

# Spotfire-Assignment-Dashboard

Dashboard link: <https://spotfire-next.cloud.tibco.com/spotfire/wp/OpenAnalysis?file=1773e29b-62ce-4c03-add9-4bd5b5087637>

\*please make sure to trust scripts to see js script for KPIs and other python datafunctions

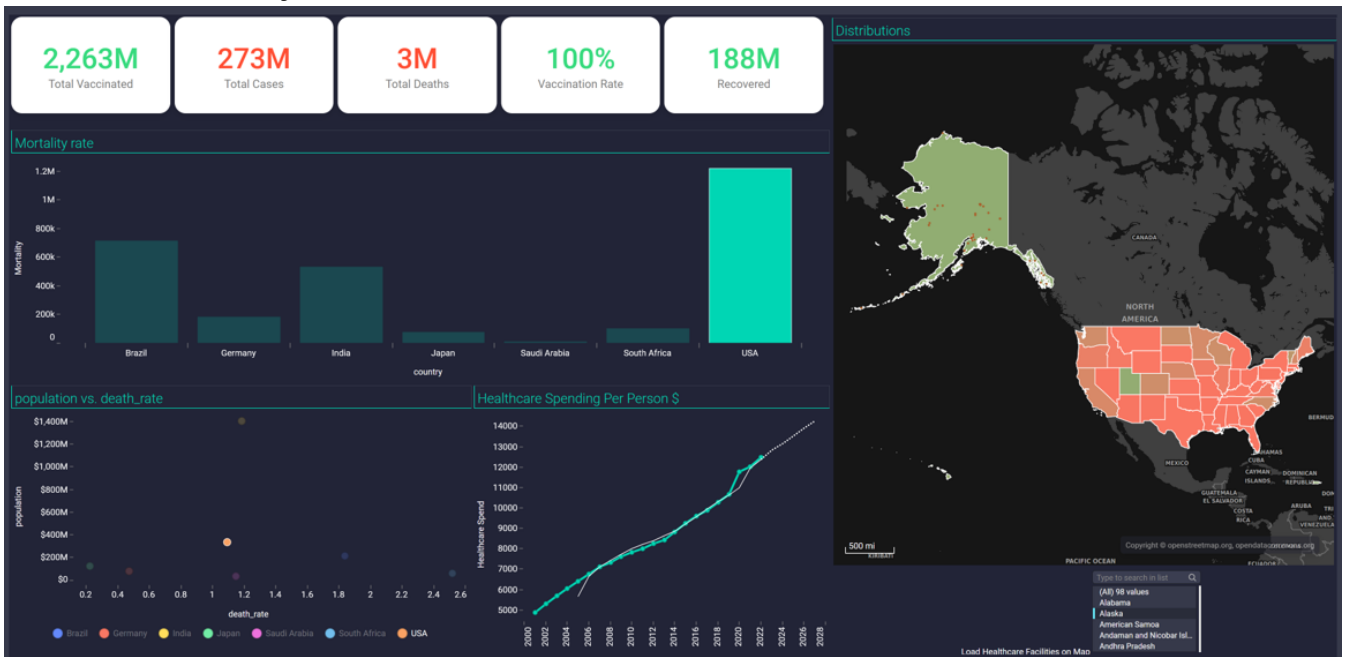
## Objective

Analyse COVID-19 and mortality based on healthcare spending and vaccination across multiple countries , focusing details on USA and India data from COVID-19 Data API.

1. Analyze Global COVID-19 Impact:
  - A. Visualize total cases, deaths, recoveries, and vaccinations at a high level.
  - B. Compare mortality rates across different countries.
2. Understand Healthcare Correlation:
  - A. Examine the relationship between healthcare spending per person and mortality or death rates.
  - B. Identify if countries with higher healthcare expenditure have lower death rates.
3. Identify Trends and Outliers:
  - A. Evaluate population vs. death rate patterns through scatter plots.
  - B. Forecast for next 5 years
4. Geospatial Distribution:
  - A. Map COVID-19 cases, vaccination rates, or healthcare facilities by country.
  - B. Highlight hotspots and regions with higher/lower healthcare investment.

## Dashboard Snapshots

Tab 1: Covid Mortality and Healthcare



Tab 2 : EDA



## Data

Data is loaded from

### 1. COVID-19 Data API

#### A. CountrywiseCovid19

Countrywise data for selected countries (India,USA,Brazil,Germany,Japan,South Africa,Saudi Arabia)

#### B. Statewise Covid19

Statewise data for India and USA

### 2. World Bank API

#### A. WorldBankData

Year wise economic indicators for {selected countries}

#### B. OpenStreetMap (OSM)

##### a. Healthcare facilities

hospitals and clinics for {drop down selected states} for India and USA

##### b. Excel data embedded

##### i. country\_code\_mapping

Country mapping for API data and maps

## Model

CountrywiseCovid19 api data table is left joined with country\_code\_mapping excel data.

-This is one to one relationship

Other API data tables are mapped by editing table relationships on state or country.

-All of this are many to one relationship or vice versa

Transformations used: Filter rows, exclude columns

## Custom R/Python functions

Mainly used for connecting data from various API. Used only built in packages like http.client, json etc. Data is manipulated to convert from JSON to spotfire row major column type tables. Tried Predictions using sklearn but sklearn is not available by default to import, so used spotfire's default forecasting.

## Architecture of the dashboard

User interactions

- Markings are used to filter data from main visual
- Markings used for loading api data online
- Filter used for healthcare facilities
- On EDA all filters as included which have a different filtering scheme to separate from main analysis

Big Picture

- Get data from various public API get requests, made time consuming api on demand when user interacts.
- Joined necessary tables
- Tables which doesn't have a unique id or clear join natures are virtually joined using relationships.

## Spotfire infrastructure

- Spotfire Analyst : desktop client for dashboard development, data manipulation and advanced analytics
- Spotfire Web Player – web based client that allows users to interact with dashboards through a browser.
- Spotfire Server – Server where spotfire is hosted. The central component that manages users, licenses, and access control.
- Data Connectors – Facilitate connections to databases, cloud storage, and big data systems.
- Spotfire Node Manager – Handles scaling by distributing workloads across multiple nodes.

Scaling Consideration

- Vertical Scaling: Increase CPU, RAM, and storage of the existing Spotfire Server
- Horizontal Scaling: Add more Spotfire Web Player instances or Spotfire Servers to distribute load
- Cluster Configuration: Deploy Spotfire in a clustered environment to ensure high availability and failover
- Session Pooling: Optimize session pooling to handle more users by reusing Web Player sessions efficiently.
- Resource : Set resource quotas to prevent a single user or dashboard from consuming excessive resources