Tecnológico de Monterrey

Campus Querétaro, Querétaro

Reto Datos

Inteligencia artificial avanzada para la ciencia de datos II | TC3007

Profesor:

Ismael Solis Moreno

Alumnos:

Vacas Saturno Saturnitas 🐮🪐

Kevin Alejandro Ramírez Luna		A01711063
Diego Antonio García Padilla		A01710777
José Eduardo Viveros Escamilla		A01710605
Fidel Alexander Bonilla Montalvo		A01798199
Guadalupe Paulina López Cuevas		A01701095
Ángel Mauricio Ramírez Herrera		A01710158
Cristian Chávez Guía		A01710680

19 oct 2025

Índice

Introducción

El uso de la inteligencia artificial se ha vuelto crucial en la estrategia comercial de múltiples industrias, y la industria ganadera no es la excepción. En este reto, por medio de la creación e implementación de un modelo de machine learning realizaremos un producto el cual sea capaz de ayudar en la toma de decisiones con respecto al periodo de secado de las vacas con el propósito de optimizar la producción de la leche. Este reto es importante ya que nos brinda un punto de perspectiva de una industria la cual a no parece tener mucha relación con la inteligencia artificial, sin embargo, su sistema de automatización el cual por medio de un robot ordeñan la vaca, genera una gran cantidad de datos que nos pueden servir para encontrar patrones entre el periodo de secado de una vaca el cual es parte del ciclo de lactancia de una vaca y la producción que esta tiene al momento de estar en el periodo de lactancia.

Todos esos datos son registrados como anteriormente se mencionó, por medio del robot de ordeña y del propio sistema con el que cuenta nuestro socio formador, el CAETEC. Nuestro objetivo es predecir una fecha en la cual sea ideal poner a la vaca en estado de secado por medio de un modelo de machine learning. Cada vaca es diferente al resto, por lo que puede o no rondar el tiempo de lactancia, el cual es de 305 días. Algunas vacas pueden llegar a superar por más de 100 días esa fecha, así que es importante destacar que si se prolonga por mucho tiempo el tiempo de ordeña de una vaca sin ponerla en secado, esto puede lastimarla y con ello ocasionar problemas de salud como es la mastitis, ya que al estar bajando su producción de leche, cada vez más le costará a la vaca producirla y no será igual que cuando está recién preñada.

Herramientas y tecnologías

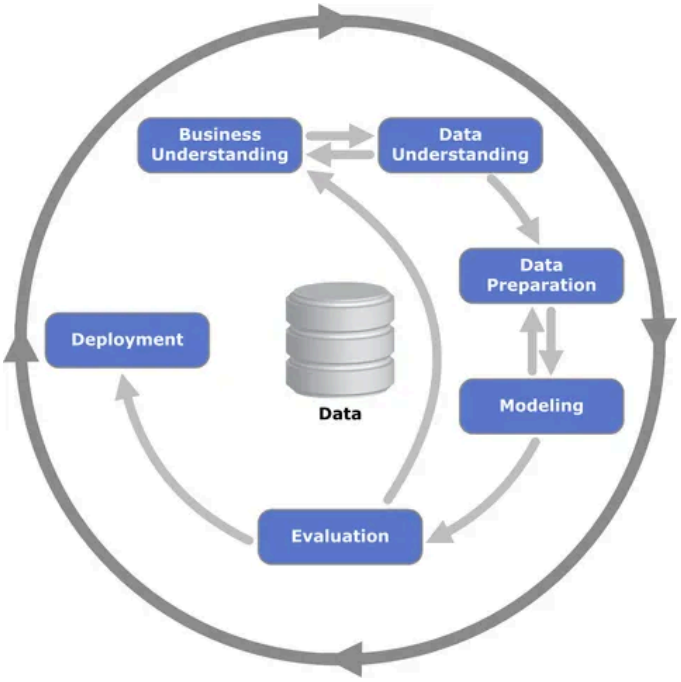

Figura 1. Fases de CRISP-DM

CRISP-DM se divide en 6 fases principales para la producción de conocimiento. En los siguientes apartados estaremos relatando cada una de las diversas tecnologías y herramientas que se usaron por cada fase.

