

Curso: Inteligencia Artificial

Unidad 1: Conceptos fundamentales de inteligencia artificial

Sesión 7: Introducción al lenguaje Python:

Tema Adicional: Minería de Datos

Docente: Carlos R. P. Tovar

INICIO

Objetivo de la sesión

Al finalizar la sesión, el alumno será capaz de:

- Comprender qué es la minería de datos y sus fases principales.
- Reconocer técnicas y algoritmos más usados.
- Aplicar herramientas básicas de minería de datos.
- Evaluar la utilidad de la minería en diferentes áreas.



Introducción

- **Frase:**

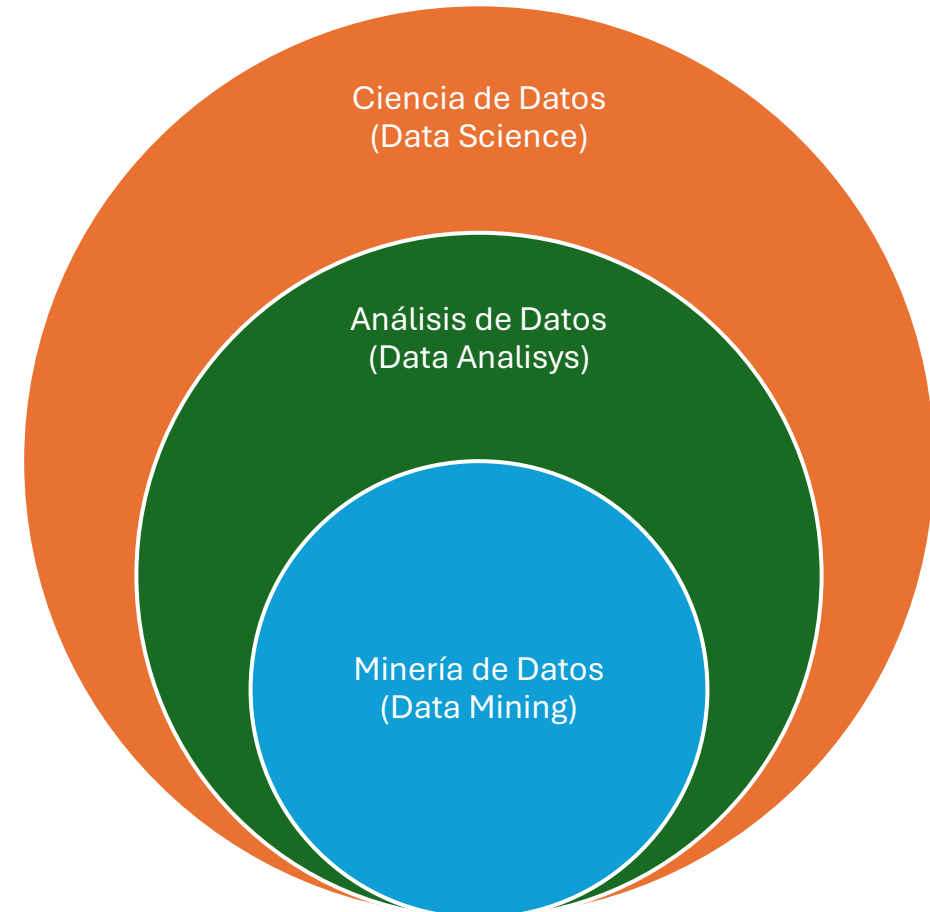
“Without data, you’re just another person with an opinion.”

