Evaluación comparativa de modelos de aprendizaje automático para la optimización del factor de secado en el proceso industrial de descafeinado del café

Diego Andrés Orrego Grisales

Angélica María Ruíz Rodríguez

Asesor

Julio Eduardo Mejia Manzano

Universidad Nacional Abierta y a Distancia UNAD

Escuela de Ciencias Básicas, Tecnología e Ingeniería ECBTI

Especialización en Ciencia de Datos y Analítica

2026

Resumen

Este trabajo de grado contempla una problemática en la empresa descafeinadora en una de las etapas críticas del proceso productivo, que afecta directamente la rentabilidad, calidad y productividad de la empresa. Esta etapa crítica se denomina secado, en ella se busca alcanzar el rango de humedad de café establecido por el cliente, como parte del postratamiento en la línea, este secado se realiza de manera indirecta con un control de presión de vacío y temperatura. Para obtener el valor deseado de humedad los supervisores y jefes de producción asignan un factor de secado que es afectado por diferentes parámetros y condiciones como el tipo de café, tipo de secador, humedad relativa, entre otros, algunos van relacionados con la materia prima y otros con la tecnología usada, sin embargo, solo se asigna un factor para todas las variables, con el fin de facilitar su escogencia. Este factor históricamente ha sido asignado de manera empírica basado en la experiencia de los trabajadores, lo que ha generado una alta variabilidad de los resultados, que en ocasiones afecta la calidad (propiedades organolépticas, vida útil del producto) y rentabilidad de los procesos.

Para estandarizar el proceso productivo, en este trabajo se desarrolla un modelo de aprendizaje automático para determinar de manera óptima este factor de secado, para esto estudio se emplea la metodología CRISP-DM (Cross-Industry Standard Process for Data Mining), por lo que se solicita a la empresa los datos históricos de los clientes, parámetros de secado, secadores y resultados obtenidos para cada una de las operaciones realizadas, se caracteriza el comportamiento de los datos y se detecta si existen sesgos que puedan afectar el modelo predictivo. Posteriormente, se realiza la limpieza y organización de los datos para usarlos en los modelos de machine learning. Dentro de los pasos a seguir se encuentran: eliminar datos

duplicados o inconsistentes, imputar valores faltantes, estandarizar variables numéricas y codificar las variables categóricas.

Se analiza el ajuste de los datos con los modelos: Regresión lineal múltiple (MAE: 16,812 y R2: 0,389), Árboles de decisión (MAE: 14,119 y R^2: 0,576), Random Forest con hiperparámetros optimizados (MAE: 12,539 y R^2: 0,667), Gradient Boosting con hiperparámetros optimizados (MAE: 12,364 y R^2: 0,681), Support Vector Regression (SVR) (MAE: 16,725 y R^2: 0,408) y Redes neuronales artificiales (MAE: 17,103 y R^2: 0,372), y se seleccionó el modelo Gradient Boosting que tiene el mejor ajuste y el menor error. Posteriormente, se realiza la integración del modelo predictivo con el proceso dentro de la empresa descafeinadora, en donde se visualiza el factor de secado recomendado para cada lote. Luego de tomar 100 registros para la implementación del modelo se evalúa el mejoramiento de los resultados de las humedades del café en el proceso, y se dan las recomendaciones respectivas a la empresa, para lograr el mantenimiento de la mejora en el proceso, las posibles variables extras que se pueden estudiar para aumentar la precisión en la toma de decisión respecto al secado, y realizar los ajustes respectivos a los modelos con el paso del tiempo.

Palabras clave: Café descafeinado, Secado de café, Factor de secado, Aprendizaje automático, Modelo predictivo.

Abstract

This degree project addresses a problem in the decaffeination company during one of the critical stages of the production process, which directly affects the company's profitability, quality, and productivity. This critical stage is called drying, in which the objective is to achieve the coffee moisture range established by the client as part of the post-treatment process in the production line. This drying process is carried out indirectly through vacuum pressure and temperature control. To obtain the desired moisture value, supervisors and production managers assign a drying factor that is affected by different parameters and conditions such as the type of coffee, type of dryer, relative humidity, among others; some are related to the raw material and others to the technology used. However, only one factor is assigned for all variables in order to facilitate its selection. Historically, this factor has been assigned empirically based on the workers' experience, which has generated high variability in the results, occasionally affecting quality (organoleptic properties, product shelf life) and process profitability.

To standardize the production process, this study develops a machine learning model to optimally determine this drying factor. For this study, the CRISP-DM (Cross-Industry Standard Process for Data Mining) methodology is employed. Therefore, the company's historical data regarding clients, drying parameters, dryers, and results obtained for each operation performed are requested. The behavior of the data is characterized, and it is determined whether biases exist that may affect the predictive model. Subsequently, data cleaning and organization are carried out in order to use the data in machine learning models. The steps to be followed include: removing duplicate or inconsistent data, imputing missing values, standardizing numerical variables, and encoding categorical variables.

The fit of the data is analyzed using the following models: Multiple linear regression (MAE: 16.812 and R2: 0.389), Decision trees (MAE: 14.119 and R2: 0.576), Random Forest with optimized hyperparameters (MAE: 12.539 and R2: 0.667), Gradient Boosting with optimized hyperparameters (MAE: 12.364 and R2: 0.681), Support Vector Regression (SVR) (MAE: 16.725 and R2: 0.408), and Artificial neural networks (MAE: 17.103 and R2: 0.372). The Gradient Boosting model was selected because it achieved the best fit and the lowest error. Subsequently, the predictive model is integrated into the process within the decaffeination company, where the recommended drying factor for each batch is displayed. After taking 100 records for the implementation of the model, the improvement in coffee moisture results within the process is evaluated, and the respective recommendations are provided to the company in order to maintain process improvement, identify possible additional variables that can be studied to increase precision in decision-making regarding drying, and make the respective adjustments to the models over time.

Keywords: Decaffeinated coffee, Coffee drying, Drying factor, Machine learning, Predictive model.

Tabla de Contenido

Lista de Tablas

Lista de Figuras

Lista de Apéndices

Introducción

El proceso de secado de café es una de las etapas más críticas del descafeinado, debido a que se debe garantizar la conservación de la calidad fisicoquímica y sensorial del café. En la empresa descafeinadora, la determinación del factor de secado se ha realizado de manera empírica basados en la experiencia y el criterio de los supervisores, presentando una alta variabilidad en los resultados y posibilidad de desviación respecto a la humedad final solicitada por el cliente a la hora de programar el secado de café. En la literatura se evidencia que el proceso de secado es una operación clave en la conservación de las propiedades fisicoquímicas y organolépticas del café, mostrando que un correcto secado puede evitar impactos negativos en el valor comercial del café (Hurtado Cortés et al., 2024).

A nivel investigativo el control de la humedad del café es un desafío ampliamente documentado. En investigaciones realizadas se ha demostrado la influencia de factores como la temperatura del aire, el flujo de este y el tiempo de secado, pueden modificar la calidad del grano y causar un deterioro del mismo (Peñuela-Martínez et al., 2023).

También se ha investigado sobre el efecto de las temperaturas excesivas o cuando existen controles inestables en la degradación de propiedades organolépticas como sabor y aroma (Abreu et al., 2025). En otros estudios se ha demostrado como la extracción de cafeína genera pérdidas adicionales en los aromas, haciendo que sea más susceptible durante el proceso del secado (Chindapan et al., 2025; Zou et al., 2022).

En cuanto a investigación con el uso de herramientas de aprendizaje automático existen varios estudios que demuestran su utilidad para procesos de secado. Se demostró que algoritmos como Random Forest (RF), Support Vector Regression (SVR) y redes neuronales, son capaces de modelar las isotermas de sorción del café con una precisión mayor al 99% (Collazos-Escobar,

Bahamón-Monje, et al., 2025; Collazos-Escobar, Gutiérrez-Guzmán, et al., 2025). En otras investigaciones emplearon ANFIS y redes neuronales para predecir el contenido humedad, ayudando a reducir errores y en la optimización del consumo energético (Le et al., 2025).

El problema actual en la descafeinadora se presenta por la alta variabilidad en la asignación de factores de secado empleados por los supervisores y jefes de turno para alcanzar los valores de humedad solicitados por el cliente a la hora de programar el secado de café. Esta variabilidad surge porque se deben considerar diversos factores como el histórico de cliente, histórico de secador, humedad final del café, entre otros factores. Como consecuencia se han presentado lotes de café fuera de las especificaciones de cliente y de los estándares de calidad.

Justificación

Determinar exactamente el factor de secado dentro de un proceso de descafeinado representa un desafío industrial debido a que hay diversos factores que pueden afectar su elección como la variabilidad del grano, las diferencias entre las solicitudes de los clientes y las diferencias entre secadores. La industria actual está enfocada en la estandarización de los procesos y la calidad del producto para poder ser competitivo, es allí donde se genera la necesidad de realizar un análisis de datos que permitan reducir la incertidumbre en la elección del factor de secado generado por su selección de forma empírica. Diferentes estudios comprueban que la forma en la que se lleva a cabo el secado influye en la preservación de las propiedades organolépticas como aroma y sabor (Abreu et al., 2025; Peñuela-Martínez et al., 2023). Es por esto, que el problema adquiere relevancia en especial para la empresa descafeinadora, donde una mala elección del factor de secado puede conllevar pérdidas económicas y deterioro de la calidad.

En cuanto al ámbito académico e investigativo, este estudio aparta en el campo de la ingeniería de alimentos – ingeniería química y las tecnologías aplicadas al procesamiento y secado del café. En estudios recientes se muestra el uso de modelos de aprendizaje automático con herramientas como Random Forest, Support Vector Regression y redes neuronales que estiman con precisión parámetros importantes dentro del proceso de secado (Collazos-Escobar, Gutiérrez-Guzmán, et al., 2025; Le et al., 2025). Implementar estas herramientas no es útil únicamente para el sector cafetero y para el proceso de secado, sino que puede servir como referente para otras aplicaciones especialmente en la agroindustria.

Este estudio también es útil directamente para los involucrados en los procesos de secado de café especialmente cuando el café se somete a descafeinado. De acuerdo con Hurtado Cortés

et al. (2024) y Zou et al. (2022) diseñar un modelo predictivo para determinar las variables óptimas de secado puede evitar problemas como obtener valores de humedad por fuera de los parámetros establecidos y disminuir la variabilidad entre lotes. También otros estudios demuestran que cuando el café es sometido a procesos de descafeinado se pueden perder compuestos aromáticos y posteriormente ser más sensible a la temperatura (Chindapan et al., 2025; Pietsch, 2017).

El secado del café ha sido ampliamente investigado en la literatura junto con parámetros para el control del secado y cinéticas de sorción, sin embargo, no se encontró un referente bibliográfico que diseñe un modelo específicamente para predecir el factor de secado en café descafeinado, que integre factores como parámetros definidos por el cliente, el tipo de secador, pérdidas por enfriamiento, entre otros parámetros relevantes, encontrando un vacío en la literatura. La mayoría de los referentes bibliográficos encontrados están basados en determinar las curvas de secado e isotermas, pero no en la determinación de factores industriales que se apliquen directamente al control de la operación (Abreu et al., 2025; Peñuela-Martínez et al., 2023).

Por ellos surge la pregunta de investigación: ¿Cómo puede el uso de modelos de aprendizaje automático contribuir a determinar de manera precisa el factor de secado en el proceso de descafeinado del café, reduciendo los errores asociados a su determinación empírica por parte de los supervisores de producción?

Objetivos

Objetivo General

Implementar modelos de aprendizaje automático para estimar el factor de secado óptimo en el proceso de descafeinado del café, con el fin de reducir la variabilidad operativa y mejorar la productividad.

Objetivos Específicos

Identificar las variables físicas, químicas y operativas que influyen en la selección del factor de secado.

Implementar diferentes modelos de aprendizaje automático para la estimación del factor de secado óptimo.

Evaluar el impacto del modelo con mejor desempeño en la eficiencia y estabilidad del proceso productivo.

Marco Teórico

El secado del café es una de las etapas más críticas especialmente dentro del proceso de descafeinado, debido a que la estructura del grano es modificada por tratamientos previos como la vaporización, la humectación y la extracción de la cafeína, generando mayor sensibilidad a la temperatura y variabilidad en la humedad del producto final (Clarke & Macrae, 1987; Zou et al., 2022). En la literatura Hurtado Cortés et al. (2024) y Peñuela-Martínez et al. (2023) también muestran como el secado influye en las características organolépticas del café y en su precio en el mercado.

Por medio de los avances tecnológicos y en aprendizaje supervisado es posible aproximarse a los procesos de secado prediciendo de forma más precisa las variables o resultados dentro de un proceso y de esta forma ser menos dependiente solo de la experiencia humana. En estudios realizados se ha demostrado la efectividad de los algoritmos de machine learning para modelar variables como humedad y el comportamiento de las variables dentro del proceso de secado (Collazos-Escobar, Gutiérrez-Guzmán, et al., 2025). Es por esto, que dentro de este marco teórico se analizarán los conceptos del descafeinado y secado del café, y la generación de modelos que se ajusten a los procesos productivos.

Proceso de Descafeinado del Café

Durante el proceso de descafeinado del café, este es sometido a condiciones de presión, temperatura y adición del solvente para remover la cafeína, inicialmente se realiza una vaporización con vapor a temperatura controlada donde se ablanda la pared del grano y se expanden los poros del grano; en esta etapa la humedad puede aumentar aproximadamente un 6% (Clarke & Macrae, 1987). Se continúa con una humectación donde se facilita la solubilización de la cafeína aumentando la permeabilidad de la membrana celular; la expansión

del grano aumenta la superficie de contacto que el solvente tendrá con la cafeína, pero también produce un deterioro que hace al grano más sensible a la temperatura (Chindapan et al., 2025). Posteriormente se añade el solvente al café continuando con la etapa de extracción en la cual el solvente retiene la cafeína (Clarke & Macrae, 1987), sin embargo, esta difusión también disminuye la cantidad de azúcares, lípidos y compuestos aromáticos, haciendo que le grano sea más susceptible dentro del proceso de secado (Zou et al., 2022). Después de la extracción de la cafeína, se someten los granos de café a vapor directo (Stripping) para eliminar los restos de solvente, y allí se pueden producir pérdidas adicionales de compuestos solubles y cambios en la humedad (Zou et al., 2022).