Business Understanding

Para business understanding estuvimos usando la plataforma de Notion para la administración, gestión y documentación del proyecto. Así mismo, de manera complementaria, usamos como servicio de almacenamiento en la nube este Google Drive.

Data Understanding

En el data understanding el objetivo de esta etapa es explorar los datos, observar cómo ha sido su comportamiento a lo largo del tiempo, si es que tienen alguna distribución en específico y si existen relaciones fuertes entre los atributos. De igual forma con el entendimiento de los datos podemos saber si es que son útiles para el objetivo del proyecto o si son atributos relevantes para las predicciones del modelo.

- https://colab.research.google.com/ -Colab es el entorno inicial y útil para la ejecución ya que con este entorno podemos hacer uso de todas las librerías necesarias sin tener que instalarlas de manera local
- https://pandas.pydata.org/ - Esta es una librería que usaremos con más recurrencia debido a que es útil para la exploración de dataframes de los datos; así como el merge de dataframes.

- https://numpy.org/es/ - Esta es otra de las librerías que es útil para el uso de funciones matemáticas para la búsqueda de atributos u operaciones en los datos.
- https://matplotlib.org/ -Esta libreria nos ayuda a la visualización y graficación de datos para encontrar correlaciones significativas y ver el comportamiento de los datos.
- https://seaborn.pydata.org/ -Esta libreria de igual forma sirve para la visualización de manera un poco más avanzada

Data Preparation

Para el proceso de EDA, y en consiguiente: el proceso ETL, se estuvo usando como lenguaje de programación principal Python. Ya que cuenta con facilidades para las transformaciones, imputaciones y exploración de datos. Las librerías e tecnologías usadas se listan a continuación:

- https://colab.research.google.com/ -Entorno inicial y útil para la ejecución de todas nuestras librerías
- https://pandas.pydata.org/ - Útil para la exploración de dataframes de los datos; así como el merge de dataframes.
- https://numpy.org/es/ - Útil para el uso de funciones matemáticas para la búsqueda de atributos en los datos.
- https://matplotlib.org/ -Visualización y graficación de datos para encontrar correlaciones significativas.
- https://seaborn.pydata.org/ -Visualización un poco más avanzada

Estas tecnologías permiten una exploración de datos eficiente, reproducible y exhaustiva. NumPy y Pandas proporcionan la base computacional y de manipulación, mientras que Matplotlib y Seaborn ofrecen un espectro completo de visualización, desde lo básico y personalizable hasta lo estadístico y de alto nivel.

Cabe recalcar que como entorno para la creación de los códigos Python se usó Google Colab por su facilidad y rápida conexión con Google Drive (lugar en donde están almacenados los archivos.csv).

Además a diferencia de la utilización de notebooks con el entorno de Anaconda, Visual Studio Code o Pycharm, nos permite tenerlo de manera remota para que en cualquier momento, otro miembro pueda ingresar en cualquier momento y realizar sus propias modificaciones.

Modeling

Durante la fase de modelado, utilizaremos Matplotlib para la visualización de datos y PyTorch para el desarrollo e implementación de nuestros modelos. Esta combinación nos permitirá construir desde modelos básicos hasta arquitecturas avanzadas, asegurando que cumplan con los requisitos del proyecto. Asimismo, aprovecharemos las capacidades de estas librerías para realizar el ajuste de hiperparámetros y calcular las métricas de evaluación necesarias, lo que las convierte en herramientas fundamentales para esta etapa.

Evaluation

La evaluación del modelo es una etapa fundamental dentro del proceso de análisis y desarrollo de modelos predictivos o de aprendizaje automático. Su propósito es verificar el desempeño del modelo, identificar su capacidad para generalizar correctamente sobre nuevos datos y determinar si los resultados obtenidos son confiables y útiles para la toma de decisiones. En esta fase, se emplean métricas y visualizaciones que permiten interpretar los resultados, comparar diferentes versiones del modelo y comunicar los hallazgos de manera clara y visual.