– W. Edwards Deming

UTILIDAD

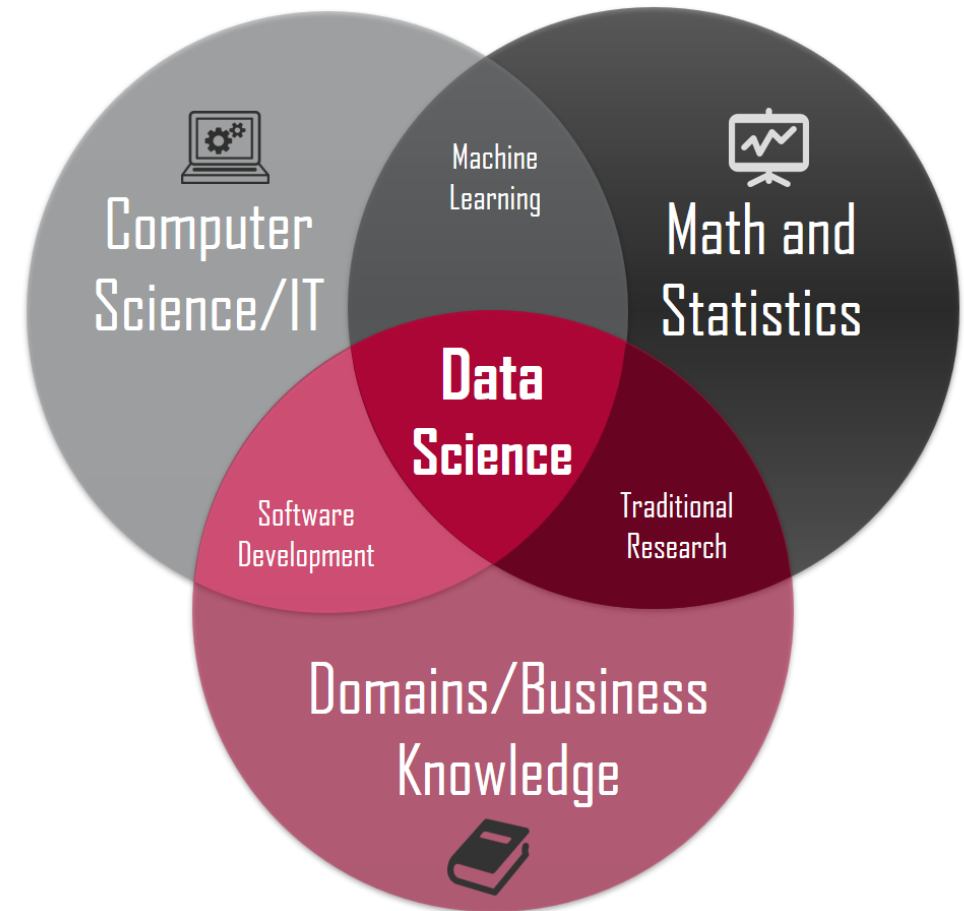
¿Qué es la minería de datos?

- Extracción de patrones y conocimiento útil de grandes volúmenes de datos.
- Se apoya en estadística, IA y aprendizaje automático.
- Va más allá de recolectar: busca entender y predecir.



Diferencia con otras disciplinas

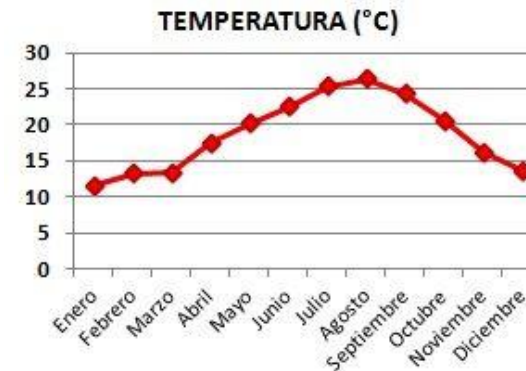
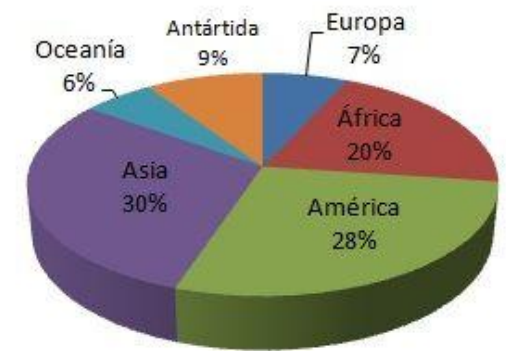
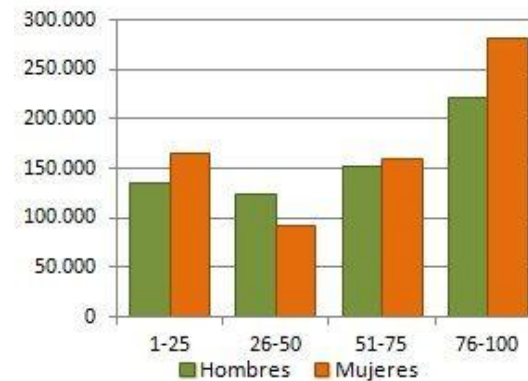
- **Estadística:** describe datos.
- **Machine Learning:** entrena modelos predictivos.
- **Minería de Datos:** integra técnicas para descubrimiento de conocimiento.



