

## Taller de la Ley de Zipf

Este taller tiene como objetivo que los estudiantes comprendan y apliquen el análisis léxico y fonológico de un conjunto de canciones de diversos géneros musicales, explorando la frecuencia de los tokens en los textos y cómo se comportan conforme a la Ley de Zipf. A través de este proceso, los estudiantes aprenderán a realizar análisis textuales, implementar tokenización, y aplicar métricas de textometría para interpretar el comportamiento de los datos.

### Objetivos del Laboratorio

Los estudiantes aprenderán a:

1. Conocimiento básico de Python y manejo de entornos como **Google Colab**.
2. Familiaridad con las bibliotecas **pandas**, **matplotlib** y **nlTK** para análisis de texto.
3. Conocimiento básico sobre la Ley de Zipf y sus aplicaciones en el análisis de frecuencias de palabras.

---

### Requisitos Previos

- Conocimiento básico de Python y manejo de entornos como **Google Colab**.
- Familiaridad con las bibliotecas **pandas**, **matplotlib** y **nlTK** para análisis de texto.
- Conocimiento sobre la Ley de Zipf en el análisis de frecuencias de palabras.

---

### 1. Análisis del Conjunto de Canciones

Cada estudiante (o grupo) deberá:

- Seleccionar al menos 3 canciones de artistas de diferentes géneros musicales.
- Extraer la letra de las canciones y realizar un análisis léxico, tokenizando el texto para estudiar la frecuencia de las palabras.
- Comprobar si la distribución de frecuencias sigue la Ley de Zipf, donde las palabras más frecuentes tienen una frecuencia inversamente proporcional a su rango.

---

### 2. Análisis Fonológico y Fonético

- Utilizar la librería **fonemas** para realizar un análisis fonológico y fonético del conjunto de canciones.
- Tokenizar y transformar los datos para obtener las representaciones fonéticas.
- Comparar cómo los resultados fonéticos se distribuyen en función de la frecuencia de las palabras.

### 3. Análisis Textométrico con la Librería Textstat

Seleccionar al menos 5 métricas de textometría para evaluar las letras de las canciones, algunas son:

- Flesch Reading Ease
  - Gunning Fog Index
  - Automated Readability Index
- 

### 4. Colaboración y Documentación en Google Colab

- Crea un **Google Colab** donde se implementen todos los análisis: desde la tokenización, pasando por el análisis fonológico y fonético, hasta las métricas de textometría.
  - Organizar y comentar el código para garantizar la claridad del proceso.
- 

### 5. Entrega y Evaluación

Al finalizar el taller, cada estudiante (o grupo) deberá entregar en el **CampusVirtual** lo siguiente:

**Informe en PDF** que incluya:

- Descripción de las canciones seleccionadas y el análisis realizado.
- Resultados del análisis léxico (frecuencia de tokens y validación de la Ley de Zipf).
- Resultados del análisis fonológico y fonético utilizando la librería **fonemas**.
- Resultados de las métricas de textometría utilizando **textstat**.
- Comparación de los resultados y discusión sobre los patrones encontrados en las canciones.
- Reflexión sobre la utilidad del análisis de texto y fonología para entender las características lingüísticas de las canciones.

**Google Colab** con los ejercicios implementados en formato **Jupyter Notebook**.

---

### 6. Recursos Adicionales

- **Fonemas:** [Librería Fonemas en PyPI](#)
- **Textstat:** [Librería Textstat en PyPI](#)
- **Ley de Zipf + código:** [Explicación sobre la Ley de Zipf](#)