Los cambios generados en la estructura del café en el proceso de descafeinado hacen que el grano se comporte de forma distinta al café tradicional durante el secado, explicando la variabilidad de humedad final que se presenta entre lotes en la empresa descafeinadora, por esto, se deben ajustar los parámetros de acuerdo con el tipo de café y las características solicitadas por el cliente.

Secado del Café

En el secado del café se remueve el agua de forma controlada aplicando calor. En este proceso por medio de la transferencia de calor y de masa, el agua dentro del interior del grano migra hacia la superficie de este. En las curvas de secado hay dos etapas, en la primera la velocidad es constante y la humedad se pierde rápidamente, en la segunda la velocidad disminuye al igual que la pérdida de humedad, y también hay mayor riesgo de sufrir daño térmico y degradación de compuestos (Brooker et al., 1992).

Durante el secado se deben controlar diferentes variables como la temperatura, debido a que temperaturas altas pueden hacer que el secado sea más rápido, pero aumentar el riesgo de

fractura del grano, en cambio, las temperaturas bajas hacen que el secado sea más controlado pero el costo energético es más elevado. Otra variable para considerar es la presión de vacío, presiones altas no remueven suficiente agua haciendo que el secado sea más lento y generar aumento en la temperatura. La distribución del vapor dentro del secador también es un factor esencial debido a que la uniformidad asegura la homogeneidad del proceso y evita las diferencias de humedad entre diferentes zonas del secador (Brooker et al., 1992).

Efectos de un Secado Incorrecto

Posterior al proceso de descafeinado, el grano se vuelve más susceptible a fluctuaciones de humedad debido a los cambios producidos como aumento de la porosidad, pérdida de compuestos solubles y aumento de la fragilidad del grano, estas variaciones generan que haya una tendencia a absorber y a liberar el agua más rápidamente en comparación con el café tradicional (Correa et al., 2009).

Cuando la humedad del café es muy baja hay una mayor degradación de compuestos (aldehídos, cetonas, fenoles, compuestos sulfurados) responsables de las características organolépticas como sabor y aroma (Kulapichitr et al., 2019). El sobre secado también genera daños mecánicos, aumenta las posibilidades de presentar micro fisuras y ruptura celular (Adamiec et al., 2014). Al disminuir mucho la humedad se generan pérdidas económicas debido a la pérdida de peso neto del producto final, por lo tanto, se recibe cierta cantidad de café, pero la cantidad final es inferior, y cada punto de porcentaje por debajo de los parámetros establecidos representa pérdidas económicas, más aún cuando los procesos son a gran escala.

En cuanto al sub secado (café con humedades por encima de los parámetros establecidos) se generan riesgos por inestabilidad microbiológica, debido a que la alta actividad de agua favorece el crecimiento de microorganismos como hongos, levaduras y bacterias, a su vez se

puede presentar fermentación indeseada generando malos sabores que son irreversibles (Murthy & Madhava Naidu, 2012).

Tener una humedad mayor a los parámetros definidos hace necesario tener un proceso extra de secado para alcanzar el rango objetivo, pero esto ocasiona costos extra a nivel operativo y energético, además de atrasos en las operaciones.

Pérdida de Humedad Durante el Enfriamiento

Para determinar la humedad final se debe tener en cuenta que durante el proceso de enfriamiento se pierde entre el 1.5% y el 2.5% de humedad, esto ocurre porque como la temperatura interna del grano sigue siendo alta, aún hay migración de agua desde el interior del grano hacia la superficie, haciendo necesario considerar este factor dentro del modelado (Brooker et al., 1992). Esta pérdida adicional de humedad explica por qué el factor de secado genera incertidumbre en la empresa descafeinadora, debido a que se producen diferencias entre la humedad programada y la humedad final obtenida después del enfriamiento, haciendo que el factor sea variable y difícil de anticipar con precisión.

Modelado

Los modelos predictivos son útiles en la optimización de las operaciones y en la selección de factores; en procesos agroindustriales como el café han tomado relevancia, debido a que el secado presenta comportamientos no lineales e involucra múltiples variables.

En el aprendizaje supervisado los modelos se entrenan por medio de unas variables de entrada y una variable de salida conocida y el modelo aprende la relación entre las variables para hacer predicciones con nuevos datos. Chindapan et al. (2025), Collazos-Escobar, Bahamón-Monje, et al. (2025) y Le et al. (2025) han demostrado que este método es capaz de predecir la humedad, el comportamiento cinético e higroscópico del secado para productos agroindustriales.

Como los procesos térmicos no son lineales y hay interacciones entre variables, los métodos tradicionales como las ecuaciones diferenciales pueden no ser tan efectivos para modelar el sistema, es por esto, que los modelos de machine learning han sido mayormente empleados para este tipo de procesos (Đaković et al., 2024).

El modelo de regresión lineal múltiple es uno de los métodos más sencillos y busca determinar una relación lineal entre las variables. Este modelo es útil como referencia para comparar el rendimiento con métodos más avanzados.

El modelo de árbol de decisión ha sido empleado en investigaciones como las de Collazos-Escobar, Gutiérrez-Guzmán, et al. (2025) para modelar procesos de deshidratación y predecir la humedad en productos agrícolas.

Con el modelo Random Forest se combinan múltiples árboles de decisión, esta técnica disminuye el sobreajuste y mejora la estabilidad del modelo. Dentro de sus investigaciones Collazos-Escobar, Gutiérrez-Guzmán, et al. (2025) mostraron la efectividad de este modelo en la predicción de factores en el proceso de secado.

Con el modelo Gradient Boosting hay una secuencia de árboles de decisión donde se corrigen los errores de los árboles anteriores. Este permite detectar patrones y dinámicas no lineales, y es muy útil cuando hay correlaciones complejas entre variables (Đaković et al., 2024).

Support Vector Regression (SVR) es un modelo que utiliza funciones kernel para modelar relaciones no lineales. Le et al. (2025) dentro de su investigación empleó el método mostrando su eficiencia en la predicción de la humedad.

Finalmente, las Redes Neuronales Artificiales (RNA) también se emplean para funciones no lineales, y se ha demostrado su precisión en investigaciones relacionadas con la humedad y el tiempo de secado (Chindapan et al., 2025).

Metodología

Dentro de este estudio se empleará la metodología CRISP-DM (Cross-Industry Standard Process for Data Mining) como una guía para desarrollar el modelo predictivo del factor de secado. Esta metodología es usada principalmente en proyectos de minería de datos y aprendizaje automático, y aplicada en la industria donde se integran datos que provienen de procesos reales (Le et al., 2025). La elección de esta metodología está alineada con los objetivos del proyecto debido a que están enfocados en el análisis del proceso de secado, en organizar los datos históricos disponibles y en construir el modelo capaz de predecir el factor de secado, disminuyendo la variabilidad existente por la elección del parámetro basado en la experiencia.

Comprensión del Negocio

En esta fase se analiza el proceso de secado de café en la empresa descafeinadora y se identifican las variables que influyen en el proceso de secado y en la selección del factor de secado, tales como: peso inicial del café, humedad final del café, número de secador empleado, origen/cliente del café y pérdidas de humedad durante el enfriamiento.

En investigaciones anteriores se muestra como estas variables pueden afectar la calidad fisicoquímica y organoléptica final del café (Abreu et al., 2025; Peñuela-Martínez et al., 2023).

Comprensión de los Datos

Durante esta etapa se realizará la recolección de los datos históricos disponibles en la empresa, se asegurará que se encuentren íntegros, en el mismo formato y que sean consistentes. En esta etapa se determinan patrones, se caracteriza el comportamiento de los datos y se detecta si existen sesgos que puedan afectar el modelo predictivo (Collazos-Escobar, Bahamón-Monje, et al., 2025).

Preparación de los Datos

En esta etapa se realiza la limpieza y organización de los datos para usarlos en los modelos de machine learning. Dentro de los pasos a seguir se encuentran: eliminar datos duplicados o inconsistentes, imputar valores faltantes, estandarizar variables numéricas y codificar las variables categóricas, por ejemplo, el cliente y el número de secador.

Modelado

Se encontrará el modelo óptimo para predecir el factor de secado con aprendizaje supervisado empleando: Regresión lineal múltiple, Árboles de decisión, Random Forest, Gradient Boosting, Support Vector Regression (SVR) y Redes neuronales artificiales. La selección de estos modelos se debe a que en investigaciones anteriores ya se ha mostrado su eficiencia en la predicción de la humedad, cinética de secado y otros factores relevantes en el secado (Chindapan et al., 2025; Collazos-Escobar, Bahamón-Monje, et al., 2025; Le et al., 2025).

Evaluación

Se compararán los resultados de los modelos para determinar cuál de ellos tiene la mayor precisión en la predicción del factor de secado. Adicionalmente se evaluará qué tan estable es el modelo frente a variaciones, su coherencia física, la interpretabilidad de este y el cumplimiento de los objetivos planteados.

Implementación

En esta última etapa se realiza la integración del modelo predictivo con el proceso dentro de la empresa descafeinadora, en donde se pueda visualizar el factor de secado recomendado para cada lote, se plantea el uso de Streamlit para facilitar la visualización y la adopción del modelo por parte de los supervisores y jefes de turno.

En la siguiente figura se muestra el resumen de la metodología anteriormente descrita.

Figura 1

Optimización del Secado de Café: Ciclo del Análisis de Datos

Nota. La imagen muestra el resumen de las fases empleadas para la optimización del secado de café. Imagen generada por inteligencia artificial (NotebookLM) el 21 de abril de 2026.

Resultados

Inicialmente se realiza un pretratamiento de los datos, para asegurar que la información empleada en el proyecto sea de calidad. Las variables que no son numéricas como el cliente, el secador y el tipo de café se convierten a formato numérico; al realizar este proceso también se puedo establecer cuáles registros no eran válidos o fueron ingresados de forma incorrecta.

Para establecer si el proceso de secado se realizó correctamente (el café no se sobre secó o resultó muy húmedo) se calcula el error en la humedad, que es la diferencia entre la humedad objetivo (que coincide con la humedad inicial) y la humedad final medida en el laboratorio; basado en el criterio aceptado en planta y por los clientes, se establece un rango de error de ±0.3 puntos porcentuales. De acuerdo con este criterio se crea la variable 'dentro de rango', que identifica los registros en los cuáles la humedad objetivo se encontraba en el rango establecido, se filtran los datos y se conservan sólo los registros que cumplen con este parámetro, esto con el objetivo de entrenar al modelo sólo con los datos dentro de especificaciones.

La base de datos inicialmente cuenta con un total de registros de 5644 y posteriormente con el filtrado de los datos se conservan sólo 2550, indicando una reducción del 54.81%.

Posterior al filtrado se realiza un análisis descriptivo de las variables más representativas para analizar su comportamiento y distribución.

Tabla 1

Análisis Descriptivo de las Variables

Variable	Cliente	Tipo de café	Humedad inicial	Peso	Secador	Factor de secado
Conteo	2550	2550	2550	2550	2550	2550
Media	15,73	2,7	11,38	4034,31	2,52	204,67
Desviación estándar	14,07	1,96	0,48	116,98	1,12	29,62
Mínimo	0	1	9,2	3366	1	102
25%	5	1	11,1	3964	2	181
50%	9	1	11,4	3996	3	200
75%	29	5	11,7	4172	4	224,5
Máximo	44	6	12,7	4306	4	317

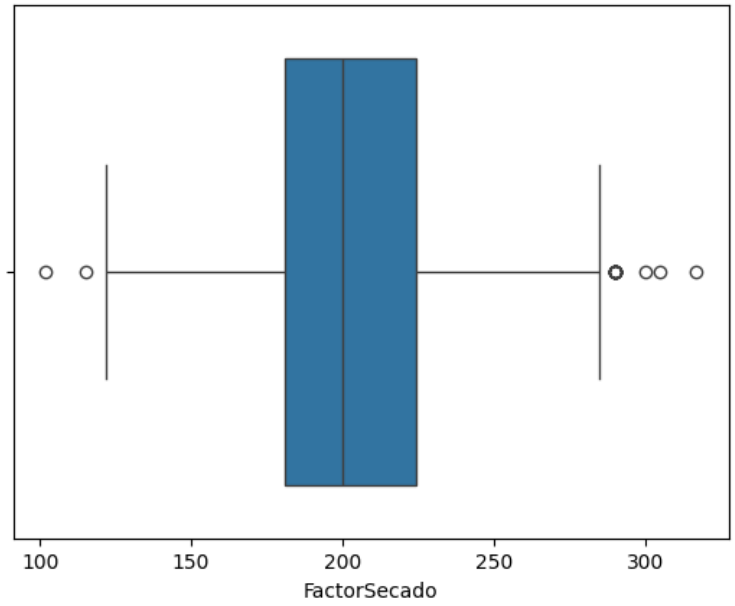
En cuanto al peso inicial, se muestra que el promedio es de 4034 kg, y una desviación de 116.98 kg, indicando que la variabilidad de pesos en los lotes procesados es moderada. La humedad inicial varía entre 9.2% y 12.7%, y tiene un promedio de 11.38%, que corresponde a las variaciones naturales de los cafés recibidos antes del descafeinado y secado.

