

# **MODELO PREDICTIVO DE RIESGO CLÍNICO DE DENGUE SEGÚN SÍNTOMAS Y GRUPO ETARIO EN PACIENTES NOTIFICADOS EN BUCARAMANGA/COLOMBIA.**

## **Integrantes**

Geraldly Bernal Bermudez  
Yeimi Tatiana Ramirez Marquez  
Brahian Alejandro Rodriguez Restrepo

## **Ejecutores**

Natalia Betancur Herrera  
Frank Yesid Zapata Castaño  
Margarita Maria Orozco

**Universidad de Antioquia, Universidad de Caldas  
Talent Tech  
BOOTCAMP Inteligencia Artificial**

**Mayo 2025**

## 1. Introducción

- El dengue es una enfermedad viral transmitida por mosquitos que representa un problema de salud pública en muchas regiones tropicales. La vigilancia epidemiológica y el análisis predictivo pueden desempeñar un papel clave en la anticipación de brotes, identificación de grupos vulnerables y mejor asignación de recursos médicos.

Según la Organización Mundial de la Salud (OMS), los casos de dengue se han multiplicado por 8 en los últimos 20 años, alcanzando cifras históricas en América Latina (OMS, 2023).

- Objetivo general:  
Desarrollar un modelo descriptivo y predictivo que permita analizar y anticipar los casos de dengue a partir de variables demográficas, clínicas y sociales, con el fin de fortalecer la toma de decisiones en salud pública.
- Objetivos específicos:
  - Realizar análisis exploratorios sobre los casos de dengue por edad, género y año de notificación.
  - Identificar los síntomas predominantes según grupo etario y tipo de clasificación clínica.
  - Predecir la probabilidad de que un paciente tenga dengue con signos de alarma usando modelos de machine learning.
  - Generar visualizaciones claras para comunicar los hallazgos a equipos médicos y autoridades sanitarias.
- Alcance del proyecto  
  
Este estudio se basa en un conjunto de datos reales sobre notificaciones de casos de dengue. Incluye variables clínicas (síntomas), demográficas (edad, género), socioeconómicas (tipo de seguridad social) y temporales (año de notificación). Se excluyen los casos clasificados como “DENGUE GRAVE”. El modelo predictivo se centra en distinguir entre “DENGUE CON SIGNOS DE ALARMA” y “DENGUE SIN SIGNOS DE ALARMA”.

---

## 2. Metodología: CRISP-DM

### 2.1. Comprensión del Negocio

- Descripción del problema desde el punto de vista del negocio

- El aumento de los casos de dengue genera presión sobre los servicios de salud pública. Una herramienta predictiva permite anticipar la severidad clínica y orientar mejor la asignación de recursos, campañas de prevención y atención médica.
- Objetivos del negocio
  - Prever el riesgo de que un paciente desarrolle dengue con signos de alarma.
  - Identificar patrones en síntomas y conducta clínica por grupo etario y género.
  - Facilitar la toma de decisiones a partir de visualizaciones claras y modelos predictivos.
- Criterios de éxito
  - AUC mayor a 0.70 en el modelo predictivo.
  - Visualizaciones comprensibles para personal médico.
  - Identificación de síntomas críticos por grupo poblacional.

## 2.2. Comprensión de los Datos

- La base de datos contiene registros clínicos y demográficos de pacientes con dengue. Se incluyen síntomas, fecha de notificación, clasificación clínica, grupo etario, sexo, conducta médica, tipo de seguridad social, entre otros.
- - Formato Excel (.xlsx)
  - Variables numéricas (edad, año)
  - Variables categóricas (sexo, grupo etario, síntomas binarios, conducta)
  - Variables de texto (clasificación final, tipo de seguridad social)
- - Valores faltantes en fechas y síntomas.
  - Códigos inconsistentes (ej. "m", "M", "masculino").
  - Casos con clasificación "DENGUE GRAVE" se excluyen del análisis predictivo.

## 2.3. Preparación de los Datos

- Limpieza de datos
  - Normalización de formatos de fecha.
  - Homogeneización de categorías (sexo, edad).
  - Exclusión de registros "NO APLICA" y "DENGUE GRAVE".
- Transformaciones realizadas
  - Codificación binaria de síntomas (1 = sí, 0 = no).
  - One-hot encoding para clasificación de edad.

- Selección de variables
  - Variables sintomáticas (30 síntomas).
  - Variables demográficas (edad codificada).
  - Variable objetivo: dengue con signos de alarma (1) vs sin signos de alarma (0).

## 2.4. Modelado

- Algoritmos seleccionados y justificación
  - Regresión logística, porque es fácil de interpretar y adecuar a variables binarias.
- Proceso de entrenamiento y validación
  - División 70/30 para entrenamiento y prueba.
  - Estratificación para balanceo de clases.
- Métricas de evaluación
  - Matriz de confusión.
  - Curva ROC y AUC (área bajo la curva).
  - Precision, recall y F1-score.
  - Accuracy
- Comparación entre modelos

Se probaron modelos con y sin escalado de variables, sin diferencias significativas. Se optó por mantener la versión sin escalar por simplicidad interpretativa.

## 2.5. Evaluación

- Análisis de resultados

El modelo alcanzó un AUC de aproximadamente 0.78, indicando buena capacidad de discriminación. Se identificaron síntomas clave asociados al dengue con signos de alarma como fiebre, cefalea, dolor abdominal y vómito.
- Validación con expertos o stakeholders
- Lecciones aprendidas
  - La limpieza y transformación de datos toma una parte considerable del tiempo.
  - Visualizaciones claras facilitan la comprensión y uso del modelo.
- Limitaciones

- Datos no balanceados entre tipos de clasificación.
- No se incluyeron datos de seguimiento posterior al diagnóstico.

## 2.6. Implementación

- Plan de despliegue

El modelo puede integrarse en sistemas de vigilancia epidemiológica locales para evaluar riesgo al momento de la notificación.

- Herramientas utilizadas

- Google Colab (Python)
- Scikit-learn (modelado)
- Matplotlib y Seaborn (visualización)
- Pandas (manipulación de datos)

- Consideraciones para mantenimiento

- Actualizar el modelo con nuevos datos regularmente.
- Validar el modelo cada 6 meses con casos recientes.

## 3. Presupuesto

Concepto	Cantidad	Costo Unitario	Total
Licencias de software (ej. Anaconda, Power BI Pro)	2	\$50.000	\$100.000
Sueldos/Data Scientist (mes)	1	\$4.000.000	\$4.000.000
Infraestructura en la nube (ej. Azure, AWS)	1 mes	\$300.000	\$300.000
Visualizaciones/Reportes	1	\$200.000	\$200.000
Otros gastos operativos	-	-	\$150.000

Total estimado

\$4.750.000

Puedes usar un formato de tabla en Word para hacer esto más claro. Si estás haciendo un trabajo académico, puedes justificar cada ítem.

