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

Cuaderno reproducible · versión 2 · seis fases · marco IPCC AR6 (amenaza, exposición y vulnerabilidad)

RESUMEN

Cartagena de Indias es cálida todo el año: en la estación del IDEAM del aeropuerto, la temperatura máxima media es de 32,7 °C y la mínima de 25,8 °C. Este cuaderno reproducible evalúa cómo cambia el riesgo por calor urbano bajo los escenarios SSP2-4.5, SSP3-7.0 y SSP5-8.5 del CMIP6, con datos abiertos reales y en seis fases: amenaza climática, isla de calor urbana, estrés térmico, exposición, vulnerabilidad y riesgo relativo. El marco es el del IPCC AR6 y el del libro *Teoría y métodos de análisis de riesgos*.

Resultados principales.

1. **Amenaza climática.** Con percentiles locales, los días con temperatura máxima sobre el percentil 90 de 1985–2014 pasan de 10 % a una mediana de 50 % en 2061–2080 bajo SSP2-4.5, 63 % bajo SSP3-7.0 y 78 % bajo SSP5-8.5. Los días con Tmáx de 35 °C o más, anclados a la estación, pasan de unos 14 al año a 95, 153 y 189 (SSP2-4.5, SSP3-7.0 y SSP5-8.5).
2. **Isla de calor urbana.** La superficie construida es 6,1 °C más caliente en la media (3,9 °C en el percentil 90) que una referencia vegetada, y el calor de la superficie se concentra en la periferia construida más que en el centro turístico.
3. **Estrés térmico.** El Humidex máximo diario observado supera 45 («peligroso») en 46 % de los días.
4. **Riesgo relativo.** Olaya Herrera y El Pozón tienen más de la mitad de su población en las dos clases más altas.

Alcance. La vulnerabilidad es parcial, porque faltan los datos sociales por manzana, y el riesgo es relativo: ordena zonas, no estima pérdidas. La versión 2 reemplaza los datos sintéticos de la versión 1 y corrige defectos que alteraban sus conclusiones (sección 1).

> **Modo de ejecución:** el cuaderno corre completo y sin credenciales con los archivos de `data/`; los scripts de `scripts/` los regeneran desde las fuentes abiertas (sección 14).

1. ALCANCE DE ESTA VERSIÓN

La versión 2 reemplaza los datos sintéticos de la versión 1 por datos abiertos reales y completa las seis fases con el alcance que esos datos permiten. La tabla indica, para cada fase, qué pregunta responde, con qué datos y hasta dónde llega.

Fase	Pregunta	Datos	Alcance en esta versión
1. Amenaza climática	¿Cuánto cambian la frecuencia y la duración del calor?	Estación IDEAM (2019–2026), NEX-GDDP-CMIP6 (5 modelos; SSP2-4.5, SSP3-7.0 y SSP5-8.5; 1985–2080), ERA5-Land	Real. Completa para un punto: la celda de 25 km del aeropuerto
2. Isla de calor urbana	¿Dónde se concentra el calor de la superficie?	Landsat 8 y 9 Collection 2 (2018–2024, 30 m), ESA WorldCover 2021	Real. Completa
3. Estrés térmico	¿Qué aporta la humedad?	Temperatura y humedad relativa de la estación IDEAM (Humidex diario)	Real para el presente. Para el futuro solo una sensibilidad con humedad constante

4. Exposición	¿Cuánta gente e infraestructura está en cada clase de amenaza?	WorldPop 2020 (100 m), OpenStreetMap	Real. Población total, sin edades
5. Vulnerabilidad	¿Quién sufre más con la misma exposición?	Cobertura arbórea, distancia a hospitales y clínicas	**Parcial.** Faltan edad, condiciones sociales y salud (censo DANE 2018, SISBÉN, RIPS)
6. Riesgo	¿Dónde priorizar?	Combinación de las fases 2, 4 y 5	**Relativo.** Clasifica zonas; no calcula pérdidas ni la pérdida anual esperada

Qué se corrigió respecto de la versión 1

La auditoría de la versión 1 encontró problemas que cambian las conclusiones, y por eso esta versión no es solo una ampliación.

- **Todo era sintético.** El ensamble de temperaturas y el mapa de isla de calor se generaron con una fórmula. Las «conclusiones» (por ejemplo, +5 a +7 °C en el centro histórico) salían de los puntos calientes que el propio código dibujó, es decir, eran circulares.
- **La climatología base no era verificable.** Se atribuía a la estación del IDEAM un promedio 1985–2014, pero el conjunto abierto de datos.gov.co solo empieza en septiembre de 2019.
- **El generador sobreestimaba los extremos.** Con el mismo clima medio, producía unos 46 días al año con Tmáx ≥ 35 °C; la estación observa unos 14. El ruido diario era independiente entre días, de modo que no reproducía la persistencia que define una ola de calor.
- **La condición de Tmín ≥ 24 °C no discriminaba.** En la estación se cumple en 9 de cada 10 días, así que la definición «Tmáx ≥ 35 °C y Tmín ≥ 24 °C» equivale casi a «Tmáx ≥ 35 °C».
- **El mapa de cambio espacial era un valor inventado** y la numeración de secciones estaba rota (no existía la sección 5 y había dos «7»).

2. MARCO TEÓRICO

2.1 Riesgo climático: amenaza, exposición y vulnerabilidad

El riesgo de desastres es una pérdida potencial que resulta de combinar un fenómeno físico, lo que está en su camino y la fragilidad de ello. La Ley 1523 de 2012 lo define como los daños o pérdidas potenciales que pueden presentarse por eventos físicos peligrosos, determinados por la vulnerabilidad de los elementos expuestos (Congreso de la República de Colombia, 2012). El Atlas de Riesgo de Colombia adopta tres componentes que no se definen unos sin otros: amenaza, exposición y vulnerabilidad (Unidad Nacional para la Gestión del Riesgo de Desastres, 2019). En el contexto del cambio climático, el riesgo cambia por dos vías: se altera la frecuencia, la intensidad y la distribución de las amenazas relacionadas con el clima, y se afectan la exposición y la vulnerabilidad (GFDRR and World Bank, 2025).

Este cuaderno sigue ese esquema. La amenaza se descompone en tres capas que responden preguntas distintas: el clima de fondo de la ciudad (Fase 1), el modulador urbano que decide qué barrios se calientan más (Fase 2) y la carga que el calor impone al cuerpo cuando se suma la humedad (Fase 3). La exposición (Fase 4) y la vulnerabilidad (Fase 5) completan el riesgo (Fase 6). El libro *Teoría y métodos de análisis de riesgos* desarrolla cada componente y se cita aquí como referencia común (ver la sección 13).

2.2 La amenaza: qué es una ola de calor

Una ola de calor es una anomalía local, no un valor absoluto. Los perfiles de información de amenazas de UNDRR y el Consejo Internacional de Ciencias la describen como un período marcado e inusual de tiempo caluroso sobre una región, que persiste al menos dos o tres días y noches consecutivos durante la época caliente del año según las condiciones climatológicas locales, con condiciones térmicas registradas por encima de umbrales dados (perfil MH0501, que cita a la OMM y la OMS) (UNDRR & ISC, 2025). Esa definición deja la elección del umbral a la climatología local, y de ahí se desprenden los demás criterios de este cuaderno.

Las metodologías difieren en cómo fijan el umbral. CLIMAAX ofrece dos para Europa: EuroHEAT, que exige al menos dos días con la temperatura aparente máxima y la mínima sobre el percentil 90 de la distribución mensual de 1971–2000, y Xclim, que usa umbrales absolutos de temperatura máxima y mínima y una duración mínima definidos por quien analiza (CLIMAAX

Consortium, 2025). Ambas usan proyecciones regionales de 12 km que no incluyen la isla de calor urbana y, por eso, no sirven para comparar barrios (CLIMAAX Consortium, 2025).

En el Caribe colombiano hay un antecedente cercano que usa percentiles. En las islas de San Andrés, Providencia y Santa Catalina, Esteban-Cantillo et al. definen la ola de calor como una racha de al menos tres días (corta) o cinco (larga) con temperatura máxima sobre el percentil 90, siguiendo los índices del ETCCDI; encuentran que las olas de calor se triplicaron en frecuencia y duplicaron su duración máxima desde finales de los años ochenta, y que para 2071–2100 los días calurosos aumentan entre 60 % y 89 % en los cuatro escenarios SSP que evaluaron (Esteban-Cantillo et al., 2024). Ávila-Díaz et al. documentan la misma familia de índices (días cálidos TX90p y duración de rachas cálidas) para América Latina y el Caribe (Avila-Diaz et al., 2023).

De estas fuentes se toman tres decisiones, que la sección 4 desarrolla. Primero, el umbral principal es un percentil de la climatología de cada modelo, no un número copiado de Europa. Segundo, se exige que el día y la noche superen el umbral, como en EuroHEAT, y se reporta también la variante de solo Tmáx del antecedente colombiano. Tercero, el umbral absoluto de 35 °C se conserva solo como indicador comunicable y anclado a la estación, porque un umbral absoluto aplicado a un modelo con sesgo de varios grados no mide el clima de la ciudad.

2.3 Isla de calor urbana y temperatura de superficie

La isla de calor urbana nace de la cobertura del suelo. Las superficies impermeables y los edificios retienen energía y la liberan como calor, y la vegetación enfría por evapotranspiración. El indicador más usado para mapearla es la temperatura de superficie terrestre (LST) de Landsat, que CLIMAAX emplea a 30 m, cada 16 días y con datos faltantes en los días nublados (CLIMAAX Consortium, 2025). La LST no es la temperatura del aire que determina el estrés fisiológico, y esa diferencia es una de las incertidumbres del método.

La evidencia local apunta en la misma dirección. En Soledad (Atlántico), Pérez-Arévalo et al. midieron temperatura del aire y de superficie en distintas zonas climáticas locales (el marco de Stewart y Oke (Stewart & Oke, 2012)) y hallaron que un menor factor de visión del cielo y un mayor índice de área foliar se asocian con temperaturas más bajas y mejor circulación del aire, mientras que los cañones urbanos densos registran temperaturas más altas y menos viento (Pérez-Arévalo et al., 2025). Un estudio sobre bosques urbanos y exposición al calor extremo en Bogotá apunta al arbolado como medida de reducción, pero aquí solo se conoce por su título (Giraldo-Charria et al., 2025).

2.4 Estrés térmico y humedad

La temperatura del aire no basta para medir el calor que siente una persona. CCDR-tools señala que la incomodidad aumenta cuando el calor se asocia con humedad alta, que el estrés térmico puede reducir la productividad y los ingresos, y que los eventos extremos aumentan la morbilidad y la mortalidad; para medirlo lista dos índices, la temperatura de globo y bulbo húmedo (WBGT) y el Índice Universal de Clima Térmico (UTCI) (GFDRR and World Bank, 2025). Ambos exigen datos de viento y radiación que la estación de referencia no publica. Aquí se usa el Humidex, que solo requiere temperatura y humedad relativa (Masterton & Richardson, 1979). Sus categorías de molestia (30 a 39 algo de molestia, 40 a 45 molestia intensa, más de 45 peligroso) son las de Environment Canada y quedan **por verificar contra el original**.

2.5 Exposición, vulnerabilidad y alerta temprana

CLIMAAX construye el riesgo local de calor cruzando la exposición, medida con la LST, con la vulnerabilidad, medida con la población mayor de 65 años y menor de 5 años (WorldPop, 100 m), en una matriz de diez por diez clases; el nivel de riesgo resultante es relativo al área analizada (CLIMAAX Consortium, 2025). Para Colombia, una revisión narrativa reciente indica que las mujeres gestantes tienen mayor riesgo de parto prematuro con altas temperaturas, que adultos y niños pueden sufrir deshidratación y que los mayores de 65 años tienen mayor riesgo de enfermedades cardíacas; además, señala que el Plan Decenal de Salud Pública 2022–2031 (Resolución 2367 de 2023) fija la meta de que los territorios cuenten con sistemas de alerta temprana hacia 2031 (Salazar-Ceballos & Álvarez-Miño, 2025) (versión anticipada, aún sin edición final). Una perspectiva reciente sobre alertas añade que reducir la mortalidad exige entender cómo responde la gente a ellas (Bubeck et al., 2025) (citada por su título).

2.6 Escenarios y proyecciones

Los escenarios sirven para explorar futuros posibles y no para predecir el que ocurrirá (CLIMAAX Consortium, 2025). Las proyecciones del CMIP6 combinan trayectorias de emisiones con trayectorias socioeconómicas compartidas (SSP) (Riahi et al., 2017). CCDR-tools recomienda SSP2-4.5 y SSP3-7.0 como escenarios principales de planificación y reserva SSP5-8.5 como prueba de estrés (GFDRR and World Bank, 2025); este cuaderno adopta esa jerarquía y la declara en cada resultado. Los

datos usados, NEX-GDDP-CMIP6, son proyecciones diarias con corrección de sesgo y reducción de escala estadística a 0,25° (Thrasher et al., 2022).

Dos advertencias de la literatura colombiana guían el uso de las proyecciones. La primera: Arias et al. mostraron que la Tercera Comunicación Nacional sobre Cambio Climático promedió las trayectorias RCP, con lo cual se pierden escenarios alternos que importan para pensar futuros distintos (Arias et al., 2022); por eso aquí cada escenario se reporta por separado y nunca se promedian. La segunda: una mayor resolución no garantiza un mejor desempeño en los extremos, según una evaluación de modelos de alta resolución para América Latina y el Caribe (Avila-Diaz et al., 2023); por eso se valida cada conjunto contra la estación antes de usarlo.

2.7 Contexto de Cartagena

Cartagena de Indias (10,42° N, 75,52° O) está al nivel del mar, junto a una bahía y a la Ciénaga de la Virgen, y su clima es cálido todo el año sin estacionalidad térmica marcada. Esa estabilidad explica por qué una definición de ola de calor copiada de latitudes templadas, que parte de una época caliente bien delimitada, funciona mal aquí, y por qué se distinguen la ola de calor (anomalía respecto de la climatología local) y el calor persistente propio del clima. Las cifras climáticas de la ciudad no se citan de memoria en este texto: se calculan con la estación del IDEAM en la sección 6.

3. CONFIGURACIÓN DEL ENTORNO

El cuaderno corre completo sin credenciales. Los datos que usa están en `data/` (series diarias de la estación IDEAM, ERA5-Land, la celda de NEX-GDDP-CMIP6 y los rásteres de Landsat, WorldCover y WorldPop recortados al área de estudio), de modo que no hace falta descargar nada. Los scripts de `scripts/` regeneran esos archivos desde las fuentes originales (sección 14).

```
# Si el cuaderno se abre en Colab sin el repositorio, se clona (los datos están en data/)
import os, subprocess
from pathlib import Path
if not Path('data/ideam_diario_rafael_nunez.csv').exists():
    subprocess.run(['git', 'clone', '--depth', '1',
                    'https://github.com/scr-ungrd/cuadernos-reproducibles-crc002.git', '_repo'], check=True)
    os.chdir('_repo')
try:
    import rasterio, xarray, netCDF4, folium, xclim            # noqa: F401
except ImportError:
    %pip install -q xarray netcdf4 rasterio folium xclim matplotlib scipy pandas
print('Directorio de trabajo:', Path.cwd())
```

```
Directorio de trabajo: /Users/mauricio/Documents/Data-science-local/A-SCR/Cuadernos-reproducibles/cuadernos-reproducibles-crc002
```

```
import warnings
warnings.filterwarnings('ignore')
import json
from pathlib import Path
import numpy as np
import pandas as pd
import xarray as xr
import matplotlib.pyplot as plt
from matplotlib.colors import LinearSegmentedColormap
from matplotlib.patches import Rectangle, Patch
import rasterio
from rasterio.warp import reproject, Resampling, transform_bounds
from scipy import stats
import folium
import xclim.indices as xci

plt.rcParams.update({
    'figure.dpi': 110, 'savefig.dpi': 150, 'font.size': 10, 'axes.titlesize': 11,
    'axes.titleweight': 'bold', 'axes.spines.top': False, 'axes.spines.right': False,
    'axes.grid': True, 'grid.alpha': 0.25, 'legend.frameon': True, 'legend.framealpha': 0.9,
})

ESC = ['ssp245', 'ssp370', 'ssp585']
ESC_LABEL = {'ssp245': 'SSP2-4.5 (principal)', 'ssp370': 'SSP3-7.0 (intermedio)', 'ssp585': 'SSP5-8.5 (prueba de estrés)'}
```

```
ESC_COLOR = {'ssp245': '#2a9d8f', 'ssp370': '#e9a03b', 'ssp585': '#c1121f'}
HEAT_CMAP = LinearSegmentedColormap.from_list('ungrd_heat', ['#FAF6F0', '#F4D58D', '#D9A21B', '#C73E1D', '#5A1E0E'],
N=256)

DATA = Path('data'); OUT_FINAL = DATA / 'final'; OUT_OBS = Path('observable/data')
for p in (OUT_FINAL, OUT_OBS):
    p.mkdir(parents=True, exist_ok=True)
print('Librerías cargadas.')
```

```
Librerías cargadas.
```

```
# ======================================================================
# Parámetros del análisis: Cartagena de Indias
# ======================================================================
CIUDAD = 'Cartagena de Indias, Bolívar'
ESTACION = {'nombre': 'Aeropuerto Rafael Núñez (IDEAM 0014015080)', 'lat': 10.44725, 'lon': -75.51603}
BBOX = {'west': -75.62, 'south': 10.30, 'east': -75.42, 'north': 10.52}      # área de estudio (AOI)

REF = ('1985', '2014')     # periodo de referencia de los modelos (el mismo del CMIP6 histórico)
PCTL = 0.90                # percentil que define "día caluroso" (calculado por día del año y por modelo)
VENTANA = 5                # días de la ventana móvil para el percentil (ETCCDI)
MIN_DIAS = 3               # duración mínima de una ola de calor
UMBRAL_TX, UMBRAL_TN = 35.0, 24.0   # definición absoluta de la versión 1 del cuaderno: aquí se evalúa, no se impone

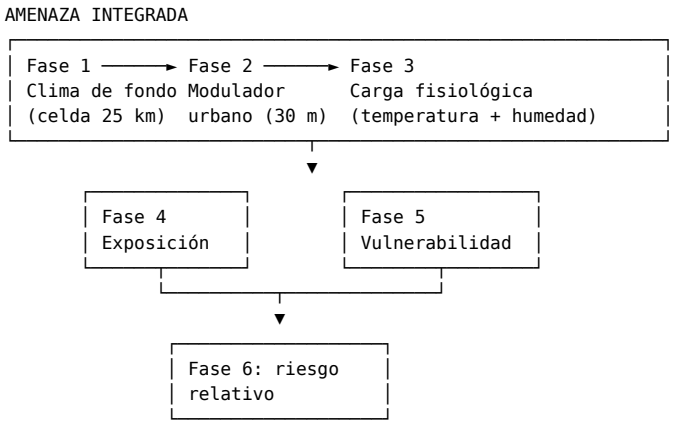
PERIODOS = {'2021–2040': ('2021', '2040'), '2041–2060': ('2041', '2060'), '2061–2080': ('2061', '2080')}
REF_OBS = ('2019', '2025')  # años con datos de la estación IDEAM que se usan para anclar los índices absolutos

print(f'Ciudad: {CIUDAD}')
print(f'Ola de calor (definición principal): Tmáx > P{int(PCTL*100)} y Tmín > P{int(PCTL*100)} del día del año, '
      f'≥{MIN_DIAS} días, con la referencia {REF[0]}–{REF[1]} de cada modelo')
```

```
Ciudad: Cartagena de Indias, Bolívar
Ola de calor (definición principal): Tmáx > P90 y Tmín > P90 del día del año, ≥3 días, con la referencia 1985–2014
de cada modelo
```

4. METODOLOGÍA

4.1 Cadena de seis fases



El orden sigue el flujo general del libro de teoría: definir el alcance, caracterizar la amenaza, la exposición y la vulnerabilidad, combinarlas y declarar la incertidumbre.

