Assembled Methods for the Prediction of the Incident GHI over the City of Puno in Tilted Solar Panels

Fernando Gonzalo Loayza-Pizarro, Estudiante de Ciencia de la Computación[1], Yuri Nuñez-Medrano, MSc. en Ingeniería de Sistemas[2]
[1]Universidad Nacional de Ingeniería, Perú, *fernando.loayza.p@uni.pe, ynunezm@uni.edu.pe*

Resumen– ***This paper focuses on the comparison of Ensemble Machine Learning (ML) methods to forecast the incident GHI on a solar panel tilted at 30° (sexagesimal degrees). In addition, the performance was compared with commonly used simple ML regression methods.***
A pre-calculation of the GHI (variable used as a target by the models) was performed for 30° tilting surfaces. The optimized Bagging Ensemble and Extra Trees methods obtained better results using the evaluation metrics MSE and R^2. In general, the Ensemble models showed satisfactory results and can be used for global solar radiation estimation at locations where only sunshine hours data are available.

Keywords-- Machine Learning (ML), Assembled methods, Solar Irradiance, Solar Panel, Global Horizontal Irradiance (GHI), Mean Squared Error (MSE).

Digital Object Identifier (DOI):
http://dx.doi.org/10.18687/LACCEI2022.1.1.529
ISBN: 978-628-95207-0-5 **ISSN:** 2414-6390

Métodos Ensamblados para la Predicción del GHI Incidente sobre la Ciudad de Puno en Paneles Solares Inclinados

Fernando Gonzalo Loayza-Pizarro, Estudiante de Ciencia de la Computación[1], Yuri Nuñez-Medrano, MSc. en Ingeniería de Sistemas[2]
[1]Universidad Nacional de Ingeniería, Perú, *fernando.loayza.p@uni.pe, ynunezm@uni.edu.pe*

Resumen– Este trabajo está enfocado en comparar métodos Machine Learning Ensamblados para la estimación del GHI incidente sobre un Panel Solar Inclinado en 30° (grados sexagesimales). Además, los desempeños fueron comparados con métodos ML de Regresión Simple habitualmente usados.

Se realizó pre-cálculo del GHI (variable usada como target en los modelos), para superficies inclinadas 30°. Los métodos de Ensamblado Bagging y Extra Trees optimizados obtuvieron mejor desempeño usando las métricas de evaluación MSE y R^2. En general, los modelos de Ensamblado mostraron resultados satisfactorios y se pueden utilizar para la estimación de la radiación solar global en sitios donde solo se cuenta con datos de horas de brillo solar.

Palabras claves-- Machine Learning (ML), Métodos Ensamblados, Irradiancia Solar, Panel Solar, Global Horizontal Irradiance (GHI), Mean Squared Error (MSE).

I. Introducción

Nuevas alternativas de energía han surgido en el último siglo, esto a causa del interés en usar fuentes de energía no renovables o que puedan no ser suficiente en potencia para satisfacer el consumo humano diario. Sistemas que almacenan energía como la solar o eólica poco a poco están ganando metros cuadrados en nuestro planeta.

Según el Servicio Nacional de Meteorología e Hidrología del Perú (Senamhi) y la Organización Mundial de la Salud (OMS), el índice de radiación ultravioletapromedio o más alta del año obtenido, cataloga como extremadamente alto [1], además de ser una de las regiones conmayor radiación llegando inclusive hasta los 19 UV.

La ciudad de Puno-Perú presenta un índice de radiación extremadamente alto, esto sumado a la ubicación geográfica, hace que los habitantes que viven en zonas con mayor irradiancia solar presenten dificultades para satisfacer necesidades básicas como iluminación, carga de celulares, radio, televisión, etc, por lo tanto, se debería hacer un análisis de las posibles zonas con alta irradiancia para el desarrollo o implementación de Paneles Solares. Se usarán medidas meteorológicas de la coordenada (-15.83, -70.02), perteneciente a la ciudad de Puno obtenidas de la National Solar Radiation Database (NSRDB) [2]. La NSRDB no tiene medición de GHI para la region Puno, entonces se debe hacer precálculo del GHI, para ello nos apoyaremos de las citas [12], [13] y [14].

