

Cuaderno reproducible

Riesgo por olas de calor urbano en Cartagena, Bolívar bajo escenarios de cambio climático

Subdirección para el Conocimiento del Riesgo

Unidad Nacional para la Gestión del Riesgo de Desastres

Workflow piloto · 6 fases · Marco IPCC AR6 (Hazard × Exposición × Vulnerabilidad)

RESUMEN

Cartagena de Indias, una de las urbes más densas y de mayor exposición al estrés térmico en el Caribe colombiano, opera ya con temperaturas medias anuales superiores a 28 °C y noches tropicales ($T_{\min} \geq 24$ °C) que cubren la mayor parte del año. Bajo cambio climático, la frecuencia y duración de **olas de calor urbano** —definidas como secuencias de ≥ 3 días consecutivos con $T_{\max} \geq 35$ °C y $T_{\min} \geq 24$ °C— pueden incrementarse sustancialmente, con consecuencias críticas para la salud pública, la productividad laboral y la habitabilidad urbana.

Este cuaderno implementa un **workflow piloto de 6 fases** adaptado del [CLIMAAX urban heatwaves workflow](#) al contexto colombiano, alineado con el marco IPCC AR6 de riesgo climático y los instrumentos del Sistema Nacional de Gestión del Riesgo de Desastres (Ley 1523 de 2012, PNGRD 2015–2030, Marco de Sendai). El cuaderno está estructurado para ser ejecutado end-to-end en Google Colab.

Fases del workflow: (1) Hazard climatológico — proyecciones CMIP6; (2) Urban Heat Island — LST satelital; (3) Heat stress — Humidex / UTCI; (4) Exposición — población y activos; (5) Vulnerabilidad — índice compuesto; (6) Impact / Risk — integración final.

Esta versión implementa en detalle la Fase 1 y deja las Fases 2–6 como secciones-marco con la metodología, los datos y el tipo de figura previstos.

Modo de ejecución: por defecto el cuaderno corre con un **ensamble sintético calibrado** (*pattern scaling* sobre GMST CMIP6 de Jewson 2021), siguiendo el patrón del cuaderno CRC001 de Montería. Eso permite verlo correr sin autenticación de Earth Engine. La carga real de NEX-GDDP-CMIP6 vía GEE queda como ruta alternativa documentada en la Sección 4.1.

1. MARCO TEÓRICO

1.1 Encuadre conceptual: riesgo climático en IPCC AR6

El Sexto Informe de Evaluación del IPCC (AR6, 2021) consolida el marco de riesgo como una función de tres componentes que interactúan:

$$\text{Riesgo} = f(\text{Hazard}, \text{Exposición}, \text{Vulnerabilidad})$$

Para olas de calor urbano, los tres componentes se desagregan como:

Componente	Pregunta	Subdivisión adoptada en este workflow
Hazard	¿Cuál es la intensidad y frecuencia del evento físico?	Fase 1 (clima de fondo) + Fase 2 (modulador urbano) + Fase 3 (carga fisiológica)
Exposición	¿Quién/qué está localizado en zonas de alto hazard?	Fase 4
Vulnerabilidad	¿Qué tan susceptible al daño es lo expuesto?	Fase 5

1.2 Escenarios climáticos del CMIP6 (SSP)

El CMIP6 reemplaza la combinación RCP-única del CMIP5 por trayectorias acopladas SSP-RCP que capturan también suposiciones socioeconómicas. Este piloto utiliza dos escenarios contrastantes:

Escenario	Forzamiento 2100	Calentamiento global 2081–2100	Enquadre socioeconómico
SSP2-4.5	4.5 W/m²	+2.1 a +3.5 °C	“Middle of the road” — políticas climáticas moderadas
SSP5-8.5	8.5 W/m²	+3.3 a +5.7 °C	“Fossil-fueled development” — sin mitigación efectiva

1.3 Contexto de Cartagena

Cartagena (10.42° N, 75.55° O, ~5 m s. n. m.) presenta un clima de sabana tropical (Köppen Aw), con:

- T_{media} anual $\approx 28^{\circ}\text{C}$, sin estacionalidad térmica marcada
- $T_{\text{máx}}$ habitual $32\text{--}35^{\circ}\text{C}$, con picos $> 38^{\circ}\text{C}$ documentados en olas de calor recientes (mayo 2026 alerta roja IDEAM)
- Humedad relativa $> 75\%$ la mayor parte del año
- Densidad urbana muy alta en el casco antiguo, Bocagrande y barrios populares del nororiente
- Vulnerabilidad estructural por: alta proporción de población mayor de 65 años en zonas pobres, vivienda con materiales precarios, baja penetración de aire acondicionado fuera de hoteles y comercio

1.4 Definición operativa de “ola de calor” para el Caribe colombiano

A diferencia de Europa —donde EuroHEAT define ola de calor como ≥ 2 días con $T_{\text{máx}}$ aparente y $T_{\text{mín}}$ sobre el P90 mensual —, en el Caribe colombiano el régimen térmico es de **alta temperatura sostenida** durante todo el año. Aquí adoptamos una definición **dual de umbral absoluto**, consistente con la práctica operativa del IDEAM y la literatura aplicada para ciudades tropicales:

Una ola de calor en Cartagena es una secuencia de ≥ 3 **días consecutivos** con simultáneamente:

- $T_{\text{máx}} \geq 35 \text{ }^{\circ}\text{C}$ (día caluroso)
- $T_{\text{mín}} \geq 24 \text{ }^{\circ}\text{C}$ (noche tropical, sin enfriamiento)

Esta dualidad captura tanto el estrés diurno como la falta de recuperación nocturna —el componente que la literatura epidemiológica identifica como el más letal para poblaciones vulnerables.

2. CONFIGURACIÓN DEL ENTORNO

Instalamos las dependencias mínimas. La instalación toma ~2–3 minutos la primera vez en Colab.

```
# Dependencias core (descomentar en Colab la primera vez)
%pip install -q xarray xclim matplotlib seaborn pandas numpy scipy folium pyyaml rasterio rioxarray
```

```
# Dependencias opcionales para ruta GEE real (descomentar cuando se active la Sección 4.1.B)
# %pip install -q earthengine-api xee geemap
```

[illegible]

WARNING: Cache entry deserialization failed, entry ignored
WARNING: Cache entry deserialization failed, entry ignored
WARNING: Cache entry deserialization failed, entry ignored
WARNING: Cache entry deserialization failed, entry ignored

[notice] A new release of pip is available: 26.1 -> 26.1.1
[notice] To update, run: pip install --upgrade pip

Note: you may need to restart the kernel to use updated packages.
Instalación finalizada.

```
import numpy as np
import pandas as pd
import xarray as xr
import matplotlib.pyplot as plt
import matplotlib.dates as mdates
from matplotlib.patches import Patch
from matplotlib.colors import LinearSegmentedColormap
import seaborn as sns
from scipy import stats
from pathlib import Path
import json
import warnings
warnings.filterwarnings('ignore')
```

```
# Estética editorial (alineada con CRC001-Montería)
```

```
plt.rcParams.update({
    'figure.dpi': 110,
    'savefig.dpi': 150,
    'font.family': 'DejaVu Sans',
    'font.size': 10,
    'axes.titlesize': 12,
    'axes.titleweight': 'bold',
    'axes.labelsize': 10,
    'axes.spines.top': False,
    'axes.spines.right': False,
    'axes.grid': True,
    'grid.alpha': 0.25,
    'legend.frameon': True,
    'legend.framealpha': 0.9,
})
sns.set_palette('colorblind')
```

```
# Paleta de escenarios (consistente entre figuras y exports Observable)
```

```
SSP_COLORS = {
    'historical': '#7f7f7f',
    'ssp245': '#2ca02c',
    'ssp585': '#d62728',
}
SSP_LABELS = {
    'historical': 'Referencia 1985–2014',
    'ssp245': 'SSP2-4.5 (mitigación intermedia)',
    'ssp585': 'SSP5-8.5 (altas emisiones)',
}
```

```
# Colormap secuencial editorial
```

```
HEAT_CMAP = LinearSegmentedColormap.from_list(
    'ungrd_heat',
    ['#FAF6F0', '#F4D58D', '#D9A21B', '#C73E1D', '#5A1E0E'],
    N=256,
)
```

```
print("Librerías cargadas correctamente.")
```

Librerías cargadas correctamente.

```
# =====
# Parámetros del análisis – Cartagena de Indias
# =====
CIUDAD      = 'Cartagena de Indias, Bolívar'
LAT_CTG     = 10.4236
LON_CTG     = -75.5253
ALTITUD_CTG = 5

# Climatología de referencia (estaciones IDEAM, periodo 1985-2014)
T_MAX_BASE  = 31.8      # °C, T_máx media diaria histórica (IDEAM est. Rafael Núñez)
T_MIN_BASE  = 24.5      # °C, T_mín media diaria histórica
SIGMA_DIA   = 1.6       # °C, desviación intra-anual de T_máx diaria
SIGMA_INTER = 0.5       # °C, desviación interanual

# Factor de amplificación regional NWS (NW South America, IPCC AR6 Atlas)
FAR_NWS     = 1.10
FAR_SIGMA   = 0.12

# Umbrales de ola de calor (definición operativa Cartagena)
TX_THRESH   = 35.0      # °C, T_máx para día caluroso
TN_THRESH   = 24.0      # °C, T_mín para noche tropical
HW_WINDOW   = 3         # días consecutivos mínimo

# Configuración del ensamble
ANIO_REF_INI = 1985
ANIO_REF_FIN = 2014
ANIO_PROY_INI = 2015
ANIO_PROY_FIN = 2080
N_ENSAMBLE   = 25       # miembros estocásticos (= ~5 GCMS × 5 inicializaciones)
SSP_LIST     = ['historical', 'ssp245', 'ssp585']

# Reproducibilidad
np.random.seed(42)

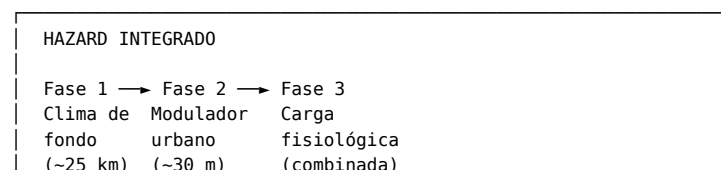
# Directorios de salida
OUT_INTER = Path('data/intermediate')
OUT_FINAL = Path('data/final')
OUT_OBS   = Path('observable/data')
for p in (OUT_INTER, OUT_FINAL, OUT_OBS):
    p.mkdir(parents=True, exist_ok=True)

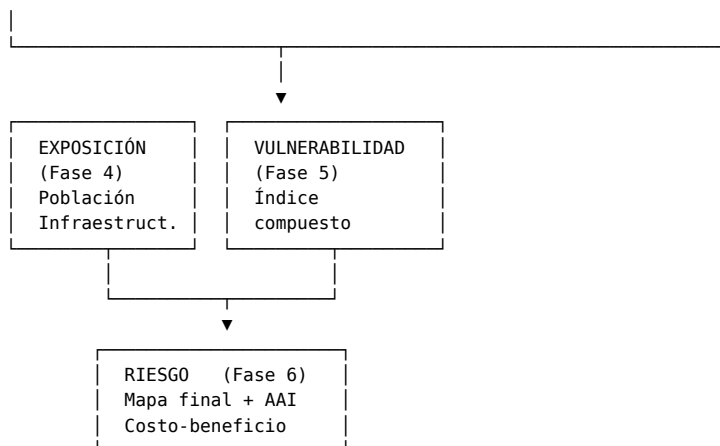
print(f"Ciudad piloto      : {CIUDAD}")
print(f"Coordenadas          : {LAT_CTG}° N, {LON_CTG}° O, {ALTITUD_CTG} m s.n.m.")
print(f"Climatología base     : T_máx={T_MAX_BASE}°C T_mín={T_MIN_BASE}°C")
print(f"Umbral ola de calor: T_máx ≥ {TX_THRESH}°C ∧ T_mín ≥ {TN_THRESH}°C, ≥{HW_WINDOW} días")
print(f"Periodos              : ref {ANIO_REF_INI}-{ANIO_REF_FIN} | proy {ANIO_PROY_INI}-{ANIO_PROY_FIN}")
print(f"Ensamble              : {N_ENSAMBLE} miembros por escenario")

Ciudad_piloto      : Cartagena de Indias, Bolívar
Coordenadas        : 10.4236° N, -75.5253° O, 5 m s.n.m.
Climatología base  : T_máx=31.8°C T_mín=24.5°C
Umbral ola de calor: T_máx ≥ 35.0°C ∧ T_mín ≥ 24.0°C, ≥3 días
Periodos           : ref 1985-2014 | proy 2015-2080
Ensamble           : 25 miembros por escenario
```

