Curso: Análisis de datos y visualización con Python

Parte del **Certificado Microsoft Python Developer Professional**, compuesto por **6 cursos**:

1. Fundamentos de programación en Python
2. Análisis de datos y visualización con Python
3. Automatización y scripting con Python
4. Desarrollo Web con Python
5. Técnicas avanzadas de desarrollo con Python
6. Desarrollo de proyectos en Python

Objetivo del curso

- Profundizar en el **análisis de datos** con Python.
- Aprender **limpieza, preparación, exploración y manipulación de datos**.
- Convertir datos en **visualizaciones e historias comprensibles** para equipos y stakeholders.
- Introducirse en **IA generativa** y **aprendizaje automático básico**.

Módulos del curso

1. **Introducción al Análisis de datos**
    - Conceptos, procesos y herramientas.
    - Uso de Jupyter Notebook.
2. **Procesamiento y manipulación de datos**

    - Uso de pandas.
    - Manejo de valores nulos y atípicos.
    - Conversión de tipos de datos.
3. **Visualización de datos**
    - Narración de datos.
    - Bibliotecas: Matplotlib, Plotly, Bokeh, Apache Superset.
    - Ejercicios con visualizaciones de datos musicales.
4. **Introducción a la IA generativa**
    - Conceptos de LLM y transformadores.
    - Diferencia con otros tipos de IA.
    - Ejemplos prácticos y consideraciones éticas.
    - Generación de datos sintéticos con GAN.
5. **Introducción al Aprendizaje automático**
    - Conceptos básicos de ML.
    - Regresión lineal y logística.
    - Evaluación de modelos (métricas de rendimiento).
    - Laboratorio con Scikit-Learn: predicción de precios de viviendas.

Metodología de aprendizaje

- **Videos, demostraciones, lecturas, ejercicios prácticos, proyectos y cuestionarios**.
- Enfoque en la **aplicación práctica** para reforzar conceptos.

En resumen: Este curso lleva a un programador de Python desde el análisis y limpieza de datos, hasta la creación de visualizaciones, exploración de IA generativa y primeros pasos en Machine Learning, con un enfoque muy práctico y orientado a proyectos.

El análisis de datos y su importancia

¿Qué es el análisis de datos?

- Proceso de **transformar datos sin procesar** en **información significativa y útil**.
- Incluye: **organizar, limpiar, interpretar y detectar patrones** en grandes volúmenes de información.
- Actúa como un **puente entre datos y conocimiento**.

Beneficios principales

1. **Mejores decisiones** → basadas en evidencia, no solo en intuición.
2. **Resolución de problemas complejos** → en negocios, salud, gobierno, etc.
3. **Impulso a la innovación** → detectar tendencias, oportunidades y anticipar desafíos.

Aplicaciones en distintos sectores

- **Empresas**: entender clientes, personalizar marketing, optimizar precios, mejorar eficiencia.
- **Salud**: analizar datos de pacientes, diseñar mejores tratamientos, reducir costos y mejorar la atención.
- **Educación**: personalizar aprendizaje, detectar dificultades de estudiantes, medir eficacia docente.
- **Ciencia**: analizar experimentos, validar hipótesis, avanzar en genómica, astronomía, clima, etc.

- **Deporte**: mejorar estrategias, entrenamientos y selección de jugadores.
- **Vida diaria**: apps de salud, relojes inteligentes, dispositivos domésticos conectados.

Rol del analista de datos

- Profesional encargado de **recopilar, limpiar e interpretar datos**.
- Transforma información compleja en conocimiento procesable.
- Muy demandado en diversas industrias.

Conclusión

El **análisis de datos** es una herramienta esencial para:

- Tomar decisiones inteligentes.
- Resolver problemas complejos.
- Generar innovación.
- Mejorar la vida diaria y avanzar en la ciencia.

En definitiva, es **clave para el futuro**, y una carrera atractiva para quienes disfrutan resolver problemas y trabajar con números.

Pregunta

¿Cuál es la esencia del Análisis de datos? Seleccione la mejor respuesta.

- (●) Transformación de datos brutos en información útil y procesable
- ( ) Creación de cuadros y gráficos visualmente atractivos.
- ( ) Predecir tendencias futuras con absoluta certeza.
- ( ) Recopilación de grandes cantidades de datos en bruto.

✓ **Correcto**
Correcto El Análisis de datos consiste fundamentalmente en convertir datos brutos en información que pueda orientar la toma de decisiones y la resolución de problemas.

Fundamentos del Análisis de Datos

Definición

El **análisis de datos** es el proceso de **examinar datos en bruto** para descubrir patrones, tendencias y relaciones. Incluye:

- Recopilación
- Limpieza
- Transformación
- Modelización de datos

Su propósito es **responder preguntas, resolver problemas y apoyar la toma de decisiones informadas**.

Importancia

- Crece exponencialmente en todos los sectores.
- Brinda **ventaja competitiva, eficiencia e innovación**.
- Permite **traducir datos en estrategias aplicables**.

Ejemplos de uso

- **Retail**: identificar tendencias de clientes y personalizar campañas.
- **Salud**: mejorar tratamientos, reducir costos, personalizar atención.
- **General**: presente en finanzas, educación, medio ambiente, tecnología y gobierno.

Aplicaciones por sectores

- **Empresas**: análisis de clientes, optimización de precios, eficiencia en cadena de suministro, uso de Power BI.
- **Sanidad**: identificar riesgos, personalizar planes, reducir reingresos hospitalarios.
- **Finanzas**: gestión de riesgos, detección de fraude, modelos predictivos de inversión.
- **Gobierno**: mejor asignación de recursos, seguridad vial, políticas públicas basadas en datos.
- **Tecnología**: IA, machine learning, recomendaciones personalizadas, autos autónomos.

Rol del Analista de Datos

Responsabilidades principales:

1. **Recopilación y limpieza de datos** (Pandas, APIs, web scraping).
2. **Análisis exploratorio (EDA)** con estadísticas y visualización (Jupyter, Matplotlib).
3. **Modelización estadística** (regresión, hipótesis, ML con Scikit-learn).
4. **Visualización y comunicación** (gráficos, dashboards en Power BI).

Retos y ética

- Riesgo de **uso indebido de datos** → privacidad, sesgos, discriminación.
- Temor a la **automatización y pérdida de empleos** → pero también genera nuevas oportunidades.
- Responsabilidad de garantizar un **uso justo, ético y transparente** de los datos.

Tendencias y expansión

- **Big Data**: manejo de terabytes/petabytes con Spark y Hadoop.

- **Machine Learning**: modelos predictivos y automatización.
- **Narración de datos**: contar historias claras y persuasivas a partir de la información.

Conclusión

El análisis de datos no trata solo de números, sino de **contar historias, descubrir verdades y generar impacto positivo**.
Dominar sus fundamentos y mantenerse actualizado en nuevas tecnologías permite a profesionales y organizaciones **tomar decisiones confiables, resolver problemas y construir un futuro basado en conocimiento**.

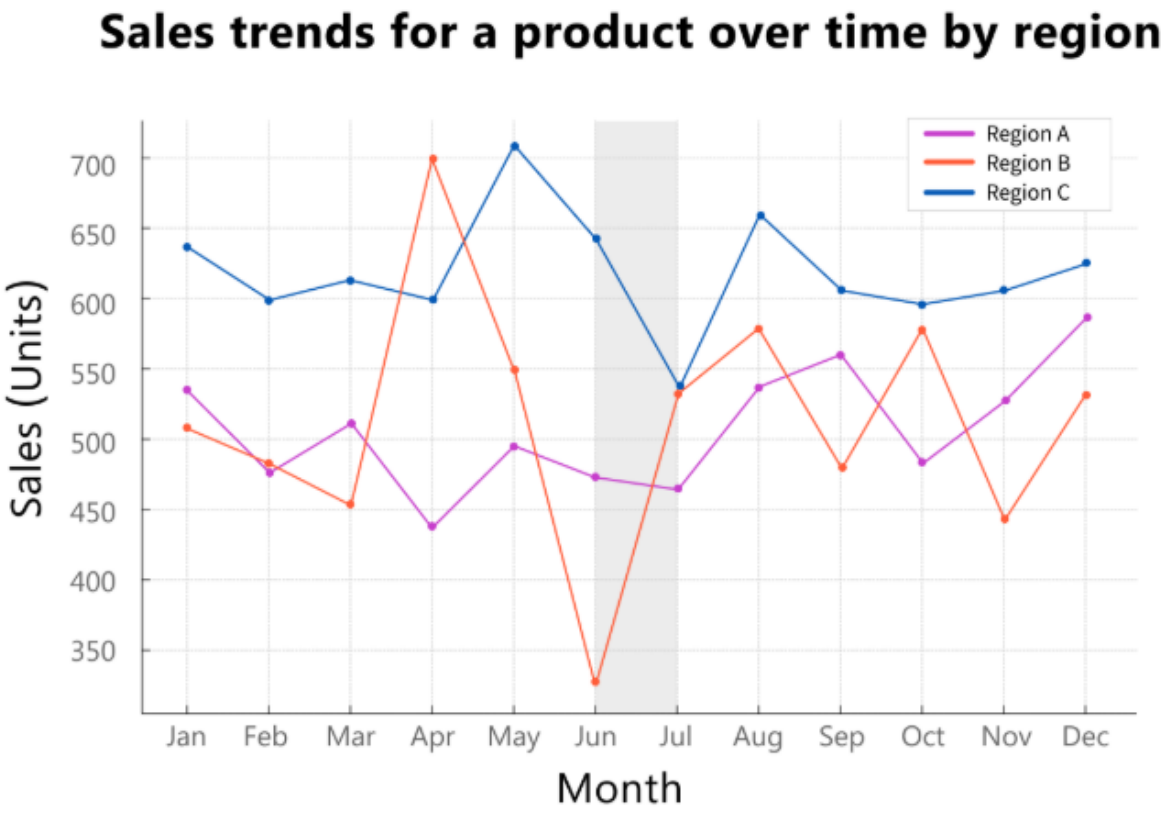
Análisis de datos vs Ciencia de datos (enfoque en análisis de datos)

¿Qué es el Análisis de datos?