En la medida de lograr predecir el GHI de incidencia que tendrá un Panel Solar Inclinado en la coordenada dada, se haceuso de algoritmos ML Simples y Ensamblados, ya que son técnicas que tienen mucho interés actual, los cuales son evaluados por métricas de Regresión como Error Cuadrático Medio (MSE) y el Coeficiente de Determinación (R^2), permitiendo compararlos y poder decidir el modelo con mejor performance para este tipo de tareas.

Así mismo, referencias como [10] y [11] usan fuentes de datos meteorológicos del NSBD de NREL y API World Weather Online, respectivamente. Para luego análizar relación de las variables climáticas que tomarán como entrada los algoritmos para cada estudio. De ambas referencias se puede deducir, la combinación de técnicas presentan major desempeño que una individual.

II. Métodos Ensamblados

Es un paradigma de aprendizaje múltiple en el que se construyen varios modelos y los resultados (predicciones) se combinan, dan lugar a una decisión final. A cada modelo del conjunto se le llama “aprendiz base” o “base learner”. Investigaciones [3] y [4] indican que a menudo la precisión del conjunto de modelos es mejor que la de los modelos base ya que aprovecharían la complementariedad de las predicciones individuales de sus integrantes.

A. Voting

Voting combina las predicciones de los modelos de base según a un esquema de votación estático (static voting scheme),que no depende de datos de aprendizaje ni de los modelos base [3]. El tipo más simple de voting es el “majority vote”, donde cadamodelo base emite un voto para su predicción [3] y [7]. La predicción que recoge más votos es la predicción final de todo el conjunto. Si se predice un valor numérico, la predicción del conjunto esla media de las predicciones de los modelos de base.

B. Bagging

Bagging o bootstrap aggregation (L. Breiman, 1996), es unmétodo voting en el que los modelos base aprenden con diferentes variantes del conjunto de datos que se generan mediante bootstrapping (muestreo bootstrap). Bootstrapping esuna técnica de muestreo con sustitución, a

partir de un conjunto inicial de datos de aprendizaje se selecciona al azar nuevos sub- conjuntos. Según [5], la probabilidad que un sub-conjuntoaparezca por lo menos una vez es alrededor de 0,632.

Tenemos variantes más robuztos de este método como Random Forest y Extremely Randomized Trees (Extra Trees).

C. Boosting

El método Boosting fue introducido por Schapire (1990) como un método para aumentar el rendimiento del modelo "aprendiz" más "débil". Boosting comprende toda una familia de métodos, al igual que Bagging, utilizan la votación para combinar las predicciones de los modelos base aprendidos. La diferencia entre los dos enfoques, comenta [6], [8] y [9], es que en el Bagging la "complementariedad" de los modelos base construidos se deja al azar, mientras que en el boosting se intenta "generar" modelos base complementarios. Otros métodos basados en Bossting son los conocidos AdaBoost, CatBoost, gradient boosting y XGBoost.

D. Stacking

El Stacking o "stacked generalization" es un método para combinar modelos base heterogéneos. Los modelos base no se combinan con un esquema fijo como en Voting, sino que se "aprende" un modelo adicional llamado "meta-modelo" (o nivel 1) y se utiliza para combinar los modelos base (o nivel 0).Según [3] y [7], la algoritmia consta de dos pasos:

1. Generar el conjunto de datos de "meta- aprendizaje" usando las predicciones de los modelos base.
2. Usa el conjunto de datos de "meta-aprendizaje" para modelar el "meta-modelo" que puede combinar las predicciones de los modelos base en una predicción final.

III. ESTIMACIÓN DE RADIACIÓN SOLAR GLOBAL PROMEDIO DIARIA SOBRE UN PANEL SOLAR NO HORIZONTAL

La Irradiación Solar que llega a un Panel Solar no es la misma con la que llega a la superficie de la tierra (generalmente). Además no todos los Paneles Solares están distribuidos horizontalmente sobre la superficie terrestre.

