Estimación del peso vivo de cerdos mediante modelos de machine learning

Nathalia Monedero Rodriguez

Asesor

Sebastián Vélez Jaramillo

Universidad Nacional Abierta y a Distancia UNAD

Escuela de Ciencias Básicas, Tecnología e Ingeniería ECBTI

Especialización en Ciencia de Datos y Analítica

2025

Dedicatoria

A Dios, por brindarme la salud, la sabiduría y la perseverancia necesarias para cerrar este proceso académico.

A mi madre, hermano y esposo, por su amor incondicional, su apoyo constante y por creer en mí incluso en los momentos más difíciles.

A mis tutores y mentores, por compartir su conocimiento y despertar en mí el interés por la ciencia de datos y el aprendizaje automático, herramientas que hoy aplico con pasión para aportar soluciones innovadoras al sector porcícola.

Y a los productores, técnicos y operarios de granja, cuya labor diaria inspira la búsqueda de tecnologías que mejoren la productividad, el bienestar animal y bienestar de las personas involucradas en el proceso.

Este trabajo es para ustedes, como símbolo del compromiso con la transformación digital del agro especialmente del sector porcícola y el desarrollo de una producción más inteligente, confiable y sostenible.

Agradecimientos

Quiero expresar mi sincero agradecimiento a todas las personas y la UNAD que hicieron posible la realización de este proyecto de especialización.

A los tutores del programa, por ofrecer una formación rigurosa y actualizada, y por fomentar el pensamiento crítico y la curiosidad científica. Su mentoría fue clave para abordar con criterio los desafíos del análisis de datos y el desarrollo de modelos de machine learning aplicados al sector porcícola.

Al equipo asesor del proyecto, por su orientación técnica y sus valiosas recomendaciones, que contribuyeron significativamente a la calidad y profundidad del trabajo.

A la empresa HBM PORCICULTURA, especialmente al Doctor HERNANDO BLANDON MONTES y al equipo técnico de granjas por facilitar el acceso a datos reales, por su disposición y compromiso con la mejora continua en el sector porcino.

A todos los investigadores y profesionales que han impulsado la transformación digital del sector, gracias por abrir el camino para que la ciencia de datos tenga un impacto directo en la productividad, el bienestar animal y la sostenibilidad.

Este proyecto es fruto del trabajo colectivo, del aprendizaje colaborativo y del deseo de aplicar la tecnología al servicio de un campo más inteligente y eficiente.

Resumen

El proyecto tiene como objetivo optimizar la selección de cerdos en la cosecha mediante un modelo predictivo de peso en pie, basado en técnicas de aprendizaje automático. A partir de datos históricos como edad de salida, rendimiento, grasa dorsal y consumo diario por animal se desarrolló un modelo capaz de predecir el peso con alta precisión.

Durante el proceso, se aplicaron técnicas de limpieza de datos, selección de variables y evaluación de distintos algoritmos, entre ellos regresión lineal simple, múltiple, no lineal, random forest y máquinas de soporte vectorial (SVR). El modelo más preciso fue el de random forest, con un 87.34 % de exactitud. Este modelo, integrado con Excel, permite predecir automáticamente el peso en pie al cargar nuevos datos. Su diseño adaptable lo hace útil para distintos tipos de granjas, mejorando la eficiencia operativa y facilitando una toma de decisiones más precisa.

***Palabras clave*:** Aprendizaje automático, cerdo, predicción, economía, matadero, eficiencia, peso.

Abstract

The objective of the project is to optimize the selection of pigs at harvest by means of a predictive live weight model based on machine learning techniques. A model capable of predicting weight with high accuracy was developed based on historical data such as age at slaughter, yield, backfat and daily consumption per animal.

During the process, data cleaning techniques, variable selection and evaluation of different algorithms were applied, including simple linear regression, multiple regression, nonlinear regression, random forest and support vector machines (SVR). The most accurate model was the random forest model, with 87.3410% accuracy. This model, integrated into an Excel tool, allows automatic prediction of standing weight when loading new data. Its adaptable design makes it useful for different types of farms, improving operational efficiency and facilitating more accurate decision making.

***Keywords*:** Machine learning, pig, prediction, economic, slaughterhouse, efficiency, weight.

Tabla de Contenido

Lista de Tablas

Lista de Figuras

Lista de Apéndices

Introducción

En la industria porcina, la estimación precisa del peso en pie de los cerdos es un indicador clave para la toma de decisiones en procesos productivos, comerciales y sanitarios. Una estimación eficiente permite optimizar el momento del beneficio, mejorar la planificación logística y garantizar la calidad del producto final. Tradicionalmente, la labor del peso en pie se ha realizado mediante métodos manuales, los cuales pueden resultar inexactos, invasivos, subjetivos y poco eficientes, especialmente en explotaciones a gran escala. Estas limitaciones han motivado la búsqueda de alternativas tecnológicas que permitan una predicción más precisa, automatizada y no invasiva del peso animal.

En este orden, el aprendizaje automático (machine learning) se ha consolidado como una herramienta poderosa para el análisis y la modelación de datos complejos en el sector agropecuario. Mediante el uso de algoritmos capaces de identificar patrones no lineales entre múltiples variables, es posible construir modelos predictivos que superen la precisión de los métodos tradicionales. Variables como la edad del animal, rendimiento, grasa dorsal, ganancia por animal por día, consumo por animal y conversión alimenticia, representan indicadores relevantes del desarrollo corporal y, por tanto, del peso en pie, constituyéndose en atributos ideales para alimentar modelos de predicción automatizada.

El presente proyecto de grado tiene como objetivo validar modelos de aprendizaje automático para la predicción del peso en pie de cerdos utilizando las variables mencionadas, con el propósito de contribuir a una gestión más eficiente, precisa y tecnológicamente avanzada en las granjas porcícolas. A través de este enfoque, se busca no solo mejorar la exactitud de las estimaciones, sino también reducir el estrés animal, optimizar recursos y facilitar la toma de

decisiones en tiempo real, promoviendo así una producción porcina más sostenible y competitiva.

Descripción del Problema

Planteamiento del Problema

En el sector Porcicola, la selección de cerdos en el proceso de cosecha es una etapa fundamental que está en el furor de su implementación, la cual afecta directamente los costos y la calidad de la carne. Sin embargo, este proceso enfrenta desafíos, como la variabilidad en el peso de los cerdos, problemas de salud no detectados y manejo ineficiente, lo que puede generar pérdidas económicas significativas y comprometer el bienestar animal (Kim et al., 2016) Actualmente, los métodos tradicionales de evaluación son subjetivos y dependen de la experiencia de las personas, lo que aumenta la probabilidad de errores y reduce la consistencia en la toma de decisiones (Xie et al., 2024)

Históricamente, los métodos utilizados para la evaluación de los cerdos en la cosecha se han basado en inspecciones visuales y mediciones manuales. Estas prácticas dependen en gran medida de la experiencia de las personas, lo que introduce subjetividad y variabilidad en las decisiones. Estudios como los de (Kim et al., 2016) han evidenciado que este enfoque tradicional puede traer errores significativos, comprometiendo tanto la uniformidad de los lotes de carne como la productividad global del sistema.

El desarrollo de tecnologías como la visión artificial (CVS) y la inteligencia artificial (IA) ha comenzado a transformar la gestión en granjas, proporcionando herramientas para una evaluación más precisa y objetiva. Investigaciones recientes han explorado el uso de imágenes en 2D y 3D para predecir el peso vivo (PV) de los cerdos (He et al., 2021) han señalado la oportunidad de integrar modelos predictivos en el manejo rutinario para mejorar la toma de decisiones.

Aunque en el campo se han evidenciado avances, todavía existen importantes

limitaciones que restringen la adopción generalizada de estas tecnologías en las granjas.

Variabilidad en las condiciones ambientales: La precisión de los modelos basados en visión artificial puede verse afectada por factores como la iluminación o la postura del animal (Bhoj et al., 2022)

Falta de validación integral: Muchos estudios se enfocan en aspectos técnicos específicos, como la predicción del peso, sin integrar variables críticas como la salud y el rendimiento general (da Cunha et al., 2024)

Escalabilidad y practicidad: Los métodos propuestos a menudo no han sido probados en escenarios operativos reales, lo que limita su aplicabilidad práctica (Ruckli et al., 2022)

La falta de herramientas eficientes y precisas para la selección de cerdos en la cosecha tiene consecuencias significativas, entre ellas se tienen las económicas, ya que las decisiones incorrectas pueden aumentar los costos de alimentación, reducir la calidad de los lotes procesados y ocasionar pérdidas económicas debido a penalizaciones en el mercado (Čandek-Potokar et al., 2024)

Otro aspecto es el Bienestar animal, ya que un manejo inadecuado en los cerdos puede incrementar el estrés y las lesiones, afectando negativamente tanto su bienestar como la percepción del consumidor (Huanca-Marca et al., 2025)

La ineficiencia en el uso de recursos limita la capacidad del sector porcicola para adaptarse a las crecientes demandas de sostenibilidad y calidad (Turner et al., 2010)

Frente a estos desafíos, se plantea la necesidad de validar un modelo predictivo que permita identificar de manera precisa los cerdos más aptos para la cosecha, considerando variables críticas como peso, salud y rendimiento. ¿Cómo puede un enfoque basado en

inteligencia artificial contribuir a mejorar la precisión y eficiencia de este proceso, minimizando el desperdicio de recursos y maximizando la productividad en granjas porcinas?

Justificación

La exploración propuesta se justifica por su relevancia científica, tecnológica, económica y social, dado que aborda problemáticas centrales del sector Porcicola a través de la integración de técnicas avanzadas de visión artificial (CVS) e inteligencia artificial (IA). A continuación, se organiza la justificación en torno a los vacíos identificados en la literatura y las limitaciones prácticas actuales.

La falta de enfoques integrales que combinen el análisis de peso, salud y rendimiento sigue siendo una barrera para la adopción masiva de tecnologías predictivas en el sector Porcicola. Esta investigación contribuirá al avance del conocimiento al integrar múltiples variables críticas: Si bien estudios como el de (Bhoj et al., 2022) se han enfocado en estrategias de procesamiento de imágenes para estimar peso vivo, la incorporación simultánea de factores como la salud y el rendimiento sigue siendo limitada. La propuesta busca llenar este vacío mediante un enfoque holístico que mejore la precisión y utilidad práctica de los modelos predictivos.

Validación bajo condiciones reales, (da Cunha et al., 2024) han destacado la necesidad de validar modelos de predicción en escenarios operativos reales, donde las condiciones ambientales y la variabilidad de los animales presentan desafíos significativos. Esta investigación abordará esta limitación mediante pruebas en entornos de granja, garantizando la aplicabilidad del modelo.