Esta foto de Autor desconocido está bajo licencia [CC BY-SA](#)

Aplicaciones reales

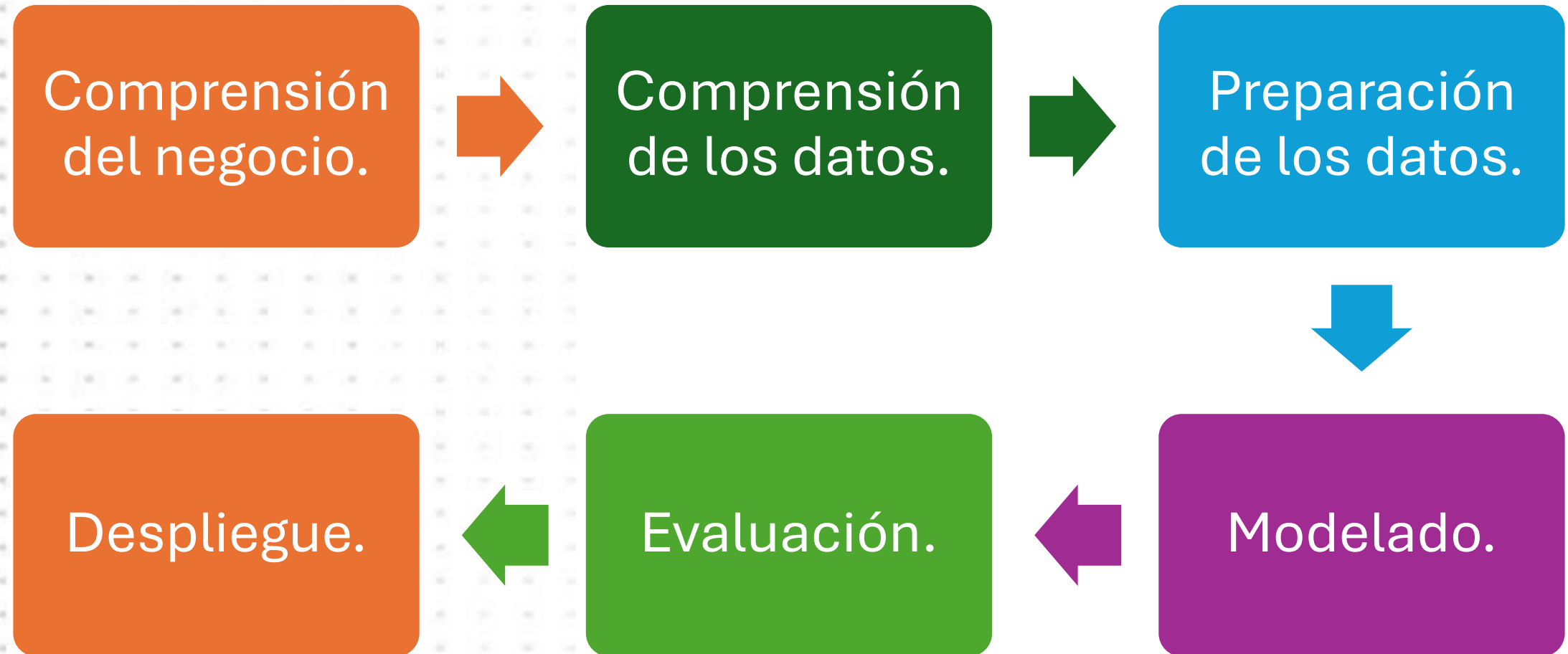
- Marketing y segmentación de clientes.
- Detección de fraudes.
- Predicción en medicina.
- Análisis de redes sociales.
- Sistemas de recomendación.



