University of Niagara Falls Canada
Master of Data Analytics
DAMO630 – Advanced Data Analytics
Assignment 1
Due date: Sunday of Week 3, 11:59 PM

Learning outcomes

1. Apply classical, statistical, and machine learning–based methods for generating synthetic data.
2. Evaluate synthetic datasets in terms of statistical similarity, utility, and privacy.
3. Implement big data mining workflows using HDFS, MapReduce, and PySpark in a personal cluster or a cloud environment.
4. Apply frequent itemset mining and clustering techniques on large-scale datasets.
5. Interpret results in terms of business value and decision-making.

Business Challenge 1: Privacy-Preserving Analytics with Synthetic Data

Organizations in healthcare and finance often cannot share raw datasets due to privacy and compliance constraints (e.g., HIPAA, GDPR). Synthetic data provides a way to generate realistic data without exposing sensitive records. You are part of an analytics consulting team hired by a healthcare startup. The startup needs synthetic patient data for research collaborations without violating privacy laws. Your task is to generate, evaluate, and analyze synthetic data that mimics real patient demographics and outcomes.

Task I: Exploratory Analysis of Real Dataset

Load the patient dataset from this link, and perform exploratory analysis:

1. Report distributions (numerical + categorical features).
2. Highlight privacy-sensitive attributes.
3. Visualize correlations between features.

Task II: Generating Synthetic Data

Apply a classical method (e.g., bootstrapping, noise injection, or rule-based generation using Faker) to create a synthetic dataset.

1. Compare distributions between real and synthetic data (histograms, summary statistics).
2. Discuss strengths and limitations of this baseline approach.

Task III: Advanced Synthetic Data Generation with SDV

Use the Synthetic Data Vault library to

1. Train at least two models (e.g., CTGAN and GaussianCopula) on the real dataset.
2. Generate synthetic datasets from each model.
3. Compare their statistical similarity to real data.

Task IV: Evaluation of Synthetic Data

Apply evaluation metrics:

1. Statistical similarity (KS test, correlation preservation)
2. Utility (TSTR: train on synthetic, test on real)
3. Privacy (row-level duplication check)
4. Interpret results in terms of business implications: How much can the startup trust synthetic data for external sharing?

Deliverables

A Jupyter Notebook with:

- EDA outputs and visualizations.
- Baseline synthetic dataset and advanced SDV-generated datasets.
- Evaluation metrics and comparison.
- Business interpretation: How synthetic data supports compliance and research use cases. (You can include these as formatted Markdown cells in your notebook)

Business Challenge 2: Mining NYC Taxi Trip Data with PySpark

The New York City Taxi & Limousine Commission (TLC) collects detailed records of every licensed taxi trip in NYC. These datasets contain millions of rides per year, including pickup/dropoff times and locations, passenger counts, trip distances, and fares.

City planners and transportation companies are interested in uncovering travel patterns and segmenting riders to improve services, reduce congestion, and optimize pricing.

The TLC has provided you with its trip data (Download any arbitrary month data from this link) to upload it to your Hadoop cluster or in a cloud environment, and you are tasked with analyzing this large-scale dataset using HDFS, MapReduce, and PySpark.

Task I: Big Data Setup & Exploration

Upload the taxi trip dataset (e.g., one month of TLC trip data) to HDFS and demonstrate:

1. Creating directories and uploading files.
2. Viewing contents in HDFS.
3. Perform an initial inspection with PySpark:
    a. Schema, row count, and sample records.
    b. Basic statistics (average fare, trip distance, passenger count).

Task II: MapReduce Analysis

1. Implement a MapReduce program to compute total fare revenue per pickup location (borough/zone).
2. Display partial outputs from the map, shuffle, and reduce steps.
3. Discuss the limitations of MapReduce in terms of speed and flexibility compared to Spark.

Task III: Frequent Travel Pattern Mining with PySpark

1. Use the FPGrowth algorithm in PySpark MLlib.
2. Treat each passenger trip as a "basket" of categorical attributes (e.g., pickup zone, dropoff zone, time-of-day bucket).
3. Identify frequent travel patterns (e.g., "rides from Midtown to JFK in the morning").
4. Report the top 10 association rules with support, confidence, and lift.
5. Interpret results as urban mobility insights (commuting flows, airport routes, tourism hotspots).

Task IV: Rider Segmentation with PySpark K-Means

1. Apply K-Means clustering to group trips or riders based on:
    a. Average trip distance
    b. Average fare amount
    c. Typical time-of-day for rides
2. Report cluster centers and interpret them as rider personas (e.g., "short-trip commuters," "airport travelers," "late-night riders").
3. Suggest how taxi companies or city planners could tailor services/pricing for each cluster.

Deliverables

A Jupyter Notebook with Markdown cells that includes:

- HDFS commands and outputs.
- MapReduce program and results.
- PySpark FPGrowth results with interpretation.
- K-Means clustering with cluster insights.
- Business interpretation: How mining at scale supports strategic decisions.

DAMO630- Rubrics of Assignment 1

Criteria	Excellent (85–100%)	Very Good (70–84%)	Good (60–69%)	Fail (<60%)
Exploratory Data Analysis (EDA)	Comprehensive analysis of healthcare dataset: distributions, correlations, privacy-sensitive features; clear visuals and insights.	Covers most attributes with some plots and comments; minor gaps in insight or clarity.	Basic descriptive stats or plots provided, but superficial insights.	No meaningful EDA of the dataset.
Baseline Synthetic Data Generation	Correct implementation of classical method (e.g., bootstrapping/Faker). Clear comparisons with real dataset; strengths/limitations discussed.	Method applied correctly, but limited analysis or weak comparison.	Attempted but incomplete or with shallow evaluation.	No baseline synthetic dataset generated.
Advanced Synthetic Data with SDV	At least two SDV models (e.g., CTGAN, GaussianCopula) implemented on cloud. Comparisons and differences clearly reported.	One SDV model applied with some evaluation; partial use of metrics.	Attempted but incomplete or misapplied; weak comparisons.	No SDV model used.
Synthetic Data Evaluation & Business Insight	Applies similarity, utility (TSTR), and privacy checks thoroughly. Results interpreted in healthcare context.	Metrics applied correctly but with limited business framing.	Partial evaluation (e.g., only statistical similarity) with weak interpretation.	No meaningful evaluation or business insight.
MapReduce Task (Taxi Data)	Correct fare-per-zone calculation; intermediate + final	Correct implementation with partial outputs	Attempted but incomplete or incorrect; weak	No working MapReduce program.

	outputs shown; limitations explained.	shown; explanation somewhat limited.	discussion of limitations.	
PySpark Frequent Pattern Mining	FPGrowth correctly applied; top 10 rules with support/confidence/lift; clear urban mobility insights.	Implementation correct; top rules shown, but interpretation is limited.	Attempted but incomplete results or unclear interpretation.	No FPGrowth results.
PySpark Clustering (Rider Segmentation)	K-Means correctly implemented; clusters described with personas (e.g., commuters, airport riders); implications discussed.	Implementation correct but cluster interpretation is somewhat limited.	Partial clustering with little or no interpretation.	No clustering attempted.
Business Insight & Reporting	Notebook is professional, reproducible, results interpreted in business/operational terms.	Well-structured report but weaker business linkage or clarity.	Technical results included but little business framing.	No coherent reporting or interpretation.