4.2 Definición de ola de calor

La definición principal es una racha de **al menos 3 días seguidos** en los que, a la vez, la temperatura máxima T_x y la mínima T_n superan su percentil 90 del día del año:

$$T_x(d) > P^{x}_{90}(\mathrm{doy}(d)) \quad \wedge \quad T_n(d) > P^{n}_{90}(\mathrm{doy}(d))$$

El percentil se calcula con una ventana móvil de 5 días centrada en cada día del año, sobre 1985–2014, **por separado para cada modelo**. Así cada modelo se compara con su propio clima y el sesgo sistemático desaparece de la comparación, que es el principio del percentil relativo. La variante de solo T_x (sin condición nocturna) reproduce la definición del antecedente colombiano (Esteban-Cantillo et al., 2024).

4.3 Índices

- **TX90p:** porcentaje de días del año con T_x sobre su percentil 90. Por construcción vale 10 % en la referencia.
- **HWF:** número de olas de calor por año.
- **HWD:** número de días del año que están dentro de una ola de calor.
- **HWML:** duración, en días, de la ola más larga del año.
- **Días con $T_x \geq 35$ °C:** indicador absoluto y comunicable, que se calcula anclado a la estación (sección 4.4).

El conteo de olas se verifica contra `xclim.heat_wave_frequency` (sección 6.2).

4.4 De los modelos a la estación: dos caminos

Un modelo de 25 km sobre una celda costera mezcla tierra y mar, y su T_x difiere de la que mide un termómetro en el aeropuerto. Por eso hay dos caminos, cada uno con su límite.

1. **Relativo (índices TX90p, HWF, HWD, HWML).** El umbral sale de la climatología de cada modelo. No exige que el modelo reproduzca el valor absoluto de la ciudad.
2. **Absoluto anclado (días con $T_x \geq 35$ °C).** A cada día observado en la estación (2019–2026) se le suma el cambio por cuantil que proyecta el modelo entre 2019–2025 y el periodo futuro:

$$T'_x = T_x + \left[\, Q^{\mathrm{mod}}_{\mathrm{fut}}(p) - Q^{\mathrm{mod}}_{2019-2025}(p) \,\right], \qquad p = F_{\mathrm{obs}}(T_x)$$

La frecuencia de días con $T'_x \geq 35$ °C, multiplicada por 365, estima los días por año. Este método conserva la relación entre días observados, pero hereda su muestra corta (unos 5 años completos de datos) y su variabilidad interanual.

4.5 Isla de calor urbana

Se construye un compuesto de LST con todas las escenas Landsat 8 y 9 de 2018 a 2024 con menos de 40 % de nubes en la escena, enmascarando nubes, sombras y cirros con la banda de calidad `QA_PIXEL`. Por píxel se guardan la media y el percentil 90 de las fechas válidas, y los mismos estadísticos para la época seca (diciembre a marzo). La cobertura del suelo (construido, árboles, agua, manglar) viene de ESA WorldCover 2021 agregada a 30 m. Las imágenes Landsat se toman hacia las 10:30 de la mañana: describen el calor de la mañana, no el de la tarde.

4.6 Estrés térmico

El Humidex combina temperatura T (°C) y presión de vapor e (hPa):

$$H = T + 0,5555\,(e - 10), \qquad e = 6,11 \exp\!\left[5417,753\left(\frac{1}{273,16} - \frac{1}{T_d + 273,15}\right)\right]$$

con el punto de rocío T_d calculado de T y la humedad relativa por la fórmula de Magnus (Masterton & Richardson, 1979).

4.7 Exposición, vulnerabilidad y riesgo relativo

La exposición es la población por celda de 100 m (WorldPop). La amenaza espacial es la LST P90. La vulnerabilidad parcial es el promedio de los rangos percentiles del déficit de arbolado y de la distancia al hospital o clínica más cercano. El riesgo relativo es la media geométrica de los rangos percentiles de las tres componentes, y sus clases son quintiles de población. Como el resultado es un ranking, dice **dónde** es mayor el riesgo dentro de la ciudad y no cuánto vale. Esa característica es la que tiene la matriz de CLIMAAX, cuyo nivel de riesgo es relativo al área elegida (CLIMAAX Consortium, 2025).

4.8 Verificación y sensibilidad

- **Conteo de olas:** igualdad exacta con xclim en una serie de prueba.
- **Modelos contra estación:** gráfico de cuantiles de T_x (Figura 1c).
- **Fórmula del riesgo:** cinco variantes y la coincidencia de su 20 % superior (sección 11).
- **Escenarios:** SSP2-4.5 como principal, SSP3-7.0 intermedio y SSP5-8.5 como prueba de estrés, siempre por separado.

5. DATOS

Conjunto	Qué es	Resolución y periodo	Cita
Estación IDEAM Aeropuerto Rafael Núñez (0014015080)	Temperatura máxima y mínima del aire; temperatura y humedad relativa	Diaria y horaria, septiembre de 2019 a noviembre de 2026	(IDEAM, 2026)
ERA5-Land vía Open-Meteo	Reanálisis de temperatura y punto de rocío	9 km, 1985–2024; solo contraste	(Muñoz-Sabater et al., 2021; Zippenfenig, 2023)
NEX-GDDP-CMIP6	Tmáx y Tmín diarias, corregidas y reducidas de escala	0,25°, 1985–2080; 5 modelos (ACCESS-CM2, MPI-ESM1-2-HR, MRI-ESM2-0, NorESM2-MM, CMCC-ESM2); SSP2-4.5, SSP3-7.0, SSP5-8.5	(Thrasher et al., 2022)
Landsat 8 y 9 Collection 2 Level 2	Temperatura de superficie (banda térmica)	30 m, 2018–2024	(Wulder et al., 2019)
ESA WorldCover 2021	Cobertura del suelo	10 m, agregada a 30 m	(Zanaga et al., 2022)
WorldPop 2020	Población (restringida a edificaciones)	100 m	(WorldPop, 2020)
OpenStreetMap	Hospitales, clínicas, hogares de mayores y colegios	Puntos, consulta de noviembre de 2026	© colaboradores de OpenStreetMap (ODbL)

El área de estudio es el recuadro de 20 por 22 km entre 75,62° y 75,42° O, y entre 10,30° y 10,52° N. Los cinco modelos de NEX-GDDP-CMIP6 se tomaron con un solo miembro (r1i1p1f1) cada uno; cinco modelos no cubren la incertidumbre completa del CMIP6, y el rango entre ellos se reporta como rango, no como intervalo de confianza.

```
ideam = pd.read_csv(DATA / 'ideam_diario_rafael_nunez.csv', parse_dates=['fecha'], index_col='fecha')
ideam_hx = pd.read_csv(DATA / 'ideam_humidex_diario_rafael_nunez.csv', parse_dates=['fecha'], index_col='fecha')
era5 = pd.read_csv(DATA / 'era5land_diario_cartagena.csv', parse_dates=['fecha'], index_col='fecha')
nex = xr.open_dataset(DATA / 'nex_gddp_celda_aeropuerto.nc')
MODELOS = [str(m) for m in nex.modelo.values]
ideam_ok = ideam[['tx', 'tn']].dropna()

print(f'IDEAM: {ideam.index.min().date()} a {ideam.index.max().date()}  |  días con Tmáx y Tmín completas:
{len(ideam_ok)} de {len(ideam)}')
print(f'ERA5-Land: {era5.index.min().date()} a {era5.index.max().date()}  |  {len(era5)} días')
print(f'NEX-GDDP-CMIP6: {len(MODELOS)} modelos {MODELOS}  |  {nex.time.dt.year.min().item()}–
{nex.time.dt.year.max().item()}')

cobertura = (ideam.assign(anio=ideam.index.year)
             .groupby('anio').agg(dias_registrados=('tx', 'size'),
                                  dias_validos=('tx', lambda s: int(s.notna().sum()))))
cobertura['tx_media'] = ideam_ok.groupby(ideam_ok.index.year).tx.mean().round(1)
cobertura['tn_media'] = ideam_ok.groupby(ideam_ok.index.year).tn.mean().round(1)
cobertura['dias_tx_ge_35'] = ideam_ok.groupby(ideam_ok.index.year).tx.apply(lambda s: int((s >= UMBRAL_TX).sum()))
cobertura
```

```
IDEAM: 2019-09-18 a 2026-11-01  |  días con Tmáx y Tmín completas: 1928 de 2602
ERA5-Land: 1985-01-01 a 2024-12-31  |  14610 días
NEX-GDDP-CMIP6: 5 modelos ['ACCESS-CM2', 'MPI-ESM1-2-HR', 'MRI-ESM2-0', 'NorESM2-MM', 'CMCC-ESM2']  |  1985–2080
```

	dias_registrados	dias_validos	tx_media	tn_media	dias_tx_ge_35
anio					
2019	105	71	33.0	25.7	1
2020	366	349	33.0	25.7	19

	dias_registrados	dias_validos	tx_media	tn_media	dias_tx_ge_35
anio					
2021	365	332	33.1	25.4	21
2022	365	204	32.5	25.2	4
2023	365	202	32.1	25.3	3
2024	366	180	32.6	26.4	7
2025	365	327	32.3	26.2	5
2026	305	275	32.8	26.4	13

6. FASE 1: AMENAZA CLIMÁTICA

6.1 Lo que dicen las observaciones

Antes de proyectar, se mira qué mide la estación. La primera tabla resume la cobertura de cada año; los días sin datos suficientes (menos de 20 horas con registro) se excluyen. La segunda mide qué tan discriminantes son los umbrales absolutos de la versión 1.

```
# Qué tan discriminantes son los umbrales absolutos de la versión 1, con datos observados
tab = pd.DataFrame({
    'Medida': ['Tmáx media (°C)', 'Tmín media (°C)', 'Días con Tmín ≥ 24 °C', 'Días con Tmáx ≥ 35 °C',
               'Días con Tmáx ≥ 35 °C y Tmín ≥ 24 °C', 'P90 de Tmáx (°C)', 'P90 de Tmín (°C)'],
    'Estación IDEAM 2019–2026': [
        f'{ideam_ok.tx.mean():.1f}', f'{ideam_ok.tn.mean():.1f}',
        f'{100*(ideam_ok.tn >= UMBRAL_TN).mean():.0f} %', f'{100*(ideam_ok.tx >= UMBRAL_TX).mean():.1f} %  (≈
{365*(ideam_ok.tx >= UMBRAL_TX).mean():.0f} días/año)',
        f'{100*((ideam_ok.tx >= UMBRAL_TX) & (ideam_ok.tn >= UMBRAL_TN)).mean():.1f} %',
        f'{ideam_ok.tx.quantile(.9):.1f}', f'{ideam_ok.tn.quantile(.9):.1f}']})
tab
```

	Medida	Estación IDEAM 2019–2026
0	Tmáx media (°C)	32.7
1	Tmín media (°C)	25.8
2	Días con Tmín ≥ 24 °C	92 %
3	Días con Tmáx ≥ 35 °C	3.8 % (≈ 14 días/año)
4	Días con Tmáx ≥ 35 °C y Tmín ≥ 24 °C	3.7 %
5	P90 de Tmáx (°C)	34.2
6	P90 de Tmín (°C)	27.4

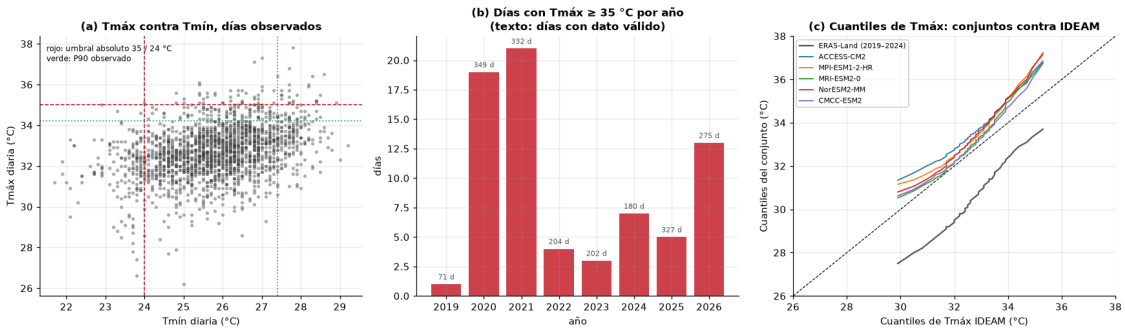
Figura 1. Observaciones de la estación IDEAM Aeropuerto Rafael Núñez: (a) Tmáx contra Tmín con los umbrales absolutos y el percentil 90 observado, (b) días con Tmáx ≥ 35 °C por año, (c) cuantiles de Tmáx de ERA5-Land y de los cinco modelos contra los de la estación.

```
# Figura 1. Observaciones IDEAM: ¿los umbrales absolutos separan algo?
fig, ax = plt.subplots(1, 3, figsize=(16, 4.8))
a = ax[0]
a.scatter(ideam_ok.tn, ideam_ok.tx, s=6, alpha=0.35, color='#444')
a.axhline(UMBRAL_TX, color='#c1121f', ls='--', lw=1); a.axvline(UMBRAL_TN, color='#c1121f', ls='--', lw=1)
a.axhline(ideam_ok.tx.quantile(.9), color='#2a9d8f', ls=':', lw=1.2); a.axvline(ideam_ok.tn.quantile(.9),
color='#2a9d8f', ls=':', lw=1.2)
a.set_xlabel('Tmín diaria (°C)'); a.set_ylabel('Tmáx diaria (°C)')
a.set_title('(a) Tmáx contra Tmín, días observados')
a.text(0.02, 0.97, 'rojo: umbral absoluto 35 / 24 °C\nverde: P90 observado', transform=a.transAxes, va='top',
fontsize=8)
```

```
a = ax[1]
cob = cobertura.copy()
a.bar(cob.index, cob.dias_tx_ge_35, color='#c1121f', alpha=.8)
for x, (v, d) in zip(cob.index, zip(cob.dias_tx_ge_35, cob.dias_validos)):
    a.text(x, v + 0.4, f'{d} d', ha='center', fontsize=7, color='#555')
a.set_title('(b) Días con Tmáx ≥ 35 °C por año\n(texto: días con dato válido)'); a.set_ylabel('días');
a.set_xlabel('año')

a = ax[2]
q = np.linspace(0.02, 0.98, 49)
j = era5[['tx']].join(ideam_ok[['tx']], rsuffix='_ideam', how='inner').dropna()
a.plot(ideam_ok.tx.quantile(q), era5.loc[era5.index.year.isin(range(2019, 2025)), 'tx'].quantile(q), color='#555',
label='ERA5-Land (2019–2024)')
sesgo = {}
for m in MODELOS:
    s = nex.sel(modelo=m, escenario='ssp245').tx.to_series().loc[REF_OBS[0]:REF_OBS[1]]
    a.plot(ideam_ok.tx.quantile(q), s.quantile(q), lw=1.2, label=m)
    sesgo[m] = (s.quantile(.5) - ideam_ok.tx.quantile(.5), s.quantile(.9) - ideam_ok.tx.quantile(.9))
lim = [26, 38]; a.plot(lim, lim, 'k--', lw=0.8)
a.set_xlim(lim); a.set_ylim(lim); a.set_xlabel('Cuantiles de Tmáx IDEAM (°C)'); a.set_ylabel('Cuantiles del conjunto
(°C)')
a.set_title('(c) Cuantiles de Tmáx: conjuntos contra IDEAM'); a.legend(fontsize=7, loc='upper left')
plt.tight_layout(); plt.savefig(OUT_FINAL / 'fig1_observaciones_ideam.png', bbox_inches='tight'); plt.show()

dif = (j.tx - j.tx_ideam)
print(f'ERA5-Land menos IDEAM (Tmáx, {len(j)} días comunes): sesgo medio {dif.mean():.2f} °C, RMSE
{np.sqrt((dif**2).mean()):.2f} °C, correlación {j.tx.corr(j.tx_ideam):.2f}')
print('Sesgo de Tmáx de cada modelo (2019–2025) frente a la estación, en la mediana y el P90 (°C):')
print(pd.DataFrame(sesgo, index=['mediana', 'P90']).T.round(1).to_string())
```




```
ERA5-Land menos IDEAM (Tmáx, 1326 días comunes): sesgo medio -2.31 °C, RMSE 2.97 °C, correlación 0.15
Sesgo de Tmáx de cada modelo (2019–2025) frente a la estación, en la mediana y el P90 (°C):
              mediana  P90
ACCESS-CM2        0.8  1.2
MPI-ESM1-2-HR     0.5  1.2
MRI-ESM2-0        0.3  1.1
NorESM2-MM        0.6  1.2
CMCC-ESM2         0.2  0.8
```

Lectura. Tres hallazgos de las observaciones condicionan el resto del análisis.