- https://www.microsoft.com/es-es/power-platform/products/power-bi.Power BI: Su uso en la evaluación del modelo facilita la interpretación de los resultados al integrar diferentes fuentes de datos, crear indicadores personalizados y representar visualmente el rendimiento del modelo en tiempo real.
- https://www.tableau.com/es-mx Tableau: Permite contrastar los resultados esperados frente a los reales, comparar distintos escenarios y analizar el comportamiento del modelo desde un enfoque exploratorio y visual.

Deployment

Para el deployment deberemos usar S3. En la sección de abajo se detalla con más detalle el uso y propósito de este servicio de AWS.

Base de datos

Para el almacenamiento y la consulta de los datos, utilizaremos el servicio AWS S3, donde los usuarios podrán subir archivos .csv con la información de ordeño de las vacas. Para gestionar la información asociada a estos archivos, emplearemos PostgreSQL como base de datos relacional, donde se almacenarán las URLs de acceso a los archivos en S3. Toda la infraestructura de la base de datos se ejecutará utilizando Docker, lo que facilitará la configuración, despliegue y portabilidad del entorno.

- AWS S3 servicio de almacenamiento en la nube que almacena datos como "objetos" en contenedores llamados "buckets".

- PostgreSQL es un sistema de gestión de bases de datos relacionales de código abierto.
- Docker es una plataforma de código abierto que automatiza el despliegue, la ejecución y la gestión de aplicaciones en contenedores.

Fase de visualización de datos

Modelo y almacenamiento de los datos

El CATEC, para la comparación e histórico de sus datos, almacena los datos producidos por el robot de la marca DELAVAL desde que este llegó al rancho (alrededor de 8 años). Entonces, considerando esto y la gran cantidad de datos que podemos llegar a tener, es importante que intentemos incorporar un sistema lo suficientemente robusto para toda esta información.

Ciclo de vida:

Fase	Generación y captura de los datos	Exploración	Limpieza	Transformación / Preparación	Modelado	Evaluación	Despliegue y mantenimiento
Acción	El robot del caetec registra los datos de ordeño de cada vaca y su ficha técnica	Por medio de librerías y herramientas como NumPy, Pandas, Matplotlib, Seaborn vemos los datos faltantes, exploramos las correlaciones y las distribuciones de los datos.	Borramos las columnas las cuales no nos sirven o no hay registros de las mismas	Combinamos todos los 44 archivos de cada vaca, se realizan transformaciones, se crean variables dummies. Se identifican las variables X y variable Predictora. Se hace la partición	Se crea la arquitectura del modelo, definiendo que tipo de estructura vamos a utilizar Pytorch	Se realiza el entrenamiento del modelo y con base en nuestras metricas determinamos si es un buen modelo, si existe overfitting, underfitting y si esta cumpliendo su objetivo	El archivo de nuestro modelo entrenado se guarda en nuestra instancia donde estará nuestra aplicación. El modelo nos servirá para predicciones

