

Bogotá D.C., 23 de junio de 2025

REPORTE FASE 3

TRABAJO FINAL DE MAESTRÍA “PRONÓSTICO DE LA PRECIPITACIÓN ACUMULADA EN COLOMBIA, UTILIZANDO TÉCNICAS DE APRENDIZAJE AUTOMÁTICO”

| | |
|--|--|
| NOMBRE: JUAN DIEGO MOGOLLÓN OVIEDO | |
| OBJETIVO: Desarrollar un modelo para el pronóstico de la precipitación acumulada en Colombia, utilizando técnicas de aprendizaje automático. | |
| OBJETIVOS ESPECÍFICOS: <ul style="list-style-type: none">• Seleccionar un conjunto de datos abiertos de precipitación acumulada en Colombia, verificando su acceso y calidad mediante algoritmos de análisis de datos.• Implementar un proceso para la extracción, transformación y carga de los datos, aplicando técnicas de ingeniería de características.• Evaluar el desempeño de diferentes algoritmos de aprendizaje automático para el pronóstico de precipitación acumulada, utilizando los datos procesados y métricas de evaluación empleadas en predicción climática.• Diseñar una arquitectura para la implementación del modelo en un entorno de prueba, aplicando prácticas y herramientas de operaciones de aprendizaje automático. | |
| PLAZO DE EJECUCIÓN: 16 semanas | |
| INICIO: 31 de marzo de 2025 | FINALIZACIÓN: 26 de julio de 2025 |

Con el objetivo de dar cumplimiento al cronograma estipulado en la “Propuesta de Trabajo Final de Maestría” en el periodo 13 de mayo de 2025 al 08 de junio de 2025, se presenta a continuación el reporte de la fase 3:

RESULTADO OBTENIDO:

Se evaluó el desempeño de diferentes algoritmos de aprendizaje automático para el pronóstico de la precipitación acumulada, utilizando datos previamente procesados y métricas comúnmente empleadas en predicción climática.

Se consideraron técnicas ampliamente utilizadas con diferentes arquitecturas, incluyendo modelos basados en árboles de decisión, redes neuronales recurrentes, Transformers y métodos híbridos con descomposición de señales. Los modelos evaluados fueron: XGBoost, LSTM, BiLSTM, Lag-Llama (solo evaluación), CEEMDAN-VMD-BiLSTM y SBiLSTM. Estos dos últimos modelos se fundamentan en los artículos [\[1\]](#) y [\[2\]](#), donde se propone una combinación de técnicas de descomposición (CEEMDAN y VMD) y entrenamiento con retroalimentación mediante una Support Vector Machine (SVM). Por su parte, Lag-Llama es un modelo fundacional de propósito general para la predicción probabilística de series de tiempo de una variable, basado en una arquitectura Transformer tipo decoder-only. Está preentrenado sobre un amplio conjunto de series temporales de distintos dominios y ha demostrado una alta capacidad de generalización zero-shot y few-shot [\[3\]](#).

La evaluación se llevó a cabo utilizando ventanas deslizantes de longitud 32 del dataset de precipitación acumulada mensual, aplicadas a la serie continua más extensa de cada estación meteorológica. Los conjuntos de entrenamiento y prueba se dividieron en proporciones 80/20 respetando el orden temporal, y la predicción se realizó para el valor siguiente a cada ventana. Los modelos entrenados se exportaron en formato json y keras.

Las métricas utilizadas fueron: Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Forecast Mean Error (Bias), Fractional Bias (FB), Nash-Sutcliffe Efficiency (NSE), Forecast Activity (FA), Observation Activity (OA), and Anomaly Correlation (AC) sobre los valores absolutos medidos en mm. La climatología mensual de cada estación meteorológica se descargó desde el portal oficial del IDEAM, correspondiente al periodo de referencia 1991–2020. Además, se empleó un proceso de categorización basado en percentiles de la precipitación observada para facilitar la interpretación de los resultados mediante matrices de confusión. Específicamente, se calcularon los percentiles P20, P50, P80, P95 y P99 sobre los datos observados, y se establecieron umbrales de clasificación utilizando los siguientes intervalos: $[-\infty, 3 \text{ mm}]$, seguido por los valores de los percentiles y un límite superior $(+\infty)$.