1. **La condición nocturna de 24 °C no discrimina.** Se cumple en 92 % de los días, de modo que la definición de la versión 1 («Tmáx ≥ 35 °C y Tmín ≥ 24 °C») identifica prácticamente los días con Tmáx ≥ 35 °C, que son 3,8 % de los días, unos 14 al año (Figura 1a). En cambio, el percentil 90 de la Tmín observada es 27,4 °C, y ese sí separa noches excepcionales.
2. **La variabilidad entre años es grande.** Entre los años con más de 200 días de datos, los días con Tmáx ≥ 35 °C van de 3 (2023) a 21 (2021) (Figura 1b). Solo tres años están casi completos (2020, 2021 y 2025), así que cualquier valor de referencia anclado en la estación hereda esa dispersión.
3. **El reanálisis no sustituye a la estación.** ERA5-Land subestima la Tmáx en promedio 2,3 °C en este punto costero (correlación diaria de 0,15), porque su celda mezcla tierra y mar; por eso solo se usa como contraste. Los cinco modelos de NEX-GDDP-CMIP6, en cambio, quedan por encima de la estación (Figura 1c): entre 0,2 y 0,8 °C en la mediana de la Tmáx

y entre 0,8 y 1,2 °C en el percentil 90. Es un sesgo cálido que cada modelo arrastra y que el umbral relativo elimina de la comparación.

6.2 Índices por modelo y escenario

Se implementan la definición y los índices de la sección 4. La celda siguiente define las funciones y las verifica contra `xclim`; la siguiente calcula los índices anuales de cada modelo y escenario con umbrales propios del modelo.

```
def doy365(idx):
    """Día del año en un calendario de 365 días (el 29 de febrero se asigna al 28)."""
    d = idx.dayofyear.values.copy()
    d = d - ((idx.is_leap_year) & (idx.month > 2)).astype(int)
    d[(idx.month == 2) & (idx.day == 29)] = 59
    return d


def umbral_doy(s, ref=REF, q=PCTL, ventana=VENTANA):
    """Percentil q por día del año (ventana móvil centrada, circular) sobre el periodo de referencia."""
    r = s.loc[ref[0]:ref[1]]
    d, v = doy365(r.index), r.values
    medio = ventana // 2
    out = np.empty(365)
    for k in range(1, 366):
        vec = [((k - 1 + j) % 365) + 1 for j in range(-medio, medio + 1)]
        out[k - 1] = np.nanquantile(v[np.isin(d, vec)], q)
    return out


def rachas(caliente, minimo=MIN_DIAS):
    """Eventos (>= minimo días seguidos): devuelve posiciones de inicio y duraciones."""
    x = np.diff(np.r_[0, caliente.astype(np.int8), 0])
    ini, fin = np.flatnonzero(x == 1), np.flatnonzero(x == -1)
    largo = fin - ini
    k = largo >= minimo
    return ini[k], largo[k]


def indices_anuales(idx, caliente):
    """HWF (eventos/año), HWD (días en ola/año) y HWML (duración del evento más largo), año por año."""
    filas = []
    for a in np.unique(idx.year):
        s = idx.year == a
        _, L = rachas(caliente[s])
        filas.append((int(a), len(L), int(L.sum()), int(L.max()) if len(L) else 0))
    return pd.DataFrame(filas, columns=['anio', 'HWF', 'HWD', 'HWML']).set_index('anio')


# Verificación cruzada con xclim (umbrales absolutos 35 / 24 °C, 3 días) en una serie de un modelo
m0 = MODELOS[0]
s_tx = nex.sel(modelo=m0, escenario='ssp585').tx.to_series()
s_tn = nex.sel(modelo=m0, escenario='ssp585').tn.to_series()
tx_da = xr.DataArray(s_tx.values, dims='time', coords={'time': s_tx.index}, attrs={'units': 'degC'}, name='tasmax')
tn_da = xr.DataArray(s_tn.values, dims='time', coords={'time': s_tn.index}, attrs={'units': 'degC'}, name='tasmin')
hwf_xclim = xci.heat_wave_frequency(tn_da, tx_da, thresh_tasmin='24 degC', thresh_tasmax='35 degC', window=3,
freq='YS')
mio = indices_anuales(s_tx.index, ((s_tx >= 35) & (s_tn >= 24)).values).HWF
iguales = np.array_equal(hwf_xclim.values.astype(int), mio.values)
print(f'Conteo propio de olas de calor igual al de xclim.heat_wave_frequency ({m0}, SSP5-8.5, 1985–2080):
{iguales}')
```

```
Conteo propio de olas de calor igual al de xclim.heat_wave_frequency (ACCESS-CM2, SSP5-8.5, 1985–2080): True
```

```
# Índices anuales por modelo y escenario, con umbrales propios de cada modelo (percentil sobre 1985–2014)
filas = []
for m in MODELOS:
    for e in ESC:
        tx = nex.sel(modelo=m, escenario=e).tx.to_series()
```

```
        tn = nex.sel(modelo=m, escenario=e).tn.to_series()
        ptx, ptn = umbral_doy(tx), umbral_doy(tn)
        d = doy365(tx.index) - 1
        cx, cn = tx.values > ptx[d], tn.values > ptn[d]
        A = indices_anuales(tx.index, cx & cn)            # definición principal: día y noche calurosos
        B = indices_anuales(tx.index, cx)                  # variante: solo Tmáx (como Esteban-Cantillo et al. 2024)
        df = pd.DataFrame({
            'TX90p': 100 * pd.Series(cx, index=tx.index).groupby(tx.index.year).mean(),
            'TN90p': 100 * pd.Series(cn, index=tx.index).groupby(tx.index.year).mean(),
            'HWF': A.HWF, 'HWD': A.HWD, 'HWML': A.HWML,
            'HWF_tx': B.HWF, 'HWD_tx': B.HWD, 'HWML_tx': B.HWML,
            'TX35': pd.Series(tx.values >= UMBRAL_TX, index=tx.index).groupby(tx.index.year).sum(),
        })
        df.insert(0, 'modelo', m); df.insert(1, 'escenario', e); df.index.name = 'anio'
        filas.append(df.reset_index())
df_idx = pd.concat(filas, ignore_index=True)
ref_mask = df_idx.anio.between(int(REF[0]), int(REF[1]))
print('Filas:', len(df_idx), '| modelos:', MODELOS)
print('\nReferencia 1985–2014 (media por modelo; por construcción TX90p ≈ 10 %):')
print(df_idx[ref_mask & (df_idx.escenario == 'ssp245')].groupby('modelo')[['TX90p', 'TN90p', 'HWF', 'HWD', 'HWML',
'TX35']].mean().round(1))
```

```
Filas: 1440 | modelos: ['ACCESS-CM2', 'MPI-ESM1-2-HR', 'MRI-ESM2-0', 'NorESM2-MM', 'CMCC-ESM2']

Referencia 1985–2014 (media por modelo; por construcción TX90p ≈ 10 %):
              TX90p  TN90p  HWF  HWD  HWML  TX35
modelo
ACCESS-CM2      9.8    9.9  1.6  6.9   2.9  37.4
CMCC-ESM2       9.9   10.1  0.9  3.6   1.2  40.7
MPI-ESM1-2-HR  10.0    9.9  1.7  6.1   2.6  45.0
MRI-ESM2-0     10.0    9.9  0.9  3.9   2.0  40.8
NorESM2-MM     10.0   10.1  1.4  5.8   1.8  35.8
```

Lectura. La verificación pasa: el conteo propio coincide exactamente con `xclim.heat_wave_frequency`. En la referencia 1985–2014, los porcentajes de días sobre el P90 salen cerca de 10 % para cada modelo, como debe ocurrir por construcción, y las olas de calor con día y noche calurosos son raras: entre 0,9 y 1,7 eventos al año, de dos a tres días en el más largo.

```
# Índices con umbral absoluto anclados a la estación: se aplica a los días observados el cambio por cuantil que
# proyecta cada modelo entre 2019–2025 (ventana de la estación) y cada periodo futuro.
QS = np.linspace(0.005, 0.995, 100)
obs_tx, obs_tn = ideam_ok.tx.values, ideam_ok.tn.values
UMBRAL_TN_NOCHE = float(np.round(np.quantile(obs_tn, 0.90), 1))     # noche cálida: P90 observado de Tmín


def dias_por_anio(obs, ref, fut, umbral):
    dq = np.quantile(fut, QS) - np.quantile(ref, QS)
    p = stats.rankdata(obs) / (len(obs) + 1)
    return 365 * np.mean(obs + np.interp(p, QS, dq) >= umbral)


filas = []
for m in MODELOS:
    for e in ESC:
        tx = nex.sel(modelo=m, escenario=e).tx.to_series(); tn = nex.sel(modelo=m, escenario=e).tn.to_series()
        rtx, rtn = tx.loc[REF_OBS[0]:REF_OBS[1]].values, tn.loc[REF_OBS[0]:REF_OBS[1]].values
        for per, (a, b) in PERIODOS.items():
            filas.append((m, e, per,
                          dias_por_anio(obs_tx, rtx, tx.loc[a:b].values, UMBRAL_TX),
                          dias_por_anio(obs_tn, rtn, tn.loc[a:b].values, UMBRAL_TN_NOCHE),
                          float(tx.loc[a:b].mean() - tx.loc[REF_OBS[0]:REF_OBS[1]].mean()),
                          float(tn.loc[a:b].mean() - tn.loc[REF_OBS[0]:REF_OBS[1]].mean())))
df_abs = pd.DataFrame(filas, columns=['modelo', 'escenario', 'periodo', 'dias_tx35', 'noches_calidas', 'dTx',
'dTn'])
base_tx35 = 365 * np.mean(obs_tx >= UMBRAL_TX)
base_noche = 365 * np.mean(obs_tn >= UMBRAL_TN_NOCHE)
print(f'Anclaje en la estación (2019–2026): {base_tx35:.1f} días/año con Tmáx ≥ {UMBRAL_TX:.0f} °C y
```

```
{base_noche:.1f} noches/año con Tmín ≥ {UMBRAL_TN_NOCHE} °C')
(df_abs.groupby(['escenario', 'periodo'])
       .agg(tx35_mediana=('dias_tx35', 'median'), tx35_min=('dias_tx35', 'min'), tx35_max=('dias_tx35', 'max'),
            dTx_mediana=('dTx', 'median'), dTx_min=('dTx', 'min'), dTx_max=('dTx', 'max')).round(1))
```

```
Anclaje en la estación (2019–2026): 13.8 días/año con Tmáx ≥ 35 °C y 39.6 noches/año con Tmín ≥ 27.4 °C
```

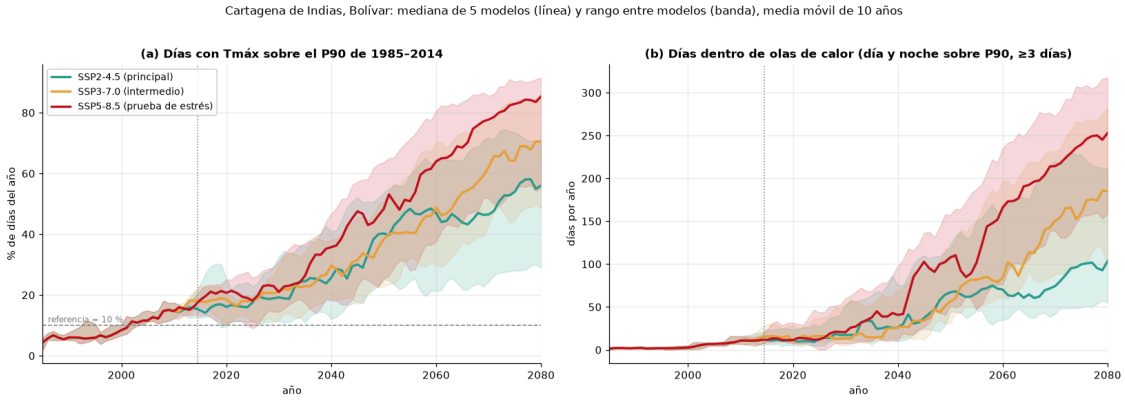
		tx35_mediana	tx35_min	tx35_max	dTx_mediana	dTx_min	dTx_max
escenario	periodo						
ssp245	2021–2040	25.4	7.0	28.4	0.3	−0.1	0.4
	2041–2060	51.1	19.5	58.9	1.0	0.3	1.1
	2061–2080	95.2	22.5	105.8	1.5	0.5	1.6
ssp370	2021–2040	28.4	13.8	28.4	0.4	0.3	0.5
	2041–2060	58.9	32.6	84.8	1.1	0.8	1.3
	2061–2080	153.0	67.2	175.3	1.9	1.3	2.2
ssp585	2021–2040	19.5	3.8	25.4	0.2	0.0	0.5
	2041–2060	76.5	17.2	84.8	1.1	0.6	1.3
	2061–2080	189.3	58.9	213.5	2.3	1.4	2.6

Lectura. Anclados en la estación, los 14 días al año con Tmáx ≥ 35 °C de 2019–2026 pasan, en la mediana de los modelos, a 25, 51 y 95 días en 2021–2040, 2041–2060 y 2061–2080 bajo SSP2-4.5; a 28, 59 y 153 bajo SSP3-7.0; y a 19, 76 y 189 bajo SSP5-8.5. Hasta 2040 los tres escenarios no se distinguen: con cinco modelos, el orden entre ellos en ese periodo no es significativo. El rango entre modelos es amplio (de 59 a 214 días en el último periodo de SSP5-8.5) porque el calentamiento de la Tmáx entre 2019–2025 y 2061–2080 va de 1,4 a 2,6 °C según el modelo en ese escenario, y los días sobre 35 °C crecen de forma no lineal con ese calentamiento. Las cifras de 2021–2040 son las menos seguras: el cambio se mide contra una ventana de solo siete años (2019–2025) y un solo año de El Niño puede mover la referencia.

Figura 2. Trayectorias de 1985 a 2080 por escenario: (a) porcentaje de días con Tmáx sobre el P90 de 1985–2014 y (b) días por año dentro de olas de calor. La línea punteada marca el inicio de las proyecciones (2015).

```
# Figura 2. Trayectorias 1985-2080: días calurosos y días en ola de calor
def suave(s, w=10):
    return s.rolling(w, center=True, min_periods=w // 2).mean()

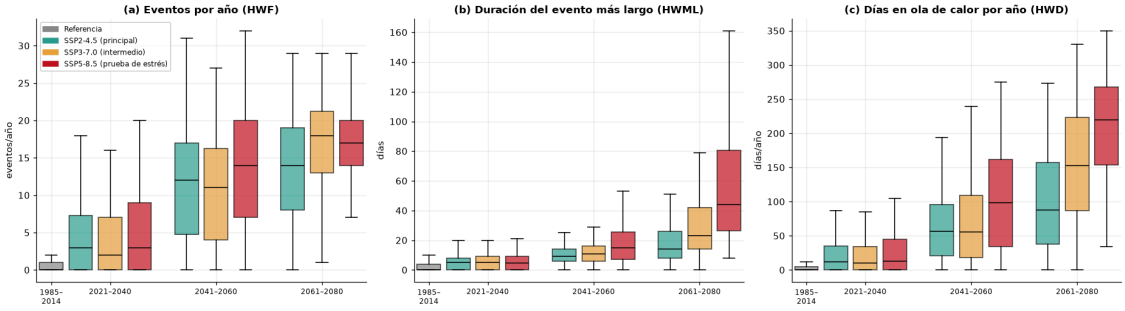
fig, ax = plt.subplots(1, 2, figsize=(15, 5.2), sharex=True)
for k, (col, tit, yl) in enumerate([('TX90p', '(a) Días con Tmáx sobre el P90 de 1985–2014', '% de días del año'),
                                    ('HWD', '(b) Días dentro de olas de calor (día y noche sobre P90, ≥3 días)',
'días por año')]):
    a = ax[k]
    for e in ESC:
        sub = df_idx[df_idx.escenario == e].pivot(index='anio', columns='modelo', values=col)
        sm = sub.apply(suave)
        a.fill_between(sm.index, sm.min(axis=1), sm.max(axis=1), color=ESC_COLOR[e], alpha=0.15)
        a.plot(sm.index, sm.median(axis=1), color=ESC_COLOR[e], lw=2.2, label=ESC_LABEL[e])
    a.axvline(2014.5, color='gray', ls=':', lw=1)                        # inicio de las proyecciones (2015)
    a.set_title(tit); a.set_xlabel('año'); a.set_ylabel(yl); a.set_xlim(1985, 2080)
ax[0].axhline(10, color='gray', ls='--', lw=1); ax[0].text(1986, 11, 'referencia = 10 %', fontsize=8, color='gray')
ax[0].legend(loc='upper left', fontsize=9)
fig.suptitle(f'{CIUDAD}: mediana de {len(MODELOS)} modelos (línea) y rango entre modelos (banda), media móvil de 10
años', y=1.02, fontsize=11)
plt.tight_layout(); plt.savefig(OUT_FINAL / 'fig2_trayectorias.png', bbox_inches='tight'); plt.show()
```



Lectura. Los días con Tmáx sobre el percentil 90 de 1985–2014 ya superan el 10 % de referencia desde cerca de 2005 y los escenarios se separan hacia 2040. En 2061–2080 la mediana de los modelos es de 50 %, 63 % y 78 % de los días bajo SSP2-4.5, SSP3-7.0 y SSP5-8.5; bajo SSP5-8.5 llega a cerca de tres cuartas partes hacia 2070. La Figura 2b muestra el efecto sobre las olas de calor: los días dentro de una ola pasan de unos 5 al año en la referencia a unos 110 en 2041–2060 bajo SSP5-8.5 (mediana de los modelos). La banda entre modelos es ancha: en SSP2-4.5, los días dentro de olas hacia 2061–2080 van de unos 44 a 205 al año según el modelo.

