UNIVERSIDAD NACIONAL DANIEL ALCIDES CARRIÓN

FACULTAD DE INGENIERÍA

ESCUELA DE FORMACIÓN PROFESIONAL DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN

MONOGRAFÍA CAPÍTULO 2

Preparación de Datos en Proyectos de Machine Learning

Autor (res):

CHACON FIRMA, Jeyson Carlos

FERNANDEZ MARCELO, Gerald Omar

HUACHO PUENTE, Ronaldo Ruben

JULCA ARZAPALO, Anthoane Camila

TORRES LUCAS, Luis Josue

Docente:

MUÑOZ ROBLES, Williams Antonio

Cerro de Pasco - Perú – 2025

ÍNDICE

INTRODUCCIÓN

La preparación de datos constituye la fase más crítica en un proyecto de Machine Learning, hasta el punto de que diversos autores estiman que puede llegar a representar entre el 70% y el 80% del esfuerzo total de desarrollo. Aunque muchas veces la atención suele centrarse en los algoritmos y modelos, en la práctica se reconoce que los datos son el verdadero corazón de un proyecto. Sin un adecuado trabajo de preprocesamiento, incluso los modelos más sofisticados están destinados a fallar, ya que no lograrán aprender patrones válidos ni realizar predicciones confiables.

En su estado original, los datos raramente están listos para ser utilizados. Generalmente provienen de fuentes heterogéneas bases de datos, sensores, encuestas, registros transaccionales, redes sociales, entre otras y presentan problemas como valores faltantes, ruido, duplicados, escalas inconsistentes o formatos no numéricos. Frente a esta realidad, el científico de datos debe realizar un conjunto de tareas que aseguren que la información esté completa, coherente y adaptada a los requerimientos de los algoritmos.

La preparación de datos no se limita a limpiar errores o corregir inconsistencias, sino que también incluye pasos más avanzados como la selección de características relevantes, la transformación de variables para que cumplan supuestos estadísticos, la reducción de dimensionalidad cuando existen demasiados atributos, y la creación de nuevas características que capturen relaciones importantes en los datos. Estas actividades requieren no solo conocimientos técnicos, sino también comprensión del problema de negocio o investigación que se está abordando, pues solo de esta forma se logran datasets que representen adecuadamente la realidad.

En consecuencia, este proceso se convierte en el verdadero motor de todo proyecto de Machine Learning. Un buen ejemplo es la frase recurrente en la comunidad científica:

“garbage in, garbage out” (basura entra, basura sale), que sintetiza la idea de que los algoritmos, por muy potentes que sean, no pueden generar valor si la información de entrada es de baja calidad. En otras palabras, la calidad del modelo depende directamente de la calidad de los datos.

Por ello, estudiar la preparación de datos no solo permite comprender un aspecto técnico del flujo de trabajo en Machine Learning, sino también valorar el impacto que este proceso tiene en la construcción de modelos predictivos exitosos. La relevancia de esta etapa radica en que constituye la base sobre la cual se fundamenta todo el proyecto: desde la correcta definición del problema hasta la evaluación final del modelo. En definitiva, la preparación de datos no es una tarea secundaria ni meramente operativa, sino un componente estratégico que determina la eficacia de las soluciones de inteligencia artificial.

CAPÍTULO 1

Data Preparation in a Machine Learning Project

1.1 Applied Machine Learning Process

El proceso aplicado de *Machine Learning* se desarrolla en cuatro fases fundamentales:

1. **Definir el problema:** en esta etapa se establecen los objetivos del proyecto, se determina la variable objetivo y se delimitan los criterios de éxito. Una definición precisa es esencial, ya que una mala formulación puede invalidar todo el estudio.

2. **Preparar los datos:** consiste en transformar la información cruda en un formato compatible con los algoritmos. Aquí se aplican técnicas de limpieza, integración, normalización, selección y transformación de variables. Esta fase suele requerir la mayor parte del tiempo de trabajo, pues los datos reales rara vez están listos para usarse directamente.

3. **Evaluar los modelos:** una vez que los datos están preparados, se prueban distintos algoritmos con el fin de medir su rendimiento mediante métricas específicas (precisión, recall, F1-score, RMSE, AUC, entre otros). El desempeño de los modelos depende en gran medida de la calidad de los datos tratados en la fase anterior.

4. **Finalizar el modelo:** consiste en elegir el algoritmo más eficiente, ajustar sus hiperparámetros y validarlo con datos independientes. El modelo resultante se prepara entonces para su implementación en un entorno real.

