Data stack for startup

# Introduction

Building a data driven environment is important for modern businesses. As the SaaS and cloud compute getting popular, there are more and more options to build data ecosystems. Among the various options, implementing a cost-efficient solution could be more beneficial than a powerful solution to small businesses or startup. Hence, this article is going to propose a simple data ecosystem using Google Cloud Platform (GCP) and open-source tools.

# Architecture

A screenshot of a computer

Description automatically generated with low confidence

There are three key components in the proposed ecosystem, which are storage, analytic, and report.

## Storage

Storage is the foundation of the entire ecosystem. It collects raw data and transforms into severable information supporting analytics or reports. To develop the storage feature, we must (1) build data in-flow pipelines, (2) design data schema, and (3) manage and monitor the data quality.

To build the data pipeline, it is necessary to understand the data volume and immediacy. The general guidelines are that the small data volume can implement cloud run or cloud function, while the big volume can use data proc or spark. The low immediacy data can also use cloud run, but high immediacy data might have to implement pub/sub or another message queue.

After loading data into the storage, the raw data must be transformed into valuable information. The most important concept in this stage is to level the data. For example, we can set three levels of data: raw, research, and serving. Raw level represents the original data. This type of data usually needs feature engineering to obtain insights. Research level is the experimental features. In many cases, analysts test different hypothesis and make up various metrics; however, the definitions or hypotheses could be changed. As a result, this level serves informative data but still under experiment. The serving data is the finalized version, and it should be the most stable and informative data.

Continuously improving the data quality and monitoring the data pipeline are critical because low quality of data pipeline is unreliable. To managing the data pipeline, we suggest that the developer should explicitly designed tests to ensure data pipeline and data quality. Each test can be host on cloud run invoke before any data in-flow or transformation.

## Analytics

Analytics is a process to generate insights, informative metrics, or algorithms. To make the analytics effective, the ecosystem must concentrate on (1) reproducibility, and (2) deployment during the design stage.

Reproducibility enhances the stability and allows people to share the same knowledge. However, many data science algorithms could produce different results due to their randomness or the unexpected new data. People could have different results or insights toward the same scenario even if they use same algorithm. It will potentially become a technical debt if people cannot setup a benchmark comparing their approaches. Consequently, it is relevant to have a general framework ruling each stage of the analysis.

Deployment is a process which transform the analytical project into a maintainable routine work. Since many data represents only part of the real population, it is possible that the analysis results would not work in the future. Deploying a analytical project could help people tracking their hypothesis. Moreover, the routine work could also deliver actions such as the prediction of machine learning models. Those actions are usually valuable for decision making or reducing operation costs. Hence, the deployment process should have good availability and can be continuously improved.

## Report

Report is one of the most efficient ways to track the result. Its function includes recognizing business insights, detecting anomaly, and validating analytical hypothesis. To fully utilize the report, we suggest that (1) build comprehensive data models and (2) understand the value before design the report.

It is possible for a company having multiple reporting tools such as Tableau or Apache Superset. Since different tools might have different structure of data connection and schema, designing a base data can significantly reduce the unnecessary work. For example, we can use the concept of data levels, mentioned in storage section, to reduce the data transformations. As the data transformation reduced in the report tools, the business meaning of each feature across various tools can be controlled.

Different tools could have different advantages. For instance, the traditional business intelligence tools are very convenient to create reports for tabular data. However, if the report requires complicated manipulation or handle unstructured data, Dash or Streamlit could hold more advantages.

## Proposed Architecture

Given the three key components, we suggest that the storage should be managed by BigQuery, and the input pipeline can be managed by Cloud Run according to the data source nature. Subsequently, we can use Cloud Build and API Gateway to manage these input pipeline. The scheduling tasks can be done by Cloud Schedule. In this architecture, we can enjoy the discount from GCP, and therefore reduce significant amount of costs.

Kedro is a data science framework which enables users to organize and reproduce their analytical results. Their main features include building data pipeline, generate documentations, and deployment services connection (i.e., kubeflow). Given this framework, we have a standard format of sharing research projects and can easily deploy on GCP Vertex AI or Kubernetes.

Although GCP Looker is a great business intelligent, its expense might be not cost-efficient. As a result, we suggest that the tabular data can be reported by Google Data Studio or PowerBI. If the complicated manipulation or interaction required, Streamlit and Dash can be very powerful.