Figura 3. Distribución de los índices por periodo y escenario (años y modelos agrupados): (a) olas por año, (b) duración del evento más largo, (c) días dentro de olas.

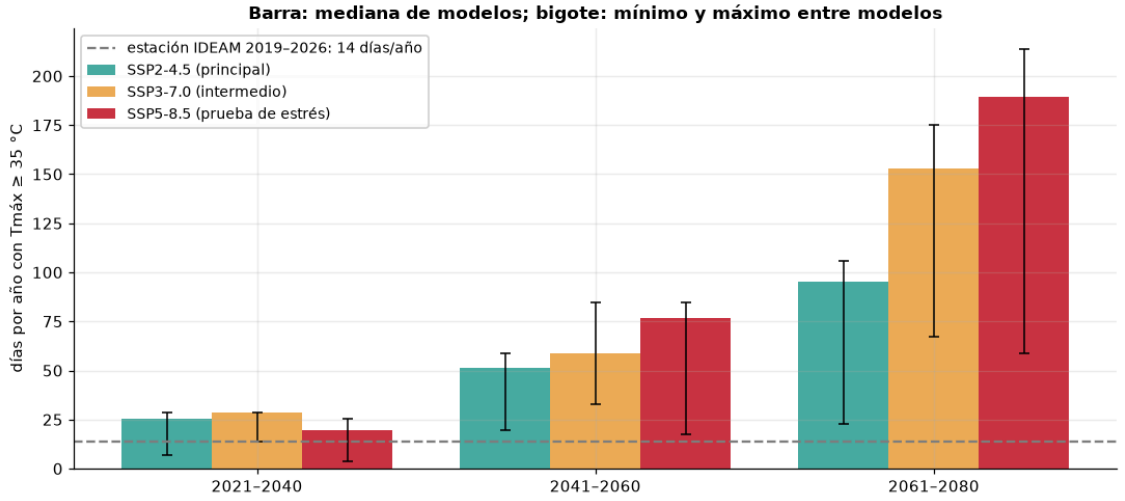
```
# Figura 3. Distribución por periodo (años × modelos)
fig, ax = plt.subplots(1, 3, figsize=(17, 4.8))
for k, (col, tit, yl) in enumerate([('HWF', '(a) Eventos por año (HWF)', 'eventos/año'),
                                    ('HWML', '(b) Duración del evento más largo (HWML)', 'días'),
                                    ('HWD', '(c) Días en ola de calor por año (HWD)', 'días/año')]):
    a = ax[k]
    datos, pos, cols = [], [], []
    x = 0
    r = df_idx[ref_mask & (df_idx.escenario == 'ssp245')][col]          # referencia (histórico, igual en los tres
escenarios)
    datos.append(r.values); pos.append(x); cols.append('#888888'); x += 1
    for per, (p0, p1) in PERIODOS.items():
        for e in ESC:
            v = df_idx[(df_idx.escenario == e) & df_idx.anio.between(int(p0), int(p1))][col]
            datos.append(v.values); pos.append(x); cols.append(ESC_COLOR[e]); x += 1
        x += 0.6
    bp = a.boxplot(datos, positions=pos, widths=0.8, patch_artist=True, showfliers=False,
medianprops=dict(color='black'))
    for b, c in zip(bp['boxes'], cols):
        b.set_facecolor(c); b.set_alpha(0.7)
    ancho_bloque = len(ESC) + 0.6
    a.set_xticks([0] + [1 + i * ancho_bloque + (len(ESC) - 1) / 2 for i in range(len(PERIODOS))])
    a.set_xticklabels(['1985–\n2014'] + list(PERIODOS.keys()), fontsize=8)
    a.set_title(tit); a.set_ylabel(yl)
ax[0].legend(handles=[Patch(color='#888888', label='Referencia')] + [Patch(color=ESC_COLOR[e], label=ESC_LABEL[e])
for e in ESC], fontsize=8, loc='upper left')
plt.tight_layout(); plt.savefig(OUT_FINAL / 'fig3_distribucion_periodos.png', bbox_inches='tight'); plt.show()
```



Lectura. La Figura 3 resume cada periodo con todos los años de todos los modelos. Las tres medidas aumentan, pero lo hacen de manera distinta: la frecuencia (HWF) se acerca a una meseta de 14 a 17 eventos al año, mientras que la duración del evento más largo (HWML) sigue creciendo: en 2061–2080 bajo SSP5-8.5 la mediana de los modelos es de 52 días (tabla de la sección 6.3). Esa combinación indica un cambio de régimen: cuando la mayoría de los días supera el umbral de la referencia, la «ola» deja de ser un episodio y se vuelve la condición típica. El percentil de 1985–2014 sigue siendo una referencia útil para medir el cambio, pero deja de describir un evento excepcional.

Figura 4. Días por año con Tmáx de 35 °C o más, anclados a la estación, por periodo y escenario.

```
# Figura 4. Días por año con Tmáx ≥ 35 °C anclados a la estación
fig, ax = plt.subplots(figsize=(10, 4.6))
w = 0.8 / len(ESC)
for i, e in enumerate(ESC):
    sub = df_abs[df_abs.escenario == e]
    med = sub.groupby('periodo').dias_tx35.median().reindex(PERIODOS.keys())
    lo = sub.groupby('periodo').dias_tx35.min().reindex(PERIODOS.keys()); hi =
sub.groupby('periodo').dias_tx35.max().reindex(PERIODOS.keys())
    x = np.arange(len(PERIODOS)) + (i - (len(ESC) - 1) / 2) * w
    ax.bar(x, med.values, w, color=ESC_COLOR[e], alpha=0.85, label=ESC_LABEL[e])
    ax.errorbar(x, med.values, yerr=[med.values - lo.values, hi.values - med.values], fmt='none', ecolor='black',
capsize=3, lw=1)
ax.axhline(base_tx35, color='gray', ls='--', label=f'estación IDEAM 2019–2026: {base_tx35:.0f} días/año')
ax.set_xticks(range(len(PERIODOS))); ax.set_xticklabels(PERIODOS.keys())
ax.set_ylabel('días por año con Tmáx ≥ 35 °C'); ax.set_title('Barra: mediana de modelos; bigote: mínimo y máximo
entre modelos')
ax.legend(loc='upper left', fontsize=9)
plt.tight_layout(); plt.savefig(OUT_FINAL / 'fig4_dias_tx35.png', bbox_inches='tight'); plt.show()
```

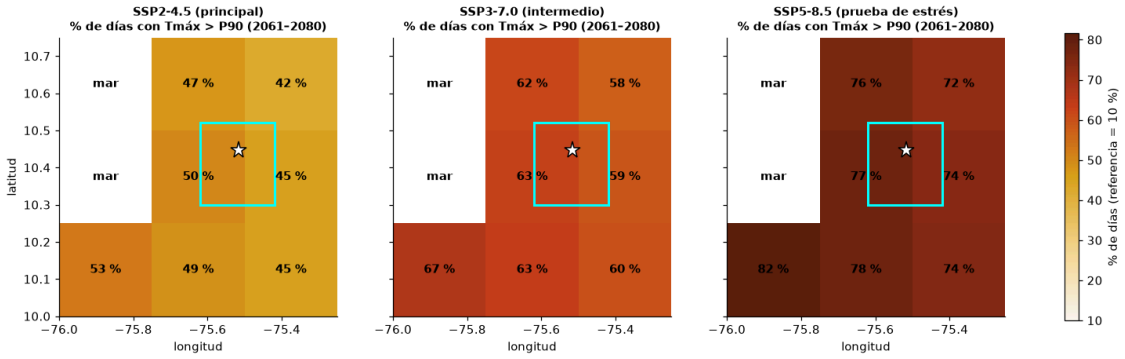


Lectura. La Figura 4 traduce el cambio a una cifra que se puede comunicar: días al año con Tmáx de 35 °C o más. La barra es la mediana de los modelos y el bigote, el rango entre ellos. La separación entre escenarios es pequeña hasta 2040 y grande

después: en 2061–2080 la mediana es de 95, 153 y 189 días bajo SSP2-4.5, SSP3-7.0 y SSP5-8.5, y los bigotes de los tres escenarios se solapan.

Figura 5. Porcentaje de días con Tmáx sobre el P90 en 2061–2080 en cada celda de 0,25° de la ventana de 3 por 3 celdas.

```
# Figura 5. Cambio espacial en la ventana 3x3 de celdas NEX-GDDP (0,25°)
cel = pd.read_csv(DATA / 'nex_gddp_celdas_3x3_anual.csv')
lats = np.sort(cel.lat.unique()); lons = np.sort(cel.lon.unique())
fig, ax = plt.subplots(1, len(ESC), figsize=(5.4 * len(ESC) + 1.5, 5.4), sharey=True)
vmax = 0
res = {}
for e in ESC:
    sub = cel[(cel.escenario == e) & cel.anio.between(2061, 2080)].groupby(['modelo', 'lat',
'lon']).p_tx90.mean().reset_index()
    M = sub.groupby(['lat', 'lon']).p_tx90.median().unstack('lon').loc[lats[::-1], lons] * 100
    res[e] = M; vmax = max(vmax, np.nanmax(M.values))
for a, e in zip(ax, ESC):
    M = res[e]
    im = a.imshow(M.values, extent=(lons.min() - 0.125, lons.max() + 0.125, lats.min() - 0.125, lats.max() + 0.125),
                  cmap=HEAT_CMAP, vmin=10, vmax=vmax)
    for i, la in enumerate(lats[::-1]):
        for j, lo in enumerate(lons):
            a.text(lo, la, 'mar' if np.isnan(M.values[i, j]) else f'{M.values[i, j]:.0f} %', ha='center',
va='center', fontsize=10, fontweight='bold')
    a.plot(ESTACION['lon'], ESTACION['lat'], marker='*', color='black', ms=14, mfc='white')
    a.add_patch(Rectangle((BBOX['west'], BBOX['south']), BBOX['east'] - BBOX['west'], BBOX['north'] - BBOX['south'],
fill=False, ec='cyan', lw=2))
    a.set_title(f'{ESC_LABEL[e]}\n% de días con Tmáx > P90 (2061–2080)', fontsize=10); a.grid(False)
    a.set_xlabel('longitud')
ax[0].set_ylabel('latitud')
plt.colorbar(im, ax=ax, shrink=0.8, label='% de días (referencia = 10 %)')
plt.savefig(OUT_FINAL / 'fig5_celdas_3x3.png', bbox_inches='tight'); plt.show()
print('Estrella: estación IDEAM. Recuadro cian: área de estudio. Cada celda mide ~25 km: no resuelve diferencias
dentro de la ciudad.')
```



Estrella: estación IDEAM. Recuadro cian: área de estudio. Cada celda mide ~25 km: no resuelve diferencias dentro de la ciudad.

Lectura. La ventana de 3 por 3 celdas muestra que el cambio proyectado es casi homogéneo en las celdas de tierra de la región: entre celdas la diferencia llega a unos 11 puntos porcentuales dentro de un mismo escenario, mientras que entre SSP2-4.5 y SSP5-8.5 es de unos 25. Las celdas de mar aparecen sin dato. Este resultado es el esperado de un modelo de 25 km, y por eso la Fase 2 aporta lo que esta fase no puede: qué barrios se calientan más.

6.3 Síntesis de la Fase 1

La tabla resume los índices por escenario y periodo. Cada celda muestra la mediana de los modelos y, entre corchetes, el mínimo y el máximo entre ellos.

```
filas = []
r = df_idx[ref_mask & (df_idx.escenario == 'ssp245')]
filas.append(['Referencia 1985–2014', '—',
```

```
            f'{r.TX90p.mean():.0f}', f'{r.HWF.mean():.1f}', f'{r.HWML.mean():.1f}', f'{r.HWD.mean():.1f}',
f'{base_tx35:.0f} (estación)'])
for e in ESC:
    for per, (a, b) in PERIODOS.items():
        s = df_idx[(df_idx.escenario == e) & df_idx.anio.between(int(a), int(b))]
        por_modelo = s.groupby('modelo')[['TX90p', 'HWF', 'HWML', 'HWD']].mean()
        ab = df_abs[(df_abs.escenario == e) & (df_abs.periodo == per)].dias_tx35
        filas.append([ESC_LABEL[e].split(' (')[0], per,
                      f'{por_modelo.TX90p.median():.0f} [{por_modelo.TX90p.min():.0f}–
{por_modelo.TX90p.max():.0f}]',
                      f'{por_modelo.HWF.median():.1f} [{por_modelo.HWF.min():.1f}–{por_modelo.HWF.max():.1f}]',
                      f'{por_modelo.HWML.median():.0f} [{por_modelo.HWML.min():.0f}–{por_modelo.HWML.max():.0f}]',
                      f'{por_modelo.HWD.median():.0f} [{por_modelo.HWD.min():.0f}–{por_modelo.HWD.max():.0f}]',
                      f'{ab.median():.0f} [{ab.min():.0f}–{ab.max():.0f}]'])
df_resumen = pd.DataFrame(filas, columns=['Escenario', 'Periodo', 'Días con Tmáx > P90 [% del año]', 'Olas de calor
[eventos/año]',
                                          'Evento más largo [días]', 'Días en ola [días/año]', 'Días con Tmáx ≥ 35
°C [días/año]'])
df_resumen
```

	Escenario	Periodo	Días con Tmáx > P90 [% del año]	Olas de calor [eventos/año]	Evento más largo [días]	Días en ola [días/año]	Días con Tmáx ≥ 35 °C [días/año]
0	Referencia 1985–2014	—	10	1.3	2.1	5.3	14 (estación)
1	SSP2-4.5	2021–2040	23 [12–31]	4.2 [2.2–8.9]	4 [3–12]	20 [11–50]	25 [7–28]
2	SSP2-4.5	2041–2060	38 [21–56]	10.8 [7.3–18.1]	9 [7–22]	56 [35–130]	51 [19–59]
3	SSP2-4.5	2061–2080	50 [25–70]	14.2 [8.3–19.6]	15 [10–44]	85 [44–205]	95 [23–106]
4	SSP3-7.0	2021–2040	21 [13–28]	2.9 [2.5–8.9]	4 [3–10]	13 [10–48]	28 [14–28]
5	SSP3-7.0	2041–2060	36 [25–54]	10.7 [7.9–16.5]	12 [6–25]	64 [42–130]	59 [33–85]
6	SSP3-7.0	2061–2080	63 [42–79]	16.4 [12.8–19.9]	26 [15–59]	144 [92–244]	153 [67–175]
7	SSP5-8.5	2021–2040	27 [15–32]	6.0 [2.1–9.6]	5 [2–12]	30 [9–59]	19 [4–25]
8	SSP5-8.5	2041–2060	52 [27–66]	15.1 [8.8–18.0]	19 [10–34]	109 [49–165]	76 [17–85]
9	SSP5-8.5	2061–2080	78 [51–88]	17.1 [14.7–18.6]	52 [26–109]	223 [137–287]	189 [59–214]

Lectura. Los resultados indican, con confianza media y sobre cinco modelos, que la frecuencia y la duración de las olas de calor en Cartagena aumentan en todos los escenarios y que el escenario importa mucho después de 2040: hacia 2061–2080 SSP3-7.0 queda entre SSP2-4.5 y SSP5-8.5 en todos los índices, y antes de 2040 el orden entre los tres no es significativo. La confianza es media y no alta porque la muestra es de cinco modelos con un solo miembro cada uno y porque la referencia observada de la estación es corta. Estas cifras describen el clima de fondo de la celda de 25 km del aeropuerto; no dicen nada sobre las diferencias entre barrios, tema de la siguiente fase.

7. FASE 2: ISLA DE CALOR URBANA

Esta fase usa el compuesto de LST descrito en la sección 4.5. Cada píxel resume en promedio 168 fechas válidas entre 228 escenas Landsat 8 y 9 (2018–2024), y la cobertura del suelo viene de WorldCover. El análisis responde dos preguntas: cuánto más caliente es la superficie construida que la no construida, y cuánto de esa diferencia explican la fracción construida y el arbolado.