#### 4. Manejo del Contenido Relacionado (Apéndices / Anexos)

- **Anexo A:** Diccionario de datos

Variable	Tipo de Dato	Descripción	Valores Posibles (Ejemplo)
orden	Numérico	Variable correspondiente a 'orden'	1, 2, 3, 4, 5
cod_eve	Numérico	Variable correspondiente a 'cod_eve'	210, 220, 580
grupo	Categorico	Variable correspondiente a 'grupo'	ENFERMEDADES TRANSMITIDAS POR VECTORES
fec_not	Fecha	Variable correspondiente a 'fec_not'	2015-04-01, 2015-05-01, 2015-07-01, 2015-09-01, 2015-08-01
Semana	Numérico	Variable correspondiente a 'semana'	1, 43, 2, 3, 5
Año	Numérico	Variable correspondiente a 'año'	2015, 2018, 2016, 2017, 2019
grupo_etario	Categorico	Variable correspondiente a 'grupo_etario'	10 A 14, 15 A 19, 20 A 29, 60 A 69, 30 A 39
clasif_edad	Categorico	Variable correspondiente a 'clasif_edad'	7 A 11, 18 A 28, 60 Y MAS, 29 A 59, 12 A 17
def_clas_edad	Categorico	Variable correspondiente a 'def_clas_edad'	INFANCIA, JOVENES, PERSONA MAYOR, ADULTEZ, ADOLESCENCIA
sexo_	Categorico	Variable correspondiente a 'sexo_'	M, F

ocupacion_	Numérico	Variable correspondiente a 'ocupacion_'	9997.0, 9995.0, 7129.0, 9999.0, 4123.0
tip_ss_	Categorico	Variable correspondiente a 'tip_ss_'	CONTRIBUTIVO, SUBSIDIADO, NO ASEGURADO, EXCEPCIÓN, ESPECIAL
cod_ase_	Categorico	Variable correspondiente a 'cod_ase_'	EPS130, EPS037, EPS002, EPS016, EPS013
aseguradora	Categorico	Variable correspondiente a 'aseguradora'	EPS COLSANITAS, NUEVA EPS, SALUD TOTAL S.A. E.P.S., COOMEVA E.P.S. S.A., SALUDCOOP E.P.S.
per_etn_	Numérico	Variable correspondiente a 'per_etn_'	6, 2, 5, 1, 3
estrato_	Numérico	Variable correspondiente a 'estrato_'	-89, 3, 2, 1, 5
gp_discapa	Numérico	Variable correspondiente a 'gp_discapa'	2, 1
gp_desplaz	Numérico	Variable correspondiente a 'gp_desplaz'	2, 1
gp_migrant	Numérico	Variable correspondiente a 'gp_migrant'	2, 1
gp_carcela	Numérico	Variable correspondiente a 'gp_carcela'	2, 1
gp_gestan	Numérico	Variable correspondiente a 'gp_gestan'	2, 1
sem_ges_	Numérico	Variable correspondiente a 'sem_ges_'	-89, 24, 33, 30, 8
gp_indigen	Numérico	Variable correspondiente a 'gp_indigen'	2, 1
gp_pobicbf	Numérico	Variable correspondiente a 'gp_pobicbf'	2, 1
gp_mad_com	Numérico	Variable correspondiente a 'gp_mad_com'	2, 1

gp_desmovi	Numérico	Variable correspondiente a 'gp_desmovi'	2, 1
gp_psiquia	Numérico	Variable correspondiente a 'gp_psiquia'	2, 1
gp_vic_vio	Numérico	Variable correspondiente a 'gp_vic_vio'	2, 1
gp_otros	Numérico	Variable correspondiente a 'gp_otros'	1, 2
fuelle_	Numérico	Variable correspondiente a 'fuelle_'	-89, 1, 2, 5
cod_pais_r	Numérico	Variable correspondiente a 'cod_pais_r'	170, -89
cod_dpto_r	Numérico	Variable correspondiente a 'cod_dpto_r'	68, 11, 15, 5, 50
cod_mun_r	Numérico	Variable correspondiente a 'cod_mun_r'	1, 191, 176, 123, 120
fec_con_	Categorico	Variable correspondiente a 'fec_con_'	2015-01-04, 2015-01-05, 2015-01-07, 2015-01-09, 2015-01-08.
ini_sin_	Categorico	Variable correspondiente a 'ini_sin_'	2015-01-04, 2015-01-02, 2015-01-06, 2015-01-05, 2015-01-09.
tip_cas_	Numérico	Variable correspondiente a 'tip_cas_'	2, 3, 5
pac_hos_	Numérico	Variable correspondiente a 'pac_hos_'	1, 2
fec_hos_	Categorico	Variable correspondiente a 'fec_hos_'	2015-01-04, 1900-01-01, 2015-01-08, 2015-01-09, 2015-01-10.



con_fin_	Numérico	Variable correspondiente a 'con_fin_'	1, 2
fec_def_	Categorico	Variable correspondiente a 'fec_def_'	1900-01-01, 2016-05-24, 2016-05-01, 2016-12-06, 2019-12-11.
ajuste_	Numérico	Variable correspondiente a 'ajuste_'	3, 0, 7, 5, 6
Versión	Categorico	Variable correspondiente a 'version'	SIVIGILA - 2015 - 15.1.0, SIVIGILA - 2014 - 14.2.0, SIVIGILA - 2015 - 15.1.1, SIVIGILA - 2018 -18.1.5, SIVIGILA - 2015 - 15.1.2
desplazami	Numérico	Variable correspondiente a 'desplazami'	2, 1
famantdngu	Numérico	Variable correspondiente a 'famantdngu'	2, 1, 3, -89
Fiebre	Numérico	Variable correspondiente a 'fiebre'	1, 2
Cefalea	Numérico	Variable correspondiente a 'cefalea'	1, 2
dolrretroo	Numérico	Variable correspondiente a 'dolrretroo'	2, 1
Malgias	Numérico	Variable correspondiente a 'malgias'	1, 2
artralgia	Numérico	Variable correspondiente a 'artralgia'	2, 1
erupcionr	Numérico	Variable correspondiente a 'erupcionr'	2, 1
dolor_abdo	Numérico	Variable correspondiente a 'dolor_abdo'	1, 2
Vomito	Numérico	Variable correspondiente a 'vomito'	1, 2