La implementación se desarrolló en Python dentro del entorno Jupyter Notebooks. Se crearon los notebooks training.ipynb y evaluation.ipynb.

A continuación, se resumen los resultados de entrenamiento y evaluación para cada modelo:

Tabla 1. Métricas de evaluación de los modelos de pronóstico de precipitación acumulada.

| Modelo | RMSE | MAE | Bias | FB | NSE | FA | OA | AC |
|--------------------|-------------|------------|-------------|-----------|------------|-----------|-----------|-----------|
| XGBoost | 103.608 | 70.041 | 3.231 | 0.016 | 0.651 | 67.725 | 110.874 | 0.410 |
| LSTM | 107.665 | 73.222 | 2.161 | 0.011 | 0.623 | 68.659 | 110.874 | 0.356 |
| BiLSTM | 107.453 | 73.366 | 2.884 | 0.015 | 0.625 | 68.194 | 110.874 | 0.357 |
| SBiLSTM | 107.830 | 73.558 | 0.863 | 0.004 | 0.622 | 66.431 | 110.874 | 0.345 |
| CEEMDAN-VMD-BiLSTM | 153.401 | 109.265 | 1.246 | 0.006 | 0.261 | 130.336 | 113.089 | 0.212 |
| Lag-Llama | 160.492 | 110.485 | -21.609 | -0.115 | 0.202 | 132.202 | 114.306 | 0.174 |

Además, se presentan las gráficas que soportan los resultados mencionados anteriormente:

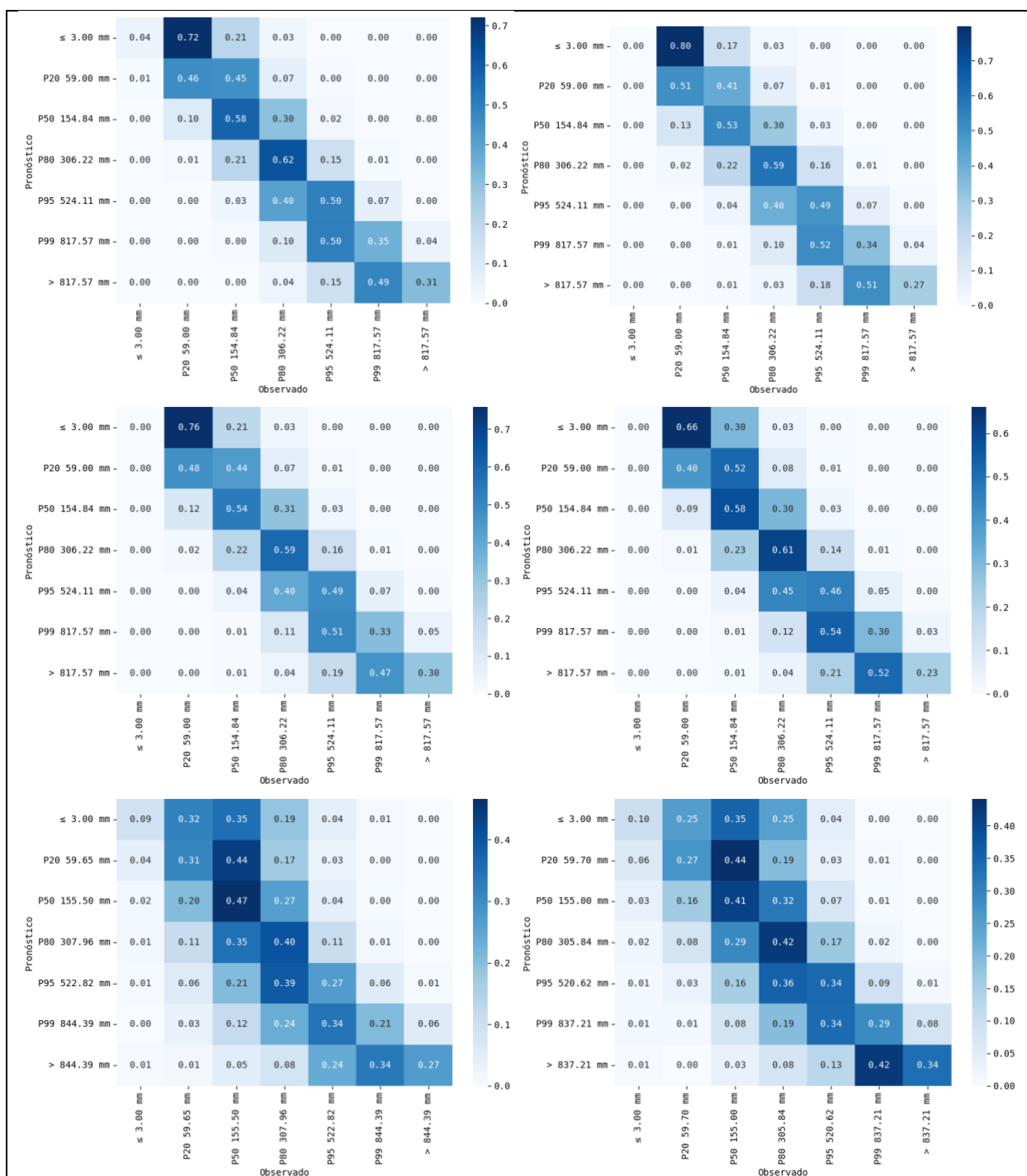


Figura 1. Matrices de confusión normalizadas para los modelos de pronóstico de precipitación acumulada. De izquierda a derecha y de arriba hacia abajo, se presentan los resultados para: (1) XGBoost (fila 1, columna 1), (2) LSTM (fila 1, columna 2), (3) BiLSTM (fila 2, columna 1), (4) SBiLSTM (fila 2, columna 2), (5) CEEMDAN-VMD-BiLSTM (fila 3, columna 1) y (6) Lag-Llama (fila 3, columna 2).

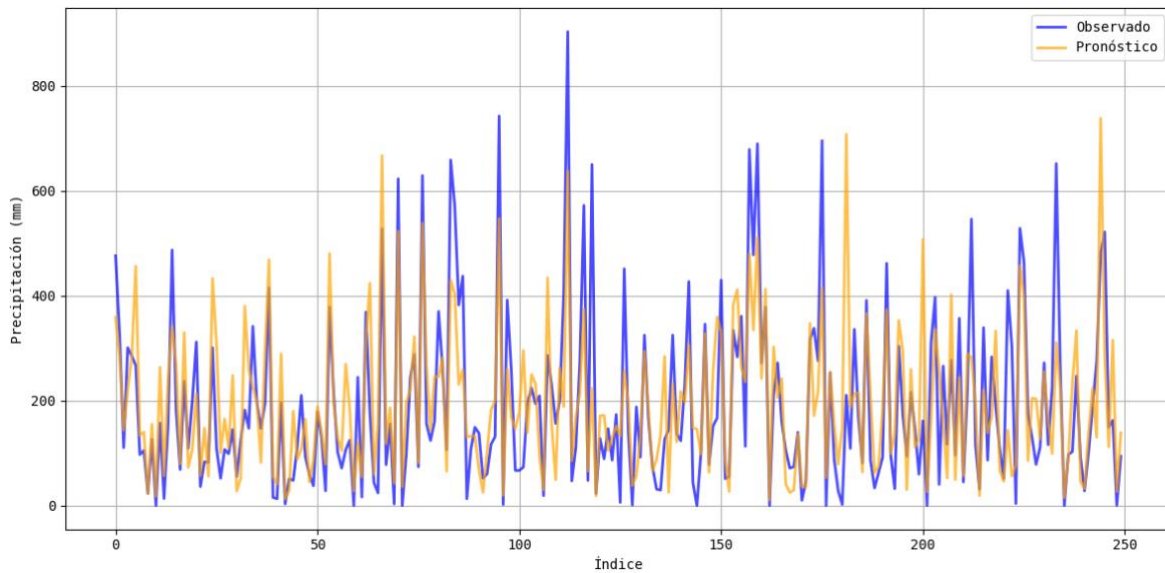


Figura 2. Predicciones vs. observaciones en una muestra aleatoria de 250 meses del conjuntos de datos de prueba utilizando el modelo XGBoost.