En cuanto al factor de secado, que corresponde a la variable objetivo y representa el peso adicional con el que se detiene el secador para compensar la pérdida de humedad posterior durante el proceso de enfriamiento, este varía entre 102 y 317 y el promedio es de 204, indicando que hay gran variabilidad y afirmando la necesidad del desarrollo de modelos para estimar de forma más precisa este factor.

Para identificar los valores atípicos se emplean diagramas de cajas (bloxpots) para el factor de secado, la humedad inicial y el peso.

Figura 2

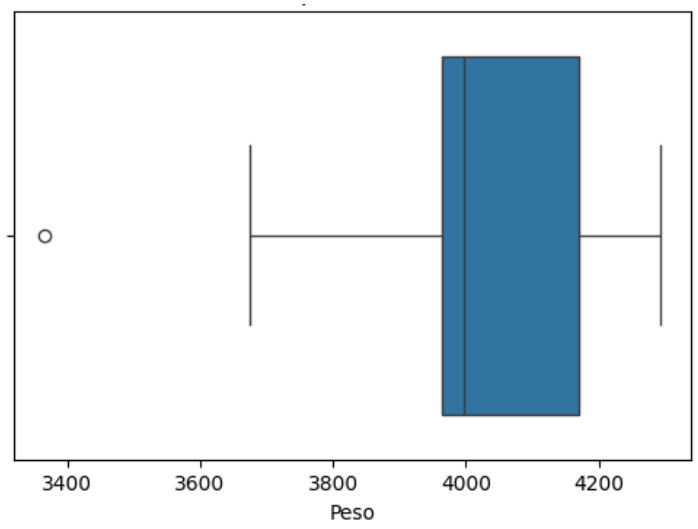
Bloxpot de la Variable Factor de Secado



El factor de secado presenta algunos valores atípicos, con valores de más de 276 y menos de 125, los cuales se pueden deber a errores de digitación o a algunas situaciones específicas donde puedo haber sido necesario seleccionar otro factor de secado, pero que no representan el comportamiento normal de la variable.

Figura 3

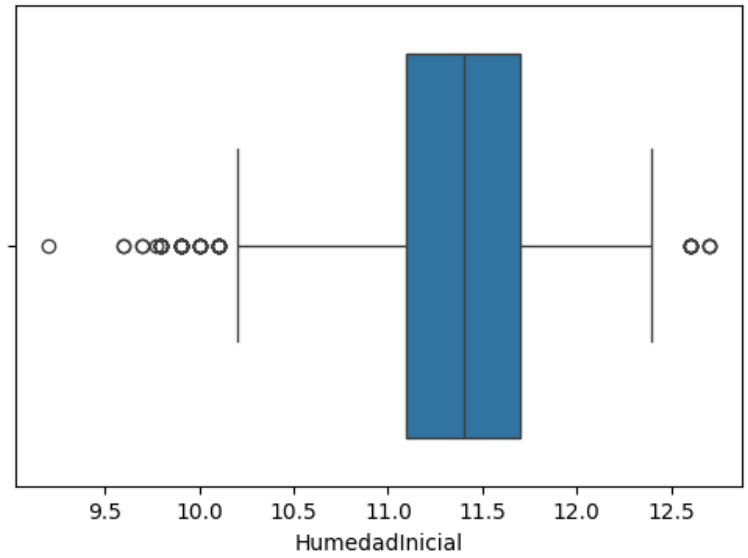
Bloxpot de la Variable Peso



Para el peso se encuentra un valor extremo, pero en este caso no necesariamente representa que corresponda a un error de digitación, y puede corresponder a un cliente que envió una cantidad menor para el proceso de descafeinado.

Figura 4

Bloxpot de la Variable Humedad Inicial



En cuanto a la humedad inicial se encuentran valores extremos, sin embargo, estos no están por fuera de las humedades lógicas que se pueden recibir de los clientes, es decir no corresponden a humedades menores a 9 que serían cafés demasiado secos o humedades por encima de 13 que representan un riesgo por el posible desarrollo de hongos, y en ocasiones algunos clientes solicitan humedades diferentes a las habituales.

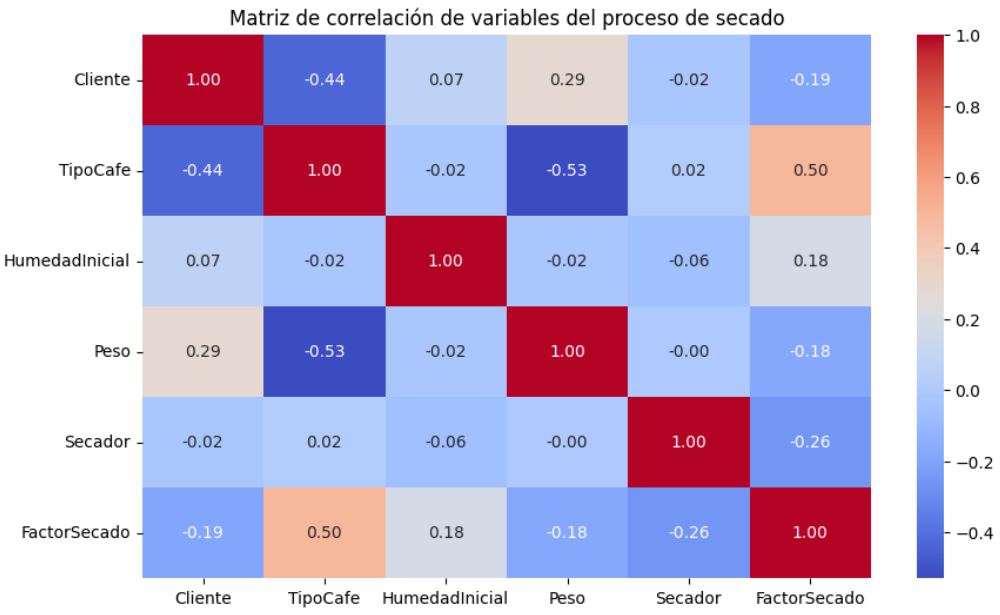
Se aplica el método del rango intercuartílico (IRQ) buscando mejorar la calidad de los datos al eliminar los valores atípicos, en este caso únicamente se eliminan los valores atípicos correspondientes al factor de secado, debido a que los correspondientes al peso y a la humedad si pueden ser valores reales aplicables dentro del proceso.

De los 2549 registros iniciales, se eliminan 23, obteniendo un valor final de 2526 registros para el entrenamiento del modelo de aprendizaje.

Con el objetivo de determinar las variables más influyentes dentro del proceso, se analizan las relaciones entre las variables empleando una matriz de correlación

Figura 5

Matriz de Correlación de Variables del Proceso de Secado



La mayor correlación observada con respecto al factor de secado corresponde al tipo de café (0.50), mostrando que el tipo de café si tiene influencia en el comportamiento del secado y se debe incluir dentro de los modelos de aprendizaje automático para predecir el factor de secado. En estudios previos también se ha demostrado que el tipo de café posee características que tienen influencia en la cinética del secado y la transferencia de masa, Tosta et al. (2020) muestra que distintos tipos de café tienen comportamientos diferentes en las velocidades de secado y el modelo que describe el proceso también tiene variaciones.

La humedad inicial también tiene una correlación positiva, en este caso no es tan fuerte, es más moderada (0.18) y muestra que entre mayor es la humedad inicial, se requiere también un mayor factor de secado para compensar la pérdida de humedad durante el enfriamiento. Esta correlación también corresponde con la realidad del secado, ya que cuando hay mayor cantidad de agua en el grano también hay una mayor pérdida de esta en el proceso de secado y enfriamiento, en las investigaciones de Burmester & Eggers (2010) se demuestra que el contenido de humedad del grano influye directamente en la difusividad de la humedad y en las propiedades de transferencia de masa.

En cuanto al secador, este presenta una correlación negativa también moderada (-0.26), indicando que a pesar de que los secadores tienen características de construcción similares no se comportan igual durante el secado, estas diferencias pueden atribuirse a los mantenimientos realizados o a la eficiencia térmica; Erbay & Icier (2010) demuestran que pequeños cambios en los diseños o en las condiciones de operación influyen en la cinética del secado, pudiendo ser algunos de los secadores más efectivos para disminuir la humedad que otros, haciendo que algunos requieran un mayor factor para compensar las pérdidas en los procesos posteriores.

El peso tiene una relación negativa y baja (-0.18) indicando que esta variable si puede tener una influencia dentro del proceso de secado, pero esta es mucho menor que las anteriores variables mencionadas y puede no genera tanto impacto en la determinación del factor de secado. Sin embargo, en la literatura se ha reportado que cuando las cantidades cargadas a los secadores son muy altas, se pueden generar resistencias en la transferencia de calor y masa (Brooker et al., 1992).

La variable cliente tiene también una correlación negativa y baja (-0.19), al igual que el peso indica que esta variable no tiene mucha influencia a la hora de establecer el factor de secado, esto puede deberse a que distintos clientes pueden tener más de un tipo de café cuyas características como densidad, la porosidad y la capacidad de retener agua varíen.

En resumen, las variables que más influencia tienen en la determinación del factor de secado son el tipo de café, la humedad inicial y el secador, mientras que las otras variables analizadas como el peso o el cliente tienen una menor influencia.

Determinación del Mejor Modelo

Con el objetivo de establecer el mejor modelo para la predicción del factor de secado se realizan diferentes pruebas.

Prueba 1

Para la primera prueba se toman únicamente las variables físicas de humedad inicial, tipo de café y secador que corresponden a las variables que obtuvieron mayor correlación con el factor de secado. Se analizaron los 6 modelos planteados en la metodología: Regresión lineal múltiple, Árboles de decisión, Random Forest, Gradient Boosting, Support Vector Regression (SVR) y Redes neuronales artificiales, y se calcula el error absoluto medio (MAE), la raíz del

error cuadrático medio (RMSE) y el coeficiente de determinación (R^2), obteniendo los siguientes los siguientes resultados.

Tabla 2

Resultados por Modelo Prueba 1

Modelo	MAE	RMSE	R2
GradientBoosting	13,028	16,527	0,647
RandomForest	13,793	17,495	0,604
ArbolDecision	14,119	18,093	0,576
SVR	16,725	21,395	0,408
RegresionLineal	16,885	21,724	0,389
RedNeuronal	17,103	22,026	0,372

El mejor modelo para esta prueba es Gradient Boosting con una MAE de 13,028. Mostrando que a pesar de tener una cantidad limitada de variables si es posible modelar parte del comportamiento del proceso de secado. El valor del R^2 no es muy alto, mostrando que el proceso no se describe completamente con sólo estas variables.

Los modelos de SVR y red neuronal presentan menores ajustes y mayores errores, esto puede deberse a que SVR es muy sensible a hiperparámetros y a la escala de las variables, en el caso de red neuronal se necesita un volumen alto de datos para obtener mejores rendimientos, y en este caso la cantidad de datos es limitada.

Prueba 2

Se realiza la misma prueba anterior empleando los mismos modelos y con los mismos cálculos estadísticos, pero en este caso con todas las variables (se añade peso y cliente) buscando

un mejor modelado de los datos. Aunque estas correlaciones son menores en comparación a las anteriores, también muestran que tienen cierta influencia dentro del modelo y se evalúa si la capacidad predictiva mejora al añadir esta información. Se obtienen los siguientes resultados:

Tabla 3

Resultados por Modelo Prueba 2

Modelo	MAE	RMSE	R2
GradientBoosting	12,465	15,931	0,672
RandomForest	12,867	16,438	0,650
ArbolDecision	16,225	21,181	0,420
RegresionLineal	16,812	21,599	0,396
RedNeuronal	20,949	26,441	0,095
SVR	22,561	27,722	0,006

Se observa que hay una mejora en los modelos de gradient boosting (MAE: 12,465) y random forest (12,867), a pesar de que esta es pequeña, muestra que añadir las variables de peso y cliente permite explicar de mejor manera el modelo; como el cambio no es muy grande, se muestra que no siempre al aumentar el número de variables mejorará considerablemente el modelo.

En cuanto a los demás modelos como árbol de decisión, red neuronal y SRV desmejoran. El de árbol de decisión puede aumentar su complejidad cuanto se integran todas las variables aumentando el riesgo de un sobreajuste y disminuyendo su capacidad para generalizar el modelo, igualmente el modelo regresión lineal no mejora al añadir otras variables por su incapacidad en el modelado de relaciones no lineales.

Los modelos de SVR y red neuronal al añadir más variables también requieren mayor cantidad de datos para aprender las relaciones entre ellas, y esto puede no cumplirse en este caso, generando que el error aumente.

Prueba 3

En esta prueba se usan modelos independientes para cada uno de los clientes, buscando establecer si el origen del grano influye en el secado del café. Se restringe la cantidad de datos mínimos para el modelado (50 registros) para asegurar que no exista un sobreajuste en los modelos.