- Proceso de **transformar datos en bruto en información práctica**.
- Busca **identificar patrones, anomalías y correlaciones** que sirvan para mejorar la toma de decisiones.
- Se centra más en **iluminar el pasado y explicar el presente** (ej.: por qué bajaron las ventas en cierta región).

Herramientas principales

- **Pandas & NumPy** → manipulación y cálculo eficiente de datos.
- **Power BI** → visualización interactiva de datos (gráficos, dashboards).
- **SQL** → extracción de información específica desde bases de datos relacionales.



Habilidades clave del Analista de datos

1. **Curiosidad** → buscar patrones y hacer las preguntas correctas.
2. **Escepticismo** → validar hipótesis y evitar conclusiones erróneas.
3. **Comunicación** → transmitir resultados de forma clara a públicos técnicos y no técnicos.

Ejemplo práctico (e-commerce)

- Un analista descubre que ciertos estilos de ropa tienen bajo rendimiento en regiones específicas.
- Acciones derivadas:
    - Ajustar inventario.
    - Ofrecer descuentos estratégicos.

    - Adaptar campañas de marketing a preferencias locales.

- Resultado: **más ventas, menos desperdicio y mayor satisfacción del cliente**.

Conclusión:
El **análisis de datos** es esencialmente **interpretar y comunicar lo que los datos nos dicen** para tomar decisiones informadas en el presente. Aunque está relacionado con la ciencia de datos, su foco está más en el **"qué pasó y por qué"**, en lugar de construir modelos predictivos o sistemas complejos (más propios de la ciencia de datos).

Ciencia de datos: Predecir y automatizar

Definición

- Campo **multidisciplinar** que combina estadística, programación y computación.
- Busca **predecir el futuro, descubrir patrones ocultos y automatizar decisiones**.

Herramientas clave

- **Machine Learning**: Scikit-learn, PyTorch (modelos predictivos, redes neuronales, aprendizaje profundo).
- **Software estadístico**: R, SAS (validación, pruebas de hipótesis).
- **Nube**: Azure (procesamiento a gran escala, despliegue de modelos).

Habilidades del científico de datos

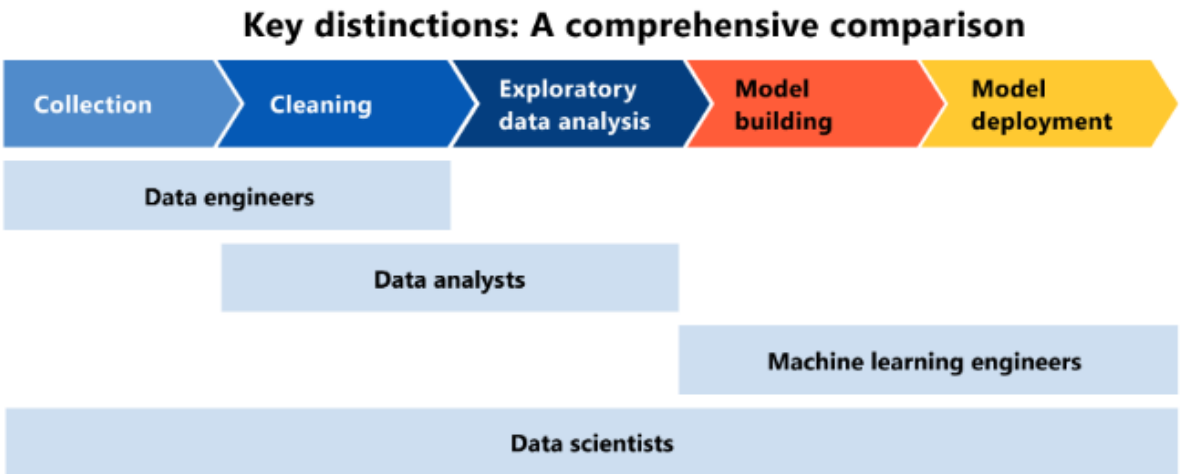
- **Experimentación**: probar, ajustar y optimizar modelos.
- **Matemáticas y estadística**: base sólida en álgebra lineal, probabilidad, inferencia.
- **Programación**: Python o R para limpieza, modelado y automatización.

Ejemplo

En **sanidad**, un científico de datos puede crear modelos predictivos para anticipar enfermedades a partir de historiales médicos, datos genómicos e imágenes, permitiendo **diagnósticos tempranos y tratamientos personalizados**.

Diferencias clave

Aspecto	Análisis de datos	Ciencia de datos
Objetivo	Describir y explicar lo que pasó.	Predecir y automatizar procesos.
Preguntas típicas	¿Qué ocurrió? ¿Por qué?	¿Qué ocurrirá? ¿Cómo optimizarlo?
Enfoque	Limpieza, manipulación, visualización.	Modelos predictivos, algoritmos, ML.
Herramientas	Pandas, NumPy, SQL, Power BI, Excel.	Scikit-learn, PyTorch, R, Azure, SAS.
Resultados	Informes, dashboards, insights prácticos.	Algoritmos, modelos de predicción, sistemas inteligentes.
Aplicaciones	BI, marketing, finanzas, reporting.	Recomendadores, fraude, medicina personalizada, coches autónomos.



Conexión entre ambos

- Muchos **científicos de datos comienzan como analistas**.
- La transición implica: más estadística, más programación y aprendizaje automático.

Elección de carrera

- **Análisis de datos**: ideal si disfrutas **descubriendo patrones y comunicando insights** para decisiones empresariales.
- **Ciencia de datos**: ideal si te motiva **crear sistemas inteligentes y predictivos** que transformen industrias.

Conclusión: Ambos campos son altamente demandados y complementarios. El análisis de datos ilumina el pasado y el presente, mientras que la ciencia de datos busca **predecir el futuro y crear automatización**. En ambos, **Python es la herramienta fundamental**.

Proceso del análisis de datos

- **Problema inicial**: María está abrumada con datos de ventas y clientes; necesita entender por qué bajaron las ventas en la región del Medio Oeste.
- **Consejo de Ben**: El análisis de datos es como un caso detectivesco; se sigue un proceso paso a paso.

Pasos del proceso de análisis de datos

1. **Definir la pregunta** → identificar qué se quiere responder (ej.: "¿Por qué bajaron las ventas?").
2. **Recolección de datos** → reunir información relevante.
3. **Limpieza de datos** → eliminar errores e inconsistencias para asegurar fiabilidad.
4. **Análisis** → usar estadísticas y visualizaciones para descubrir patrones y causas.

5. **Comunicación de resultados** → presentar hallazgos claros y útiles al equipo.

Herramientas mencionadas

- **Python + bibliotecas**: Pandas, NumPy (organización y cálculos).
- **Visualización**: Matplotlib, Apache Superset (gráficos, dashboards).

Mensaje final

El análisis de datos puede parecer abrumador al principio, pero con práctica se vuelve más fácil y revelador. Es un **viaje de aprendizaje continuo**.

Conceptos clave en Análisis de datos

1. Población y muestra

- **Población**: conjunto completo que se desea estudiar (ejemplo: todos los clientes de una librería en línea).
- **Muestra**: subconjunto representativo de la población, usado porque analizar a todos es costoso o impráctico.
- **Importancia**: una muestra bien elegida debe reflejar la diversidad de la población para evitar sesgos.

Ejemplo: tomar 1.000 clientes de una librería online para predecir preferencias de toda la base de clientes.

2. Variables: lo que medimos y observamos

Los atributos o características que se estudian en los datos.

- **Numéricas (cuantitativas)**
    - **Discretas**: valores enteros y contables (ej.: número de compras, visitas a un sitio).

    - **Continuas**: valores en un rango (ej.: tiempo en una web, temperatura, peso).

- **Categóricas (cualitativas)**
    - **Nominales**: categorías sin orden (ej.: colores, marcas, tipos de cocina).
    - **Ordinales**: categorías con jerarquía u orden (ej.: nivel educativo, satisfacción del cliente).

Ejemplo: edad, gasto promedio, géneros favoritos de libros, reseñas.

3. Uso en Python (con Pandas)

- **DataFrame**: tabla con filas (observaciones) y columnas (variables).
- **Variables numéricas**: manejadas como `int` o `float`, permiten cálculos, estadísticas y visualizaciones (histogramas, dispersión).
- **Variables categóricas**: representadas como `str` o tipo `category` en pandas → optimizan memoria y facilitan análisis como conteo de frecuencias o gráficos de barras.

En resumen:
El análisis de datos se apoya en **poblaciones, muestras y variables** para estudiar fenómenos de manera representativa. Entender los **tipos de variables** (numéricas y categóricas) es esencial porque determina qué métodos estadísticos y visualizaciones se deben usar. Con **Python y Pandas**, este trabajo se vuelve eficiente y escalable.

Tipos de variables y datos en el análisis de datos

1. Importancia de los tipos de variables

- Conocer si una variable es **numérica** o **categórica** guía las técnicas de análisis y visualización.
    - *Ejemplo 1*: gasto de clientes (numérica) → calcular medias, histogramas, identificar grandes compradores.

- *Ejemplo 2*: satisfacción del cliente (categórica) → contar frecuencias, calcular porcentajes, gráficos de barras.

2. Python en acción (ejemplo con Pandas)

```
import pandas as pd

# Datos de ejemplo
data = {'age': [25, 30, 35, 40],
        'gender': ['male', 'female', 'male', 'female'],
        'income': [50000, 60000, 75000, 55000]}

# Crear DataFrame
df = pd.DataFrame(data)

# Convertir 'gender' a variable categórica
df['gender'] = df['gender'].astype('category')

# Calcular ingreso medio
average_income = df['income'].mean()

# Contar hombres y mujeres
gender_counts = df['gender'].value_counts()

print(average_income)
print(gender_counts)
```

Salida esperada:

```
60000.0
male      2
female    2
Name: gender, dtype: int64
```

Este ejemplo muestra cómo Pandas maneja variables numéricas (`age`, `income`) y categóricas (`gender`) de manera eficiente.

