Generación de datos sintéticos y almacenamiento de información

Francisco Jurado, francisco.jurado@uam.es

Objetivos de aprendizaje:

- Aprender a generar datos sintéticos para realizar una ingesta de datos coherente con las necesidades reales del sistema a medir.
- Aprender a almacenar la información en diversos tipos de ficheros y Sistemas Gestores de Bases de Datos (SGBD).
- Aprender a automatizar el proceso de generación de datos sintéticos de forma que el código pueda reutilizarse con diferentes cantidades y formatos de datos.

Descripción:

En el marco de esta práctica, los estudiantes deberán desarrollar e implementar un código que permita la generación de datos sintéticos, así como su almacenamiento en diferentes formatos de ficheros y Sistemas Gestores de Bases de Datos (SGBD).

En particular, los estudiantes deben implementar el código que permita:

1. Generar dos conjuntos de datos de tamaño parametrizable empleando el paquete Faker[1], donde uno de ellos tenga datos relacionados con el otro:
    1. el primero contendrá datos de usuarios (nombre, dni, email, número de teléfono móvil, número de teléfono fijo, dirección, ciudad, código postal[2] y provincia[3]).
    2. el segundo contendrá datos de vehículos (matrícula, número de bastidor, año, fabricante, modelo, categoría – urbanos, sedán, berlina, cupé, descapotable, deportivo, todoterreno, monovolumen SUV–) que hayan pertenecido o pertenezcan a usuarios, empleando el DNI como identificador del usuario.
        1. un usuario puede estar relacionado con más de un vehículo.
        2. es posible que deba implementar su propio código para generar parte de la información.
        3. puede resultar útil revisar la documentación de la función `random_element`[4].
2. Almacenar la información generada en ficheros:
    1. CSV y Parquet[5]: para ambos formatos, se generará un fichero por cada conjunto de datos (usuarios y vehículos) conectando ambos conjuntos de datos por medio de referencias (el campo de la variable de un conjunto de datos será el identificador del otro conjunto de datos).
    2. JSON[6] y Avro[7]: se deberá abordar la tarea de dos modos diferentes, a saber:
        (a) un único fichero para todo el conjunto de datos anidando los datos relacionados de forma jerarquica.
        (b) un fichero por cada conjunto de datos (usuarios y vehículos) conectando ambos conjuntos de datos por medio de referencias (el campo de la variable de un conjunto de datos será el identificador del otro conjunto de datos).
    3. Señala las características más destacadas de cada formato de fichero. ¿Usarías cualquier formato de forma indistinta en cualquier situación? ¿En qué situaciones resultan más aconsejables cada uno de ellos? ¿Qué diferencias encuentras entre usar formatos de

[1] https://faker.readthedocs.io/en/master/index.html
[2] https://raw.githubusercontent.com/inigoflores/ds-codigos-postales-ine-es/master/data/codigos_postales_municipios.csv
[3] https://es.wikipedia.org/wiki/Anexo:Provincias_de_Espa%C3%B1a_por_c%C3%B3digo_postal
[4] https://faker.readthedocs.io/en/master/providers/baseprovider.html#faker.providers.BaseProvider.random_elements
[5] https://parquet.apache.org/
[6] https://www.json.org/json-es.html
[7] https://avro.apache.org/

almacenamiento basados en filas frente a los basados en columnas? ¿Qué ventajas e inconvenientes encuentras entre almacenar la información mediante ficheros separados o en estructura jerárquica?

3. Almacenar la información generada en SGBD:
    1. SQLite3[8]: sistema de gestión de bases de datos relacional autónomo, sin servidor ni necesidad de configuración, que lee y escribe directamente en archivos de disco.
    2. PostgreSQL[9]: sistema de gestión de bases de datos relacional. Al contrario que el anterior, requiere de un proceso servidor independiente al que las aplicaciones cliente deberán conectarse.
    3. MongoDB[10]: sistema de gestión de bases de datos de documentos, que emplea colecciones en lugar de guardar los datos en tablas. Al igual que el anterior, requiere de un proceso servidor independiente al que las aplicaciones cliente deberán conectarse.
    4. Señala las características más destacadas de cada SGBD. ¿Usarías cualquiera de ellos de forma indistinta en cualquier situación? ¿Para qué situaciones usarías cada uno de ellos?

Debe evitarse el uso de Pandas en todos los apartados, indicando qué implicaciones y/o limitaciones tiene su uso en las diferentes tareas.

[8] https://docs.python.org/3/library/sqlite3.html
[9] https://pypi.org/project/psycopg2/
[10] https://www.mongodb.com/es