Tabla 4

Resultados por Cliente y Modelo Prueba 3

Cliente	Modelo	MAE	RMSE	R2	Cliente	Modelo	MAE	RMSE	R2
0	RandomForest	13,864	17,496	0,576	20	RegresionLineal	14,942	18,691	0,286
0	RegresionLineal	15,19	22,53	0,297	20	SVR	15,808	22,133	-0,001
0	GradientBoosting	18,84	21,796	0,342	20	RedNeuronal	20,39	25,32	-0,31
0	ArbolDecision	19,179	23,791	0,216	20	RandomForest	20,845	24,31	-0,207
0	SVR	20,308	28,146	-0,097	20	GradientBoosting	24,686	28,205	-0,625
0	RedNeuronal	23,182	31,941	-0,412	20	ArbolDecision	31,308	37,407	-1,859
1	GradientBoosting	12,446	16,592	0,298	22	GradientBoosting	9,564	11,553	0,717
1	RandomForest	13,325	16,385	0,316	22	RandomForest	9,926	11,856	0,702
1	RegresionLineal	14,213	18,236	0,153	22	ArbolDecision	11,933	14,883	0,531
1	SVR	16,458	20,198	-0,04	22	RegresionLineal	12,463	16,707	0,409
1	RedNeuronal	17,092	20,316	-0,052	22	RedNeuronal	17,801	22,281	-0,051
1	ArbolDecision	17,587	22,506	-0,291	22	SVR	17,86	21,772	-0,004
5	RandomForest	13,233	17,163	0,586	29	RegresionLineal	12,442	16,799	0,438
5	GradientBoosting	13,251	16,849	0,601	29	GradientBoosting	12,454	15,473	0,523
5	ArbolDecision	16,486	21,971	0,321	29	RandomForest	12,722	15,946	0,494
5	RegresionLineal	18,912	22,995	0,256	29	ArbolDecision	14	17,196	0,411
5	RedNeuronal	21,381	27,25	-0,044	29	SVR	18,639	22,84	-0,039
5	SVR	21,425	27,144	-0,036	29	RedNeuronal	21,171	25,103	-0,255
8	RandomForest	13,708	16,179	0,33	33	RandomForest	12,563	15,282	0,181
8	GradientBoosting	14,175	17,43	0,223	33	RedNeuronal	12,705	16,046	0,097
8	ArbolDecision	14,857	17,885	0,182	33	RegresionLineal	13,048	14,79	0,233
8	RegresionLineal	15,401	18,338	0,14	33	SVR	13,937	17,414	-0,063
8	SVR	16,83	22,715	-0,32	33	GradientBoosting	14,207	16,162	0,084

Cliente	Modelo	MAE	RMSE	R2	Cliente	Modelo	MAE	RMSE	R2
8	RedNeuronal	17,702	23,538	-0,417	33	ArbolDecision	14,75	17,92	-0,126
9	SVR	13,39	17,238	0	38	RandomForest	9,221	12,1	0,562
9	RegresionLineal	13,599	16,661	0,066	38	GradientBoosting	10,937	14,368	0,383
9	RandomForest	16,258	20,458	-0,408	38	RegresionLineal	13,807	17,058	0,13
9	RedNeuronal	16,648	20,075	-0,356	38	ArbolDecision	14,86	20,351	-0,239
9	GradientBoosting	17,559	22,328	-0,678	38	SVR	15,677	18,3	-0,001
9	ArbolDecision	19,14	25,937	-1,264	38	RedNeuronal	16,017	20,226	-0,223
14	RandomForest	13,22	16,323	0,782	39	GradientBoosting	15,085	18,577	0,57
14	GradientBoosting	13,76	17,631	0,746	39	RandomForest	16,284	19,275	0,537
14	ArbolDecision	15,722	21,434	0,625	39	ArbolDecision	17,154	25,37	0,199
14	RegresionLineal	20,587	24,939	0,492	39	RegresionLineal	19,067	24,263	0,267
14	SVR	31,086	36,585	-0,094	39	SVR	23,793	28,348	-0,001
14	RedNeuronal	33,202	37,679	-0,16	39	RedNeuronal	24,919	30,506	-0,159
18	GradientBoosting	14,64	20,851	0,415	41	RandomForest	12,052	14,313	0,558
18	ArbolDecision	15,546	20,607	0,429	41	GradientBoosting	12,385	14,461	0,549
18	RandomForest	16,221	20,078	0,458	41	ArbolDecision	13,171	15,944	0,451
18	RegresionLineal	21,419	26,314	0,069	41	RegresionLineal	17,1	19,956	0,14
18	SVR	22,449	29,496	-0,17	41	RedNeuronal	18,001	22,638	-0,106
18	RedNeuronal	23,571	28,375	-0,083	41	SVR	18,21	21,803	-0,026

Solamente 14 de los 45 clientes cumplen con la condición de tener más de 50 registros. Se evidencia una alta variabilidad en los modelos, indicando que el proceso de secado está relacionado con las características propias de cada origen. Clientes como el 14 y 22 tienen ajustes muy altos (superiores a 0.7), mostrando que el modelo logra explicar de manera adecuada el comportamiento de las variables, sin embargo, otros cliente como el 9, 20, 33 y 39 tienen valores de R2 muy bajos e incluso negativos, indicando que el modelo no logra explicar el comportamiento de las variables, que puede deberse a que tienen poca cantidad de datos, a pesar de establecer un mínimo de 50 registros, esta puede ser insuficiente, debido a la variabilidad de los clientes, por ejemplo, clientes que tienen más de una variedad de café o diferentes cultivos que generan variabilidad en el grano.

En relación con los modelos analizados, los que menor error presentan y mayor ajuste son Random Forest y Gradient Boosting, siguiendo lo encontrado en las pruebas 1 y 2.

Entre los modelos se selecciona el que tiene el menor promedio de error absoluto, para así visualizar el resumen con todos los clientes.

Tabla 5

Resultados Error Absoluto con el Modelo de Random Forest Prueba 3

Cliente	MAE
22	9,366
38	9,371
29	12,289
0	12,548
33	12,666
1	12,848
41	13,269
5	13,798
8	14,544
14	15,941
18	16,278
9	16,318

Se presentan clientes con MAE bajo (cliente 22 y 38), en cambio, se encuentra otros clientes (18 y 9) donde el MAE es superior a 16, que muestra una menor capacidad predictiva.

Prueba 4

Similar a la prueba anterior se usan modelos independientes para cada uno de los tipos de café, buscando determinar si las características de cada tipo de grano influyen en el proceso de secado. Se obtienen los siguientes resultados:

Tabla 6

Resultados por Cliente y Modelo Prueba 3

Tipo	Modelo	MAE	R2	Tipo	Modelo	MAE	R2
1	GradientBoosting	12,36	0,508	2	RandomForest	12,05	0,322
1	RandomForest	12,944	0,468	2	GradientBoosting	12,056	0,332
1	ArbolDecision	14,819	0,278	2	ArbolDecision	14,4	-0,033
1	RegresionLineal	15,946	0,184	2	SVR	15,084	-0,006
1	SVR	17,941	0,001	2	RegresionLineal	15,745	-0,019
1	RedNeuronal	18,052	-0,021	2	RedNeuronal	132,64	-47,764
5	GradientBoosting	14,036	0,455	6	GradientBoosting	15,085	0,372
5	RandomForest	15,942	0,324	6	RegresionLineal	17,049	0,102
5	RegresionLineal	17,09	0,173	6	RandomForest	17,222	0,254
5	RedNeuronal	18,642	-0,03	6	ArbolDecision	19,185	-0,075
5	SVR	19,018	-0,108	6	SVR	19,453	-0,042
5	ArbolDecision	20,236	-0,076	6	RedNeuronal	122,509	-26,521

Gradient Boosting y Random Forest al igual que en las pruebas anteriores continúan teniendo los menores errores y los ajustes más altos. Los tipos de café 1 y 2 tienen los errores más bajos (12,36 y 12,05), mientras que los tipos 5 y 6 tiene los errores más altos y los menores ajustes.

Entre los modelos se selecciona el que tiene el menor promedio de error absoluto, para así visualizar el resumen con todos los tipos de café.

Tabla 7

Resultados Error Absoluto con el Modelo de Gradient boosting Prueba 4.

Tipo	MAE
1	12,360
2	12,056
5	14,036
6	15,085

Algunos tipos del café como el número 1 tienen el error absoluto más bajo (12,360), sin embargo, el error del tipo 6 es más alto (15,085).

Con base a los resultados se muestra que si existe variación en el comportamiento del secado entre tipos de café, que puede deberse a la estructura interna del grano, a su capacidad de retener la humedad y a su composición química.

Prueba 5

Se optimizan los parámetros de la prueba 2, para los cliente y tipos de café que por falta de datos no fueron modelados en las pruebas 3 y 4. Se realiza solamente para los dos modelos que obtuvieron los mejores resultados (Gradiente Boosting y Random Forest).

Tabla 8

Resultados Parámetros Optimizados, Modelo General Prueba 5.

Modelo	MAE	RMSE	R2
GradientBoosting Optimizado	12,364	15,702	0,681
RandomForest Optimizado	12,539	16,046	0,667

Se obtienen los siguientes parámetros óptimos de los modelos:

Random Forest: n_estimators=200; max_depth=30; min_samples_split=2; min_samples_leaf=4; bootstrap=True

Gradient Boosting: n_estimators=500; learning_rate=0,01; max_depth=5; min_samples_split=5; subsample=0,8.

Para Gradient Boosting un valor de learning rate de 0,01 es bajo, lo que permite que el aprendizaje sea más lento y que la corrección de errores sea más progresiva, mejorando la capacidad de generalizar; n_estimators es alto (500) compensando la tasa de aprendizaje baja; max_depth tiene una profundidad moderada pata evitar ajustarse también al ruido (Friedman, 2002)

En el caso de Random Forest tiene un valor de max Depth de 30, el cual es alto pudiendo generar sobreajustes, pero este se compensa empleando múltiples árboles (n_estimators: 200); boostrap se define como verdadero, mejorando la diversidad y la capacidad para generalizar porque cada árbol se entrena con una muestra diferente del conjunto de datos (Breiman, 2001).

Prueba 6

Se optimizan los parámetros de la prueba 3 para cada cliente; al igual que la prueba anterior sólo se optimizan los dos modelos con mejores resultados (Gradiente Boosting y Random Forest).

Tabla 9

Resultados Parámetros Optimizados, Modelo por Cliente Prueba 6.

Cliente	Modelo	MAE	RMSE	R2	MAE sin optimizar
0	RandomForest	12,699	17,729	0,565	13,864
0	GradientBoosting	18,166	22,385	0,306	18,84
1	RandomForest	12,35	15,442	0,392	13,325
1	GradientBoosting	12,327	15,291	0,404	12,446
5	RandomForest	13,074	16,696	0,608	13,233
5	GradientBoosting	13,033	16,523	0,616	13,251
8	RandomForest	12,354	14,801	0,44	13,708
8	GradientBoosting	14,088	16,792	0,279	14,175
9	RandomForest	14,869	18,6	-0,164	16,258
9	GradientBoosting	18,793	23,344	-0,834	17,559
14	RandomForest	13,93	17,053	0,762	13,22
14	GradientBoosting	13,899	16,952	0,765	13,76
20	RandomForest	17,851	20,041	0,179	20,845
20	GradientBoosting	21,698	24,165	-0,193	24,686
22	RandomForest	9,412	11,564	0,717	9,926

Cliente	Modelo	MAE	RMSE	R2	MAE sin optimizar
22	GradientBoosting	9,988	12,234	0,683	9,564
29	RandomForest	12,31	15,945	0,494	12,722
29	GradientBoosting	12,272	15,51	0,521	12,454
33	RandomForest	12,452	15,141	0,196	12,563
33	GradientBoosting	13,586	15,556	0,151	14,207
38	RandomForest	9,475	12,2	0,555	9,221
38	GradientBoosting	9,649	13,131	0,484	10,937
39	RandomForest	18,616	21,934	0,401	16,284
39	GradientBoosting	20,352	22,582	0,365	15,085
41	RandomForest	11,829	14,127	0,569	12,052
41	GradientBoosting	11,247	13,433	0,61	12,385

Se observan algunas mejoras, sin embargo, no ocurre en todos los clientes, donde se siguen presentando valores de R^2 negativos o muy bajos, también se presentan clientes con MAE más bajos en comparación con el modelo general como el 1, 29, 38, 22 y 41, aún así en general los errores superan los encontrados en la prueba 5 (MAE: 12,364).

A pesar de que la optimización de los hiperparámetros del modelo permitió mejoras en el desempeño para la mayoría de los clientes, se presentan casos donde hay leves incrementos en el error, lo que puede deberse a la presencia de ruidos en los datos, poca cantidad de datos por cliente, indicando que no en todos los casos la optimización de los parámetros de los modelos resulta en la disminución de los errores.

Prueba 7

Se optimizan los parámetros de la prueba 4 para cada tipo de café; se optimizan los dos modelos con mejores resultados (Gradiente Boosting y Random Forest).

Tabla 10

Resultados Parámetros Optimizados, Modelo por Tipo de Café Prueba 7.

Tipo de Café	Modelo	MAE	RMSE	R2	MAE sin optimizar
1	GradientBoosting	12,289	16,005	0,515	12,360
1	RandomForest	12,745	16,406	0,490	12,944
2	GradientBoosting	12,440	15,554	0,343	12,056
2	RandomForest	12,355	15,636	0,336	12,050
5	GradientBoosting	13,845	17,102	0,457	14,036
5	RandomForest	14,626	17,842	0,409	15,942
6	GradientBoosting	15,653	19,265	0,350	17,222
6	RandomForest	16,760	20,426	0,269	15,085

Se observan leves mejoras en el MAE con los hiperparámetros optimizados, en todos los tipos de café a excepción del tipo 6 donde el error desmejora.

En comparación con el modelo general optimizado solo el tipo 1 con el modelo de Gradient Boosting y el tipo 2 con el modelo Random Forest tienen menores MAE (12,289 y 12,355 respectivamente), sin embargo, la diferencia es muy baja y los demás tipos tienen errores relativamente más altos, mostrando que realizar diferentes modelos para cada tipo de café no

representa una mejora significativa al igual que en la prueba anterior. Adicionalmente no se modelaron los tipos de café 3 y 4 por falta de registros.

Selección del Modelo

En la prueba 1 donde se tomaron solamente 3 variables se obtuvo que los modelos de Gradient Boosting y Random Forest tienen los menores errores de 13,028 y 13,793 respectivamente. En la prueba 2 se consideraron todas las variables (5) y al igual que en la prueba anterior se obtienen Gradient Boosting (MAE: 12,465) y Random Forest (MAE: 12,867) como los mejores modelos, en comparación con la prueba 1 se observa una disminución en el error y una mejora en el coeficiente de determinación.