En este flujo de trabajo, la preparación de datos constituye el núcleo que conecta la definición del problema con la fase de modelado. Su calidad impacta directamente en la confiabilidad de los resultados y en la aplicabilidad de la solución.

1.2 What Is Data Preparation

La preparación de datos se define como el proceso mediante el cual se transforman datos crudos en representaciones adecuadas para los algoritmos de aprendizaje automático.

Este proceso abarca:

- Convertir variables categóricas en valores numéricos (por ejemplo, países transformados en variables binarias mediante *One-Hot Encoding*).
- Homogeneizar escalas para evitar que una variable con valores grandes domine el análisis.
- Asegurar la coherencia y consistencia de la información.
- Adaptar datos no estructurados (texto, imágenes, audio) en vectores numéricos que puedan ser interpretados por los algoritmos.

De esta manera, la preparación de datos no solo responde a un requisito técnico, sino también a una necesidad metodológica: garantizar que la información refleje fielmente la realidad del problema de estudio y, al mismo tiempo, pueda ser procesada por los modelos predictivos.

1.3 How to Choose Data Preparation Techniques

La elección de técnicas de preparación de datos debe realizarse de forma ética, objetiva y justificada, considerando tanto el tipo de datos como el algoritmo que se empleará.

- Según el tipo de datos:
    - **Datos numéricos:** requieren normalización o estandarización.
    - **Datos categóricos:** deben codificarse mediante *Label Encoding* o *One-Hot Encoding*.

    - **Datos textuales:** necesitan representaciones vectoriales como *Bag of Words* o *TF-IDF*.
    - **Datos temporales:** pueden descomponerse en variables como día, mes, año o tendencias.
- Según el algoritmo utilizado:
    - **KNN (K-Nearest Neighbors):** muy sensible a escalas → requiere normalización.
    - **Regresión lineal y redes neuronales:** se benefician de estandarización.
    - **Árboles de decisión y Random Forest:** menos sensibles a escalas, pero sensibles a variables irrelevantes.
    - **SVM:** funciona mejor con datos estandarizados.

Un principio metodológico clave es no aplicar técnicas de manera automática, sino experimentar con diferentes opciones, medir sus efectos y documentar el impacto en los resultados.

1.4 Summary

La preparación de datos es un paso esencial y transversal en el proceso de *Machine Learning*. Permite que los algoritmos trabajen con información clara, consistente y adaptada al problema planteado. Sin este proceso, el modelado resulta ineficiente o incluso inviable.

En conclusión, la preparación de datos no es una etapa secundaria, sino un componente estratégico que determina la calidad y confiabilidad de los resultados en cualquier proyecto de aprendizaje automático.

CAPÍTULO 2

Why Data Preparation is So Important

2.1 What is Data in Machine Learning

En el contexto de *Machine Learning*, los algoritmos solo pueden trabajar con vectores de números. Sin embargo, los datos reales no suelen estar en ese formato: se encuentran como texto, categorías, fechas, imágenes, audio o registros transaccionales.

Para que los algoritmos funcionen correctamente, es necesario transformar la información original en representaciones numéricas. Por ejemplo:

- Una columna "País" con valores *Perú, México, Chile* debe convertirse en variables binarias mediante *One-Hot Encoding*.
- Una fecha como *15/09/2025* puede descomponerse en día, mes y año para capturar patrones temporales.
- Una imagen se representa como una matriz de píxeles con valores numéricos entre 0 y 255.

Esto demuestra que la preparación de datos es indispensable para que los algoritmos puedan interpretar y procesar la información de manera efectiva.

2.2 Raw Data Must Be Prepared

Los datos en su estado bruto presentan múltiples imperfecciones que deben corregirse:

- **Valores faltantes:** registros incompletos que dificultan el análisis.
- **Ruido o inconsistencias:** errores en el ingreso de datos, como edades negativas o fechas imposibles.
- **Escalas diferentes:** por ejemplo, una variable en dólares y otra en soles.

- **Duplicados:** registros repetidos que distorsionan los patrones.

Ejemplo: un dataset con ingresos anuales (en miles) y edades (en años) sin ajustar escalas dará más peso a la variable ingresos, afectando algoritmos basados en distancias como KNN.

Por lo tanto, la preparación busca reducir el impacto de estas imperfecciones para que los modelos aprendan de manera confiable.

2.3 Predictive Modeling is Mostly Data Preparation

Aunque en teoría el interés suele centrarse en los algoritmos, en la práctica el 80% del tiempo en un proyecto de *Machine Learning* se dedica a la preparación de datos.