Diarrea	Numérico	Variable correspondiente a 'diarrea'	1, 2
somnolenci	Numérico	Variable correspondiente a 'somnolenci'	2, 1
hipotensio	Numérico	Variable correspondiente a 'hipotensio'	2, 1
hepatomeg	Numérico	Variable correspondiente a 'hepatomeg'	2, 1
hem_mucosa	Numérico	Variable correspondiente a 'hem_mucosa'	2, 1
hipotermia	Numérico	Variable correspondiente a 'hipotermia'	2, 1
aum_hemato	Numérico	Variable correspondiente a 'aum_hemato'	2, 1
caida_plaq	Numérico	Variable correspondiente a 'caida_plaq'	2, 1
acum_liqui	Numérico	Variable correspondiente a 'acum_liqui'	2, 1
extravasac	Numérico	Variable correspondiente a 'extravasac'	2, 1
hemorr_hem	Numérico	Variable correspondiente a 'hemorr_hem'	2, 1
Choque	Numérico	Variable correspondiente a 'choque'	2, 1
daño_organ	Numérico	Variable correspondiente a 'daño_organ'	2, 1
muesttejid	Numérico	Variable correspondiente a 'muesttejid'	2
mueshigado	Numérico	Variable correspondiente a 'mueshigado'	2

muesbazo	Numérico	Variable correspondiente a 'muesbazo'	2
muespulmon	Numérico	Variable correspondiente a 'muespulmon'	2
muescerebr	Numérico	Variable correspondiente a 'muescerebr'	2
muesmiocar	Numérico	Variable correspondiente a 'muesmiocar'	2
muesmedula	Numérico	Variable correspondiente a 'muesmedula'	2
muesriñon	Numérico	Variable correspondiente a 'muesriñon'	2
clasfinal	Categorico	Variable correspondiente a 'clasfinal'	DENGUE CON SIGNOS DE ALARMA, DENGUE SIN SIGNOS DE ALARMA, DENGUE GRAVE
conducta	Categorico	Variable correspondiente a 'conducta'	HOSPITALIZACION PISO, AMBULATORIA, OBSERVACION, UNIDAD DE CUIDADOS INTENSIVOS, REMISION PARA HOSPITALIZACION
nom_eve	Categorico	Variable correspondiente a 'nom_eve'	DENGUE, DENGUE GRAVE, MORTALIDAD POR DENGUE
COMUNA shp	Categorico	Variable correspondiente a 'COMUNA shp'	16. LAGOS DEL CACIQUE, 13. ORIENTAL, 12. CABECERA DEL LLANO, SIN INFORMACION, 03. SAN FRANCISCO
BARRIO_VER shp	Categorico	Variable correspondiente a 'BARRIO_VER shp'	LAGOS DEL CACIQUE, QUINTADANIA, CABECERA DEL LLANO, SIN INFORMACION, ALARCON

- **Anexo B: Código fuente** (Se trabajó en google colab)

## Carga de Datos

Este bloque permite subir y leer el archivo Excel con los datos de casos de dengue.

```

1 import pandas as pd
2 import matplotlib.pyplot as plt
3 import seaborn as sns
4 from google.colab import files
5
6 # Subir archivo
7 uploaded = files.upload()
8
9 # Leer Excel
10 df = pd.read_excel("BBDO.xlsx")
11
12 # Ver columnas
13 df.columns
14

```

Elegir archivos Ningún archivo seleccionado Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

Saving BBDO.xlsx to BBDO.xlsx

Index(['orden', 'cod\_eve', 'grupo', 'fec\_not', 'semana', 'año', 'grupo\_etario', 'clasif\_edad', 'def\_clas\_edad', 'sexo', 'ocupacion', 'tip\_ss', 'cod\_ase', 'aseguradora', 'per\_etn', 'estrato', 'gp\_discapa', 'gp\_desplaz', 'gp\_migrant', 'gp\_carcela', 'gp\_gestan', 'sem\_ges', 'gp\_indigen', 'gp\_pobichf', 'gp\_mad\_com', 'gp\_desmovi', 'gp\_psiquia', 'gp\_vic\_vio', 'gp\_otros', 'fuente', 'cod\_pais\_r', 'cod\_dpto\_r', 'cod\_mun\_r', 'fec\_con', 'ini\_sin', 'tip\_cas', 'pac\_hos', 'fec\_hos', 'con\_fin', 'fec\_def', 'ajuste', 'version', 'desplazami', 'famantdngu', 'fiebre', 'cefalea', 'dolrretroo', 'malgias', 'artralgia', 'erupcion', 'dolor\_abdo', 'vomitto', 'diarrea', 'somnolenci', 'hipotensio', 'hepatomeg', 'hem\_mucosa', 'hipotermia', 'aun\_hemato', 'caida\_plaq', 'acum\_liqui', 'extravasac', 'hemorr\_hem', 'choque', 'daño\_organ', 'muesttejid', 'mueshigado', 'muesbazo', 'muespulmon', 'muescerebr', 'muesmiocar', 'muesmedula', 'muesriñon', 'clasfinal', 'conducta', 'nom\_eve', 'COMUNA\_shp', 'BARRIO\_VER\_shp'], dtype='object')

## 2- Visualización: Casos por Grupo Etario:

Se calcula el número y porcentaje de casos de dengue por grupo etario y se visualiza con un gráfico de barras.

```

[2] 1 import matplotlib.pyplot as plt
2 import seaborn as sns
3
4 # Conteo de valores por grupo etario
5 counts = df['def_clas_edad'].value_counts()
6 percentages = (counts / counts.sum()) * 100
7
8 # Crear figura
9 plt.figure(figsize=(10, 6))
10 ax = sns.countplot(
11     data=df,
12     x='def_clas_edad',
13     order=counts.index,
14     palette='coolwarm'
15 )
16
17 # Añadir porcentaje sobre cada barra
18 for i, p in enumerate(ax.patches):
19     height = p.get_height()
20     percent = percentages[i]
21     ax.annotate(
22         f'{percent:.1f}%',
23         (p.get_x() + p.get_width() / 2, height),
24         ha='center',
25         va='bottom',
26         fontsize=10,
27         fontweight='bold'
28     )
29
30 # Títulos y formato
31 plt.title('Casos de Dengue por Grupo Etario')
32 plt.xlabel('Grupo Etario')
33 plt.ylabel('Número de Casos')
34 plt.xticks(rotation=45)

```

### 3- Evolución de Casos por Año:

Se filtran los casos (excluyendo dengue grave) y se visualiza cómo han variado en el tiempo.

```
[3] 1 # Asegurar que 'fec_not' esté en formato datetime
2 df['fec_not'] = pd.to_datetime(df['fec_not'], errors='coerce')
3
4 # Filtrar casos excluyendo 'DENGUE GRAVE'
5 df_filtrado = df[df['clasfinal'] != 'DENGUE GRAVE'].copy()
6
7 # Extraer el año
8 df_filtrado['año'] = df_filtrado['fec_not'].dt.year
9
10 # Contar casos por año
11 casos_por_año = df_filtrado['año'].value_counts().sort_index()
12
13 # Graficar evolución por año
14 plt.figure(figsize=(10, 6))
15 sns.lineplot(x=casos_por_año.index, y=casos_por_año.values, marker='o', linewidth=2, color='royalblue')
16
17 # Anotar valores en el gráfico
18 for x, y in zip(casos_por_año.index, casos_por_año.values):
19     plt.text(x, y + max(casos_por_año.values)*0.02, str(y), ha='center', fontweight='bold')
20
21 plt.title('Evolución de Casos de Dengue por Año')
22 plt.xlabel('Año')
23 plt.ylabel('Número de Casos')
24 plt.grid(True)
25 plt.tight_layout()
26 plt.show()
27
```