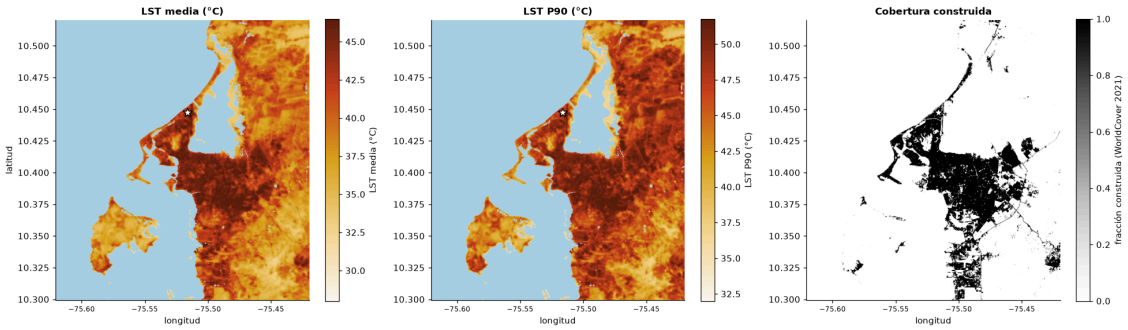
```
lst = rasterio.open(DATA / 'lst_cartagena_2018_2024.tif'); L = lst.read()
wc = rasterio.open(DATA / 'worldcover_cartagena_2021.tif'); W = wc.read()
LST_MEDIA, LST_P90, N_FECHAS, LST_MEDIA_SECA, LST_P90_SECA = L[0], L[1], L[2], L[3], L[4]
CONSTRUIDO, ARBOLES, AGUA, MANGLAR, OTRA_VEG = W
tierra = AGUA < 0.5
```

```
b = transform_bounds(lst.crs, 'EPSG:4326', *lst.bounds)
EXT = (b[0], b[2], b[1], b[3])           # oeste, este, sur, norte (grados)
print(f'Compuesto Landsat 8/9, 2018–2024: {lst.width}×{lst.height} píxeles de 30 m, {int(np.nanmedian(N_FECHAS))}
fechas válidas por píxel (mediana)')
print(f'Tierra: {100*tierra.mean():.0f} % del área; construido {100*CONSTRUIDO[tierra].mean():.0f} %, árboles
{100*ARBOLES[tierra].mean():.0f} % de la tierra')
```

```
Compuesto Landsat 8/9, 2018–2024: 732×813 píxeles de 30 m, 168 fechas válidas por píxel (mediana)
Tierra: 47 % del área; construido 25 %, árboles 30 % de la tierra
```

Figura 6. Temperatura de superficie (LST) media y P90 del compuesto Landsat 8 y 9 de 2018 a 2024, y fracción construida de WorldCover 2021.

```
# Figura 6. LST media y P90, con la fracción construida
lo, hi = np.nanpercentile(LST_MEDIA[tierra], [2, 98]), np.nanpercentile(LST_P90[tierra], [2, 98])
fig, ax = plt.subplots(1, 3, figsize=(18, 6.2))
for a, M, t, vmin, vmax in [(ax[0], LST_MEDIA, 'LST media (°C)', 28, lo[1]), (ax[1], LST_P90, 'LST P90 (°C)', 32,
hi[1])]:
    im = a.imshow(np.where(AGUA > 0.5, np.nan, M), extent=EXT, cmap=HEAT_CMAP, vmin=vmin, vmax=vmax)
    a.imshow(np.where(AGUA > 0.5, 1.0, np.nan), extent=EXT, cmap='Blues', vmin=0, vmax=2, alpha=0.6)
    plt.colorbar(im, ax=a, shrink=0.75, label=t); a.set_title(t); a.grid(False)
im = ax[2].imshow(CONSTRUIDO, extent=EXT, cmap='Greys', vmin=0, vmax=1)
plt.colorbar(im, ax=ax[2], shrink=0.75, label='fracción construida (WorldCover 2021)'); ax[2].set_title('Cobertura
construida'); ax[2].grid(False)
for a in ax:
    a.plot(ESTACION['lon'], ESTACION['lat'], marker='*', color='black', ms=11, mfc='white');
a.set_xlabel('longitud')
    a.set_xticks(np.arange(-75.6, -75.41, 0.05)); a.tick_params(axis='x', labelsize=8)
ax[0].set_ylabel('latitud')
plt.tight_layout(); plt.savefig(OUT_FINAL / 'fig6_lst_media_p90.png', bbox_inches='tight'); plt.show()
```



Lectura. El núcleo construido de la ciudad, entre la bahía y la Ciénaga de la Virgen, concentra las superficies más calientes, tanto en la media como en el P90, y el patrón coincide con la mancha construida del tercer panel. El agua y el manglar quedan como las superficies más frías. El P90 acentúa el contraste: en los días más calurosos de la mañana, las superficies construidas del centro-sur superan los 50 °C.

```
# LST por clase de cobertura (píxeles de 30 m dominados por cada clase)
clases = {
    'Construido (> 50 %)': (CONSTRUIDO > 0.5) & tierra,
    'Árboles (> 50 %)': (ARBOLES > 0.5) & tierra,
    'Manglar (> 50 %)': (MANGLAR > 0.5) & tierra,
    'Otra vegetación (> 50 %)': (OTRA_VEG > 0.5) & tierra,
    'Agua (> 90 %)': (AGUA > 0.9),
}
filas = []
for k, m in clases.items():
    v, p = LST_MEDIA[m], LST_P90[m]
    filas.append((k, int(m.sum()), np.nanmedian(v), np.nanpercentile(v, 25), np.nanpercentile(v, 75),
np.nanmedian(p), np.nanmedian(LST_P90_SECA[m])))
df_clases = pd.DataFrame(filas, columns=['Clase', 'Píxeles (30 m)', 'LST media: mediana', 'LST media: P25', 'LST
media: P75', 'LST P90: mediana', 'LST P90 época seca: mediana'])
df_clases.iloc[:, 2:] = df_clases.iloc[:, 2:].astype('float64').round(1)
ref_m = (CONSTRUIDO < 0.02) & ((ARBOLES + OTRA_VEG + MANGLAR) > 0.5) & tierra
```

```
ref_veg, ref_veg90 = np.nanmedian(LST_MEDIA[ref_m]), np.nanmedian(LST_P90[ref_m])
uhi = df_clases.set_index('Clase').loc['Construido (> 50 %)', 'LST media: mediana'] - ref_veg
uhi90 = df_clases.set_index('Clase').loc['Construido (> 50 %)', 'LST P90: mediana'] - ref_veg90
print(f'Referencia no urbana (vegetación, < 2 % construido): LST media {ref_veg:.1f} °C, P90 {ref_veg90:.1f} °C')
print(f'Intensidad de isla de calor superficial (construido menos referencia): {uhi:+.1f} °C en la media y {uhi90:
+.1f} °C en el P90')
df_clases
```

```
Referencia no urbana (vegetación, < 2 % construido): LST media 38.4 °C, P90 45.4 °C
Intensidad de isla de calor superficial (construido menos referencia): +6.1 °C en la media y +3.9 °C en el P90
```

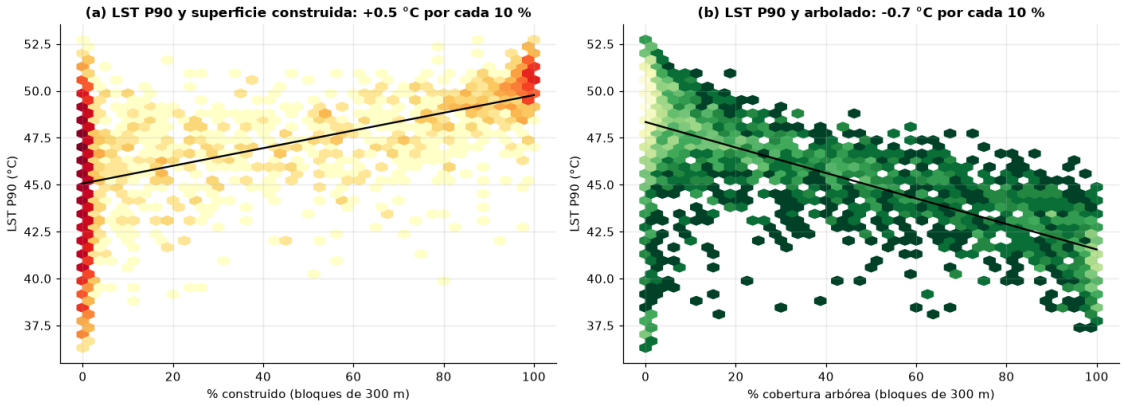
	Clase	Píxeles (30 m)	LST media: mediana	LST media: P25	LST media: P75	LST P90: mediana	LST P90 época seca: mediana
0	Construido (> 50 %)	70443	44.500000	42.700001	45.599998	49.299999	47.099998
1	Árboles (> 50 %)	83127	36.900002	35.700001	38.200001	43.200001	41.900002
2	Manglar (> 50 %)	12288	34.599998	33.599998	36.000000	37.900002	36.400002
3	Otra vegetación (> 50 %)	106965	39.799999	38.700001	41.099998	47.400002	46.099998
4	Agua (> 90 %)	312417	30.200001	29.799999	31.100000	33.400002	31.500000

Lectura. La superficie construida tiene una LST mediana de 44,5 °C frente a 36,9 °C de las zonas arboladas y 34,6 °C del manglar. Contra una referencia no urbana (vegetación con menos de 2 % de construcción), la intensidad de la isla de calor superficial es de +6,1 °C en la media y +3,9 °C en el P90. El contraste es menor en el P90 porque en los días más calurosos también se calientan las superficies no urbanas secas. Estas diferencias son de **temperatura de superficie** a las 10:30 de la mañana, no de temperatura del aire, y no se deben leer como tales.

Figura 7. Relación entre la LST P90 y (a) la fracción construida y (b) la cobertura arbórea, en bloques de 300 m.

```
# Cuánto sube la LST por cada 10 % más de superficie construida, a 300 m (promedio de bloques de 10x10 píxeles)
def bloques(A, f=10):
    h, w = (A.shape[0] // f) * f, (A.shape[1] // f) * f
    return A[:h, :w].reshape(h // f, f, w // f, f)

cb = np.nanmean(bloques(CONSTRUIDO), axis=(1, 3)); l90 = np.nanmean(bloques(LST_P90), axis=(1, 3))
lm = np.nanmean(bloques(LST_MEDIA), axis=(1, 3)); ag = np.nanmean(bloques(AGUA), axis=(1, 3)); ar =
np.nanmean(bloques(ARBOLES), axis=(1, 3))
ok = (ag < 0.2) & np.isfinite(l90)
reg_m = stats.linregress(cb[ok], lm[ok]); reg_90 = stats.linregress(cb[ok], l90[ok]); reg_ar =
stats.linregress(ar[ok], l90[ok])
xx = np.array([0, 100])
fig, ax = plt.subplots(1, 2, figsize=(13, 4.8))
ax[0].hexbin(cb[ok] * 100, l90[ok], gridsize=40, cmap='YlOrRd', mincnt=1, bins='log')
ax[0].plot(xx, reg_90.intercept + reg_90.slope * xx / 100, 'k', lw=1.5)
ax[0].set_xlabel('% construido (bloques de 300 m)'); ax[0].set_ylabel('LST P90 (°C)'); ax[0].set_title(f'(a) LST P90
y superficie construida: {reg_90.slope/10:+.1f} °C por cada 10 %')
ax[1].hexbin(ar[ok] * 100, l90[ok], gridsize=40, cmap='YlGn_r', mincnt=1, bins='log')
ax[1].plot(xx, reg_ar.intercept + reg_ar.slope * xx / 100, 'k', lw=1.5)
ax[1].set_xlabel('% cobertura arbórea (bloques de 300 m)'); ax[1].set_ylabel('LST P90 (°C)'); ax[1].set_title(f'(b)
LST P90 y arbolado: {reg_ar.slope/10:+.1f} °C por cada 10 %')
plt.tight_layout(); plt.savefig(OUT_FINAL / 'fig7_lst_vs_cobertura.png', bbox_inches='tight'); plt.show()
print(f'Media: {reg_m.slope/10:+.2f} °C por cada 10 % construido (r² = {reg_m.rvalue**2:.2f}); P90:
{reg_90.slope/10:+.2f} °C (r² = {reg_90.rvalue**2:.2f}); arbolado P90: {reg_ar.slope/10:+.2f} °C por cada 10 % (r² =
{reg_ar.rvalue**2:.2f})')
```




```
Media: +0.71 °C por cada 10 % construido (r² = 0.65); P90: +0.47 °C (r² = 0.28); arbolado P90: -0.68 °C por cada 10
% (r² = 0.51)
```

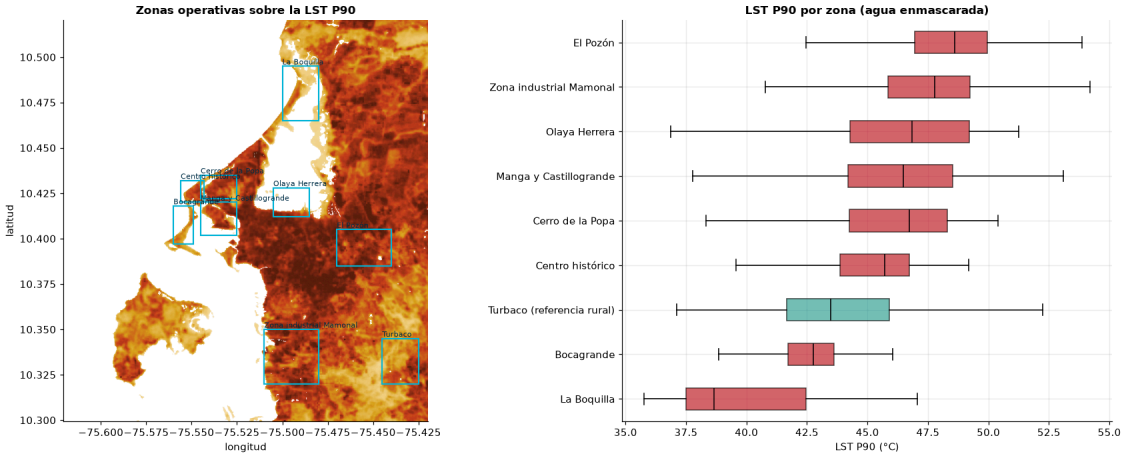
Lectura. A escala de 300 m, cada 10 % más de superficie construida se asocia con +0,71 °C en la LST media ($r^2 = 0,65$) y +0,47 °C en el P90 ($r^2 = 0,28$); cada 10 % más de cobertura arbórea, con −0,68 °C en el P90 ($r^2 = 0,51$). El arbolado explica mejor el P90 que la superficie construida. Es una asociación espacial y no prueba por sí sola que sembrar árboles reduzca la LST en esa cantidad, pero señala el arbolado como la palanca de reducción más clara, en línea con lo que reporta la literatura sobre bosques urbanos en ciudades tropicales (Giraldo-Charria et al., 2025).

Figura 8. Zonas operativas sobre la LST P90 y distribución de la LST P90 por zona.

```
# Zonas urbanas operativas: rectángulos aproximados (no son límites oficiales), con el agua enmascarada
ZONAS = {
    'Centro histórico':       (-75.556, 10.420, -75.543, 10.432),
    'Bocagrande':             (-75.560, 10.397, -75.549, 10.418),
    'Manga y Castillogrande': (-75.545, 10.402, -75.525, 10.420),
    'Cerro de la Popa':       (-75.545, 10.422, -75.525, 10.435),
    'Olaya Herrera':          (-75.505, 10.412, -75.485, 10.428),
    'El Pozón':               (-75.470, 10.385, -75.440, 10.405),
    'Zona industrial Mamonal': (-75.510, 10.320, -75.480, 10.350),
    'La Boquilla':            (-75.500, 10.465, -75.480, 10.495),
    'Turbaco (referencia rural)': (-75.445, 10.320, -75.425, 10.345),
}
lon_c = np.linspace(EXT[0], EXT[1], lst.width); lat_c = np.linspace(EXT[3], EXT[2], lst.height)
LON, LAT = np.meshgrid(lon_c, lat_c)
filas, zdata = [], {}
for z, (w, s, e, n) in ZONAS.items():
    m = (LON >= w) & (LON <= e) & (LAT >= s) & (LAT <= n) & tierra & np.isfinite(LST_MEDIA)
    zdata[z] = m
    filas.append((z, int(m.sum()), 100 * CONSTRUIDO[m].mean(), 100 * ARBOLES[m].mean(), np.nanmean(LST_MEDIA[m]),
np.nanmean(LST_P90[m])))
df_zonal = pd.DataFrame(filas, columns=['Zona', 'Píxeles de tierra', '% construido', '% árboles', 'LST media [°C]',
'LST P90 [°C]'])
rural = df_zonal.set_index('Zona').loc['Turbaco (referencia rural)']
df_zonal['Δ media vs rural [°C]'] = df_zonal['LST media [°C]'] - rural['LST media [°C]']
df_zonal['Δ P90 vs rural [°C]'] = df_zonal['LST P90 [°C]'] - rural['LST P90 [°C]']
df_zonal.iloc[:, 2:] = df_zonal.iloc[:, 2:].astype('float64').round(1)

fig, ax = plt.subplots(1, 2, figsize=(16, 6.4), gridspec_kw={'width_ratios': [1.1, 1]})
im = ax[0].imshow(np.where(AGUA > 0.5, np.nan, LST_P90), extent=EXT, cmap=HEAT_CMAP, vmin=32, vmax=hi[1])
for z, (w, s, e, n) in ZONAS.items():
    ax[0].add_patch(Rectangle((w, s), e - w, n - s, fill=False, ec='#00b4d8', lw=1.4)); ax[0].text(w, n, z.split('
(')[0], fontsize=7, color='#003049', va='bottom')
ax[0].set_title('Zonas operativas sobre la LST P90'); ax[0].grid(False); ax[0].set_xlabel('longitud');
ax[0].set_ylabel('latitud')
orden = df_zonal.sort_values('LST P90 [°C]').Zona.tolist()
datos = [LST_P90[zdata[z]] for z in orden]
bp = ax[1].boxplot(datos, vert=False, patch_artist=True, showfliers=False, medianprops=dict(color='black'))
```

```
ax[1].set_yticklabels(orden)
for bx, z in zip(bp['boxes'], orden):
    bx.set_facecolor('#c1121f' if 'rural' not in z else '#2a9d8f'); bx.set_alpha(0.65)
ax[1].set_xlabel('LST P90 (°C)'); ax[1].set_title('LST P90 por zona (agua enmascarada)')
plt.tight_layout(); plt.savefig(OUT_FINAL / 'fig8_uhi_por_zona.png', bbox_inches='tight'); plt.show()
df_zonal
```



	Zona	Píxeles de tierra	% construido	% árboles	LST media [°C]	LST P90 [°C]	Δ media vs rural [°C]	Δ P90 vs rural [°C]
0	Centro histórico	1270	84.900002	0.9	41.299999	45.299999	3.3	1.4
1	Bocagrande	1049	87.599998	0.5	38.900002	42.599998	0.8	−1.3
2	Manga y Castillo-grande	3013	79.000000	7.1	42.000000	46.099998	4.0	2.2
3	Cerro de la Popa	3104	71.300003	13.2	41.799999	46.000000	3.7	2.2
4	Olaya Her-rera	1043	66.500000	3.2	41.799999	46.099998	3.8	2.2
5	El Pozón	7982	37.799999	8.1	42.500000	48.500000	4.5	4.6
6	Zona in-dustrial Ma-monal	11505	43.400002	16.9	42.000000	47.400002	4.0	3.6
7	La Boquilla	3177	28.200001	2.1	36.599998	40.000000	−1.5	−3.8
8	Turbaco (referencia rural)	6700	12.500000	55.0	38.000000	43.799999	0.0	0.0

Lectura. Las zonas más calientes en el P90 no son las del centro turístico, sino las de la periferia construida: El Pozón (48,5 °C, +4,6 °C sobre la referencia rural), la zona industrial de Mamonal (47,4 °C) y Olaya Herrera, Manga y el Cerro de la Popa (46 °C). El centro histórico queda en 45,3 °C, y Bocagrande y La Boquilla, junto al mar, son más frescas que la referencia rural (42,6 y 40,0 °C). Dos advertencias: los rectángulos son aproximados y no son los límites oficiales de los barrios, y las zonas costeras parecen frescas en parte por la brisa y la mezcla con el agua, que la máscara solo corrige en parte.

