Project Proposal Report

Title: Resolving Open Issues in data-toolset Python Package

Topic Area: In-Memory & File Formats

GitHub Repository used: https://github.com/luminousmen/data-toolset/issues

Project Summary:

The current data-toolset utility is a tool for reading and writing a variety of file formats.

However there a lot of issues/bugs I found while testing the functionality which we will explain below and will propose further resolutions for each of these.

Issues in the existing Code:

1. File Format Compatibility:

The utility lacks the capability to read CSV or JSON files.

Implementation is needed to enable reading these formats within the utility.

2. All round File Format Conversion:

A function is required to convert between various file formats, including Avro, Parquet, CSV, and JSON.

The utility should support round-trip conversions among these formats.

3. Sample Data Generation Function:

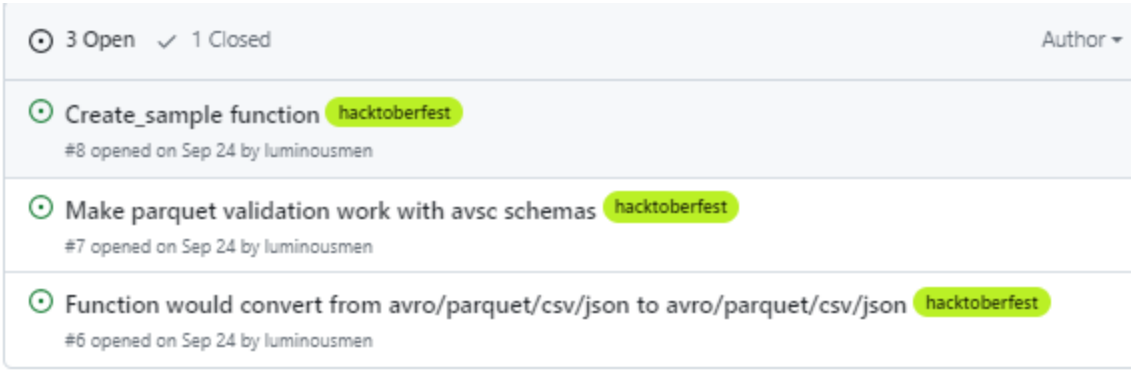
Development of a create_sample function is needed.

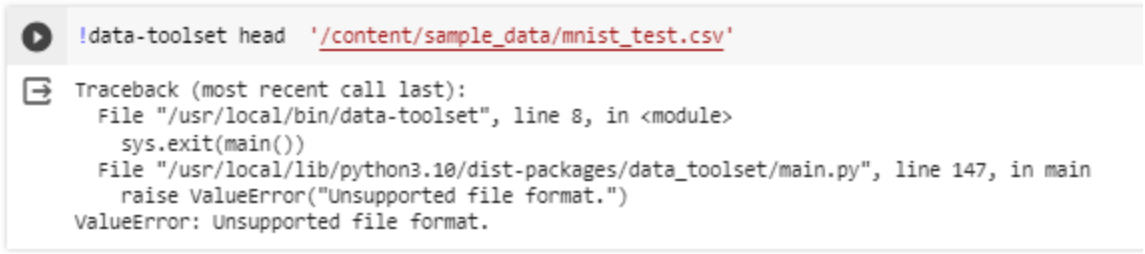
This function will generate synthetic data for testing, experimentation, or demonstration purposes.

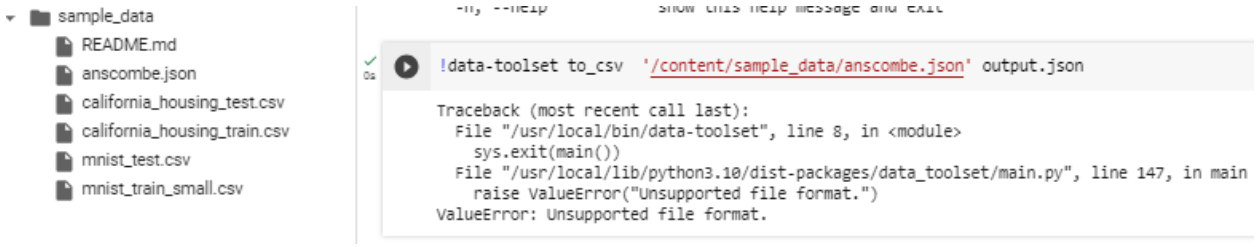
It should closely mimic real data and be applicable to Avro and Parquet utility classes.

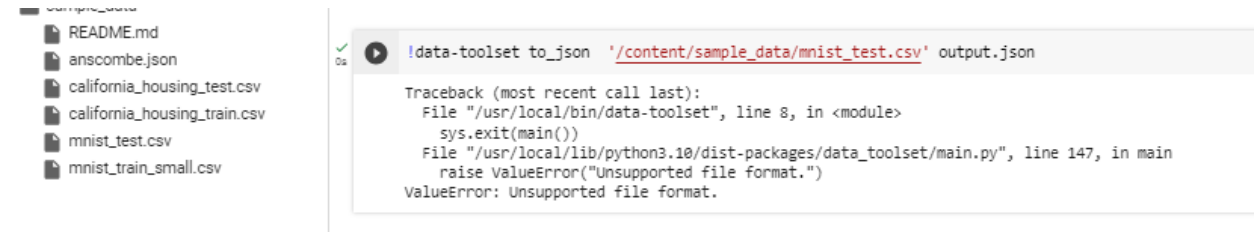
The function should accept a specified data schema or structure as input.

Here are some screenshot of issues-









Proposed changes and new features:

1. Enhancing File Format Compatibility:

Implementation is required to read CSV or JSON file formats.

2. Universal File Format Conversion:

A versatile function is needed to facilitate seamless conversion across multiple file formats, including Avro, Parquet, CSV, and JSON.The utility should support bidirectional conversions among these formats.

3. Introducing a Sample Data Generation Function:

The development of a create_sample function. This function aims to generate synthetic data suitable for testing, experimentation, or demonstration purposes.

Technologies:

Python

Future possible areas of enhancement:

- Adding support for additional file formats.
- Adding a few more file operations.
- Further deploying this package over Apache Hadoop and Spark for testing of humungous files having millions of records leveraging in-memory data processing, minimizing operation execution time, and enhancing the efficiency of the operations.
- Developing a GUI for the file manager utility package