Las razones son:

- Los algoritmos ya están disponibles en librerías como *scikit-learn*, *TensorFlow* o *PyTorch*.
- Cada problema presenta datos únicos, lo que exige un trabajo personalizado de limpieza, transformación y selección.
- La calidad de los datos influye más en los resultados que el algoritmo elegido.

Ejemplo real: dos equipos pueden usar la misma librería de regresión logística, pero si uno realiza un preprocesamiento exhaustivo y el otro no, los resultados serán radicalmente diferentes.

En otras palabras, lo que diferencia un proyecto exitoso de uno fallido no es el algoritmo, sino el manejo de los datos.

2.4 Further Reading

La bibliografía sugiere reforzar conocimientos en temas relacionados como:

- **Data Quality:** criterios para medir la integridad, consistencia y precisión de los datos.
- **Data Cleansing:** técnicas de detección y corrección de errores.
- **Data Preprocessing:** métodos de normalización, estandarización y codificación.

Estos recursos complementarios ayudan a profundizar en las buenas prácticas de preparación de datos.

2.5 Summary

La preparación de datos es crucial porque:

- Los algoritmos solo entienden números, y los datos reales vienen en múltiples formatos.
- Los datos crudos presentan errores y escalas inconsistentes que deben corregirse.
- La mayor parte del esfuerzo en *Machine Learning* se concentra en esta fase.

En conclusión, la calidad de un modelo depende directamente de la calidad de los datos. La preparación no es opcional, sino un requisito indispensable para garantizar resultados confiables y aplicables en el mundo real.

CAPÍTULO 3

Tour of Data Preparation Techniques

3.1 Common Data Preparation Tasks

Las tareas de preparación de datos son diversas y varían según el tipo de información y el algoritmo a utilizar. Sin embargo, existen prácticas comunes que suelen aparecer en la mayoría de los proyectos:

- **Limpieza de datos (Data Cleaning):** detección y corrección de errores.
- **Selección de características (Feature Selection):** identificar las variables más relevantes.
- **Transformación de datos (Data Transforms):** ajustar escalas y formatos.
- **Ingeniería de características (Feature Engineering):** crear nuevas variables útiles.
- **Reducción de dimensionalidad (Dimensionality Reduction):** simplificar el dataset eliminando redundancias.

Estas tareas no son excluyentes, sino que se combinan para asegurar un dataset robusto y confiable.

3.2 Data Cleaning

La limpieza busca garantizar que los datos sean consistentes, completos y válidos. Ejemplos de técnicas:

- **Eliminación de duplicados:** registros repetidos que distorsionan los resultados.
- **Imputación de valores faltantes:** usar la media, mediana o métodos avanzados (KNN Imputer, MICE).

- **Detección y tratamiento de outliers:** valores atípicos que alteran estadísticas.
- **Corrección de inconsistencias:** como códigos mal ingresados o formatos incompatibles.

Ejemplo práctico: si en un dataset de estudiantes aparece la edad de 150 años, se considera un outlier y debe revisarse o eliminarse.

3.3 Feature Selection

No todas las variables aportan información útil. De hecho, incluir datos irrelevantes puede provocar sobreajuste y modelos más complejos sin mejorar el desempeño.

Métodos comunes:

- **Métodos de filtro:** selección basada en estadísticas (correlación, chi-cuadrado).
- **Métodos wrapper:** prueban combinaciones de variables (ejemplo: *Recursive Feature Elimination*).
- **Métodos embebidos:** selección automática durante el modelado (ejemplo: *Lasso Regression*).

Ejemplo: en un modelo de predicción de crédito, variables como "número de mascotas" podrían no aportar nada relevante y conviene eliminarlas.

3.4 Data Transforms

Los datos requieren ajustes para cumplir con los supuestos de los algoritmos. Entre las técnicas más usadas están:

- **Normalización:** escala de valores entre 0 y 1, útil en KNN y redes neuronales.
- **Estandarización:** media = 0, desviación estándar = 1, útil en regresión logística o SVM.

- **Codificación de categorías:** *Label Encoding* para variables ordinales, *One-Hot Encoding* para variables nominales.
- **Transformaciones logarítmicas:** reducen el sesgo en variables con distribuciones muy asimétricas.

Ejemplo en Python:

```
from sklearn.preprocessing import MinMaxScaler

scaler = MinMaxScaler()

X_scaled = scaler.fit_transform(X)
```

3.5 Feature Engineering

La ingeniería de características es la creación de nuevas variables a partir de las existentes para capturar mejor la información del problema.