3. Tipos de datos en Python

- **Numéricos**: `int` (enteros), `float` (decimales).
- **Texto**: `str`.
- **Booleanos**: `bool` (True/False).
- **Colecciones**: `list` (listas ordenadas), `dict` (pares clave-valor).

Con **NumPy** y **Pandas**, se usan estructuras especializadas como **arrays** y **DataFrames** que permiten cálculos, filtrado, agrupación y visualización más eficientes.

4. Retos en el análisis de datos

- **Definir población y muestra representativa** → riesgo de sesgos si no reflejan bien al conjunto.
- **Errores de medición y sesgos** → respuestas autodeclaradas pueden ser poco fiables.
- **Datos faltantes y valores atípicos** → deben tratarse con:
    - **Imputación** (relleno de valores faltantes).
    - **Métodos estadísticos robustos** menos sensibles a outliers.

En conclusión:
Dominar los conceptos de **población, muestra, variables y tipos de datos**, junto con las herramientas de **Python (Pandas, NumPy)**, proporciona la base para un análisis fiable y útil, aunque siempre con cuidado frente a sesgos, datos faltantes y valores extremos.

Ética y privacidad de los datos

1. Ética de los datos

- Va más allá del cumplimiento legal: implica actuar con **responsabilidad**, **equidad** y **transparencia**.

- Busca proteger la **privacidad** y la **dignidad** de las personas.
- Requiere evaluar el impacto de nuestras acciones en la sociedad.

2. Principales consideraciones éticas

- **Consentimiento y transparencia**: informar claramente a las personas sobre el uso de sus datos y permitirles optar por no participar.
- **Seguridad y protección**: usar cifrado, controles de acceso y copias de seguridad para evitar filtraciones o accesos no autorizados.
- **Sesgo y equidad**: vigilar los algoritmos para evitar discriminación y resultados injustos.

3. Problemas clave de privacidad

- **Minimización de datos**: recolectar solo lo estrictamente necesario.
- **Almacenamiento seguro**: proteger información sensible con técnicas sólidas de seguridad.
- **Acceso de terceros**: compartir datos únicamente con acuerdos claros y responsables.
- **Anonimización y desidentificación**: eliminar identificadores personales para proteger la identidad de los usuarios.

4. Buenas prácticas

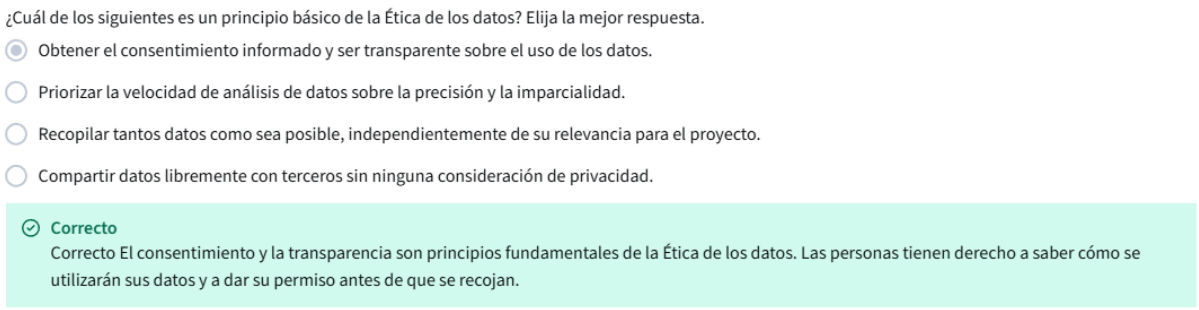
- **Privacidad desde el diseño**: integrar la privacidad desde el inicio de los sistemas y procesos.
- **Gobernanza y cumplimiento**: establecer políticas claras que sigan leyes y estándares del sector.
- **Evaluaciones de impacto ético**: analizar riesgos y consecuencias antes de implementar soluciones.

- **Educación continua**: mantenerse actualizado sobre los desafíos emergentes en ética y privacidad.

Conclusión

La ética y la privacidad en el manejo de datos son pilares de la innovación responsable. Los desarrolladores de Python tienen la oportunidad de construir un futuro digital donde los datos se usen para el bien común, protegiendo al mismo tiempo los **derechos individuales** y fomentando la **confianza**.

Pregunta



Gobierno de Datos en Python

1. Gobierno de datos

- Es un **marco de normas y regulaciones** para administrar los datos durante todo su ciclo de vida.
- Asegura: **precisión, coherencia, integridad, seguridad** y **cumplimiento legal**.
- Importante para: tomar decisiones informadas, mitigar riesgos y mantener la reputación de una organización.

2. Principales regulaciones

- **HIPAA (EE.UU.)**: protege información médica de pacientes.
- **FERPA (EE.UU.)**: protege registros educativos de estudiantes.

- **GDPR (UE)**: regula la recolección y uso de datos personales de residentes europeos, aplicable globalmente.
- **COPPA (EE.UU.)**: protege la privacidad de niños menores de 13 años en línea (consentimiento de padres obligatorio).
- **SOX (EE.UU.)**: regula registros financieros y controles internos de empresas para prevenir fraudes.

3. Responsabilidades del programador de Python

- **Seguridad de datos**:
    - Cifrado (bibliotecas como *cryptography*).
    - Autenticación segura y multifactor.
    - Controles de acceso.
- **Calidad de datos**:
    - Validación de entradas.
    - Limpieza y preprocesamiento (ej. con *pandas*).
    - Eliminación de duplicados y manejo de valores faltantes.
- **Respeto a la privacidad**:
    - Minimizar recolección de datos.
    - Avisos de privacidad claros y consentimiento informado.
    - Técnicas de anonimización o seudonimización.
- **Retención de datos**:
    - Definir cuánto tiempo se almacenan.
    - Automatizar eliminación o archivado según regulaciones.
- **Colaboración**:
    - Trabajar con administradores y dueños de datos.

- Participar en discusiones sobre políticas de gobernanza.
- Educar a otros sobre privacidad y seguridad.

Conclusión

La **gobernanza de datos** no es solo cumplir reglas, sino manejar la información de forma **responsable y ética**. Como programador de Python, aplicar estos principios ayuda a **proteger usuarios**, **cumplir la ley** y **fortalecer la confianza** en las aplicaciones que desarrolles.

Cuestionario: Desvelar el análisis de datos

1. Estás trabajando en un proyecto que implica recopilar y almacenar datos confidenciales de clientes, como nombres, direcciones e historial de compras. ¿Por qué es crucial cifrar estos datos? Selecciona la mejor respuesta.

- (●) Para proteger los datos de accesos no autorizados.
- ( ) Para reducir el coste del almacenamiento de datos.
- ( ) Para reducir el espacio de almacenamiento necesario.
- ( ) Mejorar la precisión de los análisis de datos.

Correcto
Correcto La encriptación de datos sensibles ayuda a protegerlos de accesos no autorizados y salvaguarda la privacidad del cliente.

2. Es gestor de productos en una empresa de Redes sociales que opera en todo el mundo. Es responsable de garantizar la privacidad de los datos. Necesita conocer una normativa que obliga a las organizaciones a obtener el consentimiento explícito de las personas antes de recopilar sus datos personales. ¿De qué normativa se trata? Seleccione la mejor respuesta.

- (●) GDPR
- ( ) SOX
- ( ) HIPAA
- ( ) FERPA

Correcto
Correcto El GDPR exige el consentimiento explícito antes de recopilar datos personales.

3. Está investigando posibles salidas profesionales en el campo del Análisis de datos. Desea identificar sectores con oportunidades importantes en este campo. ¿Cuál de los siguientes sectores ofrece oportunidades profesionales importantes en el análisis de datos? Seleccione todas las que corresponda.

- [x] Sanidad

Correcto
Correcto La sanidad es uno de los muchos sectores en los que el análisis de datos es crucial.

- [ ] Entrenamiento personal
- [x] Agricultura

Correcto
Correcto La agricultura también se beneficia del Análisis de datos.

- [x] Venta al por menor

Correcto
Correcto El Análisis de datos tiene una gran repercusión en el sector minorista.

4. Está formando un equipo para analizar las opiniones de los clientes sobre el lanzamiento de un nuevo producto. Antes de poder analizar los datos, hay que limpiarlos y organizarlos. ¿A quién asignaría normalmente esta tarea? Seleccione la mejor respuesta.

- ( ) Estadístico
- ( ) Científico de datos
- ( ) Analista de negocios
- (•) Ingeniero de datos

Correcto
Correcto Los ingenieros de datos suelen encargarse de limpiar y organizar los datos antes de analizarlos.

5. Usted es un desarrollador que comienza un nuevo proyecto de Análisis de datos. ¿Cuál de los siguientes pasos debe completar primero? Seleccione la mejor respuesta.

- ( ) Limpieza de datos
- ( ) Pruebas de hipótesis
- ( ) Visualización de datos
- (•) Recogida de datos

Correcto
Correcto La recopilación de datos suele ser el primer paso de un flujo de trabajo de análisis de datos, ya que sienta las bases para el análisis posterior.

6. Usted es un analista financiero que estudia las ventajas del análisis de datos en su sector. ¿De cuál de las siguientes formas puede beneficiar el Análisis de datos a las instituciones financieras? Seleccione todas las que correspondan.

- [x] Mejora de la gestión de riesgos.

Correcto
Correcto El Análisis de datos mejora considerablemente la gestión de riesgos en el sector financiero.

- [ ] Rentabilidad de la inversión garantizada.
- [x] Mejor detección del fraude.

Correcto
Correcto La detección del fraude es una de las principales ventajas del análisis de datos en el sector financiero.

- [x] Cumplimiento de la normativa.

Correcto
Correcto El Análisis de datos ayuda a garantizar el cumplimiento de la normativa mediante el control y la notificación de las actividades financieras.

7. Estás aprendiendo los conceptos básicos del análisis de datos y los distintos tipos de datos que se utilizan en este campo. ¿Cuáles de los siguientes tipos de datos se utilizan habitualmente en el Análisis de datos? Seleccione todos los que corresponda.

- [x] Ordinal

Correcto
Correcto Los Tipos de datos ordinales también se utilizan en el Análisis de datos.

