

# **ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**

## **Facultad de Ingeniería en Ciencias de la Tierra**

Desarrollo de Modelos de Machine Learning para Predecir la Resistencia a Compresión en Pasta de Cemento con Adiciones Minerales en la Industria de la Construcción

### **PROYECTO DE GRADUACIÓN**

Previo la obtención del Título de: Magíster en Ingeniería Civil.

**Mención: Construcción y Saneamiento**

Presentado por:

Jorge Esteban Campoverde Padilla

Sebastián Josué Vélez Altamirano

GUAYAQUIL - ECUADOR

Año: 2024

## DEDICATORIA

El presente proyecto lo dedico a mis padres y mi hermana quienes han sido mi ancla a tierra, y a mi enamorada, quienes me apoyan y motivan día a día para cumplir mis metas, a Jorge con quien he compartido a más de este camino de aprendizaje una gran amistad.

- Sebastián

El presente proyecto lo dedico a mis padres, mis abuelitos, y mi hermana quienes son el pilar fundamental que me ha impulsado a mejora día a día para siempre cumplir con mis metas, agradezco de una manera especial a mi colega y amigo de vida Sebastián con quien he tenido la oportunidad de formarme académicamente por varios años y con el cual he podido compartir esta nueva experiencia.

- Jorge

## **AGRADECIMIENTOS**

Nuestro más sincero agradecimiento a la Escuela Superior Politécnica del Litoral por acogernos en este posgrado, a la coordinación de la maestría por su apoyo durante todo este proceso. Una mención especial para el Ph.D. Mauricio Cornejo, quién nos supo compartir su conocimiento en la elaboración de nuestro proyecto final y siempre presto a ayudarnos.

Y nuestro agradecimiento a la empresa Ejeproy, misma que ha sido un gran complemento en este proceso de formación académica.

## DECLARACIÓN EXPRESA

“Los derechos de titularidad y explotación, nos corresponde conforme al reglamento de propiedad intelectual de la institución; *Jorge Esteban Campoverde Padilla* y *Sebastián Josué Vélez Altamirano* damos nuestro consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual”

---

Ing. Jorge Esteban  
Campoverde Padilla

---

Ing. Sebastián Josué  
Vélez Altamirano

# EVALUADORES

---

**Tutor Académico**

Ph.D. Mauricio Cornejo

---

**Profesora de Materia de Titulación**

M.Sc Nadia Quijano Arteaga

---

**Profesora de Materia de Titulación**

M.Sc Samantha Hidalgo Astudillo

## RESUMEN

Hoy en día existe un enfoque avanzado acerca de la inteligencia artificial, de la cual podemos vernos beneficiados. El proyecto trata acerca de una evaluación de modelos de machine learning (ML), los cuales permiten predecir la resistencia a compresión en pastas de cemento con adiciones minerales. Se menciona la importancia de esta propiedad característica en la industria de la construcción por lo que se propone la necesidad imperativa de desarrollar modelos predictivos que sean útiles. Se inicia con la recopilación de información experimental ya existente, después se analizan los datos a través de algoritmos de ML. Primero con su respectivo entrenamiento, después con la validación de los modelos, y finalmente con la evaluación de su precisión en la predicción.

Durante el desarrollo de este proyecto, se proponen el uso de varios modelos de ML encontrados en la revisión de la literatura científica, tales como Random Forest Regressor, Gradient Boosting Regressor, Extra Tree Regressor, Extreme Gradient Boosting, Adaboost y Decision Tree Regressor.

Los resultados obtenidos que han sido evaluados mediante varios parámetros estadísticos mostraron que los modelos de ML, i.e., Extra Tree Regressor y Random Forest Regressor presentan los mejores resultados en la predicción de la resistencia a la compresión. Estos resultados han sido comparados con los de la literatura que muestran coherencia en los mismos. Además, cabe resaltar que, a nuestro conocimiento, este estudio es el primero de su clase que se hace con cemento y adiciones minerales del Ecuador.

Finalmente se concluye que, a partir de los resultados obtenidos, el número de variables de entradas juega un papel crucial en el ajuste de los modelos de ML a los resultados experimentales.

Palabras Clave: Resistencia a Compresión, Machine Learning, cross validation, pasta de cemento

## **ABSTRACT**

Today there is an advanced approach to artificial intelligence, from which we can benefit. The project deals with an evaluation of machine learning (ML) models, which allow predicting the compressive strength of cement pastes with mineral additions. The importance of this characteristic property in the construction industry is mentioned, so the imperative need to develop useful predictive models is proposed. It begins with the collection of existing experimental information, then the data is analyzed through ML algorithms. First with their respective training, then with the validation of the models, and finally with the evaluation of their prediction accuracy.

During the development of this project, the use of several ML models found in the review of scientific literature is proposed, such as Random Forest Regressor, Gradient Boosting Regressor, Extra Tree Regressor, Extreme Gradient Boosting, Adaboost and Decision Tree Regressor.

The results obtained that have been evaluated using several statistical parameters showed that the ML models, i.e., Extra Tree Regressor and Random Forest Regressor present the best results in predicting compressive strength. These results have been compared with those of the literature, which show consistency. Furthermore, it should be noted that, to our knowledge, this study is the first of its kind that is carried out with cement and mineral additions from Ecuador.

Finally, it is concluded that, based on the results obtained, the number of input variables plays a crucial role in the adjustment of the ML models to the experimental results.

**Keywords:** Compression Strength, Machine Learning, cross validation, cement paste

# ÍNDICE GENERAL

EVALUADORES.....	5
RESUMEN .....	I
ÍNDICE GENERAL .....	III
ABREVIATURAS.....	VI
SIMBOLOGÍA.....	VI
ÍNDICE DE FIGURAS .....	VII
ÍNDICE DE TABLAS.....	VIII
CAPÍTULO 1 .....	9
1.1    Introducción.....	9
1.2    Antecedentes .....	9
1.3    Problemática a resolver .....	11
1.4    Justificación.....	12
1.5    Objetivos .....	13
1.5.1    Objetivo General .....	13
1.5.2    Objetivos Específicos.....	13
CAPÍTULO 2 .....	14
2.    Desarrollo del proyecto .....	14
2.1    Marco conceptual .....	14
2.1.1    Resistencia a Compresión en Pasta de Cemento con Adiciones Minerales 14	
2.1.2    Machine Learning en Ingeniería de Materiales.....	14
2.1.3    Relevancia en la Industria de la Construcción.....	15
2.1.4    Selección de Algoritmos de Machine Learning.....	15
2.1.5    Algoritmos de Machine Learning.....	15



2.2	Marco metodológico .....	17
2.2.1	Metodología en cascada: .....	17
2.2.2	Trabajo de laboratorio o gabinete.....	18
2.2.3	Tabulación de datos.....	19
2.2.4	Solución a diseñar.....	19
CAPÍTULO 3 .....		22
3.	RESULTADOS Y DISCUSIÓN .....	22
3.1	Resultados .....	22
3.1.1	Ingreso y Preprocesamiento de Datos .....	22
3.1.2	K-Fold Cross Validation.....	23
3.1.3	Creación y Entrenamiento del Modelo:.....	24
3.1.4	Predicción del Modelo .....	26
3.2	Optimización de Hiperparámetros .....	29
3.3	Comparación .....	32
3.3.1	MAE (Error Absoluto Medio) .....	32
3.3.2	MSE (Error Cuadrático Medio) .....	34
3.3.3	RMSE (Raíz del Error Cuadrático Medio).....	35
3.3.4	R <sup>2</sup> (Coeficiente de Determinación) .....	36
3.3.5	RMSLE (Error Cuadrático Medio del Logaritmo de las Predicciones)....	38
3.3.6	MAPE (Error Porcentual Absoluto Medio) .....	39
3.4	Análisis de resultados.....	42
3.5	Discusiones .....	45
CAPÍTULO 4 .....		48
4.	Conclusiones Y Recomendaciones.....	48
4.1	Conclusiones.....	48

4.2 Recomendaciones .....	49
BIBLIOGRAFÍA .....	51
ANEXOS .....	53
Datos de ingreso en variables independientes .....	54
Código PYCARET .....	60

## ABREVIATURAS

ESPOL	Escuela Superior Politécnica del Litoral
ML	Machine Learning
CS	Resistencia a la Compresión
ETR	Extra Tree Regressor
RF	Random Forest
ADABOOST	Adaptive Boosting
XGBOOST	Extreme Gradient Boosting
ODS	Objetivo de Desarrollo Sostenible
MAE	Error Absoluto Medio
MSE	Error Cuadrático Medio
RMSE	Raíz del Error Cuadrático Medio
R2	Coefficiente de Determinación
RMSLE	Error Cuadrático Medio del Logaritmo de las Predicciones
MAPE	Error Porcentual Absoluto Medio

## SIMBOLOGÍA

Kg/cm<sup>2</sup>      Kilogramo / centímetro cuadrado

## ÍNDICE DE FIGURAS

Figura 1: Características de Algoritmos de Machine Learning implementados .....	17
Figura 2: Error de predicción para Random Forest Regressor .....	27
Figura 3: Error de predicción para Gradient Boosting Regressor .....	28
Figura 4: Error de predicción para Extra Tress Regressor .....	28
Figura 5: Error de predicción para XGB Regressor .....	29
Figura 6: Error de predicción para AdaBoostRegressor .....	29
Figura 7: Error de predicción para Decision Tree Regressor .....	29
Figura 8: Comparativo Error Absoluto Medio .....	34
Figura 9: Comparativo Error Cuadrático Medio .....	35
Figura 10: Comparativo Raíz del Error Cuadrático Medio .....	36
Figura 11: Comparativo Coeficiente de Determinación .....	37
Figura 12: Comparativo Error Cuadrático Medio del Logaritmo de las Predicciones .	39
Figura 13: Comparativo Error Porcentual Absoluto Medio .....	41
Figura 14: Comparativo Extra Tree Regressor Vs GPR Exponencial .....	46
Figura 15: Comparativo Extra Tree Regressor Vs XGBoost .....	47
Figura 16: Importancia de las variables de entrada .....	49

## ÍNDICE DE TABLAS

Tabla 1: Resultados de los modelos ML aplicados en Pycaret. Fuente: Pycaret .....	23
Tabla 2: Resultados Random Forest Model. Fuente: Pycaret.....	24
Tabla 3: Resultados Gradient Boosting Regressor Model. Fuente: Pycaret.....	24
Tabla 4: Resultados Extra Tree Regressor Model. Fuente: Pycaret .....	25
Tabla 5: Resultados Extreme Gradient Boosting Model. Fuente: Pycaret.....	25
Tabla 6: Resultados Adaboost. Fuente: Pycaret.....	26
Tabla 7: Resultados Decision Tree Regressor Model. Fuente: Pycaret .....	26
Tabla 8: Resultados Random Forest Optimizado. Fuente: Pycaret.....	30
Tabla 9: Resultados Gradient Boosting Optimizado. Fuente: Pycaret.....	30
Tabla 10: Resultados Extra Tree Regressor Optimizado. Fuente: Pycaret .....	30
Tabla 11: Resultados Extreme Gradient Boosting Optimizado. Fuente: Pycaret.....	31
Tabla 12: Resultados Adaboost Optimizado. Fuente: Pycaret .....	31
Tabla 13: Resultados Decision Tree Regressor Optimizado. Fuente: Pycaret.....	31
Tabla 14: Comparativo Error Absoluto Medio. Fuente: Elaboración propia.....	33
Tabla 15: Comparativo Error Cuadrático Medio. Fuente: Elaboración propia.....	34
Tabla 16: Comparativo Raíz del Error Cuadrático Medio. Fuente: Elaboración propia .....	35
Tabla 17: Comparativo Coeficiente de Determinación. Fuente: Elaboración propia..	36
Tabla 18: Comparativo Error Cuadrático Medio del Logaritmo de las Predicciones. Fuente: Elaboración propia.....	38
Tabla 19: Comparativo Error Porcentual Absoluto Medio. Fuente: Elaboración propia .....	41
Tabla 20: Random Forest Regressor VS Extra Tress Regressor. Fuente: Elaboración propia .....	44
Tabla 21: Comparativo Extra Tree Regressor Vs GPR Exponencial. Fuente: Elaboración propia.....	46
Tabla 22: Comparativo Extra Tree Regressor Vs XGBoost. Fuente: Elaboración propia .....	47

# CAPÍTULO 1

## 1.1 Introducción

La resistencia a la compresión es una propiedad que se torna importante en la industria constructiva, en este caso con adiciones minerales se puede evaluar la proporción de reemplazo del cemento Portland. Una evaluación concisa de esta propiedad es clave, que permita garantizar una durabilidad y calidad de estructuras construidas con estos materiales. La necesidad de predecir efectivamente la resistencia a la compresión implica una búsqueda de nuevos enfoques, como los modelos de ML.

La resistencia a compresión es una propiedad clave que determina la capacidad que posee un material para resistir cargas. En este caso, la predicción mencionada se torna importante para optimizar la producción y mejorar la calidad del producto final en la industria cementicia. Debido a una falta de consenso sobre los distintos modelos predictivos, la propuesta actual plantea seleccionar entre distintos modelos predictivos que permitan predecir la propiedad mencionada basado en las características particulares de los componentes.

La aplicación de varios modelos (ML) se justifica debido a que estos poseen una alta capacidad de aprendizaje para captar patrones que por medio de métodos tradicionales estadísticos pueden obviarse. Se plantea aportar con información acerca de los modelos predictivos que permitan evaluar ciertas propiedades ingenieriles, en este caso la resistencia a la compresión, a su vez resolviendo la problemática de la predicción. Este planteamiento nos permite gestar mejoras de calidad y sobre todo durabilidad de estructuras edificadas con estos materiales, optimizando procesos productivos en la industria.

## 1.2 Antecedentes

Feng et al., (2020) presenta la problemática de la predicción en resistencia a la compresión del concreto y su importancia en la construcción de estructuras seguras y duraderas. Los autores inician su enfoque basado en el algoritmo de Adaboost (Adaptive Boosting), el cual usa una combinación de modelos de aprendizaje automático para mejorar paulatinamente la precisión de las predicciones. Los autores también mencionan acerca de la obtención de datos iniciales para entrenar al algoritmo, donde se incluye el

porcentaje de mezcla, tiempo de curado y resistencia a la compresión. Se discute acerca de la adhesión de varios ingredientes de mezcla que puedan afectar la resistencia del concreto.

Feng et al., (2020) muestra resultados y compara con distintos métodos de predicción, concluyendo que el enfoque utilizado basado en el algoritmo de Adaboost (Adaptive Boosting) superó a otros métodos (ML) en cuanto a precisión de predicción y robustez del conjunto. Además, se exploran las posibles aplicaciones prácticas de este enfoque en la construcción y cómo puede ayudar a mejorar la seguridad y durabilidad de las estructuras. También se discuten las posibles limitaciones y desventajas de este enfoque.

Recientemente, Chen et al., (2023) enfoca su proyecto en la optimización del diseño con proporciones de mezcla de hormigón de alto rendimiento mediante tecnologías de aprendizaje automático, cuyo objetivo es aumentar la calidad de la optimización, ampliando la base de datos de proporciones de mezcla de concreto para mejorar la generalidad del método propuesto. Chen et al., también presenta un marco inteligente híbrido para la optimización multiobjetivo basado en Random Forest (RF) y el algoritmo Non-dominated Sorting Genetic Algorithm II (NSGA II). Este marco se utiliza para predecir de manera eficiente la durabilidad del concreto y optimizar la proporción de mezcla de concreto. El módulo elástico dinámico relativo del hormigón después de 300 ciclos de congelación- descongelación y el coeficiente de permeabilidad del ion cloruro a los 28 días se definen como medidas estándar de durabilidad. La proporción de mezcla de concreto se toma como parámetro de influencia y los datos de pruebas ortogonales y los datos de prácticas de ingeniería se recopilan como conjuntos de datos.

En general, Chen et al., (2023) es una guía útil para aquellos interesados en la construcción de ferrocarriles de alta velocidad, puentes y alcantarillas de gran escala en China. El uso de tecnologías de aprendizaje automático y el marco inteligente híbrido presentado en el archivo pueden ayudar a mejorar la calidad y la eficiencia del diseño de proporciones de mezcla de concreto para garantizar la durabilidad del concreto diseñado cumpla con los requisitos específicos y al mismo tiempo minimizar su costo.

Stergio et al., (2023) proporciona una exploración detallada sobre la aplicación del aprendizaje automático en el ámbito de la ciencia de materiales. La premisa central del documento es la convergencia entre datos experimentales y simulaciones, utilizados de manera sinérgica para predecir propiedades de materiales y optimizar el diseño de estos.

En primera instancia, se destaca la importancia crítica de prever con precisión las propiedades de los materiales en el ámbito de la ciencia de materiales. Se introduce el aprendizaje automático como una herramienta clave para alcanzar este objetivo, subrayando su potencial para sintetizar y analizar grandes cantidades de datos heterogéneos.