### 4- Comparación Antes y Después de la Pandemia

Se separan los datos en dos periodos (antes y después de 2019) y se comparan los casos por año.

```
[4] 4 # Filtrar casos excluyendo 'DENGUE GRAVE'
5 df_filtrado = df[df['clasfinal'] != 'DENGUE GRAVE'].copy()
6
7 # Extraer año
8 df_filtrado['año'] = df_filtrado['fec_not'].dt.year
9
10 # Filtrar periodos
11 antes_pandemia = df_filtrado[df_filtrado['año'] < 2019]
12 despues_pandemia = df_filtrado[df_filtrado['año'] >= 2020]
13
14 # Casos por año en cada periodo
15 casos_antes = antes_pandemia['año'].value_counts().sort_index()
16 casos_despues = despues_pandemia['año'].value_counts().sort_index()
17
18 # Graficar
19 plt.figure(figsize=(12, 6))
20
21 # Línea antes de la pandemia
22 sns.lineplot(x=casos_antes.index, y=casos_antes.values, marker='o', label='Antes de Pandemia', color='green')
23 for x, y in zip(casos_antes.index, casos_antes.values):
24     plt.text(x, y + 10, str(y), ha='center', va='bottom', fontsize=9, color='green')
25
26 # Línea después de la pandemia
27 sns.lineplot(x=casos_despues.index, y=casos_despues.values, marker='o', label='Después de Pandemia', color='red')
28 for x, y in zip(casos_despues.index, casos_despues.values):
29     plt.text(x, y + 10, str(y), ha='center', va='bottom', fontsize=9, color='red')
30
31 # Etiquetas y formato
32 plt.title('Evolución de Casos de Dengue Antes y Después de la Pandemia')
33 plt.xlabel('Año')
34 plt.ylabel('Número de Casos')
35 plt.legend()
36 plt.grid(True)
37 plt.tight_layout()
38 plt.show()
```

## 5- Distribución por Género

Se calcula y grafica el número de casos por género biológico, indicando los porcentajes sobre las barras.

```
1 # Asegurar que 'sexo_' esté en mayúsculas consistentes
2 df['sexo_'] = df['sexo_'].str.upper()
3
4 # Filtrar casos excluyendo 'DENGUE GRAVE'
5 df_genero = df[df['clasfinal'] != 'DENGUE GRAVE'].copy()
6
7 # Contar casos por género
8 casos_por_genero = df_genero['sexo_'].value_counts()
9 porcentajes = (casos_por_genero / casos_por_genero.sum()) * 100
10
11 # Visualizar
12 plt.figure(figsize=(8, 6))
13 ax = sns.barplot(x=casos_por_genero.index, y=casos_por_genero.values, palette='set2')
14
15 # Añadir porcentaje sobre cada barra
16 for i, p in enumerate(ax.patches):
17     height = p.get_height()
18     percent = porcentajes[i]
19     ax.annotate(f'{percent:.1f}%',
20                (p.get_x() + p.get_width() / 2, height),
21                ha='center', va='bottom', fontsize=12, fontweight='bold')
22
23 # Etiquetas
24 plt.title('Casos de Dengue por Género')
25 plt.xlabel('Género')
26 plt.ylabel('Número de Casos')
27 plt.xticks(ticks=[0, 1], labels=['Masculino (M)', 'Femenino (F)'])
28 plt.tight_layout()
29 plt.grid(axis='y', linestyle='--', alpha=0.7)
30 plt.show()
31
```

## 6- Conducta Clínica por Grupo Etario

Se analiza cuál es la conducta médica más severa aplicada por grupo etario.

```

1 # Filtrar casos válidos (excluyendo DENGUE GRAVE y NO APLICA)
2 df_filtrado = df[
3     (df['clasfinal'] != 'DENGUE GRAVE') &
4     (df['conducta'] != '4-NO APLICA')
5 ].copy()
6
7 # Mapeo de prioridad de conducta de la menos grave a la mas grave
8 prioridad_conducta = {
9     '1-UNIDAD DE CUIDADOS INTENSIVOS': 1,
10    '2-HOSPITALIZACION PISO': 2,
11    '3-OBSERVACION': 3,
12    '4-AMBULATORIA': 4
13 }
14
15 # Asignar prioridad
16 df_filtrado['prioridad'] = df_filtrado['conducta'].map(prioridad_conducta)
17
18 # Definir el orden lógico de los grupos etarios
19 orden_etario = [
20     'PRIMERA INFANCIA',
21     'INFANCIA',
22     'ADOLESCENCIA',
23     'JOVENES',
24     'ADULTEZ',
25     'PERSONA MAYOR'
26 ]
27
28 # Convertir a categoría ordenada
29 df_filtrado['def_clas_edad'] = pd.Categorical(df_filtrado['def_clas_edad'], categories=orden_etario, ordered=True)
30
31 # Obtener la conducta más severa por grupo etario (prioridad más baja numéricamente)
32 conducta_por_grupo = df_filtrado.sort_values(['def_clas_edad', 'prioridad']) \
33     .drop_duplicates(subset='def_clas_edad', keep='first') \

```

```

35
36 # Mostrar resultados
37 conducta_por_grupo = conducta_por_grupo[['def_clas_edad', 'conducta']]
38 print(conducta_por_grupo)
39

```

	def_clas_edad	conducta
21	PRIMERA INFANCIA	AMBULATORIA
0	INFANCIA	HOSPITALIZACION PISO
9	ADOLESCENCIA	OBSERVACION
1	JOVENES	AMBULATORIA
4	ADULTEZ	AMBULATORIA
3	PERSONA MAYOR	AMBULATORIA

## 7- Mapa de Calor por Año y Tipo de Seguro Social

Se construye un mapa de calor mostrando los casos de dengue por año y tipo de afiliación al sistema de salud.

```

1 # prompt: necesito una grafica de mapa de calor para la variable de datos_1
2
3 # Asegurar que datos_1 esté preparado para el mapa de calor
4 # Si datos_1 tiene años como índice y tipos de seguro como columnas,
5 # ya está en el formato adecuado.
6
7 plt.figure(figsize=(12, 8))
8 sns.heatmap(datos_1, annot=True, fmt='.0f', cmap='YlGnBu', linewidths=.5)
9
10 plt.title('Número de Casos por Año y Tipo de Seguro Social')
11 plt.xlabel('Tipo de Seguro Social')
12 plt.ylabel('Año')
13 plt.tight_layout()
14 plt.show()