Según [12], [13] y [14], la Irradiación Solar que llega a un Panel Solar Inclinado puede ser calculado de la siguiente manera:

- Factor de excentricidad (E_0): Es la razón entre el promedio de la distancia sol-tierra durante todo el año sobre la distancia sol-tierra en cualquier época o día (d_n) del año.

$$E_0 = 1 + 0.033 \cos\left(\frac{360 d_n}{365}\right) \quad (1)$$

- Ángulo de Declinación (δ): Es el ángulo que forma la línea sol-Tierra y el plano del ecuador o su proyección.

$$\delta = 0.409 \sin\left[\frac{360 d_n}{365} - 1.39\right] \quad (2)$$

- Ángulo horario (ω): Es el primer ángulo que forman los rayos del sol del lugar de estudio con la horizontal del mismo. Este ángulo es calculado a partir de la ecuación (3) y φ es la latitud de la zona de studio.

$$\omega = -\arccos(-\tan(\phi)\tan(\delta)) \quad (3)$$

- Radiación solar global extraterrestre (Re): Es la radiación medida en la frontera extraterrestre de la tierra y es calculado usando la ecuación (4). Donde I0 es la constante solar y T es el periodo terrestre de rotación (24h).

$$R_e = \frac{T}{\pi} E_0 I_0 (\omega \sin\phi \sin\delta + \cos\phi \cos\delta \sin\omega) \quad (4)$$

- Radiación solar global (Rg): Es la radiación que proporciona una base de datos reconocida. Los valores A, B y C son valores empíricos los cuales se pueden obtener del Atlas Energía Solar del Perú.

$$R_g = A R_e \left(1 - e^{-B(T_max - T min)^C}\right) \quad (5)$$

IV. METODOLOGÍA

A. *Análisis y pre-procesamientos de los datos*

El dataset completo obtenido de la NSRDB tiene un archivo por cada año (1998-2019), cada archivo contiene valores meteorológicos como el DNI, DHI y GHI tomados cada 30 minutos, mediante un lenguaje de programación se logró unirlos automáticamente, este dataset cuenta con 385679 filas y 24 columnas, además se ha pre-calculado el GHI para una inclinación (φ=30°). Es necesario análizar como es el comportamiento del GHI en algunos días tomados al azar en diferentes años (ver Fig. 1).

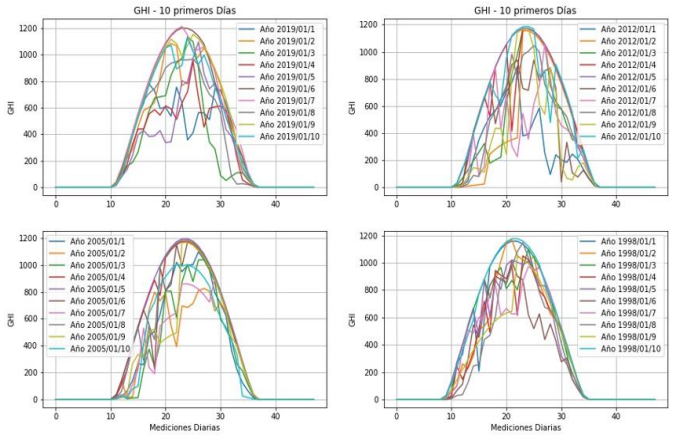

Fig. 1 Distribución GHI - 10 Primeros días de los años (2019, 2012, 2005 y 1998).

La distribución GHI por estaciones (ver la Fig. 2), se puede observar que las estaciones Primavera y Verano presentan mayores picos de GHI.

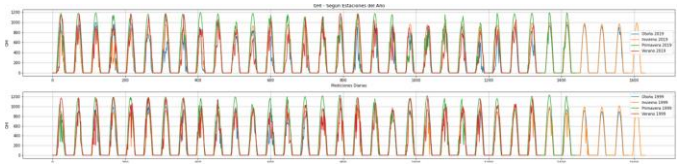

Fig. 2 Distribución GHI - Por estaciones de los años (2019 y 1999).

Procedemos a identificar el valor del Día Solar Promedio en este trabajo se considera 6 horas, que comprende desde las 9h hasta las 15h. Este filtro se puede visualizar en la Fig. 3, en el rango dado se cree obtener el mejor rendimiento de los Paneles Solares porque hay mayor radiación durante todo el día. Al final con este filtro el número de filas se reduce a 112490.

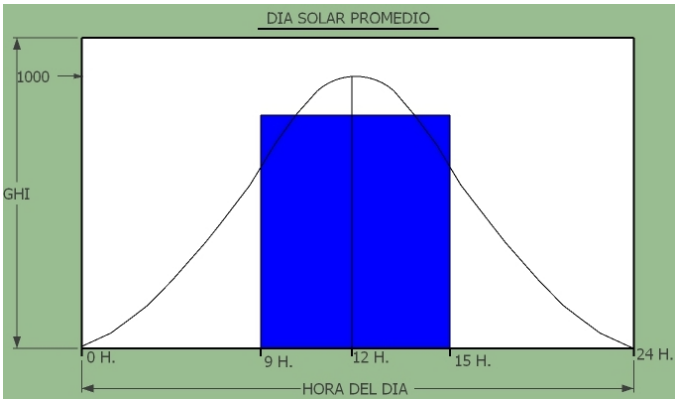

Fig. 3 Día Solar Promedio (6h).