El documento se adentra en los enfoques de aprendizaje automático utilizados específicamente en la ciencia de materiales. Se discuten los tipos de datos que pueden ser empleados, como datos experimentales y de simulación, y se ofrecen ejemplos concretos que ejemplifican cómo estos enfoques se han implementado para predecir el comportamiento de los materiales en diversas condiciones.

En resumen, el trabajo de Stergio et al., (2023) se erige como una contribución integral y actualizada que aborda la convergencia entre el aprendizaje automático y la ciencia de materiales, proporcionando una visión detallada de enfoques, algoritmos y tendencias emergentes en este fascinante cruce disciplinario.

Con respecto a la realidad de la región, hasta el momento no se ha aplicado esta técnica a materiales de construcción local, siendo la primera en su clase, donde mediante algoritmos de ML se busca la mejor opción para proponer el modelo mejor optimizado a través de una comparación, sirviendo de base a futuras investigaciones las cuales puedan adherirse y mejorar los resultados.

### **1.3 Problemática a resolver**

Hoy en día, la búsqueda de materiales amigables con el medio ambiente es un tema muy común de abordar, lo cual se vuelve un desafío a enfrentar en la industria de la construcción, de este modo una buena estimación de la resistencia a compresión en la pasta de cemento con adiciones minerales conlleva a garantizar una buena calidad y durabilidad de las estructuras.

Este problema promueve una investigación sobre la implementación de modelos de ML, para optimizar la producción y mejorar la calidad del producto final. La necesidad de obtener métodos precisos para predecir la resistencia a compresión se ubica en el contexto principal de evitar fallos estructurales y sobre todo mejorar la eficiencia en los procesos productivos.



El problema se sustenta en lo crítico que se torna la resistencia a la compresión para la seguridad y durabilidad de las estructuras, que tiene consecuencias directamente proporcionales en la parte integral de la construcción, llevando a fallos estructurales y costos adicionales por errores, los cuales generan pérdidas de recursos.

Esta investigación tiene como enfoque el desarrollo de modelos ML, ligándose con estudios y datos previos ya existentes acerca de la resistencia a compresión y aplicaciones ML en la industria constructiva. Se busca aportar con una evaluación que permita comparar los algoritmos y determinar el más adecuado.

Existe una falta de soluciones, lo que continúa siendo preocupante, ante posibles consecuencias negativas como construcciones inseguras, pérdida de recursos e incluso consecuencias legales. La metodología planteada se centrará en una evaluación sistemática de algoritmos ML para responder a la siguiente pregunta de investigación: ¿Cuál es el algoritmo de machine learning más adecuado para predecir la resistencia a compresión en la pasta de cemento con adiciones minerales en la industria de la construcción, considerando métricas de ajuste y optimización de la calidad del producto final?

#### **1.4 Justificación**

La problemática planteada, que consiste en la predicción de la resistencia a compresión en pastas de cemento con adiciones minerales mediante modelos ML, tiene suma importancia dentro de la industria constructiva, la ingeniería e incluso alineada con el desarrollo sostenible. Poder optimizar la predicción de esta propiedad característica, permite respaldar una calidad y durabilidad en las estructuras construidas, contribuyendo al Objetivo de Desarrollo Sostenible (ODS) No. 9, acerca de industria, innovación e infraestructura, el cual busca construir infraestructuras resilientes, promover la industrialización inclusiva y sostenible, y fomentar la innovación.

Al lograr una predicción más precisa de la resistencia a compresión, se pueden tomar decisiones informadas en la selección de materiales y procesos de construcción, reduciendo así el desperdicio de recursos y mejorando la eficiencia en la producción de estructuras. Esto está alineado con el ODS 12, que aboga por la producción y consumo responsables. Además, al contribuir al conocimiento sobre modelos predictivos precisos, el proyecto tiene el potencial de influir en las prácticas de la industria del cemento,

promoviendo en última instancia la sostenibilidad y la innovación en la construcción de infraestructuras.

## **1.5 Objetivos**

### **1.5.1 Objetivo General**

Proponer un modelo predictivo optimizado utilizando algoritmos de machine learning para la predicción de la resistencia a compresión en pasta de cemento con adiciones minerales, con el fin de mejorar la calidad y durabilidad de las infraestructuras.

### **1.5.2 Objetivos Específicos**

- Recopilar datos experimentales de resistencia a compresión en pasta de cemento con diversas adiciones minerales de la literatura para el aprendizaje de modelo ML
- Seleccionar los algoritmos de machine learning más adecuados para el desarrollo del modelo predictivo.
- Validar el modelo predictivo con los datos experimentales recopilados.
- Evaluar la precisión del modelo usando varios parámetros estadísticos, para la comparación con métodos convencionales.

# CAPÍTULO 2

## 2. DESARROLLO DEL PROYECTO

### 2.1 Marco conceptual

La importancia de este proyecto se sitúa en la intersección de la ingeniería de materiales, la industria de la construcción y la ciencia de datos, específicamente en la aplicación de modelos de machine learning para prever la resistencia a compresión en pasta de cemento con adiciones minerales. Este enfoque multidisciplinario busca abordar la falta de consenso en la identificación de modelos predictivos precisos para este material específico.

#### 2.1.1 Resistencia a Compresión en Pasta de Cemento con Adiciones Minerales

La resistencia a compresión en pasta de cemento con adiciones minerales es una propiedad clave que determina la capacidad de un material para soportar cargas. Este parámetro es fundamental en la industria de la construcción, ya que influye directamente en la calidad y durabilidad de las estructuras construidas. La pasta de cemento, en combinación con adiciones minerales, ha ganado popularidad debido a sus propiedades mejoradas en términos de resistencia y durabilidad, pero la precisión en la predicción de esta resistencia aún representa un desafío.

#### 2.1.2 Machine Learning en Ingeniería de Materiales

El uso de modelos de machine learning en ingeniería de materiales ha ganado terreno en la última década debido a su capacidad para analizar conjuntos de datos complejos y capturar patrones que métodos estadísticos tradicionales pueden pasar por alto. En este contexto, diversos algoritmos de machine learning, como extreme gradient boosting, decision tree regressor, gradient boosting, adaboost, random forest regressor, y extra tree regressor, ofrecen una amplia gama de enfoques para la predicción precisa de propiedades materiales.

### **2.1.3 Relevancia en la Industria de la Construcción**

La industria de la construcción está en constante búsqueda de mejoras en la eficiencia y calidad de los materiales utilizados. La predicción precisa de la resistencia a compresión en pasta de cemento con adiciones minerales no solo optimizará la producción, reduciendo el desperdicio de recursos, sino que también mejorará la durabilidad de las estructuras construidas.

### **2.1.4 Selección de Algoritmos de Machine Learning**

La selección de algoritmos de machine learning es una fase crucial en el desarrollo del modelo predictivo. La revisión de la literatura científica y técnica revela que cada uno de los algoritmos propuestos tiene ventajas y limitaciones específicas. El extreme gradient boosting, por ejemplo, destaca en la optimización del rendimiento, mientras que el random forest regressor es conocido por manejar bien conjuntos de datos grandes y complejos. La elección dependerá de la naturaleza de los datos experimentales recopilados y la complejidad inherente a la resistencia a compresión en pasta de cemento con adiciones minerales.

### **2.1.5 Algoritmos de Machine Learning**

El Machine Learning es un enfoque computacional que permite a los sistemas aprender patrones y tomar decisiones sin intervención humana explícita. En el contexto de la regresión, se centra en predecir valores numéricos.

### Adaboost (Adaptive Boosting)

- Se trata de un algoritmo de conjunto en donde se asigna pesos a las instancias de entrenamiento y corrige los errores ponderados mediante el ajuste de modelos secuenciales.
- **Características:**
  - Forma un modelo fuerte a través de la combinación de modelos débiles.
  - Es sensible a datos atípicos y al ruido.

### Random Forest Regressor

- Trata de un extensión de los árboles de decisión que desarrollan múltiples árboles y combinan sus predicciones para reducir y mejorar la generalización de sobreajuste, generando resultados más precisos y estables. Chen et al.
- **Características:**
  - Cada árbol se entren con un subconjunto de características
  - Se promedia las predicciones de los árboles para obtener una variable resultante final
  - Maneja gran cantidad de conjuntos de datos con características aleatorias.

### Extra Tree Regressor (Extremely Randomized Trees)

- Se trata de una variante del algoritmo Random Forest en la cual se agrega una capa de aleatoriedad durante la generación de los árboles de decisiones
- **Características**
  - Construcción de árboles sin poda adicional
  - Selección aleatoria de decisiones según nodos internos

### Gradient Boosting

- Es un algoritmo de ML que combina múltiples modelos de manera secuencial, con el objeto de corregir los errores del modelo anterior.
- **Características:**
  - Mediante la incorporación de nuevos modelos minimiza la función de pérdida.
  - Como modelo base puede implementar árboles de decisión.
  - Mediante la combinación de modelos débiles mejora su precisión.

### Extreme Gradient (XGBoost)

- Es un algoritmo de ML basado en árboles de decisión que utiliza un conjunto de aprendices débiles para crear un aprendiz fuerte, generando de esta forma un modelo más preciso. Stergio et al.
- **Características:**
  - Elabora árboles de decisión de manera secuencial para corregir errores anteriormente generados.
  - Puede manejar datos faltantes.

### Decisión Tree Regressor

- Se utiliza para la clasificación y regresión en el aprendizaje supervisado, puesto que se basa en un método de modelado de conjunto utilizando árboles de decisión para modelar relaciones no lineales entre las variables de entrada y la variable de salida u objetivo. Stergio et al.
- **Características:**
  - Divide el conjunto de datos en nodos de decisión.
  - El modelo puede manejar datos no lineales.

**Figura 1:** Características de Algoritmos de Machine Learning implementados

## 2.2 Marco metodológico

### 2.2.1 Metodología en cascada:

La metodología propuesta se estructura linealmente en procesos sucesivos:

### **1. Recopilación de datos experimentales**

Mediante la revisión exhaustiva de literatura científica se recopila datos de resistencia a la compresión en pasta de cemento, que servirán como base para el entrenamiento y validación del modelo.

### **2. Selección de Algoritmos**

En base a la literatura revisada, se selecciona algoritmos de ML adecuados para desarrollar el modelo predictivo. Su selección dependerá de la capacidad de cada algoritmo para abordar la complejidad del caso.

### **3. Entrenamiento y validación del modelo**

Los datos recopilados de la literatura se utilizarán para entrenar y validar el modelo predictivo. Dentro de este proceso se asegura la capacidad del modelo para pronosticar la resistencia a la compresión de la pasta de cemento.

### **4. Evaluación de la precisión del modelo**

La eficacia del modelo se definirá mediante la implementación de parámetros estadísticos. Adicionalmente, se compararán los resultados obtenidos con estudios relacionados a la obtención de la resistencia a la compresión mediante algoritmos ML.

## **2.2.2 Trabajo de laboratorio o gabinete**

Estos datos nacen de ensayos experimentales específicos. M.Cornejo (2014), en su investigación analiza depósitos de zeolitas naturales, los cuales han sido estudiados durante 10 años, en el período (2004 – 2014) ubicados desde la ciudad de Guayaquil

hasta la costa del mar a lo largo de más de 100 Km, es decir, la Cordillera Chongon Colonche, cuya investigación es un intento de arrojar luz sobre las aplicaciones prácticas de dichas materias primas, analiza también propiedades como el porcentaje de adiciones minerales (%SCM), tipo de adiciones, tipo de curado, envejecimiento y metodología aplicada en pastas de cemento, estableciendo una base sólida que aporta a la investigación y modelado posterior.

Una vez recopilada la información mencionada, se estructuró en una tabla de Excel para facilitar su procesamiento. Se realizó un análisis exploratorio de datos para identificar posibles anomalías y se dividió el conjunto de datos en conjuntos de entrenamiento y prueba. En cuanto a los métodos de machine learning, se analizaron algoritmos como Extreme Gradient Boosting, Decision Tree Regressor, Gradient Boosting, Adaboost, Random Forest Regressor y Extra Tree Regressor. La elección y aplicación de estos métodos se basaron en la literatura revisada, que proporcionó pautas sobre la idoneidad de cada algoritmo para abordar la problemática de predecir la resistencia a compresión en pastas de cemento con adiciones minerales.

### **2.2.3 Tabulación de datos**

El análisis de datos se realizó mediante una tabla de Excel, donde ingresamos distintos parámetros de diseño.

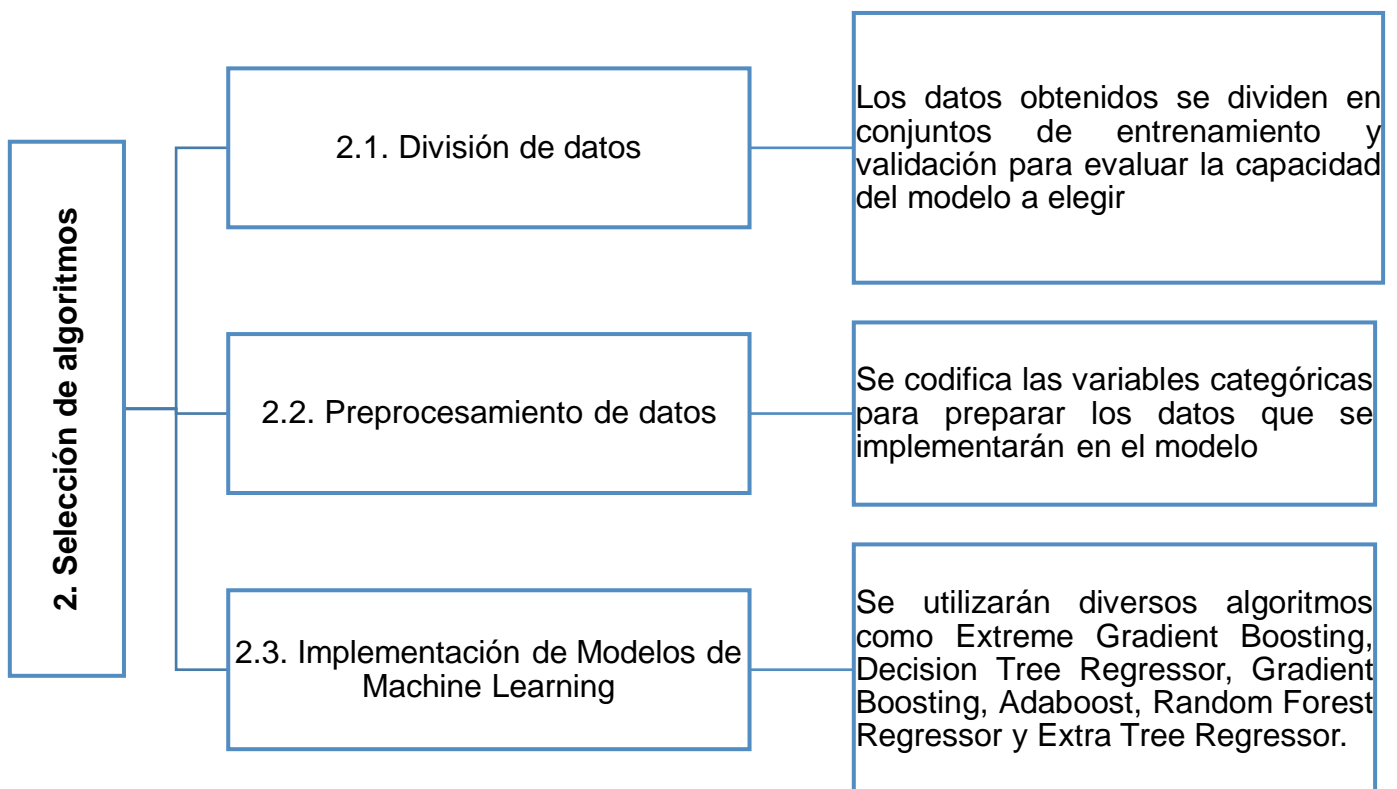
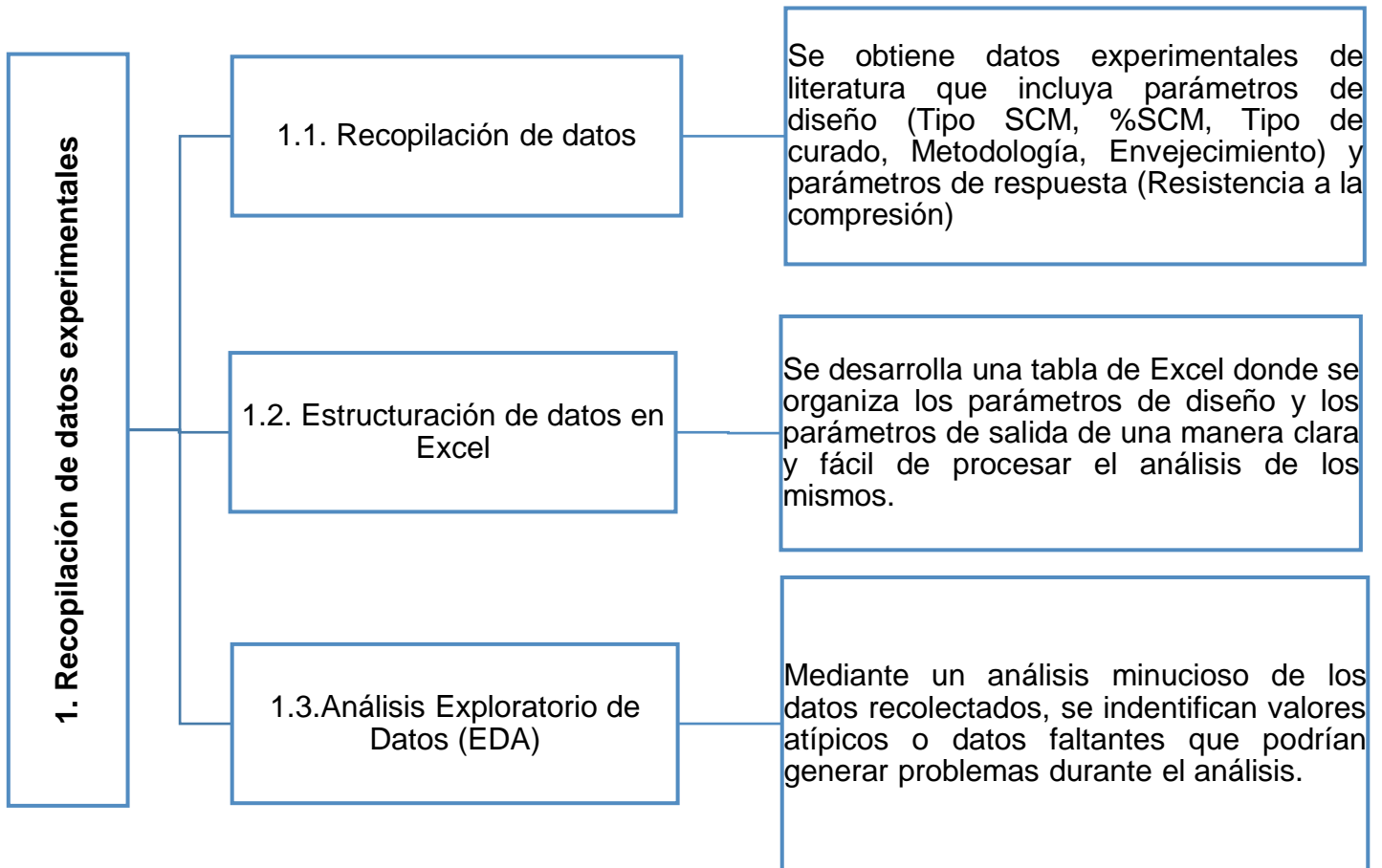
- Contenido de SCM
- Tipo SCM
- Tipo de curado
- Edad de envejecimiento
- Metodología