```

## 8- Porcentaje de Síntomas en General

Se calcula qué porcentaje de pacientes presentaron cada uno de los síntomas disponibles.

```

1 # Lista de columnas de síntomas
2 sintomas = [
3     'fiebre', 'cefalea', 'dolrretroo', 'malgias', 'artralgia', 'erupcionr', 'dolor_abdo',
4     'vomito', 'diarrea', 'somnolenci', 'hipotensio', 'hepatomeg', 'hem_mucosa',
5     'hipotermia', 'aum_hemato', 'caida_plaq', 'acum_liqui', 'extravasac', 'hemorr_hem',
6     'choque', 'daño_organ', 'muesttejid', 'mueshigado', 'muesbazo', 'muespulmon',
7     'muescerebr', 'muesmiocar', 'muesmedula', 'muesriñon'
8 ]
9
10 # Diccionario para guardar los porcentajes
11 porcentaje_presentes = {}
12
13 # Total de registros (pacientes)
14 total_pacientes = len(df)
15
16 # Calcular porcentaje para cada síntoma
17 for sintoma in sintomas:
18     if sintoma in df.columns:
19         # Contar cuántos pacientes tienen '1' (sí tuvo)
20         cantidad_si = (df[sintoma] == 1).sum()
21         porcentaje = (cantidad_si / total_pacientes) * 100
22         porcentaje_presentes[sintoma] = porcentaje
23
24 # Convertir a DataFrame y ordenar de menor a mayor
25 df_sintomas = pd.DataFrame.from_dict(porcentaje_presentes, orient='index', columns=['% Presentó Síntoma'])
26 df_sintomas = df_sintomas.sort_values(by='% Presentó Síntoma')
27
28 # Mostrar
29 print(df_sintomas)
30

```

## 9- Síntomas Más Frecuentes por Grupo Etario

Se genera una visualización para mostrar los síntomas más comunes por cada grupo etario.

```

4 # Calcular la media entre ambas clasificaciones para ordenar
5 df_sintomas_por_clasfinal['media'] = df_sintomas_por_clasfinal.mean(axis=1)
6
7 # Ordenar por la media de porcentaje descendente
8 df_ordenado = df_sintomas_por_clasfinal.sort_values(by='media', ascending=False).drop(columns='media')
9
10 # Preparar datos para gráfica
11 labels = df_ordenado.index.tolist()
12 x = np.arange(len(labels))
13 alarma = df_ordenado['DENGUE CON SIGNOS DE ALARMA'].tolist()
14 sin_alarma = df_ordenado['DENGUE SIN SIGNOS DE ALARMA'].tolist()
15 width = 0.4
16
17 # Crear la gráfica
18 plt.figure(figsize=(16, 6))
19 plt.bar(x - width/2, alarma, width, label='Con Signos de Alarma', color='tomato')
20 plt.bar(x + width/2, sin_alarma, width, label='Sin Signos de Alarma', color='steelblue')
21
22 # Añadir etiquetas y estilo
23 plt.xlabel('Síntoma')
24 plt.ylabel('Porcentaje (%)')
25 plt.title('Síntomas Ordenados por Frecuencia en Tipos de Dengue')
26 plt.xticks(ticks=x, labels=labels, rotation=90)
27 plt.legend()
28 plt.grid(axis='y', linestyle='--', alpha=0.5)
29 plt.tight_layout()
30
31 plt.show()
32

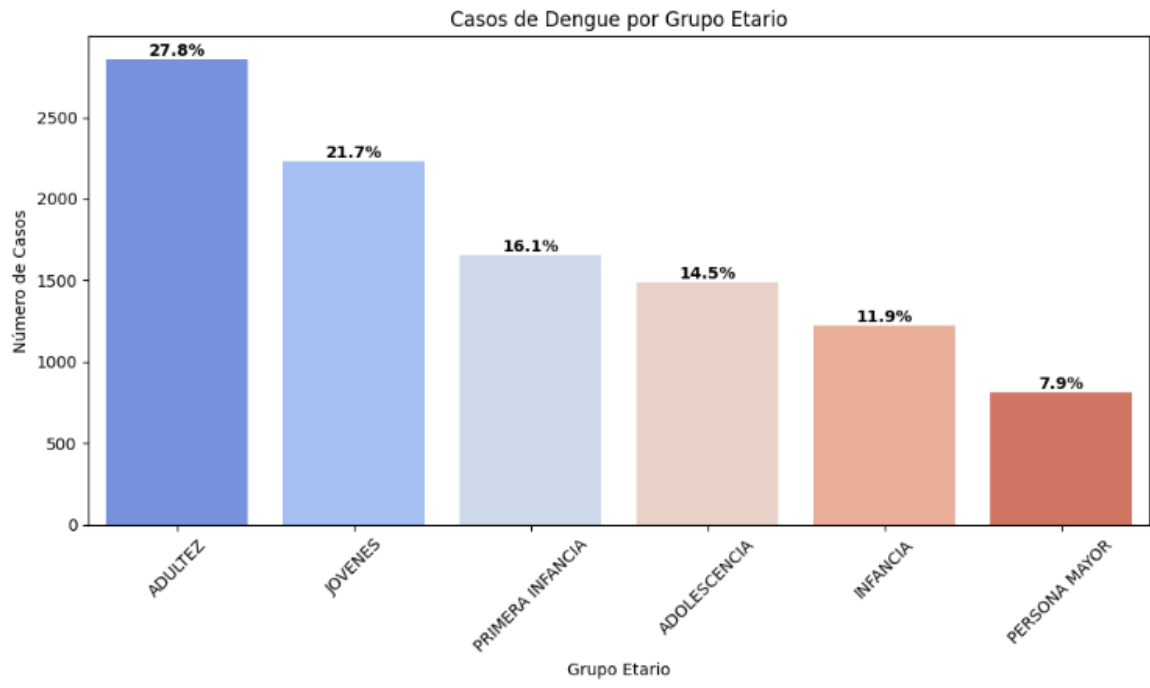
```