Innovación en técnicas predictivas**,** los trabajos de (He et al., 2021) y (Xie et al., 2024) han demostrado que las redes de regresión y modelos de aprendizaje profundo son eficaces para la predicción de peso, pero no han explorado suficientemente la integración de algoritmos avanzados de corrección de postura y ruido. Este estudio tratara de incluir técnicas para

maximizar la precisión en diferentes condiciones.

La implementación de un modelo predictivo que utilice visión e inteligencia artificial representa un avance significativo para el sector. Este modelo, aumentará la objetividad y precisión: Frente a la subjetividad inherente de las evaluaciones manuales (Brossard et al., 2009), este modelo reducirá la dependencia de la experiencia humana al automatizar la selección de animales para la cosecha.

Optimizará los recursos, (Turner et al., 2010) señalaron la importancia de reducir el uso ineficiente de recursos mediante la optimización del manejo animal. Esta investigación permitirá identificar a los cerdos más aptos para la cosecha, minimizando el desperdicio de alimentos y tiempo.

El sector Porcicola enfrenta constantes presiones para reducir costos y garantizar la calidad de la carne. Este proyecto tiene el potencial de reducir pérdidas económicas, según (Ruckli et al., 2022), la ineficiencia en la selección de animales genera pérdidas significativas. Este modelo predictivo disminuirá estas pérdidas al identificar con mayor precisión a los cerdos listos para el sacrificio.

Mejorar la competitividad, (Čandek-Potokar et al., 2024) resaltaron que la calidad homogénea de las canales es un factor clave para la competitividad. El objetivo de este proyecto es contribuir a la garantía de la homogeneidad mediante una selección más precisa y basada en datos.

El bienestar animal es una prioridad creciente para los consumidores y reguladores. Estudios como el de (Chidgey, 2024) han demostrado que un manejo adecuado reduce el estrés y mejora la calidad de la carne. Este proyecto tendrá un impacto positivo en el bienestar animal al reducir el estrés, minimizando el manejo físico y evitar métodos invasivos.

Fomentar prácticas sostenibles, (Ruckli et al., 2022) subrayaron la necesidad de adoptar herramientas que optimicen los recursos en granjas, contribuyendo a la sostenibilidad del sector.

El consumidor moderno exige productos de alta calidad y producidos bajo estándares éticos y sostenibles. Este proyecto contribuirá a satisfacer estas demandas al mejorar la trazabilidad y calidad, (García-Infante et al., 2024) enfatizaron que el uso de tecnologías avanzadas mejora la trazabilidad y aumenta la confianza del consumidor. Este proyecto facilitará la clasificación de animales según estándares de calidad.

Promover el bienestar animal como un valor de mercado, estudios como el de (Huanca-Marca et al., 2025) destacan que la percepción del bienestar animal está estrechamente vinculada con la aceptación de los productos porcinos.

Esta investigación abordará vacíos significativos en la literatura y en la práctica, proporcionando un modelo predictivo innovador que transformará la selección de cerdos en la cosecha. Al hacerlo, no solo mejorará la eficiencia y sostenibilidad del sector, sino que también garantizará un mayor bienestar animal y una mejor percepción pública del producto porcino

Objetivos

Objetivo General

Validar modelos de aprendizaje automático para la predicción del peso en pie de cerdos utilizando variables como la edad, rendimiento, grasa dorsal y consumo por animal, con el fin de optimizar la gestión en granjas porcícolas.

Objetivos Específicos

Explorar la base de datos para la identificación de patrones y correlaciones entre las variables, utilizando EDA.

Analizar modelos de regresión y aprendizaje automático (Random Forest, SVR, regresión lineal múltiple, regresión lineal simple, regresión no lineal) para la predicción del peso en pie.

Implementar una herramienta práctica para la carga datos desde un archivo Excel, aplicando el modelo validado de predicción del peso en pie y exportando los resultados a un nuevo archivo.

Marco de Referencia

Marco Conceptual y Teórico

La selección de cerdos en el proceso de cosecha implica la identificación de animales que han alcanzado un peso y condición óptimos para el sacrificio, lo cual es crítico para la rentabilidad y sostenibilidad de la industria porcina (Brossard et al., 2009). Tradicionalmente, esta selección se ha basado en métodos manuales y la experiencia del personal, sin embargo, estos métodos presentan limitaciones relacionadas con la subjetividad y la variabilidad individual (Kim et al., 2016). En este contexto, el peso vivo se ha identificado como una de las variables más importantes para determinar la idoneidad de los animales para cosecha (da Cunha et al., 2024).

El concepto de bienestar animal también es fundamental, ya que el manejo adecuado durante la selección no solo mejora la calidad de la carne, sino que también responde a la creciente demanda de consumidores por prácticas éticas (Huanca-Marca et al., 2025). La implementación de tecnologías no invasivas, como la visión artificial (CVS), está transformando este proceso al permitir evaluaciones objetivas y precisas sin causar estrés adicional a los animales (Taylor et al., 2022).

En los últimos años, las tecnologías basadas en inteligencia artificial (IA) y CVS han mostrado un gran potencial en la mejora de la precisión y eficiencia de los procesos en la industria porcina. Estudios recientes han demostrado que el uso de imágenes 2D y 3D, combinado con algoritmos de aprendizaje profundo, permite estimar el peso y la composición corporal de los cerdos con alta precisión (He et al., 2021); (Masoumi et al., 2021). Estas tecnologías son particularmente valiosas para abordar desafíos como la variabilidad en las condiciones ambientales y la postura de los animales durante la medición (Tu & Jørgensen,

2023).

Una técnica destacada es la aplicación de redes neuronales convolucionales (CNN) para analizar imágenes en tiempo real. Por ejemplo, (Wei et al., 2024) desarrollaron un modelo basado en aprendizaje profundo que logra predecir con precisión el contenido graso intramuscular, una métrica clave para la calidad de la carne. Del mismo modo, (Xie et al., 2024) emplearon una variante modificada de Mask R-CNN para estimar el peso vivo, logrando una mejora significativa en comparación con los métodos tradicionales.

La automatización del proceso de selección de cerdos tiene implicaciones económicas y operativas significativas. Según (Ruckli et al., 2022), la integración de herramientas tecnológicas no solo reduce el tiempo y los costos operativos, sino que también mejora la consistencia en la toma de decisiones. Por ejemplo, la implementación de sensores y sistemas de visión por computadora ha permitido identificar patrones de comportamiento y salud que podrían pasar desapercibidos mediante evaluaciones manuales (Drexl et al., 2024).

Por otro lado, la sostenibilidad es otro beneficio importante. (Chidgey, 2024) señala que el uso de tecnologías no invasivas contribuye a minimizar el desperdicio de recursos y el estrés animal, factores que están estrechamente vinculados con la percepción pública y la aceptación de los productos porcinos en el mercado global.

Aunque las tecnologías avanzadas han mostrado resultados prometedores, todavía enfrentan desafíos significativos, como la necesidad de una infraestructura inicial costosa y la capacitación del personal para operar sistemas complejos (García-Infante et al., 2024). Además, el éxito de estas herramientas depende de la calidad de los datos utilizados para entrenar los modelos, lo que puede variar considerablemente entre granjas y regiones (Turner et al., 2010).

Otro desafío destacado es la variabilidad en las condiciones de luz y el comportamiento

animal durante la captura de imágenes, lo que puede introducir errores en las estimaciones (Čandek-Potokar et al., 2024). A pesar de estas limitaciones, los avances en algoritmos de corrección y preprocesamiento de datos, como los desarrollados por (Drexl et al., 2024), están mejorando la robustez y aplicabilidad de estas tecnologías.

En síntesis, el desarrollo y validación de modelos predictivos basados en IA y CVS representa una oportunidad única para transformar la selección de cerdos en la industria porcina. Este enfoque aborda los vacíos de los métodos tradicionales al proporcionar mediciones objetivas, reducir costos y mejorar tanto la productividad como el bienestar animal. La investigación actual se basa en avances recientes en aprendizaje profundo y visión por computadora, sin embargo, aún enfrenta desafíos relacionados con la implementación práctica y la adaptabilidad a diferentes contextos productivo.

Metodología

La metodología elegida para el proyecto es el marco metodológico CRISP-DM (Cross-Industry Standard Process for Data Mining) debido a su flexibilidad, enfoque cíclico y capacidad para adaptarse a proyectos de análisis predictivo y minería de datos. Este enfoque permite estructurar las etapas de trabajo desde la comprensión inicial del problema hasta la implementación final del modelo.

La naturaleza del proyecto, orientada al desarrollo de un modelo predictivo basado en inteligencia artificial para la estimación del peso vivo (PV) o peso en pie, se alinea con la estructura modular de CRISP-DM, maximizando la eficiencia en el manejo de datos e interpretación de resultados.

La metodología CRISP-DM se ajusta bien a este proyecto debido a que se centra en el análisis de datos y la implementación de modelos predictivos en entornos industriales. Este marco proporciona una estructura clara y práctica que permite abordar de manera efectiva los desafíos asociados con la recopilación, preprocesamiento y modelado de datos derivados de captura de datos en campo, es decir en las granjas porcícolas. Su enfoque iterativo facilita la mejora continua del modelo, asegurando que los resultados sean precisos y útiles en un contexto práctico.

Fases

Fase 1 Comprensión del problema: identificar los requerimientos clave y las preguntas que el modelo debe responder.

Actividades:

Análisis del dominio porcino para entender la relevancia del peso vivo (PV) en la selección de cerdos para beneficio.

Revisión de literatura para establecer criterios de medición y estándares actuales en la industria.

Definición de métricas de éxito, como coeficiente de determinación (R^2), error absoluto medio (MAE) y raíz del error cuadrático medio (RMSE).

Fase 2 Comprensión de los datos: explorar y validar los datos disponibles para garantizar su calidad y representatividad.

Actividades:

Revisión de la base de datos de cerdos obtenidas mediante captura de datos en campo.

Exploración inicial para identificar patrones, inconsistencias o valores atípicos.

Clasificación de los datos por categorías relevantes, como peso, edad, rendimiento, grasa dorsal y consumo por animal.

Fase 3 Preparación de los datos: procesar y transformar los datos para que sean aptos para el modelo.

Actividades:

Normalización de variables, como peso vivo, edad, rendimiento, grasa dorsal y consumo por animal, asegurando la consistencia en las estimaciones.

Generación de conjuntos de datos de entrenamiento y prueba en una proporción adecuada (80%-20%).

Normalización de variables para garantizar escalas homogéneas.

Generación de interacciones polinomiales para capturar relaciones no lineales

Fase 4 Modelado: desarrollar modelos predictivos para estimar el peso vivo.

Actividades:

Entrenamiento y optimización de Random Forest mediante búsqueda aleatoria de

hiperparámetros.

Entrenamiento y optimización de SVR, regresión lineal múltiple, regresión lineal simple y regresión no lineal.

Fase 5 Evaluación: validar el modelo en términos de precisión y aplicabilidad.

Actividades:

Prueba del modelo en un conjunto de datos independiente y comparación de resultados con métodos tradicionales.