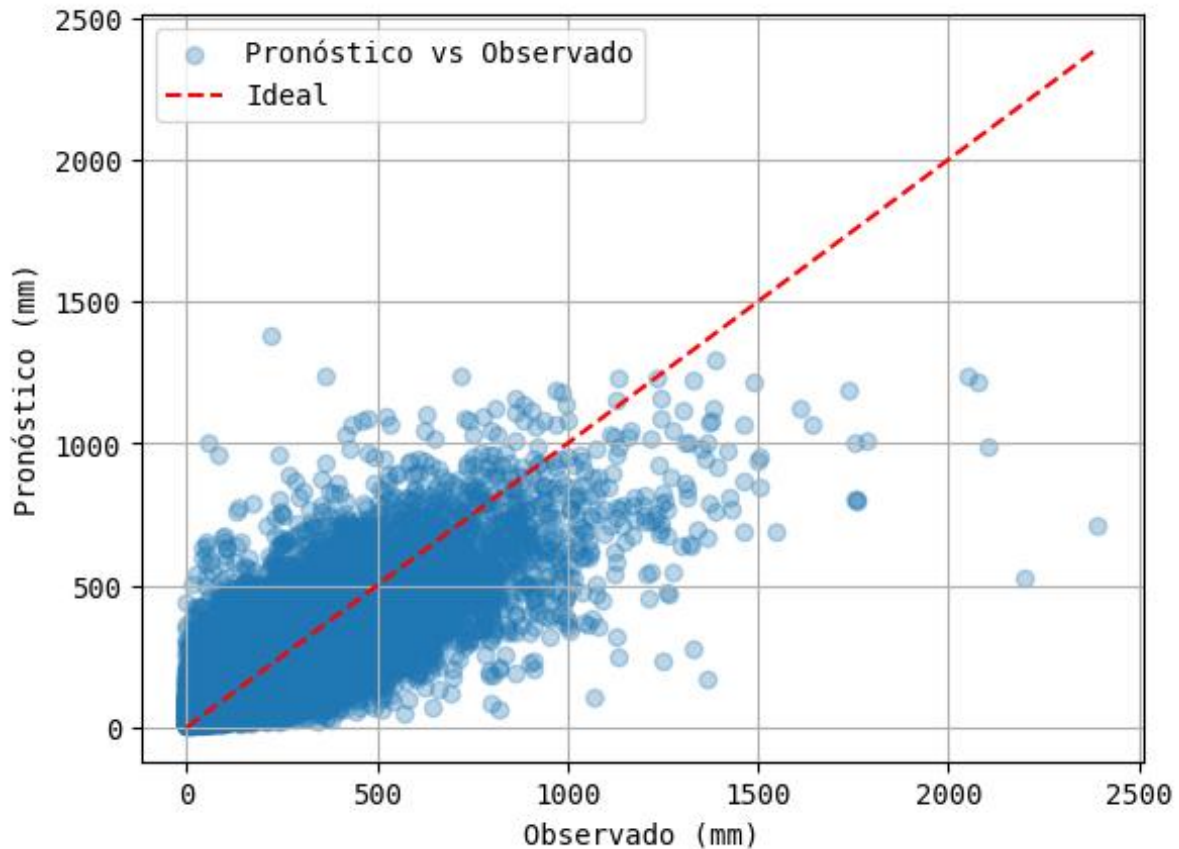


Figura 3: Diagrama de dispersión del conjunto de datos de prueba utilizando el modelo XGBoost.

ACTIVIDADES REALIZADAS:

1. Se desarrolló el script “training.ipynb” para la ingeniería y entrenamiento de los modelos.
2. Se desarrolló el script “evaluation.ipynb” para la evaluación de los modelos.

ENLACE: [Repositorio - GitHub](#)

Jorge Eliecer Camargo Mendoza
Director

Juan Diego Mogollón Oviedo
Estudiante