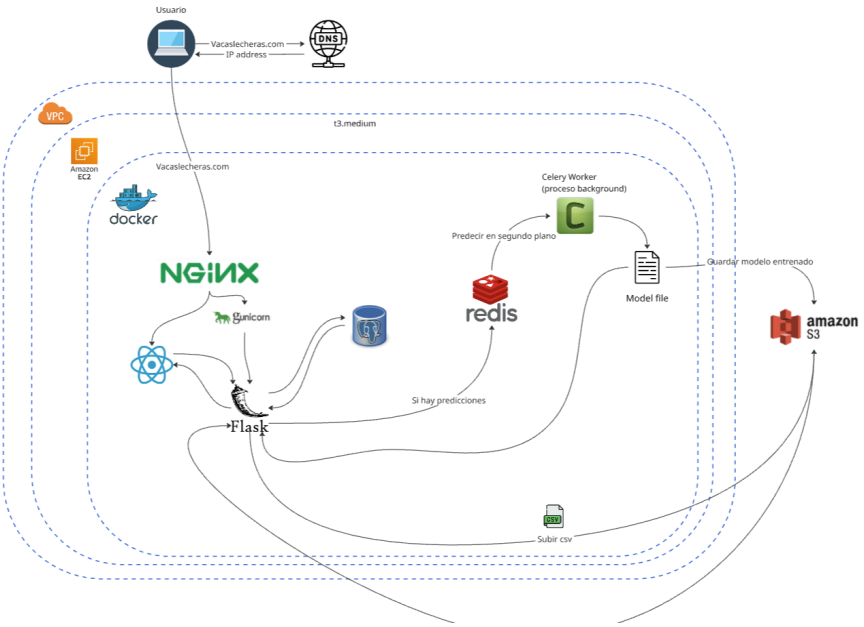

Figura 2. Arquitectura propuesta

En general la recolección de datos lo hará el CAETEC por medio de su robot, y los miembros del CAETEC podrán descargar esos datos por medio de archivos .csv los cuales estarán por así decirlo crudos. Para nuestra solución realizaremos una aplicación que nos permita realizar predicciones parar el mejor momento de poner a una vaca en el periodo de secado, nuestro archivo de nuestro modelo ya entrenado se guardará en nuestra instancia, para esa predicción el usuario guardará un archivo csv del registro del ordeño de esa vaca en específico, ese archivo se guardará en un S3 y de ahí será utilizado por nuestro sistema para las predicciones. En nuestro S3 se van a ir guardando los csv de cada vaca de las cuales se les quiera hacer una predicción, ya que se podrán consultar y volver a realizar predicciones con estos mismos archivos por lo que será importante preservar y guardar estos datos. Además se podrán actualizar los datos por datos más recientes de las vacas con el objetivo de tener mejores predicciones. En cuanto a nuestros usuarios, los datos de los mismos serán guardados en una base de datos de Postgres DB. El almacenamiento de los datos en Postgres no será una gran cantidad de datos puesto que los usuarios serán una cantidad reducida, y no tendrá mucho crecimiento. En cuanto a los csv estos sí podrán llegar a ser una cantidad considerable y pueden ser hasta cierto punto algo pesados dependiendo de la cantidad de lactancias de la vaca que se vaya a predecir, para eso utilizaremos el servicio de S3 ya que si intentamos almacenar todos esos csv en nuestra instancia, el precio saldría demasiado caro, estos csv serán preservados en nuestra instancia y serán actualizados si se vuelve a subir un nuevo archivo de una vaca en específico ya que este nuevo csv tendrá los nuevos datos y nos servirá para predicciones más precisas.

Limpieza y preparación de los datos

Para el análisis exploratorio de los registros dados por CAETEC, se inició con cuatro archivos principales que contenían la información base:

- ID Vaca.csv (con registros desde la vaca 1204 hasta la 8794)
- patadas.csv
- inventario.csv
- reporte.csv.

Estos archivos sirvieron como punto de partida para establecer un proceso estandarizado de limpieza y transformación de datos.

Ahora, a modo de listado, en la siguiente sección se detalla lo que se puede observar en cada csv.

1. El proceso comenzó con la carga del dataset principal (descargado de manera local) en el entorno de Google Colab, donde se montó la unidad de Drive y se accedió al archivo CSV correspondiente al análisis. Una vez cargado el dataset, se procedió a verificar sus dimensiones originales, identificando el número total de registros y variables disponibles para su registro.

2. La siguiente etapa consistió en una evaluación exhaustiva de la calidad de los datos. Se realizó una búsqueda de registros duplicados, contabilizando aquellos que aparecían repetidos en el dataset. Para, posteriormente, examinar la presencia de valores nulos por cada columna, creando un inventario completo de los datos faltantes en el conjunto original.
    a. Dada la importancia de contar con datos completos para el análisis, se estableció un criterio de limpieza basado en el porcentaje de valores nulos. Se identificaron aquellas columnas que presentaban más del 50% de datos faltantes y se procedió a eliminarlas del dataset, conservando únicamente las variables con suficiente información para ser útiles en el análisis posterior.

    b. Adicionalmente, se detectaron columnas específicas que contenían exclusivamente valores cero, las cuales no aportan información relevante para el modelo. Estas columnas fueron eliminadas explícitamente, resultando en un dataset más significativo y cómodo para nuestras implementaciones futuras.