Evaluación inicial del desempeño del modelo con métricas de rendimiento como RMSE, MAE y R^2.

Fase 6: Implementación: desplegar el modelo en un entorno práctico.

Actividades:

Desarrollo de un método manual para la aplicación del modelo como la creación de un script que permita la carga de nuevos datos a través de un archivo de Excel, el modelo predice el peso en pie y permite la exportación de un nuevo archivo en Excel con los resultados.

Monitoreo continuo del desempeño del modelo para ajustes futuros.

Método

El método utilizado es cuantitativo de tipo predictivo, basado en la aplicación de técnicas estadísticas y de aprendizaje automático para modelar relaciones entre variables y predecir un valor objetivo continuo (peso en pie o peso vivo).

Tipo de Estudio

Es un estudio aplicado de enfoque descriptivo-predictivo. Es aplicado porque busca resolver un problema práctico dentro del contexto productivo porcino. Es descriptivo en tanto se

analizan las variables disponibles en la base de datos, y predictivo porque el objetivo final es generar modelos que permitan estimar el peso en pie a partir de variables observables.

Recolección de Datos

Los datos utilizados en este estudio provienen de registros históricos de un grupo de granjas porcícolas, recopilados a partir de procesos operativos de producción. La base de datos incluye información por cerdo relacionada con: edad, rendimiento, grasa dorsal y consumo por animal.

Estos datos fueron almacenados de forma estructurada en hojas de cálculo de Excel y para garantizar la calidad, se realizó una limpieza y depuración de los datos para eliminar registros incompletos, valores atípicos o inconsistencias.

Resultados

Primer Resultado

Aplicación EDA

Se realizó la carga del dataset, validando un análisis exploratorio, en el cual se incluyó la información general, verificación de valores nulos, distribución tanto de la variable objetivo, obteniendo 4.540 datos.

Figura 1

Resumen Estadístico Descriptivo de Variables Productivas en Cerdos

	Promedio de rendimiento	Promedio de grasa_dorsal	Edad	\
count	4540.000000	4540.000000	4540.000000	
mean	83.892141	15.202643	160.203524	
min	59.160000	11.000000	144.000000	
25%	79.887500	14.000000	155.000000	
50%	84.130000	15.000000	160.000000	
75%	88.280000	17.000000	166.000000	
max	100.000000	20.000000	185.000000	
std	6.564165	2.264683	8.157893	

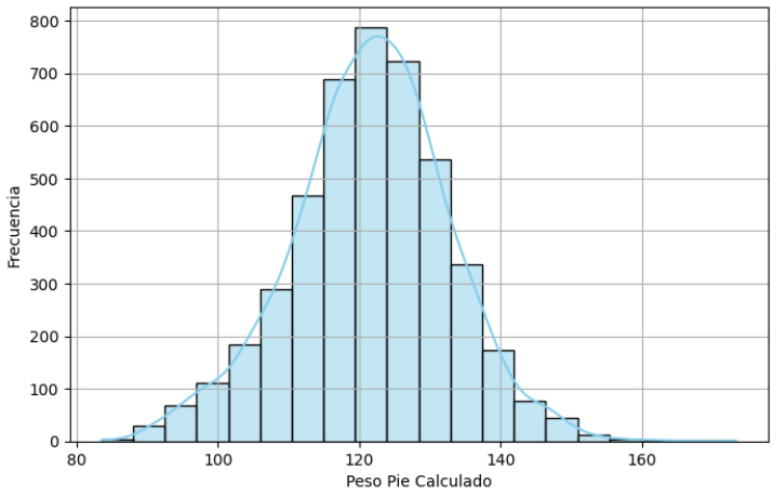
	Ganancia/animal/día	Consumo por animal	CA	% mortalidad
count	4540.000000	4540.000000	4540.000000	4540.000000
mean	1.001468	193.913584	2.238163	0.873500
min	0.946000	141.230000	2.020000	0.000000
25%	0.956000	182.690000	2.230000	0.580000
50%	1.004000	195.670000	2.280000	0.670000
75%	1.029000	202.550000	2.310000	1.110000
max	1.149000	240.250000	2.400000	3.280000
std	0.053437	18.936872	0.114152	0.607587

En la figura 1 se puede apreciar el estadístico de 7 variables, las cuales podrían ser candidatas para el modelo. El rendimiento es un porcentaje de cumplimiento que se asigna a cada cerdo de acuerdo con la tabla de rendimientos asignada por la casa genética, la cual presenta

mediciones diarias de peso de acuerdo con la edad del cerdo y para el respectivo análisis, presentó una media de 83.89% y una desviación de 6.56, la grasa dorsal presenta un rango entre 11 y 20, lo cual puede ser normal por el tamaño y la edad del cerdo. El comportamiento de la edad es muy similar al de la grasa dorsal, presentando un rango entre 144 y 184 días, una media de 160 días y una desviación de 8.15 días. La ganancia/animal/día muestra una variabilidad extremadamente baja de 0.053, lo que quiere decir que los cerdos están bajo condiciones muy homogéneas, este dato puede ser bueno para control zootécnico, sin embargo, podría no aportar demasiada discriminación en el modelo. El consumo por animal presentó variación de 18.93, lo cual es positivo para el modelo con un rango entre 141 y 240 kg. La conversión y la mortalidad no aportan al modelo ya que presentan desviaciones de 0.11 y 0.60. Realizando un análisis más detallado, se encuentra que las variables ganancia/animal/día y CA (Conversión alimenticia), tienen poca variación lo que podría limitar su utilidad predictiva individual por tal motivo no se tendrán en cuenta en las variables predictoras.

Figura 2

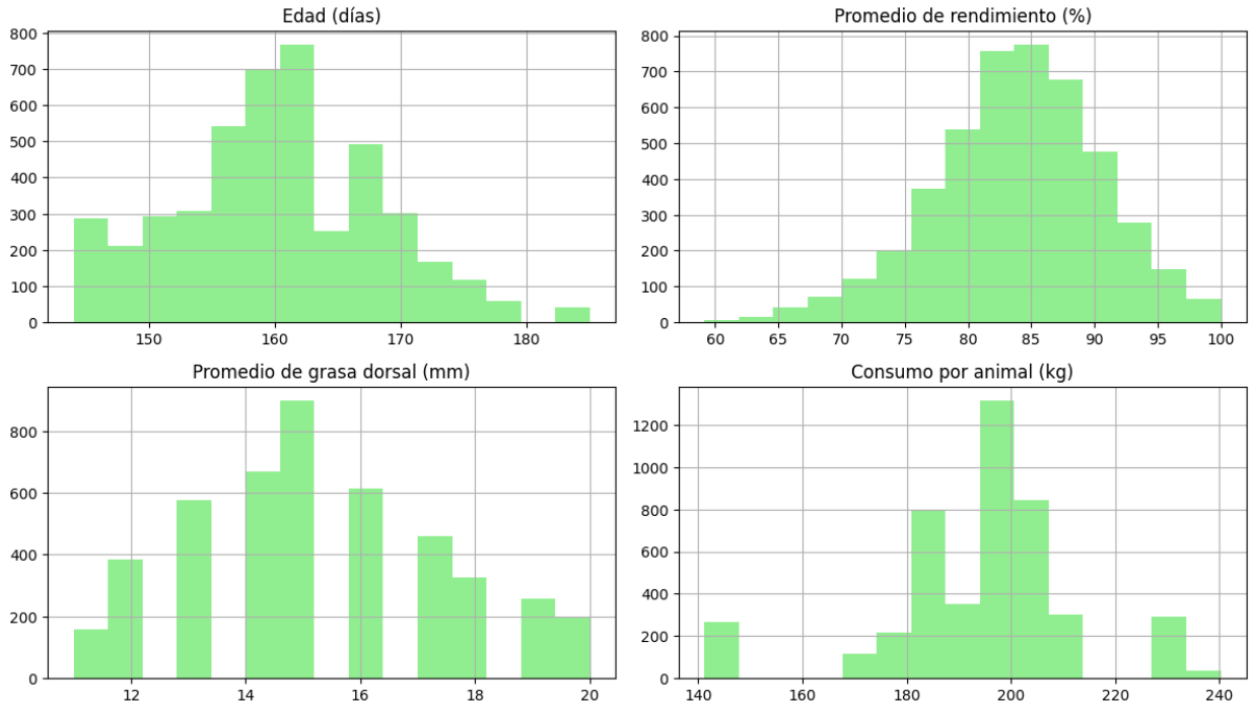
Distribución de la Variable Objetivo



En la figura 2 se puede observar la distribución de la variable Peso en Pie Calculado (peso en pie real de cada cerdo), la cual muestra que la mayoría de los cerdos tienen un peso en pie cercano a 120 kg. La distribución tiene una forma de campana, aunque con una ligera asimetría hacia la derecha, lo cual indica que hay algunos cerdos con pesos más altos. El peso varía entre 80 y 165 kg, este dato demuestra que los datos son consistentes y biológicamente sensatos.

Figura 3

Distribución de las Variables Predictoras



En la figura 3 se pueden apreciar gráficos de distribución para cada variable predictora, en la cual se observa en la edad una distribución levemente sesgada a la derecha, mostrando valores poco frecuentes de 185 días. El rendimiento muestra una distribución normal, evidenciando un rango entre 80% y 90%, con la presencia de pocos valores menores a 70% y

mayores a 95%. La grasa dorsal presenta distribución simétrica a la derecha, con valores poco comunes menores a 11 mm y mayores a 20 mm, lo que puede indicar un comportamiento normal en cerdos.

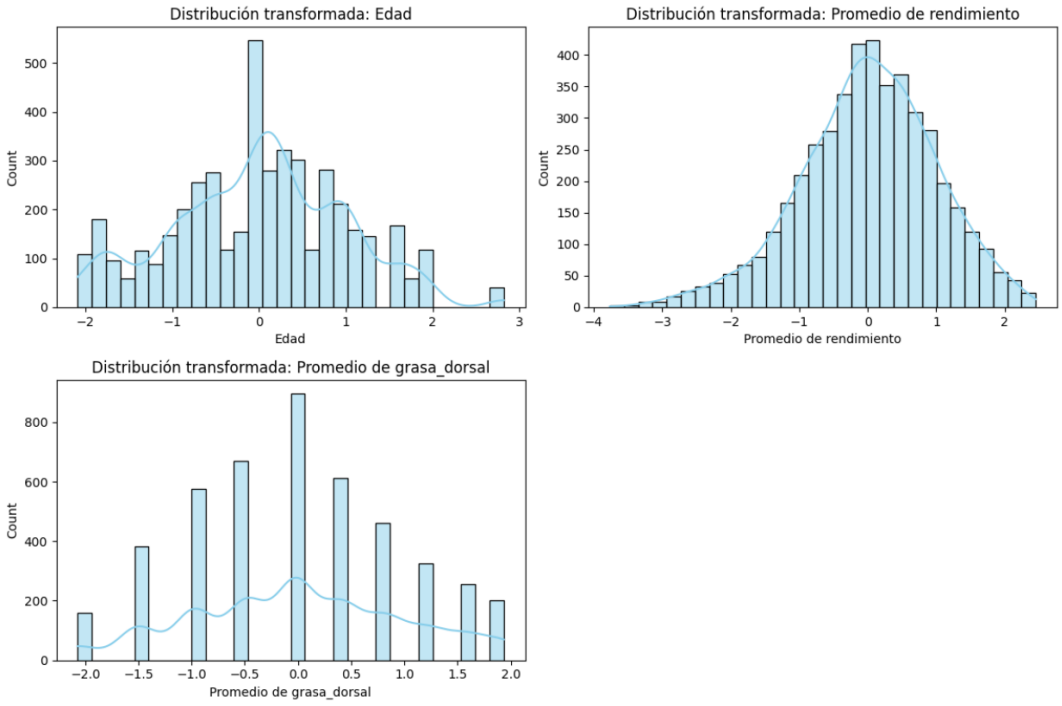
En el consumo muestra que la mayoría de los valores se concentran entre 180 y 210 kg, con un pico cerca de 195–200 kg. La distribución es ligeramente asimétrica a la derecha, este comportamiento es normal en el consumo de los cerdos puesto que el tipo de comedero puede ser diferente para cada lote.

