**Explicación del Código**

1. **Importaciones y Configuración**:
   * Importamos numpy para la generación de datos y matplotlib.pyplot para la visualización.
   * Importamos time para medir el tiempo de generación.
2. **Medición del Tiempo de Generación**:
   * Capturamos el tiempo antes y después de la generación de los datos para medir cuánto tiempo tarda el proceso.
3. **Generación del Dataset**:
   * n\_samples es el número de muestras que queremos generar.
   * n\_features es el número de características de cada muestra.
   * n\_classes es el número de clases para el problema de clasificación.
   * Generamos características usando np.random.randn, que crea una matriz de muestras de una distribución normal estándar.
   * Generamos etiquetas aleatorias usando np.random.choice.
4. **Visualización de los Datos**:
   * Creamos una función plot\_data que utiliza matplotlib para visualizar los datos generados.
   * La función muestra un scatter plot con diferentes colores para cada clase.
5. **Mostrar Información del Dataset**:
   * Imprimimos la cantidad de datos generados y el tiempo de generación.
6. **Llamada a la Función de Visualización**:
   * Finalmente, llamamos a la función plot\_data para mostrar los datos generados.

Este código te proporciona una base sólida para generar y analizar datasets sintéticos desde cero en Python. Puedes ajustarlo y expandirlo según tus necesidades específicas.