B. Implementación y entrenamiento de los algoritmos ML Simples y Ensamblados

En este proceso se implementan y entrenan métodos ML con los valores X_train e Y_train, aquí se usarán los hyperparámetros configurados por defecto para cada método tanto los de regresión simple como los métodos ensamblados.

C. Comparación y selección de los algoritmos ML más exitosos

Los resultados del proceso anterior se compararán según la métrica Negative Mean Squared Error (nMSE) y de la desviación estándar para luego seleccionar aquellos con los que se obtiene un nMSE cercano a 0 (cero) y una desviación estandar pequeña, ya que se necesita que los mejores modelos sean estables.

D. Optimización, testeo y validación de la eficiencia de los mejores algoritmos ML

Para la optimización de nuestros mejores algoritmos ML, se propone tunear con diferentes hyperparámetros y asi lograr optimizarlos. En todo trabajo ML es necesario verificar el rendimiento de los modelos. Para ello se utilizarán los datos X_test y Y_test que previamente fueron divididos. Finalmente, se utilizan como índices de rendimiento el Accuracy y Loss.

V. Resultados

Se muestra primero, los resultados de los métodos ML simples. En la Fig. 4 y Tabla I, se pueden apreciar que los métodos de Regresión Lineal, Ridge, Lasso y ElasticNet presentan valores similares para las métricas nMSE y las desviaciones estándar. Respecto al método Support Vector Regression (SVR) no logra terminar el entrenamiento, se evidencia lo costoso que es el modelo.

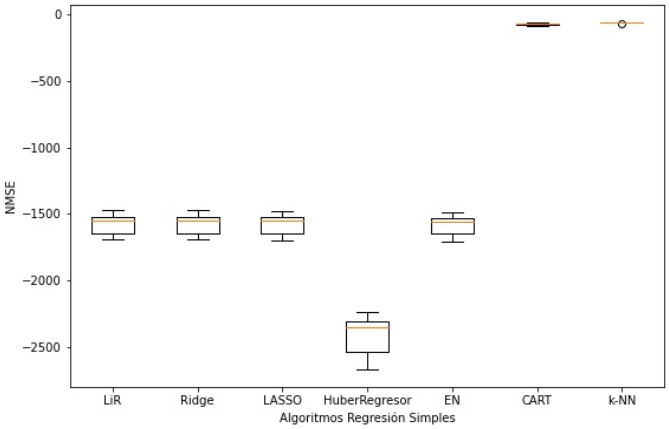

Fig. 4 Box-Plot del nMSE de los métodos ML Simples.

Los mejores modelos ML de Regresión Simple que trabajan con el dataset son: Classification And Regression Trees (CART), k-Nearest-Neighbor (k-NN). Estos modelos como se observa (Fig. 4 y Tabla I), presentan nMSE cercanos a 0 con desviaciones estándar pequeñas y tiempo de ejecución aceptable. Los resultados para el método SVR no se pudieron obtener ya que no logr´o terminar este proceso.

TABLA I
MÉTRICAS DE ENTRENAMIENTO MÉTODOS ML SIMPLES

Métodos Regresion Simple	nMSE	Desviación Estándar	Tiempo Ejecución (s)
Regresión Lineal (LiR)	-1,576.20	74.934	0.842
Ridge	-1,576.20	74.984	0.696
Lasso	-1,579.87	76.241	9.635
Huber	-2,420.48	145.906	17.633
Elastic Net	-1,587.11	75.859	9.928
Decision Tree	-77.941	8.301	9.692
k-NN	-65.115	3.495	12.372
SVR	-	-	-

Ahora mostramos los resultados de los modelos ML simples usando como una entrada el dataset estándarizado y escalado. La transformación realizada puede perjudicar a los modelos, estos modelos son (ver Fig. 5 y Tabla II): ElasticNet y k-NN; el modelo que se beneficia de esta transformación es CART. Al aplicar estas transformaciones el tiempo de entrenamiento prácticamente se mantiene en la mayoría de métodos, a excepción del método k-NN, donde el tiempo de entrenamiento pasó de 12s a 312s.