Ejemplos:

- Extraer el año y el mes de una variable fecha.
- Generar variables polinómicas (ejemplo: x^2, x^3) para modelos lineales.
- Agrupar categorías con poca frecuencia en una sola clase (“otros”).

Esta técnica es considerada un arte dentro del *Machine Learning*, ya que requiere tanto conocimiento técnico como comprensión del dominio del problema.

3.6 Dimensionality Reduction

Cuando un dataset tiene demasiadas variables, puede presentarse la *maldición de la dimensionalidad*. Esto significa que el modelo se vuelve más complejo, lento y con riesgo de sobreajuste.

Técnicas más usadas:

- **PCA (Análisis de Componentes Principales):** transforma variables originales en un conjunto más pequeño de componentes que retienen la mayor parte de la varianza.
- **LDA (Linear Discriminant Analysis):** maximiza la separabilidad entre clases.
- **SVD (Singular Value Decomposition):** útil en sistemas de recomendación y procesamiento de texto.

Ejemplo: en un dataset con 500 variables, PCA puede reducirlo a 20 componentes con el 95% de la información original.

3.7 Summary

Las técnicas de preparación de datos abarcan desde tareas básicas de limpieza hasta procesos avanzados de reducción de dimensionalidad. Cada técnica cumple un rol específico: mejorar la calidad de la información, reducir el ruido, resaltar los patrones relevantes y optimizar el aprendizaje de los modelos.

En conclusión, la preparación de datos no es un conjunto de pasos aislados, sino un proceso integral y estratégico que asegura la solidez y confiabilidad de los modelos predictivos.

CAPÍTULO 4

Data Preparation Without Data Leakage

4.1 Problem With Naive Data Preparation

Un error común en la práctica es preparar los datos antes de dividirlos en entrenamiento y prueba. Esto parece eficiente, pero en realidad genera un problema grave conocido como data leakage o fuga de datos.

El *data leakage* ocurre cuando información del conjunto de prueba se filtra, de forma directa o indirecta, en la fase de entrenamiento. Esto provoca que el modelo tenga acceso a datos que no debería conocer, inflando artificialmente su desempeño.

Ejemplo:
Si se normalizan todos los datos (train + test) antes de dividirlos, el cálculo de la media y desviación estándar incluye información del conjunto de prueba, lo cual genera un modelo irrealmente "mejorado".

Este tipo de error lleva a resultados engañosos y modelos poco confiables en producción.

4.2 Data Preparation With Train and Test Sets

La forma correcta de preparar datos es:

1. Dividir el dataset en entrenamiento (train) y prueba (test).
2. Calcular las transformaciones únicamente con el conjunto de entrenamiento.
3. Aplicar esas transformaciones al conjunto de prueba.

De esta manera se garantiza que el modelo no conozca información futura y que su evaluación sea honesta.

Ejemplo en Python (escalado con MinMaxScaler):

```
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import MinMaxScaler

# División de datos
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3,
random_state=1)

# Escalado solo con train
scaler = MinMaxScaler()
scaler.fit(X_train)

# Aplicar transformaciones
X_train = scaler.transform(X_train)
X_test = scaler.transform(X_test)
```

Esto asegura que los parámetros de escalado no incluyan información del *test set*.

4.3 Data Preparation With k-fold Cross-Validation

En proyectos más avanzados se utiliza la validación cruzada (k-fold cross-validation). Aquí, el dataset se divide en k subconjuntos (folds), y cada uno funciona como conjunto de prueba en una iteración.

El desafío es evitar fugas en cada iteración. Para ello se recomienda:

- Realizar la preparación de datos dentro de cada fold.
- Usar pipelines, que encapsulan todos los pasos (preprocesamiento + modelo) y aseguran un flujo correcto.

Ejemplo en Python con Pipeline:

```
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression


# Definición del pipeline
steps = [('scaler', StandardScaler()), ('model', LogisticRegression())]
pipeline = Pipeline(steps=steps)
```

Con pipelines, cada vez que se entrena y valida un modelo, la preparación se ajusta solo con los datos de entrenamiento de ese fold, evitando fugas.

4.4 Further Reading

El autor recomienda profundizar en buenas prácticas de validación, bibliografía sobre fugas de datos y guías de librerías como *scikit-learn*, que implementan mecanismos automáticos para evitar errores en la preparación.