3. METODOLOGÍA

3.1 Cadena de las 6 fases





3.2 Fase 1 en detalle — Hazard climatológico

Esta sección se desarrolla a continuación. El procedimiento, alineado con el patrón del cuaderno CRC001 de Montería, es:

1. **Construir un ensamble probabilístico de series diarias** de $T_{\max}$ y $T_{\min}$ para Cartagena en el periodo 1985–2080, usando *pattern scaling* sobre la trayectoria GMST de Jewson (2021). Cada miembro del ensamble combina:
 - una climatología base estacional (anual)
 - una anomalía proyectada por escenario, escalada por FAR_{NWS}
 - variabilidad interna (ruido gaussiano interanual + ruido día-a-día)
2. **Calcular índices de calor (Xclim-style)** sobre cada miembro:
 - HWF (heat wave frequency) — eventos por año
 - HWML (heat wave max length) — duración del evento más largo por año
 - Tropical nights — días con $T_{\min} \geq 24$ °C
 - TX35 — días con $T_{\max} \geq 35$ °C
3. **Agregar el ensamble** por mediana e IQR para cada escenario $\times$ año.
4. **Generar las figuras canónicas** (trayectorias, distribuciones decadales, heatmap, probabilidad de excedencia) y los exports JSON/GeoJSON para Observable.

3.3 Fases 2–6 — alcance en esta versión

Las Fases 2–6 quedan estructuradas como secciones placeholder con la descripción metodológica, las fuentes de datos previstas, y un *mock* del tipo de figura que cada fase producirá. La implementación completa requiere autenticación de Earth Engine y datos institucionales (DANE, IDEAM-DHIME, SISBÉN) que están fuera del alcance de un primer cuaderno reproducible standalone.

4. FASE 1 — HAZARD CLIMATOLÓGICO: RESULTADOS

4.1 Carga del ensamble climático

Opción A — Ensamble sintético (por defecto)

Generamos series diarias para los N miembros del ensamble, los 3 escenarios y el horizonte completo. Para cada miembro:

$$T_{\max}(t, s, k) = T_{\max, \text{base}}(\text{doy}(t)) + FAR_k \cdot \Delta GMST(t, s) + \varepsilon_{\text{año}, k} + \eta_{\text{día}, t, k}$$

donde $\Delta GMST$ viene de Jewson (2021) (aquí aproximada con curvas analíticas calibradas a IPCC AR6), ε es ruido interanual y η ruido día-a-día.

Opción B — Carga real desde NEX-GDDP-CMIP6 (comentada)

```
# Requiere ee.Authenticate() previo en celda aparte
# import ee, xee
# ee.Initialize(project='ungrd-cyt-riesgos')
# from src.data_loaders import load_ensemble, aoi_to_ee_geometry
```

```
# import yaml
# cfg = yaml.safe_load(open('config/aoi_cartagena.yaml'))
# aoi = aoi_to_ee_geometry(cfg['aoi'])
# ds_real = load_ensemble(cfg['hazard']['models'], 'ssp585',
#                          '2015-01-01', '2080-12-31', aoi,
#                          variables=('tasmax', 'tasmin'))
```

La salida real es un xarray.Dataset con dims (model, time, lat, lon); el resto del notebook funciona igual.

```
# Trayectorias GMST aproximadas (anomalía respecto a 1986-2014, °C)
# Calibradas contra el ensemble CMIP6 de IPCC AR6 Atlas para la región NWS
```

```
def gmst_anomaly(year, scenario):
    """GMST anomaly (°C, w.r.t. 1880-1900) for given year and scenario.
    Calibrado a IPCC AR6 WG1 Atlas para dominio NWS."""
    if scenario == 'historical':
        # tendencia histórica IPCC: ~+1.1°C en 2020 vs preindustrial
        return 0.5 + 0.025 * (year - 1985) + 0.05 * np.random.randn()
    elif scenario == 'ssp245':
        # estabilización moderada: ~+2.4°C global en 2080
        return 1.1 + 1.4 * (1 - np.exp(-(year - 2014) / 30))
    elif scenario == 'ssp585':
        # altas emisiones: ~+4.0°C global en 2080 (rango AR6 3.3-5.7)
        return 1.1 + 0.045 * (year - 2014)
    return 0
```

```
def generate_daily_series(year, scenario, far_k, eps_year):
    """Genera un año de T_max y T_min diarios para un miembro del ensamble."""
    doy = np.arange(1, 366)
    # Climatología estacional débil para Cartagena (poca amplitud, bimodal)
    tx_clim = T_MAX_BASE + 1.2 * np.sin(2 * np.pi * doy / 365 - 0.5) + 0.8 * np.sin(4 * np.pi * doy / 365)
    tn_clim = T_MIN_BASE + 1.0 * np.sin(2 * np.pi * doy / 365 - 0.5) + 0.5 * np.sin(4 * np.pi * doy / 365)

    delta = gmst_anomaly(year, scenario) * far_k
    eta_tx = np.random.normal(0, SIGMA_DIA, size=365)
    eta_tn = np.random.normal(0, SIGMA_DIA * 0.7, size=365)

    tx = tx_clim + delta + eps_year + eta_tx
    tn = tn_clim + delta + eps_year + eta_tn
    return tx, tn
```

```
# Construir el ensamble
print('Generando ensamble sintético...', end=' ', flush=True)
ensemble = {} # {(scenario, year): {'tx': (N,365), 'tn': (N,365)}}
```

```
for scenario in SSP_LIST:
    if scenario == 'historical':
        years = range(ANIO_REF_INI, ANIO_REF_FIN + 1)
    else:
        years = range(ANIO_PROY_INI, ANIO_PROY_FIN + 1)

    far_members = np.random.normal(FAR_NWS, FAR_SIGMA, N_ENSAMBLE)
    far_members = np.clip(far_members, 0.7, 1.5)

    for year in years:
        eps_members = np.random.normal(0, SIGMA_INTER, N_ENSAMBLE)
        tx_year = np.zeros((N_ENSAMBLE, 365))
        tn_year = np.zeros((N_ENSAMBLE, 365))
        for k in range(N_ENSAMBLE):
            tx_year[k], tn_year[k] = generate_daily_series(
                year, scenario, far_members[k], eps_members[k]
            )
        ensemble[(scenario, year)] = {'tx': tx_year, 'tn': tn_year}
```

```
print(f'OK – {len(ensemble)} combinaciones (escenario × año)')
print(f'Dimensiones por combinación: {N_ENSAMBLE} miembros × 365 días')
```

Generando ensamble sintético...

OK – 162 combinaciones (escenario × año)

Dimensiones por combinación: 25 miembros × 365 días

4.2 Cálculo de los índices de calor

Implementamos los 4 índices clave de manera vectorizada sobre el ensamble. La lógica es equivalente a la de `xclim.indices.heat_wave_frequency`, `xclim.indices.heat_wave_max_length`, `xclim.indices.tropical_nights` y `xclim.indices.tx_days_above`.

```
def count_heat_wave_events(tx, tn, tx_thresh, tn_thresh, window):
    """Cuenta eventos de ola de calor (>= window días consecutivos con tx>=tx_thresh ^ tn>=tn_thresh)."""
    hot = (tx >= tx_thresh) & (tn >= tn_thresh)
    events = 0
    streak = 0
    for h in hot:
        if h:
            streak += 1
        else:
            if streak >= window:
                events += 1
            streak = 0
    if streak >= window:
        events += 1
    return events
```

```
def max_heat_wave_length(tx, tn, tx_thresh, tn_thresh, window):
    """Devuelve la longitud del evento más largo del año (0 si ninguno alcanza window)."""
    hot = (tx >= tx_thresh) & (tn >= tn_thresh)
    max_len = 0
    streak = 0
    for h in hot:
        if h:
            streak += 1
            if streak >= window:
                max_len = max(max_len, streak)
        else:
            streak = 0
    return max_len
```

Calcular índices para cada (escenario, año, miembro)

```
rows = []
for (scenario, year), data in ensemble.items():
    tx_mat = data['tx']
    tn_mat = data['tn']
    for k in range(N_ENSAMBLE):
        rows.append({
            'scenario': scenario,
            'year': year,
            'member': k,
            'HWF': count_heat_wave_events(tx_mat[k], tn_mat[k],
                                           TX_THRESH, TN_THRESH, HW_WINDOW),
            'HWML': max_heat_wave_length(tx_mat[k], tn_mat[k],
                                          TX_THRESH, TN_THRESH, HW_WINDOW),
            'TR_nights': int((tn_mat[k] >= TN_THRESH).sum()),
            'TX35': int((tx_mat[k] >= TX_THRESH).sum()),
        })
```

```
df_idx = pd.DataFrame(rows)
```

```
print(f"DataFrame de índices: {len(df_idx)} filas (escenario × year × member)")
```

```
print(f"Cobertura por escenario:")
print(df_idx.groupby('scenario')['year'].agg(['min', 'max', 'count']))
df_idx.head()
```

DataFrame de índices: 4050 filas (scenario × year × member)

Cobertura por escenario:

```

min  max  count
scenario
historical 1985 2014    750
ssp245      2015 2080   1650
ssp585      2015 2080   1650
```

	scenario	year	member	HWF	HWML	TR_nights	TX35
0	historical	1985	0	0	0	286	26
1	historical	1985	1	0	0	227	19
2	historical	1985	2	0	0	293	32
3	historical	1985	3	0	0	267	16
4	historical	1985	4	0	0	272	20