- [x] Categórica

Correcto
Correcto Los tipos de datos categóricos se utilizan con frecuencia en el análisis de datos.

- [x] Numérico

Correcto
Correcto Los tipos de datos numéricos se utilizan habitualmente en el Análisis de datos.

- [ ] Temporal

Configuración del entorno para el análisis de datos

1. **Editor de código**:
    - Elegir un editor confiable, como **PyCharm**.
    - Facilita la **finalización automática** y la **depuración**, ahorrando tiempo y errores.
2. **Cuadernos interactivos**:
    - **Jupyter Notebooks** permiten ejecutar código por fragmentos, visualizar resultados y añadir explicaciones.
    - Actúa como un **cuaderno de laboratorio digital** para trabajar con datos.
3. **Bibliotecas esenciales**:
    - **pandas**: organiza, limpia y transforma los datos (como un bibliotecario).
    - **NumPy**: realiza operaciones numéricas y cálculos complejos (el “genio de las matemáticas”).
4. **Visualización de datos**:
    - **Matplotlib** y **Apache Superset**: crean gráficos y tablas para mostrar patrones y tendencias (como pintar una imagen de tus datos).
5. **Claves del aprendizaje**:
    - Practicar, experimentar y disfrutar del proceso.
    - Comenzar con lo esencial antes de avanzar a técnicas más complejas.

Trucos y consejos para Jupyter notebook

1. **Qué es Jupyter Notebook**
    - Entorno de codificación interactivo que combina **código, texto y visualizaciones**.
    - Permite ejecutar código en fragmentos (“celdas”) y ver los resultados de inmediato.
    - Ideal para **prototipos, experimentación y documentación**.
2. **Tipos de celdas**
    - **Celdas de código**: Ejecutan Python, importan bibliotecas, realizan cálculos y muestran resultados.
    - **Celdas de marcado (Markdown)**: Añaden texto formateado, encabezados, listas, hipervínculos o ecuaciones para documentar y organizar el trabajo.
3. **Funciones clave y productividad**
    - **Atajos de teclado**: Crear celdas, cambiar tipos de celdas, copiar/pegar, navegar rápido.
    - **Comandos mágicos (%):**
        - `%matplotlib inline`: mostrar gráficos dentro del notebook.
        - `%timeit`: medir tiempo de ejecución.
        - `%ls`, `%pwd`, `%run`: gestión de archivos y ejecución de scripts externos.
    - **Extensiones útiles**: Jupyter NB Extensions Configurator (plegado de código, inspector de variables, tabla de contenidos).
4. **Visualización y análisis**
    - Integración con bibliotecas como **Matplotlib** y **Apache Superset** para crear gráficos y tablas.
    - Combina celdas de código y Markdown para notebooks informativos y visualmente atractivos.
5. **Colaboración y compartición**
    - Compartir notebooks en **GitHub**, **NB Viewer** o exportar a **HTML/PDF**.

- Integración con plataformas en la nube como **Azure Notebooks** para colaboración en tiempo real y uso de recursos potentes.

6. **Avances y productividad**
    - IA generativa (como Copilot) puede **generar código, documentación o funciones completas** a partir de lenguaje natural.
    - Mantenerse actualizado y explorar estas herramientas potencia la productividad y creatividad.

Pregunta

¿Cuál es la función principal de los "comandos mágicos" en el Notebook de Jupyter? Elija la mejor respuesta.

- Para instalar y gestionar extensiones para Notebooks de Jupyter
- Para proporcionar funcionalidades adicionales más allá de la ejecución estándar de código Python
- Crear informes y presentaciones visualmente atractivos
- Para colaborar y compartir cuadernos con otros usuarios

Correcto
Correcto Los comandos mágicos ofrecen características especiales como trazado, temporización de la ejecución de código, etc., que amplían las capacidades de Python.

Demostración: Métodos abreviados y consejos de productividad para Jupyter notebook (sirve para VScode)

Atajos y consejos en Jupyter Notebook

1. **Interfaz y estructura**
    - Barra de menú: Archivo, Editar, Kernel.
    - **Celdas**: donde se escribe código o texto.
        - **Celdas de código**: ejecutar Python.
        - **Celdas de Markdown**: texto, encabezados, listas, imágenes.
    - Ejecutar celda: botón Ejecutar o **Ctrl+Enter**.
    - Ejecutar y pasar a la siguiente celda: **Shift+Enter**. O **May+intro**

2. **Modos de Jupyter Notebook**
    - **Modo Edición**: editar el contenido de la celda.
    - **Modo Comando**: acciones sobre celdas (añadir, borrar, mover).
        - Entrar: tecla **Esc**.
        - Atajos útiles:
            - `A`: nueva celda arriba
            - `B`: nueva celda abajo
            - `D D`: borrar celda
            - `Z`: deshacer borrado
            - `M`: cambiar a Markdown
            - `Y`: cambiar a código
            - `H`: mostrar ayuda de atajos
3. **Productividad y autocompletado**
    - **Tabulador**: autocompleta comandos y nombres de variables.
    - **Shift+Tab**: muestra documentación de funciones.
4. **Comandos mágicos (%)**
    - `%matplotlib inline`: mostrar gráficos dentro del notebook.
    - `%timeit`: medir tiempo de ejecución de código.
    - `!pip install nombre_librería`: instalar librerías desde el notebook.
5. **Trabajo con datos en Python**
    - Uso de **pandas** para análisis de datos:

        - `head()`: primeras filas
        - `describe()`: estadísticas resumidas
        - `groupby()` + `sum()`: agrupar y sumar datos
    - Visualización de resultados con **Matplotlib** (`%matplotlib inline`).
6. **Buenas prácticas**
    - Usar **atajos de teclado** para agilizar el flujo de trabajo.
    - Organizar notebooks con **encabezados y comentarios**.
    - Experimentar y probar comandos mágicos para mejorar análisis y gestión del entorno.
    - Mantener notebooks claros y documentados.

Actividad: Un análisis sencillo en Jupyter Notebooks

Escenario

Acabas de unirte a un equipo de astrónomos que exploran las maravillas de nuestro sistema solar. Deseoso de contribuir, te han encargado que analices datos sobre los planetas, sus tamaños y el número de lunas que poseen. Sin embargo, hay un problema: los datos están dispersos en diferentes archivos y formatos, ¡y nunca antes has utilizado Notebooks de Jupyter! Los astrónomos, entusiasmados con el potencial de esta herramienta interactiva, te han pedido que aprendas a utilizarla mientras abordas esta tarea de análisis de datos.

Tu misión ahora implica no sólo cargar, limpiar y analizar los datos utilizando tus conocimientos de Python, sino también dominar el entorno de Notebook de Jupyter para crear visualizaciones y comunicar eficazmente tus hallazgos al equipo.

Objetivo

El objetivo de esta actividad es introducirte en los fundamentos del análisis y visualización de datos en Python utilizando Jupyter Notebooks. Aprenderás a importar librerías esenciales como NumPy y pandas, cargar datos de archivos CSV en DataFrames y realizar cálculos básicos y estadísticas descriptivas usando NumPy. Al final de esta actividad, tendrás una base sólida en el uso de Python para la exploración de datos y estarás preparado para cuadernos Jupyter más complejos.

Instrucciones

Descarga PYTC2M1P01_Activity_AsimpleanalysisinJupyterNotebook.zip y guárdalo en el escritorio de tu máquina y extráelo a una carpeta donde sea fácilmente accesible.

PYTC2M1P01_Activity_AsimpleanalysisinJupyterNotebook.zip
ZIP File

Abra la carpeta, que contiene los siguientes archivos

- un archivo de Notebook de Jupyter llamado "PYTC2M1P01_Activity_ A simple analysis in Notebook de Jupyter.ipynb"
- un conjunto de datos de valores separados por comas (CSV) llamado "planets_with_moons_all.csv"

Paso 1: ¡Hola, mundo! Tus primeros pasos con los Notebooks de Jupyter

Los Notebooks de Jupyter son documentos interactivos que combinan código, texto y visualizaciones. Familiarízate con la interfaz de Notebook de Jupyter, incluyendo las celdas (código y markdown), el núcleo y la barra de herramientas.

- Se le ha proporcionado una celda de código con una sentencia `print`.
- Sustituye `"insert code here"` por `"Hello, World!"`.
- Ejecute la celda utilizando `Shift + Enter` o el botón "Ejecutar".

Paso 2: Expandiendo el poder de Python con la librería numpy

Las capacidades de Python pueden ampliarse en gran medida mediante el uso de bibliotecas y paquetes. Estos son colecciones de código pre-escritas que proporcionan funcionalidad adicional. La biblioteca `numpy` es la piedra angular de la computación numérica en Python.

- En la celda de código vacía, escribe `"import numpy as np"` en la celda y ejecútalo.

Paso 3: numpy en acción

Ahora que hemos importado `numpy`, verás cómo puede simplificar los cálculos. Lo usarás para encontrar el promedio de una lista de números.

- Se te ha proporcionado un array de números (`numbers`) y el código para calcular la media usando `np.mean`, pero necesitas pasar el array `numbers` a `np.mean`.

Paso 4: Lectura de datos CSV en Jupyter

Los Archivos CSV (Valores separados por comas) son una forma común de almacenar datos en un formato tabular, similar a una hoja de cálculo. También puede cargar estos Archivos CSV en un Notebook de Jupyter

- El código para importar Pandas y leer un Archivo CSV ha sido proporcionado para usted.
- Proporcione el nombre de archivo correcto a `pd.read_csv` y ejecute la celda.

Paso 5: Combinar datos CSV con numpy

Ahora que tienes tus datos en un DataFrame, usarás `numpy` para calcular algunas estadísticas descriptivas sobre los planetas de nuestro sistema solar.

- Se te ha proporcionado el código para calcular la media y la mediana tanto para el diámetro como para el número de lunas.
- Ejecuta la celda.

1. ¿Cuál es el propósito principal de las bibliotecas y paquetes en Python?