Las pruebas 3 y 6 que corresponden a los modelos por cliente antes y después de la optimización de los hiperparámetros, mostraron buenos resultados para algunos clientes, sin embargo, no fue posible mejorar la consistencia de todos los clientes, mostrando que dividir los datos por cliente no mejora el desempeño del modelo, lo que se debe a la limitación en la cantidad de datos y a la variabilidad propia del proceso.

En las pruebas 4 y 7 donde se realizan modelos por tipo de café ocurre algo similar, también se obtienen mejoras en ciertos tipos de café, pero en otros los errores son más altos. A pesar de que el tipo de café es una variable importante y con una de las correlaciones más altas en relación con el factor de secado los modelos individuales no tienen una mejora significativa en comparación con el modelo general.

En la prueba 5 se realizó la optimización de los hiperparámetros del modelo general, el mejor resultado obtenido fue con el modelo de Gradient Boosting, superando levemente al modelo de Random Forest, obteniendo un MAE de 12,364, RMSE de 15,702 y R^2 de 0,681. Este

modelo tiene en general el menor error en la predicción en comparación con las demás pruebas realizadas, y el mejor ajuste (R^2).

Por lo tanto, se define el modelo de gradient boosting con los hiperparámetros optimizados como el mejor modelo para la predicción del factor de secado, debido a que tiene la mayor estabilidad, precisión en los resultados y desempeño global.

Implementación del Modelo Dentro del Proceso Productivo

Se realiza la integración del modelo de secado seleccionado, buscando apoyar la toma de decisiones dentro del proceso operativo para la estimación del factor de secado, para esto se emplea una aplicación desarrollada en Streamlit en donde se ingresan las variables del proceso y se obtienen el factor para la operación, como se muestra en la figura 6.

Se realiza un instructivo para el uso del aplicativo en el apéndice C.

Figura 6

Interfaz Aplicación de Predicción del Factor de Secado

Para el registro de la información se emplea el formato de la tabla 11, se ingresan las mismas variables que en la aplicación y el factor de secado calculado por esta, se registra el valor de secado aproximado que es el valor redondeando por encima a 5 o a 0, es decir, si es valor calculado es de 203 se aproxima a 205, debido al nivel de incertidumbre de las básculas.

Los supervisores basados en el valor calculado y su experiencia determinan si este valor es óptimo para ser usado dentro del proceso, en caso de usarlo se ingresa el mismo valor en la casilla de factor de secado utilizado, en caso contrario, se registra el valor considerado por el supervisor.

Al finalizar el proceso se ingresa la humedad de laboratorio, se calcula la diferencia entre el factor aproximado y el utilizado, y a partir de la humedad de laboratorio y el factor de secado utilizado se extrapola la humedad que se hubiera obtenido con el factor de secado aproximado. Con el criterio de que la humedad debe estar en el rango de ±0.3% de la humedad inicial/objetivo, se determina si el factor aproximado cumple con las especificaciones.

Tabla 11

Formato de Factores de Secado Empleados

Fecha inicio proceso (aaaa/mm/dd)	Cliente	Tipo de café	Humedad Inicial/objetivo	Peso	Secador	Factor de secado calculado

Factor de secado aproximado	Factor de secado utilizado	Humedad Laboratorio	Diferencia entre factores	Humedad calculada	¿El factor aproximado cumple con la humedad?

Se tomaron 100 datos reales de planta y se registraron en el formato de tabla 11, para tener una evaluación inicial del desempeño del modelo en planta.

Tabla 12

Registro de Factores de Secado Empleados

Fecha inicio proceso	Client	Tipo Cafe	Hum Inicial / obj	Peso	Secad	Factor de secado calc	Factor de secado aprox	Factor de secado progr	Hum Lab	Hum calc	¿El factor aproxim cumple con la humedad?
6/04/2026	39	2	11,6	4186	1	189,5	190	200	11,3	10,7	no
6/04/2026	39	2	11,6	4190	2	201,81	205	199	10,8	11	no
6/04/2026	14	4	10,6	4024	3	204,31	205	210	10,4	10,1	no
6/04/2026	14	4	10,6	3896	4	181,09	185	175	10,7	11,1	si
6/04/2026	14	4	10,6	4306	1	190,36	195	180	10,6	11,2	si
6/04/2026	14	4	10,6	4068	1	199,36	200	180	11	12,2	no
6/04/2026	14	4	10,6	4050	2	198,19	200	185	11	11,8	no
6/04/2026	14	4	10,6	4054	4	194,26	195	210	10,7	9,9	no
6/04/2026	14	4	10,6	4076	2	199,22	200	190	10,4	10,9	si
6/04/2026	14	4	10,6	4064	1	200,03	200	190	11,4	12	no
7/04/2026	5	4	11,7	3974	2	212	215	207	12,1	12,4	si
7/04/2026	5	4	11,7	3976	3	223,58	225	208	11,8	12,7	no
7/04/2026	5	4	11,7	3974	1	212	215	190	11,3	12,6	si
7/04/2026	5	4	11,7	3992	4	199,12	200	180	11,1	12,3	si
7/04/2026	5	6	11	3932	2	231,83	235	224	10,7	11,1	si
7/04/2026	5	6	11	3942	3	237,01	240	203	10,8	12,6	no
8/04/2026	5	6	11	3930	1	228,02	230	213	11,4	12,2	no
8/04/2026	5	6	11,5	3974	4	219,49	220	202	11,3	12,3	si
8/04/2026	5	6	11,5	3966	2	234,35	235	226	11,4	11,8	si
8/04/2026	5	6	11,5	3964	3	244,42	245	221	11,5	12,7	no
8/04/2026	5	5	11,4	3988	1	227,28	230	229	11,2	11,1	si
8/04/2026	5	5	11,4	3984	4	218,34	220	230	11,2	10,6	no
8/04/2026	5	5	11,4	3966	2	233,42	235	230	11,6	11,8	si
8/04/2026	5	5	11,4	3992	3	239,74	240	240	11,5	11,5	si
8/04/2026	5	5	11,4	3989	1	227,28	230	213	11,5	12,3	si
8/04/2026	5	5	11,5	3980	4	221,3	225	206	11,5	12,4	si
8/04/2026	5	5	11,6	3986	2	237,62	240	224	11,5	12,2	si
9/04/2026	5	5	11,6	3978	3	247,32	250	234	11,9	12,6	no
9/04/2026	5	5	11,6	3772	1	215,84	220	220	11,4	11,2	no
9/04/2026	5	5	11,6	4198	4	224,46	225	210	11,4	12,2	si
9/04/2026	5	5	11,6	3960	2	238,58	240	230	11,2	11,6	si
9/04/2026	5	5	11,7	3970	3	245,67	250	240	11,8	12,1	si
9/04/2026	5	5	11,77	3998	1	229,82	230	218	12,2	12,9	no
9/04/2026	5	5	11,77	3996	4	218,14	220	224	11,2	10,9	no
9/04/2026	5	5	11,77	3974	2	237,72	240	249	11,5	11	no
9/04/2026	5	5	11,77	3940	3	246,39	250	238	11,9	12,3	si
9/04/2026	5	5	11,22	4034	1	231,32	235	220	10,9	11,5	si

Fecha inicio proceso	Client	Tipo Cafe	Hum Inicial / obj	Peso	Secad	Factor de secado calc	Factor de secado aprox	Factor de secado progr	Hum Lab	Hum calc	¿El factor aproxim cumple con la humedad?
9/04/2026	5	5	10,6	3990	4	213,87	215	219	10,9	10,6	si
10/04/2026	5	5	10,6	3986	2	226,85	230	239	10,4	9,9	no
10/04/2026	5	5	10,6	3966	3	239,94	240	231	11,5	11,9	no
10/04/2026	5	5	10,6	3926	1	223,69	225	198	11,3	12,8	no
10/04/2026	5	5	10,6	4014	4	211,46	215	189	10,8	12,1	no
10/04/2026	5	5	10,6	3978	2	227,34	230	230	10,8	10,7	si
10/04/2026	5	5	10,6	3972	3	240	240	236	10,2	10,4	si
10/04/2026	5	5	10,6	3976	1	223,83	225	176	10,6	13,5	no
10/04/2026	5	5	10,3	3982	4	196,6	200	200	10,8	10,6	si
10/04/2026	5	5	10,3	4006	2	193,87	195	210	11	10,2	si
10/04/2026	5	5	10,3	3980	3	250	250	230	10,8	11,7	no
10/04/2026	5	5	10,3	3994	1	198,87	200	219	10,3	9,4	no
11/04/2026	5	5	10,3	3992	4	193,31	195	192	10,8	10,9	si
11/04/2026	5	5	10,3	3990	2	201,19	205	214	10,6	10	si
11/04/2026	5	5	10,3	3986	3	247,63	250	230	10,7	11,5	no
11/04/2026	5	5	10,3	4004	1	195,38	200	189	10,8	11,2	si
11/04/2026	14	5	11,4	3730	4	203,56	204	212	11,6	11,1	si
11/04/2026	14	5	11,4	3720	2	224,28	225	203	11	12,2	si
11/04/2026	14	5	11,4	3728	3	233,2	250	229	11,7	11,9	si
11/04/2026	14	5	11,4	3724	1	211,32	215	161	10,6	13,9	no
11/04/2026	14	5	11,4	3734	4	203,56	205	187	11,4	12,4	no
11/04/2026	14	5	11,4	3738	3	234,47	235	214	12	13,1	no
12/11/2026	14	5	11,4	3726	1	214,38	215	190	11,6	13,1	no
12/11/2026	14	5	11,4	3724	4	201,8	205	190	12	12,7	no
12/04/2026	14	5	11,4	3714	3	232,46	235	232	11,9	11,9	si
12/04/2026	30	1	11,4	4192	2	198,9	200	230	12,1	10,5	no
12/04/2026	30	1	11,3	4188	1	196,5	200	210	11,9	11,1	si
12/04/2026	30	1	11,1	4198	4	191,85	195	174	11,6	12,8	no
12/04/2026	32	1	11,7	4186	3	219,08	220	201	11,8	12,9	no
12/04/2026	22	1	11,8	4178	2	208,58	210	202	11,7	12,1	si
12/04/2026	22	1	11,8	4184	1	204,38	205	200	11,8	12,1	si
13/04/2026	22	1	11,8	4184	4	197,42	200	180	12,1	13,3	no
13/04/2026	22	1	11,8	4178	3	222,73	225	210	11,6	12,3	si
13/04/2026	22	1	11,8	4180	2	208,7	210	210	11	10,9	no
13/04/2026	22	1	11,5	4174	1	201,86	205	210	11,7	11,2	si
13/04/2026	22	1	11,5	4182	4	196,98	200	200	11,6	11,4	si
13/04/2026	22	1	11,5	4174	3	224,02	225	220	11,5	11,7	si
13/04/2026	22	1	11,6	4162	2	206,59	210	210	12,1	11,9	si
13/04/2026	22	1	11,7	4190	1	200,88	205	188	12,3	13,1	no
13/04/2026	28	1	11,4	4174	4	197,48	200	172	11,9	13,7	no
13/04/2026	40	1	11,3	4162	3	202,43	205	200	11,6	11,7	si
14/04/2026	40	1	11,3	4168	2	202,16	205	200	11,2	11,3	si
14/04/2026	38	1	11,2	4164	1	196,78	200	190	11,8	12,2	no
14/04/2026	37	1	11,3	4158	4	200,65	205	180	12	13,4	no
14/04/2026	13	1	11,5	4164	3	222,1	225	200	11,8	13,1	no
14/04/2026	0	1	11,7	4178	2	198,99	200	200	11,7	11,6	si

Fecha inicio proceso	Client	Tipo Cafe	Hum Inicial / obj	Peso	Secad	Factor de secado calc	Factor de secado aprox	Factor de secado progr	Hum Lab	Hum calc	¿El factor aproxim cumple con la humedad?
14/04/2026	0	1	11,7	4174	1	196,84	200	200	11,9	11,7	si
14/04/2026	9	1	12,1	4196	4	190,09	195	190	12	12	si
14/04/2026	9	1	12,1	4202	2	196,67	200	220	11,8	10,5	no
14/04/2026	9	1	12	4202	3	204,7	205	220	11,8	11	no
14/04/2026	9	1	12	4226	1	195,4	200	210	12	11,2	no
15/04/2026	9	1	12	4186	4	192,26	195	190	12,1	12,2	si
15/04/2026	1	1	11,5	4176	2	197,86	200	210	10,9	10,3	no
15/04/2026	1	1	11,5	4170	3	222,63	225	224	11,1	11	no
15/04/2026	1	1	11,5	4180	1	194,11	195	200	12	11,6	si
15/04/2026	1	1	11,5	4182	4	190,52	195	200	12	11,4	si
15/04/2026	1	1	11,5	4180	2	198,32	200	210	11,9	11,2	si
21/04/2026	1	1	11,7	4210	1	187,29	190	159	11,7	13,8	no
21/04/2026	1	1	11,7	4194	2	193,81	195	210	11,4	10,5	no
21/04/2026	1	1	11,7	4208	3	207,86	210	209	12,2	12,1	si
21/04/2026	1	1	11,9	4208	4	190,93	195	190	11,3	11,4	no
21/04/2026	1	1	11,9	4190	2	200,25	205	210	11,2	10,7	no
21/04/2026	0	1	11,3	4198	1	192,47	195	198	12	11,7	si

En la tabla 12 se obtienen inicialmente 29 registros con factor de secado dentro de especificaciones sin necesidad de reinicios y 71 registros fuera de especificaciones, que representa un porcentaje 29 de exactitud del modelo. Sin embargo, considerando un reinicio de secado de solamente 10 min que es valor máximo para no afectar el rendimiento del proceso, se obtienen el 51% de datos dentro de especificaciones, en comparación con un 47% obtenido con los valores seleccionados por el supervisor, es decir, hay una mejora del 4% en la selección del factor de secado empleando el modelo.