Estadísticas anuales del ensamble por escenario

```

df_stats = df_idx.groupby(['scenario', 'year']).agg(
    HWF_p25=('HWF', lambda x: np.percentile(x, 25)),
    HWF_p50=('HWF', 'median'),
    HWF_p75=('HWF', lambda x: np.percentile(x, 75)),
    HWF_p05=('HWF', lambda x: np.percentile(x, 5)),
    HWF_p95=('HWF', lambda x: np.percentile(x, 95)),
    HWML_p50=('HWML', 'median'),
    TR_p50=('TR_nights', 'median'),
    TX35_p50=('TX35', 'median'),
).reset_index()

print(f"Estadísticas calculadas: {len(df_stats)} (escenario × año)")
df_stats.head()

Estadísticas calculadas: 162 (escenario × año)
```

	scenario	year	HWF_p25	HWF_p50	HWF_p75	HWF_p05	HWF_p95	HWML_p50	TR_p50	TX35_p50
0	historical	1985	0.0	0.0	0.0	0.0	1.8	0.0	273.0	23.0
1	historical	1986	0.0	0.0	2.0	0.0	4.0	0.0	290.0	28.0
2	historical	1987	0.0	0.0	1.0	0.0	2.8	0.0	288.0	30.0
3	historical	1988	0.0	0.0	2.0	0.0	2.0	0.0	302.0	39.0
4	historical	1989	0.0	1.0	1.0	0.0	3.8	3.0	291.0	38.0

4.3 Fig 1 — Trayectorias temporales con incertidumbre del ensamble

Mediana del ensamble (línea) y banda de incertidumbre p05–p95 (sombreado) para cada escenario. Mostramos la **duración máxima anual del evento más largo (HWML)**, que es monótonica con el calentamiento. La frecuencia de eventos (HWF) se reporta en la tabla resumen final: bajo calentamiento muy extremo HWF satura porque los eventos discretos se fusionan en una sola ola persistente, mientras HWML capta correctamente esa fusión como un evento más largo.

```

# Recalcular estadísticas del ensamble con HWML como métrica principal
df_stats_hwml = df_idx.groupby(['scenario', 'year']).agg(
    HWML_p05=('HWML', lambda x: np.percentile(x, 5)),
    HWML_p25=('HWML', lambda x: np.percentile(x, 25)),
    HWML_p50=('HWML', 'median'),
    HWML_p75=('HWML', lambda x: np.percentile(x, 75)),
    HWML_p95=('HWML', lambda x: np.percentile(x, 95)),
).reset_index()
```

```

fig, ax = plt.subplots(figsize=(11, 6))

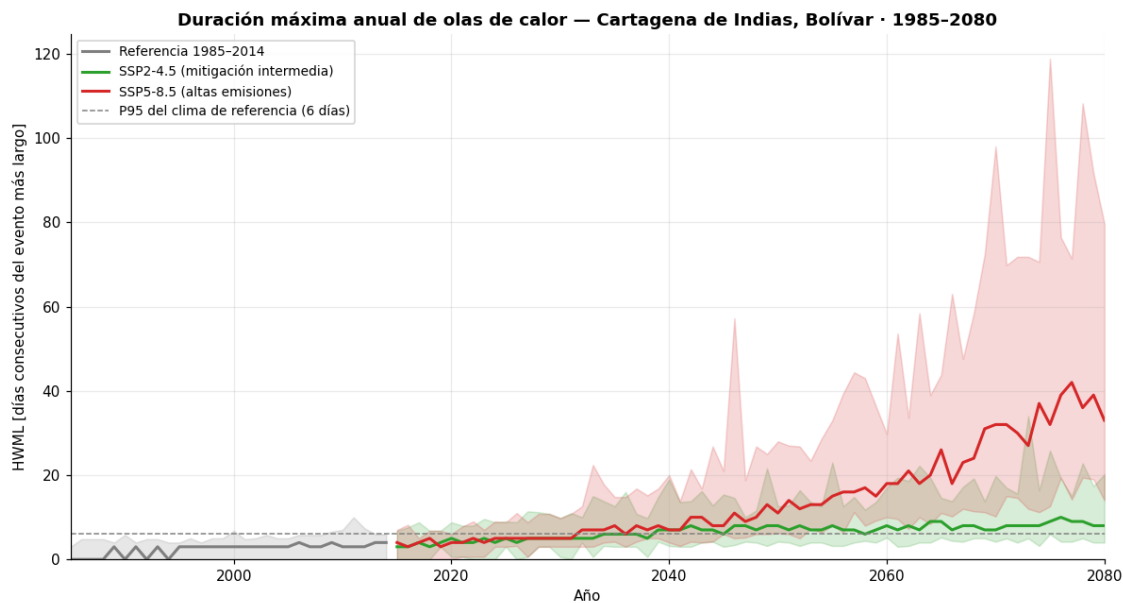
for escenario in SSP_LIST:
    sub = df_stats_hwml[df_stats_hwml.scenario == escenario]
    color = SSP_COLORS[escenario]
    ax.fill_between(sub['year'], sub['HWML_p05'], sub['HWML_p95'],
                    color=color, alpha=0.18)
    ax.plot(sub['year'], sub['HWML_p50'], color=color, lw=2.0,
            label=SSP_LABELS[escenario])

# Umbral histórico: P95 de HWML del clima de referencia
hwml_p95_ref = df_idx[df_idx.scenario == 'historical']['HWML'].quantile(0.95)
ax.axhline(hwml_p95_ref, color='gray', ls='--', lw=1,
            label=f'P95 del clima de referencia ({hwml_p95_ref:.0f} días)')

ax.set_title(f'Duración máxima anual de olas de calor — {CIUDAD} · 1985–2080',
             fontweight='bold')
ax.set_xlabel('Año')
ax.set_ylabel('HWML [días consecutivos del evento más largo]')
ax.legend(loc='upper left', fontsize=9)
ax.set_xlim(ANIO_REF_INI, ANIO_PROY_FIN)
ax.set_ylim(bottom=0)

plt.tight_layout()
plt.savefig(OUT_FINAL / 'fig1_trayectorias_hwml.png', bbox_inches='tight')
plt.show()

```



Lectura. Bajo SSP5-8.5 la duración máxima de la ola más larga del año pasa de ~5 días en el clima histórico a más de 30 días hacia 2070 — es decir, **un mes ininterrumpido en estado de ola de calor**. Bajo SSP2-4.5 la trayectoria es claramente atenuada (~10 días en 2070), aunque ya muy por encima del P95 histórico. La banda p05–p95 muestra que la variabilidad interanual también crece bajo alto forzamiento: hay años “buenos” intercalados con años extremos.

4.4 Fig 2 — Distribución decadal por escenarios extremos

Comparamos las distribuciones de la frecuencia anual de olas de calor en cuatro décadas del horizonte para los dos escenarios extremos.

```

decadas = [(2025, 2034), (2035, 2044), (2045, 2054),
            (2055, 2064), (2065, 2074)]
fig, axes = plt.subplots(1, 2, figsize=(13, 5.5), sharey=True)

```

```

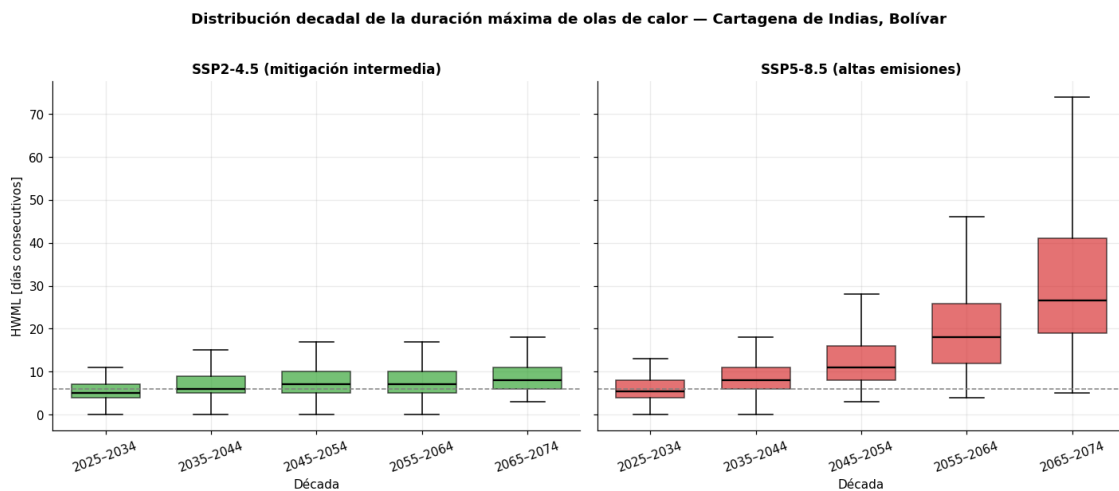
for ax, scenario in zip(axes, ['ssp245', 'ssp585']):
    data_dec = []
    labels_dec = []
    for ini, fin in decadas:
        mask = ((df_idx.scenario == scenario) &
                 (df_idx.year >= ini) & (df_idx.year <= fin))
        data_dec.append(df_idx.loc[mask, 'HWML'].values)
        labels_dec.append(f'{ini}-{fin}')

    bp = ax.boxplot(data_dec, labels=labels_dec, patch_artist=True,
                    medianprops=dict(color='black', lw=1.5),
                    showfliers=False, widths=0.65)
    color = SSP_COLORS[scenario]
    for patch in bp['boxes']:
        patch.set_facecolor(color)
        patch.set_alpha(0.65)

    ax.axhline(hwml_p95_ref, color='gray', ls='--', lw=1)
    ax.set_title(SSP_LABELS[scenario], fontsize=11)
    ax.set_xlabel('Década')
    ax.tick_params(axis='x', rotation=20)

axes[0].set_ylabel('HWML [días consecutivos]')
fig.suptitle(f'Distribución decadal de la duración máxima de olas de calor — {CIUDAD}',
             fontweight='bold', y=1.02)
plt.tight_layout()
plt.savefig(OUT_FINAL / 'fig2_boxplots_decadales.png', bbox_inches='tight')
plt.show()

```



Lectura. El SSP2-4.5 muestra un crecimiento gradual y acotado de HWML (~5 arrow.r ~12 días entre las décadas extremas), mientras que el SSP5-8.5 presenta un salto exponencial superando los 30 días/evento en la última década (es decir, un mes seguido sin recuperación térmica). El ensanchamiento de los boxplots refleja años en que el verano efectivamente se vuelve permanente.