Transformaciones para Normalizar Distribuciones Sesgadas

Con el fin de evitar distribuciones sesgadas, se decide aplicar transformaciones para normalizar, obteniendo el siguiente resultado.

Figura 4

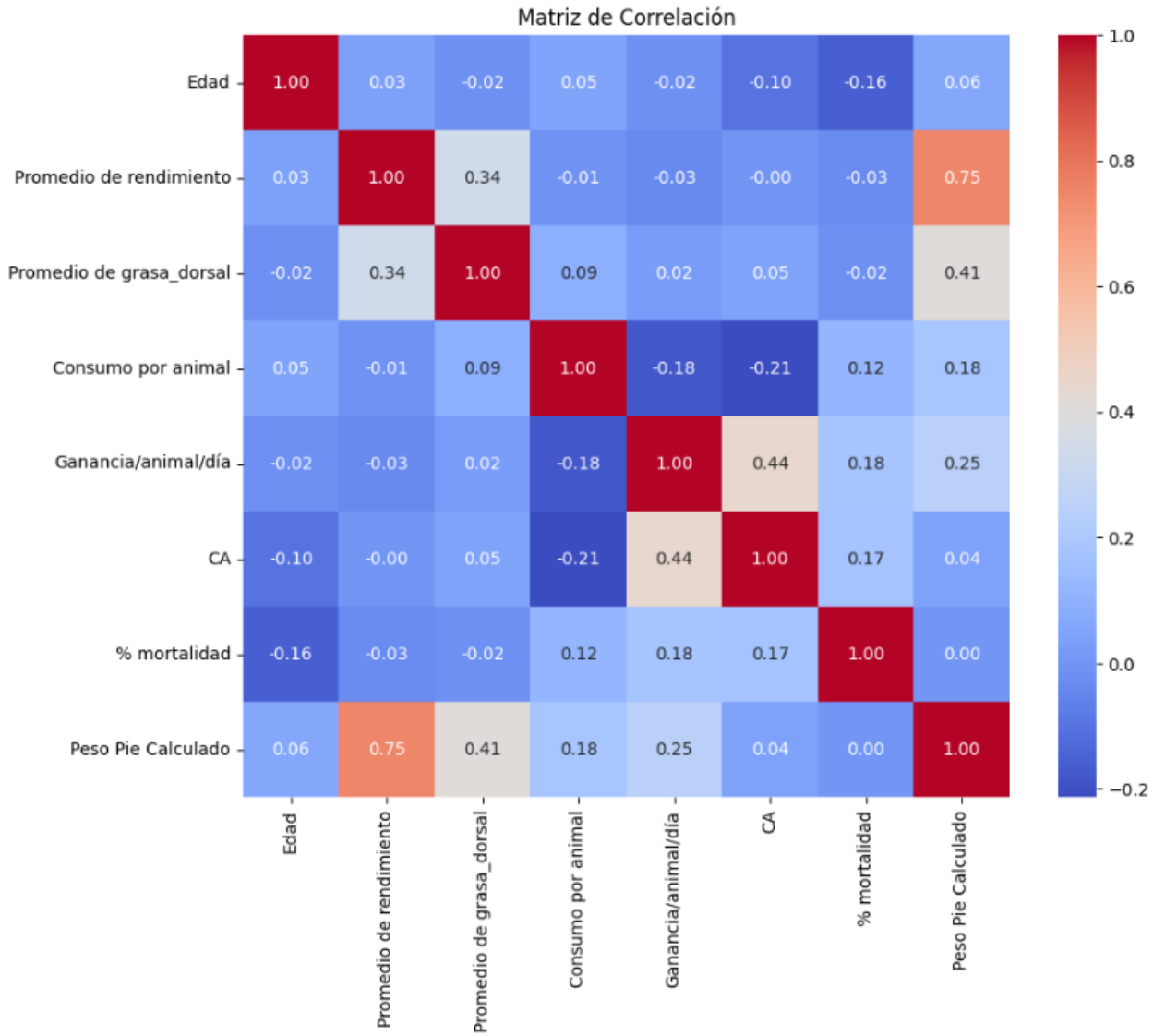
Normalizar Distribuciones Sesgadas



De acuerdo con la figura 4, se evidencia la normalización de las distribuciones para edad, rendimiento y grasa dorsal. Para el rendimiento muestra una distribución transformada notablemente normal y simétrica, para la edad y grasa dorsal, aunque presentan ciertas asimetrías, han sido centradas y escaladas correctamente, lo que mejora significativamente su comportamiento en modelos.

Figura 5

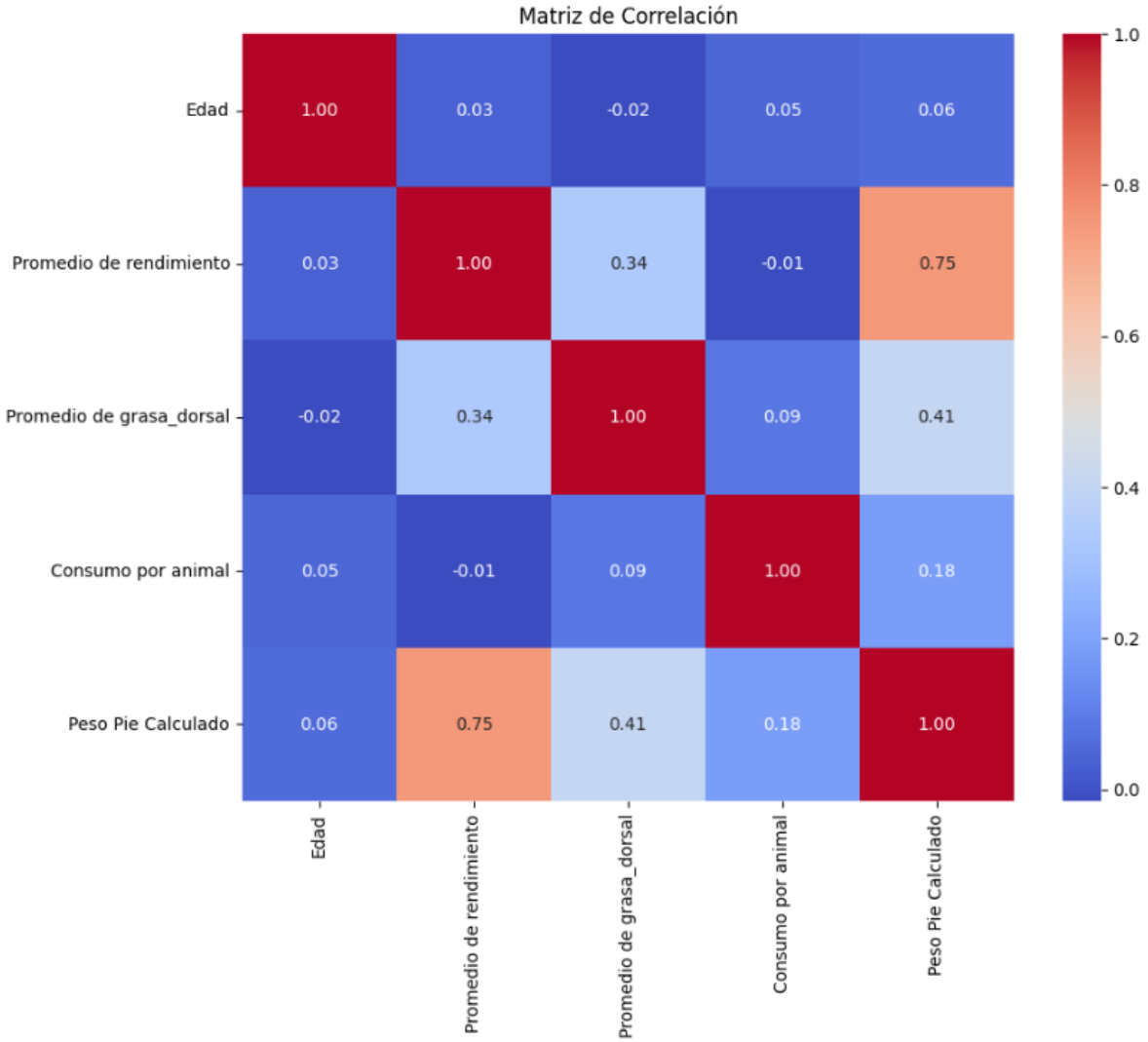
Matriz de Correlación para 7 Variables



En la figura 5 se puede apreciar la matriz de correlación para las 7 variables inicialmente candidatas para el procesamiento del modelo, sin embargo, la CA (conversión alimenticia) y el % mortalidad, tienen una relación muy baja de 0.04 y 0.00 respectivamente, lo que no aportaría al modelo. Para la Ganancia/animal/día aunque tiene una correlación de 0.25, se descarta porque para obtener este valor es necesario conocer el peso final, lo que no es consecuente ya que el peso es la variable objetivo.

Figura 6

Matriz de Correlación para 4 Variables Elegidas



En la figura 6, se muestra que el rendimiento presenta una fuerte correlación positiva (0.75) con el peso en pie, lo que quiere decir que animales con mayor rendimiento tienden a tener pesos mayores. La grasa dorsal tiene una correlación moderada (0.41), lo que quiere decir que, incluirla puede mejorar la precisión del modelo. El consumo por animal presenta una correlación baja (0.18), sin embargo, puede seguir siendo útil como variable complementaria.

Para definir el modelo, se descartaron las variables Ganancia/animal/día y conversión alimenticia.

