upGrad

ETL Assignment:
Bank Data Analysis

Assignment ID: ETL/02
Total Marks: 200

Problem Statement

Objective

In this case study, you will work on banking data analysis using Apache Spark, a powerful distributed computing framework designed for big data processing. This assignment aims to give you hands-on experience in analysing large-scale banking data sets using PySpark. You will apply techniques learnt in data analytics to clean, transform and explore banking data, drawing meaningful insights to support financial decision-making. Apart from understanding how big data tools can optimise performance on a single machine and across clusters, you will develop a structured approach to analysing market capitalisation trends, currency conversions and global banking performance.

Business Value:

The banking industry operates in a highly competitive and globalised market where financial institutions must continuously monitor their market position, performance and currency fluctuations. To stay competitive, banks must leverage data-driven insights to optimise their financial strategies, assess market trends and make informed decisions. In this assignment, you will analyse banking data to uncover patterns in market capitalisation, currency conversions and global rankings. With Apache Spark's ability to handle large datasets efficiently, financial institutions can process vast amounts of data in real time, which helps them make faster and more informed decisions.

As an analyst at a financial institution, your task is to examine historical banking data to derive actionable insights that can drive strategic growth. Your analysis will help identify the top 10 largest banks by market capitalisation, convert market cap values into multiple currencies (USD, GBP, EUR and INR) and store the processed data for easy retrieval. By leveraging big data analytics and cloud services, financial institutions can streamline operations, enhance decision-making and maximise revenue opportunities.

Dataset Overview

Context

The dataset used in this analysis comprises global banking data extracted from a Wikipedia page (List of Largest Banks) archived as of September 2023. It includes rankings of the world's largest banks based on market capitalisation (in USD billions) and is structured into a single table with three columns. The data was scraped programmatically, cleaned and transformed using PySpark to support cross-currency analysis. Exchange rate data (for USD, GBP, EUR and INR) was sourced from a CSV file to enable currency conversions. The processed dataset is stored in CSV and SQLite formats for accessibility and efficient querying.

Content

- The dataset consists of a single table (Largest_banks) with the following three key attributes.
- Key attributes include:
  - **Rank**: Numerical position of the bank based on market capitalization.
  - **Bank Name**: Name of the financial institution.
  - **Market Cap (USD Billion)**: Market valuation of the bank in USD billions.
- This dataset enables analysis of global banking trends, cross-currency valuations, and the relative market dominance of financial institutions.

Acknowledgements

This dataset is sourced from publicly available information on Wikipedia and is intended for educational and analytical purposes.

Download

The dataset files can be accessed here in the next section.

Scoring and Penalty

- **Total Marks**: **200** (130 for code notebook and 70 for report)
- **Extension and Penalty**: As given in your learner handbooks

Instructions

1. Each learner should attempt this assignment individually.
2. Programming Language: Python
3. You will be provided with the dataset and a starter notebook. You have to perform analyses in the starter notebook only.
4. It is very important that you do not change any headings, subheadings, questions or tasks in your notebook as it can cause problems with grading.
5. For analyses and processing tasks, you should use only the following libraries: NumPy, Pandas, Matplotlib, Seaborn, and Plotly.
6. The data will have inconsistencies and outliers please handle them as per your understanding and mention them in your report.
7. You are encouraged to search the web and consult AI tools for conceptual understanding. However, using plagiarized or AI-generated code is strictly prohibited and strongly discouraged.
8. Submitting plagiarized and AI-generated code or reports will result in significant penalties to your scores.

Submission Guidelines

1. You are required to upload your solution files in the submission field.
2. You are required to submit **two** files:
   (a) an **Interactive Python Notebook** (.ipynb) that contains your code
   (b) a **Report Document** (.pdf) that presents your visualisations, analysis, results, insights, and outcomes.
3. Note that these files should only be generated from the starter files provided to you.
4. Both your Jupyter notebook and report should contain your name and the assignment title in the format: `"ETL_Bank_Data_Analysis_<your_name>"`
5. Mention all assumptions made in the report.
6. Your answers to all the tasks mentioned in the starter notebook should be present in the report. Any graphs/plots you generate for analysis should also be attached to the report.