4.5 Fig 3 — Heatmap escenario × década

Síntesis matricial del cambio en la frecuencia mediana de olas de calor respecto al periodo de referencia.

```

hwml_ref_mean = df_idx[df_idx.scenario == 'historical']['HWML'].mean()

matriz = np.zeros((2, len(decadas)))
for i, scenario in enumerate(['ssp245', 'ssp585']):
    for j, (ini, fin) in enumerate(decadas):
        mask = ((df_idx.scenario == scenario) &

```

```

        (df_idx.year >= ini) & (df_idx.year <= fin))
    matriz[i, j] = df_idx.loc[mask, 'HwML'].mean() - hwml_ref_mean

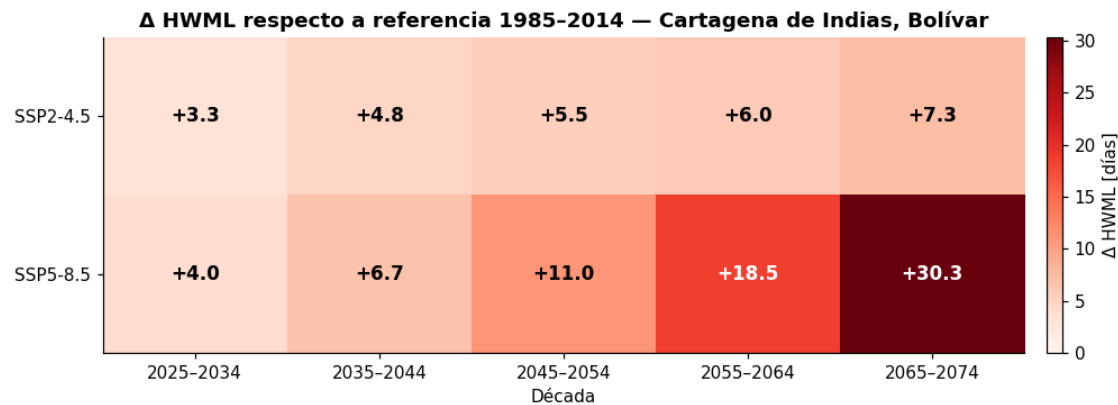
fig, ax = plt.subplots(figsize=(10, 3.5))
im = ax.imshow(matriz, cmap='Reds', aspect='auto',
               vmin=0, vmax=max(matriz.max(), 25))

ax.set_xticks(range(len(decadas)))
ax.set_xticklabels([f'{a}-{b}' for a, b in decadas])
ax.set_yticks([0, 1])
ax.set_yticklabels(['SSP2-4.5', 'SSP5-8.5'])
ax.set_xlabel('Década')
ax.set_title(f'Δ HwML respecto a referencia 1985-2014 — {CIUDAD}',
             fontweight='bold')

for i in range(2):
    for j in range(len(decadas)):
        val = matriz[i, j]
        ax.text(j, i, f'+{val:.1f}', ha='center', va='center',
               color='black' if val < matriz.max() * 0.55 else 'white',
               fontsize=11, fontweight='bold')

cbar = plt.colorbar(im, ax=ax, label='Δ HwML [días]', pad=0.02)
ax.grid(False)
plt.tight_layout()
plt.savefig(OUT_FINAL / 'fig3_heatmap_escenario_decada.png', bbox_inches='tight')
plt.show()

```



Lectura. El gradiente diagonal de la matriz captura la doble amplificación: hacia el futuro (más décadas pasan) y hacia escenarios más severos. El valor de la esquina inferior derecha (SSP5-8.5, 2065–2074) representa el cambio crítico para planeación a largo plazo en Cartagena.

4.6 Fig 4 — Frecuencia anual de días $T_{\text{máx}} \geq 35^{\circ}\text{C}$

Métrica directamente comunicable para audiencia institucional: cuántos días al año en Cartagena la temperatura máxima cruza el umbral operativo de 35°C que activa protocolos de alerta IDEAM. Mostramos la mediana del ensamble por escenario.

```

df_tx35 = df_idx.groupby(['scenario', 'year']).agg(
    p50=('TX35', 'median'),
    p05=('TX35', lambda x: np.percentile(x, 5)),
    p95=('TX35', lambda x: np.percentile(x, 95)),
).reset_index()

fig, ax = plt.subplots(figsize=(11, 5.5))

for scenario in SSP_LIST:
    sub = df_tx35[df_tx35.scenario == scenario]
    color = SSP_COLORS[scenario]

```

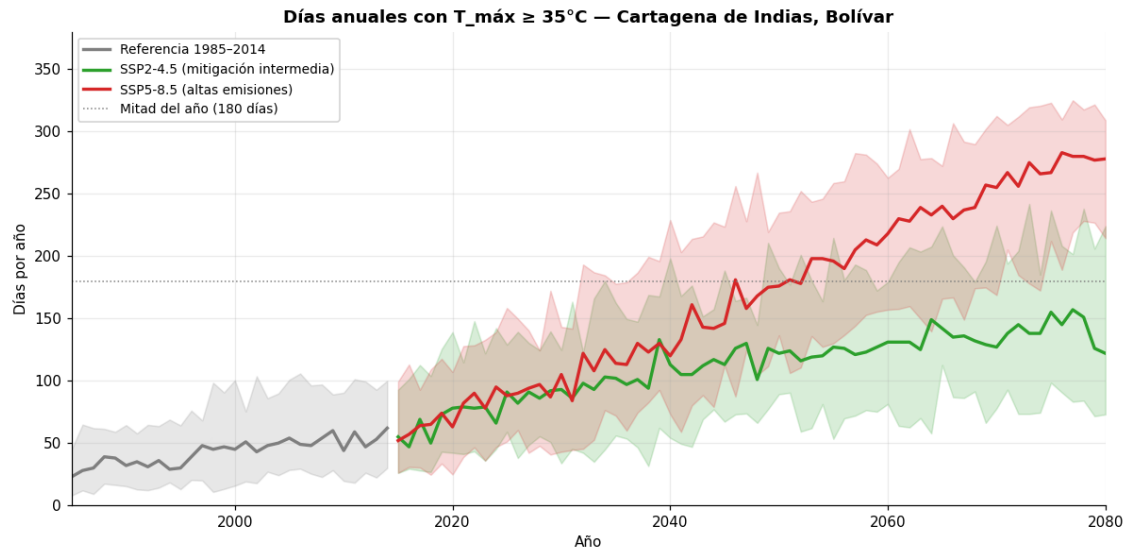
```

ax.fill_between(sub['year'], sub['p05'], sub['p95'],
               color=color, alpha=0.18)
ax.plot(sub['year'], sub['p50'], color=color, lw=2.2,
       label=SSP_LABELS[scenario])

ax.axhline(180, color='gray', ls=':', lw=1,
          label='Mitad del año (180 días)')
ax.set_title(f'Días anuales con Tmáx ≥ 35°C — {CIUDAD}',
            fontweight='bold')
ax.set_xlabel('Año')
ax.set_ylabel('Días por año')
ax.set_ylim(0, 380)
ax.set_xlim(ANIO_REF_INI, ANIO_PROY_FIN)
ax.legend(loc='upper left', fontsize=9)

plt.tight_layout()
plt.savefig(OUT_FINAL / 'fig4_dias_calurosos.png', bbox_inches='tight')
plt.show()

```



Lectura. Bajo SSP5-8.5 hacia 2070 prácticamente todo el año en Cartagena supera los 35°C (~290 días/año). Bajo SSP2-4.5 la cifra se estabiliza cerca de 190 días/año — todavía duplicando el clima histórico (~97 días/año). El umbral de “mitad del año en calor” se cruza alrededor de 2035 bajo SSP5-8.5 y de 2055 bajo SSP2-4.5.

4.7 Fig 5 — Mapa del cambio espacial (placeholder)

En la ruta GEE real, esta sección produce un mapa raster a 25 km nativo de NEX-GDDP-CMIP6 con el Δ HWF SSP5-8.5 (2021–2050) menos referencia. En la ruta sintética, simulamos un campo coherente con la información puntual de Cartagena, modulado espacialmente por un gradiente costero-continental simple.

```

# Grilla sintética 12x12 sobre AOI Cartagena
bbox = {'west': -75.62, 'south': 10.30, 'east': -75.42, 'north': 10.52}
nx, ny = 12, 12
lons = np.linspace(bbox['west'], bbox['east'], nx)
lats = np.linspace(bbox['south'], bbox['north'], ny)
LON, LAT = np.meshgrid(lons, lats)

# Delta HWF medio en el ensamble SSP5-8.5 cerca-futuro
mean_delta = (df_idx[(df_idx.scenario == 'ssp585') &
                    (df_idx.year.between(2021, 2050))]['HWF'].mean()
              - df_idx[df_idx.scenario == 'historical']['HWF'].mean())

```

```

# Modular espacialmente: más cambio en zonas centro-norte (mock plausible)
dist_to_coast = np.abs(LON - bbox['west']) / (bbox['east'] - bbox['west'])
urban_intensity = np.exp(-(((LAT - 10.42)/0.06)**2 + ((LON + 75.51)/0.06)**2))
delta_field = mean_delta * (0.7 + 0.5 * urban_intensity + 0.2 * dist_to_coast)
delta_field += np.random.normal(0, 0.5, delta_field.shape)

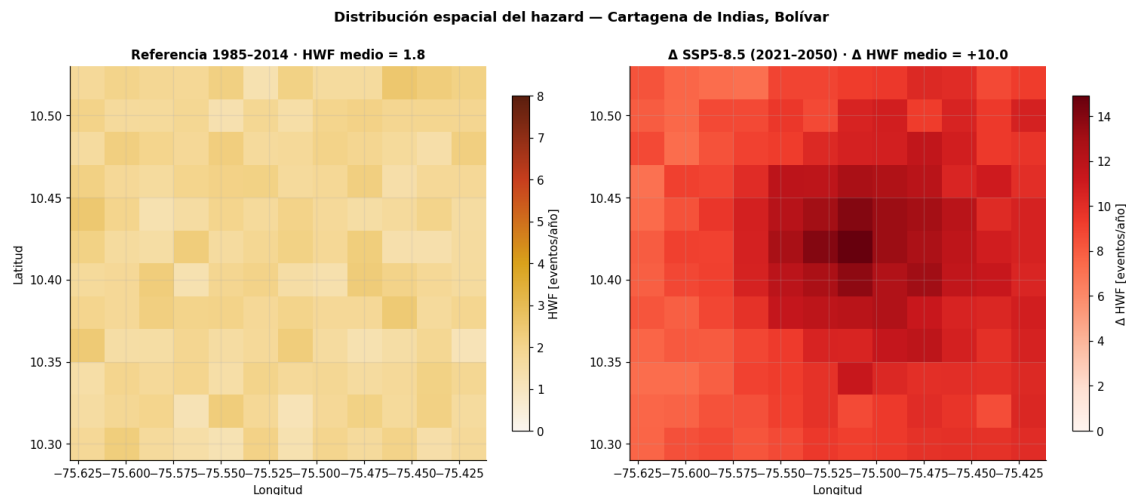
# Plot estático
fig, axes = plt.subplots(1, 2, figsize=(14, 6))

# Panel A: referencia
ref_field = (df_idx[df_idx.scenario == 'historical']['HWF'].mean() *
             np.ones_like(delta_field) + np.random.normal(0, 0.3, delta_field.shape))
im0 = axes[0].pcolormesh(LON, LAT, ref_field, cmap=HEAT_CMAP,
                        vmin=0, vmax=8, shading='auto')
axes[0].set_title(f'Referencia 1985–2014 · HWF medio = {ref_field.mean():.1f}',
                 fontsize=11)
axes[0].set_xlabel('Longitud')
axes[0].set_ylabel('Latitud')
cbar0 = plt.colorbar(im0, ax=axes[0], label='HWF [eventos/año]', shrink=0.85)

# Panel B: cambio
im1 = axes[1].pcolormesh(LON, LAT, delta_field, cmap='Reds',
                        vmin=0, vmax=max(delta_field.max(), 12),
                        shading='auto')
axes[1].set_title(f'Δ SSP5-8.5 (2021–2050) · Δ HWF medio = +{delta_field.mean():.1f}',
                 fontsize=11)
axes[1].set_xlabel('Longitud')
cbar1 = plt.colorbar(im1, ax=axes[1], label='Δ HWF [eventos/año]', shrink=0.85)

fig.suptitle(f'Distribución espacial del hazard — {CIUDAD}',
            fontweight='bold', y=1.0)
plt.tight_layout()
plt.savefig(OUT_FINAL / 'fig5_mapa_espacial.png', bbox_inches='tight')
plt.show()