3. Una vez completada la limpieza básica, se inició la fase de exploración mediante visualizaciones. Se generaron histogramas y gráficos de barras para todas las variables restantes, excluyendo aquellas relacionadas con fechas e identificadores. Esto permitió observar las distribuciones de los datos y detectar patrones o anomalías en las variables numéricas y categóricas en contraste al dataframe.

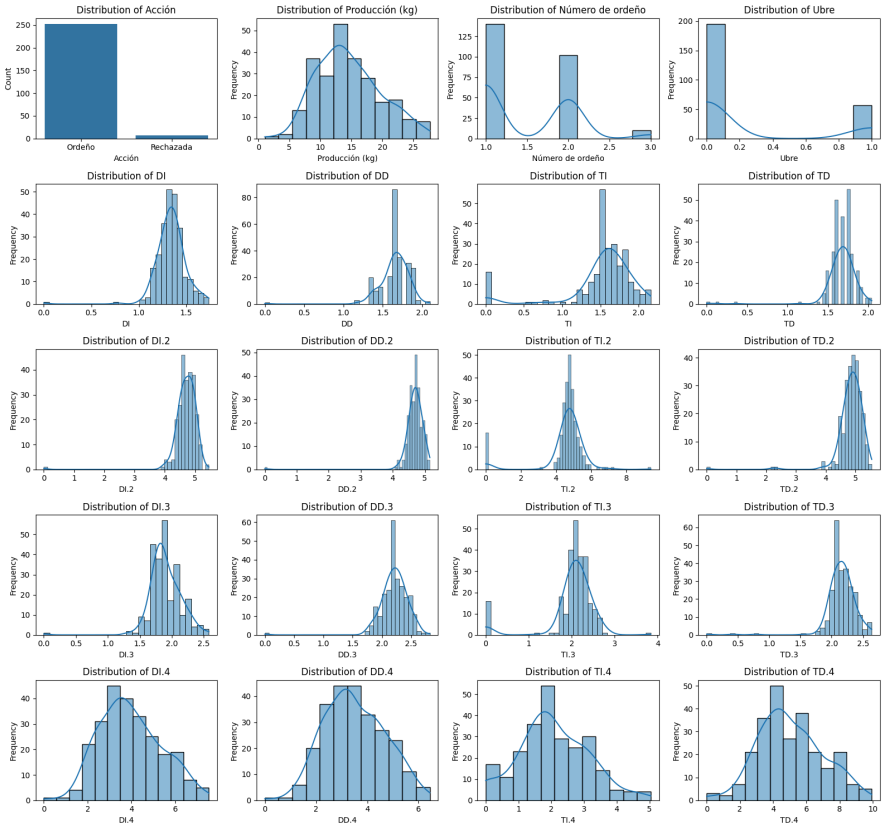

Figura 3. Histogramas generados para el csv de vacas

4. El análisis de valores faltantes continuó con la creación de un mapa de calor que mostraba visualmente la distribución de los datos nulos a lo largo del dataset. Se calcularon los porcentajes de valores faltantes por columna, priorizando aquellas variables que requerirían estrategias de imputación más elaboradas.