Figura 9. Mapa interactivo de la LST P90 con hospitales y clínicas de OpenStreetMap.

```
# Figura 9. Mapa interactivo: LST P90 e infraestructura de salud (OpenStreetMap)
from rasterio.transform import from_bounds
```

```
def a_wgs84(A, res=0.0003):
    w, e, s, n = EXT
    ancho, alto = int((e - w) / res), int((n - s) / res)
    dst = np.full((alto, ancho), np.nan, dtype='float32')
    reproject(A.astype('float32'), dst, src_transform=lst.transform, src_crs=lst.crs,
              dst_transform=from_bounds(w, s, e, n, ancho, alto), dst_crs='EPSG:4326',
              resampling=Resampling.nearest, src_nodata=np.nan, dst_nodata=np.nan)
    return dst

p90_wgs = a_wgs84(np.where(AGUA > 0.5, np.nan, LST_P90))
norm = np.clip((p90_wgs - 32) / (hi[1] - 32), 0, 1)
rgba = HEAT_CMAP(norm); rgba[np.isnan(p90_wgs)] = 0
osm = pd.read_csv(DATA / 'osm_infraestructura_cartagena.csv')
m = folium.Map(location=[10.41, -75.52], zoom_start=12, tiles='CartoDB positron')
folium.raster_layers.ImageOverlay(image=rgba, bounds=[[EXT[2], EXT[0]], [EXT[3], EXT[1]]], opacity=0.75, name='LST
P90 (°C)').add_to(m)
gsal = folium.FeatureGroup(name='Hospitales y clínicas (OSM)', show=True)
for _, r in osm[osm.tipo.isin(['hospital', 'clinic'])].iterrows():
    folium.CircleMarker([r.lat, r.lon], radius=3, color='#003049', fill=True, fill_opacity=0.9,
                        tooltip=f"{r.tipo}: {r.nombre if isinstance(r.nombre, str) else 's/n'}").add_to(gsal)
gsal.add_to(m)
folium.Marker([ESTACION['lat'], ESTACION['lon']], tooltip=ESTACION['nombre'], icon=folium.Icon(color='black',
icon='cloud')).add_to(m)
folium.LayerControl().add_to(m)
m
```

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

Lectura. El mapa permite acercarse a cada barrio y ver dónde caen los hospitales y las clínicas de OpenStreetMap respecto de las superficies más calientes. Las capas se encienden y apagan desde el control de la esquina.

```
# Época seca (dic-mar) frente al compuesto de todo el año
d_seca = (LST_MEDIA_SECA - LST_MEDIA)[tierra]
print(f'LST media en época seca menos todo el año (tierra): mediana {np.nanmedian(d_seca):+.1f} °C; construido
{np.nanmedian((LST_MEDIA_SECA - LST_MEDIA)[clases["Construido (> 50 %)"]]):+.1f} °C')
print(f'Fechas válidas por píxel: {int(np.nanmedian(N_FECHAS))} (todo el año) y {int(np.nanmedian(L[5]))} (época
seca)')
```

```
LST media en época seca menos todo el año (tierra): mediana -0.2 °C; construido -0.9 °C
Fechas válidas por píxel: 168 (todo el año) y 95 (época seca)
```

Lectura. La época seca (diciembre a marzo) tiene una LST ligeramente menor que el compuesto de todo el año (-0,2 °C en la mediana de la tierra y −0,9 °C en las superficies construidas), y menos fechas válidas por píxel (95 frente a 168). Esto se debe en parte a los vientos alisios de esos meses y no significa que la época seca sea más segura frente al calor; la Fase 3 muestra que la humedad hace que los meses de mayo a septiembre sean los más exigentes.

8. FASE 3: ESTRÉS TÉRMICO

La temperatura sola subestima la carga que el calor impone cuando la humedad es alta, como en el Caribe. Con la temperatura y la humedad relativa de la estación IDEAM (sección 4.6) se calcula el Humidex máximo diario en los 867 días con datos suficientes entre 2019 y 2026.

Figura 10. Humidex máximo diario en la estación IDEAM: (a) ciclo anual y (b) distribución frente a ERA5-Land.

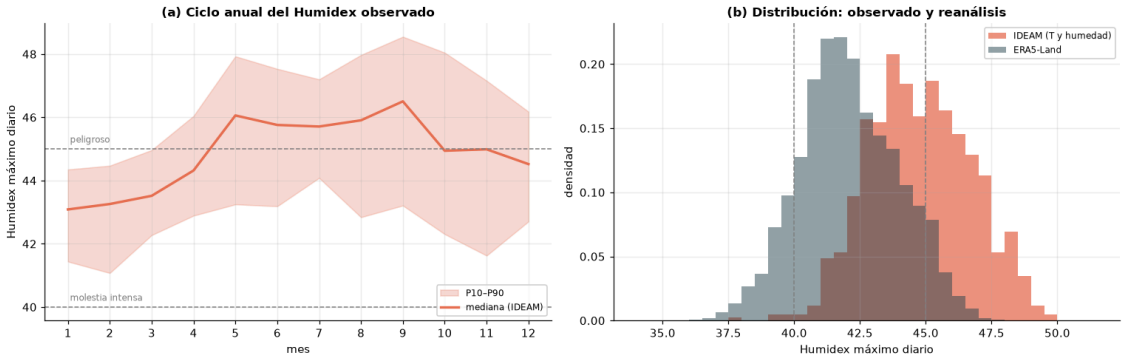
```
def humidex(t, rh):
    """Humidex (Masterton y Richardson, 1979) con temperatura (°C) y humedad relativa (%); el punto de rocío sale de
Magnus."""
    a, b = 17.62, 243.12
    g = np.log(np.clip(rh, 1, 100) / 100) + a * t / (b + t)
    td = b * g / (a - g)
    e = 6.11 * np.exp(5417.7530 * (1 / 273.16 - 1 / (td + 273.15)))
    return t + 0.5555 * (e - 10.0)

hx = ideam_hx.dropna(subset=['hx_max']).copy()
CAT = [(0, 30, 'Sin molestia'), (30, 40, 'Algo de molestia'), (40, 45, 'Molestia intensa'), (45, 200, 'Peligroso')]
fr = {n: 100 * ((hx.hx_max >= a) & (hx.hx_max < b)).mean() for a, b, n in CAT}
```

```
hx['mes'] = hx.index.month
mensual = hx.groupby('mes').hx_max.agg(['median', lambda s: s.quantile(.1), lambda s: s.quantile(.9)])
mensual.columns = ['mediana', 'p10', 'p90']
print(f'Días con Humidex diario máximo observado: {len(hx)} (2019–2026); mediana {hx.hx_max.median():.1f}')
print({k: f'{v:.0f} %' for k, v in fr.items()})
print('Mediana mensual del Humidex máximo:', {int(k): round(v, 1) for k, v in mensual['mediana'].items()})
e19 = era5.loc['2019':'2024', 'hx_max']
print(f'ERA5-Land 2019–2024: mediana {e19.median():.1f}; diferencia de medianas con la estación {e19.median() -
hx.hx_max.median():+.1f}')

fig, ax = plt.subplots(1, 2, figsize=(14, 4.6))
a = ax[0]
a.fill_between(mensual.index, mensual.p10, mensual.p90, color='#e76f51', alpha=0.25, label='P10–P90')
a.plot(mensual.index, mensual['mediana'], color='#e76f51', lw=2.2, label='mediana (IDEAM)')
for y, t in [(40, 'molestia intensa'), (45, 'peligroso')]:
    a.axhline(y, color='gray', ls='--', lw=1); a.text(1.05, y + 0.2, t, fontsize=8, color='gray')
a.set_xticks(range(1, 13)); a.set_xlabel('mes'); a.set_ylabel('Humidex máximo diario'); a.set_title('(a) Ciclo anual
del Humidex observado'); a.legend(loc='lower right', fontsize=8)
a = ax[1]
a.hist(hx.hx_max, bins=np.arange(34, 52, 0.5), color='#e76f51', alpha=0.75, density=True, label='IDEAM (T y
humedad)')
a.hist(e19, bins=np.arange(34, 52, 0.5), color='#264653', alpha=0.5, density=True, label='ERA5-Land')
for xx in (40, 45):
    a.axvline(xx, color='gray', ls='--', lw=1)
a.set_xlabel('Humidex máximo diario'); a.set_ylabel('densidad'); a.set_title('(b) Distribución: observado y
reanálisis'); a.legend(fontsize=8)
plt.tight_layout(); plt.savefig(OUT_FINAL / 'fig10_humidex.png', bbox_inches='tight'); plt.show()
```

```
Días con Humidex diario máximo observado: 867 (2019–2026); mediana 44.7
{'Sin molestia': '0 %', 'Algo de molestia': '1 %', 'Molestia intensa': '54 %', 'Peligroso': '46 %'}
Mediana mensual del Humidex máximo: {1: 43.1, 2: 43.3, 3: 43.5, 4: 44.3, 5: 46.1, 6: 45.8, 7: 45.7, 8: 45.9, 9:
46.5, 10: 44.9, 11: 45.0, 12: 44.5}
ERA5-Land 2019–2024: mediana 42.0; diferencia de medianas con la estación -2.7
```



Lectura. El Humidex máximo diario observado tiene una mediana de 44,7 y en 46 % de los días entra en la categoría «peligroso» (45 o más) de Environment Canada; el resto está casi todo en «molestia intensa» (40 a 45) y solo 1 % de los días queda por debajo de 40. El ciclo anual no sigue a la temperatura: los meses de mayo a septiembre tienen la mediana más alta (45,7 a 46,5) y enero a marzo la más baja (43,1 a 43,5). ERA5-Land, por la subestimación de temperatura de la celda costera, desplaza la mediana 2,7 unidades hacia abajo (42,0 frente a 44,7). Dos advertencias: las categorías del Humidex se diseñaron para el verano canadiense y su uso en un clima tropical, donde la población está aclimatada, es una convención, no un umbral médico; y la estación es una sola.

```
# Sensibilidad (no es proyección): el Humidex observado si la temperatura sube lo que proyectan los modelos
# y la humedad relativa se mantiene. Es un supuesto simple; no incluye cambios en la humedad ni en el viento.
dT = df_abs.groupby(['escenario', 'periodo']).dTx.median()
filas = [('Observado 2019–2026', '–', 0.0, f'{100*(hx.hx_max >= 40).mean():.0f}', f'{100*(hx.hx_max >=
45).mean():.0f}', f'{hx.hx_max.median():.1f}')]
for e in ESC:
    for per in PERIODOS:
        d = float(dT.loc[(e, per)])
```

```
        h2 = humidex(hx.t_hxmax + d, hx.rh_hxmax)
        filas.append((ESC_LABEL[e].split(' (')[0], per, d, f'{100*(h2 >= 40).mean():.0f}', f'{100*(h2 >=
45).mean():.0f}', f'{h2.median():.1f}'))
df_hx = pd.DataFrame(filas, columns=['Escenario', 'Periodo', 'ΔTmáx aplicado [°C]', '% días con Humidex ≥ 40', '%
días con Humidex ≥ 45', 'Mediana del Humidex máximo'])
df_hx['ΔTmáx aplicado [°C]'] = df_hx['ΔTmáx aplicado [°C]'].round(1)
df_hx
```

	Escenario	Periodo	ΔTmáx aplicado [°C]	% días con Humidex ≥ 40	% días con Humidex ≥ 45	Mediana del Humidex máximo
0	Observado 2019–2026	—	0.0	99	46	44.7
1	SSP2-4.5	2021–2040	0.3	100	56	45.3
2	SSP2-4.5	2041–2060	1.0	100	83	46.8
3	SSP2-4.5	2061–2080	1.5	100	94	47.8
4	SSP3-7.0	2021–2040	0.4	100	59	45.4
5	SSP3-7.0	2041–2060	1.1	100	87	47.1
6	SSP3-7.0	2061–2080	1.9	100	98	48.9
7	SSP5-8.5	2021–2040	0.2	100	51	45.0
8	SSP5-8.5	2041–2060	1.1	100	87	47.1
9	SSP5-8.5	2061–2080	2.3	100	99	49.6

Lectura. Esta tabla es una **sensibilidad**, no una proyección: suma a la temperatura observada el calentamiento de la Tmáx de los modelos y mantiene constante la humedad relativa. Con +1,0 °C (SSP2-4.5 hacia 2041–2060) los días en categoría «peligroso» pasan de 46 % a 83 %, y con +2,3 °C (SSP5-8.5 hacia 2061–2080) a 99 %. El supuesto de humedad constante es simple; cambios en la humedad o en el viento modificarían estas cifras. La conclusión robusta es de dirección y orden de magnitud: una parte pequeña del calentamiento basta para que casi todos los días queden en la categoría más alta.

9. FASE 4: EXPOSICIÓN

La exposición es la población que vive en cada clase de amenaza. Se alinean la LST P90, la cobertura del suelo y la población de WorldPop (100 m) en una misma rejilla y se clasifica la amenaza con intervalos fijos de la LST P90, como hace CLIMAAX con la LST pero con cortes ajustados al rango local. La población de WorldPop es una estimación modelada, no un censo, y no incluye la edad.

Figura 11. Clase de amenaza por LST P90 en las celdas pobladas y población por clase.

```
# Alinear todas las capas a la rejilla de WorldPop (~92 m) con promedio de los píxeles de 30 m
pop_ds = rasterio.open(DATA / 'worldpop_cartagena_2020.tif'); POP = pop_ds.read(1)
def a_pop(A):
    dst = np.full(POP.shape, np.nan, dtype='float32')
    reproject(A.astype('float32'), dst, src_transform=lst.transform, src_crs=lst.crs,
              dst_transform=pop_ds.transform, dst_crs=pop_ds.crs, resampling=Resampling.average,
              src_nodata=np.nan, dst_nodata=np.nan)
    return dst
P90p, MEDp, CONp, ARBp, AGUp = a_pop(LST_P90), a_pop(LST_MEDIA), a_pop(CONSTRUIDO), a_pop(ARBOLES), a_pop(AGUA)
pob = (POP > 0) & np.isfinite(P90p) & (AGUp < 0.5)
print(f'Población en el AOI (WorldPop 2020): {POP.sum():,.0f}  |  en celdas con LST válida y sin agua dominante:
{POP[pob].sum():,.0f} ({100*POP[pob].sum()/POP.sum():.1f} %)')
o = np.argsort(P90p[pob]); cw = np.cumsum(POP[pob][o]) / POP[pob].sum()
pq = {q: float(P90p[pob][o][np.searchsorted(cw, q)]) for q in (0.05, 0.25, 0.5, 0.75, 0.95, 0.99)}
print(f'Celdas pobladas: {pob.sum():,}')
print('LST P90 ponderada por población, percentiles:', {int(100*k): round(v, 1) for k, v in pq.items()}, '°C')
```

```
Población en el AOI (WorldPop 2020): 1,203,526  |  en celdas con LST válida y sin agua dominante: 1,152,338 (95.7 %)
Celdas pobladas: 9,956
LST P90 ponderada por población, percentiles: {5: 42.1, 25: 46.5, 50: 48.8, 75: 50.3, 95: 51.7, 99: 52.6} °C
```

```
# Clases de amenaza por LST P90 con intervalos fijos (como CLIMAAX, pero con cortes calibrados al rango local)
cortes = [0, 45, 47.5, 50, 52, 99]
nombres = ['Muy baja (< 45 °C)', 'Baja (45–47,5)', 'Media (47,5–50)', 'Alta (50–52)', 'Muy alta (≥ 52 °C)']
clase_H = np.digitize(P90p, cortes[1:-1]) + 1           # 1..5
tot = POP[pob].sum()
tabla = pd.DataFrame({
    'Clase LST P90': nombres,
    'Personas': [POP[pob & (clase_H == k)].sum() for k in range(1, 6)],
})
tabla['% de la población'] = 100 * tabla.Personas / tot
tabla['Personas'] = tabla.Personas.round(-2).astype(int)
tabla['% de la población'] = tabla['% de la población'].round(1)

# Infraestructura crítica por clase
def clase_en_punto(lat, lon):
    r, c = pop_ds.index(lon, lat)
    if 0 <= r < POP.shape[0] and 0 <= c < POP.shape[1] and np.isfinite(P90p[r, c]):
        return int(clase_H[r, c])
    return np.nan
osm = pd.read_csv(DATA / 'osm_infraestructura_cartagena.csv')
osm['clase'] = [clase_en_punto(a, b) for a, b in zip(osm.lat, osm.lon)]
infra = osm.dropna(subset=['clase']).groupby(['tipo',
'clase']).size().unstack(fill_value=0).reindex(columns=range(1, 6), fill_value=0)
infra.columns = nombres
print('Infraestructura por clase de LST P90 (número de sitios):')
print(infra)
tabla
```

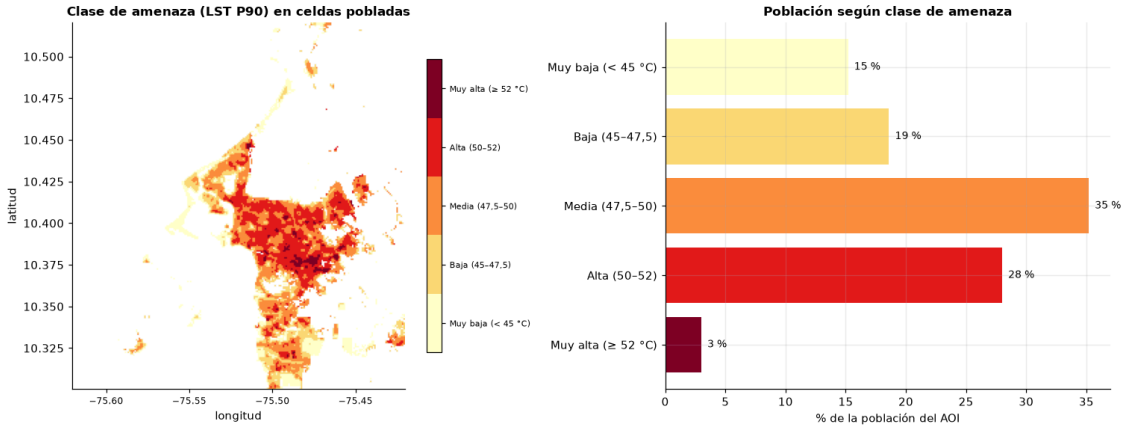
```
Infraestructura por clase de LST P90 (número de sitios):
              Muy baja (< 45 °C)  Baja (45–47,5)  Media (47,5–50)  \
tipo
clinic                         6              10               13
hospital                       4               5               10
nursing_home                   0               0                1
school                        21              34              139

              Alta (50–52)  Muy alta (≥ 52 °C)
tipo
clinic                  14                   0
hospital                10                   0
nursing_home             0                   0
school                 156                   6
```