Results Expected from Learners

Present the overall approach of the analysis in a report document. Mention the problem statement and the analysis approach briefly.

In the starter notebook, you will find headings, subheadings, and checkpoints stating the tasks you need to perform. The marks associated with each checkpoint will also be mentioned in the notebook. Keep in mind not to edit the cells with marking schemes and questions. You can find a brief description of the tasks below.

1. **Data Preparation [5 marks]**

   The dataset consists of a structured table containing global banking data. Before performing any analysis, it is crucial to prepare the data to ensure consistency, and efficiency in processing.

   (a) Check for data consistency and ensure all columns are correctly formatted.

   (b) Apply sampling techniques if needed to extract a representative subset for analysis.

   (c) Structure and prepare the data for further processing and analysis.

2. **Data Cleaning [20 marks]**

   (a) Fixing Columns [5 marks]

   (b) Handling Missing Values [10 marks]

   (c) Handling Outliers [5 marks]

   Hints:

   - Note that it is not necessary to replace the missing value in EDA, if you have to replace it, what should be the approach? Mention the approach.
   - Identify if there are outliers in the dataset. Also, mention why you think it is an outlier. Again, remember that for this exercise, it is not necessary to remove any data points.

3. **EDA: Finding Patterns [40 marks]**

   (a) Convert PySpark DataFrame to Pandas DataFrame for visualization.

   (b) Analyze the distribution of market capitalization using a histogram.

   (c) Identify the top 10 banks by market capitalization using a bar chart.

   (d) Visualize the relationship between market capitalization and bank ranking using a scatter plot.

   (e) Use a boxplot to examine the spread and outliers in market capitalization.

   (f) Display the quartile distribution of market capitalization using a violin plot.

   (g) Compute cumulative market share and visualize it with a line plot.

   (h) Categorize banks into market capitalization ranges and analyze their distribution using a bar chart.

   (i) Calculate and display market share distribution of top 10 banks using a pie chart.

4. **Banking Data ETL and Querying [55 marks]**

   (a) Extract banking data from Wikipedia and process it using PySpark.

   (b) Transform banking data by converting market capitalization values into multiple currencies.

   (c) Save the transformed dataset as CSV files and Load the processed dataset into an SQLite database for structured querying.

   (d) Perform Advanced Market Capitalization Analysis with Growth Metrics.

   (e) Analyze Market Concentration and Categorize Banks Based on Market Share Tiers.

   (f) Examine Statistical Distribution of Market Capitalization Using Quartile Analysis.

   (g) Conduct Comparative Size Analysis to Classify Banks by Relative Market Size.

   (h) Evaluate Market Growth and Identify Gaps Between Consecutive Banks.

   (i) Assess Market Dominance by Measuring Cumulative Share and Dominance Score.

   (j) Analyze Segment-Wise Bank Performance Based on Market Capitalization Ranges.

(k) Generate a Comprehensive Performance Dashboard for Bank Rankings and Metrics.

5. **Conclusion [10 marks]**

Final insights and recommendations:

(a) Recommendations to track and compare market capitalisation of the top global banks to evaluate competitiveness and dominance.

(b) Suggestions to use cross-currency analysis (USD, GBP, EUR, INR) for consistent benchmarking of financial institutions across regions.

(c) Propose continuous monitoring of market share concentration to identify growth opportunities for mid-tier banks.

(d) Identify potential regions or banking segments for expansion by analysing gaps between tiers of banks and regional trends.

Points to note:

- Conclude the analysis by summarizing key findings and business implications.
- Explain the results of univariate, segmented univariate, bivariate analysis, etc. in business terms.
- Include visualisations and summarise the most important results in the report. You are free to choose the graphs which explain the numerical/categorical variables. Insights should explain why the variable is important.

6. **Visualization Integration [Optional]**

Enhance the project by incorporating a visualization component that connects the processed data stored in an S3 bucket to a business intelligence tool such as Tableau or Power BI. This involves:

(a) Setting up the connection between the S3 bucket and the chosen visualization tool.