Esta foto de Autor desconocido está bajo licencia [CC BY-NC-ND](#)

TRANSFORMACIÓN

Proceso de Minería de Datos (CRISP-DM)



Comprensión del negocio

- Identificar problemas y objetivos.
- Ejemplo: ¿Qué clientes dejarán el servicio el próximo mes?

Comprensión y preparación de datos

- Recolección, limpieza y transformación.
- Manejo de valores faltantes.
- Normalización y reducción de dimensionalidad.

Técnicas principales

- Clasificación.
- Regresión.
- Agrupamiento (clustering).
- Reglas de asociación.
- Detección de anomalías.

Clasificación

- Asignar etiquetas a objetos.
- Algoritmos: Árboles de decisión, SVM, Redes Neuronales.
- Ejemplo: Diagnóstico médico.

Regresión

- Predecir valores numéricos.
- Algoritmos: Regresión lineal, regresión polinómica.
- Ejemplo: Predicción del precio de viviendas.

Clustering

- Agrupar datos sin etiquetas.
- Algoritmos: K-Means, DBSCAN, Hierárquico.
- Ejemplo: Segmentación de clientes.

Reglas de asociación

- Encontrar patrones frecuentes en datos.
- Algoritmo: Apriori, FP-Growth.
- Ejemplo: “Quien compra pan también compra leche.”

Detección de anomalías

- Identificación de datos inusuales.
- Aplicaciones: Fraude en tarjetas de crédito, fallas en sensores.

Herramientas de minería de datos

- Python: scikit-learn, pandas, Orange, Weka.
- R: caret, mlr.
- Plataformas empresariales: RapidMiner, SAS.

PRACTICA

Ejemplo práctico

- Dataset: Alquiler de Viviendas por AIRBNB
- Objetivo: Clasificar viviendas que van a ser alquiladas y cual determinar cual sera el precio para eso.
- Herramienta: Python con mlxtend o Weka.

Ejemplo práctico

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, confusion_matrix
import matplotlib.pyplot as plt
import seaborn as sns
```

1. Generar datos simulados

```
np.random.seed(42)
n = 500
precio_noche = np.random.randint(30, 500, n)
calificacion = np.round(np.random.uniform(2.5, 5.0, n), 2)
```

Ejemplo práctico

```
# Lógica para determinar si fue arrendada (simplificada)
# Supongamos que propiedades baratas y bien calificadas tienen más chance
prob_arrendada = 1 / (1 + np.exp(-0.02*(300 - precio_noche) + 2*(calificacion - 3.5)))
arrendada = np.random.binomial(1, prob_arrendada)
```

```
# Crear DataFrame
df = pd.DataFrame({
    'precio_noche': precio_noche,
    'calificacion': calificacion,
    'arrendada': arrendada
})
```

Ejemplo práctico

2. Dividir en conjuntos de entrenamiento y prueba

```
X = df[['precio_noche', 'calificacion']]
```

```
y = df['arrendada']
```

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

3. Entrenar modelo