	Clase LST P90	Personas	% de la población
0	Muy baja (< 45 °C)	175100	15.200000
1	Baja (45–47,5)	213900	18.600000
2	Media (47,5–50)	406000	35.200001
3	Alta (50–52)	322600	28.000000
4	Muy alta (≥ 52 °C)	34700	3.000000

```
fig, ax = plt.subplots(1, 2, figsize=(14, 5.4))
Hm = np.where(pob, clase_H, np.nan).astype(float)
cm = plt.get_cmap('YlOrRd', 5)
im = ax[0].imshow(Hm, extent=(pop_ds.bounds.left, pop_ds.bounds.right, pop_ds.bounds.bottom, pop_ds.bounds.top),
cmap=cm, vmin=0.5, vmax=5.5)
cb_ = plt.colorbar(im, ax=ax[0], ticks=range(1, 6), shrink=0.8); cb_.ax.set_yticklabels(nombres, fontsize=7)
ax[0].set_title('Clase de amenaza (LST P90) en celdas pobladas'); ax[0].grid(False); ax[0].set_xlabel('longitud');
ax[0].set_ylabel('latitud')
ax[0].set_xticks(np.arange(-75.6, -75.41, 0.05)); ax[0].tick_params(axis='x', labelsize=8)
ax[1].barh(nombres[::-1], tabla['% de la población'].values[::-1], color=[cm(i) for i in range(5)][::-1])
for i, v in enumerate(tabla['% de la población'].values[::-1]):
    ax[1].text(v + 0.5, i, f'{v:.0f} %', va='center', fontsize=9)
```

```
ax[1].set_xlabel('% de la población del AOI'); ax[1].set_title('Población según clase de amenaza')
plt.tight_layout(); plt.savefig(OUT_FINAL / 'fig11_exposicion.png', bbox_inches='tight'); plt.show()
```



Lectura. De los 1,15 millones de habitantes con LST válida, 15 % vive en celdas de la clase más baja (menos de 45 °C) y 31 % en las dos clases más altas (50 °C o más): 28 % en «alta» y 3 % en «muy alta» (unas 357 000 personas). En la infraestructura de OpenStreetMap, 24 de los 72 hospitales y clínicas están en la clase «alta» y ninguno en la «muy alta»; entre los colegios, 162 de 356 (46 %) están en las dos clases más altas. El mapa de OpenStreetMap está incompleto, así que estos conteos son una cota y no un inventario.

10. FASE 5: VULNERABILIDAD (PARCIAL)

Esta fase es parcial. La vulnerabilidad al calor depende sobre todo de quién es la persona expuesta: edad, salud, vivienda e ingresos (Salazar-Ceballos & Álvarez-Miño, 2025). Esos datos (censo DANE 2018 por manzana, SISBÉN, RIPS) no están disponibles de forma abierta y se dejan como tarea pendiente. Con datos abiertos solo se pueden construir dos componentes del entorno: el déficit de arbolado (sensibilidad física) y la distancia al hospital o clínica más cercano de OpenStreetMap (acceso a atención). Son aproximaciones y su cobertura en OpenStreetMap es incompleta.

```
# Fase 5 (parcial): componentes de vulnerabilidad que se pueden construir hoy con datos abiertos
# (a) déficit de arbolado, (b) lejanía a hospitales y clínicas de OpenStreetMap.
# Pendiente: composición por edad y condiciones sociales (censo DANE 2018 por manzana, SISBÉN).
from scipy.spatial import cKDTree
ys, xs = np.mgrid[0:POP.shape[0], 0:POP.shape[1]]
tr = pop_ds.transform
lon_p, lat_p = tr.c + (xs + 0.5) * tr.a, tr.f + (ys + 0.5) * tr.e
salud = osm[osm.tipo.isin(['hospital', 'clinic'])]
kx = np.cos(np.deg2rad(10.4))
arbol_tree = cKDTree(np.c_[salud.lon.values * kx, salud.lat.values])
dist_km = arbol_tree.query(np.c_[lon_p[pob] * kx, lat_p[pob]])[0] * 111.2
DIST = np.full(POP.shape, np.nan); DIST[pob] = dist_km
DEF_ARB = 1 - np.clip(ARBp / 0.6, 0, 1)                    # 0 con >= 60 % de copa en la celda, 1 sin árboles

def rango(A):
    """Rango percentil (0-1) entre celdas pobladas."""
    out = np.full(A.shape, np.nan)
    out[pob] = stats.rankdata(A[pob]) / pob.sum()
    return out
rH, rE, rA, rD = rango(P90p), rango(POP), rango(DEF_ARB), rango(DIST)
VULN = (rA + rD) / 2
print(f'Distancia a hospital o clínica (km) en celdas pobladas: mediana {np.nanmedian(DIST[pob]):.2f}, P90
{np.nanpercentile(DIST[pob], 90):.2f}')
corr = pd.DataFrame({'amenaza (LST P90)': P90p[pob], 'población por celda': POP[pob], 'déficit de arbolado':
DEF_ARB[pob], 'distancia a salud': DIST[pob]}).corr(method='spearman').round(2)
print('\nCorrelación de Spearman entre componentes (celdas pobladas):')
corr
```

```
Distancia a hospital o clínica (km) en celdas pobladas: mediana 1.20, P90 6.98

Correlación de Spearman entre componentes (celdas pobladas):
```

	amenaza (LST P90)	población por celda	déficit de arbolado	distancia a salud
amenaza (LST P90)	1.00	0.14	0.44	−0.31
población por celda	0.14	1.00	0.16	−0.47
déficit de arbolado	0.44	0.16	1.00	−0.29
distancia a salud	−0.31	−0.47	−0.29	1.00

Lectura. La mediana de la distancia al servicio de salud más cercano es de 1,2 km, pero el 10 % de la población más lejana está a más de 7 km. El déficit de arbolado y la LST P90 se correlacionan (ρ = 0,44), de modo que el componente de arbolado aporta información parcialmente repetida con la amenaza, y la distancia a salud es mayor en las celdas menos pobladas (ρ = −0,47 con la población). Esa colinealidad parcial se prueba en la sensibilidad de la Fase 6.

11. FASE 6: RIESGO RELATIVO

El riesgo relativo combina la amenaza (LST P90), la exposición (población por celda) y la vulnerabilidad parcial de la sección anterior, como se describe en la sección 4.7. Cada componente se convierte en rango percentil entre las celdas pobladas y se combinan con media geométrica; las clases son quintiles de población, de modo que cada una agrupa a la quinta parte de los habitantes. Este resultado **ordena** las zonas; no estima pérdidas, y no se calcula la pérdida anual esperada porque no hay una función de impacto de calor validada para Cartagena.

```
# Fase 6: riesgo relativo = media geométrica de los rangos percentiles de amenaza, exposición y vulnerabilidad
RIESGO = (rH * rE * VULN) ** (1 / 3)
orden_r = np.argsort(RIESGO[pob])
w = POP[pob][orden_r]; cum = np.cumsum(w) / w.sum()
cls = np.digitize(cum, [0.2, 0.4, 0.6, 0.8]) + 1                    # quintiles de población: cada clase suma ~20
%
CLASE_R = np.full(POP.shape, np.nan); tmp = np.empty(len(orden_r)); tmp[orden_r] = cls; CLASE_R[pob] = tmp
tab_r = []
for k in range(1, 6):
    m = pob & (CLASE_R == k)
    tab_r.append((k, int(m.sum()), POP[m].sum(), np.nanmean(P90p[m]), np.nanmean(POP[m]), np.nanmean(DIST[m]),
np.nanmean(DEF_ARB[m])))
df_riesgo = pd.DataFrame(tab_r, columns=['Clase de riesgo relativo', 'Celdas', 'Personas', 'LST P90 media [°C]',
'Personas por celda', 'Distancia a salud [km]', 'Déficit de arbolado'])
df_riesgo['Personas'] = df_riesgo.Personas.round(-2).astype(int)
df_riesgo.iloc[:, 3:] = df_riesgo.iloc[:, 3:].astype('float64').round(2)

# Sensibilidad a la fórmula: variantes y coincidencia de la clase más alta
variantes = {
    'Base (geométrica, pesos iguales)': RIESGO,
    'Media aritmética': (rH + rE + VULN) / 3,
    'Amenaza con doble peso': (rH ** 2 * rE * VULN) ** (1 / 4),
    'Sin vulnerabilidad': (rH * rE) ** 0.5,
    'Solo déficit de arbolado como vulnerabilidad': (rH * rE * rA) ** (1 / 3),
}
top = lambda R: R >= np.nanpercentile(R[pob], 80)
filas = []
for k, R in variantes.items():
    rho = stats.spearmanr(RIESGO[pob], R[pob]).correlation
    filas.append((k, round(rho, 2), round(100 * (top(R) & top(RIESGO) & pob).sum() / (top(RIESGO) & pob).sum())))
df_sens = pd.DataFrame(filas, columns=['Variante', 'Spearman con la base', '% de celdas del 20 % superior que
coinciden'])
print('Sensibilidad del ranking a la fórmula:')
print(df_sens.to_string(index=False))
df_riesgo
```

```
Sensibilidad del ranking a la fórmula:
                                Variante  Spearman con la base  % de celdas del 20 % superior que coinciden
          Base (geométrica, pesos iguales)                  1.00                                          100
```

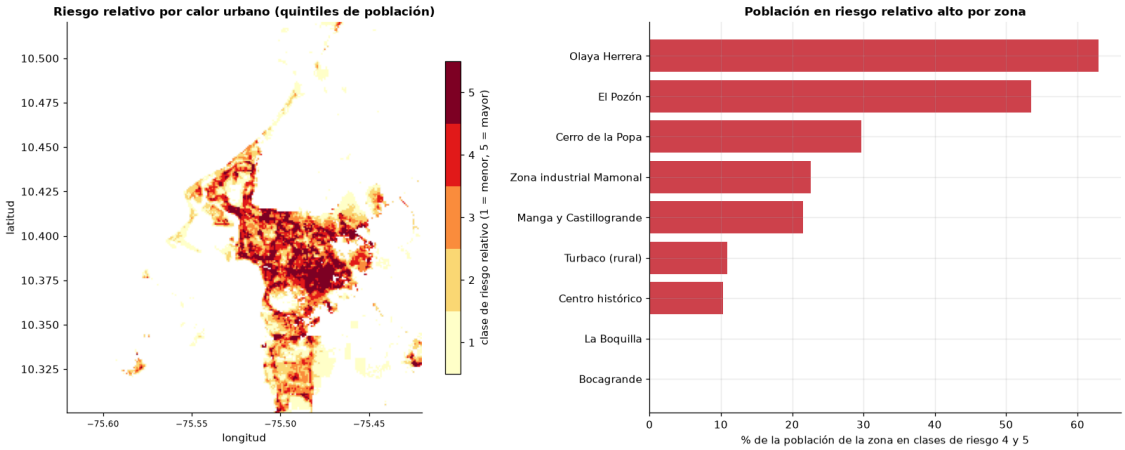
```
                                  Media aritmética                0.96                                  91
                              Amenaza con doble peso              0.97                                  86
                                  Sin vulnerabilidad              0.93                                  74
Solo déficit de arbolado como vulnerabilidad                      0.95                                  82
```

	Clase de riesgo relativo	Celdas	Personas	LST P90 media [°C]	Personas por celda	Distancia a salud [km]	Déficit de arbolado
0	1	2539	230400	44.730000	90.750000	3.44	0.61
1	2	2105	230400	47.610001	109.470001	2.77	0.84
2	3	1963	230400	48.900002	117.370003	2.69	0.93
3	4	1792	230500	49.880001	128.600006	1.77	0.98
4	5	1557	230600	50.790001	148.119995	1.41	1.00

Lectura. Cada clase agrupa a unas 230 000 personas por construcción. De la clase 1 a la 5, la LST P90 media sube de 44,7 a 50,8 °C, la población por celda de 91 a 148 personas y el déficit de arbolado de 0,61 a 1,00; en cambio, la distancia a salud baja de 3,4 a 1,4 km, porque las zonas densas tienen más hospitales y clínicas. Por eso, las clases altas lo son por calor, población y falta de árboles, y no por lejanía a la atención en salud: la vulnerabilidad parcial no crece junto con la amenaza. La sensibilidad a la fórmula es baja: las cuatro variantes mantienen una correlación de Spearman de 0,93 a 0,97 con la base, y entre 74 % y 91 % de las celdas del 20 % superior coinciden. La variante sin vulnerabilidad es la que más se separa (74 %), de modo que la componente parcial aporta información, pero no cambia el orden global.

Figura 12. Riesgo relativo por calor urbano y porcentaje de la población de cada zona en las clases 4 y 5.

```
fig, ax = plt.subplots(1, 2, figsize=(15, 6))
cm5 = plt.get_cmap('YlOrRd', 5)
im = ax[0].imshow(CLASE_R, extent=(pop_ds.bounds.left, pop_ds.bounds.right, pop_ds.bounds.bottom,
pop_ds.bounds.top), cmap=cm5, vmin=0.5, vmax=5.5)
plt.colorbar(im, ax=ax[0], ticks=range(1, 6), shrink=0.8, label='clase de riesgo relativo (1 = menor, 5 = mayor)')
ax[0].set_title('Riesgo relativo por calor urbano (quintiles de población)'); ax[0].grid(False);
ax[0].set_xlabel('longitud'); ax[0].set_ylabel('latitud')
ax[0].set_xticks(np.arange(-75.6, -75.41, 0.05)); ax[0].tick_params(axis='x', labelsize=8)
filas = []
for z, (w_, s_, e_, n_) in ZONAS.items():
    m = (lon_p >= w_) & (lon_p <= e_) & (lat_p >= s_) & (lat_p <= n_) & pob
    if m.sum() > 5:
        filas.append((z, POP[m].sum(), 100 * POP[m & (CLASE_R >= 4)].sum() / POP[m].sum()))
df_zr = pd.DataFrame(filas, columns=['Zona', 'Personas', '% en clases 4-5']).sort_values('% en clases 4-5')
ax[1].barh(df_zr.Zona.str.replace(' (referencia rural)', ' (rural)'), df_zr['% en clases 4-5'], color='#c1121f',
alpha=.8)
ax[1].set_xlabel('% de la población de la zona en clases de riesgo 4 y 5'); ax[1].set_title('Población en riesgo
relativo alto por zona')
plt.tight_layout(); plt.savefig(OUT_FINAL / 'fig12_riesgo_relativo.png', bbox_inches='tight'); plt.show()
df_zr = df_zr.assign(Personas=lambda d: d.Personas.round(-2).astype(int)).round({'% en clases 4-5': 0})
df_zr.sort_values('% en clases 4-5', ascending=False)
```



	Zona	Personas	% en clases 4-5
4	Olaya Herrera	8200	63.0
5	El Pozón	49300	53.0
3	Cerro de la Popa	42900	30.0
6	Zona industrial Mamonal	111200	23.0
2	Manga y Castillogrande	44300	22.0
8	Turbaco (referencia rural)	7800	11.0
0	Centro histórico	23700	10.0
1	Bocagrande	20100	0.0
7	La Boquilla	13600	0.0

Lectura. Olaya Herrera (63 %) y El Pozón (53 %) tienen más de la mitad de su población en las dos clases más altas de riesgo relativo; siguen el Cerro de la Popa (30 %), la zona industrial de Mamonal (23 %), Manga y Castillogrande (22 %), el centro histórico (10 %) y, sin población en esas clases, Bocagrande y La Boquilla. La referencia rural de Turbaco queda en 11 %. Los rectángulos cubren fracciones de los barrios (por ejemplo, unas 8 000 personas en el de Olaya Herrera), de modo que las cifras por zona son ilustrativas y no estadísticas oficiales de los barrios.

```
# Persona-días con Tmáx >= 35 °C al año: población del AOI por días calurosos (la estación se aplica a toda la ciudad)
pob_total = POP[pob].sum(); pob_alta = POP[pob & (clase_H >= 4)].sum()
filas = [('Observado 2019–2026', '—', base_tx35, pob_total * base_tx35 / 1e6, pob_alta * base_tx35 / 1e6)]
for e in ESC:
    for per in PERIODOS:
        d = df_abs[(df_abs.escenario == e) & (df_abs.periodo == per)].dias_tx35.median()
        filas.append((ESC_LABEL[e].split(' (')[0], per, d, pob_total * d / 1e6, pob_alta * d / 1e6))
df_pd = pd.DataFrame(filas, columns=['Escenario', 'Periodo', 'Días/año con Tmáx ≥ 35 °C', 'Millones de persona-días (toda la población)', 'Millones de persona-días (clases de amenaza 4-5)']).round(1)
df_pd
```

	Escenario	Periodo	Días/año con Tmáx ≥ 35 °C	Millones de persona-días (toda la población)	Millones de persona-días (clases de amenaza 4-5)
0	Observado 2019–2026	—	13.8	15.9	4.9
1	SSP2-4.5	2021–2040	25.4	29.2	9.1
2	SSP2-4.5	2041–2060	51.1	58.9	18.3
3	SSP2-4.5	2061–2080	95.2	109.7	34.0

	Escenario	Periodo	Días/año con Tmáx ≥ 35 °C	Millones de persona-días (toda la población)	Millones de persona-días (clases de amenaza 4-5)
4	SSP3-7.0	2021–2040	28.4	32.7	10.1
5	SSP3-7.0	2041–2060	58.9	67.8	21.0
6	SSP3-7.0	2061–2080	153.0	176.3	54.7
7	SSP5-8.5	2021–2040	19.5	22.5	7.0
8	SSP5-8.5	2041–2060	76.5	88.1	27.3
9	SSP5-8.5	2061–2080	189.3	218.2	67.6

Lectura. Con una sola estación para toda la ciudad, las persona-días son una medida gruesa de la carga de calor: resultan de multiplicar una población fija por los días de la Fase 1, y por eso crecen en la misma proporción que ellos. Pasan de unos 16 millones al año con el clima observado a 110, 176 y 218 millones en 2061–2080 bajo SSP2-4.5, SSP3-7.0 y SSP5-8.5; para la población de las dos clases de amenaza más altas, de 5 a 34, 55 y 68 millones. La tabla no incorpora crecimiento de la población ni cambios en su estructura por edad.