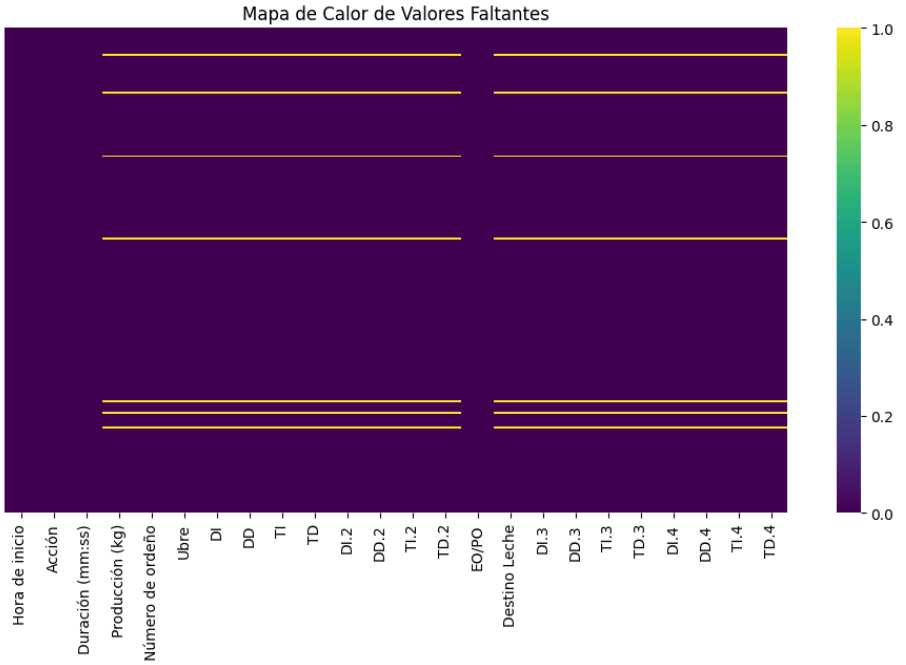

Figura 4. Mapa de calor generado para el csv de vacas

5. Para comprender mejor las características del dataset, se generaron estadísticas descriptivas completas que incluían medidas de tendencia central y dispersión para variables numéricas, así como distribuciones de frecuencia para variables categóricas.

6. Finalmente, se realizó un análisis de correlaciones entre las variables numéricas mediante un mapa de calor, identificando las relaciones más fuertes tanto positivas como negativas. Este análisis proporcionó insights valiosos sobre la estructura subyacente de los datos y las posibles interdependencias entre variables para la consideración en el modelo de Machine Learning.