```
clf = RandomForestClassifier(n_estimators=100, random_state=42)
```

```
clf.fit(X_train, y_train)
```

4. Evaluación

```
y_pred = clf.predict(X_test)
```

```
print(confusion_matrix(y_test, y_pred))
```

```
print(classification_report(y_test, y_pred))
```

Ejemplo práctico

5. Visualización del espacio de decisión

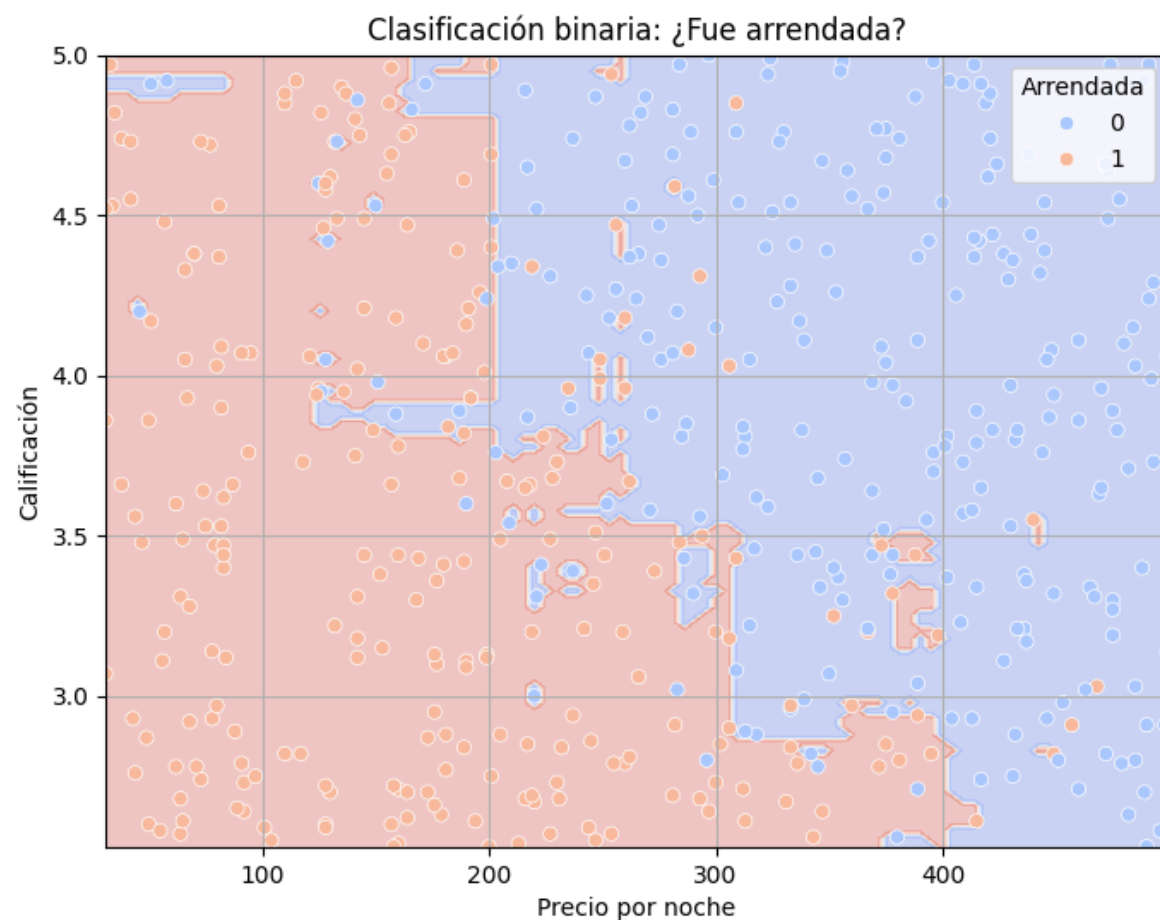
```
xx, yy = np.meshgrid(
    np.linspace(X['precio_noche'].min(),
X['precio_noche'].max(), 100),
    np.linspace(X['calificacion'].min(),
X['calificacion'].max(), 100)
)
Z = clf.predict(np.c_[xx.ravel(), yy.ravel()])
Z = Z.reshape(xx.shape)
```

```
plt.figure(figsize=(8, 6))
plt.contourf(xx, yy, Z, alpha=0.3,
cmap='coolwarm')
sns.scatterplot(data=df,
x='precio_noche', y='calificacion',
hue='arrendada', palette='coolwarm')
plt.title('Clasificación binaria: ¿Fue
arrendada?')
plt.xlabel('Precio por noche')
plt.ylabel('Calificación')
plt.legend(title='Arrendada')
plt.grid(True)
plt.show()
```

Ejemplo práctico: Resultados

Datos	precision	recall	f1-score	support
0	0.76	0.82	0.79	50
1	0.80	0.74	0.77	50
accuracy			0.78	100
macro avg	0.78	0.78	0.78	100
weighted avg	0.78	0.78	0.78	100

	Verdadero	Falso
Alquilada	41	9
No Alquilada	13	37



PRACTICA

Actividades

- Implementar un clustering con K-Means en Python.
- Aplicar reglas de asociación con datos de supermercado.
- Comparar resultados con distintos parámetros.

CIERRE

Conclusiones

- La minería de datos transforma datos en conocimiento.
- CRISP-DM guía el proceso.
- Diferentes técnicas sirven según el tipo de problema.
- Aplicaciones en múltiples industrias



**Universidad
Tecnológica
del Perú**