Bienvenidos!
Semana 3

Datos No Estructurados y Semiestructurados

Especialización en Economía, opción Ciencia de Datos

FCS - Udelar

Guillermo Lezama

La clase de hoy

- HTML
- Scraping and APIs
- Una intro a texto
- Actividad grupal y Proyecto Final

- ¿Dudas del cuestionario o alguna de las tareas?

- Comentarios sobre Ejemplos

HTML

- HTML (*HyperText Markup Language*) es el lenguaje con el que están construidas la mayoría de las páginas web.

- Actividad con un txt. Este es un párrafo.

- → etiqueta de apertura

- class="intro" → atributo

- "Este es un párrafo." → contenido textual

¿Por qué nos importa en análisis de datos?

- Los datos que necesitamos están en páginas web.
- Podemos usar HTML como fuente de datos no estructurados.
- Podemos extraer texto, títulos, tablas, listas, etc.

En XML:

- Las etiquetas (<ciudad>, <nombre>) definen estructura y contenido de los datos.
- Los atributos (<ciudad nombre="Montevideo">) agregan información adicional dentro de la etiqueta.
- En XML se espera que las etiquetas describan datos y no estilo.

En HTML:

- También usa etiquetas y atributos, pero la mayoría están pensadas para presentación visual (dónde mostrar algo, cómo se ve, qué hace un botón, etc.)
- Por eso hablamos de etiquetas de presentación:
- <h1>, <p>, <b>, <table>, <img>
- Atributos como style=, class=, width=, href=
- En HTML, las etiquetas no describen datos, sino cómo deben mostrarse.

JSON, XML y HTML

- Son las formas comunes de representar datos no estructurados o semiestructurados.
- Son la base para leer, extraer y procesar información desde la web o servicios (como APIs).
- Aparecen en casi todo: desde respuestas de APIs, hasta documentos legislativos o páginas de noticias.

JSON, XML y HTML: Similitudes

- Todos son texto legible por humanos.
- Tienen una estructura jerárquica o anidada.
- Podemos recorrerlos con Python (diccionarios, árboles, selectores).

JSON, XML y HTML: Diferencias

- Estructura (clave → valor, etiquetas y atributos y etiquetas de presentación)
- Json.loads(), ElementTree, BeautifulSoup

API

- API significa *Application Programming Interface*. Es una forma estructurada de pedir datos a un servidor.
- Muchas páginas web y servicios ofrecen APIs para que puedas consultar datos directamente en formato JSON.
- En vez de hacer scraping del HTML, simplemente pedís datos como si fuera una "conversación de robots".

¿Por qué usarlas?

- Devuelven datos limpios (sin etiquetas HTML)
- Más estables y rápidas que el scraping
- Muchas veces pensadas para ser reutilizadas en apps o análisis

- Usamos `requests.get()` para consultar la URL del servicio.

¿Qué son los datos de texto?

- Datos no estructurados compuestos por caracteres, palabras y oraciones.
- Ejemplos: noticias, tuits, libros, respuestas de encuestas.

Texto: Flujo de trabajo básico

- Pre-procesamiento
  - ‘Hola’ no es lo mismo que ‘hola’
  - ‘de’ o ‘y’ son palabras muy comunes que no aportan información
  - Puntuación, símbolos y números pueden interferir.
- Transformar y extraer información
  - Identificar frecuencia de palabras clave.
  - Encontrar frases relevantes o expresiones comunes.
  - Clasificar textos según sentimiento o categorías temáticas.
  - Crear representaciones gráficas, como nubes de palabras.
- Analizar la información

Proyecto Final

- Similar a actividad grupal.
- La diferencia: si van a trabajar con los datos.
- Objetivo: Transformar un conjunto de datos desestructurados en algo estructurado + contar algo de esos datos + articularlo con una historia a contar.
- 9 de Junio: Contar qué piensan hacer
- Entrega: 31 de Julio
- 3 opciones: Texto, imagen, Sonido.
- Grupos de a 4.
- Fin de semana **(ANTES DEL LUNES DE NOCHE)**: formulario sobre qué formato quieren trabajar
- Agregar si hay algún tipo de texto, imágenes o sonidos que quieran trabajar

Para próxima clase

- Cuestionario
- Llenar formulario
- Tareas del notebook
- Tarea adicional en mi notebook Guillermo.ipynb