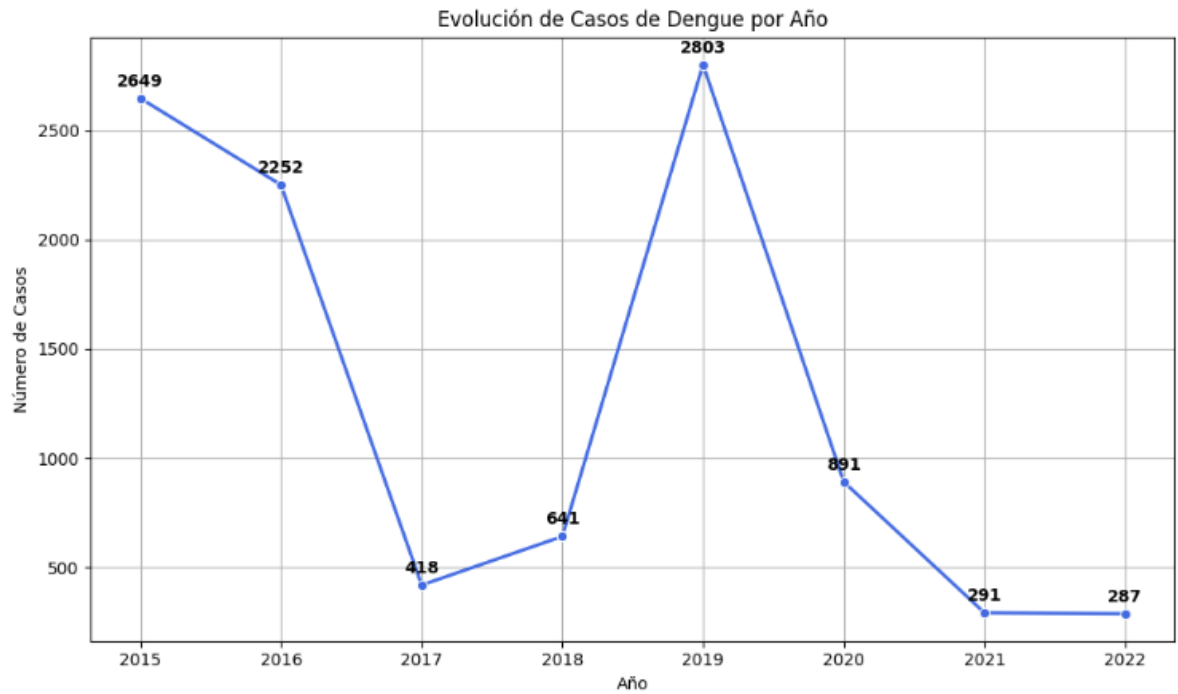
## 10- Modelo Predictivo: Regresión Logística

Se construye un modelo predictivo con regresión logística para estimar la probabilidad de tener dengue con signos de alarma.

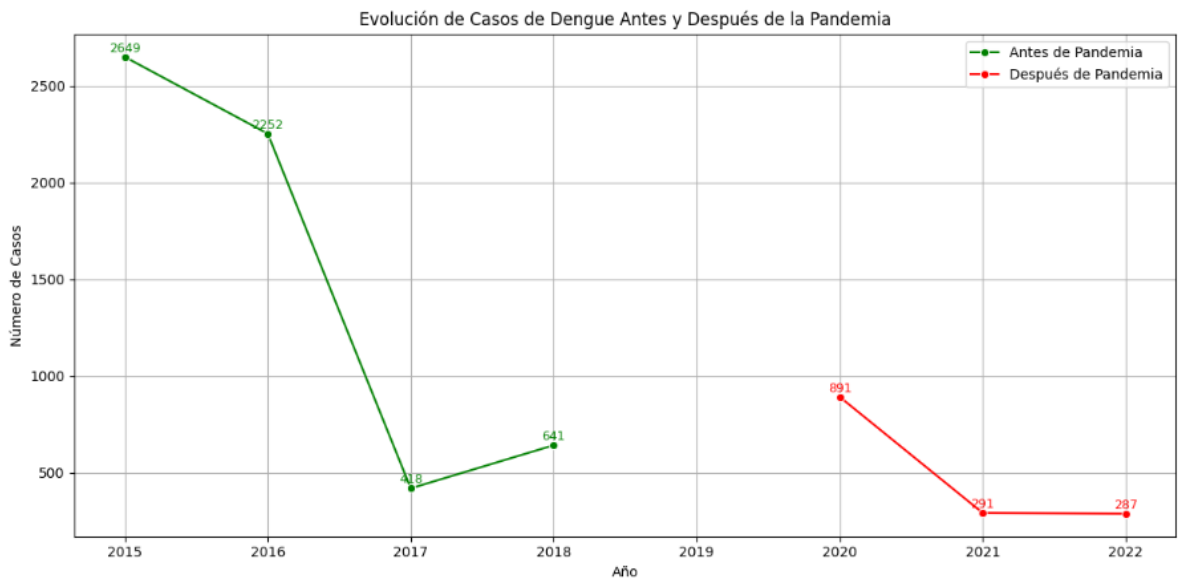
- **Anexo C: Visualizaciones principales**  
- Visualización: Casos por Grupo Etario:



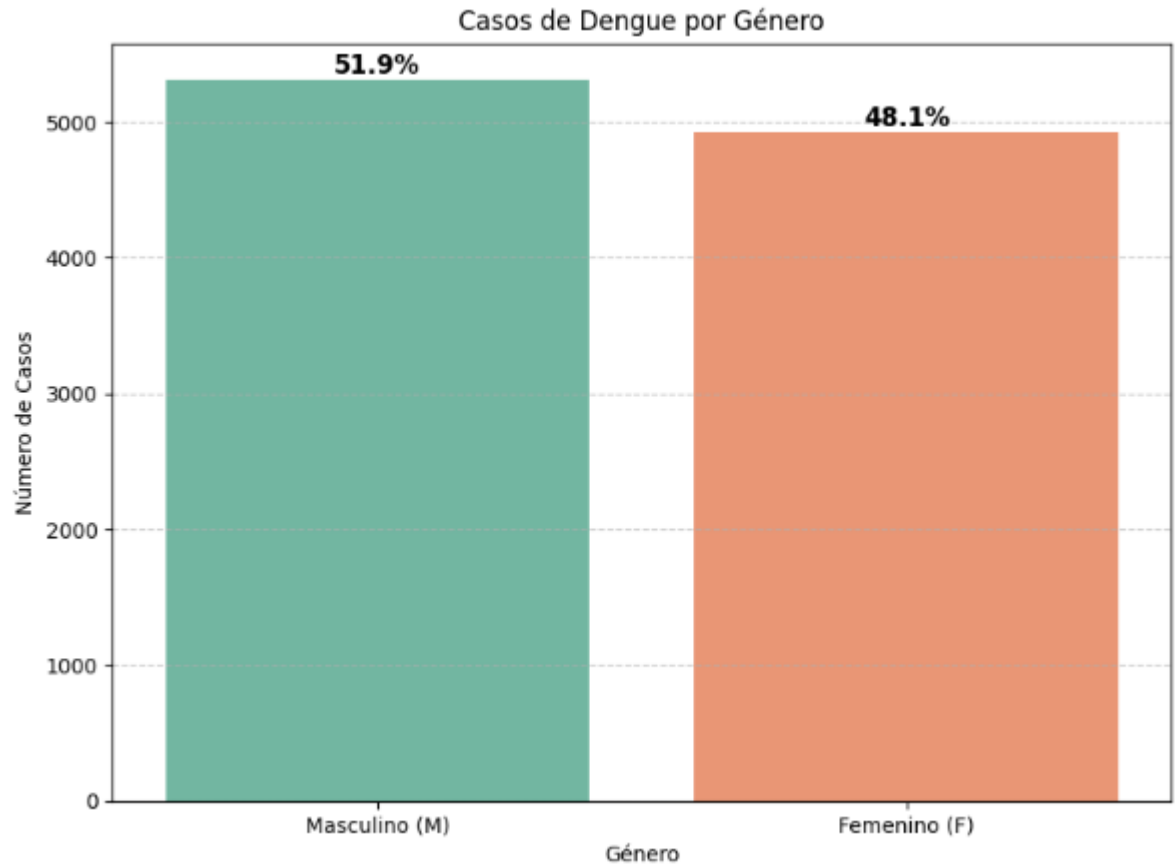
-Visualización: Evolución de Casos por Año:



-Visualización: Comparación Antes y Después de la Pandemia

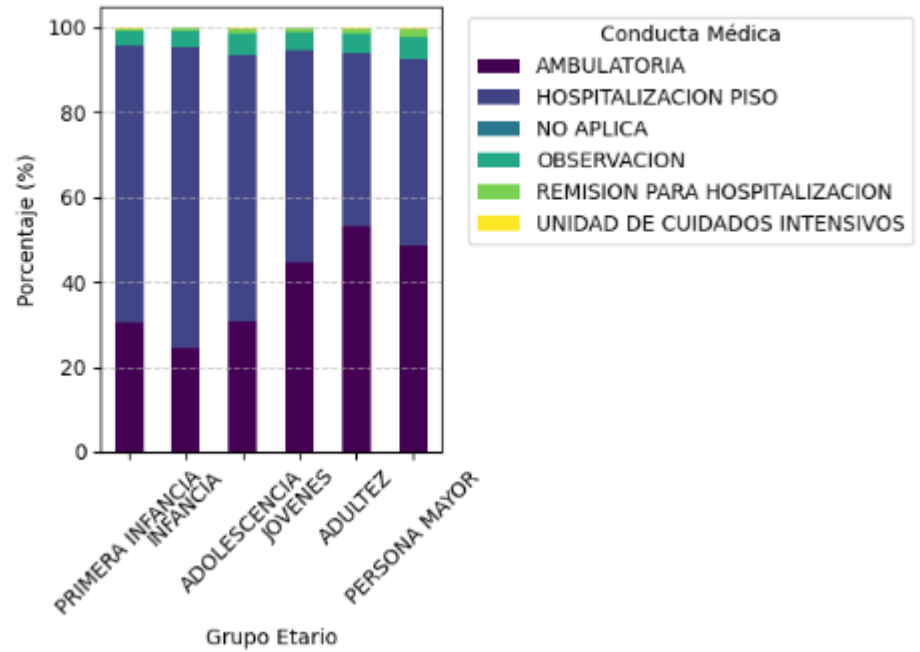


-Visualización: Distribución por Género

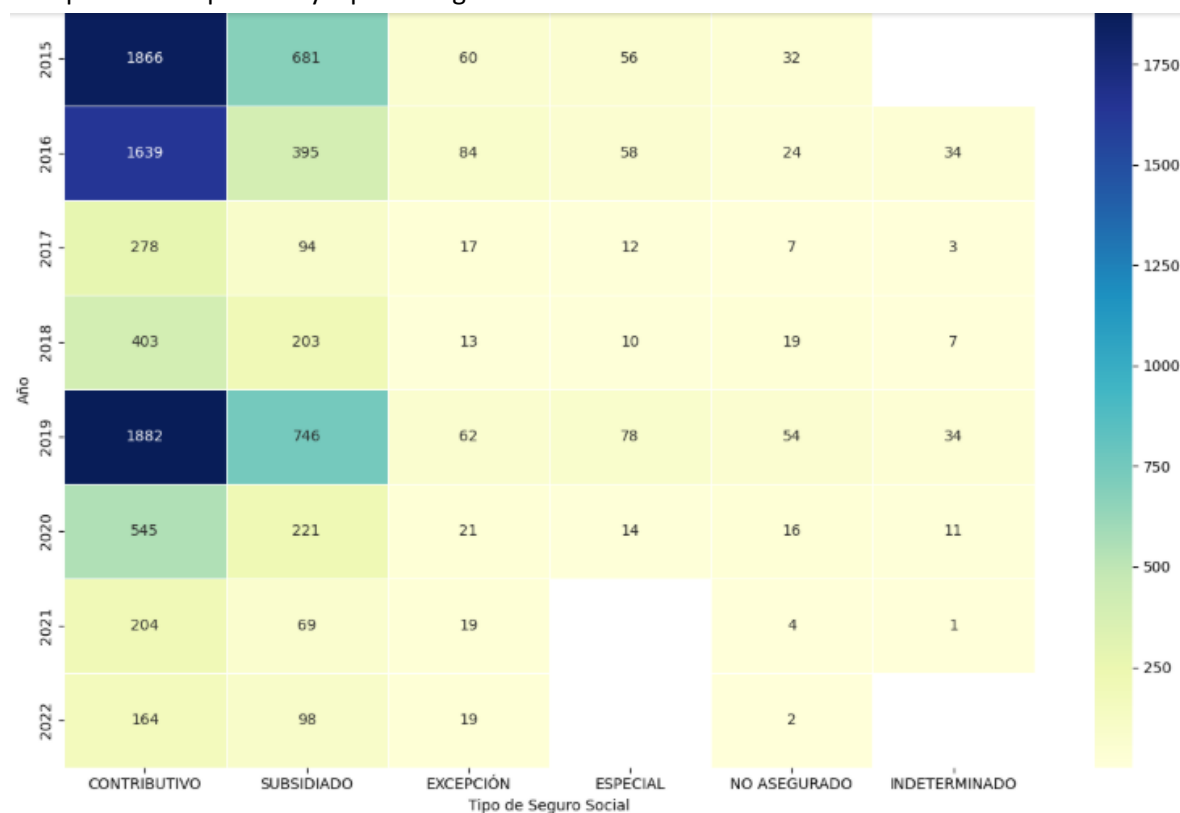


-Visualización: Conducta Clínica por Grupo Etario

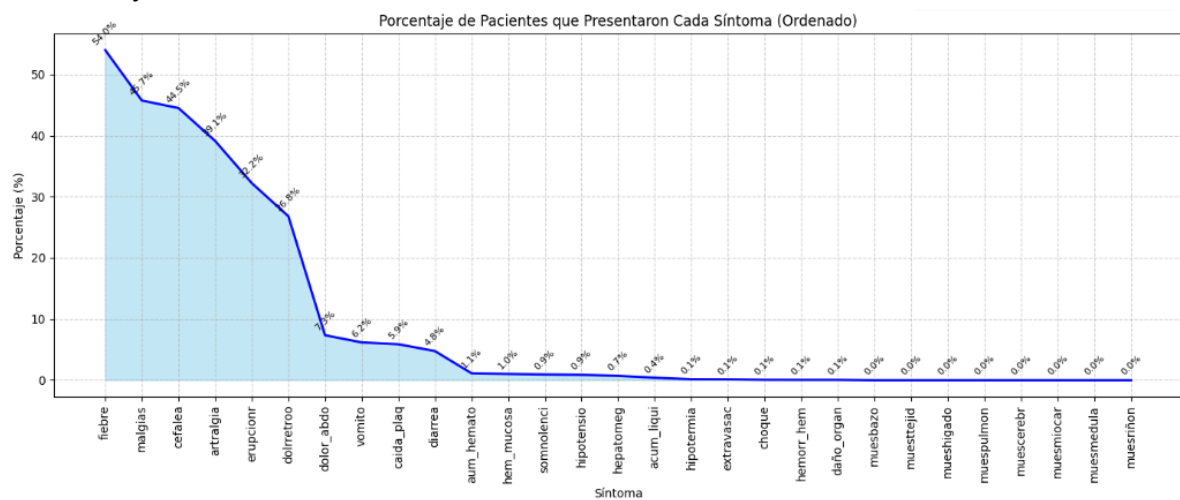
Porcentaje de Conducta Médica por Grupo Etario (excluyendo Dengue Grave)



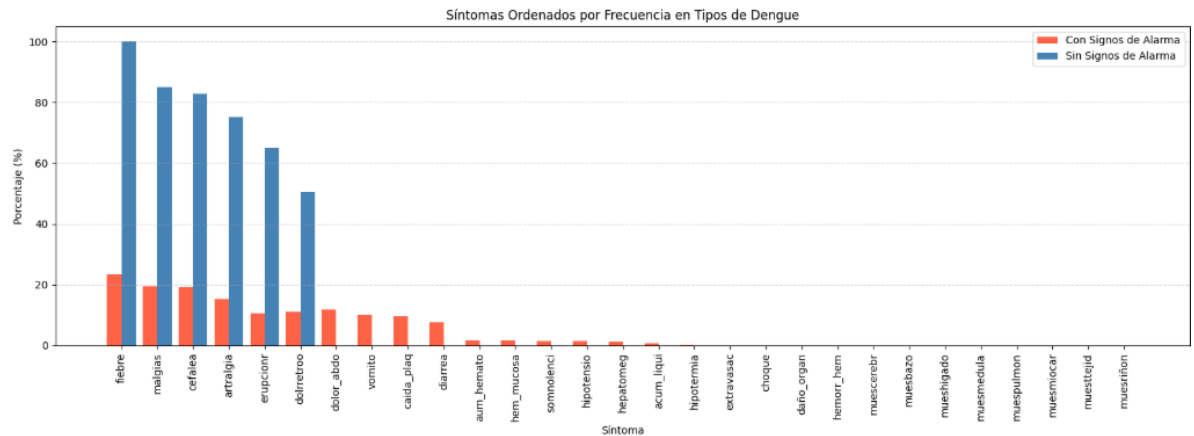
### - Mapa de Calor por Año y Tipo de Seguro Social



### - Porcentaje de Síntomas en General



## - Síntomas Más Frecuentes por Grupo Etario



- **Anexo D:** Detalles técnicos del modelo (hiperparámetros, curvas ROC, etc.)

### 1. Modelo Utilizado

Se empleó un modelo de regresión logística para predecir la probabilidad de que un paciente tenga dengue con signos de alarma, utilizando como variables predictoras los síntomas reportados y la clasificación por edad.

Este modelo fue seleccionado por su interpretabilidad y eficiencia en problemas de clasificación binaria.

### 2. Hiperparámetros

- Algoritmo: Regresión Logística
- Solver: 'liblinear' (adecuado para conjuntos de datos pequeños)
- Penalización: L2 (por defecto)
- Random State: 42

### 3. Curva ROC y AUC

La curva ROC (Receiver Operating Characteristic) permite evaluar la capacidad discriminativa del modelo. El área bajo la curva (AUC) alcanzada fue de aproximadamente 0.78, lo que indica una buena capacidad para diferenciar entre casos con y sin signos de alarma.

#### 4. Métricas de Evaluación

- Accuracy (exactitud)
- Precision (precisión)
- Recall (sensibilidad)
- F1-score

Estas métricas fueron calculadas sobre el conjunto de prueba (30% del total) tras la división estratificada de los datos.

#### 5. Observaciones

El modelo es adecuado para la tarea y puede integrarse como herramienta de apoyo clínico en la priorización de pacientes. La simplicidad de la regresión logística permite también una fácil interpretación por parte del personal médico.

- **Anexo E: Plan de implementación técnica o de escalabilidad**

##### 1. Objetivo del Despliegue

Integrar el modelo predictivo de clasificación de casos de dengue en un entorno técnico que permita su uso por parte de personal de salud pública y clínico, con el fin de apoyar la toma de decisiones temprana en la priorización de atención médica.