En cuanto a pérdidas por peso (sobre secado) con el modelo se obtienen 56,7 kg perdidos dentro de las 100 operaciones, en comparación con 21 kg perdidos con los factores seleccionados por los supervisores, mostrando que con este último hay un 38% menos de pérdidas por secado. Por otra parte, con factores obtenidos por medio del modelo sería necesario reiniciar en total 40 min los secadores para llegar a la humedad deseada, en comparación con 278 min reiniciados

con los factores seleccionados por los supervisores, representando un 85% más de retrabajo con este último.

Hay una pequeña mejora con el modelo en la cantidad de factores de secado seleccionados y una mejora significativa en la cantidad de reinicios o retrabajo, sin embargo, hay un aumento en las pérdidas por secado representando pérdidas económicas.

Observando los datos a detalle, ese presenta que 9 de los 16 registros con sobre secado corresponden al secador número 2, que puede deberse a la variabilidad de la tara de este secador en el tiempo en el que tomaron los registros (se estaba presentando más variabilidad de los usual) y a que este valor no siempre se reportó en la base de datos, por lo que no se tomó en cuenta para el ajuste del modelo. Para el secador 4 se encuentran 3 datos con sobre secado, que puede deberse a variaciones que se realizaron en el equipo a principio de 2026, lo que representa que el 70% de los registros de la base de datos para este secador no incluyen estas modificaciones que afectan directamente factor de secado. También se observa que para algunos clientes como el 9 y el 14 (donde se presenta sobre secado) los registros históricos son pocos y tienen más de un año de antigüedad, que es una posible consecuencia de cambios realizados en los equipos o en la materia prima del cliente.

Conclusiones

Por medio del análisis exploratorio y las correlaciones encontradas, se determinaron las variables con mayor influencia dentro del proceso de secado, destacándose el tipo de café y la humedad inicial, indicando que las características del grano y su contenido de humedad influyen de manera significativamente en el comportamiento del proceso de secado y enfriado. El secador también tiene impacto en el factor de secado, mostrando que pequeñas diferencias en los equipos afectan la eficiencia de esta etapa. En cuanto al peso inicial la correlación no es tan fuerte, sin embargo, se incluye dentro de las variables a usar debido a su influencia en la eficiencia del secado llegando a afectar el resultado final de la humedad, esto mismo ocurre con el cliente (presenta una correlación relativamente débil) debido a que cada proveedor tiene unas características fisicoquímicas asociadas que afectan el factor de secado (densidad, porosidad, actividad de agua).

Se implementaron y evaluaron los modelos de aprendizaje automático regresión lineal múltiple (MAE: 16,812 y R2: 0,389), árboles de decisión (MAE: 14,119 y R^2: 0,576), Random Forest con hiperparámetros optimizados (MAE: 12,539 y R^2: 0,667), Gradient Boosting con hiperparámetros optimizados (MAE: 12,364 y R^2: 0,681), support vector regression (SVR) (MAE: 16,725 y R^2: 0,408) y redes neuronales artificiales (MAE: 17,103 y R^2: 0,372), para la determinación del factor de secado. Se encontró que los modelos con mejor precisión y capacidad explicativa fueron Random Forest y Gradient Boosting y se identificó que al incluir más variables dentro del modelo aumenta moderadamente su capacidad predictiva (mejoras del MAE entre 0,563 y 0,926) y que al realizar segmentaciones por clientes o por tipo las mejoras obtenidas no son consistentes en todos los casos debido a la variabilidad y a la limitación en la cantidad de datos (valores de MAE entre 9,412 y 21,698). Es por esto, que finalmente se

selecciona el modelo Gradient Boosting con hiperparámetros optimizados para el cálculo del factor de secado.

Al validar el modelo dentro de planta se evidencia una mejora para la toma de decisiones especialmente al disminuir los reinicios y retrabajos. A pesar de que inicialmente el modelo acierta únicamente el 29% con los factores escogidos, cuando se toma en cuenta el margen de reinicio de 10 min, el acierto aumenta al 51% superando el 47% que se obtiene con los factores seleccionados por los supervisores. Entre los resultados más relevantes con el uso del modelo se encuentra la disminución del tiempo de secado pasando de 278 min (escogido por supervisores) a 40 min empleando el modelo. Sin embargo, se observa un aumento en las pérdidas por sobre secado pasando de 21 kg con el factor seleccionado por los supervisores a 56,7 kg con el modelo, mostrando que es necesario realizar ajustes para disminuir las pérdidas económicas que conlleva el sobre secado.

En conclusión, el modelo empleado se considera una herramienta útil y viable para apoyar la toma de decisiones dentro del proceso de secado, reduciendo el retrabajo; sin embargo, las variaciones y errores del modelo pueden disminuir al actualizar frecuentemente la base de datos, incorporar variables fisicoquímicas adicionales y al realizar actualizaciones cuando se efectúen cambios en los equipos.

Recomendaciones

Para la implementación se recomienda continuar con un modelo dual, es decir, correr el modelo en paralelo con el proceso actual (factor asignado por el supervisor), registrando su factor y el recomendado por la aplicación para evaluar el desempeño del modelo durante por lo menos 200 datos más que los ya analizados, para realizar los ajustes que sean necesarios.

Realizar un análisis de otras variables que no se registran actualmente directamente relacionadas con la materia prima, que pueden influir en la humedad final como la densidad, tamaño, distribución de partícula y actividad de agua. Además de analizar humedad inicial por operación y no por recepción (una recepción incluye más de una operación) para aumentar la exactitud y precisión del modelo, adicionalmente se recomienda hacer un modelo de reentrenamiento cada 6 meses, para asegurar la fiabilidad del modelo e incluir datos más actuales y además para incluir las variaciones realizadas a los equipos (mantenimiento, calibración). Con lo que también se recomienda hacer un monitoreo del desempeño (drift) que evalúe la desviación del modelo, con un umbral definido en caso de que se deba hacer un reentrenamiento fuera de cronograma.

Se recomienda implementar un sistema de recolección de datos automáticos teniendo en cuenta que en el proceso de evaluación del modelo se encontró un 18% de errores de digitación en el número de secador en las últimas 100 recolecciones realizadas que afectan directamente la predicción del factor de secado, por lo pronto, se sugieren realizar revisiones periódicas de los registros para asegurar su confiabilidad.

Referencias Bibliográficas

Abreu, D. J. M. de, Lorenço, M. S., Machado, G. G. L., Silva, J. M., Azevedo, E. C. de, & Carvalho, E. E. N. (2025). Influence of Drying Methods on the Post-Harvest Quality of Coffee: Effects on Physicochemical, Sensory, and Microbiological Composition. *Foods*, *14*. https://doi.org/10.3390/foods14091463

Adamiec, J., Kamiński, W., Markowski, A. S., & Strumiłło, C. (2014). Drying of biotechnological products. In *Handbook of Industrial Drying, Fourth Edition* (pp. 895–916). CRC Press. https://doi.org/10.1201/b17208

Breiman, L. (2001). Random forests. *Machine Learning*, *45*, 5–32. https://doi.org/10.1023/A:1010933404324

Brooker, D. B., Bakker-Arkema, F. W., & Carl W. Hall. (1992). *Drying and Storage Of Grains and Oilseeds*.

Burmester, K., & Eggers, R. (2010). Heat and mass transfer during the coffee drying process. *Journal of Food Engineering*, *99*, 430–436. https://doi.org/10.1016/j.jfoodeng.2009.12.021

Chindapan, N., Puangngoen, C., & Devahastin, S. (2025). Caffeine removal and compositions losses from whole Robusta coffee beans during conventional and ultrasound-assisted aqueous decaffeination. *Journal of Food Engineering*, *387*. https://doi.org/10.1016/j.jfoodeng.2024.112349

Clarke, & Macrae. (1987). *Coffee Technology*. Springer Netherlands. https://doi.org/10.1007/978-94-009-3417-7

Collazos-Escobar, G. A., Bahamón-Monje, A. F., & Gutiérrez-Guzmán, N. (2025). Dataset and machine learning-based computer-aided tools for modeling working sorption isotherms in

dried parchment and green coffee beans. *Data in Brief*, *61*. https://doi.org/10.1016/j.dib.2025.111738

Collazos-Escobar, G. A., Gutiérrez-Guzmán, N., Váquiro, H. A., García-Pérez, J. V., & Cárcel, J. A. (2025). Analysis of Machine Learning Algorithms for the Computer Simulation of Moisture Sorption Isotherms of Coffee Beans. *Food and Bioprocess Technology*, *18*, 5419–5430. https://doi.org/10.1007/s11947-025-03785-x

Correa, P. C., Goneli, A. L. D., Afonso, P. C., Oliveira, G. H. H. de, & Valente, D. S. M. (2009). *Moisture sorption isotherms and isosteric heat of sorption of coffee in different processing levels.* https://doi.org/10.1111/j.1365-2621.2010.02373.x

Đaković, D., Kljajić, M., Milivojević, N., Doder, Đ., & Anđelković, A. S. (2024). Review of Energy-Related Machine Learning Applications in Drying Processes. In *Energies* (Vol. 17). Multidisciplinary Digital Publishing Institute (MDPI). https://doi.org/10.3390/en17010224

Erbay, Z., & Icier, F. (2010). A review of thin layer drying of foods: Theory, modeling, and experimental results. In *Critical Reviews in Food Science and Nutrition* (Vol. 50, pp. 441–464). https://doi.org/10.1080/10408390802437063

Friedman, J. H. (2002). Stochastic gradient boosting. *Computational Statistics and Data Analysis*, *38*, 367–378. https://doi.org/10.1016/S0167-9473(01)00065-2

Hurtado Cortés, V., Orozco Blanco, D. A., Salas Calderón, K. T., Ramón Ossa, A. L., Bustos Vanegas, J. D., & Gutiérrez Guzmán, N. (2024). Mechanical Drying of Coffee: Influence of Operating Parameters on Cup Quality. *Biotecnología En El Sector Agropecuario y Agroindustrial*, *23*. https://doi.org/10.18684/rbsaa.v23.n1.2025.2483

Kulapichitr, F., Borompichaichartkul, C., Suppavorasatit, I., & Cadwallader, K. R. (2019). Impact of drying process on chemical composition and key aroma components of Arabica coffee. *Food Chemistry*, *291*, 49–58. https://doi.org/10.1016/j.foodchem.2019.03.152

Le, T. M., Tran, T. T., Tran, H. M., & Dao, S. V. T. (2025). Application of Machine Learning in Moisture Content Prediction of Coffee Drying Process. In *Artificial Intelligence and Machine Learning for Industry 4.0* (pp. 145–167). Wiley. https://doi.org/10.1002/9781394275076.ch6

Murthy, P. S., & Madhava Naidu, M. (2012). Sustainable management of coffee industry by-products and value addition - A review. In *Resources, Conservation and Recycling* (Vol. 66, pp. 45–58). https://doi.org/10.1016/j.resconrec.2012.06.005

Peñuela-Martínez, A. E., Sanz-Uribe, J. R., & Medina-Rivera, R. D. (2023). Influence of drying air temperature on coffee quality during storage. *Revista Facultad Nacional de Agronomia Medellin*, *76*, 10493–10503. https://doi.org/10.15446/rfnam.v76n3.104115

Pietsch, A. (2017). Decaffeination-Process and Quality. In *The Craft and Science of Coffee* (pp. 225–143). Elsevier Inc. https://doi.org/10.1016/B978-0-12-803520-7.00010-4

Tosta, M. F., Salvio, L. G. A., Corrêa, J. L. G., & Andrade, E. T. de. (2020). Drying kinetics mathematical modeling of coffee (Coffea arabica L.) processed in different ways and with the use of enzymes and yeast. *Research, Society and Development*, *9*, e908974359. https://doi.org/10.33448/rsd-v9i7.4359

Zou, Y., Gaida, M., Franchina, F. A., Stefanuto, P. H., & Focant, J. F. (2022). Distinguishing between Decaffeinated and Regular Coffee by HS-SPME-GC×GC-TOFMS, Chemometrics, and Machine Learning. *Molecules*, *27*. https://doi.org/10.3390/molecules27061806