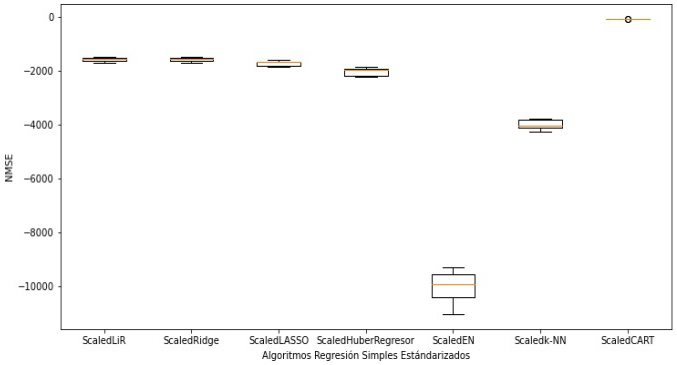

Fig. 5 Box-Plot del nMSE de los métodos ML Simples.

TABLA II
MÉTRICAS DE ENTRENAMIENTO MÉTODOS ML SIMPLES ESTÁNDARIZADOS

Métodos Regresion Simple Estándarizado	nMSE	Desviación Estándar	Tiempo Ejecución (s)
Regresión Lineal	-1576.2	74.934	1.35
Ridge	-1576.201	74.976	0.848
Lasso	-1720.755	91.679	2.46
Huber	-2040.884	135.737	20.989
Elastic Net	-10042.574	564.804	1.305
Decision Tree	-77.83	8.628	10
k-NN	-3997.231	168.429	312.427
SVR	-	-	-

Los métodos de Regresión Simple donde se obtuvieron mejores resultados son: CART y k-NN, estos modelos se optimizarán más adelante.

En cuánto a resultados de entrenamiento de los métodos ML de Ensamblado (ver Fig. 6 y Tabla III), los valores del nMSE mejoran mucho con respecto a los presentados para los métodos anteriores, sin embargo, el tiempo de ejecución se ha incrementado. Los modelos más lentos en entrenamiento son: Voting, AdaBoost, Gradient Boosting y el Stacking, de éstos modelos el AdaBoost tiene los peores resultados de desviación estándar. Los métodos Bagging, Random Forest, Extra Trees, XGBoost y CatBoost son muy buenos, tienen un nMSE menor a −50, una desviación estándar pequeña lo que los hace más confiables. En cuánto a los tiempos de ejecución se puede decir en general, que se ha prevalecido obtener un mejor nMSE a costa de un mayor tiempo de ejecución; XGBoost es el método con menos tiempo de ejecución, logrando el entrenamiento en tal solo 49s.

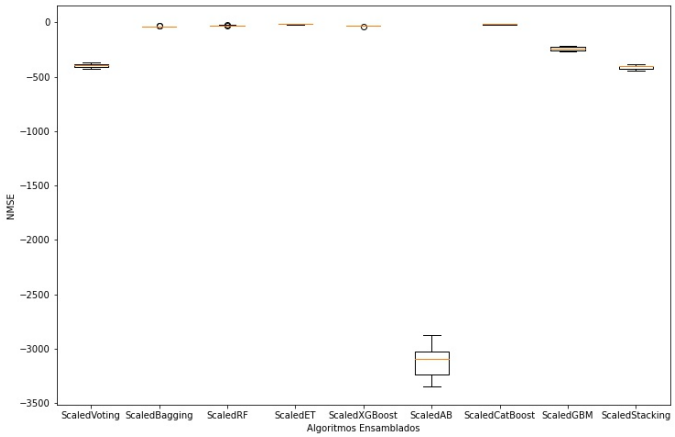

Fig. 6 Box-Plot del nMSE de los métodos ML Ensamblados

TABLA III
MÉTRICAS DE ENTRENAMIENTO MÉTODOS ML ENSAMBLADOS

Métodos Ensamblados	**nMSE**	**Desviación Estándar**	**Tiempo Ejecución (s)**
Voting	-396.32	19.13	66.565
Bagging	-35.8	3.84	68.595
Random Forest	-28.83	3.48	644.463
Extra Trees	-15.41	1.89	338.996
XGBoost	-33.09	1.9	48.782
AdaBoost	-3134.73	132.02	124.926
CatBoost	-17.19	0.92	164.095
Gradient Boosting	-242.45	16.35	251.367
Stacking	-411.02	18.37	99.76