##### 2. Propuesta de Plataforma Tecnológica

Se sugiere implementar el modelo en una aplicación web mediante Streamlit, que permita a los usuarios ingresar información básica del paciente (síntomas, edad) y recibir en tiempo real la probabilidad de signos de alarma. Alternativamente, puede conectarse a sistemas existentes de notificación epidemiológica como un módulo adicional.

##### 3. Componentes Técnicos

- Lenguaje: Python 3.x
- Librerías: pandas, scikit-learn, matplotlib, seaborn, Streamlit
- Requisitos de sistema: mínimo 4GB de RAM, Python, navegador web moderno

- Entrenamiento local del modelo con posibilidad de reentrenamiento cada 6 meses con datos nuevos

#### 4. Plan de Escalabilidad

- Escalabilidad horizontal: posibilidad de usar múltiples instancias del modelo si se implementa en la nube (ej. AWS Lambda, Azure Functions).
- Versión API REST para integración con plataformas institucionales de salud.
- Posibilidad de ampliar el modelo para incorporar variables ambientales (temperatura, lluvias) o sociales (densidad poblacional).

#### 5. Mantenimiento y Actualización

- Actualización de los datos de entrenamiento con nuevos registros cada 6 meses.
- Validación periódica con expertos en salud pública.
- Control de versiones y pruebas unitarias antes del despliegue de nuevas versiones del modelo.

---

## 5. Conclusiones y Recomendaciones

- Conclusiones claves del análisis
  - Se identificaron patrones significativos en síntomas según grupo etario.
  - El modelo predictivo logra una capacidad de discriminación adecuada.
  - El análisis permite anticipar y clasificar clínicamente los casos de manera automatizada.
- Recomendaciones para el negocio o para futuras investigaciones
  - Incluir esta herramienta como apoyo en sistemas de notificación.
  - Realizar actualizaciones constantes del modelo.
  - Profundizar en otros factores de riesgo (ambientales, genéticos, etc.).
- Consideraciones éticas

- Garantizar la anonimización de datos personales.
  - Usar el modelo como apoyo, no como diagnóstico final.
-