Apéndices

Apéndice A

Código Para la Selección del Modelo Para la Predicción del Factor de Secado

```
from google.colab import drive
drive.mount('/content/drive')
import pandas as pd
ruta = "/content/drive/MyDrive/Proyecto de grado/BD/secado_cafe_filtrado.xlsx"

df1 = pd.read_excel(ruta)

df1.head()

"""## Pretratamiento de datos"""

df1["Cliente"] = pd.to_numeric(df1["Cliente"], errors='coerce')
df1["HumedadInicial"] = pd.to_numeric(df1["HumedadInicial"], errors='coerce')
df1["Peso"] = pd.to_numeric(df1["Peso"], errors='coerce')
df1["FactorSecado"] = pd.to_numeric(df1["FactorSecado"], errors='coerce')
df1["HumedadLab"] = pd.to_numeric(df1["HumedadLab"], errors='coerce')

len(df1)

# Se filtran los datos para que sólo los que se encuentran dentro de rango sean tenidos en cuenta

df1["ErrorHumedad"] = df1["HumedadLab"] - df1["HumedadInicial"]
df1["DentroRango"] = abs(df1["ErrorHumedad"]) <= 0.3

df = df1[df1["DentroRango"] == True]

len(df)

datos_iniciales2 = len(df)

import pandas as pd
import numpy as np

from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split

# Se seleccionan las variables a analizar

df[[
"Cliente",
```

```
"TipoCafe",
"HumedadInicial",
"Peso",
"Secador",
"FactorSecado"
]].describe()

# Eliminar duplicados
df = df.drop_duplicates()

# Valores faltantes
df.isnull().sum()

# Se separa la variable objetivo

X = df[[
"Cliente",
"TipoCafe",
"HumedadInicial",
"Peso",
"Secador"
]]

y = df["FactorSecado"]

# Estandarizar variables

scaler = StandardScaler()

X_scaled = scaler.fit_transform(X)

X = pd.DataFrame(X_scaled, columns=X.columns)

# Dividir datos de entrenamaiento y prueba

X_train, X_test, y_train, y_test = train_test_split(
X,
y,
test_size=0.2,
random_state=42
)

print("Datos entrenamiento:", len(X_train))
print("Datos prueba:", len(X_test))
print("Total datos usados:", len(X_train) + len(X_test))
```

```
# Correlación entre variables

variables = [
"Cliente",
"TipoCafe",
"HumedadInicial",
"Peso",
"Secador",
"FactorSecado"
]

df[variables].corr()

import seaborn as sns
import matplotlib.pyplot as plt

plt.figure(figsize=(10,6))

sns.heatmap(
    df[variables].corr(),
    annot=True,
    cmap="coolwarm",
    fmt=".2f"
)

plt.title("Matriz de correlación de variables del proceso de secado")

plt.show()

# Outliers

import seaborn as sns
import matplotlib.pyplot as plt

sns.boxplot(x=df["FactorSecado"])

plt.show()

import seaborn as sns
import matplotlib.pyplot as plt

variables_num = [
"HumedadInicial",
"Peso",
"Secador",
"FactorSecado"
```

```
]

for var in variables_num:

    plt.figure(figsize=(6,4))

    sns.boxplot(x=df[var])

    plt.title(f"Boxplot de {var}")

    plt.show()

datos_iniciales = len(df)

variables_outliers = [
"FactorSecado"
]

for col in variables_outliers:

    Q1 = df[col].quantile(0.25)
    Q3 = df[col].quantile(0.75)

    IQR = Q3 - Q1

    limite_inferior = Q1 - 1.5 * IQR
    limite_superior = Q3 + 1.5 * IQR

    df = df[(df[col] >= limite_inferior) & (df[col] <= limite_superior)]

print("Datos iniciales:", datos_iniciales)
print("Datos finales:", len(df))

"""# Primera prueba
Solo las variables Humedad Inicial, tipo de café y secador
"""

import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.tree import DecisionTreeRegressor
from sklearn.ensemble import RandomForestRegressor, GradientBoostingRegressor
from sklearn.svm import SVR
from sklearn.neural_network import MLPRegressor
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
```

```
X = df[["HumedadInicial", "TipoCafe", "Secador"]]
y = df["FactorSecado"]

X_train, X_test, y_train, y_test = train_test_split(
    X, y,
    test_size=0.2,
    random_state=42
)

modelos = {
    "RegresionLineal": LinearRegression(),
    "ArbolDecision": DecisionTreeRegressor(random_state=42),
    "RandomForest": RandomForestRegressor(random_state=42),
    "GradientBoosting": GradientBoostingRegressor(random_state=42),
    "SVR": SVR(),
    "RedNeuronal": MLPRegressor(max_iter=2000, random_state=42)
}

# Entrenamiento y recolección de métricas
resultados_lista = []

for nombre, modelo in modelos.items():
    modelo.fit(X_train, y_train)

    pred = modelo.predict(X_test)

    mae = mean_absolute_error(y_test, pred)
    rmse = np.sqrt(mean_squared_error(y_test, pred)) # Raíz del error cuadrático medio
    r2 = r2_score(y_test, pred)                      # R cuadrado

    resultados_lista.append({
        "Modelo": nombre,
        "MAE": round(mae, 4),
        "RMSE": round(rmse, 4),
        "R2": round(r2, 4)
    })

df_resultados = pd.DataFrame(resultados_lista)

# Se ordena por MAE de menor (mejor) a mayor (peor)
df_resultados = df_resultados.sort_values(by="MAE", ascending=True).reset_index(drop=True)

print("="*45)
print(" COMPARATIVA DE MODELOS - FACTOR DE SECADO")
print("="*45)
```

```
print(df_resultados.to_string())

"""# Segunda prueba
Con todas las variables
"""

X = df[["HumedadInicial", "TipoCafe", "Secador", "Cliente", "Peso"]]
y = df["FactorSecado"]

X_train, X_test, y_train, y_test = train_test_split(
    X, y,
    test_size=0.2,
    random_state=42
)

modelos = {
    "RegresionLineal": LinearRegression(),
    "ArbolDecision": DecisionTreeRegressor(random_state=42),
    "RandomForest": RandomForestRegressor(random_state=42),
    "GradientBoosting": GradientBoostingRegressor(random_state=42),
    "SVR": SVR(),
    "RedNeuronal": MLPRegressor(max_iter=2000, random_state=42)
}

resultados_lista = []

for nombre, modelo in modelos.items():
    # Entrenamiento y predicción
    modelo.fit(X_train, y_train)

    pred = modelo.predict(X_test)

    mae = mean_absolute_error(y_test, pred)
    rmse = np.sqrt(mean_squared_error(y_test, pred)) # Raíz del error cuadrático medio
    r2 = r2_score(y_test, pred)                      # R cuadrado

    # Guardar en la lista
    resultados_lista.append({
        "Modelo": nombre,
        "MAE": round(mae, 4),
        "RMSE": round(rmse, 4),
        "R2": round(r2, 4)
    })

df_resultados = pd.DataFrame(resultados_lista)
```

```
# Se ordena por MAE de menor (mejor) a mayor (peor)
df_resultados = df_resultados.sort_values(by="MAE", ascending=True).reset_index(drop=True)

print("="*45)
print(" COMPARATIVA DE MODELOS - FACTOR DE SECADO")
print("="*45)
print(df_resultados.to_string())

"""# Prueba 3
Con cada uno de los clientes con todos los modelos
"""

clientes = df["Cliente"].unique()

modelos_clientes = {}

resultados = []

for cliente in clientes:
    print(f"\n{'-'*50}")
    print(f" Procesando Cliente: {cliente} ")
    print(f"{'-'*50}")

    # Filtrar cliente
    df_cliente = df[df["Cliente"] == cliente]

    X = df_cliente[[
        "HumedadInicial",
        "TipoCafe",
        "Secador",
        "Cliente",
        "Peso"
    ]]
    y = df_cliente["FactorSecado"]

    # Limpiar NaN
    datos = pd.concat([X, y], axis=1).dropna()
    X = datos[X.columns]
    y = datos[y.name]

    # Solo si hay suficientes datos
    if len(X) > 50:
        X_train, X_test, y_train, y_test = train_test_split(
            X, y,
            test_size=0.2,
            random_state=42
```

```
        )

        modelos = {
            "RegresionLineal": LinearRegression(),
            "ArbolDecision": DecisionTreeRegressor(random_state=42),
            "RandomForest": RandomForestRegressor(random_state=42),
            "GradientBoosting": GradientBoostingRegressor(random_state=42),
            "SVR": SVR(),
            "RedNeuronal": MLPRegressor(max_iter=2000, random_state=42)
        }

        modelos_clientes[cliente] = {}

        for nombre, modelo in modelos.items():
            modelo.fit(X_train, y_train)

            pred = modelo.predict(X_test)

            mae = mean_absolute_error(y_test, pred)
            rmse = np.sqrt(mean_squared_error(y_test, pred))
            r2 = r2_score(y_test, pred)

            print(f"{nombre:20s} | MAE: {mae:.4f} | RMSE: {rmse:.4f} | R2: {r2:.4f}")

            # Se guarda en la lista para la tabla final
            resultados.append({
                "Cliente": cliente,
                "Modelo": nombre,
                "MAE": round(mae, 4),
                "RMSE": round(rmse, 4),
                "R2": round(r2, 4)
            })

            modelos_clientes[cliente][nombre] = modelo

    else:
        print(f"Muy pocos datos (N={len(X)}). Se omitió el entrenamiento.")

# Tabla final
tabla_resultados = pd.DataFrame(resultados)

# Se ordenan los datos: Primero por Cliente (ascendente) y luego por MAE (de mejor a peor)
if not tabla_resultados.empty:
    tabla_resultados = tabla_resultados.sort_values(by=["Cliente", "MAE"], ascending=[True,
True]).reset_index(drop=True)
```

```
    print("\n" + "="*60)
    print(" RESUMEN FINAL DE MODELOS POR CLIENTE (ORDENADO POR MAE)")
    print("="*60)
    print(tabla_resultados.to_string())
else:
    print("\nNo se generaron resultados. Ningún cliente superó el mínimo de 50 registros.")

tabla_resultados.loc[
tabla_resultados.groupby("Cliente")["MAE"].idxmin()
]

"""# Prueba 4
Con cada uno de los tipos de café, con todos los modelos
"""

tipos_cafe = df["TipoCafe"].unique()

modelos_por_tipo = {}
resultados_tipo = []

for tipo in tipos_cafe:
    # Filtrar datos por Tipo de Café
    df_tipo = df[df["TipoCafe"] == tipo]

    X = df_tipo[["HumedadInicial", "Peso", "Secador", "Cliente"]]
    y = df_tipo["FactorSecado"]

    # Limpieza de nulos
    datos = pd.concat([X, y], axis=1).dropna()
    X = datos[X.columns]
    y = datos[y.name]

    # Solo entrenar si hay suficientes registros (N > 50)
    if len(X) > 50:
        print(f">>> Analizando Tipo de Café: {tipo} (Registros: {len(X)})")

        X_train, X_test, y_train, y_test = train_test_split(
            X, y,
            test_size=0.2,
            random_state=42
        )

        modelos = {
            "RegresionLineal": LinearRegression(),
            "ArbolDecision": DecisionTreeRegressor(random_state=42),
            "RandomForest": RandomForestRegressor(random_state=42),
```

```
            "GradientBoosting": GradientBoostingRegressor(random_state=42),
            "SVR": SVR(),
            "RedNeuronal": MLPRegressor(max_iter=2000, random_state=42)
        }

        modelos_por_tipo[tipo] = {}

        for nombre, modelo in modelos.items():
            modelo.fit(X_train, y_train)

            pred = modelo.predict(X_test)

            mae = mean_absolute_error(y_test, pred)
            rmse = np.sqrt(mean_squared_error(y_test, pred))
            r2 = r2_score(y_test, pred)

            # Guardar resultados
            resultados_tipo.append({
                "TipoCafe": tipo,
                "Modelo": nombre,
                "MAE": round(mae, 4),
                "RMSE": round(rmse, 4),
                "R2": round(r2, 4)
            })

            modelos_por_tipo[tipo][nombre] = modelo

    else:
        print(f"Tipo {tipo}: Datos insuficientes para entrenamiento (N={len(X)}).")

# Tabla final
tabla_resultados_tipo = pd.DataFrame(resultados_tipo)

if not tabla_resultados_tipo.empty:
    # Ordenar por Tipo de Café y luego por el menor MAE
    tabla_resultados_tipo = tabla_resultados_tipo.sort_values(
        by=["TipoCafe", "MAE"],
        ascending=[True, True]
    ).reset_index(drop=True)

    print("\n" + "="*75)
    print(" RESUMEN DE RENDIMIENTO POR TIPO DE CAFÉ")
    print("="*75)
    print(tabla_resultados_tipo.to_string(index=False))
else:
    print("No se generaron resultados suficientes.")
```

```
tabla_resultados_tipo.loc[
tabla_resultados_tipo.groupby("TipoCafe")["MAE"].idxmin()
]

"""# Prueba 5
Optimización de parámetros del modelo general de la prueba 2
"""

import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split, RandomizedSearchCV
from sklearn.ensemble import RandomForestRegressor, GradientBoostingRegressor
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

X = df[["HumedadInicial", "TipoCafe", "Secador", "Cliente", "Peso"]]
y = df["FactorSecado"]

X_train, X_test, y_train, y_test = train_test_split(
    X, y,
    test_size=0.2,
    random_state=42
)

# Definición de los Hiperparámetros
# Para Random Forest
param_grid_rf = {
    'n_estimators': [100, 200, 500],
    'max_depth': [10, 20, 30, None],
    'min_samples_split': [2, 5, 10],
    'min_samples_leaf': [1, 2, 4],
    'bootstrap': [True, False]
}

# Para Gradient Boosting
param_grid_gb = {
    'n_estimators': [100, 300, 500],
    'learning_rate': [0.01, 0.05, 0.1],
    'max_depth': [3, 4, 5],
    'subsample': [0.8, 0.9, 1.0],
    'min_samples_split': [2, 5]
}

modelos_base = {
    "RandomForest": (RandomForestRegressor(random_state=42), param_grid_rf),
    "GradientBoosting": (GradientBoostingRegressor(random_state=42), param_grid_gb)
```

```
}

resultados_optimizados = []

print("="*60)
print("INICIANDO OPTIMIZACIÓN DE HIPERPARÁMETROS")
print("="*60)

for nombre, (modelo, malla) in modelos_base.items():
    print(f"⚙️ Optimizando {nombre}...")

    # Se busca la mejor combinación con 10 iteraciones
    random_search = RandomizedSearchCV(
        estimator=modelo,
        param_distributions=malla,
        n_iter=10,
        cv=3,
        scoring='neg_mean_absolute_error',
        random_state=42,
        n_jobs=-1
    )

    random_search.fit(X_train, y_train)

    # Se encuentra el mejor modelo
    mejor_modelo = random_search.best_estimator_
    predicciones = mejor_modelo.predict(X_test)

    # Cálculo de métricas finales
    mae = mean_absolute_error(y_test, predicciones)
    rmse = np.sqrt(mean_squared_error(y_test, predicciones))
    r2 = r2_score(y_test, predicciones)

    # Se guardan los resultados
    resultados_optimizados.append({
        "Modelo": f"{nombre} Optimizado",
        "MAE": round(mae, 4),
        "RMSE": round(rmse, 4),
        "R2": round(r2, 4),
        "Mejores Parámetros": random_search.best_params_
    })

df_final = pd.DataFrame(resultados_optimizados)
print("\n" + "="*60)
print("RESULTADOS FINALES TRAS LA OPTIMIZACIÓN")
print("="*60)
```

```
print(df_final[["Modelo", "MAE", "RMSE", "R2"]].to_string(index=False))

# Se muestran los mejores parámetros
print("\n" + "="*60)
print("MEJORES PARÁMETROS ENCONTRADOS")
print("="*60)
for res in resultados_optimizados:
    print(f"\n{res['Modelo']}:")
    print(res['Mejores Parámetros'])

# Se selecciona el mejor modelo
mejor_resultado = min(resultados_optimizados, key=lambda x: x["MAE"])

mejor_nombre = mejor_resultado["Modelo"]
mejores_parametros = mejor_resultado["Mejores Parámetros"]

from sklearn.ensemble import RandomForestRegressor, GradientBoostingRegressor

if "RandomForest" in mejor_nombre:

    modelo_final = RandomForestRegressor(**mejores_parametros, random_state=42)

else:

    modelo_final = GradientBoostingRegressor(**mejores_parametros, random_state=42)

modelo_final.fit(X, y)

import pickle

pickle.dump(modelo_final, open("modelo_general.pkl","wb"))

print("Modelo guardado sin modificar código original")

columnas = ["HumedadInicial","TipoCafe","Secador","Cliente","Peso"]

pickle.dump(columnas, open("columnas.pkl","wb"))

from google.colab import files
#files.download("modelo_general.pkl")

"""# Prueba 6
Optimización de parámetros del modelo de la prueba 3
"""

param_grid_rf = {
```

```
    'n_estimators': [100, 200, 500],
    'max_depth': [10, 20, None],
    'min_samples_split': [2, 5, 10],
    'bootstrap': [True]
}

param_grid_gb = {
    'n_estimators': [100, 300, 500],
    'learning_rate': [0.01, 0.05, 0.1],
    'max_depth': [3, 4, 5],
    'subsample': [0.8, 0.9]
}

resultados_detallados = []
clientes = df["Cliente"].unique()

print("="*70)
print("OPTIMIZACIÓN INDIVIDUAL POR CLIENTE (RF & GB)")
print("="*70)

for cliente in clientes:
    df_cliente = df[df["Cliente"] == cliente].copy()

    # Solo entrenar si hay suficientes registros (N > 50)
    if len(df_cliente) < 50:
        continue

    print(f"Analizando Cliente {cliente}...")

    X_c = df_cliente[["HumedadInicial", "TipoCafe", "Secador", "Peso"]]
    y_c = df_cliente["FactorSecado"]

    X_train, X_test, y_train, y_test = train_test_split(X_c, y_c, test_size=0.2, random_state=42)

    modelos = {
        "RandomForest": (RandomForestRegressor(random_state=42), param_grid_rf),
        "GradientBoosting": (GradientBoostingRegressor(random_state=42), param_grid_gb)
    }

    for nombre, (mod, malla) in modelos.items():
        search = RandomizedSearchCV(
            estimator=mod,
            param_distributions=malla,
            n_iter=10,
            cv=3,
            scoring='neg_mean_absolute_error',
```

```
            random_state=42,
            n_jobs=-1
        )

        search.fit(X_train, y_train)

        # Se encuentra el mejor modelo
        mejor_mod = search.best_estimator_
        preds = mejor_mod.predict(X_test)

        resultados_detallados.append({
            "Cliente": cliente,
            "Modelo": nombre,
            "MAE": mean_absolute_error(y_test, preds),
            "RMSE": np.sqrt(mean_squared_error(y_test, preds)),
            "R2": r2_score(y_test, preds),
            "Parametros": search.best_params_
        })

# Tabla con métricas
df_res = pd.DataFrame(resultados_detallados)
df_tabla = df_res.copy()

for col in ["MAE", "RMSE", "R2"]:
    df_tabla[col] = df_tabla[col].map(lambda x: "{:,.3f}".format(x).replace(".", ","))

print("\n" + "="*70)
print("TABLA DE MÉTRICAS OPTIMIZADAS POR CLIENTE")
print("="*70)
print(df_tabla[["Cliente", "Modelo", "MAE", "RMSE", "R2"]].to_string(index=False))

# Se imprimen los mejores parámetros por cliente
print("\n" + "="*70)
print("DESGLOSE DE MEJORES PARÁMETROS POR CLIENTE")
print("="*70)

for res in resultados_detallados:
    print(f"\n CLIENTE: {res['Cliente']} | ALGORITMO: {res['Modelo']}")
    print(f"   Configuración: {res['Parametros']}")
    print("-" * 50)

"""# Prueba 7
Optimización de parámetros del modelo de la prueba 4
"""

param_grid_rf = {
```

```
    'n_estimators': [100, 200, 500],
    'max_depth': [10, 20, None],
    'min_samples_split': [2, 5, 10],
    'bootstrap': [True]
}

param_grid_gb = {
    'n_estimators': [100, 300, 500],
    'learning_rate': [0.01, 0.05, 0.1],
    'max_depth': [3, 4, 5],
    'subsample': [0.8, 0.9]
}

resultados_tipos_opt = []
tipos_cafe = df["TipoCafe"].unique()

print("="*75)
print(" OPTIMIZACIÓN INDIVIDUAL POR TIPO DE CAFÉ (RF & GB)")
print("="*75)

modelos_por_tipo = {}

for tipo in tipos_cafe:
    # Se filtran datos por cada Tipo de Café
    df_tipo = df[df["TipoCafe"] == tipo].copy()

    # Solo entrenar si hay suficientes registros (N > 50)
    if len(df_tipo) < 50:
        print(f" ⚠ Tipo {tipo}: Datos insuficientes ({len(df_tipo)}), omitiendo...")
        continue

    print(f" Analizando Tipo de Café: {tipo}...")

    X_t = df_tipo[["HumedadInicial", "Secador", "Cliente", "Peso"]]
    y_t = df_tipo["FactorSecado"]

    X_train, X_test, y_train, y_test = train_test_split(X_t, y_t, test_size=0.2, random_state=42)

    modelos_competencia = {
        "RandomForest": (RandomForestRegressor(random_state=42), param_grid_rf),
        "GradientBoosting": (GradientBoostingRegressor(random_state=42), param_grid_gb)
    }

    for nombre, (mod, malla) in modelos_competencia.items():

        search = RandomizedSearchCV(
```

```
        estimator=mod,
        param_distributions=malla,
        n_iter=10,
        cv=3,
        scoring='neg_mean_absolute_error',
        random_state=42,
        n_jobs=-1
    )

    search.fit(X_train, y_train)

    # Se evalúa el mejor modelo para el tipo de café
    mejor_mod = search.best_estimator_
    preds = mejor_mod.predict(X_test)

    resultados_tipos_opt.append({
        "Tipo_Cafe": tipo,
        "Modelo": nombre,
        "MAE": mean_absolute_error(y_test, preds),
        "RMSE": np.sqrt(mean_squared_error(y_test, preds)),
        "R2": r2_score(y_test, preds),
        "Parametros": search.best_params_
    })

df_temp = pd.DataFrame([
    r for r in resultados_tipos_opt
    if r["Tipo_Cafe"] == tipo
])

if len(df_temp) > 0:

    mejor_fila = df_temp.loc[df_temp["MAE"].idxmin()]

    mejor_modelo_nombre = mejor_fila["Modelo"]
    mejores_parametros = mejor_fila["Parametros"]

    print(f" Mejor modelo para Tipo {tipo}: {mejor_modelo_nombre}")


    if mejor_modelo_nombre == "RandomForest":
        modelo_final = RandomForestRegressor(
            **mejores_parametros,
            random_state=42
        )
    else:
        modelo_final = GradientBoostingRegressor(
```

```
            **mejores_parametros,
            random_state=42
        )

        modelo_final.fit(X_t, y_t)

        # Guardar modelo
        modelos_por_tipo[tipo] = modelo_final

    else:
        print(f" No hay resultados para Tipo {tipo}")

# Tabla de resultados
df_res_tipos = pd.DataFrame(resultados_tipos_opt)
df_tabla_tipos = df_res_tipos.copy()

# Cambiar puntos por comas y redondear a 3 decimales
for col in ["MAE", "RMSE", "R2"]:
    df_tabla_tipos[col] = df_tabla_tipos[col].map(lambda x: "{:,.3f}".format(x).replace(".", ","))

print("\n" + "="*75)
print(" TABLA DE MÉTRICAS OPTIMIZADAS POR TIPO DE CAFÉ")
print("="*75)
print(df_tabla_tipos[["Tipo_Cafe", "Modelo", "MAE", "RMSE", "R2"]].to_string(index=False))

# Se imprimen los mejores parámetros
print("\n" + "="*75)
print(" DESGLOSE DE PARÁMETROS GANADORES POR TIPO")
print("="*75)

for res in resultados_tipos_opt:
    print(f"\n TIPO DE CAFÉ: {res['Tipo_Cafe']} | MODELO: {res['Modelo']}")
    print(f"   Configuración: {res['Parametros']}")
    print("-" * 60)

import pickle

for tipo, modelo in modelos_por_tipo.items():
    pickle.dump(modelo, open(f"modelo_tipo_{tipo}.pkl","wb"))

from google.colab import files

# for tipo in modelos_por_tipo.keys():
#    files.download(f"modelo_tipo_{tipo}.pkl")
```

