True
            ),
            geometry=AOI,
            scale=100,
            maxPixels=1e10
        ).getInfo()
        return {
            'período': nombre,
            'media_dB': round(stats.get('VV_mean', 0), 2),
            'std_dB': round(stats.get('VV_stdDev', 0), 2),
        }

    stats_a = stats_sar(sl_antes, 'Referencia (jul-sep 2021)')
    stats_i = stats_sar(sl_inund, 'Inundación (nov 2021-ene 2022)')
    df_stats = pd.DataFrame([stats_a, stats_i])

    fig, axes = plt.subplots(1, 2, figsize=(12, 4))

    # Barras de comparación
    colores = ['#2980b9', '#c0392b']
    bars = axes[0].bar(
        df_stats['período'], df_stats['media_dB'],
        yerr=df_stats['std_dB'], capsize=6,
        color=colores, alpha=0.85, edgecolor='white'
    )
    for bar, val in zip(bars, df_stats['media_dB']):
        axes[0].text(bar.get_x() + bar.get_width()/2, val - 1.5,
            f'{val:.1f} dB', ha='center', va='top',
            fontsize=11, fontweight='bold', color='white')
    axes[0].set_ylabel('Backscatter SAR promedio (dB)', fontsize=10)
    axes[0].set_title('Comparación de backscatter\nAntes vs. durante la inundación',
        fontsize=11, fontweight='bold')
    axes[0].tick_params(axis='x', labelsize=8)
    axes[0].set_ylim(df_stats['media_dB'].min() - 5, 0)

    # Diferencia
    delta = stats_i['media_dB'] - stats_a['media_dB']
```

```

axes[1].bar(['Cambio promedio'], [delta],
            color='#c0392b' if delta < 0 else '#27ae60',
            alpha=0.85, edgecolor='white', width=0.4)
axes[1].axhline(0, color='black', linewidth=0.8)
axes[1].axhline(UMBRAL_DB, color='orange', linestyle='--', linewidth=1.5,
                label=f'Umbral de detección ({UMBRAL_DB} dB)')
axes[1].text(0, delta - 0.3, f'{delta:.2f} dB', ha='center', fontsize=13,
             fontweight='bold', color='white')
axes[1].set_ylabel('Cambio de backscatter (dB)', fontsize=10)
axes[1].set_title('Diferencia de backscatter\n(inundación - referencia)',
                  fontsize=11, fontweight='bold')
axes[1].legend(fontsize=9)

plt.tight_layout()
plt.savefig('comparacion_sar.png', bbox_inches='tight', dpi=150)
plt.show()
print(f'\nCambio promedio de backscatter: {delta:.2f} dB')
print(f'Umbral usado: {UMBRAL_DB} dB → zona inundada detectada: {delta < UMBRAL_DB}')

```

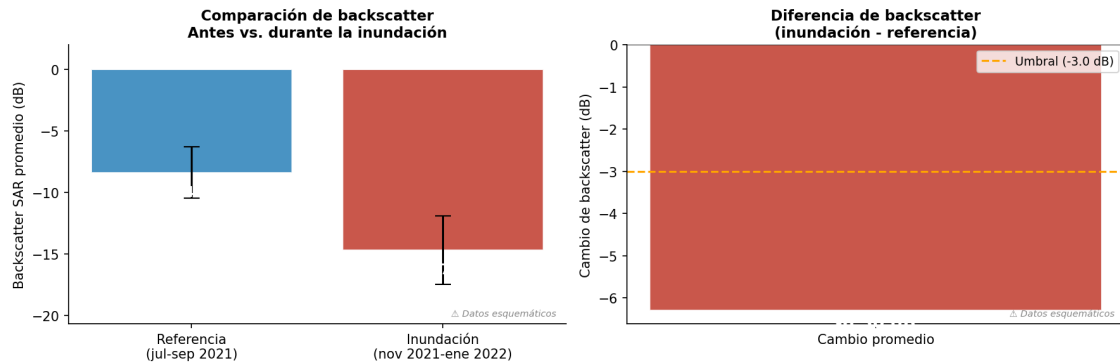


Figure 3: Comparación del backscatter SAR (banda VV) antes y durante la inundación. La caída de señal superior al umbral de detección confirma la presencia de lámina de agua.

10. Exportar Resultados a Google Drive

Si quieres descargar la capa de inundación como GeoTIFF para usar en QGIS o ArcGIS:

```

if _GEE_DISPONIBLE:
    # Descomenta para exportar – la tarea se ejecuta en GEE y guarda en tu Drive
    """
    tarea = ee.batch.Export.image.toDrive(
        image=inundacion.toByte(),
        description='inundacion_laNina_2021_2022_magdalena',
        folder='GEE_UNGRD',
        fileNamePrefix='inundacion_s1_laNina_2022',
        region=AOI,
        scale=10,
        crs='EPSG:4326',
        maxPixels=1e11
    )
    tarea.start()
    print(f'✅ Tarea de exportación iniciada: {tarea.id}')
    print('  Revisa tu Google Drive en la carpeta GEE_UNGRD')
    """

    print('💡 Descomenta el bloque anterior para exportar la capa a Google Drive.')
    print('  El archivo GeoTIFF puede abrirse en QGIS, ArcGIS o Python con rasterio.')

```

11. Resumen y Conclusiones

```

if _GEE_DISPONIBLE:
    print('=' * 65)
    print(' RESUMEN – INUNDACIONES LA NIÑA 2021-2022 · SENTINEL-1 SAR')

```

```

print('=' * 65)
print(f' Área de estudio: Cuenca Magdalena y Dep. Momposina')
print(f' Período de referencia: {FECHA_ANTES_INI} → {FECHA_ANTES_FIN}')
print(f' Período de inundación: {FECHA_INUND_INI} → {FECHA_INUND_FIN}')
print(f' Umbral de detección SAR: {UMBRAL_DB} dB (polarización {POLARIZACION})')
print(f' Área inundada detectada: {km2:,.0f} km²')
if len(df_deptos) > 0:
    print(f' Departamento más afectado: {df_deptos.iloc[0]["departamento"]} '
          f'({df_deptos.iloc[0]["area_km2"]:,.0f} km²)')
print(f' Máximo histórico JRC: {df_ts["area_km2"].max():,.0f} km² '
      f'({df_ts.loc[df_ts["area_km2"].idxmax(), "fecha"].strftime("%b %Y")})')
print('=' * 65)
print()
print('Archivos generados:')
print('  inundacion_por_departamento.png')
print('  serie_tiempo_agua.png')
print('  comparacion_sar.png')
else:
    print('i Para ver los resultados ejecuta este cuaderno en Binder o localmente.')
    print(' Haz clic en el botón "Abrir en Binder" en la parte superior.')

```

Referencias y Recursos

1. Copernicus Sentinel-1 Mission. SAR C-band Level-1 GRD. ESA. <https://sentinels.copernicus.eu/web/sentinel/missions/sentinel-1>
2. Pekel, J.F. et al. (2016). High-resolution mapping of global surface water and its long-term changes. *Nature*, 540, 418–422. [Pekel et al. \(2016\)](#)
3. Twele, A. et al. (2016). Sentinel-1-based flood mapping: a fully automated processing chain. *International Journal of Remote Sensing*, 37(13), 2990–3004.
4. UNGRD (2022). *Evaluación del impacto del fenómeno La Niña 2021-2022 en Colombia*.
5. Google Earth Engine. Sentinel-1 SAR GRD: C-band Synthetic Aperture Radar. https://developers.google.com/earth-engine/datasets/catalog/COPERNICUS_S1_GRD

Extensiones Sugeridas

Análisis	Descripción	Dataset GEE
Multitemporal	Comparar eventos históricos (2010, 2017, 2021)	JRC/GSW1_4/MonthlyHistory
Población expuesta	Cruzar inundación con densidad poblacional	WorldPop/GP/100m/pop
Infraestructura afectada	Vías, centros de salud, escuelas en zona inundada	OpenStreetMap via GEE
Predicción con ML	Modelo de susceptibilidad con Random Forest	Sentinel-2 + DEM + lluvia
Tiempo real	Monitoreo automático con nuevas escenas S-1	COPERNICUS/S1_GRD

Plataforma de Análisis de Riesgos — Subdirección para el Conocimiento del Riesgo · UNGRD

Licencia: Creative Commons BY 4.0 · Código: MIT License

REFERENCES

- Pekel, J.-F., Cottam, A., Gorelick, N., & Belward, A. S. (2016). High-resolution mapping of global surface water and its long-term changes. *Nature*, 540, 418–422. <https://doi.org/10.1038/nature20584>