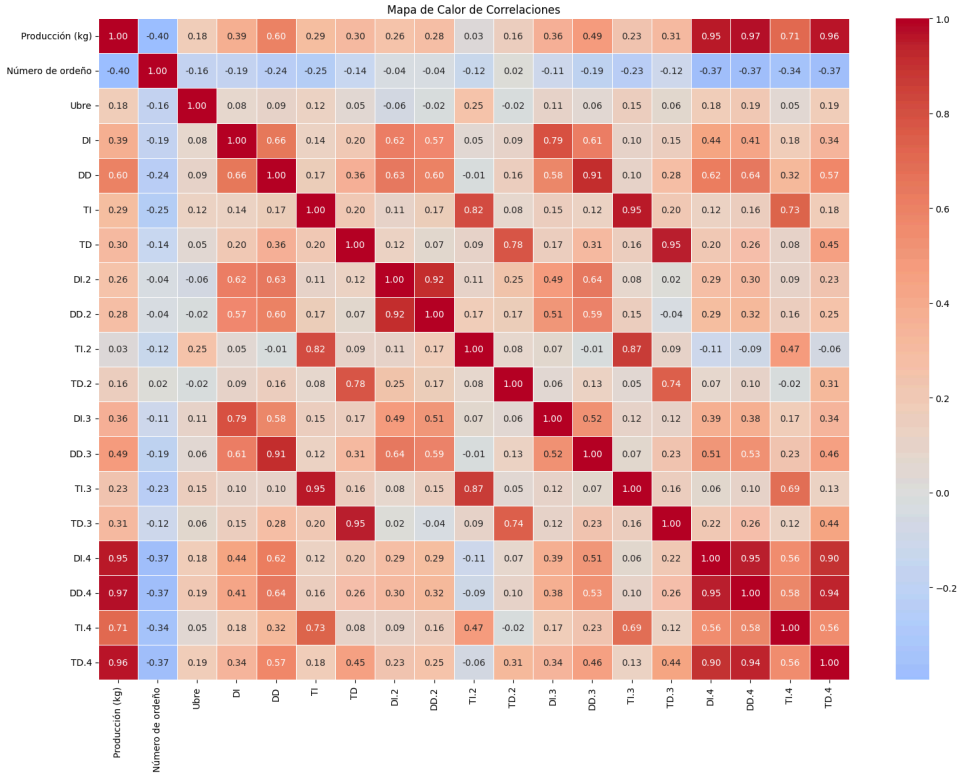

Figura 5. Mapa de correlaciones generado para el csv de vacas

Este proceso sistemático de limpieza y exploración sentará las bases para etapas posteriores de imputación y modelado, asegurando que los análisis se realizan sobre datos de calidad y con una comprensión profunda de sus características fundamentales.

Scripts usados

Los archivos de collab se listan a continuación:

- Vacas Data Exploration.ipynb
- reporte_DataUnderstanding.ipynb
- Patadas_Data exploration
- Datos de ordeña - Todas las vacas
- events.ipynb

Para más detalle de los datos, se recomienda checar el enlace al Notion del equipo:
Notion - Data Description

Separación de sets de entrenamiento

En el siguiente Google Colab es una versión preliminar para la división de datos de Test y Train del .csv de los de registros de ordeña de las 44 vacas, la segmentación de los datos la hicimos a través de KFolds, escogiendo el mejor comportamiento de 4 algoritmos (Random Forest, Ridge, Linear Regression, Lasso), el proceso se documentó en el archivo siguiente:

edve_Datos de ordeña - Todas las vacas

¿Es necesario usar Big Data?

Los datos de las vacas comprenden alrededor de 50,000 registros, y el archivo pesa aproximadamente 7 MB. Dado que estos registros abarcan los últimos 3 años, esto significa que, incluso si el número de registros creciera durante los próximos 10 años, el tamaño estimado del archivo sería de aproximadamente 23 MB, con un total de entre 166,670 y 170,000 registros. Por esta razón, seguimos hablando de Small Data, cuyo rango abarca desde miles hasta cientos de miles de registros.

El concepto de Big Data se refiere a millones, miles de millones o incluso petabytes de datos. Por lo tanto, un archivo CSV con 50,000 registros es fácilmente manejable. Esto implica que, incluso trabajando con herramientas como Google Colab, que proporciona hasta 12 GB de RAM, podemos utilizar pandas u otras librerías tradicionales de Python sin problema. En otras palabras, no necesitamos usar una "bazuca para matar una hormiga".

Implementar tecnologías como Hadoop o Spark, con procesamiento distribuido y clústeres de servidores, agregaría una complejidad innecesaria, además de costos adicionales, tiempo extra de desarrollo y mayor mantenimiento, recursos que actualmente tenemos limitados como equipo.

Las ventajas de trabajar con este volumen de datos incluyen que la limpieza se realiza en segundos y que el entrenamiento de modelos de machine learning es mucho más rápido. En conclusión, podemos manejar los datos eficientemente utilizando las librerías tradicionales del stack de Python.

Según la definición de Coursera:
"Small data, as you might guess, comprises data sets small enough for human comprehension and analysis. It concerns identifying precise causations within an isolated ecosystem and is often used to address immediate needs or answer specific questions."

Esto quiere decir que nuestro proyecto cumple exactamente con las características de Small Data: trabajamos con un dataset comprensible, buscamos identificar causaciones específicas (la relación entre el periodo de secado y la producción de leche), operamos en un ecosistema aislado osea el rancho del CAETEC, y respondemos una pregunta específica (cuándo es el momento óptimo para secar cada vaca). Además, según las "3 V's" que definen Big Data (Volumen, Velocidad y Variedad), nuestro proyecto no cumple con ninguna: tenemos bajo volumen (7 MB vs terabytes), baja velocidad (datos históricos de 3 años vs generación continua en tiempo real), y baja variedad (archivos CSV

estructurados vs múltiples fuentes no estructuradas). Por lo tanto, no solo es innecesario utilizar Big Data, sino que sería contraproducente, ya que las herramientas de Small Data nos permiten obtener insights más rápidamente, con menor costo.