- ○ Sustituir las funciones básicas del lenguaje Python
- ○ Hacer el código Python más complejo y difícil de entender
- ○ Para ralentizar la ejecución de programas Python
- ◉ Ampliar las capacidades de Python y proporcionar herramientas especializadas para diversas tareas

Correcto
¡Correcto! Las bibliotecas y los paquetes ofrecen una gran cantidad de funciones y módulos predefinidos que le ahorrarán tiempo y esfuerzo

2. ¿Qué biblioteca se utiliza habitualmente para cálculos numéricos y para trabajar con matrices en Python?

- ○ pytest
- ○ matplotlib
- ◉ numpy
- ○ pandas

Correcto
Correcto! numpy es la librería para el manejo de arrays, matrices y operaciones matemáticas en Python

3. ¿Cuál es la principal ventaja de utilizar la sintaxis `import ... as ...` al importar bibliotecas en Python?

- ○ Oculta las funciones de la biblioteca de otras partes de su código
- ○ Evita conflictos con otras bibliotecas que puedan tener nombres de función similares
- ◉ Proporciona un alias más corto y cómodo para la biblioteca
- ○ Acelera la ejecución del código

Correcto
Correcto Esto hace que su código sea más legible y fácil de escribir

4. ¿Qué formato de archivo se utiliza habitualmente para almacenar datos tabulares, similares a los de una hoja de cálculo?

- ○ PDF
- ◉ CSV
- ○ JSON
- ○ TXT

Correcto
Correcto CSV (Comma-Separated Values) es un formato muy utilizado para almacenar datos en filas y columnas

Bibliotecas esenciales de Python para análisis de datos: NumPy

1. Qué es NumPy

- NumPy es una biblioteca para cálculo numérico en Python.
- Proporciona la estructura `ndarray` (matriz n-dimensional) para almacenar datos numéricos de forma eficiente.
- Permite realizar operaciones matemáticas avanzadas de manera rápida y concisa.

2. Vectorización

- Reemplaza bucles explícitos por operaciones a nivel de array, aumentando eficiencia y claridad del código.

- **Ejemplo: cálculo de variación porcentual de precios:**

```
import numpy as np

prices = np.array([100, 105, 112, 98])
percentage_change = (prices[1:] - prices[:-1]) / prices[:-1] * 100
```

- Resultado: NumPy calcula todo de una vez, evitando bucles.

3. Difusión (Broadcasting)

- Permite operaciones entre arrays de diferentes formas al expandir automáticamente dimensiones.

- Útil para operaciones elemento a elemento sin bucles explícitos.

Ejemplo de difusión básica:

```
import numpy as np

arr1 = np.array([[1, 2, 3],
                 [4, 5, 6]])  # Shape: (2, 3)
arr2 = np.array([10, 20])       # Shape: (2,)

# Error: dimensiones incompatibles
# result = arr1 + arr2  # ValueError

# Solución: remodelar arr2
arr2_reshaped = arr2.reshape(2, 1)  # Shape: (2, 1)
result = arr1 + arr2_reshaped
print(result)
# [[11 12 13]
#  [24 25 26]]
```

Otro ejemplo usando `np.newaxis`:

```
arr2_newaxis = arr2[:, np.newaxis]  # Shape: (2, 1)
result = arr1 + arr2_newaxis        # Shape: (2, 3)
print(result)
# [[11 12 13]
#  [24 25 26]]
```

- Clave: las dimensiones se alinean desde la derecha; si no coinciden correctamente, se produce un error (`ValueError`).

4. Recomendaciones

- Comprender la vectorización y la difusión es esencial para usar NumPy eficazmente.
- Siempre revisar las formas de los arrays antes de operaciones para evitar errores de broadcasting.

Estrategias para NumPy

1. Visualizar las formas antes de difundir

- Antes de usar difusión (broadcasting), revisa cómo se expandirán las dimensiones para asegurarte de que la operación sea la deseada.

2. Ajustar formas con `reshape()` o `np.newaxis`

- Útiles para alinear matrices antes de operaciones de difusión.

3. Depuración de difusión

- Imprime las formas de las matrices en diferentes etapas para identificar desajustes.

4. Precisión numérica y errores silenciosos

- La aritmética de coma flotante puede generar resultados inesperados.
- Ejemplo de suma de valores pequeños que no da el resultado exacto:

```
import numpy as np
```

```
# Crear un array con un millón de elementos de 0.1
small_values = np.full(1000000, 0.1)

# Sumar los valores
sum_value = np.sum(small_values)

print(sum_value)  # Resultado: 99999.9999999998 (no exactamente 100000)
```

- Para cálculos críticos, considera bibliotecas especializadas o algoritmos como la **suma de Kahan**.

Estrategias para Pandas

1. Indexación

- Diferencia entre `iloc` (por posición) y `loc` (por etiqueta).
- Error común: usar `df[0]` esperando `df.loc[0]`.

2. Alineación de datos

- Al sumar DataFrames, asegúrate de que los índices y nombres de columna coincidan.

3. Datos faltantes

- La imputación incorrecta puede sesgar los resultados.
- Ejemplo: rellenar ingresos faltantes con la media puede ocultar diferencias importantes.

4. Agregación y transformación

- Operaciones pueden modificar DataFrames en el lugar.
- Mejor crear copias si vas a cambiar los datos:

```
df_copy = df.copy()
```

Estrategias para Matplotlib

1. Enfoque por capas

- Crear gráficos complejos requiere práctica.
- Divide gráficos complicados en subplots para mayor claridad.

2. Personalización

- Evita exceso de colores, fuentes y decoraciones.
- Prioriza la claridad sobre la complejidad.

3. Integración con otras bibliotecas

- Mantén librerías actualizadas y usa entornos virtuales para evitar conflictos.

Conclusión

- **NumPy**: vectorización y difusión optimizan cálculos numéricos, pero requiere atención a formas y precisión.
- **Pandas**: facilita limpieza, preparación y manipulación de datos, cuidado con indexación, alineación y valores faltantes.
- **Matplotlib**: permite visualizaciones detalladas, con control total, pero requiere práctica y enfoque en claridad.
- La clave es **conocer los matices de cada biblioteca**, practicar y acercarse a los datos con conocimiento, para transformar datos crudos en información significativa.

Casos de uso de las bibliotecas de Python

NumPy

- Base de la computación numérica en Python.
- Ofrece **ndArray**: estructuras multidimensionales rápidas y eficientes.
- Casos de uso:
    - Simulaciones científicas e ingeniería (ej. patrones climáticos, cargas en estructuras).
    - Procesamiento de imágenes (filtros, detección de bordes, reconocimiento, efectos visuales).
- Incluye funciones optimizadas para **álgebra lineal, transformadas de Fourier y operaciones matriciales**.

Pandas

- Especializado en **manipulación y análisis de datos**.
- Presenta el **DataFrame**, similar a una tabla SQL o Excel.
- Casos de uso:
    - Limpieza de datos: manejo de valores faltantes, conversión de tipos, filtrado y unión de datasets.
    - Exploración: estadísticas resumidas, agrupación, agregación.
    - Análisis de series temporales: seguimiento de tendencias y pronósticos.
- Muy usado por **analistas de negocio y científicos de datos**.

Matplotlib

- Biblioteca de **visualización de datos**.
- Casos de uso:
    - **Análisis exploratorio de datos (EDA):** histogramas, diagramas de dispersión, gráficos de barras y líneas para detectar patrones, outliers y

correlaciones.

    - Visualizaciones con **calidad de publicación** para artículos, reportes y presentaciones.

- Ofrece control total sobre estilo, colores, etiquetas, leyendas y diseño.
- Convierte datos en **historias visuales claras y profesionales**.

Apache Superset

- Construido sobre Matplotlib, pero con un nivel más alto de abstracción.
- Casos de uso:
    - Crear **gráficos estadísticos avanzados** como mapas térmicos, gráficos de violín o gráficos de pares.
    - Comunicar relaciones de datos complejas de forma clara a audiencias técnicas y no técnicas.
- Ventajas:
    - Paletas y estilos predeterminados que generan visualizaciones atractivas.
    - Combinación de **estética + capacidades estadísticas**, ideal para presentaciones impactantes.

Conclusión

NumPy, Pandas, Matplotlib y Apache Superset forman un **cuarteto esencial para el análisis de datos en Python**. Cada uno cumple un rol:

- **NumPy**: cálculos numéricos y científicos.
- **Pandas**: manipulación y organización de datos.
- **Matplotlib**: visualización detallada y personalizable.
- **Apache Superset**: visualizaciones estadísticas avanzadas, claras y estéticas.

Con ellos, los profesionales de datos pueden abordar problemas complejos con confianza y eficiencia.

Pregunta

¿Qué biblioteca introduce el DataFrame, una estructura de datos para la manipulación y el análisis de datos? Seleccione la mejor respuesta.

- ○ NumPy
- ○ Matplotlib
- ◉ pandas
- ○ Superset Apache

Correcto
Pandas presenta el DataFrame, una estructura de datos potente e intuitiva que simplifica la limpieza, transformación y exploración de datos.

Cuestionario: Configuración del conjunto de herramientas de análisis de datos

1. Estás aprendiendo sobre diferentes bibliotecas de Python utilizadas en el Análisis de datos. Quieres entender el propósito principal de la biblioteca Matplotlib. ¿Para qué se utiliza principalmente? Selecciona la mejor respuesta.

- ○ Cálculos numéricos
- ◉ Visualización de datos
- ○ Tratamiento de datos de series temporales
- ○ Manipulación de datos

Correcto
Correcto Matplotlib se utiliza principalmente para crear visualizaciones estáticas, animadas e interactivas en Python.

2. Estás trabajando en un proyecto de Análisis de datos y necesitas crear algunas visualizaciones para representar tus hallazgos. ¿Cuál de las siguientes bibliotecas de Python se utiliza principalmente para la visualización de datos? Selecciona la mejor respuesta.

- ○ pandas
- ○ SciPy
- ○ NumPy
- ◉ Matplotlib

Correcto
Correcto Matplotlib es una biblioteca popular para la Visualización de datos.

3. Estás trabajando en Notebook de Jupyter y quieres ejecutar la celda actual y pasar a la siguiente. ¿Qué atajo de teclado debes utilizar? Selecciona la mejor respuesta.