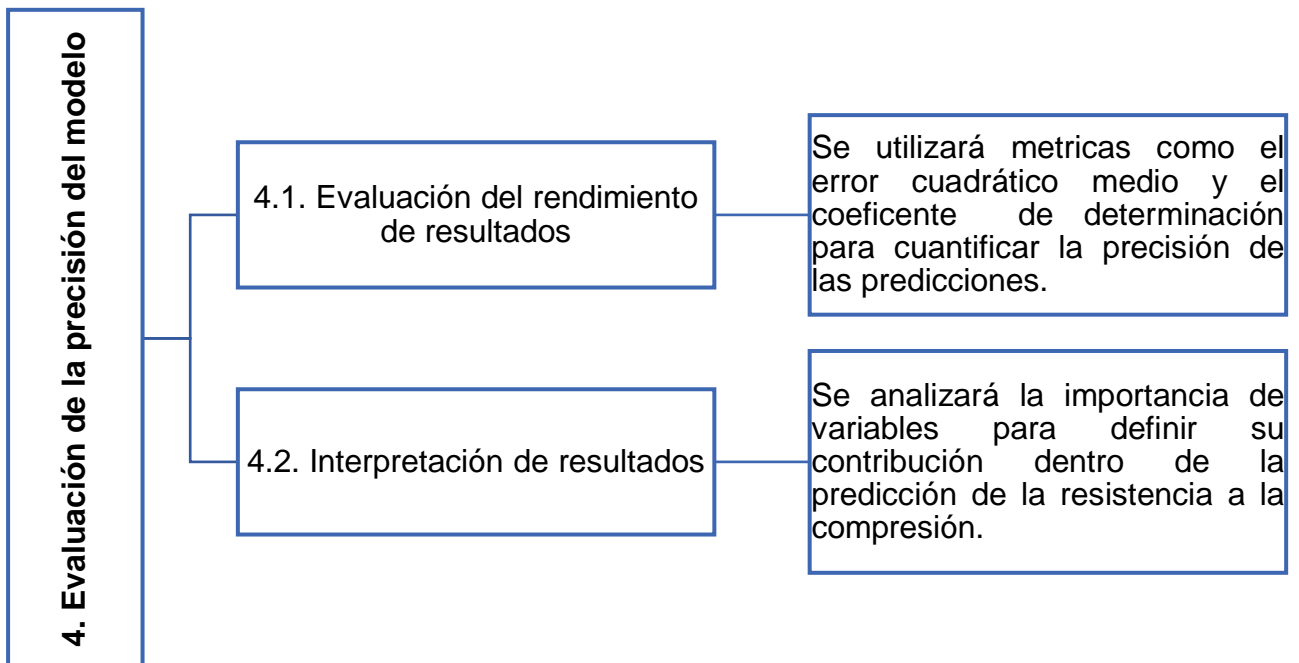
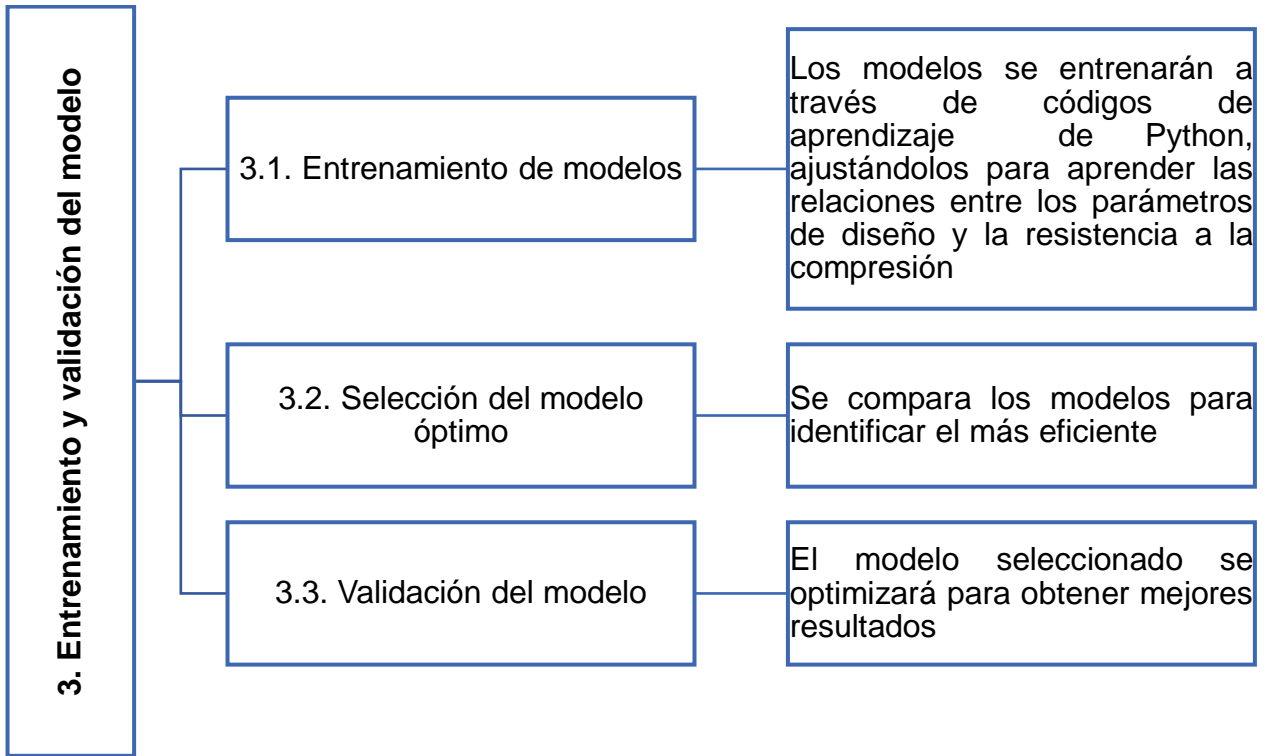
Para pastas de cemento con distintas adiciones minerales, seguido del parámetro de respuesta (Resistencia a la compresión), ligado a la inicial.

### **2.2.4 Solución a diseñar**

Este marco metodológico establece una secuencia de pasos desde la recopilación de datos hasta la interpretación de resultados, guiando el desarrollo de modelos de machine learning para predecir la resistencia a compresión en pasta de cemento con adiciones minerales.







# CAPÍTULO 3

## 3. RESULTADOS Y DISCUSIÓN

### 3.1 Resultados

En la fase de análisis de los resultados obtenidos mediante el uso de modelos de regresión en la predicción de la resistencia a la compresión de la pasta de cemento con adiciones minerales, se destacan varios modelos de regresión, como Extreme Gradient Boosting, Decision Tree Regressor, Gradient Boosting, Adaboost y Extra Tree Regressor.

#### 3.1.1 Ingreso y Preprocesamiento de Datos

El proceso comienza con la entrada de datos, que consiste en un conjunto de observaciones divididas en 5 subgrupos de 300, donde cada observación incluye datos de entrada relacionados con porcentajes de adiciones minerales, tipos de curado, envejecimiento, metodología, entre otros. Estos datos se someten a un proceso de preprocesamiento que incluye la metodología zscore para normalizarlos, garantizando así que todas las variables tengan la misma escala y contribuyan de manera equitativa al modelo.

**Tabla 1:** Resultados de los modelos ML aplicados en Pycaret. Fuente: Pycaret

	Model	MAE	MSE	RMSE	R2	RMSLE	MAPE	TT (Sec)
<b>rf</b>	Random Forest Regressor	8.1296	119.6337	10.7292	0.7541	0.1618	0.1265	0.2880
<b>gbr</b>	Gradient Boosting Regressor	8.8067	128.9087	11.2446	0.7328	0.1749	0.1381	0.1440
<b>et</b>	Extra Trees Regressor	8.6556	138.8095	11.5530	0.7152	0.1721	0.1327	0.2240
<b>lightgbm</b>	Light Gradient Boosting Machine	9.0194	137.3858	11.5692	0.7128	0.1764	0.1411	0.1460
<b>xgboost</b>	Extreme Gradient Boosting	9.1289	144.8475	11.8107	0.7037	0.1776	0.1413	0.1460
<b>ada</b>	AdaBoost Regressor	9.7282	150.4629	12.1388	0.6874	0.1833	0.1513	0.1340
<b>dt</b>	Decision Tree Regressor	10.9070	202.3217	14.1518	0.5742	0.2164	0.1666	0.1280
<b>huber</b>	Huber Regressor	10.6730	220.1356	14.6765	0.5433	0.2182	0.1727	0.1640
<b>br</b>	Bayesian Ridge	11.1020	219.4050	14.7040	0.5431	0.2190	0.1774	0.1700
<b>ridge</b>	Ridge Regression	11.0827	219.4893	14.7054	0.5428	0.2195	0.1772	0.0960
<b>lr</b>	Linear Regression	11.0783	219.5906	14.7083	0.5425	0.2197	0.1771	1.1860
<b>lar</b>	Least Angle Regression	11.0783	219.5906	14.7083	0.5425	0.2197	0.1771	0.1000
<b>lasso</b>	Lasso Regression	11.1878	222.8900	14.7934	0.5395	0.2182	0.1786	0.1100
<b>llar</b>	Lasso Least Angle Regression	11.1878	222.8905	14.7934	0.5395	0.2182	0.1786	0.1060
<b>knn</b>	K Neighbors Regressor	12.2588	245.0120	15.4492	0.4992	0.2266	0.1928	0.1180
<b>par</b>	Passive Aggressive Regressor	11.8820	242.6255	15.4681	0.4987	0.2326	0.1873	0.1900
<b>en</b>	Elastic Net	12.4970	252.6704	15.8090	0.4771	0.2295	0.1971	0.0960
<b>omp</b>	Orthogonal Matching Pursuit	12.4924	266.1812	15.9795	0.4597	0.2281	0.1959	0.1420
<b>dummy</b>	Dummy Regressor	18.9008	490.7326	22.1036	-0.0213	0.3102	0.2905	0.1040

### 3.1.2 K-Fold Cross Validation

La validación cruzada con k-fold cross validation es esencial para evaluar el rendimiento del modelo de manera robusta. En este caso, se divide el conjunto de datos en k (en este contexto, 5) partes iguales, y el modelo se entrena y valida k veces, cada vez utilizando una partición diferente como conjunto de prueba mientras las otras actúan como conjunto de entrenamiento. Esto asegura que el modelo se evalúe de manera integral en diferentes subconjuntos de datos, evitando sesgos debido a la aleatoriedad de la división.

### 3.1.3 Creación y Entrenamiento del Modelo:

Los distintos algoritmos por analizar se eligen modelos iniciales debido a su alto coeficiente de determinación ( $R^2$ ) en la fase de exploración. Este modelo se crea utilizando la biblioteca Pycaret, que facilita la creación y entrenamiento de modelos de ML con una sintaxis simplificada. Durante el entrenamiento, el modelo aprende patrones y relaciones en los datos de entrada y salida. El coeficiente de determinación ( $R^2$ ) es una métrica fundamental que proporciona información sobre la proporción de la varianza en la variable dependiente que es predecible a partir de las variables independientes. En este contexto, un  $R^2$  más alto indica que el modelo tiene una mejor capacidad para explicar la variabilidad en la resistencia a la compresión.

**Tabla 2:** Resultados Random Forest Model. Fuente: Pycaret

	MAE	MSE	RMSE	R2	RMSLE	MAPE
<b>Fold</b>						
<b>0</b>	7.5496	82.0184	9.0564	0.8448	0.1441	0.1161
<b>1</b>	5.7678	66.3342	8.1446	0.8292	0.1190	0.0869
<b>2</b>	11.0774	204.6186	14.3045	0.6154	0.2498	0.2069
<b>3</b>	8.2621	127.4259	11.2883	0.7619	0.1532	0.1142
<b>4</b>	7.9911	117.7713	10.8522	0.7191	0.1428	0.1082
<b>Mean</b>	8.1296	119.6337	10.7292	0.7541	0.1618	0.1265
<b>Std</b>	1.7112	48.0466	2.1255	0.0830	0.0454	0.0416

**Tabla 3:** Resultados Gradient Boosting Regressor Model. Fuente: Pycaret

	MAE	MSE	RMSE	R2	RMSLE	MAPE
<b>Fold</b>						
<b>0</b>	8.4239	102.2492	10.1118	0.8065	0.1697	0.1313
<b>1</b>	7.1737	88.1923	9.3911	0.7729	0.1354	0.1051
<b>2</b>	10.8447	195.7266	13.9902	0.6321	0.2445	0.2026
<b>3</b>	8.7028	132.8846	11.5276	0.7517	0.1712	0.1283
<b>4</b>	8.8882	125.4907	11.2023	0.7006	0.1535	0.1233
<b>Mean</b>	8.8067	128.9087	11.2446	0.7328	0.1749	0.1381
<b>Std</b>	1.1819	37.0423	1.5709	0.0610	0.0372	0.0335

**Tabla 4:** Resultados Extra Tree Regressor Model. Fuente: Pycaret

	MAE	MSE	RMSE	R2	RMSLE	MAPE
<b>Fold</b>						
<b>0</b>	7.3972	89.6735	9.4696	0.8303	0.1550	0.1129
<b>1</b>	5.9073	78.8609	8.8804	0.7969	0.1231	0.0842
<b>2</b>	11.9334	231.8046	15.2251	0.5643	0.2597	0.2198
<b>3</b>	9.5488	165.0336	12.8465	0.6917	0.1718	0.1290
<b>4</b>	8.4914	128.6750	11.3435	0.6930	0.1507	0.1174
<b>Mean</b>	8.6556	138.8095	11.5530	0.7152	0.1721	0.1327
<b>Std</b>	2.0350	55.5818	2.3102	0.0936	0.0465	0.0460