Es momento de optimizar nuestros modelos, se han elegido los mejores siete modelos tanto de Regresión Simple como Ensamblados. Los resultados se muestran en la Tabla IV, los métodos de Regresión Simple (CART y k-NN) no logran mejora significativa con el valor nMSE, de igual manera ocurre en los métodos Ensamblados como Random Forest, Extra Trees, pero los valores son mucho más bajos. El método Bagging es el que mejora positivamente con un buen margen. No se ha podido tunear con más hyperparámetros debido a problemas de cómputo, puesto que se crean tantas instancias de acuerdo al número de hyperparámetros utilizados.

Por otro lado, se logra observar, tunear hyperparámetros a veces no garantiza la mejora de las métricas de los modelos, ejemplo de ello se tiene a los métodos XGBoost y CatBoost, en donde se aprecia una caída del rendimiento, esto implica que fue mejor usar los hyperparámetros por defecto configurados para estos modelos.

TABLA IV
MÉTRICAS DE ENTRENAMIENTO MEJORES MÉTODOS ML CON TUNING

Método Regresion	**nMSE (Antes)**	**nMSE**	**Mejores Hyperparámetros**
Decision Tree	-77.941	-77.559	random state = 1, splitter = best
k-NN	-3997.231	-3923.811	n neighbors = 78
Bagging	-35.8	-28.501	n estimators = 150, random state = 1
Random Forest	-28.83	-28.786	n estimators = 125
Extra Trees	-15.41	-15.333	n estimators = 150
XGBoost	-33.09	-173.361	colsample bytree = 1.0, n estimators = 150, objective = reg:squarederror
CatBoost	-17.19	-41.513	n estimators = 150

Resultados en la etapa de validación de los modelos, se usaron las métricas de evaluación MSE y R^2, si bien no son suficientes para decidir el/los mejor(es) modelo(s), ayudan a entender el error promedio y el grado de ajuste de valores predichos frente a los reales, respectivamente. La Tabla V contiene los resultados de esta validación. Partiremos de los valores nMSE como se sabe (por las secciones anteriores) estos siete métodos presentan los valores más pequeños, con esto se quiere decir, mientras más cercano se está de 0 será mejor, a esto se le debe agregar la observación de tener una desviación estándar pequeña. Todos los métodos tienen un buen ajuste entre los valores predichos con los valores reales, además cabe decir que la métrica R^2 no es muy confiable ya que no logra medir posibles sesgos, etc.

De los modelos presentados podríamos pensar que Extra Trees sería adecuado, esto no es así ya que en la Tabla III se observa que necesita cierto tiempo de entrenamiento. Entonces, otro de los modelos métodos comparando las métricas en este estudio, sería el método Bagging, como se ha venido comentando, este método logra conseguir que los errores se vayan compensando entre sí, usando subconjuntos de entrenamiento aleatoriamente y con repetición, esta característica favorece a la obtención de mejores resultados. La Fig. 7 muestra el GHI real comparado con el predecido por el modelo Bagging para paneles inclinados 30°. Un método con similares valores de las métricas es Random Forest pero el tiempo que toma para entrenar es grande, pero puede servir como una alternativa si se está priorizando performance en lugar de tiempo.

TABLA V
COMPARACIÓN DE VALIDACIÓN MEJORES MÉTODOS ML

Método Regresion	**Hyperparámetros por defecto**	**MSE**	**R^2**	**Cercano al Entrenamiento**
Decision Tree	No	21.438	0.99881	No (menor)
k-NN	No	3178.215	0.9989	Si
Bagging	No	10.126	0.9995	No (menor)
Random Forest	No	10.237	0.9995	No (menor)
Extra Trees	No	4.305	0.9997	No (menor)
XGBoost	Si	233.624	0.9959	Si
CatBoost	Si	14.777	0.9997	No