- ○ Ctrl + Intro
- ○ Alt + Intro
- ○ Ctrl + Mayús + Intro
- ◉ Mayús + Intro

Correcto
Correcto Mayús + Intro ejecuta la celda actual y pasa a la siguiente, lo que le permite agilizar su flujo de trabajo.

4. Estás aprendiendo sobre el entorno Notebook de Jupyter y sus componentes clave. ¿Cuál es la función principal del Kernel en un Notebook de Jupyter? Selecciona la mejor respuesta.

- ○ Para gestionar el sistema de archivos del Notebook
- ○ Para resaltar la sintaxis de las celdas de código
- ○ Para renderizar celdas markdown como HTML
- ◉ Para ejecutar el código contenido en las celdas del cuaderno

Correcto
Correcto El Kernel es responsable de ejecutar el código en las celdas del Notebook.

5. Estás trabajando con dos matrices NumPy, `array_a` con forma `(3, 4)` y `array_b` con forma `(3,)`. Intentas realizar una suma de elementos: `result = array_a + array_b`. ¿Cuál es el resultado más probable de esta operación debido a las reglas de emisión de NumPy? Selecciona la mejor respuesta.

- ○ NumPy cambiará automáticamente la forma de `array_b` a `(3, 1)` y, a continuación, realizará la suma de elementos.
- ○ La operación se ejecutará correctamente y `array_b` se difundirá por las filas de `array_a`.
- ○ La operación se ejecutará correctamente, y array_b se difundirá a través de las columnas de array_a.
- ◉ NumPy lanzará un mensaje `ValueError` indicando que las matrices no pueden emitirse juntas debido a formas incompatibles.

Correcto
Correcto Las formas `(3, 4)` y `(3,)` son incompatibles para la difusión porque sus dimensiones finales (4 y 3) no son ni iguales ni una.

6. Estás explorando las ventajas de utilizar los widgets interactivos de Notebook de Jupyter. ¿Cuáles de los siguientes son beneficios de usar estos widgets? Selecciona todas las que correspondan.

- [x] Permiten la manipulación y visualización de datos en tiempo real.

Correcto
Correcto Los widgets interactivos permiten manipular y visualizar datos en tiempo real, lo que mejora el análisis.

- [ ] Se integran con otras herramientas de análisis de datos.
- [ ] Muestran los resultados de la ejecución del código.
- [x] Pueden utilizarse para crear cuadros de mando interactivos.

Correcto
Correcto Los widgets se pueden utilizar para crear cuadros de mando interactivos dentro de los Notebooks de Jupyter.

Comprender los conjuntos de datos

Un **conjunto de datos** es como una gran hoja de cálculo formada por **filas** (cada fila representa un cliente u observación) y **columnas** (atributos o características como edad, bebida favorita, hora del día, etc.).

- Ejemplo: clientes en una cafetería con datos de edad, bebida preferida y momento de visita.
- Los conjuntos de datos permiten **detectar patrones**, como jóvenes que piden café helado por la tarde.
- Se pueden enriquecer con más información: clima, uso de tarjeta de fidelización, reseñas, etc., lo que abre la puerta a análisis más profundos (ej: bebidas calientes en días fríos, clientes fieles que gastan más).
- Estos datos pueden **visualizarse** con gráficos de barras, circulares, etc., para comprender mejor las tendencias.
- En definitiva, los **datos son el combustible** de proyectos de análisis o predicción, y **Python es la herramienta** para limpiarlos, analizarlos y usarlos para tomar decisiones o predecir comportamientos futuros.

Tipos de datos y fuentes habituales

Tipos de conjuntos de datos

1. **Series temporales (Time Series)**
    - Datos recogidos en orden cronológico, con una **marca de tiempo** en cada observación.
    - Ejemplos: precios de acciones diarios, temperatura, ritmo cardíaco.
    - Usos:
        - Finanzas → predicción de precios, tendencias de mercado.
        - Salud → monitoreo de pacientes, detección de brotes.
        - Medio ambiente → patrones climáticos, desastres naturales.
    - **Fuentes**: Yahoo Finanzas, Quandl, Alpha Vantage, NOAA, Data.gov, Kaggle.

2. **Transversales (Cross-Sectional)**
    - Datos que capturan una **instantánea en un momento específico**, describiendo varios individuos o grupos.
    - Ejemplos: encuestas de ingresos, edad y gasto mensual.
    - Usos:
        - Ciencias sociales → opiniones, comportamientos, prevalencia de características.
        - Marketing → preferencias del consumidor, segmentación.
        - Salud → comparación de resultados entre grupos demográficos.
    - **Fuentes**: Pew Research Center, ICPSR, encuestas (SurveyMonkey, Qualtrics), Kaggle, Data.gov.

3. **De panel o longitudinales (Panel Data)**

- Combinan **tiempo + comparación de individuos**, siguiendo a las mismas personas o entidades en varios periodos.
- Ejemplos: rendimiento académico de estudiantes durante años, evolución de ingresos familiares.
- Usos:
  - Medir impacto de políticas, intervenciones o eventos en individuos.
  - Educación → trayectoria académica y profesional.
  - Salud → resultados de tratamientos, uso de servicios médicos.
- **Fuentes**:
  - Encuestas longitudinales (educación, ingresos, salud).
  - Datos administrativos (registros fiscales, reclamaciones médicas, matriculación escolar).
  - Repositorios especializados en economía, salud y educación.

Conclusión:

- **Series temporales** → cambios a lo largo del tiempo.
- **Transversales** → instantánea en un momento.
- **Panel** → combinación de ambas (misma entidad observada en el tiempo).
  Cada tipo de conjunto de datos permite distintos tipos de análisis y proviene de fuentes específicas como encuestas, plataformas abiertas (Kaggle, Data.gov) o instituciones de investigación.

Otros tipos de conjuntos de datos

1. **Basados en eventos**
   - Capturan datos en el momento exacto en que ocurre un evento (intervalos irregulares).
   - Ejemplos: transacciones financieras, clics web, sensores, visitas médicas.

- Enfocados en la **secuencia y el momento** → útiles para analizar sistemas dinámicos.

2. **Geoespaciales**
    - Contienen información de **ubicación y características geográficas**.
    - Usos: mapas interactivos, rutas de transporte, urbanismo, análisis ambiental.
    - Herramientas: Azure Synapse, Data Explorer, Azure Maps.
3. **De texto**
    - Incluyen redes sociales, noticias, reseñas, documentos legales.
    - Desafíos: lenguaje humano, ambigüedad.
    - Herramientas: Azure Cognitive Services (análisis de sentimiento, extracción de frases), Azure ML (NLP avanzado).
4. **De imágenes**
    - Fotos y videos → base de la visión por computador.
    - Usos: coches autónomos, análisis médico, realidad aumentada.
    - Herramientas: OpenCV, PyTorch.

Calidad de los datos (clave para un análisis fiable)

Un buen conjunto de datos debe ser:

- **Exacto** → sin errores.
- **Íntegro** → sin lagunas.
- **Coherente** → mismo formato y definiciones.
- **Actual** → reflejar la realidad presente.
- **Pertinente** → útil para la pregunta de investigación.

Prácticas clave:

- Recopilación cuidadosa (evitar sesgos/errores).
- Limpieza de datos (corrección de errores, imputación de faltantes, valores atípicos).
- Validación de datos (comparación con fuentes externas).

Ética de los datos (brújula moral)

Principios fundamentales:

- **Privacidad** → proteger datos personales, consentimiento informado, seguridad.
- **Evitar sesgos** → prevenir discriminación y resultados distorsionados.
- **Transparencia** → claridad en fuentes, métodos y análisis.
- **Responsabilidad** → asumir consecuencias y corregir errores.
- **Equidad** → uso justo de datos, sin explotación ni exclusión.

Los datos pueden usarse tanto para **empoderar** (salud pública, ciencia, innovación) como para **dañar** (vigilancia, manipulación, discriminación).

Conclusión

- El mundo de los datos es diverso: series temporales, transversales, panel, eventos, geoespaciales, texto e imágenes.
- La **calidad** y la **ética** son tan importantes como el análisis técnico.
- El analista de datos no solo procesa números: **es un narrador** que revela patrones y tendencias para tomar decisiones responsables y justas.

Encontrar y acceder a conjuntos de datos en el mundo real

Importancia de los conjuntos de datos del mundo real

- Base de la innovación en múltiples campos:
    - **Salud** → identificar patrones de enfermedades, diseñar tratamientos, mejorar resultados.
    - **Finanzas** → trading algorítmico, gestión de riesgos, detección de fraudes.
    - **Marketing** → entender clientes, personalizar campañas, optimizar publicidad.
- Ejemplo: **apps de recomendación musical** → usan historial de escucha, metadatos y análisis de estado de ánimo para personalizar experiencias.

Plataformas y fuentes de datos

1. **Microsoft Azure Open Datasets**
    - Datos preparados para IA y ML.
    - Integración fácil con Azure Machine Learning.
2. **Microsoft Research Data Repository**
    - Colección de datasets de investigación (IA, visión artificial, salud).
    - Muchos de acceso público.
3. **APIs (Interfaces de Programación de Aplicaciones)**
    - Acceso a datos actualizados/en tiempo real.
    - Ejemplo: API de redes sociales, API de mapas.
    - Más complejas que descargar un archivo, pero más flexibles.
4. **Recursos especializados**

- **Microsoft Planetary Computer** → datos geoespaciales y ambientales.
- **Banco Mundial (Open Data)** → pobreza, educación, salud, economía.
- **Azure Marketplace (finanzas)** → datos financieros y económicos.
- **AI for Health (Microsoft)** → datasets de investigación en salud.

Cómo descargar y trabajar con los datos

- Descarga directa desde repositorios → formatos comunes: **CSV, JSON, ZIP, RAR**.
- Python facilita el trabajo con **pandas** y **NumPy**.

Aspectos clave a considerar

- **Licencias** → algunos datos son abiertos, otros tienen restricciones legales.
- **Calidad y relevancia** del dataset → determinan el éxito de un proyecto.
- **Formato y limitaciones** → siempre revisarlos antes de usar.