**Tabla 5:** Resultados Extreme Gradient Boosting Model. Fuente: Pycaret

	MAE	MSE	RMSE	R2	RMSLE	MAPE
<b>Fold</b>						
<b>0</b>	9.5537	121.7914	11.0359	0.7696	0.1783	0.1521
<b>1</b>	6.0105	67.2459	8.2004	0.8268	0.1171	0.0888
<b>2</b>	11.8095	236.7189	15.3857	0.5550	0.2621	0.2172
<b>3</b>	9.1829	152.5234	12.3500	0.7151	0.1688	0.1254
<b>4</b>	9.0880	145.9577	12.0813	0.6518	0.1619	0.1229
<b>Mean</b>	9.1289	144.8475	11.8107	0.7037	0.1776	0.1413
<b>Std</b>	1.8490	54.8709	2.3143	0.0943	0.0472	0.0429

**Tabla 6:** Resultados Adaboost. Fuente: Pycaret

	MAE	MSE	RMSE	R2	RMSLE	MAPE
<b>Fold</b>						
<b>0</b>	9.1933	116.5436	10.7955	0.7795	0.1646	0.1373
<b>1</b>	7.9011	106.8387	10.3363	0.7248	0.1577	0.1239
<b>2</b>	12.2067	236.5393	15.3798	0.5554	0.2657	0.2272
<b>3</b>	9.3732	145.0991	12.0457	0.7289	0.1626	0.1296
<b>4</b>	9.9666	147.2940	12.1365	0.6486	0.1657	0.1385
<b>Mean</b>	9.7282	150.4629	12.1388	0.6874	0.1833	0.1513
<b>Std</b>	1.4107	45.8292	1.7645	0.0782	0.0413	0.0383

**Tabla 7:** Resultados Decision Tree Regressor Model. Fuente: Pycaret

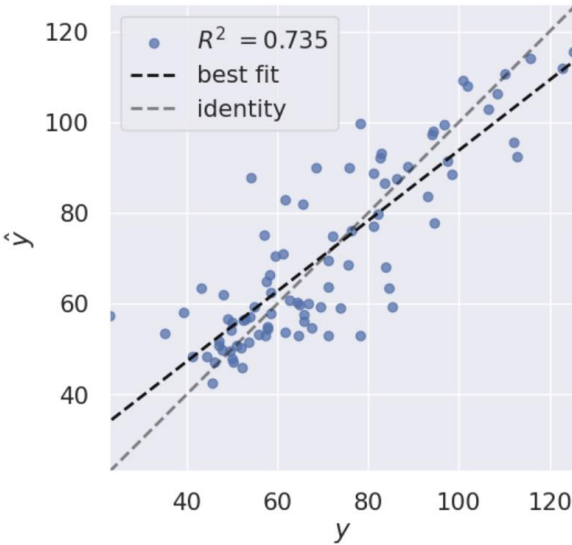
	MAE	MSE	RMSE	R2	RMSLE	MAPE
<b>Fold</b>						
<b>0</b>	10.6810	183.1863	13.5346	0.6534	0.2073	0.1621
<b>1</b>	10.9252	206.7705	14.3795	0.4674	0.2277	0.1528
<b>2</b>	12.4352	283.0402	16.8238	0.4679	0.2723	0.2202
<b>3</b>	10.6383	174.1817	13.1978	0.6746	0.1886	0.1511
<b>4</b>	9.8552	164.4296	12.8230	0.6077	0.1860	0.1466
<b>Mean</b>	10.9070	202.3217	14.1518	0.5742	0.2164	0.1666
<b>Std</b>	0.8444	42.7296	1.4317	0.0896	0.0317	0.0273

### 3.1.4 Predicción del Modelo

Una vez creado y entrenado, los distintos modelos se utilizan para realizar predicciones en un conjunto de datos de prueba independiente. Estas predicciones representan las estimaciones de la resistencia a la compresión para las observaciones no vistas durante el entrenamiento.

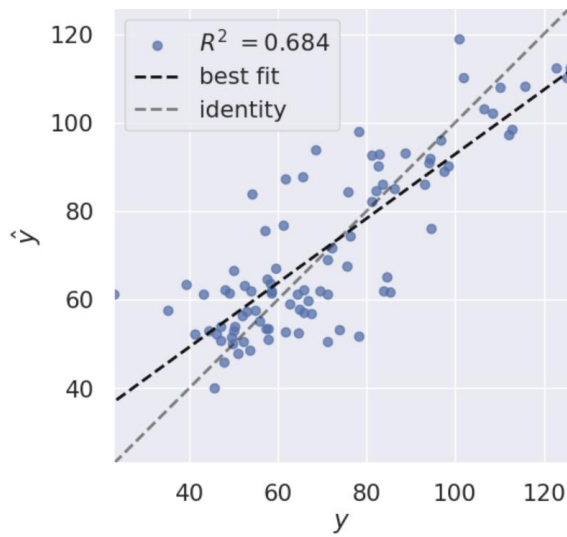
Se representa un modelo específico, Al ejecutar el código se genera una visualización que muestra cómo varía el error de los modelos en el conjunto de entrenamiento a

medida que se aumenta el tamaño del conjunto de datos. Esto puede ser útil para comprender cómo se comporta el modelo en diferentes tamaños de datos y para identificar posibles problemas de sobreajuste o subajuste.

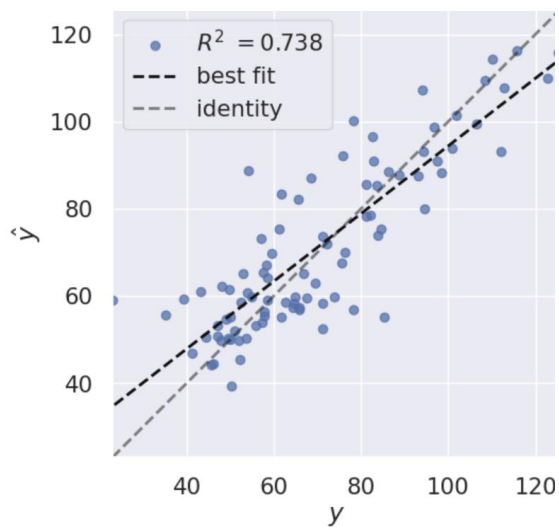


**Figura 2:** Error de predicción para Random Forest Regressor. Fuente: Pycaret

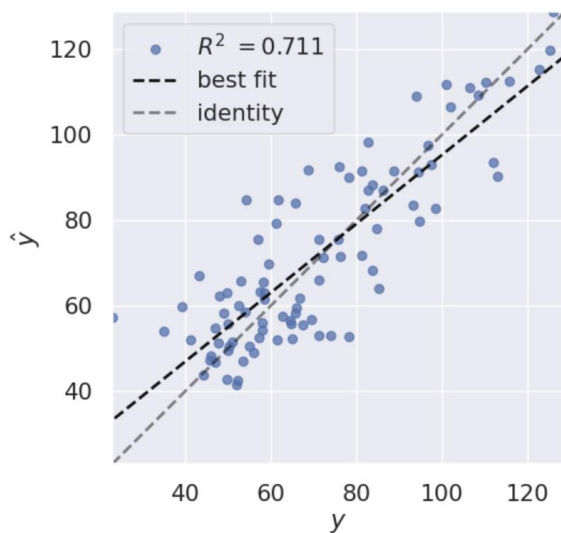




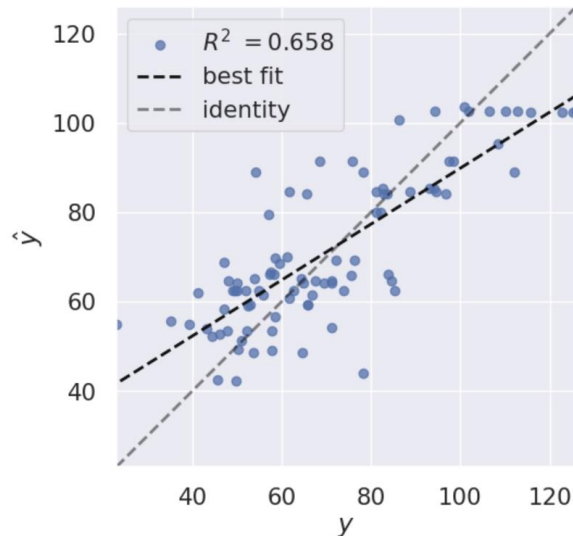
**Figura 3:** Error de predicción para Gradient Boosting Regressor. Fuente: Pycaret



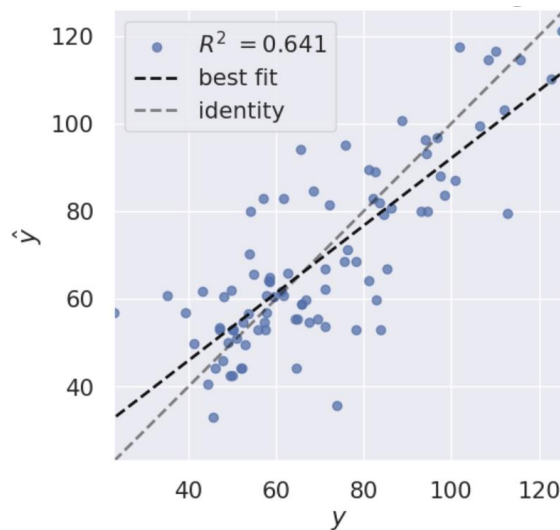
**Figura 4:** Error de predicción para Extra Tree Regressor. Fuente: Pycaret



**Figura 5:** Error de predicción para XGB Regressor. Fuente: Pycaret



**Figura 6:** Error de predicción para AdaBoost Regressor. Fuente: Pycaret



**Figura 7:** Error de predicción para Decision Tree Regressor. Fuente: Pycaret

### 3.2 Optimización de Hiperparámetros

La optimización de hiperparámetros es una fase crítica para mejorar el rendimiento del modelo. Utilizando técnicas como búsqueda en cuadrícula o búsqueda aleatoria, se exploran diferentes combinaciones de hiperparámetros para el modelo. El objetivo es encontrar la configuración óptima que maximice el rendimiento según métricas específicas, como el coeficiente de determinación ( $R^2$ ) o el error cuadrático medio (MSE).

**Tabla 8:** Resultados Random Forest Optimizado. Fuente: Pycaret

	MAE	MSE	RMSE	R2	RMSLE	MAPE
<b>Fold</b>						
<b>0</b>	7.6547	77.2917	8.7916	0.8538	0.1393	0.1177
<b>1</b>	5.9239	73.1077	8.5503	0.8117	0.1288	0.0907
<b>2</b>	11.4302	211.0949	14.5291	0.6032	0.2532	0.2123
<b>3</b>	7.9408	119.1917	10.9175	0.7773	0.1467	0.1092
<b>4</b>	7.9667	114.1011	10.6818	0.7278	0.1405	0.1098
<b>Mean</b>	8.1833	118.9574	10.6941	0.7548	0.1617	0.1279
<b>Std</b>	1.7907	49.7015	2.1435	0.0863	0.0461	0.0431

**Tabla 9:** Resultados Gradient Boosting Optimizado. Fuente: Pycaret

	MAE	MSE	RMSE	R2	RMSLE	MAPE
<b>Fold</b>						
<b>0</b>	7.8179	77.6812	8.8137	0.8530	0.1397	0.1189
<b>1</b>	7.1557	79.9266	8.9402	0.7941	0.1357	0.1099
<b>2</b>	11.2452	216.3328	14.7083	0.5933	0.2579	0.2134
<b>3</b>	8.2904	116.6077	10.7985	0.7822	0.1479	0.1172
<b>4</b>	8.8140	125.5305	11.2040	0.7005	0.1534	0.1208
<b>Mean</b>	8.6646	123.2158	10.8929	0.7446	0.1669	0.1360
<b>Std</b>	1.4011	50.3332	2.1354	0.0899	0.0459	0.0388

**Tabla 10:** Resultados Extra Tree Regressor Optimizado. Fuente: Pycaret

	MAE	MSE	RMSE	R2	RMSLE	MAPE
<b>Fold</b>						
<b>0</b>	7.2458	80.1678	8.9536	0.8483	0.1427	0.1123
<b>1</b>	6.1316	73.1554	8.5531	0.8116	0.1289	0.0925
<b>2</b>	11.4801	217.1811	14.7371	0.5917	0.2589	0.2181
<b>3</b>	7.9525	118.2418	10.8739	0.7791	0.1451	0.1091
<b>4</b>	8.0398	101.4621	10.0728	0.7580	0.1318	0.1098
<b>Mean</b>	8.1700	118.0416	10.6381	0.7577	0.1615	0.1284
<b>Std</b>	1.7906	52.0629	2.2073	0.0884	0.0491	0.0454

**Tabla 11:** Resultados Extreme Gradient Boosting Optimizado. Fuente: Pycaret

	MAE	MSE	RMSE	R2	RMSLE	MAPE
<b>Fold</b>						
<b>0</b>	7.5908	79.5608	8.9197	0.8495	0.1431	0.1159
<b>1</b>	6.7899	83.2789	9.1257	0.7855	0.1364	0.1026
<b>2</b>	11.1325	198.5759	14.0917	0.6267	0.2462	0.2075
<b>3</b>	8.0417	110.2628	10.5006	0.7940	0.1475	0.1156
<b>4</b>	9.1785	127.1485	11.2760	0.6967	0.1493	0.1264
<b>Mean</b>	8.5467	119.7654	10.7827	0.7505	0.1645	0.1336
<b>Std</b>	1.5060	43.1360	1.8702	0.0789	0.0411	0.0377

**Tabla 12:** Resultados Adaboost Optimizado. Fuente: Pycaret

	MAE	MSE	RMSE	R2	RMSLE	MAPE
<b>Fold</b>						
<b>0</b>	9.1943	111.0270	10.5369	0.7899	0.1560	0.1335
<b>1</b>	7.3574	96.7723	9.8373	0.7508	0.1467	0.1124
<b>2</b>	11.6017	219.5804	14.8182	0.5872	0.2573	0.2159
<b>3</b>	8.1039	121.2548	11.0116	0.7735	0.1468	0.1111
<b>4</b>	9.4599	138.8961	11.7854	0.6687	0.1576	0.1293
<b>Mean</b>	9.1435	137.5061	11.5979	0.7140	0.1729	0.1404
<b>Std</b>	1.4434	43.2727	1.7306	0.0759	0.0425	0.0388

**Tabla 13:** Resultados Decision Tree Regressor Optimizado. Fuente: Pycaret

	MAE	MSE	RMSE	R2	RMSLE	MAPE
<b>Fold</b>						
<b>0</b>	9.0666	114.5112	10.7010	0.7833	0.1631	0.1350
<b>1</b>	7.2760	92.9292	9.6400	0.7607	0.1439	0.1077
<b>2</b>	12.5620	265.5654	16.2962	0.5008	0.2633	0.2194
<b>3</b>	9.5752	144.9587	12.0399	0.7292	0.1604	0.1296
<b>4</b>	8.0140	115.5902	10.7513	0.7243	0.1552	0.1139
<b>Mean</b>	9.2988	146.7109	11.8857	0.6996	0.1772	0.1411
<b>Std</b>	1.8178	61.6883	2.3328	0.1017	0.0435	0.0404

En resumen, el proceso completo implica la entrada y preprocesamiento de datos, validación cruzada para evaluar el rendimiento del modelo, creación y entrenamiento del modelo elegido, predicciones en un conjunto de prueba y, finalmente, la optimización de hiperparámetros para mejorar aún más la precisión del modelo en la predicción de la resistencia a la compresión de la pasta de cemento con adiciones minerales.

### 3.3 Comparación

Al comparar modelos de regresión, especialmente en el contexto de machine learning, es crucial evaluar diversas métricas para obtener una visión completa del rendimiento de cada modelo.

#### 3.3.1 MAE (Error Absoluto Medio)

Interpretación: Cuanto menor sea el MAE, mejor será el modelo. Indica la magnitud promedio de los errores en las predicciones.

**Random Forest Regressor:** El MAE disminuyó ligeramente después de la optimización de hiperparámetros, lo que indica que el modelo mejoró su capacidad de generalización y ajuste a los datos. Esto sugiere que los hiperparámetros seleccionados durante la optimización probablemente ayudaron a reducir el sesgo y la varianza del modelo.

**Gradient Boosting Regressor:** Aunque el MAE disminuyó después de la optimización de hiperparámetros, la mejora no fue tan significativa en comparación con otros modelos. Esto podría indicar que el modelo ya estaba bastante bien ajustado en su forma original, y los cambios realizados durante la optimización tuvieron un impacto limitado.

**Extra Tree Regressor:** Este modelo mostró una mejora notable en el MAE después de la optimización de hiperparámetros, lo que sugiere que los cambios realizados durante la optimización ayudaron significativamente a mejorar la capacidad predictiva del modelo.