Segundo Resultado

Se evaluaron 5 modelos con enfoques de regresión, tanto lineales como no lineales. En la Tabla 1 se pueden apreciar los resultados obtenidos para cada uno con su respectivo coeficiente de determinación (R cuadrado), error absoluto medio (MAE), y raíz del error cuadrático medio (RMSE), igualmente la observación sobre el desempeño, mostrando sobreajuste para algunos modelos, como la regresión múltiple y la no lineal, que aunque se obtiene un R cuadrado relativamente alto, los errores son elevados, lo que hace pensar que el modelo aprende bien los datos de entrenamiento, sin embargo, no funciona igual de bien con datos nuevos.

Tabla 1

Comparativos Modelos Aplicados

Modelo	R cuadrado	MAE	RMSE	Observación
Regresión lineal simple	0.0010	8.7427	11.1603	Baja capacidad predictiva.
Regresión lineal múltiple	0.6196	5.0875	6.8864	Buen desempeño, pero con riesgo de sobreajuste; no generaliza bien.

Modelo	R cuadrado	MAE	RMSE	Observación
Regresión no lineal	0.6617	4.7041	6.4949	Ligeramente mejor que la múltiple, pero tiende al sobreajuste.
Random forest	0.8734	2.9679	3.9736	Excelente precisión y buena capacidad predictiva.
SVR	0.7462	3.9597	5.6257	Falla al generalizar; predice valores fijos.

Nota. Métricas de desempeño para diferentes modelos de regresión.

Los resultados obtenidos en la validación cruzada de los modelos aplicados permiten concluir que el modelo Random forest representa la alternativa más sólida y confiable para la predicción del peso en pie de cerdos a partir de las variables aplicadas. A pesar de que modelos como SVR y las regresiones polinómicas y múltiples reportaron coeficientes de determinación (R^2) mayores a 0.6, excepto la regresión lineal simple.

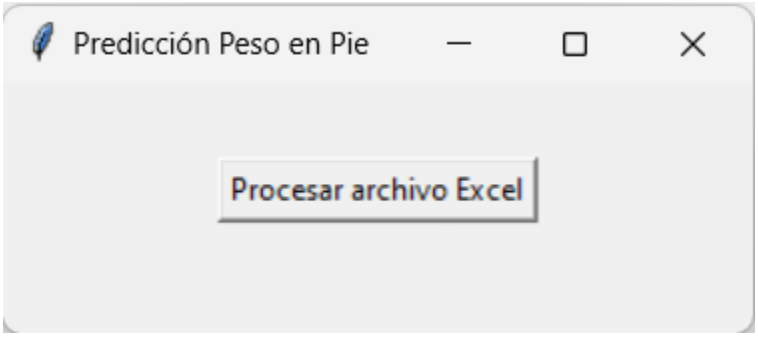
En consecuencia, se recomienda enfáticamente el uso del modelo Random forest en aplicaciones operativas dentro del sector porcino, dado que su desempeño demuestra no solo solidez estadística, sino también viabilidad tecnológica en entornos reales de producción animal.

Tercer Resultado

Para finalizar, se creó una herramienta sencilla que facilita el uso del modelo predictivo por parte de los usuarios, la cual permite aplicar el modelo sin necesidad de conocimientos en programación o análisis de datos.

Figura 7

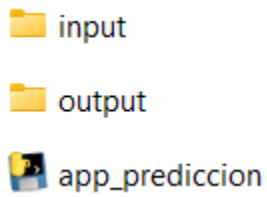
Botón para Generar Archivo con las Nuevas Predicciones



En la figura 7 se puede apreciar el tercer resultado del proyecto, donde básicamente se construyó como herramienta, un botón que le permite al usuario final procesar el modelo sin que tenga contacto con temas técnicos, el cual funciona pegando un archivo con los nuevos datos para predecir el peso en pie en una carpeta específica, para esto se cuenta con una herramienta como Excel, en la cual ya está estructurada la base con los datos que el modelo necesita. Después de procesar los datos este botón exporta un nuevo archivo de Excel en una carpeta de salida con el peso en pie predicho.

Figura 8

Herramientas para Procesar el Modelo



En la figura 8 se puede apreciar la herramienta creada para procesar el modelo, la carpeta de input de entrada, que es donde el usuario va a pegar la base de dato estructurada con datos

nuevos y la carpeta output de salida, es donde se va a alojar el resultado final con el peso en pie predicho.

Figura 9

Salida Después de Procesar el Modelo

	A	B	C	D	E	F
1	tiquete	Promedio de rendimiento	Promedio de grasa_dorsa	Edad	Consumo por animal	Predicción_RF
2	1830260	58,36	9	166	186,94	126,3287273
3	1830258	59,79	11	166	186,94	126,4184242
4	1806386	57,83	8	161	173,14	124,3706667
5	1832897	52,38	9	164	193,66	129,038303
6	1824946	59,58	9	153	218,96	153,5629091
7	1807968	55,99	8	156	184,63	126,5260606
8	1828375	55,35	9	164	186,94	126,3287273
9	1830242	58,51	12	166	186,94	126,4184242
10	1812061	57,31	13	160	184,63	126,5260606
11	1831057	64,62	8	164	218,96	153,5629091
12	1830230	63,57	14	166	186,94	126,4184242

En la figura 9 se puede evidenciar el resultado final del ejercicio con el peso predicho, aplicado con datos nuevos reales de granja.

Conclusiones

Durante la exploración detallada de la base de datos, se identificaron patrones, distribuciones, posibles valores atípicos y relaciones de correlación entre las variables. Este análisis permitió seleccionar las variables más relevantes para la construcción del modelo predictivo.

En la implementación y comparación de diversos enfoques de regresión incluyendo regresión lineal múltiple, regresión no lineal, SVR y Random Forest, el modelo Random Forest demostró el mejor desempeño global. Su capacidad de generalización y precisión en la predicción del peso en pie fue superior, como lo evidencian sus métricas: alto coeficiente de determinación (R^2), bajo error absoluto medio (MAE) y menor raíz del error cuadrático medio (RMSE), manteniendo estos resultados incluso bajo validación cruzada.

Se desarrolló un script funcional que permite al usuario cargar datos desde un archivo Excel, ejecutar manualmente a través de un botón el modelo validado en Python y exportar un nuevo archivo con las predicciones generadas. Aunque no se trata de una herramienta autónoma con interfaz gráfica, esta solución es práctica y aplicable en contextos reales de producción porcina, facilitando la toma de decisiones basada en datos.

Finalmente, se realizó una prueba con datos reales no utilizados en el entrenamiento, en la cual el modelo predijo con alta precisión, lo que confirma su aplicabilidad en escenarios productivos reales, consolidándolo como una herramienta confiable para la gestión en granjas porcícolas.

Recomendaciones

A partir de los resultados obtenidos y el análisis comparativo de los modelos de predicción del peso en pie de cerdos, se proponen las siguientes recomendaciones:

Implementación práctica del modelo Random forest en las granjas porcícolas, mediante el uso de herramientas sencillas como la carga de datos en Excel, que permitan a los productores ingresar variables básicas y obtener estimaciones inmediatas del peso animal.

Capacitación al usuario final explicándole en qué carpeta debe subir el archivo y en qué estructura, enseñándole el uso del botón que ejecuta el proceso y dirigiéndolo a la carpeta donde queda el peso en pie predicho.

Ampliación de la base de datos, a través de la incorporación de diferentes granjas en ubicación geográfica, fases productivas (sitio 3 o Wean to finish) y condiciones ambientales (temperatura y ventilación), con el objetivo de dar más estabilidad al modelo.

Incorporación de variables adicionales como genética, medidas morfométricas obtenidas mediante visión artificial, lo que podría potenciar aún más la precisión del modelo.

Evaluación en tiempo real con sensores o cámara ya existentes en el mercado, con el fin de capturar variables morfométricas sin contacto, lo que permitiría automatizar completamente la estimación del peso sin intervención humana directa.

Desarrollo de un tablero de fácil visualización y alcance a los operarios que permita utilizar el modelo predictivo desde cualquier dispositivo, puede ser desde la herramienta Power BI, que es fácil de implementar y usar.

Referencias Bibliográficas

Bhoj, S., Tarafdar, A., Chauhan, A., Singh, M., & Gaur, G. K. (2022). *Image processing strategies for pig liveweight measurement: Updates and challenges.* In Computers and Electronics in Agriculture (Vol. 193). Elsevier B.V. https://doi.org/10.1016/j.compag.2022.106693

Brossard, L., Dourmad, J. Y., Rivest, J., & Van Milgen, J. (2009). *Modelling the variation in performance of a population of growing pig as affected by lysine supply and feeding strategy.* Animal, 3(8), 1114–1123. https://doi.org/10.1017/S1751731109004546

Čandek-Potokar, M., Lebret, B., Gispert, M., & Font-i-Furnols, M. (2024). *Challenges and future perspectives for the European grading of pig carcasses – A quality view*. Meat Science, 208. https://doi.org/10.1016/j.meatsci.2023.109390

Chidgey, K. L. (2024). *Review: Space allowance for growing pigs: animal welfare, performance and on-farm practicality.* In Animal (Vol. 18). Elsevier B.V. https://doi.org/10.1016/j.animal.2023.100890

da Cunha, A. C. R., Antunes, R. C., da Costa, W. G., Rebouças, G. F., Leite, C. D. S., & do Carmo, A. S. (2024). *Body weight prediction in crossbred pigs from digital images using computer vision.* Livestock Science, 282. https://doi.org/10.1016/j.livsci.2024.105433

Drexl, V., Dittrich, I., Wilder, T., Diers, S., Janssen, H., & Krieter, J. (2024). *Prediction of tail biting in pigs using partial least squares regression and artificial neural networks*. Computers and Electronics in Agriculture, 216. https://doi.org/10.1016/j.compag.2023.108477

García-Infante, M., Castro-Valdecantos, P., Delgado-Pertíñez, M., Teixeira, A., Guzmán, J. L., & Horcada, A. (2024). *Effectiveness of machine learning algorithms as a tool to meat*

traceability system. a case study to classify Spanish mediterranean lamb carcasses. Food Control, 164. https://doi.org/10.1016/j.foodcont.2024.110604

He, H., Qiao, Y., Li, X., Chen, C., & Zhang, X. (2021). *Automatic weight measurement of pigs based on 3D images and regression network.* Computers and Electronics in Agriculture, 187. https://doi.org/10.1016/j.compag.2021.106299

Huanca-Marca, N. F., Estévez-Moreno, L. X., Espinosa, N. L., & Miranda-de la Lama, G. C. (2025). *Assessment of pig welfare at slaughterhouse level: A systematic review of animal-based indicators suitable for inclusion in monitoring protocols*. In Meat Science (Vol. 220). Elsevier Ltd. https://doi.org/10.1016/j.meatsci.2024.109689