4.5 Summary

El *data leakage* es uno de los errores más comunes y dañinos en *Machine Learning*. Preparar los datos de manera ingenua —sin separar train y test— puede generar modelos con métricas engañosamente altas, pero que fracasan en el mundo real.

Para evitarlo, es necesario:

- Preparar datos solo con el conjunto de entrenamiento.
- Usar pipelines para garantizar un flujo seguro en cada validación.
- Aplicar correctamente la validación cruzada.

En conclusión, una preparación de datos responsable y ética garantiza que los modelos sean confiables, reproducibles y aplicables en contextos reales.

CONCLUSIÓN

La preparación de datos se ha evidenciado a lo largo de esta monografía como el pilar fundamental en cualquier proyecto de *Machine Learning*. Más allá de la elección de algoritmos o librerías, la calidad del modelo está directamente determinada por la calidad de los datos con los que se entrena. Este principio, que suele resumirse en la expresión *"garbage in, garbage out"*, reafirma que no es posible obtener predicciones confiables si la información de entrada no ha sido tratada de manera rigurosa y adecuada.

En primer lugar, se comprendió que la preparación de datos no es una tarea aislada, sino un proceso transversal que conecta la definición del problema con la construcción y evaluación de modelos. Representa entre el 70% y 80% del esfuerzo en la práctica, lo que demuestra que el verdadero reto en la ciencia de datos no reside únicamente en los algoritmos, sino en el tratamiento de la información.

Asimismo, se destacó la importancia de convertir datos crudos en representaciones numéricas consistentes, ya que los algoritmos solo trabajan con números. Este paso es imprescindible cuando los datos provienen en formatos heterogéneos como texto, categorías, fechas o imágenes. Además, se hizo énfasis en la necesidad de resolver problemas habituales en datasets reales, tales como valores faltantes, duplicados, outliers y escalas desiguales, que de no atenderse comprometen el aprendizaje del modelo.

Por otro lado, se realizó un recorrido por las principales técnicas de preparación, como limpieza, selección de características, transformaciones, ingeniería de variables y reducción de dimensionalidad. Cada una de ellas responde a desafíos específicos: garantizar la coherencia de los datos, reducir ruido, resaltar patrones relevantes y simplificar la complejidad de los modelos. En este sentido, se concluye que la preparación

de datos no es un procedimiento mecánico, sino un proceso creativo y estratégico que requiere conocimiento técnico y comprensión del dominio de aplicación.

Un aspecto clave analizado fue la prevención de la data leakage, considerado uno de los errores más perjudiciales en *Machine Learning*. La preparación de datos debe hacerse de forma ética y responsable, evitando fugas de información del conjunto de prueba hacia el entrenamiento. Para ello, se recomienda preparar datos únicamente con el conjunto de entrenamiento, aplicar correctamente la validación cruzada y utilizar pipelines que garanticen flujos reproducibles. Estas prácticas no solo fortalecen la validez de los modelos, sino que también aseguran su aplicabilidad en entornos reales.

En conclusión, la preparación de datos debe entenderse como una fase estratégica que determina la confiabilidad y el éxito de los proyectos de *Machine Learning*. Si bien los algoritmos son importantes, el verdadero valor de la ciencia de datos radica en el trabajo previo con la información, que permite construir modelos robustos, éticos y capaces de responder a problemas complejos de la realidad.

REFERENCIAS BIBLIOGRAFICAS

Brownlee, J. (2019). *Data Preparation for Machine Learning: Data Cleaning, Feature Selection, and Data Transforms in Python*. Machine Learning Mastery.

Kuhn, M., & Johnson, K. (2013). *Applied Predictive Modeling*. Springer. https://doi.org/10.1007/978-1-4614-6849-3

Kuhn, M., & Johnson, K. (2019). *Feature Engineering and Selection: A Practical Approach for Predictive Models*. CRC Press.

Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow* (2nd ed.). O'Reilly Media.

Han, J., Pei, J., & Kamber, M. (2011). *Data Mining: Concepts and Techniques* (3rd ed.). Morgan Kaufmann.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2021). *An Introduction to Statistical Learning: with Applications in R* (2nd ed.). Springer. https://doi.org/10.1007/978-1-0716-1418-1

Zhang, Z. (2016). Missing data imputation: focusing on single imputation. *Annals of Translational Medicine, 4*(1), 9. https://doi.org/10.3978/j.issn.2305-5839.2015.12.38

Wikipedia contributors. (2023). *Data preparation*. In *Wikipedia*. https://en.wikipedia.org/wiki/Data_preparation