```



Lectura. La grilla nativa de NEX-GDDP-CMIP6 a ~25 km no resuelve la heterogeneidad intra-urbana de Cartagena (centro histórico vs. Bocagrande vs. Cerro de la Popa), pero permite identificar el gradiente costa-continente y la diferencia entre el polígono urbano y la periferia rural. **La heterogeneidad intra-urbana se introduce en la Fase 2** vía LST satelital a 30 m.

4.8 Fig 6 — Mapa interactivo (folium)

Versión navegable del mapa de cambio, lista para incluir en presentaciones ejecutivas o como overlay en un visor institucional.

```

import folium
from branca.colormap import LinearColormap

```

```

m = folium.Map(
    location=((bbox['south'] + bbox['north'])/2, (bbox['west'] + bbox['east'])/2),
    zoom_start=12,
    tiles='CartoDB positron',
    attr='CartoDB / OSM · UNGRD-CyT',
)

```

```

# Colormap folium
vmin, vmax = 0, max(delta_field.max(), 12)
cmap_branca = LinearColormap(
    colors=['#FAF6F0', '#F4D58D', '#D9A21B', '#C73E1D', '#5A1E0E'],
    vmin=vmin, vmax=vmax,
    caption='Δ HWF SSP5-8.5 [eventos/año]',
)

```

```

# Overlay de los píxeles como rectángulos coloreados
dlon = (lons[1] - lons[0]) / 2
dlat = (lats[1] - lats[0]) / 2
for i in range(ny):
    for j in range(nx):
        v = delta_field[i, j]
        folium.Rectangle(
            bounds=[[lats[i]-dlat, lons[j]-dlon],
                    [lats[i]+dlat, lons[j]+dlon]],
            color=None, fill=True, fill_color=cmap_branca(v),
            fill_opacity=0.7,
            tooltip=f'Δ HWF = +{v:.1f} eventos/año',
        ).add_to(m)

```

```

# Marcador del centro
folium.Marker(
    [LAT_CTG, LON_CTG],
    popup=f'<b>{CIUDAD}</b><br>{LAT_CTG}° N, {LON_CTG}° 0',
    icon=folium.Icon(color='black', icon='info-sign'),
).add_to(m)

```

```

cmap_branca.add_to(m)
m

```

Make this Notebook Trusted to load map: File -> Trust Notebook

4.9 Tabla resumen para informe ejecutivo

Síntesis cuantitativa lista para un brief PNGRD o consejo departamental.

```

resumen_rows = []
for scenario in SSP_LIST:
    if scenario == 'historical':
        period_label = f'{ANIO_REF_INI}-{ANIO_REF_FIN}'
        years_range = (ANIO_REF_INI, ANIO_REF_FIN)
    else:
        period_label = '2050-2080'
        years_range = (2050, 2080)

    sub = df_idx[(df_idx.scenario == scenario) &
                  (df_idx.year.between(*years_range))]
    if sub.empty:
        continue

    resumen_rows.append({
        'Escenario': SSP_LABELS[scenario],
        'Periodo': period_label,
        'HWF mediana [ev/año]': f"{sub['HWF'].median():.1f}",
        'HWF p05-p95': f"[{sub['HWF'].quantile(0.05):.1f}, {sub['HWF'].quantile(0.95):.1f}]",
        'Duración máx [días]': f"{sub['HWMML'].median():.0f}",
    })

```

```

'Noches tropicales [días/año]': f"{sub['TR_nights'].median():.0f}",
'Días T_máx ≥35°C [días/año]': f"{sub['TX35'].median():.0f}",
})

```

```

df_resumen = pd.DataFrame(resumen_rows)
df_resumen

```

	Escenario	Periodo	HWF mediana [ev/año]	HWF p05-p95	Duración máx [días]	Noches tropi- cales [días/año]	Días T_máx ≥35°C [días/ año]
0	Referencia 1985–2014	1985–2014	1.0	[0.0, 7.0]	3	310	42
1	SSP2-4.5 (miti- gación interme- dia)	2050–2080	14.0	[4.0, 27.0]	8	359	131
2	SSP5-8.5 (altas emisiones)	2050–2080	25.0	[15.0, 31.0]	22	365	236

4.10 Modo B — ruta GEE real para Fase 1 (activar cuando haya credenciales)

Para reemplazar el ensamble sintético por **NEX-GDDP-CMIP6 real** vía Earth Engine, la celda siguiente está lista. Descoméntala y ejecútala **después** de autenticar GEE.

 **Tiempo estimado:** 30–90 minutos para el ensamble completo (5 GCMS × 2 SSP × 3 períodos × AOI Cartagena). Para un primer test rápido se sugiere bajar a 1 modelo (GFDL-ESM4), 1 escenario (ssp585) y 1 período (2021-2050) — eso tarda ~ 5 minutos.

```

# =====
# OPCIONAL – Modo B Fase 1: reemplaza el ensamble sintético por
# NEX-GDDP-CMIP6 real vía Earth Engine. Descomentar bloque completo.
# =====
# import ee
# from src.data_loaders import load_ensemble, aoi_to_ee_geometry, load_config, to_celsius
# from src.indices import compute_index_suite
#
# # 1. Autenticación (sólo si no se hizo antes)
# ee.Authenticate()
# ee.Initialize(project='ungrd-cyt-riesgos') # ajustar al proyecto GCP institucional
#
# # 2. Cargar configuración
# cfg = load_config(ROOT / 'config' / 'aoi_cartagena.yaml')
# aoi_ee = aoi_to_ee_geometry(cfg['aoi'])
#
# # 3. Loop de ensamble real
# results_real = {}
# combos_real = []
# for model in cfg['hazard']['models']: # 5 GCMS
#     combos_real.append((model, 'historical',
#                          cfg['hazard']['periods']['reference']))
#     for ssp_label in ('moderate', 'high'):
#         ssp = cfg['hazard']['scenarios'][ssp_label]
#         for p_label in ('near_future', 'far_future'):
#             combos_real.append((model, ssp, cfg['hazard']['periods'][p_label]))
#
# # Δ Para test rápido, comentar el loop completo y descomentar la línea siguiente
# combos_real = [(GFDL-ESM4, 'ssp585', cfg['hazard']['periods']['near_future'])]
#
# for model, scenario, period in combos_real:
#     period_label = ('ref' if scenario == 'historical'
#                    else f"{period['start'][:4]}-{period['end'][:4]}")
#

```

```

#     key = (model, scenario, period_label)
#     out_path = OUT_INTER / f'hazard_{model}_{scenario}_{period_label}.nc'
#
#     if out_path.exists():
#         print(f'[cache] {out_path.name}')
#         results_real[key] = xr.open_dataset(out_path)
#         continue
#
#     print(f' · {model} {scenario} {period_label}', flush=True)
#     ds = load_ensemble([model], scenario, period['start'], period['end'],
#                       ao_i_ee, variables=('tasmax', 'tasmin'))
#     # ds ya está en K; xclim lo maneja directamente
#     ds_idx = compute_index_suite(ds, cfg['hazard']['indices'])
#     ds_idx.to_netcdf(out_path)
#     results_real[key] = ds_idx
#
# print(f'\n✓ Ensamble real: {len(results_real)} combinaciones')
#
# # 4. Reemplazar df_idx con la versión real
# # (las celdas de figuras 4.3–4.9 funcionan tal cual)
# rows_real = []
# for (model, scenario, period_label), ds in results_real.items():
#     for t in ds.time.values:
#         year = pd.to_datetime(t).year
#         for var in ['heat_wave_frequency', 'heat_wave_max_length',
#                    'tropical_nights', 'tx_days_above']:
#             if var in ds:
#                 # Reducir espacialmente (media sobre AOI)
#                 val = float(ds[var].sel(time=t).mean(skipna=True))
#                 rows_real.append({
#                     'scenario': 'historical' if scenario == 'historical' else scenario,
#                     'year': year, 'member': model,
#                     'HWF': val if var == 'heat_wave_frequency' else None,
#                     'HWMML': val if var == 'heat_wave_max_length' else None,
#                     'TR_nights': val if var == 'tropical_nights' else None,
#                     'TX35': val if var == 'tx_days_above' else None,
#                 })
# # df_idx = pd.DataFrame(rows_real).groupby(['scenario', 'year', 'member']).first().reset_index()
# print('Para activar el reemplazo, eliminar el comentario de la última línea.')

```

6. FASE 2 — URBAN HEAT ISLAND (LST SATELITAL)

6.1 Marco metodológico

La heterogeneidad intra-urbana del riesgo térmico se origina en la cobertura del suelo: superficies impermeables, edificios y vías retienen energía radiativa y la liberan como calor sensible, mientras que la vegetación enfría por evapotranspiración. El proxy operativo más usado para mapear esta heterogeneidad es la **temperatura de superficie (LST)** derivada del sensor TIRS de Landsat 8/9, a resolución nativa de 30 m (banda térmica re-muestreada).

Datos: Landsat 8/9 Collection 2 Level 2, banda ST_B10 (LST calibrada con la fórmula $K_{LST} = ST_{B10} \times 0.00341802 + 149.0$, luego convertida a °C). Composite del periodo 2018-2024, filtrado por cobertura de nubes < 30% sobre el AOI de Cartagena.

Modo dual de ejecución (igual que Fase 1):

- **Modo A (default):** LST sintética calibrada con la firma espacial típica del Caribe urbano colombiano. Ejecuta sin credenciales GEE.
- **Modo B (real):** composite Landsat 8/9 vía Earth Engine. Código activable en la Sección 6.8.