Kim, K. H., Cho, E. S., Kim, K. S., Kim, J. E., Seol, K. H., Sa, S. J., Kim, Y. M., & Kim, Y. H. (2016). *Effects of stocking density on growth performance, carcass grade and immunity of pigs housed in sawdust fermentative pigsties*. South African Journal of Animal Science, 46(3), 294–301. https://doi.org/10.4314/sajas.v46i3.9

Masoumi, M., Marcoux, M., Maignel, L., & Pomar, C. (2021). *Weight prediction of pork cuts and tissue composition using spectral graph wavelet.* Journal of Food Engineering, 299. https://doi.org/10.1016/j.jfoodeng.2021.110501

Ruckli, A. K., Hörtenhuber, S. J., Ferrari, P., Guy, J., Helmerichs, J., Hoste, R., Hubbard, C., Kasperczyk, N., Leeb, C., Malak-Rawlikowska, A., Valros, A., & Dippel, S. (2022). *Integrative Sustainability Analysis of European Pig Farms: Development of a Multi-Criteria Assessment Tool. S*ustainability (Switzerland), 14(10). https://doi.org/10.3390/su14105988

Taylor, C., Guy, J., & Bacardit, J. (2022). *Prediction of growth in grower-finisher pigs using recurrent neural networks*. Biosystems Engineering, 220, 114–134. https://doi.org/10.1016/j.biosystemseng.2022.05.016

Tu, G. J., & Jørgensen, E. (2023). *Vision analysis and prediction for estimation of pig weight in slaughter pens*. Expert Systems with Applications, 220. https://doi.org/10.1016/j.eswa.2023.119684

Turner, S. P., D'Eath, R. B., Roehe, R., & Lawrence, A. B. (2010). *Selection against aggressiveness in pigs at re-grouping: Practical application and implications for long-term behavioural patterns.* In Animal Welfare (Vol. 19, Issue SUPPL. 1, pp. 123–132). https://doi.org/10.1017/s0962728600002323

Wei, J., Wu, Y., Tang, X., Liu, J., Huang, Y., Wu, Z., Li, X., & Zhang, Z. (2024). *Deep Learning-Based Automated Approach for Determination of Pig Carcass Traits*. Animals, 14(16). https://doi.org/10.3390/ani14162421

Xie, C., Cang, Y., Lou, X., Xiao, H., Xu, X., Li, X., & Zhou, W. (2024). *A novel approach based on a modified mask R-CNN for the weight prediction of live pigs.* Artificial Intelligence in Agriculture, 12, 19–28. https://doi.org/10.1016/j.aiia.2024.03.001

Apéndices

Apéndice A

Código Fuente

```
# Importar librerías

import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split, RandomizedSearchCV
from sklearn.ensemble import RandomForestRegressor
from sklearn.svm import SVR
from sklearn.preprocessing import PolynomialFeatures, StandardScaler,
PowerTransformer
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
import matplotlib.pyplot as plt
import seaborn as sns

# Cargar datos
file_path = "C:/Users/nathalia.monedero/Desktop/Nathalia Monedero
2024/Nathalia especializacion/Tercer corte/Trabajo de grado 2/Fase 1/Data
depurada.xlsx"
sheet_name = "Hoja1"
df = pd.read_excel(file_path, sheet_name=sheet_name)

# Información general
print("Información general del dataset:")
print(df.info())
print("\nPrimeras filas del dataset:")
print(df.head())
print("\nDescripción estadística:")
print(df.describe())

# Verificar valores nulos
print("\nValores nulos por columna:")
print(df.isnull().sum())

# Distribución de la variable objetivo
plt.figure(figsize=(8,5))
sns.histplot(df['Peso Pie Calculado'], bins=20, kde=True, color='skyblue')
plt.title('Distribución del Peso Pie Calculado')
plt.xlabel('Peso Pie Calculado')
plt.ylabel('Frecuencia')
plt.grid(True)
plt.show()
```

```
# Distribución de variables predictoras
features = ['Edad', 'Promedio de rendimiento', 'Promedio de grasa_dorsal',
'Consumo por animal']
df[features].hist(bins=15, figsize=(12, 10), layout=(3, 2),
color='lightgreen')
plt.suptitle('Distribución de Variables Predictoras')
plt.tight_layout()
plt.show()

# Identificación de Outliers usando rango intercuartil
def tratar_outliers_iqr(df, columna):
    Q1 = df[columna].quantile(0.25)
    Q3 = df[columna].quantile(0.75)
    IQR = Q3 - Q1
    limite_inferior = Q1 - 1.5 * IQR
    limite_superior = Q3 + 1.5 * IQR
    df[columna] = np.where(df[columna] > limite_superior, limite_superior,
                  np.where(df[columna] < limite_inferior, limite_inferior,
df[columna]))
    return df

# Variables con potenciales outliers
vars_outliers = ['Promedio de grasa_dorsal']
for var in vars_outliers:
    df = tratar_outliers_iqr(df, var)

# Transformaciones para normalizar distribuciones sesgadas ---
# PowerTransformer usa Yeo-Johnson por defecto (funciona con valores
negativos o cero)
pt = PowerTransformer(method='yeo-johnson')

# Variables con distribución sesgada
vars_sesgadas = ['Edad', 'Promedio de grasa_dorsal']
df[vars_sesgadas] = pt.fit_transform(df[vars_sesgadas])

# Estandarización
scaler = StandardScaler()
vars_modelo = ['Edad', 'Promedio de rendimiento', 'Promedio de grasa_dorsal']
df[vars_modelo] = scaler.fit_transform(df[vars_modelo])

# Visualización rápida para comprobar normalidad transformada


plt.figure(figsize=(12, 8))
```

```
for i, var in enumerate(vars_modelo):
    plt.subplot(2, 2, i+1)
    sns.histplot(df[var], bins=30, kde=True, color='skyblue')
    plt.title(f'Distribución transformada: {var}')
plt.tight_layout()
plt.show()
Información general del dataset:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4540 entries, 0 to 4539
Data columns (total 13 columns):
 #   Column                              Non-Null Count  Dtype
---  ------                              --------------  -----
 0   tiquete                             4540 non-null   int64
 1   Última fecha: ConsecutivoDespacho   4540 non-null   int64
 2   Fecha Sacrificio                    4540 non-null   datetime64[ns]
 3   Granja Procedencia                  4540 non-null   object
 4   Peso Pie Calculado                  4540 non-null   float64
 5   Promedio de rendimiento             4540 non-null   float64
 6   Promedio de grasa_dorsal            4540 non-null   int64
 7   Edad                                4540 non-null   int64
 8   Lote granja                         4540 non-null   object
 9   Ganancia/animal/día                 4540 non-null   float64
 10  Consumo por animal                  4540 non-null   float64
 11  CA                                  4540 non-null   float64
 12  % mortalidad                        4540 non-null   float64
dtypes: datetime64[ns](1), float64(6), int64(4), object(2)
memory usage: 461.2+ KB
None

Primeras filas del dataset:
   tiquete  Última fecha: ConsecutivoDespacho Fecha Sacrificio  \
0  1735768                                7234       2024-12-30
1  1736327                                7236       2024-12-30
2  1736326                                7236       2024-12-30
3  1735778                                7234       2024-12-30
4  1735777                                7234       2024-12-30

  Granja Procedencia  Peso Pie Calculado  Promedio de rendimiento  \
0         LA FABIOLA           90.060606                    67.61
1         LA FABIOLA           91.515152                    67.05
2         LA FABIOLA           92.727273                    67.94
3         LA FABIOLA           96.509091                    70.15
4         LA FABIOLA           98.545455                    71.63

   Promedio de grasa_dorsal  Edad Lote granja  Ganancia/animal/día  \
```

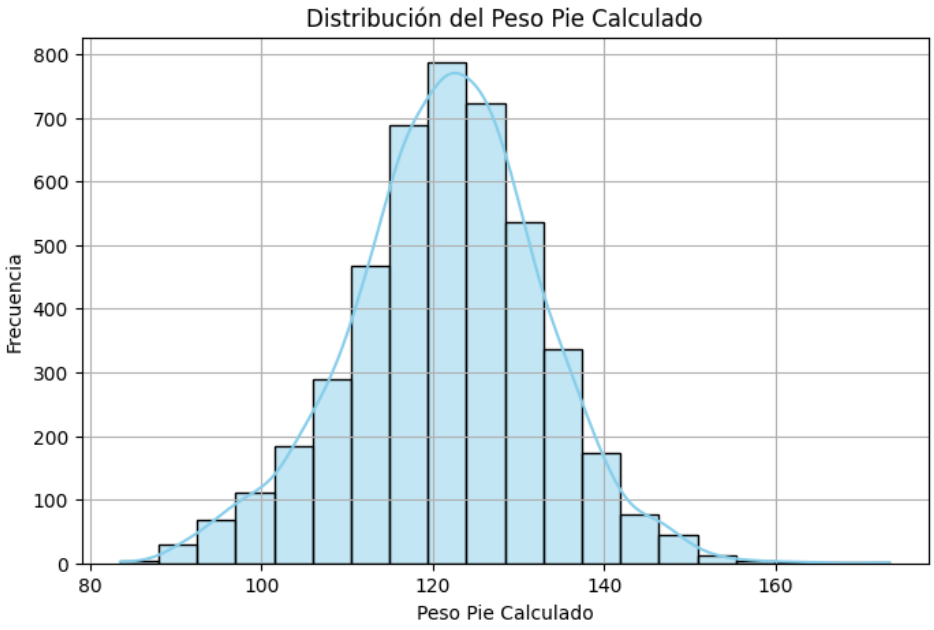
```
0                        12   144        2433                1.004
1                        12   144        2433                1.004
2                        12   144        2433                1.004
3                        13   144        2433                1.004
4                        15   144        2433                1.004

   Consumo por animal    CA  % mortalidad
0              182.69  2.29          1.11
1              182.69  2.29          1.11
2              182.69  2.29          1.11
3              182.69  2.29          1.11
4              182.69  2.29          1.11

Descripción estadística:
            tiquete  Última fecha: ConsecutivoDespacho  \
count  4.540000e+03                         4540.000000
mean   1.670716e+06                         6565.045595
min    1.544636e+06                         5291.000000
25%    1.593130e+06                         5787.000000
50%    1.674908e+06                         6607.000000
75%    1.743943e+06                         7292.000000
max    1.758690e+06                         7450.000000
std    6.858028e+04                          690.909676

                    Fecha Sacrificio  Peso Pie Calculado  \
count                           4540         4540.000000
mean   2024-10-07 18:43:46.255506432          121.540574
min              2024-04-24 00:00:00           83.490909
25%              2024-06-27 00:00:00          114.836364
50%              2024-10-15 00:00:00          121.939394
75%              2025-01-10 00:00:00          128.727273
max              2025-01-31 00:00:00          173.333333
std                              NaN           11.167096

       Promedio de rendimiento  Promedio de grasa_dorsal         Edad  \
count              4540.000000               4540.000000  4540.000000
mean                 83.892141                 15.202643   160.203524
min                  59.160000                 11.000000   144.000000
25%                  79.887500                 14.000000   155.000000
50%                  84.130000                 15.000000   160.000000
75%                  88.280000                 17.000000   166.000000
max                 100.000000                 20.000000   185.000000
std                   6.564165                  2.264683     8.157893

       Ganancia/animal/día  Consumo por animal          CA  % mortalidad
```