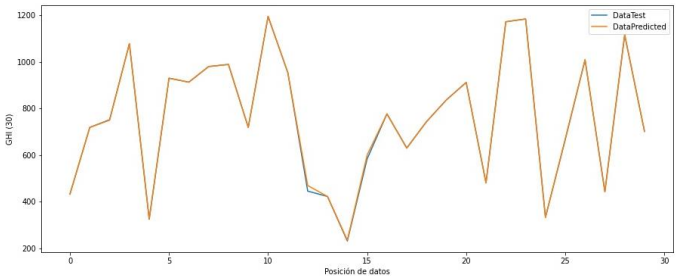

Fig. 7 Comparación de datos de Prueba y de Predicción - Método Bagging

VI. CONCLUSIONES

En la primera parte del trabajo se logra obtener un target adecuado, estos son los GHI incidentes para un Panel Solar Inclinado 30°. Esta obtención requirió de cálculos matemáticos como el Factor de Excentricidad, Angulo de Declinación, Ángulo horario, Radiación Solar Global Extraterrestre, etc. El método de Machine Learning SVR no terminó el entrenamiento del modelo. Así mismo, se logró implementar y comparar todos los métodos Ensamblados propuestos con los algoritmos ML de regresión.

También se pudo optimizar los modelos con mayor performance mediante tuneo de hyperpárametros. Se compararon los métodos de ML CART, k-NN, Bagging, Random Forest, Elastic Net, XGBoost y CatBoost, estos tuvieron mejores valores en las métricas de evaluación (MSE, nMSE, R^2) y el tiempo de entrenamiento que tomó cada modelo. El método de Ensamblado Bagging obtuvo el mejor rendimiento con un MSE (10.126), un R^2 (0.9995) y un tiempo de ejecución de entrenamiento (68.595 segundos). Para trabajos donde se requiera mayor rendimiento a cambio de un mayor tiempo de entrenamiento, se aconseja la implementación del modelo Extra Trees, el cual obtuvo un MSE (4.305) y R^2 (0.9997). No se pudo comparar resultados obtenidos en otros trabajos ya que la mayoría de trabajos procesa sus entrenamientos con datos que se obtienen directamente de un Sistema Fotovoltaico.

AGRADECIMIENTO

Se agradece a la Universidad Nacional de Ingeniería (UNI-Perú), especialmente a la Facultad de Ciencias y al Vicerrectorado de Investigación de la UNI.

REFERENCIAS

[1] L. Lozano, A. Llacza y O. Sánchez, "Pronóstico con cobertura nacional del índice de radiación solar ultravioleta", Senamhi, 2016.
[2] NSRDB: National Solar Radiation Database, https://nsrdb.nrel.gov/.
[3] S. Dzeroski, P. Panov y B. Zenko. Machine Learning, "Ensemble Methods in", Eslovenia, 2009.
[4] F. Merchán. "Aplicación de Técnicas de Machine Learning a la Seguridad", Junio 2018.
[5] Zhi-Hua Zhou. Ensemble Learning. China.
[6] Zhi-Hua Zhou. Ensemble Methods Foundations and Algorithms. 2012.
[7] Aurélien Géron. Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. Junio 2019.
[8] P. Harrington. "Machine Learning in Action", 2012.
[9] G. Ridgeway, D. Madigan y T. Richardson. "Boosting Methodology for Regression Problems". Universidad de Washington.
[10] E. Obando-Paredes, S. Carvajal-Quintero, J. Pineda. "Comparación metodológica para pronosticar capacidad de generación de energía fotovoltaica basado en datos climáticos", 2017.
[11] L.-E. Ordoñez-Palacios, D.-A. León-Vargas, V.-A. Bucheli-Guerrero, H.-A. Ordoñez-Eraso, "Solar Radiation Prediction on Photovoltaic Systems Using Machine Learning Techniques", Revista Facultad de Ingeniería, vol. 29 (54), e11751, 2020.
[12] O. Guzmán-Martínez, J.-Vicente Baldión-Rincón, O. Simbaqueva-Fonseca, H. Josué-Zapata y C. Chacón-Cardona. "Coeficientes para estimar la radiación solar global a partir del brillo solar en la zona cafetera colombiana". Revista Cenicafé 64(1): 60-76, 2013.
[13] J.-R Gómez-Sarduy, J.-F Puerta-Fernández, A. González-Alén, M. Gessa Gálvez. "Direct and diffuse solar radiation determinationat the Venezuelan seashore by using meteorological variables". Rev. Téc. Ing. Univ. Zulia. Vol. 38, Nº 2, 150 – 158, 2015.
[14] Lelia Quispe Huamán. "Determinación y Análisis Espacio Temporal de la Radiación Solar Global en el Altiplano de Puno". Universidad Nacional del Altiplano, Perú, 2018.