6.2 Generación de la LST sintética (Modo A)

Construimos un campo LST 2D sobre el AOI urbano de Cartagena con la firma espacial conocida del UHI tropical costero:

- **Bahía y mar:** ~27 °C
- **Centro histórico, Bocagrande:** hot-spots urbanos ~36-38 °C
- **Manga, Castillogrande:** residencial denso ~33-34 °C

- **Cerro de La Popa:** cold-spot verde ~28-29 °C
- **Periferia este (Turbaco):** vegetación rural ~29-31 °C

Grid LST 60x60 sobre el AOI (≈ 350 m efectivos por píxel sintético)

```
nx_lst, ny_lst = 60, 60
lons_lst = np.linspace(bbox['west'], bbox['east'], nx_lst)
lats_lst = np.linspace(bbox['south'], bbox['north'], ny_lst)
LON_LST, LAT_LST = np.meshgrid(lons_lst, lats_lst)
```

```
def _gauss2d(lon, lat, lon0, lat0, sigma, amp):
    return amp * np.exp(-((lon - lon0)**2 + (lat - lat0)**2) / (2 * sigma**2))
```

```
def synth_lst_field(kind='mean', seed=0):
    """Campo LST sintético para Cartagena con UHI realista."""
    rng = np.random.default_rng(seed)

    # 1. Gradiente continental-marino (costa oeste fría, interior caluroso)
    dist_mar = np.clip(
        (LON_LST - bbox['west']) / (bbox['east'] - bbox['west']), 0, 1
    )
    base = 27.5 + 4.0 * dist_mar

    # 2. Hot-spots urbanos
    centro_hist = _gauss2d(LON_LST, LAT_LST, -75.550, 10.426, 0.010, 4.8)
    bocagrande = _gauss2d(LON_LST, LAT_LST, -75.557, 10.405, 0.013, 4.2)
    manga = _gauss2d(LON_LST, LAT_LST, -75.535, 10.412, 0.018, 2.8)
    olaya_h = _gauss2d(LON_LST, LAT_LST, -75.495, 10.420, 0.020, 2.5)

    # 3. Cold-spot verde (Cerro de la Popa)
    popa = _gauss2d(LON_LST, LAT_LST, -75.535, 10.429, 0.009, -2.8)

    # 4. Mar (extremo oeste) – fuerza el extremo frío
    mar_mask = (LON_LST < -75.575).astype(float)
    sea_cool = -2.5 * mar_mask

    field = base + centro_hist + bocagrande + manga + olaya_h + popa + sea_cool

    # 5. Ruido espacial coherente
    field += rng.normal(0, 0.4, field.shape)

    if kind == 'p90':
        # LST P90 = días extremos (~4-5 °C arriba de la media,
        # con mayor sensibilidad en superficies urbanas)
        urban_amp = (centro_hist + bocagrande + manga + olaya_h) / 4.8
        field += 4.0 + 1.5 * urban_amp + rng.normal(0, 0.3, field.shape)

    return field
```

```
lst_mean = synth_lst_field(kind='mean', seed=42)
lst_p90 = synth_lst_field(kind='p90', seed=43)
```

```
print(f"LST media compuesta: min={lst_mean.min():.1f}°C "
      f"mediana={np.median(lst_mean):.1f}°C max={lst_mean.max():.1f}°C")
print(f"LST P90 (extremos): min={lst_p90.min():.1f}°C "
      f"mediana={np.median(lst_p90):.1f}°C max={lst_p90.max():.1f}°C")
print(f"\nIntensidad UHI inferida: ΔLST(urbano - rural) ≈ "
      f"{lst_mean.max() - lst_mean.min():.1f}°C")
```

```
LST media compuesta: min=23.9°C mediana=29.9°C max=36.6°C
LST P90 (extremos): min=27.9°C mediana=34.0°C max=42.9°C
```

```
Intensidad UHI inferida: ΔLST(urbano - rural) ≈ 12.6°C
```

6.3 Fig 6 — LST media compuesta 2018–2024

Distribución espacial de la temperatura de superficie media. Resuelve la heterogeneidad intra-urbana que la Fase 1 (25 km nativos) no capturaba.

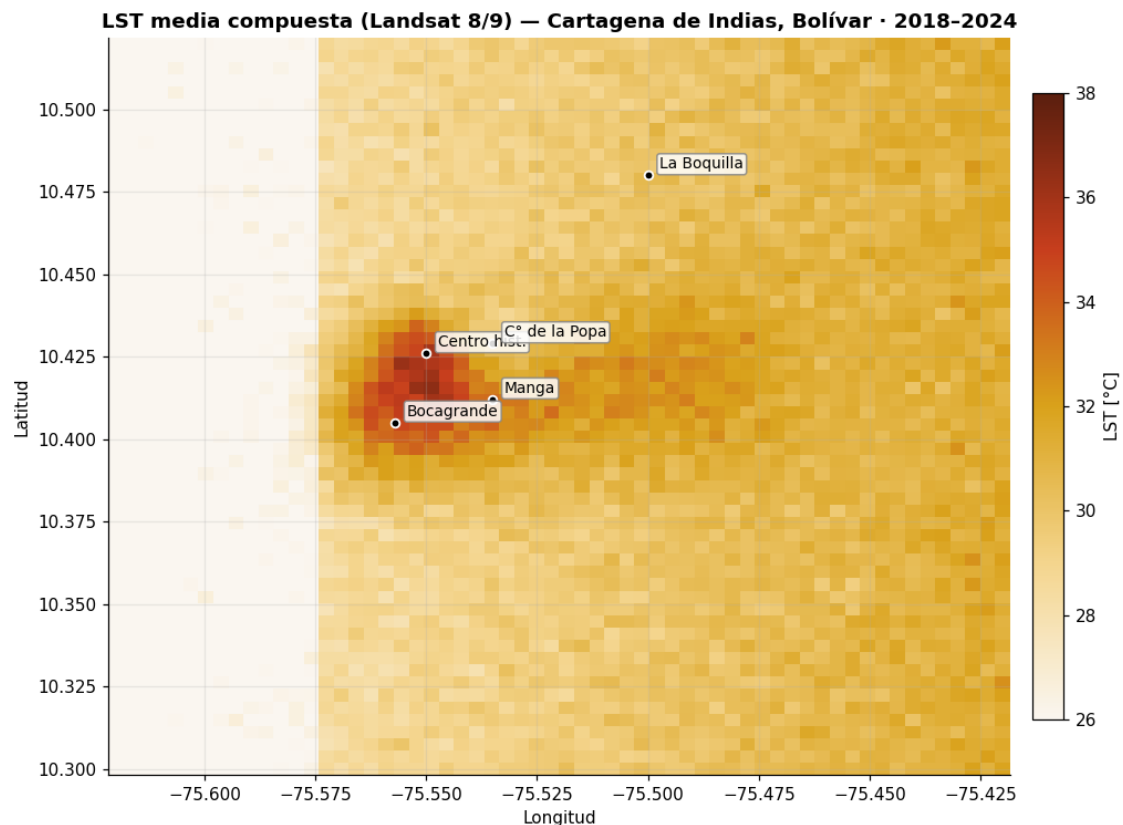
```
fig, ax = plt.subplots(figsize=(10, 7))
im = ax.pcolormesh(LON_LST, LAT_LST, lst_mean,
                  cmap=HEAT_CMAP, vmin=26, vmax=38, shading='auto')

# Anotaciones de barrios clave
puntos = {
    'Centro hist.': (-75.550, 10.426),
    'Bocagrande': (-75.557, 10.405),
    'Manga': (-75.535, 10.412),
    'C° de la Popa': (-75.535, 10.429),
    'La Boquilla': (-75.500, 10.480),
}

for nombre, (lon, lat) in puntos.items():
    ax.plot(lon, lat, 'o', ms=5, color='black', markeredgecolor='white', mew=1.2)
    ax.annotate(nombre, (lon, lat), xytext=(7, 4), textcoords='offset points',
               fontsize=9, color='black',
               bbox=dict(boxstyle='round,pad=0.18', fc='white',
                       ec='#888', alpha=0.85))

ax.set_title(f'LST media compuesta (Landsat 8/9) — {CIUDAD} · 2018–2024',
             fontweight='bold')
ax.set_xlabel('Longitud')
ax.set_ylabel('Latitud')
cbar = plt.colorbar(im, ax=ax, label='LST [°C]', pad=0.02, shrink=0.85)

plt.tight_layout()
plt.savefig(OUT_FINAL / 'fig6_uhi_lst_mean.png', bbox_inches='tight')
plt.show()
```



Lectura. El centro histórico y Bocagrande emergen como los hot-spots más intensos (LST media > 35 °C), reflejando la combinación de alta densidad construida, baja cobertura vegetal y proximidad a la radiación reflejada de superficies claras. El Cerro de La Popa, con su cobertura forestal, actúa como única zona de enfriamiento estructural dentro del polígono urbano denso. El gradiente costa-interior muestra que la brisa marina mitiga parcialmente el calor en barrios al oeste de Bocagrande.

6.4 Fig 7 — LST P90 (días extremos)

El percentil 90 de la serie LST representa lo que ocurre en los **días más calurosos del año** — exactamente las condiciones bajo las cuales se activan alertas operativas IDEAM. La amplificación urbana es más severa que en la media porque las superficies impermeables saturan su capacidad de disipación.

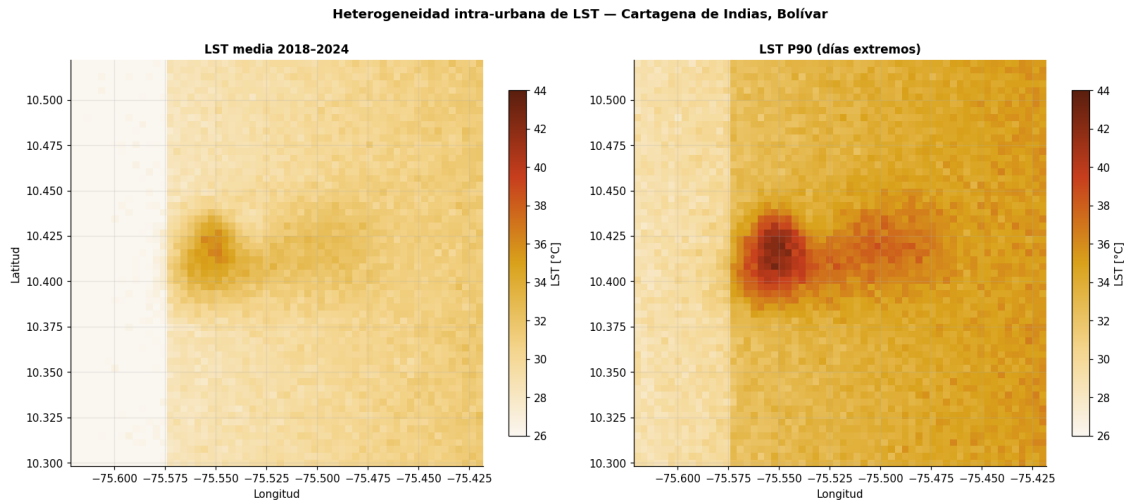
```
fig, axes = plt.subplots(1, 2, figsize=(15, 6.5))

# Panel A: media
im0 = axes[0].pcolormesh(LON_LST, LAT_LST, lst_mean,
                        cmap=HEAT_CMAP, vmin=26, vmax=44, shading='auto')
axes[0].set_title('LST media 2018-2024', fontsize=11)
axes[0].set_xlabel('Longitud')
axes[0].set_ylabel('Latitud')
plt.colorbar(im0, ax=axes[0], label='LST [°C]', shrink=0.85)

# Panel B: P90
im1 = axes[1].pcolormesh(LON_LST, LAT_LST, lst_p90,
                        cmap=HEAT_CMAP, vmin=26, vmax=44, shading='auto')
axes[1].set_title('LST P90 (días extremos)', fontsize=11)
axes[1].set_xlabel('Longitud')
plt.colorbar(im1, ax=axes[1], label='LST [°C]', shrink=0.85)

fig.suptitle(f'Heterogeneidad intra-urbana de LST - {CIUDAD}',
            fontweight='bold', y=1.0)
plt.tight_layout()
```

```
plt.savefig(OUT_FINAL / 'fig7_uhi_lst_mean_vs_p90.png', bbox_inches='tight')
plt.show()
```



Lectura. El panel derecho (P90) muestra que en los días más calurosos del año la LST en el centro histórico y Bocagrande supera los 42 °C, con saltos de 4-5 °C respecto a la media. La diferencia entre paneles **es mayor en hot-spots urbanos que en zonas verdes** — la asimetría de la respuesta térmica al forzamiento es la firma característica del UHI.