Conclusión

Los conjuntos de datos del mundo real son el **combustible** de la ciencia de datos y la IA. Conocer **dónde buscarlos, cómo descargarlos y respetar licencias** es tan importante como analizarlos. La exploración cuidadosa abre la puerta a descubrimientos y soluciones innovadoras.

Pregunta

Estás trabajando en un proyecto que requiere el análisis de datos geográficos para predecir la propagación de incendios forestales. ¿Cuál de los siguientes recursos mencionados en la lección sería el más adecuado para encontrar los conjuntos de datos pertinentes? Selecciona la mejor respuesta.

- ○ Iniciativa de Microsoft sobre IA para la salud
- ○ Mercado Microsoft Azure
- ○ Datos abiertos del Banco Mundial
- ◉ Ordenador planetario de Microsoft

Correcto
Correcto Microsoft Planetary Computer está especializado en el suministro de conjuntos de datos geoespaciales y medioambientales, por lo que resulta idóneo para su proyecto de predicción de incendios forestales.

Limpieza de datos 101

¿Qué es la limpieza de datos?

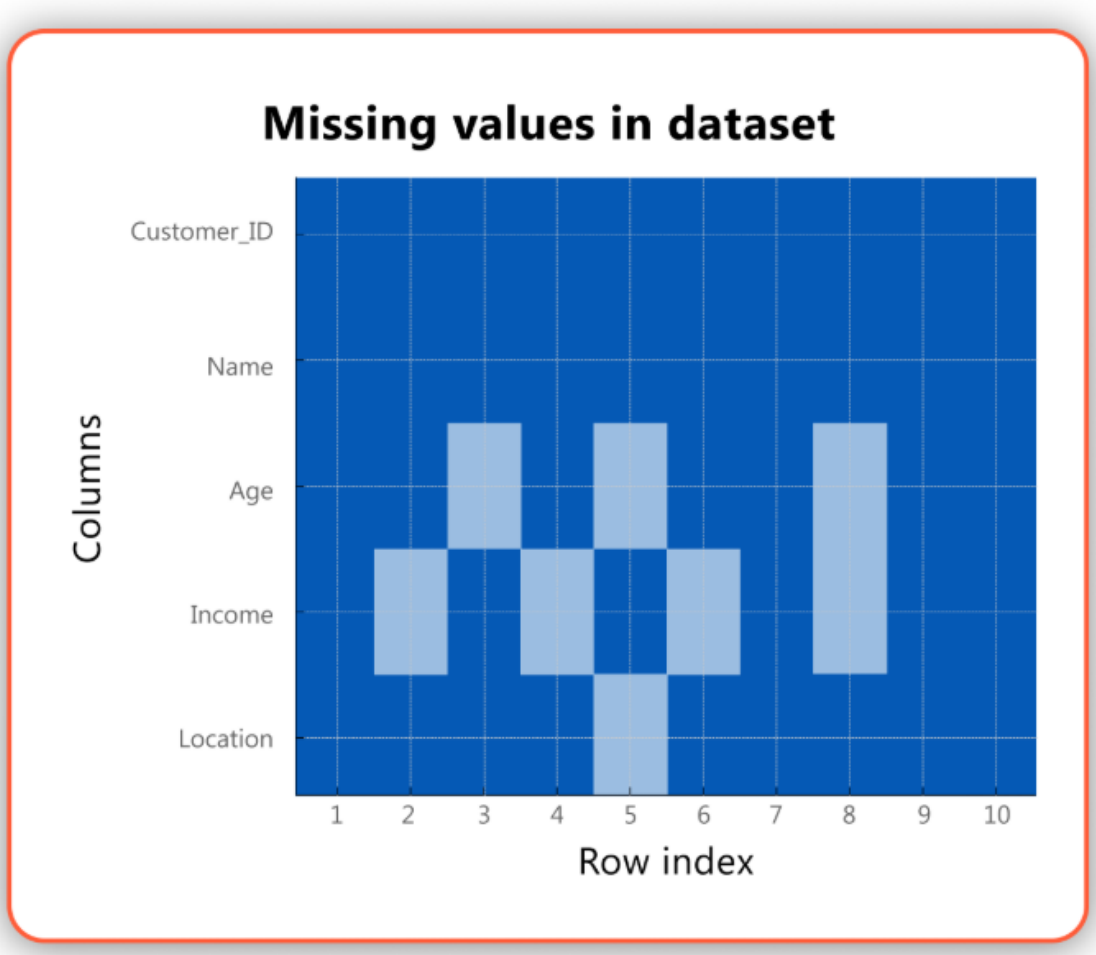
- También llamada *data wrangling* o *data munging*.
- Proceso de transformar datos en bruto (desordenados, incompletos, incoherentes) en un formato **limpio, estructurado y utilizable**.
- Incluye:
    - Corregir errores.
    - Estandarizar encabezados.
    - Asegurar que los datos sean comprensibles y listos para análisis o modelos.

Importancia

- Garantiza **precisión, fiabilidad y solidez** en análisis y modelos.
- Evita conclusiones erróneas por datos defectuosos.
- Ahorra tiempo al reducir problemas de depuración.
- Facilita la colaboración y el intercambio de información.

Problemas comunes en la calidad de datos

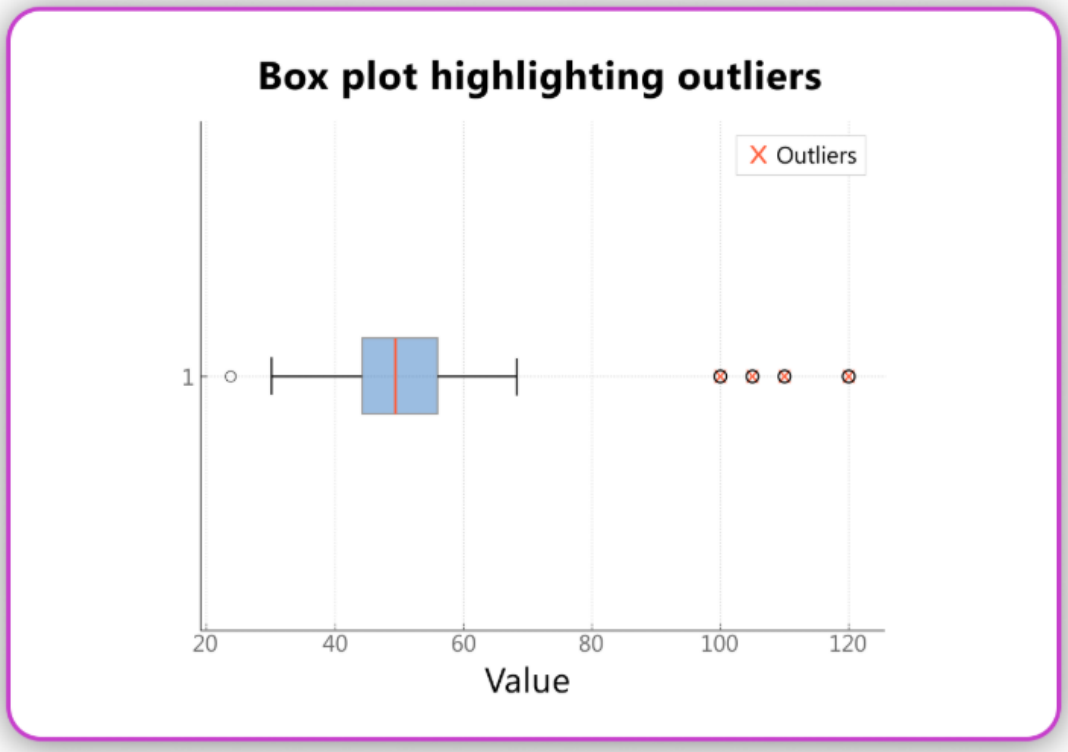
1. **Valores perdidos**
    - Como piezas faltantes de un rompecabezas.
    - Causas: errores de entrada, fallos técnicos, datos no disponibles.
    - Si no se tratan → análisis sesgado.



Cada bloque azul claro de este gráfico indica la presencia de un valor omitido en una columna específica del conjunto de datos.

2. **Valores atípicos (outliers)**
    - Datos que se desvían mucho del resto.
    - Pueden ser anomalías reales o errores.

- Distorsionan medidas estadísticas (ej: precios de casas con mansiones extremadamente caras).



Este gráfico muestra cómo los valores atípicos son puntos de datos que se desvían significativamente de la norma. Estos valores atípicos pueden distorsionar el análisis estadístico.

3. **Inconsistencias**
    - Diferencias en nombres, formatos o métodos de recopilación.
    - Ejemplos: "ID cliente" vs. "ID_cliente", "St." vs. "Street".
    - Dificultan la combinación y el análisis de datos.

En conclusión: La limpieza de datos es la base del análisis de calidad. Permite tener información confiable, procesable y útil, evitando sesgos y errores que puedan comprometer los resultados.

Estrategias para problemas de calidad de datos

1. **Valores perdidos**

    - **Eliminar datos**: rápido, pero arriesgado porque puede sesgar los resultados.
    - **Imputar valores**: sustituir por media, mediana o moda.
    - **Métodos avanzados**: imputación por regresión o múltiple para mayor precisión.
2. **Valores atípicos (outliers)**
    - Evaluar si son **errores** o **información valiosa**.
    - Opciones: eliminarlos, aplicar *capping* (limitar extremos), usar métodos robustos (mediana, rango intercuartílico) o escaladores como **RobustScaler** en Machine Learning.
3. **Inconsistencias**
    - Normalizar formatos, corregir errores, reconciliar datos contradictorios.
    - En **Python/Pandas**:
        - `replace()` para sustituir valores.
        - `apply()` para transformaciones personalizadas.
        - Expresiones regulares para estandarizar texto.
        - Diccionarios/metadatos para definir rangos y valores válidos.

Ejemplos prácticos

- **Escenario 1 (clientes con edades faltantes):** imputar edad usando la media de clientes con patrones similares.
- **Escenario 2 (temperaturas con outliers):** eliminar o corregir valores extremos debidos a fallos de sensores.
- **Escenario 3 (nombres de empleados incoherentes):** estandarizar formatos ("John Doe" vs "Doe, John").