(b) Importing the processed dataset for analysis and visualization.

(c) Creating interactive dashboards to explore key trends and insights.

(d) Ensuring data updates are reflected dynamically in the visualization tool.

Evaluation Rubrics

The following rubrics will be used while judging your solutions to the above tasks.

Table 1: Rubrics

Criteria	Meets expectations	Does not meet expectations
Data Understanding	1. All data quality issues are correctly identified and reported. 2. Wherever required, the meanings of the variables are correctly interpreted and written either in the comments or text.	1. Data quality issues are overlooked or are not identified correctly such as missing values, outliers and other data quality issues. 2. The variables are interpreted incorrectly or the meaning of variables is not mentioned.
Data Cleaning and Manipulation	1. Data quality issues are addressed in the right way (missing value imputation analysis and other kinds of data redundancies, etc.). 2. If applicable, data is converted to a suitable and convenient format to work with using the right methods. 3. Manipulation of strings and dates is done correctly wherever required	1. Data quality issues are not addressed correctly. 2. The variables are not converted to an appropriate format for analysis. 3. String and date manipulation is not done correctly or is done using complex methods

Continued on next page

Table 1: Rubrics (Continued)

Criteria	Meets expectations	Does not meet expectations
Data Analysis (EDA)	*These come from the analyses in your code and insights reported in your report.* 1. The analysis effectively addresses key business objectives, uncovering trends, patterns, and relationships in banking data. The structure is clear and logical. 2. The ETL process is correctly implemented, ensuring clean, structured data for analysis, including handling missing values and converting market capitalization into multiple currencies. 3. At least five key variables are identified (e.g., market cap, ranking, currency conversions), focusing on metrics relevant to financial insights. 4. Derived metrics are created (e.g., top 10 banks by market cap, aggregated market share) with clear justification and effective usage in analysis. 5. Univariate and bivariate analyses comprehensively identify meaningful trends and relationships in market capitalization, rankings, and currency fluctuations. 6. Insights are clearly communicated in comments/documentation, highlighting key patterns such as dominant financial institutions and market trends. 7. Well-designed visualizations are used to present findings effectively, with clear labels and relevance to conclusions.	*The code and report will be graded for following a reasonable order along with the notebook and implementing the points mentioned here.* 1. The analysis fails to address key business objectives or lacks a clear structure, making it difficult to interpret. 2. The ETL process is incomplete or incorrect, resulting in inconsistencies in the dataset. 3. Key variables are not identified, and the analysis lacks depth in exploring financial insights. 4. Derived metrics are missing, irrelevant, or not effectively used in the analysis. 5. Univariate and bivariate analyses are incomplete or fail to reveal meaningful relationships. 6. Insights are missing, unclear, or misinterpreted, overlooking important trends. 7. Visualizations are missing, poorly formatted, or unclear, making interpretation difficult.

Continued on next page

Table 1: Rubrics (Continued)

Criteria	Meets expectations	Does not meet expectations
Presentation and Recommendations	1. The report has a clear structure, is not too long, and explains the most important results concisely in simple language. 2. The recommendations to solve the problems or the outcomes and insights, whichever is applicable, are realistic, actionable and coherent with the analysis. 3. If any assumptions are made, they are stated clearly.	1. The report lacks structure, is too long or does not put emphasis on the important observations. The language used is complicated for business people to understand. 2. The recommendations to solve the problems or the outcomes are either unrealistic, non-actionable or incoherent with the analysis. 3. Contains unnecessary details or lacks important ones. 4. Assumptions made, if any, are not stated clearly.
Conciseness and readability of the code	1. The code is concise and syntactically correct. Wherever appropriate, built-in functions and standard libraries are used instead of writing long code (if-else statements, loops, etc.). 2. Custom functions are used to perform repetitive tasks. 3. The code is readable with appropriately named variables and detailed comments are written wherever necessary.	1. Long and complex code is used instead of shorter built-in functions. 2. Custom functions are not used to perform repetitive tasks resulting in the same piece of code being repeated multiple times. 3. Code readability is poor because of vaguely named variables or lack of comments wherever necessary.