Build a Real-Time GCP Pipeline for Household Data Analysis

Business Overview

A structured data pipeline is critical for collecting, processing, and analyzing household income and expenditure data, especially in economies transitioning from centrally planned to market-based systems, such as Eastern Europe and the former Soviet Union. This project uses scalable Google Cloud Platform services including Cloud Storage, Cloud Functions, BigQuery, and Looker Studio to manage large datasets efficiently through automated ingestion, transformation, and analysis. The goal is to examine income trends, rising inequality, and the effectiveness of social assistance programs in supporting vulnerable households and reducing poverty.

Aim:

The objective of this project is to examine household income and expenditure patterns across different regions and demographic groups in Eastern Europe and the former Soviet Union. It focuses on understanding income distribution, spending behavior, and the role of social assistance programs in transitioning economies. The analysis is supported by an automated Google Cloud pipeline designed to efficiently process and manage incoming data batches.

Tech Stack: ➔ Language: SQL, Python

➔ Services: Google Cloud Storage, Google Cloud Functions, Google BigQuery, Looker Studio

Google Cloud Storage:
Google Cloud Storage (GCS) provides scalable cloud-based storage for securely storing and managing raw data files.

Google Cloud Functions:
Google Cloud Functions is a serverless compute service that executes code automatically in response to events, eliminating the need to manage infrastructure.

Google BigQuery:
Google BigQuery is a fully managed, serverless data warehouse that enables fast and cost-efficient SQL queries on large-scale datasets.

Looker Studio:
Looker Studio is a data visualization platform used to create interactive dashboards and shareable reports from structured datasets.

Data Description:
The dataset contains multiple columns grouped into categories such as household income, expenditure patterns, asset ownership, taxation details, and demographic information across different countries.

Unique Identifier (Essential Column)

hhid: This column represents the unique ID for each household.

Expenditure Categories: These columns represent the expenditure patterns of households:

foodx: Expenditure on food items.

healthx: Health-related expenditures, such as medical services and medications.

rentx: Expenses incurred for rent.

transx: Costs associated with transportation, such as public transport or fuel.

clothx: Expenditure on clothing items.

housex: Costs related to housing, including utilities and rent.

educulx: Educational expenses, such as tuition and books.

otherx: Miscellaneous expenditures not categorized in other columns.

Income Categories: These columns represent the income sources for households:

wagey: Total wage income from employment.

wagemy: Monthly wage income.

wageky: Income from wages of other household members.

selfemy: Earnings from self-employment or personal business activities.

totpeny: Total pension income received by the household.

familyy: Financial support received from family members.

socassy: Social assistance received in the form of government benefits.

unempy: Unemployment benefits received by household members.

othsocy: Other social benefits that are not specified elsewhere.

asoctry: Position of a household in the social structure, based on factors like income, education, and occupation.

imrenty: Income from renting out a property or other assets.

othery: Other income sources not covered in other categories.

Demographic and Regional Information: These columns provide demographic and regional details about households:

amenita: Availability of amenities, such as water supply and sanitation, in the household.

landa: Agricultural land owned by the household.

local: Type of locality where the household is situated (1 - Capital, 2 - Other City, 3 - Rural).

region1: Region or geographical location of the household. **seg:** Socioeconomic group classification of the household.

hhsize: Household size, defined as the number of members in the household.

Taxation and Utility Information: These columns represent the taxes paid by households and other related utility information:

sstaxy: State or sales tax paid by the household.

pitaxy: Personal income tax paid by the household.

othtaxy: Other taxes applicable, such as property tax, VAT, or local taxes.

Asset and Property Categories: These columns describe the assets and properties owned by households:

durabla: Durable goods owned, such as vehicles or appliances.

carda: Ownership of a car or other vehicles.

tvclda: Ownership of a television or similar devices.

refigda: Ownership of a refrigerator.

tenanca: Ownership of land or agricultural properties.

Note: We recommend monitoring GCP Services and deleting those that are not in use after project execution, as they can result in high costs.

Approach:

1. A project was created in the GCP console, which allows seamless creation of other GCP services

2. The raw household data collected was loaded into Google Cloud Storage (GCS) as the central storage for efficient data management.

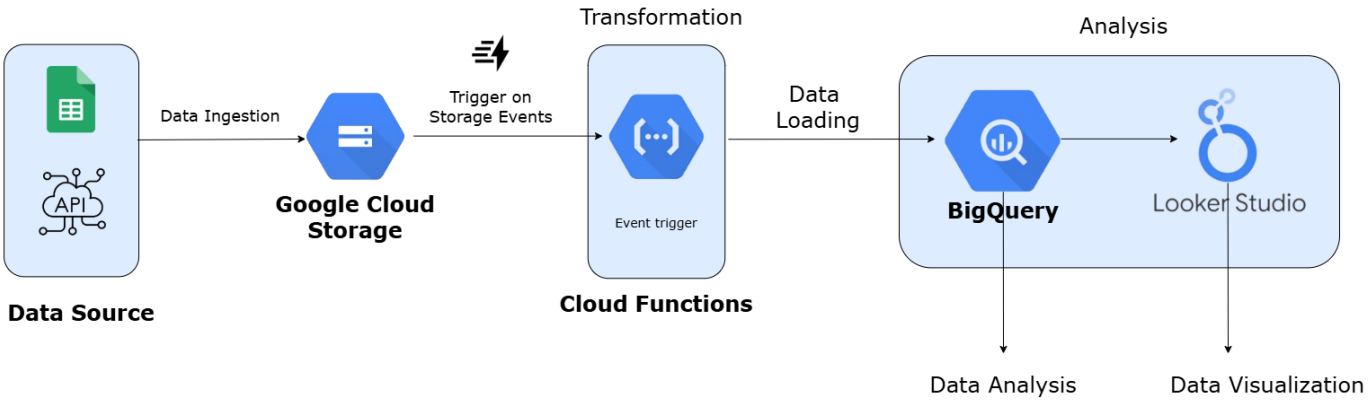
3. Cloud Functions was set up to trigger automatically whenever new data files are uploaded to GCS. These functions transformed the data into the required format and prepared it for analysis.

4. The transformed data was then loaded into a Google BigQuery table for analysis.

5. Looker Studio was used to create interactive visualizations from the BigQuery table, to create visualization insights such as poverty rate comparisons and trends across countries.

Note: We recommend monitoring GCP Services and deleting those that are not in use. The unused resources, such as Cloud Functions, storage buckets, or BigQuery, might continue to incur charges even when not in use.

Architecture Diagram:



Key Takeaways:

Creation of a Google Cloud Storage bucket for storing raw household data

How to set up event triggers in Cloud Functions to process new data uploads?

Implementation of Google Cloud Functions for automated data transformation

Understanding of transformation scripts to clean and structure raw data

Integration of Cloud Functions with BigQuery for data loading

Creation of BigQuery datasets and tables to manage transformed data

Understanding of SQL queries in BigQuery to perform analysis

How to connect BigQuery with Looker Studio for data visualization?

Creation of interactive dashboards in Looker Studio using different visuals

How to use filters, scorecards, and charts to enable dynamic visualization

Understanding how to analyze income and expenditure patterns through visualization layers