**Extreme Gradient Boosting (XGBoost):** Similar al Gradient Boosting Regressor, el XGBoost mostró una disminución en el MAE después de la optimización de

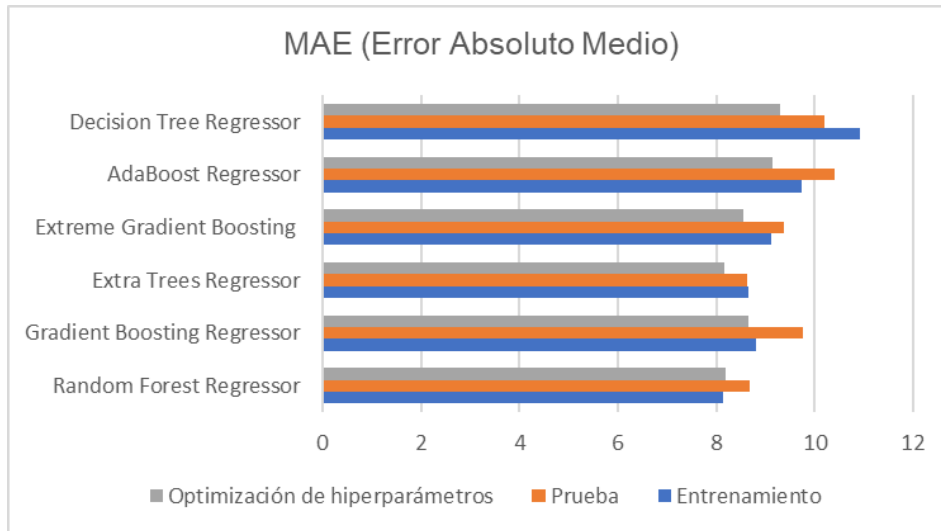
hiperparámetros, aunque la mejora no fue tan pronunciada. Esto podría indicar que el modelo ya estaba relativamente bien ajustado antes de la optimización.

**AdaBoost Regressor:** El MAE disminuyó de manera significativa después de la optimización de hiperparámetros, lo que sugiere que la selección adecuada de hiperparámetros ayudó a mejorar considerablemente la capacidad de generalización del modelo.

**Decision Tree Regressor:** Este modelo mostró una mejora sustancial en el MAE después de la optimización de hiperparámetros, lo que indica que los cambios realizados durante la optimización tuvieron un impacto significativo en la capacidad predictiva del modelo.

**Tabla 14:** Comparativo Error Absoluto Medio. Fuente: Elaboración propia

ALGORITMO/ PARAMETRO	MAE (Error Absoluto Medio)		
	Entrenamiento	Prueba	Optimización de hiperparámetros
Random Forest Regressor	8,13	8,68	8,18
Gradient Boosting Regressor	8,81	9,75	8,66
Extra Trees Regressor	8,66	8,63	8,17
Extreme Gradient Boosting	9,13	9,37	8,55
AdaBoost Regressor	9,73	10,41	9,14
Decision Tree Regressor	10,91	10,21	9,30



**Figura 8:** Comparativo Error Absoluto Medio. Fuente: Elaboración Propia

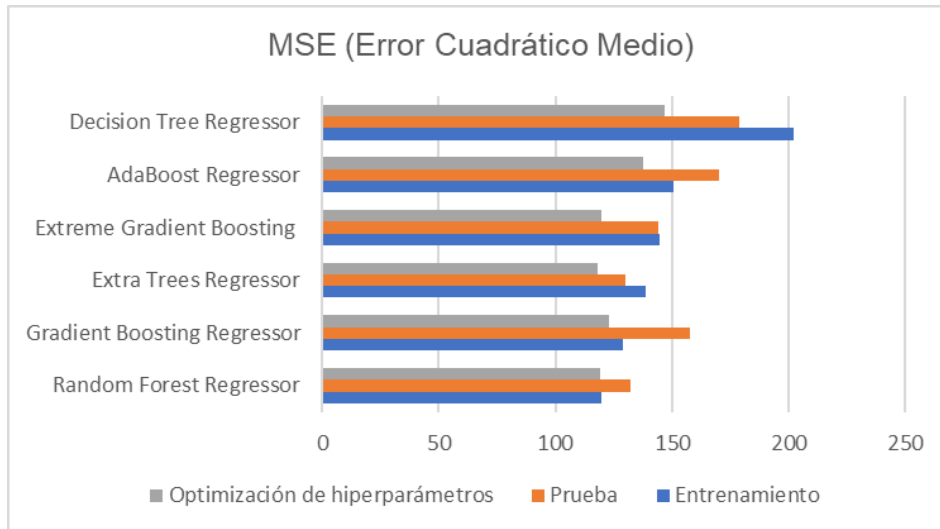
En resumen, la optimización de hiperparámetros puede tener un impacto variable en diferentes modelos de regresión. Algunos modelos pueden beneficiarse significativamente de la optimización, mientras que otros pueden mostrar mejoras más modestas. Es importante realizar una evaluación detallada del rendimiento del modelo antes y después de la optimización para determinar el impacto real de los cambios realizados durante este proceso.

### 3.3.2 MSE (Error Cuadrático Medio)

Interpretación: Al igual que MAE, se busca minimizar el MSE. Penaliza más fuertemente los errores grandes.

**Tabla 15:** Comparativo Error Cuadrático Medio. Fuente: Elaboración propia

ALGORITMO/ PARAMETRO	MSE (Error Cuadrático Medio)		
	Entrenamiento	Prueba	Optimización de hiperparámetros
Random Forest Regressor	119,63	132,13	118,96
Gradient Boosting Regressor	128,91	157,52	123,22
Extra Trees Regressor	138,81	130,19	118,04
Extreme Gradient Boosting	144,85	144,06	119,77
AdaBoost Regressor	150,46	170,21	137,510
Decision Tree Regressor	202,32	178,58	146,71



**Figura 9:** Comparativo Error Cuadrático Medio. Fuente: Elaboración Propia

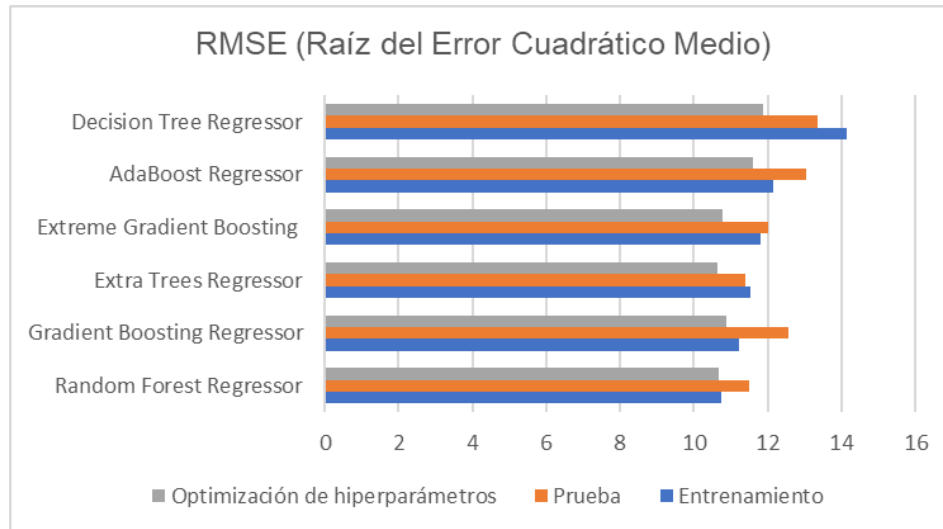
### 3.3.3 RMSE (Raíz del Error Cuadrático Medio)

Interpretación: Similar a MSE, pero interpretable en la misma unidad que la variable objetivo.

**Tabla 16:** Comparativo Raíz del Error Cuadrático Medio. Fuente: Elaboración propia

ALGORITMO/ PARAMETRO	RMSE (Raíz del Error Cuadrático Medio)		
	Entrenamiento	Prueba	Optimización de hiperparámetro
Random Forest Regressor	10,73	11,49	10,69
Gradient Boosting Regressor	11,24	12,55	10,89
Extra Trees Regressor	11,55	11,41	10,64
Extreme Gradient Boosting	11,81	12,00	10,78
AdaBoost Regressor	12,14	13,05	11,600
Decision Tree Regressor	14,15	13,36	11,89





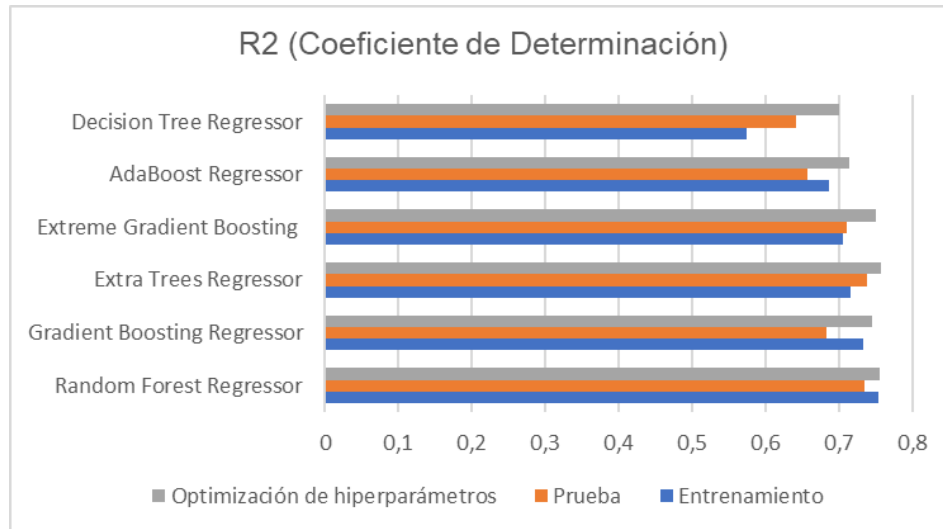
**Figura 10:** Comparativo Raíz del Error Cuadrático Medio. Fuente: Elaboración Propia

### 3.3.4 R<sup>2</sup> (Coeficiente de Determinación)

Interpretación: Cuanto más cercano a 1 sea el R<sup>2</sup>, mejor es el modelo. Representa la calidad general de las predicciones.

**Tabla 17:** Comparativo Coeficiente de Determinación. Fuente: Elaboración propia

ALGORITMO/ PARAMETRO	R <sup>2</sup> (Coeficiente de Determinación)		
	Entrenamiento	Prueba	Optimización de hiperparámetros
Random Forest Regressor	0,7541	0,7345	0,7548
Gradient Boosting Regressor	0,7328	0,6835	0,7446
Extra Trees Regressor	0,7152	0,7384	0,7577
Extreme Gradient Boosting	0,7057	0,7106	0,7505
AdaBoost Regressor	0,6874	0,658	0,714
Decision Tree Regressor	0,5742	0,6412	0,6996



**Figura 11:** Comparativo Coeficiente de Determinación. Fuente: Elaboración Propia

**Random Forest Regressor:** El  $R^2$  muestra una mejora después de la optimización de hiperparámetros, lo que indica que los cambios realizados durante este proceso ayudaron a mejorar la capacidad de generalización del modelo.

**Gradient Boosting Regressor:** Aunque el  $R^2$  aumenta ligeramente después de la optimización, la mejora no es tan significativa. Sin embargo, el  $R^2$  indica una capacidad de generalización razonable después de la optimización.

**Extra Trees Regressor:** El  $R^2$  muestra una mejora considerable, la más representativa después de la optimización, lo que indica que los cambios realizados durante este proceso beneficiaron significativamente la capacidad de generalización del modelo.

**Extreme Gradient Boosting (XGBoost):** Similar al Extra Trees Regressor, el  $R^2$  muestra una mejora considerable después de la optimización, lo que sugiere una mejora significativa en la capacidad de generalización del modelo.

**AdaBoost Regressor:** El  $R^2$  también muestra una mejora después de la optimización, lo que indica que los cambios realizados durante este proceso ayudaron a mejorar la capacidad de generalización del modelo.

**Decision Tree Regressor:** Aunque el  $R^2$  aumenta después de la optimización, sigue siendo el más bajo entre todos los modelos. Esto sugiere que el modelo Decision Tree

podría beneficiarse de más ajustes en los hiperparámetros o podría ser menos adecuado para el conjunto de datos en comparación con otros modelos.

En resumen, la optimización de hiperparámetros contribuyó a mejorar la capacidad de generalización de la mayoría de los modelos, como se refleja en el aumento del coeficiente de determinación  $R^2$  después de la optimización. Esto indica que los modelos fueron capaces de explicar una mayor proporción de la variabilidad en los datos de prueba después de la optimización.

### 3.3.5 RMSLE (Error Cuadrático Medio del Logaritmo de las Predicciones)

Interpretación: Es útil cuando las diferencias porcentuales son más relevantes que las diferencias absolutas.

El RMSLE se utiliza para evaluar la precisión de las predicciones en escala logarítmica. En general, todos los modelos muestran una disminución en el RMSLE después de la optimización de hiperparámetros, lo que indica una mejora en la precisión de las predicciones.

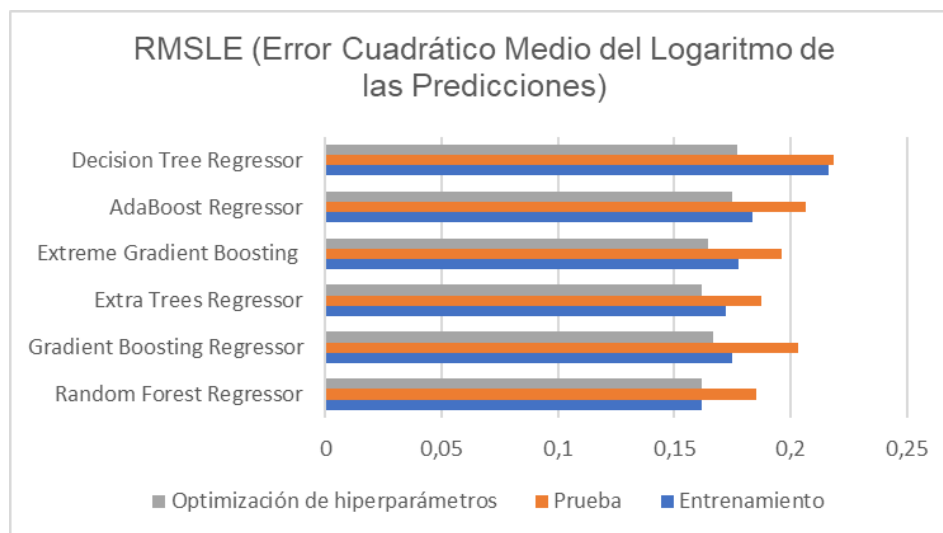
Extra Trees Regressor tiene el RMSLE más bajo después de la optimización de hiperparámetros, seguido de Random Forest Regressor y Extreme Gradient Boosting.

Decision Tree Regressor tiene el RMSLE más alto, incluso después de la optimización de hiperparámetros, lo que sugiere que podría beneficiarse de ajustes adicionales o puede ser menos adecuado para el conjunto de datos en comparación con otros modelos.

**Tabla 18:** Comparativo Error Cuadrático Medio del Logaritmo de las Predicciones. Fuente: Elaboración propia

ALGORITMO/ PARAMETRO	RMSLE (Error Cuadrático Medio del Logaritmo de las Predicciones)		
	Entrenamiento	Prueba	Optimización de hiperparámetros
Random Forest Regressor	0,1618	0,1854	0,1617
Gradient Boosting Regressor	0,1749	0,2033	0,1669
Extra Trees Regressor	0,1721	0,1876	0,1615
Extreme Gradient Boosting	0,1776	0,1958	0,1645
AdaBoost Regressor	0,1833	0,2064	0,175

<b>Decision Tree Regressor</b>	0,2164	0,2186	0,1772
--------------------------------	--------	--------	--------



**Figura 12:** Comparativo Error Cuadrático Medio del Logaritmo de las Predicciones. Fuente: Elaboración Propia

En resumen, la optimización de hiperparámetros juega un papel crucial en mejorar la precisión de los modelos de regresión en términos de RMSLE. La selección cuidadosa de los parámetros es esencial para obtener resultados precisos y mejorar la capacidad de generalización de los modelos. Además, algunos modelos pueden ser más sensibles que otros a los ajustes de los hiperparámetros, lo que destaca la importancia de realizar un ajuste fino durante el proceso de optimización.

### 3.3.6 MAPE (Error Porcentual Absoluto Medio)

Interpretación: Cuanto menor sea el MAPE, mejor será el modelo. Mide la precisión porcentual promedio.

**Random Forest Regressor (RF):** RF muestra un MAPE inicialmente bajo en la creación del modelo (0.127), lo que indica una buena precisión en las predicciones. Durante el entrenamiento, el MAPE aumenta ligeramente (0.142), pero vuelve a disminuir después de la optimización de hiperparámetros (0.128), lo que sugiere que la optimización mejoró la precisión del modelo.

Importante: RF ofrece un rendimiento sólido en términos de MAPE, lo que lo convierte en una opción confiable para la predicción precisa en este conjunto de datos.

**Gradient Boosting Regressor (GB):** GB muestra un MAPE inicial relativamente bajo en la creación del modelo (0.138). Durante el entrenamiento, el MAPE aumenta (0.162), pero luego disminuye después de la optimización de hiperparámetros (0.136), lo que indica que la optimización mejoró la precisión del modelo.