Apéndice B

Código Para la Aplicación de Predicción del Factor de Secado

```
import streamlit as st
import pandas as pd
import pickle

st.title(" Predicción del Factor de Secado")

st.markdown("Ingrese las condiciones del lote para estimar el factor de secado")

# ----------------------------
#  Inputs del usuario
# ----------------------------

tipo = st.number_input("Tipo de café (ID)", min_value=1, step=1)
humedad = st.number_input("Humedad Inicial (%)", value=11.5)
secador = st.number_input("Secador (1 - 4)", min_value=1, max_value=4, step=1)
cliente = st.number_input("Cliente (ID)", min_value=0, step=1)
peso = st.number_input("Peso del lote (kg)", value=4000)

# ----------------------------
#  Botón de predicción
# ----------------------------

if st.button("Predecir Factor de Secado"):

    # Cargar modelo
    modelo = pickle.load(open("modelo_general.pkl", "rb"))

    # Crear dataframe
    nuevo = pd.DataFrame({
        "HumedadInicial":[humedad],
        "TipoCafe":[tipo],
        "Secador":[secador],
        "Cliente":[cliente],
        "Peso":[peso]
    })

    # Predicción
    pred = modelo.predict(nuevo)[0]

    # ----------------------------
    #  ◇  Resultado
    # ----------------------------
```

```
    st.subheader("Resultado")

    st.write(f" **Factor de Secado estimado:** {round(pred,2)}")

    # ----------------------------
    #  ◇  Interpretación
    # ----------------------------

    st.markdown("###  Interpretación")

    st.write(
        "El **factor de secado** representa el ajuste necesario en el proceso "
        "para compensar la pérdida de humedad del café durante las etapas de secado y
enfriamiento. "
        "Este valor permite definir condiciones operativas que aseguren que el café alcance la
humedad final deseada."
    )

    # ----------------------------
    #  ◇  Nota técnica
    # ----------------------------

    st.markdown("###  Nota técnica")

    st.write(
        "Este valor es una estimación basada en un modelo de Machine Learning entrenado con
datos históricos. "
        "Debe ser utilizado como apoyo a la toma de decisiones y no como sustituto del criterio
operativo del proceso."
    )
```

Apéndice C

Instructivo de Uso del Aplicativo de Predicción del Factor de Secado

Instructivo de uso del aplicativo de predicción del factor de secado con Streamlit

Objetivo

Estimar el factor de secado con base en las variables operativas, como apoyo a la toma de decisiones dentro del proceso productivo y buscando la reducción de la variabilidad en el proceso de secado.

Requisitos del sistema

- Contar con Phyton instalado
- Tener las librerías de streamlit, pandas y scikit-learn instaladas
- Contar con los archivos de la aplicación (app.py) y el modelo (modelo_general.pkl) dentro de la misma carpeta
- Computador con Windows 10 o superior

Uso de la aplicación

- Abril la terminal, ya sea usando CMD o PowerShell
- Establecer la ruta de la carpeta donde se encuentran los archivos
- Ejecutar el comando streamlit run app.py
- Se abre automáticamente el navegador y se visualiza el aplicativo de la siguiente manera

- Se ingresan los datos teniendo en cuenta lo siguiente:
- Tipo de café: Corresponde a la categoría del café y es un valor numérico entero entre 1 y 6 (referirse a la tabla física donde se muestran las equivalencias).
- Humedad inicial (%): Corresponde al porcentaje de humedad inicial que coincide con la humedad final buscada.
- Secador: Número del secador, corresponde a un valor numérico entero entre 1 y 4.
- Cliente: Número de identificador del cliente (referirse a la tabla física donde se muestran las equivalencias entre nombre del cliente y el ID).
- Peso (kg): Peso del lote de café.
- Hacer clic en el botón “predecir el factor”.
- Se muestra el factor de secado estimado, de la siguiente manera

Recomendaciones para el uso del aplicativo

- El modelo para predecir el factor de secado es un apoyo para la toma de decisiones, no se debe usar como única decisión.
- Verificar que los valores ingresados sean correctos.
- Comprar el resultado con la experiencia operativa del supervisor.