12. CONCLUSIONES

1. **El calor extremo aumenta de forma robusta en todos los escenarios y el escenario decide después de 2040.** Los días con Tmáx sobre el P90 de 1985–2014 pasan de 10 % a una mediana de 50 %, 63 % y 78 % hacia 2061–2080 bajo SSP2-4.5, SSP3-7.0 y SSP5-8.5; los días dentro de olas de calor, de unos 5 al año a 85, 144 y 223 respectivamente. Los cinco modelos coinciden en el signo; difieren en la magnitud.
2. **La definición importa más que el modelo para decidir qué es una ola.** La condición de Tmín ≥ 24 °C de la versión 1 no separaba nada (92 % de los días la cumplen); el percentil local sí. Con el percentil, la «ola de calor» pasa a ser la condición típica hacia 2060–2080 en el escenario más cálido, lo que cambia el significado de la palabra.
3. **El calor superficial se concentra en la periferia construida y no en el centro turístico.** La superficie construida es 6,1 °C más caliente en la media (+3,9 °C en el P90) que una referencia vegetada, y cada 10 % más de arbolado se asocia con −0,68 °C de LST P90 a escala de 300 m.
4. **La humedad cambia el orden de los meses críticos.** El Humidex es más alto de mayo a septiembre, y en 46 % de los días ya cae en la categoría más alta.
5. **El riesgo relativo es mayor en los barrios del sureste.** Olaya Herrera y El Pozón tienen más de la mitad de su población en las dos clases más altas. Este resultado ordena zonas con una vulnerabilidad parcial; debe contrastarse con los datos sociales que faltan antes de usarse para priorizar inversión.
6. **Implicaciones para el SNGRD.** El aumento del calor presiona a la salud pública, que cuenta con una meta de alerta temprana territorial al 2031 (Salazar-Ceballos & Álvarez-Miño, 2025); a la productividad laboral y al consumo de energía para enfriamiento. Esta última afirmación no se evaluó aquí.

12.1 Limitaciones

- **Cinco modelos, un miembro cada uno.** El rango entre modelos no es un intervalo de confianza del CMIP6 completo.
- **Una celda de 25 km para el clima de fondo.** La celda mezcla tierra y mar; los modelos quedan por encima de la estación en los cuantiles altos y el método relativo elimina ese sesgo, pero el absoluto solo lo corrige con el anclaje a la estación.
- **Estación de corta duración.** Siete años, con 26 % de los días sin datos suficientes. Los índices absolutos heredan su variabilidad interanual.
- **LST no es temperatura del aire.** Es del momento de paso del satélite (10:30) y de la superficie, y no mide la noche, que es clave para la recuperación del cuerpo.
- **Vulnerabilidad sin variables sociales.** Sin edad, salud ni condiciones de vivienda, el riesgo relativo ordena sobre todo calor, población y acceso a salud.
- **Humidex.** Categorías diseñadas para otro clima, una sola estación y humedad constante en la sensibilidad.
- **Sin evaluación de pérdidas.** No hay una función de impacto de calor validada para Cartagena; la pérdida anual esperada queda pendiente.
- **Zonas aproximadas.** Los rectángulos no son límites oficiales de barrios.

12.2 Próximos pasos

1. **Datos sociales por manzana.** Incorporar el censo DANE 2018 (población mayor de 65 y menor de 5 años, hogares, materiales de la vivienda) y reemplazar la vulnerabilidad parcial por un índice compuesto, con sensibilidad a los pesos.
2. **Datos de salud.** Cruzar con RIPS o registros de urgencias para estimar la relación entre calor y atenciones, y poder calcular una función de impacto propia.
3. **Más estaciones y más años.** Pedir al IDEAM las series históricas de Rafael Núñez anteriores a 2019 y de otras estaciones de la ciudad, para validar los umbrales y reducir la incertidumbre de la referencia.
4. **Humedad y viento en el futuro.** Añadir la humedad relativa proyectada (variable `hurs` de NEX-GDDP-CMIP6) para pasar de la sensibilidad a una proyección del Humidex.
5. **Imágenes de la tarde y la noche.** Evaluar LST de otros sensores para ver el calor de la tarde y de la noche.
6. **Validar con la Subdirección de Manejo de Desastres** la utilidad del producto para alertas y refugios climáticos.

13. TRAZABILIDAD CON EL LIBRO «TEORÍA Y MÉTODOS DE ANÁLISIS DE RIESGOS»

Este cuaderno es un caso aplicado del libro (https://mromerot.github.io/teoria-riesgos/), y los dos se alimentan mutuamente.

Elemento	Libro de teoría	Qué hizo este cuaderno
Marco del riesgo (amenaza, exposición, vulnerabilidad)	Capítulos 11 y 12	Se adoptó tal cual
Escenarios	Capítulo 23: SSP2-4.5 y SSP3-7.0 como principales, SSP5-8.5 como prueba de estrés	Se adoptó la jerarquía; los escenarios no se promedian
Rastro de evidencia	Cada afirmación con cita; lo no verificado, marcado	Se adoptó; los datos sintéticos de la versión 1 se retiraron
Definición de ola de calor	Capítulo 309: umbral local, ola frente a calor persistente	Se implementó con percentiles locales y se mostró que el umbral absoluto de la versión 1 no discriminaba
Isla de calor urbana con LST	Capítulo 309: Landsat 30 m de CLIMAAX	Se implementó con datos reales de 2018 a 2024
Matriz de riesgo relativo	Capítulo 309: matriz de 10 por 10 de CLIMAAX	Se adaptó a rangos percentiles y quintiles de población
Pérdida anual esperada	Capítulo 22	**No** se calculó: falta una función de impacto
Exposición y vulnerabilidad	Capítulos 41 y 43	Parcial: sin datos sociales

En sentido inverso, este cuaderno aporta al capítulo de olas de calor del libro la prueba de que el umbral nocturno absoluto no discrimina en una ciudad tropical, el uso de los datos abiertos del IDEAM, el método de anclaje por cuantiles y la cuantificación de la isla de calor superficial en una ciudad del Caribe.

14. REPRODUCIBILIDAD Y DATOS

El cuaderno corre completo con los archivos de `data/`. Para regenerarlos desde las fuentes originales (sin credenciales), en este orden:

Script	Qué hace
`scripts/descargar_observaciones.py`	Series diarias de la estación IDEAM, Humidex observado y ERA5-Land
`scripts/extraer_nex_gddp.py`	Series de NEX-GDDP-CMIP6 para una ventana de 3 por 3 celdas por OPeNDAP (tarda ∼1 h)
`scripts/procesar_nex.py`	Ensambla la celda del aeropuerto y los índices de la ventana 3 por 3
`scripts/extraer_lst_landsat.py`	Compuesto de LST de Landsat 8 y 9 en Planetary Computer

Script	Qué hace
`scripts/extraer_worldcover.py`	Fracciones de cobertura WorldCover 2021
`scripts/extraer_poblacion_worldpop.py`	Población WorldPop 2020 recortada al área de estudio
`scripts/extraer_osm.py`	Hospitales, clínicas, hogares de mayores y colegios de OpenStreetMap

La semilla no interviene: no hay componentes aleatorios. Los cinco archivos de `observable/data/` se regeneran en la última celda de código.

```
# Exports planos para Observable (se regeneran en cada ejecución)
ts = df_idx.copy(); ts['escenario_etiqueta'] = ts.escenario.map(ESC_LABEL)
ts.round(2).to_json(OUT_OBS / 'hazard_timeseries.json', orient='records', force_ascii=False)

estad = (df_idx.groupby(['escenario', 'anio'])[['TX90p', 'HWF', 'HWD', 'HWML', 'TX35']]
         .agg(['median', 'min', 'max']).round(2))
estad.columns = ['_'.join(c) for c in estad.columns]
estad.reset_index().to_json(OUT_OBS / 'hazard_ensemble_stats.json', orient='records', force_ascii=False)

# GeoJSON: % de días con Tmáx > P90 en cada celda 3x3 (2061–2080, mediana de modelos)
feats = []
for e in ESC:
    M = res[e]
    for i, la in enumerate(lats[::-1]):
        for j, lo in enumerate(lons):
            d = 0.125
            if np.isnan(M.values[i, j]):
                continue
            feats.append({'type': 'Feature', 'geometry': {'type': 'Polygon', 'coordinates': [[[lo - d, la - d], [lo
+ d, la - d], [lo + d, la + d], [lo - d, la + d], [lo - d, la - d]]]},
                          'properties': {'escenario': e, 'tx90p_2061_2080': round(float(M.values[i, j]), 1)}})
with open(OUT_OBS / 'hazard_delta_hwf.geojson', 'w', encoding='utf-8') as f:
    json.dump({'type': 'FeatureCollection', 'name': '% de días con Tmáx > P90 de 1985–2014, 2061–2080',
               'metadata': {'unidades': '% de días', 'fuente': 'NEX-GDDP-CMIP6, mediana de 5 modelos'}, 'features':
feats}, f, ensure_ascii=False, separators=(',', ':'))

# GeoJSON: LST a 360 m (bloques de 12x12 píxeles)
f12 = 12
lm12 = np.nanmean(bloques(LST_MEDIA, f12), axis=(1, 3)); l90_12 = np.nanmean(bloques(LST_P90, f12), axis=(1, 3));
ag12 = np.nanmean(bloques(AGUA, f12), axis=(1, 3))
cb12 = np.nanmean(bloques(CONSTRUIDO, f12), axis=(1, 3))
dx = (EXT[1] - EXT[0]) / lst.width * f12; dy = (EXT[3] - EXT[2]) / lst.height * f12
feats = []
for i in range(lm12.shape[0]):
    for j in range(lm12.shape[1]):
        if np.isfinite(lm12[i, j]) and ag12[i, j] < 0.5:
            x0, y1 = EXT[0] + j * dx, EXT[3] - i * dy
            feats.append({'type': 'Feature', 'geometry': {'type': 'Polygon', 'coordinates': [[[x0, y1], [x0 + dx,
y1], [x0 + dx, y1 - dy], [x0, y1 - dy], [x0, y1]]]},
                          'properties': {'lst_media': round(float(lm12[i, j]), 1), 'lst_p90': round(float(l90_12[i,
j]), 1), 'construido': round(float(cb12[i, j]), 2)}})
with open(OUT_OBS / 'uhi_lst_raster.geojson', 'w', encoding='utf-8') as f:
    json.dump({'type': 'FeatureCollection', 'name': 'LST Landsat 8/9 2018–2024, bloques de ~360 m',
               'metadata': {'unidades': '°C', 'fuente': 'Landsat C2 L2 (Planetary Computer), WorldCover 2021'},
'features': feats}, f, ensure_ascii=False, separators=(',', ':'))
df_zonal.to_json(OUT_OBS / 'uhi_zonal_summary.json', orient='records', force_ascii=False)
for n in ('hazard_timeseries.json', 'hazard_ensemble_stats.json', 'hazard_delta_hwf.geojson',
'uhi_lst_raster.geojson', 'uhi_zonal_summary.json'):
    print(f'  {n}: {(OUT_OBS / n).stat().st_size / 1024:.0f} KB')
```

```
  hazard_timeseries.json: 288 KB
  hazard_ensemble_stats.json: 76 KB
  hazard_delta_hwf.geojson: 4 KB
  uhi_lst_raster.geojson: 622 KB
  uhi_zonal_summary.json: 2 KB
```

15. REFERENCIAS

Las obras citadas en el texto se listan a continuación, en el orden en que las genera el sitio a partir de `references.bib`.

Disponibilidad de datos y código. Los datos recortados al área de estudio y los scripts que los generan están en el repositorio del cuaderno (`data/` y `scripts/`). Las fuentes originales son abiertas: datos.gov.co (IDEAM), NASA NEX-GDDP-CMIP6, Microsoft Planetary Computer (Landsat Collection 2 Level 2 y ESA WorldCover), WorldPop y OpenStreetMap. Código bajo licencia Apache-2.0 y texto bajo CC-BY-4.0.

REFERENCES

Arias, P. A., Villegas, L. D., Mesa, O. J., & others. (2022). Implicaciones metodológicas e inconsistencias de la Tercera Comunicación Nacional sobre Cambio Climático de Colombia. *Revista De La Academia Colombiana De Ciencias Exactas, Físicas Y Naturales*, *46*(180), 769–794. https://doi.org/10.18257/raccefyn.1705

Avila-Diaz, A., Torres, R. R., Zuluaga, C. F., Cerón, W. L., Oliveira, L., Benezoli, V., Rivera, I. A., Marengo, J. A., Wilson, A. B., & Medeiros, F. (2023). Current and Future Climate Extremes Over Latin America and Caribbean: Assessing Earth System Models from High Resolution Model Intercomparison Project (HighResMIP). *Earth Systems and Environment*, *7*, 99–130. https://doi.org/10.1007/s41748-022-00337-7

Bubeck, P., Ozturk, U., Aristizabal, E., Thieken, A. H., & Wagener, T. (2025). Mortality reduction despite changing climate extremes requires better understanding of human behavioral response to warnings. *Environmental Research Letters*, *20*, 101004. https://doi.org/10.1088/1748-9326/ae034f

CLIMAAX Consortium. (2025,). *CLIMAAX CRA Handbook*. https://handbook.climaax.eu/

Congreso de la República de Colombia. (2012,). *Ley 1523 de 2012, por la cual se adopta la política nacional de gestión del riesgo de desastres y se establece el Sistema Nacional de Gestión del Riesgo de Desastres*.

Esteban-Cantillo, O. J., Clerici, N., Avila-Diaz, A., & Quesada, B. (2024). Historical and future extreme climate events in highly vulnerable small Caribbean Islands. *Climate Dynamics*, *62*, 7233–7250. https://doi.org/10.1007/s00382-024-07276-1

GFDRR and World Bank. (2025,). *Risk Data Library: Country Climate and Disaster Risk Screening Tools (CCDR-tools)*. https://gfdrr.github.io/CCDR-tools/home.html

Giraldo-Charria, D. L., Escobedo, F. J., Clerici, N., & Quesada, B. (2025). Urban forests mitigate extreme heat exposure in a vulnerable tropical city. *Urban Climate*, *59*, 102311. https://doi.org/10.1016/j.uclim.2025.102311

IDEAM. (2026,). *Temperatura máxima del aire (ccvq-rp9s) y temperatura mínima del aire (afdg-3zpb), estación Aeropuerto Rafael Núñez*. https://www.datos.gov.co/

Masterton, J. M., & Richardson, F. A. (1979). *Humidex: a method of quantifying human discomfort due to excessive heat and humidity* (Techreport Nos. CLI 1–79).

Muñoz-Sabater, J., Dutra, E., Agustí-Panareda, A., & others. (2021). ERA5-Land: a state-of-the-art global reanalysis dataset for land applications. *Earth System Science Data*, *13*, 4349–4383. https://doi.org/10.5194/essd-13-4349-2021

Pérez-Arévalo, R., Jiménez-Caldera, J. E., Serrano-Montes, J. L., Rodrigo-Comino, J., Ortiz Royero, J. C., & Caballero-Calvo, A. (2025). Impacts of Urban Morphology on Micrometeorological Parameters and Cyclonic Phenomena in Northern Colombian Caribbean. *Climate*, *13*(5), 87. https://doi.org/10.3390/cli13050087

Riahi, K., Vuuren, D. P. van, Kriegler, E., Edmonds, J., O'Neill, B. C., Fujimori, S., & others. (2017). The Shared Socioeconomic Pathways and their energy, land use, and greenhouse gas emissions implications: An overview. *Global Environmental Change*, *42*, 153–168. https://doi.org/10.1016/j.gloenvcha.2016.05.009

Salazar-Ceballos, A., & Álvarez-Miño, L. (2025). Sistema de alerta temprana para altas temperaturas y olas de calor: la necesidad de una política de salud pública en Colombia. *Biomédica*, *45*(Sp. 2), 7–16. https://doi.org/10.7705/biomedica.7934

Stewart, I. D., & Oke, T. R. (2012). Local Climate Zones for urban temperature studies. *Bulletin of the American Meteorological Society*, *93*(12), 1879–1900. https://doi.org/10.1175/BAMS-D-11-00019.1

Thrasher, B., Wang, W., Michaelis, A., Melton, F., Lee, T., & Nemani, R. (2022). NASA Global Daily Downscaled Projections, CMIP6. *Scientific Data*, *9*, 262. https://doi.org/10.1038/s41597-022-01393-4

UNDRR, & ISC. (2025,). *2025 Hazard Information Profiles (HIPs) online reference*. PreventionWeb, DRR glossary. https://www.preventionweb.net/drr-glossary/hips

Unidad Nacional para la Gestión del Riesgo de Desastres. (2019). *Atlas de Riesgo de Colombia: revelando los desastres latentes*. UNGRD.

WorldPop. (2020,). *Población y estructuras de edad y sexo de Colombia 2020 (100 m, no restringida)*. https://www.worldpop.org/

Wulder, M. A., Loveland, T. R., Roy, D. P., Crawford, C. J., Masek, J. G., Woodcock, C. E., & others. (2019). Current status of Landsat program, science, and applications. *Remote Sensing of Environment*, *225*, 127–147. https://doi.org/10.1016/j.rse.2019.02.015

Zanaga, D., Van De Kerchove, R., Daems, D., & others. (2022,). *ESA WorldCover 10 m 2021 v200*. https://doi.org/10.5281/zenodo.7254221

Zippenfenig, P. (2023,). *Open-Meteo.com Weather API*. https://doi.org/10.5281/zenodo.7970649