Importante: La optimización de hiperparámetros es crucial para mejorar el rendimiento de GB, lo que sugiere que la selección cuidadosa de los parámetros es fundamental para obtener resultados precisos.

**Extra Trees Regressor (ETR):** ETR muestra un MAPE inicialmente bajo en la creación del modelo (0.133), que aumenta ligeramente durante el entrenamiento (0.145), pero luego disminuye significativamente después de la optimización de hiperparámetros (0.128).

Importante: La optimización de hiperparámetros es fundamental para mejorar la precisión de ETR, lo que indica que la selección adecuada de los parámetros puede conducir a una mejor capacidad de generalización.

**Extreme Gradient Boosting (XGBoost):** XGBoost muestra un MAPE inicial relativamente bajo en la creación del modelo (0.141), que aumenta durante el entrenamiento (0.157), pero luego disminuye después de la optimización de hiperparámetros (0.134).

Importante: La optimización de hiperparámetros es esencial para mejorar la precisión de XGBoost, lo que sugiere que la afinación cuidadosa de los parámetros puede conducir a una mejora significativa en la capacidad de generalización.

**AdaBoost Regressor (AB):** AB muestra un MAPE inicialmente moderado en la creación del modelo (0.151), que aumenta durante el entrenamiento (0.171), pero luego disminuye después de la optimización de hiperparámetros (0.140).

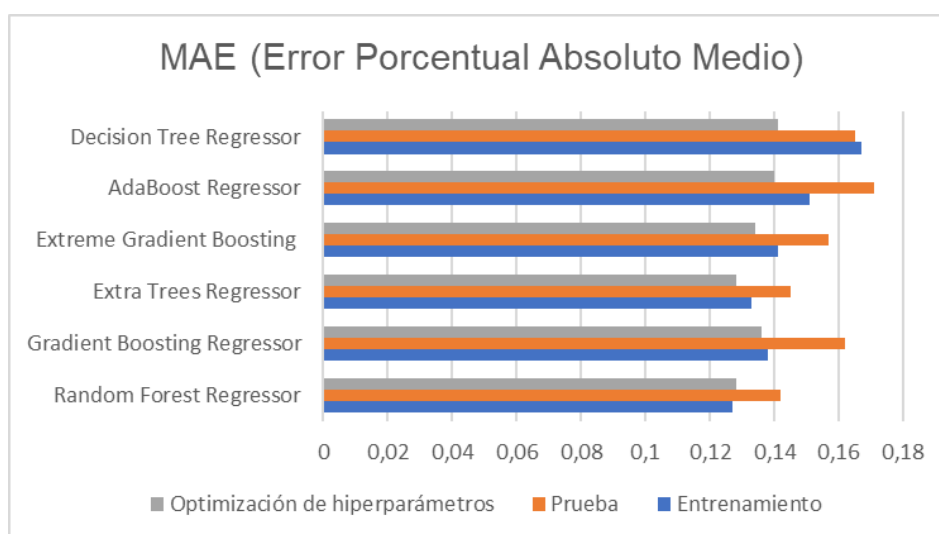
Importante: La optimización de hiperparámetros es crucial para mejorar la precisión de AB, lo que sugiere que la selección adecuada de los parámetros puede conducir a una mejor capacidad de generalización.

**Decision Tree Regressor (DT):** DT muestra un MAPE relativamente alto en la creación del modelo (0.167), que disminuye durante el entrenamiento (0.165), pero luego disminuye aún más después de la optimización de hiperparámetros (0.141).

Importante: A pesar de la disminución del MAPE después de la optimización, DT sigue teniendo el MAPE más alto entre todos los modelos. Esto sugiere que podría requerir más ajustes o ser menos adecuado para el conjunto de datos en comparación con otros modelos.

**Tabla 19:** Comparativo Error Porcentual Absoluto Medio. Fuente: Elaboración propia

ALGORITMO/ PARAMETRO	MAPE (Error Porcentual Absoluto Medio)		
	Entrenamiento	Prueba	Optimización de hiperparámetro
Random Forest Regressor	0,127	0,142	0,128
Gradient Boosting Regressor	0,138	0,162	0,136
Extra Trees Regressor	0,133	0,145	0,128
Extreme Gradient Boosting	0,141	0,157	0,134
AdaBoost Regressor	0,151	0,171	0,140
Decision Tree Regressor	0,167	0,165	0,141



**Figura 13:** Comparativo Error Porcentual Absoluto Medio. Fuente: Elaboración Propia

En resumen, la optimización de hiperparámetros juega un papel crucial en mejorar la precisión de los modelos de regresión en términos de MAPE. La selección cuidadosa de

los parámetros es esencial para obtener resultados precisos y mejorar la capacidad de generalización de los modelos. Además, algunos modelos pueden ser más sensibles que otros a los ajustes de los hiperparámetros, lo que destaca la importancia de realizar un ajuste fino durante el proceso de optimización

Al comparar modelos, se busca un equilibrio entre estas métricas, considerando la naturaleza específica del problema. Una elección bien fundamentada implicaría la revisión de todas estas métricas en conjunto y la ponderación de la importancia relativa de cada una según los objetivos específicos del proyecto. La elección del "mejor" modelo dependerá de cómo estas métricas se alineen con los requisitos y metas del problema de predicción de la resistencia a la compresión en pasta de cemento con adiciones minerales.

### **3.4 Análisis de resultados**

- **Valores Iniciales**

Estos valores iniciales proporcionan una referencia del rendimiento del modelo antes de cualquier entrenamiento u optimización de hiperparámetros.

La literatura existente respalda la eficacia de Random Forest en problemas de regresión debido a su capacidad para manejar múltiples características y reducir el sobreajuste. La combinación de árboles de decisión correlacionados y la aleatorización en la selección de características lo hace robusto.

Mientras que modelos como Gradient Boosting y XGBoost también son eficaces, Random Forest a menudo se destaca por su rendimiento equilibrado y su capacidad para manejar datos ruidosos y complejos.

- **Después del Entrenamiento**

Después del entrenamiento, se observa un aumento en el error absoluto medio (MAE) y el error cuadrático medio (MSE), lo que indica que el modelo puede haber experimentado un ligero sobreajuste durante el entrenamiento. La disminución en el coeficiente de determinación ( $R^2$ ) también sugiere una posible reducción en la capacidad de generalización del modelo. Es esencial destacar que la elección de Random Forest Regressor no solo se basa en el  $R^2$  más alto, sino también en una evaluación integral que considera otras métricas estadísticas como el Error Absoluto Medio (MAE), Error

Cuadrático Medio (MSE), Raíz del Error Cuadrático Medio (RMSE), Raíz Cuadrada del Error Cuadrático Logarítmico (RMSLE), Error Porcentual Absoluto Medio (MAPE), entre otras. La elección de Random Forest Regressor se basa en un análisis exhaustivo y en la búsqueda de un equilibrio entre precisión y generalización.

- **Después de la Optimización de Hiperparámetros**

La optimización de hiperparámetros ha mejorado significativamente el rendimiento del modelo en términos de métricas clave.

La disminución sustancial en MAE y MSE, así como el aumento en  $R^2$ , indican una mejora en la capacidad de predicción y generalización del modelo.

El análisis de folds revela cierta variabilidad en el rendimiento entre diferentes subconjuntos de datos, destacando la importancia de la validación cruzada para evaluar la robustez del modelo.

Aunque la media de las métricas se mantiene en niveles similares a los valores iniciales, la mejora en algunos folds sugiere que el modelo es capaz de realizar predicciones precisas en conjuntos de datos específicos.

Considerando los 2 modelos de regresión que presentan mejores valores dentro de los parámetros estadísticos, se realiza una comparación de los mismos. (Random Forest y Extra Tree Regressor)

- MAE (Error Absoluto Medio): Random Forest presenta valores menores en las fases (entrenamiento y prueba) comparado con Extra Tree Regressor, que presenta un menor valor en la optimización, indicando una menor diferencia promedio entre las predicciones y los valores reales.
- MSE (Error Cuadrático Medio): Similar a MAE, Random Forest muestra valores menores en las dos fases, reflejando una menor dispersión de los errores. Mientras que Extra Tree Regressor presenta el valor menor luego de la optimización de hiperparámetros.
- RMSE (Raíz del Error Cuadrático Medio): Random Forest tiene valores más bajos en las fases de entrenamiento y prueba, indicando una menor magnitud promedio de los errores. Mientras que Extra Tree Regressor presenta el valor menor luego



de la optimización de hiperparámetros, lo que indica un mejor ajuste de este parámetro.

- $R^2$  (Coeficiente de Determinación): En la optimización de hiperparámetros, Extra Tree Regressor tiene un  $R^2$  ligeramente superior (0.7577 contra 0.7548 en Random Forest).
- RMSLE (Error Cuadrático Medio del Logaritmo de las Predicciones): Random Forest presenta valores menores en las dos primeras fases de entrenamiento y prueba, indicando una mejor precisión en la predicción de valores en una escala logarítmica, pero con la optimización de hiperparámetros Extra Tree Regressor se ajusta de mejor manera.
- MAPE (Error Porcentual Absoluto Medio): Random Forest tiene valores menores en todas las fases, indicando una menor discrepancia porcentual entre las predicciones y los valores reales.

**Tabla 20:** Random Forest Regressor VS Extra Trees Regressor. Fuente: Elaboración propia

ALGORITMO	Random Forest Regressor			Extra Trees Regressor		
FASE/ PARÁMETRO	Entrenam iento	Prueba	Optimización Hiperparáme tros	Entrenami ento	Prueba	Optimizació n Hiperparám etros
MAE	8,1296	8,6797	8,1833	8,6556	8,6310	8,1700
MSE	119,633	132,1322	118,9574	138,8095	130,1898	118,0416
RMSE	10,7292	11,4949	10,6941	11,5530	11,4101	10,6381
R2	0,7541	0,7345	0,7548	0,7152	0,7384	0,7577
RMSLE	0,1618	0,1854	0,1617	0,1721	0,1876	0,1615
MAPE	0,1265	0,1424	0,1279	0,1327	0,1453	0,1284

Extra Tree Regressor tiene un  $R^2$  ligeramente superior y basándonos en las métricas de errores estadísticos (MAE, MSE, RMSE, RMSLE, MAPE), ofrece un rendimiento general más consistente y preciso en la predicción de la resistencia a la compresión. Por lo tanto,

la decisión final se inclina hacia Extra Tree Regressor. Sin embargo, es recomendable realizar validaciones adicionales y considerar la naturaleza específica de los datos y los requisitos del problema antes de tomar una decisión definitiva. En conclusión, el enfoque técnico implementado, junto con la evaluación de métricas y la optimización de hiperparámetros, respalda la elección del Extra Tree Regressor como el modelo preferido en esta aplicación específica. Este análisis proporciona una base sólida para la toma de decisiones en la selección de modelos de regresión para la predicción de la resistencia a la compresión en pasta de cemento con adiciones minerales.

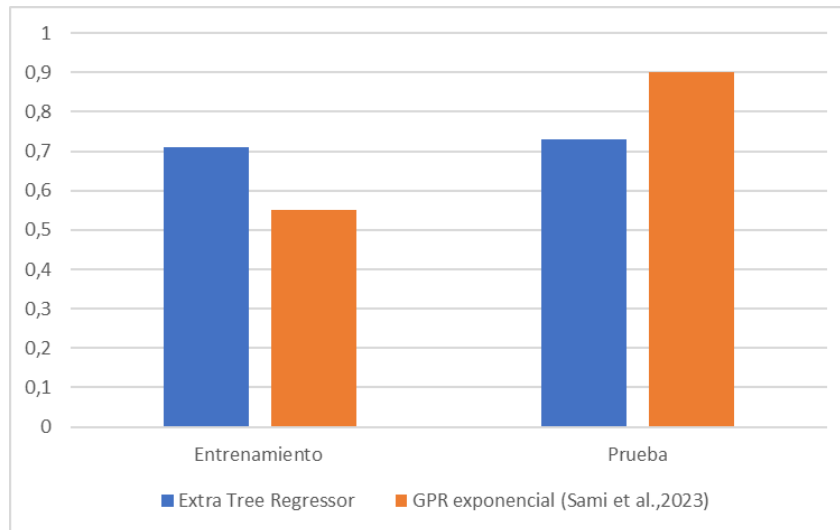
### **3.5 Discusiones**

Para fundamentar el desempeño del modelo ML propuesto sobre modelos implementados en otras literaturas, se realizaron análisis comparativos. (Sami et al., 2023), utilizó el repositorio de aprendizaje automático de la Universidad de California, Irvine (UCI) para entrenar los modelos de aprendizaje automático (ML) para la resistencia a la compresión; a lo largo de esta investigación se utilizaron algoritmos de aprendizaje automático que incluyen modelos de regresión, modelos de regresión de árbol, regresión de vector de soporte (SVR), regresión de conjunto (ER) y regresión de proceso gaussiano (GPR) para predecir la resistencia a la compresión.

En el caso de Sami et al., 2023 el GPR exponencial fue el modelo de mayor rendimiento para predecir la resistencia a la compresión, generando un coeficiente de determinación ( $R^2$ ) en la fase de entrenamiento del modelo de 0.55 y en la fase de prueba de 0.90; En lo que respecta al presente estudio con el modelo Extra Tree Regressor se obtuvo un coeficiente de determinación ( $R^2$ ) en la fase de entrenamiento de 0.71 y en la fase de prueba de 0.73, con lo cual podemos concluir que el modelo Extra Tree Regressor es mucho más eficiente para la predicción de la resistencia a la compresión durante la fase de creación, pero durante la fase de entrenamiento el modelo de Sami et al., genera mejores resultados en función de las variables de ingreso mencionadas en los apartados anteriores.

**Tabla 21:** Comparativo Extra Tree Regressor Vs GPR Exponencial. Fuente: Elaboración propia

ALGORITMO / FASE	Entrenamiento	Prueba
Extra Tree Regressor	0,71	0,73
GPR exponencial (Sami et al.,2023)	0,55	0,9



**Figura 14:** Comparativo Extra Tree Regressor Vs GPR Exponencial. Fuente: Elaboración Propia  
Cabe mencionar que, en este caso puntual solo se han analizado cinco (5) variables de entrada, pero tal como menciona Sami et al., 2023 el modelo GPR exponencial mejora sus resultados al incrementar el número de variables de entrada, tal es el caso puntual en la cual se incluye 8 variables (cemento, escoria de alto horno, cenizas volantes, superplastificante, edad en días, índice de agregado grueso, contenido de agregado fino y contenido de agua) y el valor del coeficiente de determinación ( $R^2$ ) en la fase de entrenamiento de 0.95 y en la fase de prueba de 0.89.

Otra comparativa realizada con la literatura existente, Nguyen et al., 2023 prevé la resistencia a compresión (CS) de un hormigón ultra dúctil utilizando 15 variables de entrada, evaluando criterios estadísticos de rendimiento como RMSE, MAE y  $R^2$  y como única variable de salida la CS, todo esto a través de simulaciones cross validation de 10 veces, analizando el rendimiento del algoritmo XGBOOST y sus variables, obteniendo resultados estadísticos muy prometedores tanto en los conjuntos de datos de entrenamiento como en los de prueba. Los valores calculados de RMSE, MAE y  $R^2$  en entrenamiento fueron 7,238; 5,387 y 0,969 respectivamente y en los de prueba 4,051; 2,716 y 0,99; valores de  $R^2$  muy cercanos a 1, lo que significa un ajuste lineal ideal.

Comparando con nuestros valores de R2 mejor adaptados EGR tanto en entrenamiento como en prueba de 0,71 y 0,73 existe una amplia diferencia, la que se debe principalmente por el número de variables de ingreso, en este caso de 5 vs 15, existiendo una relación directamente proporcional con el número de variables de ingreso vs el ajuste optimizado de los parámetros, pudiendo prever de mejor manera la característica estudiada.

**Tabla 22:** Comparativo Extra Tree Regressor Vs XGBoost. Fuente: Elaboración propia

ALGORITMO / FASE	Entrenamiento	Prueba
Extra Tree Regressor	0,71	0,73
XGBOOST(Nguyen et al.,2023)	0,969	0,99



**Figura 15:** Comparativo Extra Tree Regressor Vs XGBoost. Fuente: Elaboración Propia

Podemos observar los valores que desde el entrenamiento se obtienen parámetros con valores excelentes, debido a que se llevaron a cabo pruebas de error, donde la estrategia fue comenzar con los hiperparámetros de los valores predeterminados del modelo y luego ajustar cada parámetro individualmente aumentando la precisión de la predicción, a comparación de nuestro modelo que realiza el entrenamiento a través de simulación de cross validation de 5 veces con datos iniciales no hiperparametrizados, lo que se realiza después del entrenamiento y prueba, donde a medida que avanzan las fases se ve una mejora en los resultados estadísticos.