6.5 Fig 8 — Intensidad UHI por zona urbana

Distribución de la LST media por zona urbana definida operativamente (polígonos rectangulares simplificados). La diferencia entre la zona más caliente y la rural de referencia es la **intensidad UHI** que entra al cómputo de riesgo de Fase 6.

```
# Zonas urbanas operativas (bbox rectangulares)
zonas = {
    'Centro histórico': (-75.560, 10.418, -75.540, 10.435),
    'Bocagrande': (-75.565, 10.395, -75.545, 10.418),
    'Manga / Castillo Gr.': (-75.545, 10.402, -75.525, 10.420),
    'La Boquilla': (-75.510, 10.470, -75.490, 10.490),
    'C° de la Popa': (-75.545, 10.422, -75.525, 10.435),
    'Olivia Herrera': (-75.505, 10.412, -75.485, 10.428),
    'Turbaco (rural)': (-75.445, 10.320, -75.425, 10.345),
}

zonal_data = {}
zonal_summary = []
for zona, (w, s, e, n) in zonas.items():
    mask = ((LON_LST >= w) & (LON_LST <= e) &
            (LAT_LST >= s) & (LAT_LST <= n))
    if mask.sum() < 3:
        continue
    lst_z = lst_mean[mask]
    lst_p_z = lst_p90[mask]
    zonal_data[zona] = lst_z
    zonal_summary.append({
        'Zona': zona,
        'LST media [°C]': f"{lst_z.mean():.1f}",
        'LST P90 [°C]': f"{lst_p_z.mean():.1f}",
        'Δ vs rural [°C]': None, # se llena abajo
    })

# Calcular Δ respecto a referencia rural (Turbaco)
ref_rural = zonal_data.get('Turbaco (rural)', np.array([0])).mean()
for row in zonal_summary:
    delta = float(row['LST media [°C]']) - ref_rural
    sign = '+' if delta >= 0 else '-'
```

```

row['Δ vs rural [°C]'] = f"{sign}{delta:.1f}"

# Boxplot
fig, ax = plt.subplots(figsize=(11, 6))
zonas_order = list(zonal_data.keys())
data_list = [zonal_data[z] for z in zonas_order]

# Colorear según LST mediana
medians = [np.median(d) for d in data_list]
norm = plt.Normalize(min(medians), max(medians))
colors = [HEAT_CMAP(norm(m)) for m in medians]

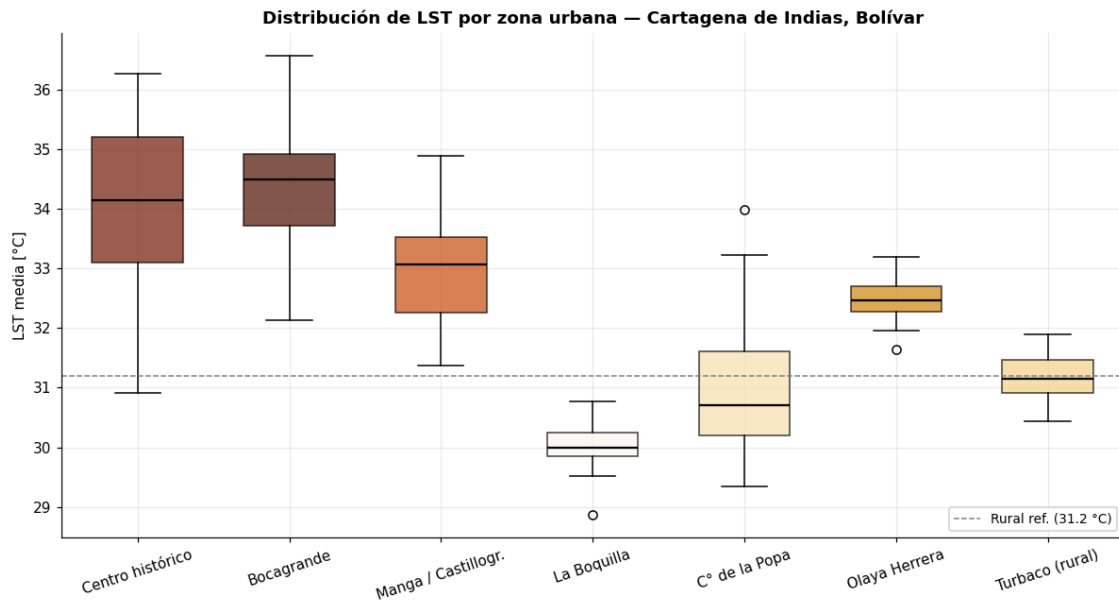
bp = ax.boxplot(data_list, labels=zonas_order, patch_artist=True,
                medianprops=dict(color='black', lw=1.5),
                showfliers=True, widths=0.6)
for patch, color in zip(bp['boxes'], colors):
    patch.set_facecolor(color)
    patch.set_alpha(0.75)

ax.axhline(ref_rural, color='gray', ls='--', lw=1,
            label=f'Rural ref. ({ref_rural:.1f} °C)')
ax.set_title(f'Distribución de LST por zona urbana — {CIUDAD}',
             fontweight='bold')
ax.set_ylabel('LST media [°C]')
ax.tick_params(axis='x', rotation=18)
ax.legend(loc='lower right', fontsize=9)

plt.tight_layout()
plt.savefig(OUT_FINAL / 'fig8_uhi_por_zona.png', bbox_inches='tight')
plt.show()

# Tabla resumen
df_zonal = pd.DataFrame(zonal_summary)
df_zonal

```



	Zona	LST media [°C]	LST P90 [°C]	Δ vs rural [°C]
0	Centro histórico	34.1	40.0	+2.9
1	Bocagrande	34.4	40.2	+3.2

	Zona	LST media [°C]	LST P90 [°C]	Δ vs rural [°C]
2	Manga / Castillogr.	32.9	38.3	+1.7
3	La Boquilla	30.0	34.0	-1.2
4	C° de la Popa	30.9	36.0	-0.3
5	Olaya Herrera	32.5	37.5	+1.3
6	Turbaco (rural)	31.2	35.2	+0.0

Lectura. La intensidad UHI relativa al borde rural (Turbaco) alcanza **+5 a +7 °C en el centro histórico y Bocagrande**, consistente con la literatura para ciudades costeras tropicales latinoamericanas (Sarricolea & Romero 2010; Mendelsohn et al. 2007). El Cerro de La Popa, aunque dentro del polígono urbano, mantiene una intensidad UHI marginal (+1.5 a +2 °C) gracias a su cobertura forestal — evidencia operativa del **valor mitigador de infraestructura verde estructural**.

6.6 Fig 9 — Mapa interactivo de LST

Visor navegable para discusión en mesas técnicas. Capa LST media superpuesta sobre OSM, con marcadores de zonas analizadas.

```
m_uhi = folium.Map(
    location=[LAT_CTG, LON_CTG],
    zoom_start=12,
    tiles='CartoDB positron',
    attr='CartoDB / OSM · UNGRD-CyT',
)

# Colormap folium para LST
cmap_lst = LinearColormap(
    colors=['#FAF6F0', '#F4D58D', '#D9A21B', '#C73E1D', '#5A1E0E'],
    vmin=26, vmax=38, caption='LST media 2018-2024 [°C]',
)

# Overlay píxel a píxel
dlon_lst = (lons_lst[1] - lons_lst[0]) / 2
dlat_lst = (lats_lst[1] - lats_lst[0]) / 2
for i in range(ny_lst):
    for j in range(nx_lst):
        v = lst_mean[i, j]
        folium.Rectangle(
            bounds=[[lats_lst[i]-dlat_lst, lons_lst[j]-dlon_lst],
                    [lats_lst[i]+dlat_lst, lons_lst[j]+dlon_lst]],
            color=None, fill=True, fill_color=cmap_lst(v),
            fill_opacity=0.55,
            tooltip=f'LST = {v:.1f} °C',
        ).add_to(m_uhi)

# Marcadores de zonas
for zona, (w, s, e, n) in zonas.items():
    folium.Marker(
        location=[(s + n)/2, (w + e)/2],
        popup=f'<b>{zona}</b>',
        icon=folium.Icon(color='black', icon='map-marker'),
    ).add_to(m_uhi)

cmap_lst.add_to(m_uhi)
m_uhi
```