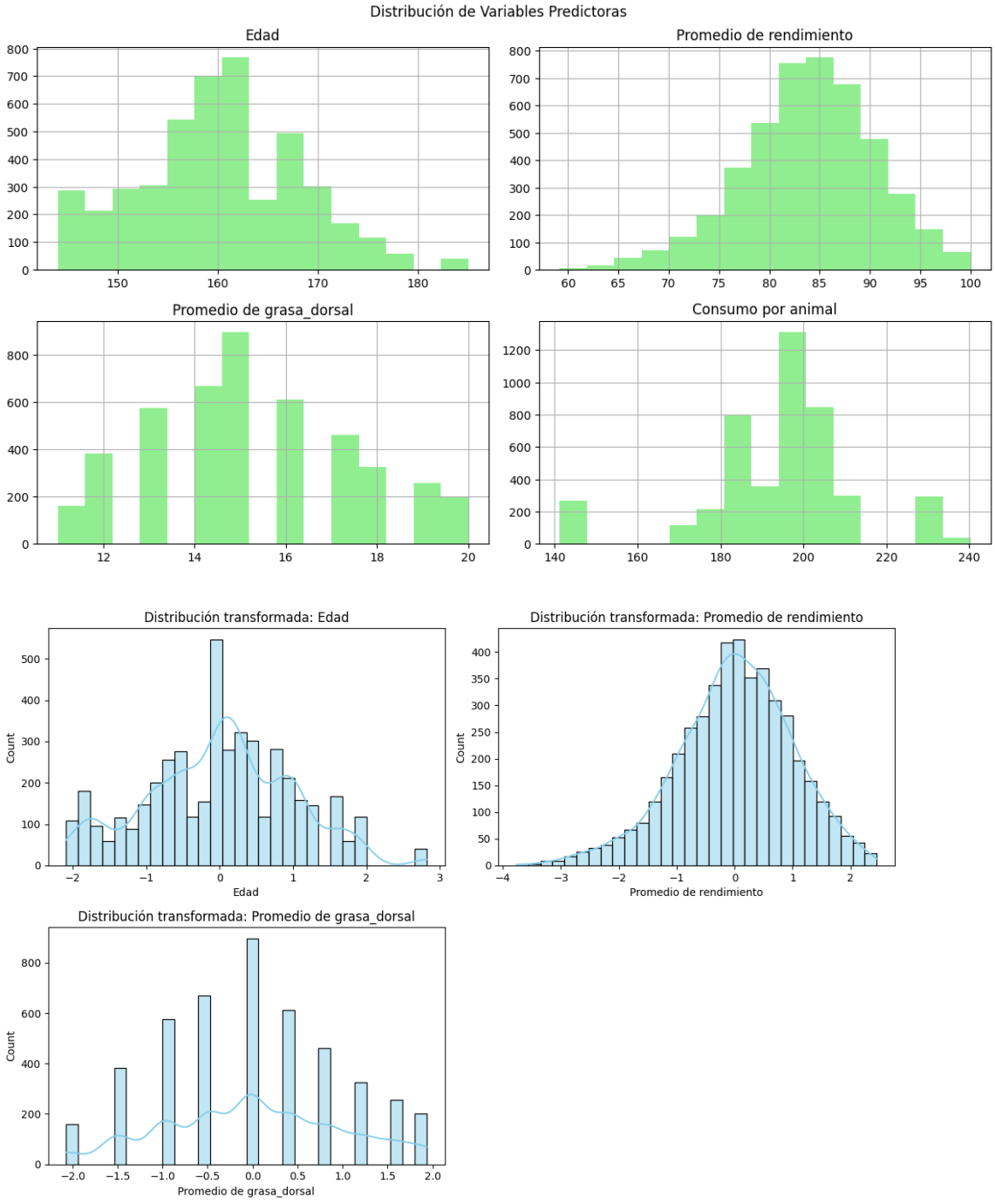
count	4540.000000	4540.000000	4540.000000	4540.000000
mean	1.001468	193.913584	2.238163	0.873500
min	0.946000	141.230000	2.020000	0.000000
25%	0.956000	182.690000	2.230000	0.580000
50%	1.004000	195.670000	2.280000	0.670000
75%	1.029000	202.550000	2.310000	1.110000
max	1.149000	240.250000	2.400000	3.280000
std	0.053437	18.936872	0.114152	0.607587

Valores nulos por columna:

tiquete	0
Última fecha: ConsecutivoDespacho	0
Fecha Sacrificio	0
Granja Procedencia	0
Peso Pie Calculado	0
Promedio de rendimiento	0
Promedio de grasa_dorsal	0
Edad	0
Lote granja	0
Ganancia/animal/día	0
Consumo por animal	0
CA	0
% mortalidad	0

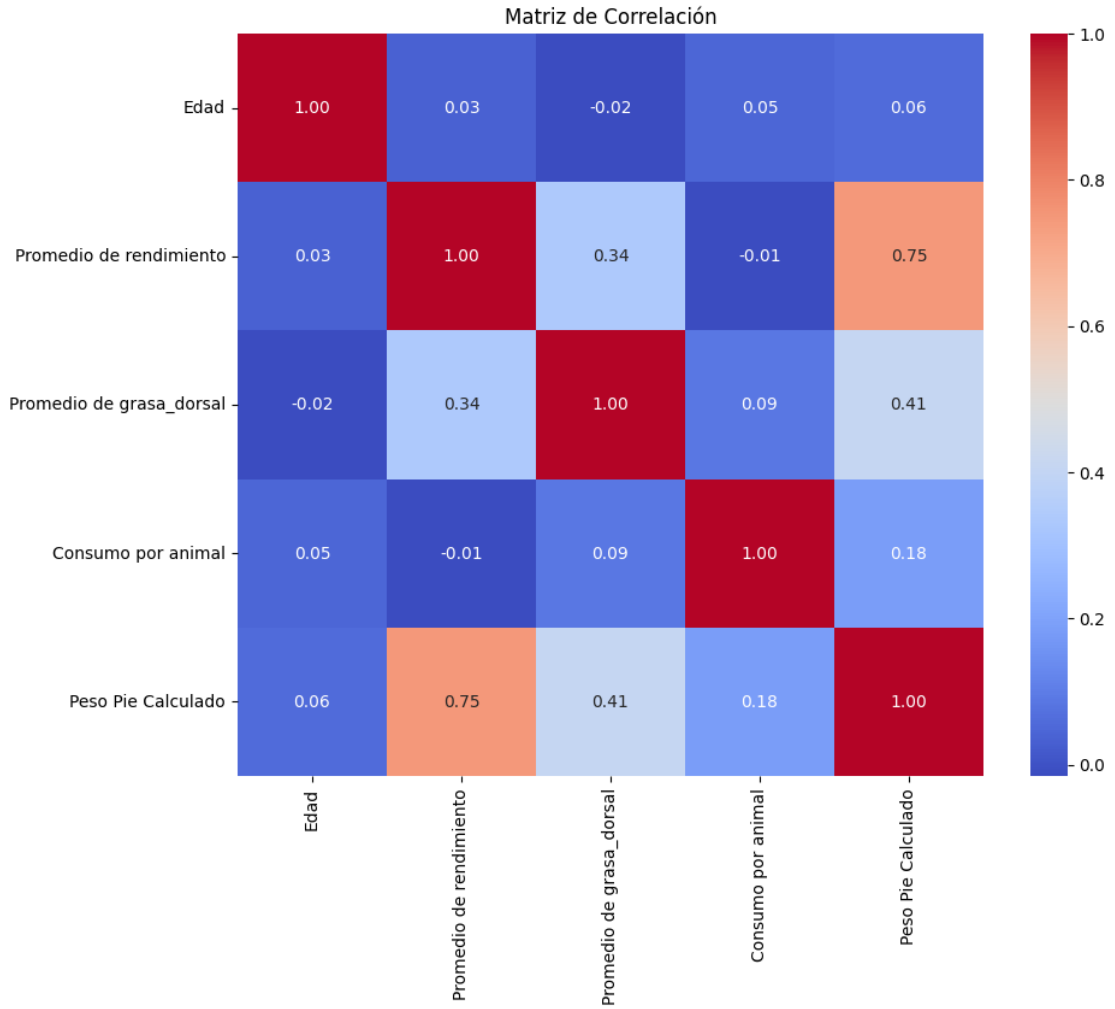
dtype: int64





In [3]:

```
# Matriz de correlación
plt.figure(figsize=(10, 8))
sns.heatmap(df[features + ['Peso Pie Calculado']].corr(), annot=True,
fmt=".2f", cmap="coolwarm")
plt.title("Matriz de Correlación")
plt.show()
```



In [4]:

```
# Definir variables predictoras y objetivo
X = df[['Edad','Promedio de rendimiento', 'Promedio de grasa_dorsal',
'Consumo por animal']]
y = df['Peso Pie Calculado']

# Dividir en conjunto de entrenamiento y prueba
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

# Normalizar las variables
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

# Generar interacciones polinomiales
poly = PolynomialFeatures(degree=2, include_bias=False)
```

```
X_train_poly = poly.fit_transform(X_train_scaled)
X_test_poly = poly.transform(X_test_scaled)

# Optimización de Random Forest
param_dist_rf = {
    "n_estimators": [50, 100, 200, 300],
    "max_depth": [5, 10, 15, 20, None],
    "min_samples_split": [2, 5, 10],
    "min_samples_leaf": [1, 2, 4]
}
rf_model = RandomForestRegressor(random_state=42)
rf_search = RandomizedSearchCV(rf_model, param_dist_rf, n_iter=10, cv=3,
scoring="neg_mean_squared_error", random_state=42, n_jobs=-1)
rf_search.fit(X_train_poly, y_train)
best_rf = rf_search.best_estimator_

# Optimización de SVR
param_dist_svr = {
    "C": [0.1, 1, 10, 100],
    "gamma": ["scale", "auto", 0.01, 0.1, 1],
    "kernel": ["rbf", "poly", "sigmoid"]
}
svr_model = SVR()
svr_search = RandomizedSearchCV(svr_model, param_dist_svr, n_iter=10, cv=3,
scoring="neg_mean_squared_error", random_state=42, n_jobs=-1)
svr_search.fit(X_train_scaled, y_train)
best_svr = svr_search.best_estimator_

# Evaluación de modelos
def evaluate_model(y_true, y_pred, model_name):
    r2 = r2_score(y_true, y_pred)
    mae = mean_absolute_error(y_true, y_pred)
    rmse = np.sqrt(mean_squared_error(y_true, y_pred))
    print(f"{model_name} - R²: {r2:.4f}, MAE: {mae:.4f}, RMSE: {rmse:.4f}")

# Predicciones
y_pred_rf = best_rf.predict(X_test_poly)
y_pred_svr = best_svr.predict(X_test_scaled)

# Evaluar modelos
evaluate_model(y_test, y_pred_rf, "Random Forest (Optimizado con
Polinomios)")
evaluate_model(y_test, y_pred_svr, "SVR (Optimizado)")
Random Forest (Optimizado con Polinomios) - R²: 0.8784, MAE: 2.9703, RMSE:
3.9453
```

```
SVR (Optimizado) - R²: 0.7636, MAE: 3.8784, RMSE: 5.5015
```