# CAPÍTULO 4

## 4. CONCLUSIONES Y RECOMENDACIONES

En esta investigación, se evaluaron varios modelos para pronosticar la resistencia a la compresión en pasta de cemento con adiciones minerales en la industria de la construcción. Los modelos se entrenaron utilizando datos (variables de entrada) obtenidos de la literatura de M.Cornejo 2014. El conjunto de variables de entrada usadas para entrenar los modelos se compone de 2100 datos clasificados según los parámetros de diseño como %SCM, Tipo SCM, Tipo de curado, Envejecimiento y Metodología.

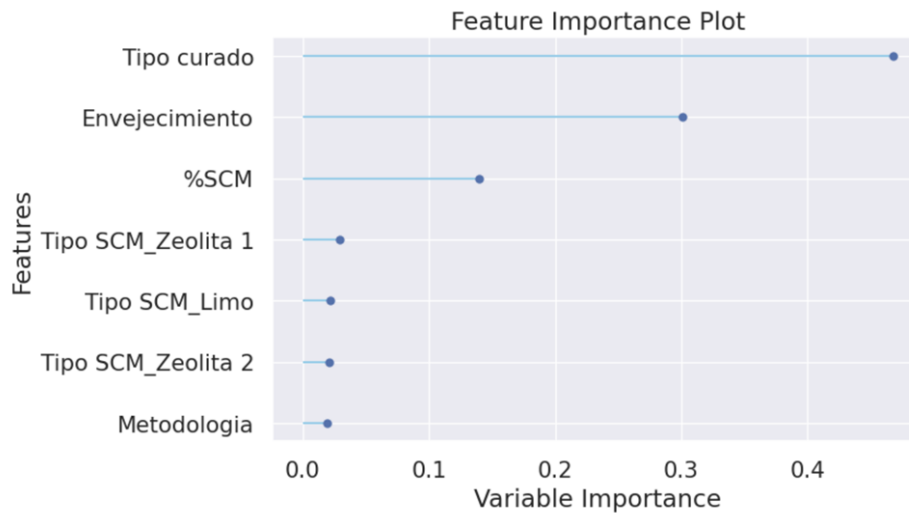
### 4.1 Conclusiones

Después de realizar el estudio se llegó a las siguientes conclusiones:

Para predecir la resistencia a la compresión de pasta de cemento se implementaron 5 variables de entrada tal como se expuso anteriormente, de la cual se pudo concluir que, de todos los modelos utilizados y entrenados, el modelo Extra Tree Regressor generó una mayor confiabilidad puesto que su coeficiente de determinación ( $R^2$ ) fue superior al resto de modelos analizados.

Las métricas estadísticas de los modelos óptimos de ML fueron comparables con los que se encuentran en la literatura.

Según el análisis, se determinó que el tipo de curado ejerce la influencia más significativa en la variable de salida (CS). La precisión de la predicción del modelo Extra Tree Regressor se ve afectada principalmente por las variables de entrada en el siguiente orden: tipo de curado, edad de las muestras y porcentaje de SCM. Los efectos de los siguientes factores (Tipo SCM y Metodología) son inconsistentes, con impactos insignificantes.



**Figura 16:** Importancia de las variables de entrada. Fuente: Elaboración Propia

En resumen, la optimización de hiperparámetros ha contribuido a un mejor rendimiento global del modelo Extra Tree Regressor, demostrando la importancia de ajustar de manera adecuada los parámetros del algoritmo para obtener resultados más precisos y generalizables.

## 4.2 Recomendaciones

La aplicación de modelos de ML en el campo de la construcción ha comenzado a desempeñar un papel cada vez más importante, ya que permite optimizar diseños, detectar defectos y realizar mantenimientos predictivos, que genera una mayor eficiencia, seguridad y calidad de los trabajos.

Por lo cual es necesario enfatizar la implementación de dichos modelos dentro del campo de la construcción pues como se ha mencionado, esto está generando una revolución a la industria de la construcción tradicional.

En cuanto a la predicción de la resistencia a la compresión ofrece beneficios como optimización de diseños y procesos, mejora la calidad y seguridad, identifica y mitiga riesgos asociados.

El éxito del aprendizaje de un modelo de ML dependerá de la calidad de los datos o variables de ingreso y de la selección del modelo adecuado, por lo que se recomienda identificar un modelo que maneje un conjunto de datos complejo y de igual forma que los

datos a ingresar sean de alta calidad, excluyendo valores atípicos y que se disponga de la suficiente cantidad de datos para el entrenamiento, validación y prueba del modelo ML.

Para determinar la resistencia a la compresión aplicando modelos de ML, a medida que se incluyen más variables de entrada, el modelo ML aumenta su capacidad para aprender de manera efectiva, lo que se traduce en respuestas más eficientes y seguras. En general para cualquier proyecto en el cual se pretenda implementar machine learning, es necesario antes de comenzar su modelado asegurarse de comprender el problema propuesto, posterior a ello explorar y comprender los datos que se dispone del caso para luego poder realizar un preprocesamiento de los mismos, en donde el objetivo sea limpiar toda la información del caso, hasta tener un conjunto ideal con características idóneas para que una vez seleccionado el modelo de ML el mismo pueda aprender adecuadamente e ir retroalimentándose en función de la optimización de sus parámetros.

# BIBLIOGRAFÍA

Feng, D.-C., Liu, Z.-T., Wang, X.-D., Chen, Y., Chang, J.-Q., Wei, D.-F., & Jiang, Z.-M. (2020). Machine learning-based compressive strength prediction for concrete: An adaptive boosting approach. *Construction and Building Materials*, 230(117000). <https://doi.org/10.1016/j.conbuildmat.2019.117000>

Chen, B., Wang, L., Feng, Z., Liu, Y., Wu, X., Qin, Y., & Xia, L. (2023). Optimization of high-performance concrete mix ratio design using machine learning. *Engineering Applications of Artificial Intelligence*, 122(106047). <https://doi.org/10.1016/j.engappai.2023.106047>

Stergiou, K., Ntakolia, C., Varytis, P., Koumoulos, E., Karlsson, P., & Moustakidis, S. (2023). Enhancing property prediction and process optimization in building materials through machine learning: A review. *Computational Materials Science*, 220(112031). <https://doi.org/10.1016/j.commatsci.2023.112031>

Sami, B. H. Z., Sami, B. F. Z., Kumar, P., Ahmed, A. N., Amieghemen, G. E., Sherif, M. M., & El-Shafie, A. (2023). Feasibility analysis for predicting the compressive and tensile strength of concrete using machine learning algorithms. *Case Studies in Construction Materials*, 18, e01893. <https://doi.org/10.1016/j.cscm.2023.e01893>

Nguyen, M. H., Nguyen, T., & Ly, H. (s. f.). Ensemble XGBoost schemes for improved compressive strength prediction of UHPC. *Structures*, 57, 105062. <https://doi.org/10.1016/j.istruc.2023.105062>

Vakharia, V., & Gujar, R. (2019). Prediction of compressive strength and portland cement composition using cross-validation and feature ranking techniques. *Construction And Building Materials*, 225, 292-301. <https://doi.org/10.1016/j.conbuildmat.2019.07.224>



Pham, B. T., Nguyen, M. D., Van Dao, D., Prakash, I., Ly, H., Le, T., Ho, L. S., Nguyen, K. T., Trinh, P. T., Hoang, V., Sơn, L. H., Ngo, H. T. T., Tran, H., Minh, N. D., Van Le, H., Loc, H. H., & Bui, D. T. (2019). Development of artificial intelligence models for the prediction of Compression Coefficient of soil: An application of Monte Carlo sensitivity analysis. *Science Of The Total Environment*, 679, 172-184. <https://doi.org/10.1016/j.scitotenv.2019.05.061>

Yaseen, Z. M., Deo, R. C., Hilal, A. A., Abbas, M., Bueno, L. C., Salcedo–Sanz, S., & Nehdi, M. L. (2018). Predicting compressive strength of lightweight foamed concrete using extreme learning machine model. *Advances In Engineering Software*, 115, 112-125. <https://doi.org/10.1016/j.advengsoft.2017.09.00>

Cornejo, M. (2014). *Natural and thermomechanical treated zeolite tuffs used as supplementary cementitious materiales* [Tesis de doctorado]. Ku Leuven.

# **ANEXOS**

## DATOS DE INGRESO EN VARIABLES INDEPENDIENTES

<b>%SCM</b>	<b>Tipo SCM</b>	<b>Tipo curado</b>	<b>Envejecimiento</b>	<b>Metodologia</b>	<b>Density</b>	<b>Compressive strength</b>
5	Zeolita 1	Agua	7	Calibracion	2,119	88,2
5	Zeolita 1	Agua	14	Calibracion	2,098	83,1
5	Zeolita 1	Agua	21	Calibracion	2,129	94,2
5	Zeolita 1	Agua	28	Calibracion	2,164	107,9
5	Zeolita 1	Agua	91	Calibracion	2,175	114,1
7,5	Zeolita 1	Agua	4	Validacion	2,12	61,22
7,5	Zeolita 1	Agua	11	Validacion	2,13	96,76
7,5	Zeolita 1	Agua	18	Validacion	2,13	96,76
7,5	Zeolita 1	Agua	25	Validacion	2,14	101
7,5	Zeolita 1	Agua	91	Validacion	2,15	106,2
10	Zeolita 1	Agua	7	Calibracion	2,132	80,4
10	Zeolita 1	Agua	14	Calibracion	2,136	88,6
10	Zeolita 1	Agua	21	Calibracion	2,142	100,6
10	Zeolita 1	Agua	28	Calibracion	2,127	111,8
10	Zeolita 1	Agua	91	Calibracion	2,148	115,9
12,5	Zeolita 1	Agua	4	Validacion	2,08	76,51
12,5	Zeolita 1	Agua	11	Validacion	2,14	89,41
12,5	Zeolita 1	Agua	18	Validacion	2,12	103,1
12,5	Zeolita 1	Agua	25	Validacion	2,11	104,8
12,5	Zeolita 1	Agua	91	Validacion	2,12	122,7
15	Zeolita 1	Agua	7	Calibracion	2,102	72,2
15	Zeolita 1	Agua	14	Calibracion	2,132	81,2
15	Zeolita 1	Agua	21	Calibracion	2,123	112
15	Zeolita 1	Agua	28	Calibracion	2,107	115,4
15	Zeolita 1	Agua	91	Calibracion	2,189	126
17,5	Zeolita 1	Agua	4	Validacion	2,07	56,56
17,5	Zeolita 1	Agua	11	Validacion	2,11	81,23
17,5	Zeolita 1	Agua	18	Validacion	2,14	64,13
17,5	Zeolita 1	Agua	25	Validacion	2,1	99,19
17,5	Zeolita 1	Agua	91	Validacion	2,11	125,2
20	Zeolita 1	Agua	7	Calibracion	2,07	57,5
20	Zeolita 1	Agua	14	Calibracion	2,14	61,6
20	Zeolita 1	Agua	21	Calibracion	2,166	89,8
20	Zeolita 1	Agua	28	Calibracion	2,13	111,2
20	Zeolita 1	Agua	91	Calibracion	2,155	121,1
22,5	Zeolita 1	Agua	4	Validacion	2,11	58,44
22,5	Zeolita 1	Agua	11	Validacion	2,08	82
22,5	Zeolita 1	Agua	18	Validacion	2,13	94,06
22,5	Zeolita 1	Agua	25	Validacion	2,14	111,1
22,5	Zeolita 1	Agua	91	Validacion	2,06	114,5
25	Zeolita 1	Agua	7	Calibracion	2,06	53
25	Zeolita 1	Agua	14	Calibracion	2,046	94,6
25	Zeolita 1	Agua	21	Calibracion	2,058	54,2

25	Zeolita 1	Agua	28	Calibracion	2,056	98
25	Zeolita 1	Agua	91	Calibracion	2,113	106
27,5	Zeolita 1	Agua	4	Validacion	2,05	61,85
27,5	Zeolita 1	Agua	11	Validacion	2,15	69,6
27,5	Zeolita 1	Agua	18	Validacion	2,09	79,95
27,5	Zeolita 1	Agua	25	Validacion	2,09	96,28
27,5	Zeolita 1	Agua	91	Validacion	2	96,06
5	Zeolita 2	Agua	7	Calibracion	2,114	80,6
5	Zeolita 2	Agua	14	Calibracion	2,136	114,6
5	Zeolita 2	Agua	21	Calibracion	2,163	108,4
5	Zeolita 2	Agua	28	Calibracion	2,11	101,9
5	Zeolita 2	Agua	91	Calibracion	2,161	100,9
7,5	Zeolita 2	Agua	4	Validacion	2,16	61,74
7,5	Zeolita 2	Agua	11	Validacion	2,1	96,22
7,5	Zeolita 2	Agua	18	Validacion	2,13	103,3
7,5	Zeolita 2	Agua	25	Validacion	2,13	97,11
7,5	Zeolita 2	Agua	91	Validacion	2,12	98,7
10	Zeolita 2	Agua	7	Calibracion	2,105	79,9
10	Zeolita 2	Agua	14	Calibracion	2,094	94
10	Zeolita 2	Agua	21	Calibracion	2,094	102,3
10	Zeolita 2	Agua	28	Calibracion	2,103	84,8
10	Zeolita 2	Agua	91	Calibracion	2,11	87
12,5	Zeolita 2	Agua	4	Validacion	2,06	29,55
12,5	Zeolita 2	Agua	11	Validacion	2,06	88,99
12,5	Zeolita 2	Agua	18	Validacion	2,1	88,1
12,5	Zeolita 2	Agua	25	Validacion	2,11	93,44
12,5	Zeolita 2	Agua	91	Validacion	2,12	110,2
15	Zeolita 2	Agua	7	Calibracion	2,076	81,4
15	Zeolita 2	Agua	14	Calibracion	2,066	82,6
15	Zeolita 2	Agua	21	Calibracion	2,09	97,4
15	Zeolita 2	Agua	28	Calibracion	2,086	81,7
15	Zeolita 2	Agua	91	Calibracion	2,089	119,5
17,5	Zeolita 2	Agua	4	Validacion	2,06	53,51
17,5	Zeolita 2	Agua	11	Validacion	2,02	82,86
17,5	Zeolita 2	Agua	18	Validacion	2,07	99,78
17,5	Zeolita 2	Agua	25	Validacion	2,06	85,53
17,5	Zeolita 2	Agua	91	Validacion	2,08	117,6
20	Zeolita 2	Agua	7	Calibracion	2,04	83,8
20	Zeolita 2	Agua	14	Calibracion	2,042	93,1
20	Zeolita 2	Agua	21	Calibracion	2,044	98,5
20	Zeolita 2	Agua	28	Calibracion	2,069	102,3
20	Zeolita 2	Agua	91	Calibracion	2,068	116,4
22,5	Zeolita 2	Agua	4	Validacion	2,02	63,94
22,5	Zeolita 2	Agua	11	Validacion	2,02	80,06
22,5	Zeolita 2	Agua	18	Validacion	2,04	78,28
22,5	Zeolita 2	Agua	25	Validacion	2,03	84,51
22,5	Zeolita 2	Agua	91	Validacion	2,07	115,6

25	Zeolita 2	Agua	7	Calibracion	2,013	78,2
25	Zeolita 2	Agua	14	Calibracion	1,995	78,5
25	Zeolita 2	Agua	21	Calibracion	1,993	78,5
25	Zeolita 2	Agua	28	Calibracion	2,017	80,6
25	Zeolita 2	Agua	91	Calibracion	2,045	110,1
27,5	Zeolita 2	Agua	4	Validacion	2,01	49,78
27,5	Zeolita 2	Agua	11	Validacion	2,01	78,24
27,5	Zeolita 2	Agua	18	Validacion	2,03	72,9
27,5	Zeolita 2	Agua	25	Validacion	1,98	95,13
27,5	Zeolita 2	Agua	91	Validacion	2,02	116,5
5	Limo	Agua	7	Calibracion	2,141	71,7
5	Limo	Agua	14	Calibracion	2,122	68,6
5	Limo	Agua	21	Calibracion	2,151	78,3
5	Limo	Agua	28	Calibracion	2,122	117,5
5	Limo	Agua	91	Calibracion	2,175	119,4
7,5	Limo	Agua	4	Validacion	2,03	43,12
7,5	Limo	Agua	11	Validacion	2,04	59,78
7,5	Limo	Agua	18	Validacion	2,05	82,77
7,5	Limo	Agua	25	Validacion	2,07	89,89
7,5	Limo	Agua	91	Validacion	2,13	79,54
10	Limo	Agua	7	Calibracion	2,161	85,4
10	Limo	Agua	14	Calibracion	2,126	88,1
10	Limo	Agua	21	Calibracion	2,133	102,4
10	Limo	Agua	28	Calibracion	2,124	85,4
10	Limo	Agua	91	Calibracion	2,155	112,8
12,5	Limo	Agua	4	Validacion	2,06	49,98
12,5	Limo	Agua	11	Validacion	2,12	92,56
12,5	Limo	Agua	18	Validacion	2,11	99,68
12,5	Limo	Agua	25	Validacion	2,11	90,78
12,5	Limo	Agua	91	Validacion	2,18	99,5
15	Limo	Agua	7	Calibracion	2,072	58,7
15	Limo	Agua	14	Calibracion	2,126	90,8
15	Limo	Agua	21	Calibracion	2,142	89
15	Limo	Agua	28	Calibracion	2,119	91,7
15	Limo	Agua	91	Calibracion	2,11	106,5
17,5	Limo	Agua	4	Validacion	2,13	44,1
17,5	Limo	Agua	11	Validacion	2,12	81,88
17,5	Limo	Agua	18	Validacion	2,12	83,66
17,5	Limo	Agua	25	Validacion	2,11	89,36
17,5	Limo	Agua	91	Validacion	2,13	115,29
20	Limo	Agua	7	Calibracion	2,117	62,7
20	Limo	Agua	14	Calibracion	2,118	82,9
20	Limo	Agua	21	Calibracion	2,102	83,7
20	Limo	Agua	28	Calibracion	2,121	96,7
20	Limo	Agua	91	Calibracion	2,1	93,2
22,5	Limo	Agua	4	Validacion	2,06	44,62
22,5	Limo	Agua	11	Validacion	2,04	56,96