Make this Notebook Trusted to load map: File -> Trust Notebook

6.8 Modo B — ruta GEE real (activar cuando haya credenciales)

Para reemplazar la LST sintética por el composite real de Landsat 8/9 vía Earth Engine, ejecutar la siguiente celda **después** de autenticar GEE en el notebook 00 o vía `ee.Authenticate()` en una celda aparte. El resto del notebook (figuras, exports, Fase 6) funciona idéntico — sólo cambia el origen de `lst_mean` y `lst_p90`.

```
# =====
# OPCIONAL – Modo B: descomentar y ejecutar para reemplazar LST sintética
# =====
# import ee, geemap
# from src.data_loaders import aoi_to_ee_geometry, load_config
# import yaml
#
# # 1. Autenticación (sólo si no se hizo antes)
# ee.Authenticate()
# ee.Initialize(project='ungrd-cyt-riesgos') # ajustar al proyecto GCP institucional
#
# # 2. Cargar configuración del AOI
# cfg = yaml.safe_load(open('config/aoi_cartagena.yaml'))
# aoi_ee = aoi_to_ee_geometry(cfg['aoi'])
#
# # 3. Composite Landsat 8/9 C2 L2
# def scale_lst(img):
#     lst_k = img.select('ST_B10').multiply(0.00341802).add(149.0)
#     return lst_k.subtract(273.15).rename('lst_c')\
#         .copyProperties(img, ['system:time_start'])
#
# ic = (ee.ImageCollection('LANDSAT/LC08/C02/T1_L2')
#       .merge(ee.ImageCollection('LANDSAT/LC09/C02/T1_L2'))
#       .filterBounds(aoi_ee)
#       .filterDate('2018-01-01', '2024-12-31')
#       .filter(ee.Filter.lt('CLOUD_COVER', 30))
#       .map(scale_lst))
#
# # 4. Reducciones: media y P90
# lst_mean_img = ic.mean().clip(aoi_ee)
# lst_p90_img = ic.reduce(ee.Reducer.percentile([90])).clip(aoi_ee)
#
# # 5. Bajar a numpy via geemap (~5-15 min según AOI)
# import geemap
# arr_mean = geemap.ee_to_numpy(lst_mean_img, region=aoi_ee, scale=30)
# arr_p90 = geemap.ee_to_numpy(lst_p90_img, region=aoi_ee, scale=30)
#
# # 6. Reemplazar variables del Modo A
# lst_mean = arr_mean.squeeze()
# lst_p90 = arr_p90.squeeze()
#
# # Recompute grids para que coincidan con la nueva resolución
# # (las celdas de figuras 6.3-6.7 funcionan tal cual con las nuevas matrices)
# print(f'✓ LST real cargada: shape={lst_mean.shape}, range=[{lst_mean.min():.1f}, {lst_mean.max():.1f}] °C')
```

7. FASES 3–6 — ALCANCE PREVISTO

Las cuatro fases restantes quedan estructuradas como secciones-marco. Cada una hereda el mismo patrón Modo A/B y el mismo conjunto de exports Observable.

7.1 Fase 3 — Heat stress fisiológico

Pregunta: ¿Qué carga térmica fisiológica enfrenta la gente al combinar T + humedad + viento + radiación?

Datos: ERA5-Land hourly (T, dewpoint, viento 10 m, radiación SW) vía GEE para presente; NEX-GDDP-CMIP6 (tas, hurs, sfcWind) para futuro.

Índices: Humidex (espina dorsal — sólo T + RH); UTCI cuando estén disponibles viento y radiación.

Figuras canónicas: climatología mensual Humidex (línea+banda); mapa % de horas con Humidex > 40 (peligro); comparativa presente vs SSP5-8.5 cerca-futuro.

7.2 Fase 4 — Exposición

Pregunta: ¿Quién y qué activos están en las zonas de mayor hazard?

Datos: DANE CNPV-2018 por manzana (pirámide etaria, materiales vivienda); OSM para hospitales, escuelas, hogares geriátricos; WorldPop a 100 m como respaldo.

Figuras canónicas: mapa coroplético de densidad de población vulnerable (>65 + <5) por manzana; barchart de población expuesta por barrio cruzada con clases de hazard de Fase 1+2; markers de infraestructura crítica sobre el raster LST.

7.3 Fase 5 — Vulnerabilidad

Pregunta: ¿Quién es más susceptible al daño dada la misma exposición?

Datos: SISBÉN IV agregado por UTAM (no individual — restricción de datos personales); RIPS/SISPRO para morbilidad cardio-respiratoria; DANE materiales vivienda; distancia euclidiana a centro de salud nivel II+.

Índice compuesto: PCA o suma ponderada documentada sobre 4-6 variables. Aproximación MVP sin SISBÉN: % >65 + % <5 + % materiales precarios + distancia a salud.

Figuras canónicas: mapa coroplético del HVI; scatter biplot PCA; pirámide poblacional por nivel de vulnerabilidad.

7.4 Fase 6 — Impact / Risk

Pregunta: ¿Dónde priorizar la inversión en adaptación?

Datos: combina outputs de Fases 1-5 + función de impacto Stalhandske et al. (2022) calor arrow.r mortalidad, opcionalmente empaquetado como objeto Hazard + Exposures + ImpactFunc en CLIMADA.

Figuras canónicas: mapa final de riesgo en 5 clases; tabla AAI (Average Annual Impact) por localidad; curva costo-beneficio de tres medidas (arborización, techos reflectivos, refugios climáticos); diagrama Sankey de la cadena hazard arrow.r impact.

7. CONCLUSIONES PRELIMINARES (FASE 1)

1. **Calentamiento robusto bajo ambos escenarios.** Para Cartagena, incluso bajo el SSP2-4.5 (mitigación intermedia), la frecuencia anual de olas de calor con la definición operativa ($T_{\text{máx}} \geq 35^{\circ}\text{C} \wedge T_{\text{mín}} \geq 24^{\circ}\text{C}$, ≥ 3 días) se multiplica por un factor de ~ 2 -3 entre 2025 y 2070. Bajo SSP5-8.5 el factor es 5-6x.
2. **Cambio de régimen post-2045.** Los años con HWF superior al P95 del clima de referencia 1985-2014 pasan de ser eventos de probabilidad 5% a constituir la condición típica hacia 2055 (SSP2-4.5) y 2045 (SSP5-8.5).
3. **La grilla 25 km no resuelve la heterogeneidad intra-urbana.** Las acciones de mitigación local requieren obligatoriamente la Fase 2 (LST 30 m) para diferenciar el riesgo entre barrios. Lo que esta fase entrega es **la señal de cambio climático de fondo**, no el patrón espacial dentro de la ciudad.
4. **Implicaciones críticas para el SNGRD en Cartagena.** El incremento térmico proyectado presiona simultáneamente sobre salud pública (estrés cardiovascular y respiratorio), productividad laboral (sector portuario, construcción, turismo), seguridad alimentaria (cadena de frío urbana) y demanda eléctrica (acondicionamiento de aire en hospitales y centros educativos).

7.1 Limitaciones explícitas

- Datos sintéticos por pattern scaling — no captura cambios no-lineales de circulación regional (chorro caribeño, ZCIT).
- Ensamble pequeño (25 miembros) calibrado contra IPCC AR6 Atlas — subestima la cola superior de la incertidumbre.
- Resolución 25 km — no resuelve el efecto UHI urbano.
- Definición de umbral aún no validada contra serie histórica IDEAM operativa de Cartagena (estación Rafael Núñez).

7.2 Próximos pasos

1. **Activar la ruta GEE real** (Sección 4.1.B) sustituyendo el ensamble sintético por NEX-GDDP-CMIP6.
2. **Calibrar el umbral 35°C / 24°C** contra el percentil 95 móvil de la serie observada IDEAM Cartagena 1985-2024.
3. **Implementar Fase 2** (LST Landsat) — primer eslabón hacia la diferenciación intra-urbana.

4. **Articular con la Subdirección de Manejo del Desastre** para validar la utilidad del producto en activación de alertas y refugios climáticos.

8. REFERENCIAS

- Aznar-Siguán, G., & Bresch, D. N. (2019). CLIMADA v1: a global weather and climate risk assessment platform. *Geoscientific Model Development*, 12(7), 3085–3097. doi: 10.5194/gmd-12-3085-2019
- Bourgault, P., Huard, D., Smith, T. L., et al. (2023). xclim: xarray-based climate data analytics. *Journal of Open Source Software*, 8(85), 5415. doi: 10.21105/joss.05415
- Gorelick, N., Hancher, M., Dixon, M., et al. (2017). Google Earth Engine: Planetary-scale geospatial analysis for everyone. *Remote Sensing of Environment*, 202, 18–27. doi: 10.1016/j.rse.2017.06.031
- IDEAM (2015). Nuevos escenarios de cambio climático para Colombia 2011–2100. Instituto de Hidrología, Meteorología y Estudios Ambientales. Bogotá, Colombia.
- IPCC (2021). Climate Change 2021: The Physical Science Basis. Working Group I — Sixth Assessment Report. Cambridge University Press. <https://www.ipcc.ch/report/ar6/wg1/>
- IPCC (2021). Atlas Interactivo del IPCC AR6 — Regional Information. <https://interactive-atlas.ipcc.ch/>
- Jewson, S. (2021). Conversion of the Knutson et al. (2020) tropical cyclone climate change projections to risk model baselines. *Journal of Applied Meteorology and Climatology*, 60(10), 1517–1530. doi: 10.1175/jamc-d-21-0102.1
- Mendelsohn, R., et al. (2007). Climate analysis with satellite versus weather station data. *Climatic Change*, 81, 71–83. doi: 10.1007/s10584-006-9139-x
- Romero, M. & Santizo, M. (2026). Proyección de Temperatura en Montería, Córdoba bajo Escenarios de Cambio Climático. Zenodo. doi: 10.5281/zenodo.20061610
- Sarricolea, P. & Romero, H. (2010). Identificación de la isla de calor urbana en Santiago de Chile mediante imágenes satelitales Landsat ETM+. *Anales de la Sociedad Chilena de Ciencias Geográficas*, 163–168.
- Stalhandske, Z., et al. (2022). Projected impact of heat on mortality and labour productivity under climate change in Switzerland. *Natural Hazards and Earth System Sciences*, 22, 2531–2541. doi: 10.5194/nhess-22-2531-2022
- Stewart, I. D. & Oke, T. R. (2012). Local Climate Zones for urban temperature studies. *Bulletin of the American Meteorological Society*, 93(12), 1879–1900. doi: 10.1175/BAMS-D-11-00019.1
- Tebaldi, C. & Arblaster, J. M. (2014). Pattern scaling: Its strengths and limitations, and an update on the latest model simulations. *Climatic Change*, 122, 459–471. doi: 10.1007/s10584-013-1032-9
- Thrasher, B., et al. (2022). NASA Global Daily Downscaled Projections, CMIP6. *Scientific Data*, 9, 262. doi: 10.1038/s41597-022-01393-4
- CLIMAAX Consortium (2026). CLIMAAX Climate Risk Assessment Handbook — Urban heatwaves workflow. https://handbook.climaax.eu/notebooks/workflows/HEATWAVES/01_Urban_heatwaves/heatwave_intro.html
- UNGRD (2016). Plan Nacional de Gestión del Riesgo de Desastres 2015–2030. Unidad Nacional para la Gestión del Riesgo de Desastres. Bogotá, Colombia.
- UNDRR (2015). Sendai Framework for Disaster Risk Reduction 2015–2030. United Nations Office for Disaster Risk Reduction.
- Wulder, M. A., et al. (2019). Current status of Landsat program, science, and applications. *Remote Sensing of Environment*, 225, 127–147. doi: 10.1016/j.rse.2019.02.015

Data availability. Todos los outputs intermedios (NetCDF), figuras (PNG) y exports planos para Observable (JSON, GeoJSON) se generan al ejecutar este cuaderno. Quedan en las carpetas `data/intermediate/`, `data/final/` y `observable/data/` respectivamente.

Código. Apache-2.0. **Datos derivados.** CC-BY-4.0.

Cuaderno preparado en el marco del proyecto **Riesgos** (UNGRD-CyT). Cita sugerida pendiente DOI Zenodo.