Consideraciones críticas

- La limpieza de datos puede introducir sesgos si no se hace con cuidado.
- Se deben **documentar las decisiones** y mantener la **transparencia**.
- El objetivo es mejorar la **calidad y utilidad** de los datos, no manipularlos.

Conclusión: La limpieza de datos es la base de un análisis sólido. Aplicando las estrategias adecuadas se asegura fiabilidad, precisión y conclusiones robustas en proyectos de ciencia de datos y desarrollo con Python.

Cuestionario: Sumergirse en los conjuntos de datos

1. Usted está explorando diferentes tipos de conjuntos de datos y necesita identificar las características de los conjuntos de datos transversales. ¿Cuáles de las siguientes características suelen asociarse a los conjuntos de datos transversales? Seleccione todas las que correspondan.

   - [ ] Seguimiento de los mismos sujetos a lo largo del tiempo
   - [x] Permite la comparación entre diferentes temas

     Correcto
     Correcto Los conjuntos de datos transversales son útiles para comparar diferentes sujetos en un momento dado.

   - [x] Datos recogidos en un momento dado

     Correcto
     Correcto Los conjuntos de datos transversales recogen datos de los sujetos en un momento concreto.

   - [ ] Datos recogidos en intervalos

2. Está limpiando un conjunto de datos y encuentra valores que faltan en algunas columnas. ¿Cuáles de las siguientes son soluciones básicas para tratar estos datos que faltan? Seleccione todas las que correspondan.

- [x] Utilización de algoritmos para predecir los valores que faltan.

Correcto
Correcto El uso de algoritmos para predecir los valores que faltan es un método sofisticado para tratar los datos que faltan.

- [ ] Eliminación de datos no relacionados.
- [x] Imputación de valores perdidos.

Correcto
Correcto La imputación de los valores que faltan es un método habitual para tratar los datos que faltan.

- [x] Eliminación de filas con datos ausentes.

Correcto
Correcto La eliminación de filas con datos que faltan es una solución básica para tratar estos problemas.

3. Está trabajando con un conjunto de datos y necesita comprender el propósito de una cabecera. ¿Qué función tienen las cabeceras en un conjunto de datos? Seleccione la mejor respuesta.

- ○ Para almacenar los valores de los datos
- ◉ Para definir los nombres de las columnas
- ○ Para separar diferentes tablas
- ○ Resumir el conjunto de datos

Correcto
Correcto Las cabeceras definen los nombres de las columnas de un conjunto de datos.

4. Está buscando conjuntos de datos y cuadernos compartidos por científicos de datos y profesionales del aprendizaje automático. ¿Cuál de los siguientes repositorios es el mejor para este propósito? Seleccione la mejor respuesta.

- ○ Medio
- ◉ Kaggle
- ○ Github
- ○ Quora

Correcto
¡Correcto! Kaggle es una conocida plataforma para que los entusiastas de los datos compartan conjuntos de datos y cuadernos.

5. Estás buscando conjuntos de datos en repositorios públicos y necesitas identificar los formatos de archivo más comunes utilizados para almacenar y compartir estos conjuntos de datos. ¿Cuáles de los siguientes son los formatos de archivo más comunes que encontrará? Seleccione la mejor respuesta.

- ◉ CSV, JSON, Excel
- ○ DOCX, PDF, TXT
- ○ MP3, WAV, FLAC
- ○ JPEG, PNG, GIF

Correcto
Correcto Estos formatos se utilizan ampliamente para almacenar y compartir conjuntos de datos.

ACTIVIDAD FINAL: INTRODUCCIÓN AL ANÁLISIS DE DATOS

1. Está trabajando con un conjunto de datos que se recopiló mediante la introducción manual de datos. ¿Cuál de los siguientes problemas de calidad de datos es más probable que ocurra en este escenario? Seleccione la mejor respuesta.

   - ○ Homogeneidad
   - ◉ Errores tipográficos
   - ○ Valores atípicos
   - ○ Errores sistemáticos

   Correcto
   Correcto La introducción manual de datos es propensa a errores tipográficos, lo que puede afectar a la calidad de los datos.

2. Está aprendiendo sobre diferentes tipos de conjuntos de datos y necesita identificar las características de los datos de panel. ¿Cuáles de las siguientes son características típicas de los datos de panel? Seleccione todas las que correspondan.

   - ☐ Datos recogidos en un momento dado
   - ☑ Datos que incluyen dimensiones de series temporales y transversales

   Correcto
   Correcto Los datos de panel incluyen tanto dimensiones de series temporales como transversales.

   - ☑ Datos recogidos de múltiples sujetos

   Correcto
   Correcto Los datos de panel se recogen de múltiples sujetos.

   - ☑ Datos que implican observaciones repetidas

   Correcto
   Correcto Los datos de panel implican observaciones repetidas.

3. Está trabajando en un proyecto que implica la integración de datos de múltiples fuentes. ¿Cuáles de los siguientes son problemas comunes de calidad de datos que pueden surgir en este escenario? Seleccione todo lo que corresponda.

- [ ] Errores en la introducción de datos
- [x] Datos que faltan

Correcto
Correcto Los datos que faltan pueden producirse al combinar conjuntos de datos de distintas fuentes.

- [x] Registros duplicados

Correcto
Correcto Los registros duplicados pueden ser el resultado de la integración de datos procedentes de múltiples fuentes.

- [x] Formatos de datos incoherentes

Correcto
Correcto Los formatos de datos incoherentes son un problema común cuando se integran datos de múltiples fuentes.

4. Estás aprendiendo cómo se utiliza el Análisis de datos en diferentes industrias. ¿Cuál de los siguientes sectores utiliza habitualmente el análisis de datos para mejorar sus operaciones? Seleccione todas las que correspondan.

- [x] Agricultura

Correcto
Correcto El Análisis de datos se utiliza en agricultura para la agricultura de precisión y la optimización del rendimiento.

- [ ] Construcción
- [x] Educación

Correcto
Correcto El Análisis de datos se utiliza en educación para analizar el rendimiento de los alumnos y mejorar los resultados.

- [x] Finanzas

Correcto
Correcto El sector financiero se basa en el análisis de datos para diversas operaciones.

5. Usted está aprendiendo las distintas formas en que se aplica el Análisis de datos en el sector sanitario. ¿Cuáles de las siguientes son las funciones del Análisis de datos en la sanidad? Seleccione todas las que corresponda.

- [x] Predicción de brotes de enfermedades

Correcto
Correcto El Análisis de datos se utiliza para predecir brotes de enfermedades, lo que permite adoptar medidas proactivas.

- [x] Mejorar la calidad de la atención al paciente

Correcto
Correcto La Analítica de datos ayuda a mejorar la calidad de la atención al paciente a través de la información y el análisis de tendencias.

- [x] Seguimiento de la propagación de enfermedades

Correcto
Correcto El análisis de datos se utiliza para rastrear la propagación de enfermedades y fundamentar las decisiones en materia de salud pública.

- [ ] Desarrollo de nuevos equipos médicos

6. Usted es un Analista de datos que trabaja para una empresa de telecomunicaciones. ¿Cuáles de las siguientes son aplicaciones del Análisis de datos en su sector? Seleccione todas las que procedan.

- [x] Reducción de los tiempos de inactividad de la red

Correcto
Correcto El análisis de datos puede ayudar a predecir y prevenir problemas en la red.

- [x] Mejora de la seguridad de la red

Correcto
Correcto El Análisis de datos desempeña un papel fundamental en la identificación y mitigación de las amenazas a la seguridad.

- [ ] Control del rendimiento de los empleados
- [x] Mejorar la satisfacción del cliente

Correcto
Correcto La Analítica de datos ayuda a comprender el comportamiento de los clientes y a mejorar su satisfacción.

7. ¿Cuáles de las siguientes son formas válidas de ejecutar una celda en un Notebook de Jupyter? Seleccione todas las que correspondan.

- [x] Pulsando "Mayúsculas + Intro".

  Correcto
  Correcto 'Mayús + Intro' ejecuta la celda actual y pasa a la siguiente.

- [ ] Pulsando 'Alt + R'.
- [x] Pulsando el botón "Ejecutar" de la barra de herramientas.

  Correcto
  Correcto El botón "Ejecutar" de la barra de herramientas puede utilizarse para ejecutar una celda.

- [x] Seleccionando "Ejecutar todo" en el menú "Celda".

  Correcto
  Correcto la opción "Ejecutar todo" del menú "Celda" ejecuta todas las celdas de la libreta.

8. ¿Cuáles de las siguientes son formas válidas de añadir una nueva celda en un Notebook de Jupyter? Seleccione todas las que correspondan.

- [x] Pulsando 'B'.

Correcto
Correcto Pulsando 'B' se añade una nueva celda debajo de la celda actual.

- [ ] Pulsando 'C'.
- [ ] Pulsando 'M'.
- [x] Pulsando 'A'.

Correcto
Correcto Al pulsar 'A' se añade una nueva celda encima de la celda actual.

9. ¿Qué función de los Notebooks de Jupyter permite mostrar texto enriquecido, ecuaciones y visualizaciones directamente dentro del notebook? Seleccione la mejor respuesta.

- ( ) Células brutas
- ( ) Comandos mágicos
- (•) Celdas Markdown
- ( ) Células de código

Correcto
¡Correcto! Las celdas Markdown de los Notebooks de Jupyter te permiten incluir texto enriquecido, ecuaciones y visualizaciones en tus análisis.

10. Estás creando un Notebook de Jupyter para analizar los datos de ventas de tu empresa. Quieres permitir a los usuarios filtrar los datos por categoría de producto y ver los resultados actualizarse dinámicamente dentro del Notebook. ¿Qué función interactiva de Notebook de Jupyter te permitiría lograr esta capacidad de filtrado dinámico? Selecciona la mejor respuesta.

- ( ) Plantillas
- ( ) Marcadores
- (•) Widgets
- ( ) Macros

Correcto
¡Correcto! Los widgets te permiten añadir controles interactivos a tus cuadernos.