In [5]:

```
from sklearn.pipeline import make_pipeline
from sklearn.model_selection import cross_val_predict, KFold
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
import numpy as np

# Configurar validación cruzada con 5 folds
kf = KFold(n_splits=5, shuffle=True, random_state=42)

# Crear pipelines que incluyan transformaciones
pipeline_rf = make_pipeline(
    StandardScaler(),
    PolynomialFeatures(degree=2, include_bias=False),
    best_rf
)

pipeline_svr = make_pipeline(
    StandardScaler(),
    best_svr
)

# Diccionario de modelos con sus respectivos pipelines
modelos = {
    "Random Forest (Polinómico)": pipeline_rf,
    "SVR (Escalado)": pipeline_svr
}

# Evaluar modelos con validación cruzada
for nombre, modelo in modelos.items():
    y_pred_cv = cross_val_predict(modelo, X, y, cv=kf)

    r2 = r2_score(y, y_pred_cv)
    mae = mean_absolute_error(y, y_pred_cv)
    rmse = np.sqrt(mean_squared_error(y, y_pred_cv))

    print(f"{nombre} - Validación Cruzada:")
    print(f"R²: {r2:.4f}, MAE: {mae:.4f}, RMSE: {rmse:.4f}\n")
Random Forest (Polinómico) - Validación Cruzada:
R²: 0.8734, MAE: 2.9679, RMSE: 3.9736

SVR (Escalado) - Validación Cruzada:
R²: 0.7462, MAE: 3.9597, RMSE: 5.6257
```

Apéndice B

Modelo de Regresión Lineal Simple

In [6]:

```
from sklearn.linear_model import LinearRegression

# Modelo de regresión lineal simple
# Variable predictora única
X_simple = df[['Edad']]
y = df['Peso Pie Calculado']

# Dividir y entrenar
X_train_s, X_test_s, y_train_s, y_test_s = train_test_split(X_simple, y,
test_size=0.2, random_state=42)
model_simple = LinearRegression()
model_simple.fit(X_train_s, y_train_s)

# Predicción y evaluación
y_pred_simple = model_simple.predict(X_test_s)
evaluate_model(y_test_s, y_pred_simple, "Regresión Lineal Simple")



# Modelo de regresión lineal multiple
# Variables múltiples
X_multiple = df[['Edad','Promedio de rendimiento', 'Promedio de
grasa_dorsal', 'Consumo por animal']]
y = df['Peso Pie Calculado']

# Dividir y entrenar
X_train_m, X_test_m, y_train_m, y_test_m = train_test_split(X_multiple, y,
test_size=0.2, random_state=42)
model_multiple = LinearRegression()
model_multiple.fit(X_train_m, y_train_m)

# Predicción y evaluación
y_pred_multiple = model_multiple.predict(X_test_m)
evaluate_model(y_test_m, y_pred_multiple, "Regresión Lineal Múltiple")

# Modelo de regresión no lineal
# Transformación polinómica
poly = PolynomialFeatures(degree=2, include_bias=False)
X_poly = poly.fit_transform(X_multiple)
X_train_poly, X_test_poly, y_train_poly, y_test_poly =
train_test_split(X_poly, y, test_size=0.2, random_state=42)
```

```
# Entrenar modelo
model_poly = LinearRegression()
model_poly.fit(X_train_poly, y_train_poly)

# Predicción y evaluación
y_pred_poly = model_poly.predict(X_test_poly)
evaluate_model(y_test_poly, y_pred_poly, "Regresión Polinómica")
Regresión Lineal Simple - R²: 0.0064, MAE: 8.7265, RMSE: 11.2783
Regresión Lineal Múltiple - R²: 0.6070, MAE: 5.2041, RMSE: 7.0927
Regresión Polinómica - R²: 0.6708, MAE: 4.6739, RMSE: 6.4920
```

In [7]:

```
from sklearn.model_selection import cross_val_score, cross_val_predict, KFold
from sklearn.preprocessing import PolynomialFeatures
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score, mean_absolute_error, mean_squared_error
import numpy as np

# Función de evaluación con validación cruzada
def evaluate_with_cv(model, X, y, model_name, cv=5):
    kf = KFold(n_splits=cv, shuffle=True, random_state=42)
    y_pred = cross_val_predict(model, X, y, cv=kf)

    r2 = r2_score(y, y_pred)
    mae = mean_absolute_error(y, y_pred)
    rmse = np.sqrt(mean_squared_error(y, y_pred))

    print(f"{model_name} - Validación Cruzada:")
    print(f"R²: {r2:.4f}, MAE: {mae:.4f}, RMSE: {rmse:.4f}\n")


# Regresión lineal simple
X_simple = df[['Edad']]
y = df['Peso Pie Calculado']
model_simple = LinearRegression()
evaluate_with_cv(model_simple, X_simple, y, "Regresión Lineal Simple")


# Regresión lineal múltiple
X_multiple = df[['Edad','Promedio de rendimiento', 'Promedio de
grasa_dorsal', 'Consumo por animal']]
model_multiple = LinearRegression()
evaluate_with_cv(model_multiple, X_multiple, y, "Regresión Lineal Múltiple")
```

```
# Regresión polinómica (no lineal)
poly = PolynomialFeatures(degree=2, include_bias=False)
X_poly = poly.fit_transform(X_multiple)
model_poly = LinearRegression()
evaluate_with_cv(model_poly, X_poly, y, "Regresión Polinómica")
Regresión Lineal Simple - Validación Cruzada:
R²: 0.0010, MAE: 8.7427, RMSE: 11.1603

Regresión Lineal Múltiple - Validación Cruzada:
R²: 0.6196, MAE: 5.0875, RMSE: 6.8864

Regresión Polinómica - Validación Cruzada:
R²: 0.6617, MAE: 4.7041, RMSE: 6.4949
```

In [8]:

```
import openpyxl
import sklearn

import pandas as pd
from sklearn.preprocessing import StandardScaler, PolynomialFeatures
import joblib  # para guardar y cargar modelos si quieres

# Cargar nuevos datos
archivo = "Nuevos datos.xlsx"
df_nuevos = pd.read_excel(archivo)

# Variables predictoras
columnas = ['Edad','Promedio de rendimiento', 'Promedio de grasa_dorsal',
'Consumo por animal']

X_nuevos = df_nuevos[columnas]

# Transformaciones
# Escalado
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X)
X_nuevos_scaled = scaler.transform(X_nuevos)

# Polinomios para RF
poly = PolynomialFeatures(degree=2, include_bias=False)
X_train_poly = poly.fit_transform(X_train_scaled)
X_nuevos_poly = poly.transform(X_nuevos_scaled)

# Predicciones
# Random Forest con polinomios
```

```
pred_rf = best_rf.predict(X_nuevos_poly)



# Guardar predicciones
df_nuevos['Predicción_RF'] = pred_rf

df_nuevos.to_excel("Predicciones_Peso_Pie.xlsx", index=False)

print("Predicciones generadas y guardadas en 'Predicciones_Peso_Pie.xlsx'")
Predicciones generadas y guardadas en 'Predicciones_Peso_Pie.xlsx'
```

In [9]:

```
import joblib
from sklearn.preprocessing import StandardScaler, PolynomialFeatures
from sklearn.ensemble import RandomForestRegressor

# Supongamos que tienes tus datos X_train e y_train preparados
# Aquí solo pongo un ejemplo básico, reemplaza con tu código de entrenamiento
real

# 1. Crear el escalador y ajustar con X_train
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)

# 2. Crear polinomios y ajustar con datos escalados
poly = PolynomialFeatures(degree=2, include_bias=False)
X_train_poly = poly.fit_transform(X_train_scaled)

# 3. Entrenar modelo Random Forest con los datos transformados
best_rf = RandomForestRegressor()
best_rf.fit(X_train_poly, y_train)

# 4. Guardar objetos entrenados para usarlos luego
joblib.dump(scaler, 'scaler.pkl')
joblib.dump(poly, 'poly.pkl')
joblib.dump(best_rf, 'best_rf.pkl')
```

Out[9]:

```
['best_rf.pkl']
```

In [10]:

```
import os
import pandas as pd
import joblib
from tkinter import Tk, Button, messagebox
```

```
def procesar_excel(input_path, output_path):
    # Cargar datos
    df_nuevos = pd.read_excel(input_path)

    # Variables predictoras
    columnas = ['Edad','Promedio de rendimiento', 'Promedio de grasa_dorsal',
'Consumo por animal']
    X_nuevos = df_nuevos[columnas]

    # Cargar objetos guardados
    scaler = joblib.load('scaler.pkl')
    poly = joblib.load('poly.pkl')
    best_rf = joblib.load('best_rf.pkl')

    # Transformaciones usando objetos cargados
    X_nuevos_scaled = scaler.transform(X_nuevos)
    X_nuevos_poly = poly.transform(X_nuevos_scaled)

    # Predicción
    pred_rf = best_rf.predict(X_nuevos_poly)

    # Guardar predicciones
    df_nuevos['Predicción_RF'] = pred_rf

    df_nuevos.to_excel(output_path, index=False)

def ejecutar_proceso():
    input_folder = 'input'
    output_folder = 'output'

    os.makedirs(input_folder, exist_ok=True)
    os.makedirs(output_folder, exist_ok=True)

    archivos = [f for f in os.listdir(input_folder) if f.endswith(('.xls',
'.xlsx'))]
    if not archivos:
        messagebox.showerror("Error", "No se encontró ningún archivo Excel en
la carpeta 'input'.")
        return

    input_file = os.path.join(input_folder, archivos[0])
    output_file = os.path.join(output_folder, f"Predicciones_{archivos[0]}")

    try:
```

```
        procesar_excel(input_file, output_file)
        messagebox.showinfo("Éxito", f"Archivo procesado y guardado
en:\n{output_file}")
    except Exception as e:
        messagebox.showerror("Error", f"Ocurrió un error al procesar el
archivo:\n{e}")

def main():
    root = Tk()
    root.title("Predicción Peso en Pie")
    root.geometry("300x100")

    btn = Button(root, text="Procesar archivo Excel",
command=ejecutar_proceso)
    btn.pack(pady=30)

    root.mainloop()

if __name__ == "__main__":
    main()
```