Predicción de Hospitalización por Dengue

Metodología CRISP-DM

José Ronaldo Duran Toloza

Rafael Eduardo Garcia Montes

Curso: Aprendizaje de Máquinas

Docente: John Fernando Vargas Buitrago

28 de febrero de 2026

Índice

1. Entendimiento del negocio

1.1. Descripción del negocio

El dengue es un problema de salud pública que exige decisiones oportunas para priorizar recursos clínicos, en especial la hospitalización de pacientes con mayor riesgo de complicación.

1.2. Descripción del problema

Se requiere un modelo predictivo que estime si un paciente con sospecha o confirmación de dengue es candidato a hospitalización, usando variables clínicas, demográficas y geográficas.

1.3. Objetivos de la minería

- Construir modelos de clasificación supervisada para la variable objetivo `pac_hos_`.
- Optimizar hiperparámetros con validación cruzada para maximizar desempeño.
- Comparar modelos clásicos y ensambles para seleccionar el mejor.

1.4. Diseño de solución

Se implementó un flujo CRISP-DM: comprensión de datos, preparación, modelamiento con hiperparametrización, evaluación en *hold-out* y despliegue básico con Streamlit.

Tipo de análisis	Tipo de aprendizaje	Métodos	Evaluación
Predictivo	Supervisado	Clásicos + Ensambles	F1-macro, recall, precision, accuracy

Cuadro 1: Diseño general del análisis

2. Entendimiento de los datos

2.1. Diccionario de datos

El dataset principal contiene variables demográficas, geográficas y síntomas asociados a dengue. La variable objetivo del problema es `pac_hos_` (1=Sí hospitalización, 2=No).

Variable	Descripción	Tipo	Rol
edad_	Edad del paciente	Numérica	Entrada
comuna	Comuna de residencia	Categórica	Entrada
fiebre	Presencia de fiebre	Categórica	Entrada
cefalea	Presencia de cefalea	Categórica	Entrada
dolor_abdo	Dolor abdominal	Categórica	Entrada
vomito	Vómito	Categórica	Entrada
hipotensio	Hipotensión	Categórica	Entrada

Variable	Descripción	Tipo	Rol
pac_hos_	Paciente hospitalizado (1=Sí, 2=No)	Categórica	Salida

Cuadro 2: Diccionario resumido de variables relevantes

2.2. Reglas de calidad

Se definieron reglas de validación para variables numéricas y categóricas, tomando como base el diccionario oficial y los valores observados en los datos.

Variable numérica	Valor mínimo	Valor máximo
edad_	0	120

Cuadro 3: Reglas de calidad para variables numéricas

Variable categórica	Valores posibles
pac_hos_	1=Sí, 2=No
fiebre	1=Sí, 2=No, SD
cefalea	1=Sí, 2=No, SD
hipotensio	1=Sí, 2=No, SD

Cuadro 4: Reglas de calidad para variables categóricas

3. Preparación de datos

3.1. Selección de variables

Se eliminaron variables con baja contribución y/o redundancia, manteniendo la variable objetivo al final del flujo para facilitar el procesamiento.

3.2. Descripción estadística

Se realizaron inspecciones de tipos, conteos, distribución de clases y porcentaje de nulos por variable.

3.3. Limpieza de atípicos

No se aplicó una eliminación masiva de atípicos por tratarse mayoritariamente de variables categóricas codificadas y binarias. En variables numéricas se verificó consistencia de rangos.

3.4. Limpieza de nulos

Se eliminaron registros con alta proporción de nulos y se imputaron nulos categóricos usando KNN sobre codificación temporal.

3.5. Creación de nuevas variables

No se crearon variables sintéticas adicionales; se aplicó codificación one-hot para variables categóricas.

3.6. Análisis de relaciones

Se revisó comportamiento de la variable objetivo antes y después del balanceo, y relación de síntomas/atributos con la clase.

3.7. Reducción de dimensiones

En la versión actual no se aplicó PCA. La decisión se sustentó en mantener interpretabilidad y en dimensionalidad manejable tras selección de variables.

3.8. Balanceo

Se aplicó SMOTE para balancear las clases de hospitalización y reducir sesgo por desbalance de clase.

4. Modelamiento y evaluación

4.1. Configuración y selección de métodos

Se trabajó con dos familias:

- **Modelos clásicos (5):** Regresión Logística, Árbol de Decisión, KNN, SVM y MLP.
- **Ensembles (3):** VotingClassifier (votación), Random Forest (bagging) y XGBoost (boosting).

4.2. Ajuste de hiperparámetros con 70 % de datos + CV

Se realizó partición estratificada 70/30. El ajuste de hiperparámetros se ejecutó exclusivamente en el 70 % de entrenamiento mediante `GridSearchCV` y validación cruzada estratificada.

4.2.1. Justificación de la métrica

Se utilizó **F1-macro** como métrica principal para balancear desempeño entre clases y no favorecer una clase particular.

4.2.2. Ajuste de modelos clásicos

Se ajustaron hiperparámetros de los 5 modelos clásicos definidos.

4.2.3. Ajuste de modelos de ensamble

Se ajustaron hiperparámetros de VotingClassifier, Random Forest y XGBoost.

4.3. Medida de calidad del modelo

La evaluación final se hizo sobre el 30 % de prueba no usado en tuning.

Modelo	Tipo	F1-macro (CV train)	F1-macro (test)
Regresión Logística	Clásico	—	—
Árbol de Decisión	Clásico	—	—
KNN	Clásico	—	—
SVM	Clásico	—	—
Red Neuronal MLP	Clásico	—	—
VotingClassifier	Ensamble (Votación)	—	—
Random Forest	Ensamble (Bagging)	—	—
XGBoost	Ensamble (Boosting)	—	—

Cuadro 5: Resultados comparativos de modelos

Nota: completar con valores finales exportados del notebook (bloque formal 70/30).

4.3.1. Selección del mejor modelo

El mejor modelo se seleccionó por el mayor valor de `test_f1_macro`. Además, se reportaron recall, precision y accuracy en test.

5. Despliegue

5.1. Predicción de datos futuros

Se generó un conjunto de entrada nuevo y se aplicó el modelo final para predecir hospitalización.

5.2. Construcción del conjunto de entrada

El conjunto de entrada se estructuró con el mismo esquema de variables del modelo entrenado.

5.3. Preparación del conjunto nuevo

Se reindexaron columnas según `model_input_columns.joblib` para garantizar compatibilidad con el pipeline final.

5.4. Predicción e interpretación

Las salidas del modelo se interpretaron como:

- 1: Hospitalización Sí
- 0: Hospitalización No

5.5. Aplicación web con Streamlit

Se implementó una app en `streamlit_app/app.py` para cargar CSV y obtener predicciones.

Pendiente para entrega final: incluir captura de pantalla funcionando y URL pública de Streamlit Cloud.

6. Conclusiones

El proyecto permitió construir un sistema de clasificación para estimar hospitalización por dengue con base en síntomas y variables complementarias.

La comparación sistemática entre modelos clásicos y ensambles permitió seleccionar el mejor enfoque según `F1-macro` en test.

Como trabajo futuro se recomienda:

- ampliar validación temporal por periodos epidemiológicos,
- incluir calibración de probabilidades para soporte clínico,
- monitorear deriva de datos en operación.