22,5	Limo	Agua	18	Validacion	2,08	65,59
22,5	Limo	Agua	25	Validacion	2,11	68,6
22,5	Limo	Agua	91	Validacion	2,11	101,48
25	Limo	Agua	7	Calibracion	2,058	24,1
25	Limo	Agua	14	Calibracion	2,076	74,1
25	Limo	Agua	21	Calibracion	2,046	79,3
25	Limo	Agua	28	Calibracion	2,064	86,2
25	Limo	Agua	91	Calibracion	2,112	94,4
27,5	Limo	Agua	4	Validacion	2,08	56,05
27,5	Limo	Agua	11	Validacion	2,14	64,64
27,5	Limo	Agua	18	Validacion	2,14	77,57
27,5	Limo	Agua	25	Validacion	2,09	75,84
27,5	Limo	Agua	91	Validacion	2,14	92,49
5	Zeolita 1	Aire	7	Calibracion	2,14	65,9
5	Zeolita 1	Aire	14	Calibracion	2,15	58,2
5	Zeolita 1	Aire	21	Calibracion	2,18	74,5
5	Zeolita 1	Aire	28	Calibracion	2,13	80,5
5	Zeolita 1	Aire	91	Calibracion	2,17	48,76
7,5	Zeolita 1	Aire	4	Validacion	2,16	22,9
7,5	Zeolita 1	Aire	11	Validacion	2,16	62,7
7,5	Zeolita 1	Aire	18	Validacion	2,26	68,6
7,5	Zeolita 1	Aire	25	Validacion	2,18	75
7,5	Zeolita 1	Aire	91	Validacion	2,23	49,61
10	Zeolita 1	Aire	7	Calibracion	2,08	65,7
10	Zeolita 1	Aire	14	Calibracion	2,06	64,05
10	Zeolita 1	Aire	21	Calibracion	2,15	66,72
10	Zeolita 1	Aire	28	Calibracion	2,08	62,27
10	Zeolita 1	Aire	91	Calibracion	2,26	42,43
12,5	Zeolita 1	Aire	4	Validacion	2,13	39,22
12,5	Zeolita 1	Aire	11	Validacion	2,15	62,27
12,5	Zeolita 1	Aire	18	Validacion	2,2	75,61
12,5	Zeolita 1	Aire	25	Validacion	2,12	80,06
12,5	Zeolita 1	Aire	91	Validacion	2,27	49,97
15	Zeolita 1	Aire	7	Calibracion	2,12	58,84
15	Zeolita 1	Aire	14	Calibracion	2,04	64,94
15	Zeolita 1	Aire	21	Calibracion	2,08	66,72
15	Zeolita 1	Aire	28	Calibracion	2,06	71,17
15	Zeolita 1	Aire	91	Calibracion	2,12	41,22
17,5	Zeolita 1	Aire	4	Validacion	2,12	56,87
17,5	Zeolita 1	Aire	11	Validacion	2,12	59,22
17,5	Zeolita 1	Aire	18	Validacion	2,16	62,27
17,5	Zeolita 1	Aire	25	Validacion	2,15	71,16
17,5	Zeolita 1	Aire	91	Validacion	2,34	49,17
20	Zeolita 1	Aire	7	Calibracion	2,15	54,63
20	Zeolita 1	Aire	14	Calibracion	2,09	58,4
20	Zeolita 1	Aire	21	Calibracion	2,07	68,08
20	Zeolita 1	Aire	28	Calibracion	2,08	87,34

20	Zeolita 1	Aire	91	Calibracion	2,09	49,78
22,5	Zeolita 1	Aire	4	Validacion	2,07	52,37
22,5	Zeolita 1	Aire	11	Validacion	2,13	62,23
22,5	Zeolita 1	Aire	18	Validacion	2,13	70,93
22,5	Zeolita 1	Aire	25	Validacion	2,14	76,15
22,5	Zeolita 1	Aire	91	Validacion	2,2	47,51
25	Zeolita 1	Aire	7	Calibracion	2,07	60,71
25	Zeolita 1	Aire	14	Calibracion	2,09	67,55
25	Zeolita 1	Aire	21	Calibracion	2,06	54,72
25	Zeolita 1	Aire	28	Calibracion	2,2	51,81
25	Zeolita 1	Aire	91	Calibracion	2,13	47,8
27,5	Zeolita 1	Aire	4	Validacion	2,09	35,06
27,5	Zeolita 1	Aire	11	Validacion	2,07	61,56
27,5	Zeolita 1	Aire	18	Validacion	2,07	57,29
27,5	Zeolita 1	Aire	25	Validacion	2,15	49,59
27,5	Zeolita 1	Aire	91	Validacion	2,24	51,18
5	Zeolita 2	Aire	7	Calibracion	2,12	57,82
5	Zeolita 2	Aire	14	Calibracion	2,18	66,68
5	Zeolita 2	Aire	21	Calibracion	2,15	66,68
5	Zeolita 2	Aire	28	Calibracion	2,2	68,46
5	Zeolita 2	Aire	91	Calibracion	2,26	73,62
7,5	Zeolita 2	Aire	4	Validacion	2,15	49,98
7,5	Zeolita 2	Aire	11	Validacion	2,25	49
7,5	Zeolita 2	Aire	18	Validacion	2,17	49,98
7,5	Zeolita 2	Aire	25	Validacion	2,12	50,96
7,5	Zeolita 2	Aire	91	Validacion	2,19	63,4
10	Zeolita 2	Aire	7	Calibracion	2,36	57,82
10	Zeolita 2	Aire	14	Calibracion	2,23	57,82
10	Zeolita 2	Aire	21	Calibracion	2,2	44,1
10	Zeolita 2	Aire	28	Calibracion	2,17	65,66
10	Zeolita 2	Aire	91	Calibracion	2,13	56,87
12,5	Zeolita 2	Aire	4	Validacion	2,11	56,84
12,5	Zeolita 2	Aire	11	Validacion	2,01	58,8
12,5	Zeolita 2	Aire	18	Validacion	2,11	52,92
12,5	Zeolita 2	Aire	25	Validacion	2,1	41,75
12,5	Zeolita 2	Aire	91	Validacion	2,24	54,89
15	Zeolita 2	Aire	7	Calibracion	2,23	52,92
15	Zeolita 2	Aire	14	Calibracion	2,14	55,86
15	Zeolita 2	Aire	21	Calibracion	2,25	51,94
15	Zeolita 2	Aire	28	Calibracion	0,12	54,88
15	Zeolita 2	Aire	91	Calibracion	2,33	62,12
17,5	Zeolita 2	Aire	4	Validacion	2,07	49,98
17,5	Zeolita 2	Aire	11	Validacion	2,07	49
17,5	Zeolita 2	Aire	18	Validacion	2,08	49,98
17,5	Zeolita 2	Aire	25	Validacion	2,05	50,96
17,5	Zeolita 2	Aire	91	Validacion	2,14	55,69
20	Zeolita 2	Aire	7	Calibracion	2,21	47,04

20	Zeolita 2	Aire	14	Calibracion	2,04	49
20	Zeolita 2	Aire	21	Calibracion	2,18	51,94
20	Zeolita 2	Aire	28	Calibracion	2,14	67,57
20	Zeolita 2	Aire	91	Calibracion	2,16	69,43
22,5	Zeolita 2	Aire	4	Validacion	2,04	57,82
22,5	Zeolita 2	Aire	11	Validacion	2,04	60,76
22,5	Zeolita 2	Aire	18	Validacion	2,13	86,24
22,5	Zeolita 2	Aire	25	Validacion	2,05	74,68
22,5	Zeolita 2	Aire	91	Validacion	2,18	55,49
25	Zeolita 2	Aire	7	Calibracion	2,03	45,67
25	Zeolita 2	Aire	14	Calibracion	2,16	45,08
25	Zeolita 2	Aire	21	Calibracion	2,18	43,12
25	Zeolita 2	Aire	28	Calibracion	2,1	50,96
25	Zeolita 2	Aire	91	Calibracion	2,18	53,01
27,5	Zeolita 2	Aire	4	Validacion	2,04	32,93
27,5	Zeolita 2	Aire	11	Validacion	2,11	41,16
27,5	Zeolita 2	Aire	18	Validacion	2,09	43,12
27,5	Zeolita 2	Aire	25	Validacion	2,2	37,24
27,5	Zeolita 2	Aire	91	Validacion	2,17	50,27
5	Limo	Aire	7	Calibracion	2,33	35,55
5	Limo	Aire	14	Calibracion	2,21	73,83
5	Limo	Aire	21	Calibracion	2,3	68,49
5	Limo	Aire	28	Calibracion	2,31	77,39
5	Limo	Aire	91	Calibracion	2,28	45,5
7,5	Limo	Aire	4	Validacion	2,17	59,84
7,5	Limo	Aire	11	Validacion	2,16	66,71
7,5	Limo	Aire	18	Validacion	2,24	79,17
7,5	Limo	Aire	25	Validacion	2,21	78,28
7,5	Limo	Aire	91	Validacion	2,22	62,72
10	Limo	Aire	7	Calibracion	2,33	46,06
10	Limo	Aire	14	Calibracion	2,21	52,92
10	Limo	Aire	21	Calibracion	2,13	85,23
10	Limo	Aire	28	Calibracion	2,18	71,26
10	Limo	Aire	91	Calibracion	2,23	62,77
12,5	Limo	Aire	4	Validacion	2,13	64,68
12,5	Limo	Aire	11	Validacion	2,14	52,92
12,5	Limo	Aire	18	Validacion	2,12	84,64
12,5	Limo	Aire	25	Validacion	2,17	77,51
12,5	Limo	Aire	91	Validacion	2,28	65,78
15	Limo	Aire	7	Calibracion	2,28	52,19
15	Limo	Aire	14	Calibracion	2,25	49,44
15	Limo	Aire	21	Calibracion	2,18	51,01
15	Limo	Aire	28	Calibracion	2,24	51,01
15	Limo	Aire	91	Calibracion	2,33	59,96
17,5	Limo	Aire	4	Validacion	2,09	44,15
17,5	Limo	Aire	11	Validacion	2,1	52,39
17,5	Limo	Aire	18	Validacion	2,13	53,96



17,5	Limo	Aire	25	Validacion	2,16	48,07
17,5	Limo	Aire	91	Validacion	2,21	65,66
20	Limo	Aire	7	Calibracion	2,3	40,57
20	Limo	Aire	14	Calibracion	2,3	47,04
20	Limo	Aire	21	Calibracion	2,25	68,45
20	Limo	Aire	28	Calibracion	2,29	60,45
20	Limo	Aire	91	Calibracion	2,3	64,82
22,5	Limo	Aire	4	Validacion	2,1	44,3
22,5	Limo	Aire	11	Validacion	2,07	52,92
22,5	Limo	Aire	18	Validacion	2,12	70,23
22,5	Limo	Aire	25	Validacion	2,16	59,56
22,5	Limo	Aire	91	Validacion	2,17	64,44
25	Limo	Aire	7	Calibracion	2,09	49
25	Limo	Aire	14	Calibracion	2,09	50,96
25	Limo	Aire	21	Calibracion	2,1	71,12
25	Limo	Aire	28	Calibracion	2,16	47,04
25	Limo	Aire	91	Calibracion	2,28	45,89
27,5	Limo	Aire	4	Validacion	2,06	51,16
27,5	Limo	Aire	11	Validacion	2,06	50,96
27,5	Limo	Aire	18	Validacion	2,09	53,69
27,5	Limo	Aire	25	Validacion	2,09	53,34
27,5	Limo	Aire	91	Validacion	2,17	50,99

## CÓDIGO PYCARET

```

!pip install matplotlib #to install matplotlib
!pip install mpl_toolkits #to install mpl_toolkits
!pip install psutil #to install psutil
!pip install scikit-learn #to install scikit-learn
!pip install --pre pycaret # to install pycaret
#to import all Pycaret regression functions
from pycaret.regression import *
import pandas as pd #to import pandas as pd
import numpy as np #to import numpy as np
from sklearn.metrics import mean_squared_error #to import MSE
from sklearn.metrics import mean_absolute_error #to import MAE
from sklearn.metrics import r2_score #to import coefficient of determination
import matplotlib.pyplot as plt #to import matplotlib as plt
import seaborn as sns #to import seaborn as sns
import math #to import math

```

```

from google.colab import files #to read documents on colab

#to access the data file from google drive
from google.colab import drive
drive.mount('/content/drive')

#to open the data file from google drive
#df = pd.read_excel(r'/content/drive/My Drive/Colab Notebooks/Dr Mauricio _ Base de
datos/Tabla tesis.xlsx') #to save the data file as df
df = pd.read_excel(r'/content/drive/My Drive/Tabla tesis sin cod.xlsx') #to save the data
file as df
df.head() #returns the first 5 rows of the data

round(df.describe(),2) # describe() returns the description of the data in the DataFrame
(mean, standard deviation, and percentiles)
# A data frame is a data structure that organizes data into a 2-dimensional table of rows
and columns
#round(): returns a value with a specific number of decimals

round(df.corr( ),2) #to calculate the correlation coefficient of the variables and round the
values to two decimal places

df_corr = round(df.corr( ),2)#to save the correlation coefficients as df_corr
plt.figure(figsize=(15,10)) #plt.figure() to graph the table
#figsize: width, height in inches of the figure
sns.set(font_scale=1.4) #font size
heat_map = sns.heatmap(df_corr,annot=True,cmap=plt.cm.Blues) #to create the
heatmap
#annot=True: to write the data value in each cell'
#cmap=plt.cm.Blues: the color of the heatmap
#to save the image
#plt.savefig("heat_map.png", dpi=600, format='png', bbox_inches='tight')
#files.download("heat_map.png")

```

```

df_cs = df.drop(columns=['Density'])
print(df_cs)

#setup(): to initialize the environment
#df is the dataset
#target is the compressive strength
#normalize = True: to normalize the data
#normalize_method = 'zscore': the method to normalize the data
#numeric_features: the input variables
#fold=5: for k-fold cross validation
#transformation = True: to transform the original data to a normal distribution
#reg1 = setup(df, target = 'Compressive strength', session_id=123, normalize = True,
normalize_method = 'zscore', fold=5, transformation = True, numeric_features = ['Mass
of chitosan','Mass of metakaolin','Mass of quartz sand','Mass of sodium Silicate','Mass of
water','Mass of NaOH', 'Curing time (days)'], use_gpu = False)
reg1 = setup(df_cs, target = 'Compressive strength', session_id=123, normalize = True,
normalize_method = 'zscore', fold=5, numeric_features = ['%SCM', 'Envejecimiento'],
categorical_features=['Tipo SCM', 'Tipo curado', 'Metodologia'], use_gpu = False)
#reg1 = setup(df_cs, target = 'Compressive strength', session_id=123, normalize = True,
normalize_method = 'zscore', fold=5, numeric_features = ['%SCM', 'Envejecimiento','Tipo
SCM', 'Tipo curado', 'Metodologia'], use_gpu = False)

best_model_cs=compare_models(fold=5)
gbr_train_cs=create_model('gbr',fold=5)
gbr_train_error_cs=plot_model(gbr_train_cs,plot='error')
dt_train_cs = create_model('dt', fold=5) #to create the model
dt_train_feature_cs = plot_model(dt_train_cs, plot='feature') #to train the Model
dt_train_error_cs = plot_model(dt_train_cs, plot='error') # prediction error
dt_train_test_cs = predict_model(dt_train_cs) #to test the Model using new unseen data
dt_tune_cs = tune_model(dt_train_cs, n_iter=100, choose_better=True, optimize='R2')

```