**COVID-19, Weather & Census Analysis Dashboard**

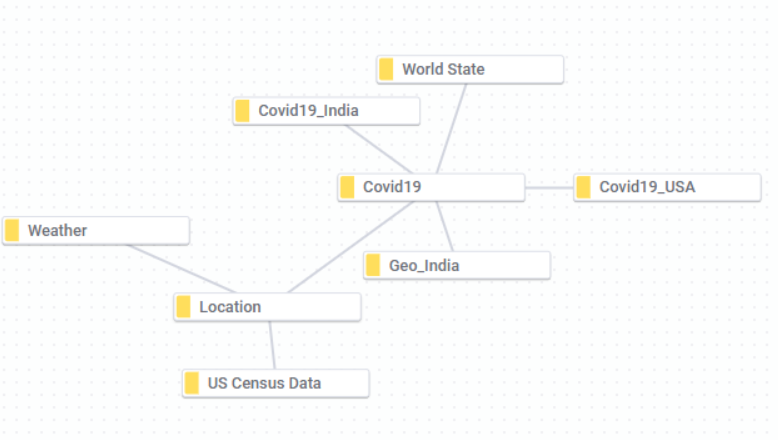
This project aims to build a dynamic, interactive dashboard in TIBCO Spotfire by integrating data from multiple open APIs. The dashboard includes predictive modeling, geospatial visualizations, calculated metrics, and user-driven interactivity for exploring Census, Temperature, COVID-19 trends and insights.

**Data Sources**

1. **OpenStreet API**
   * **API**: Location API
   * **Format**: JSON/XML
   * **Purpose**: Get the City, state, country, lat, lon
2. **Weather API**
   * **API**: OpenWeatherMap
   * **Format**: JSON
   * **Purpose**: Fetches historical weather data to compare temperature, humidity sunrise, sunset, etc.
3. **COVID-19 API**
   * **API:** [COVID Tracking Project / OWID / Rootnet / etc.]
   * **Format:** JSON
   * **Purpose:** Tracks confirmed cases, deaths, recoveries, testing rates
4. **US Census API**
   * **API:** Official U.S. Census Bureau
   * **Format:** JSON
   * **Purpose:** Tracks Population, age, income, house rent, etc.

**Data Integration & Relationships**

**Composite Data Model**

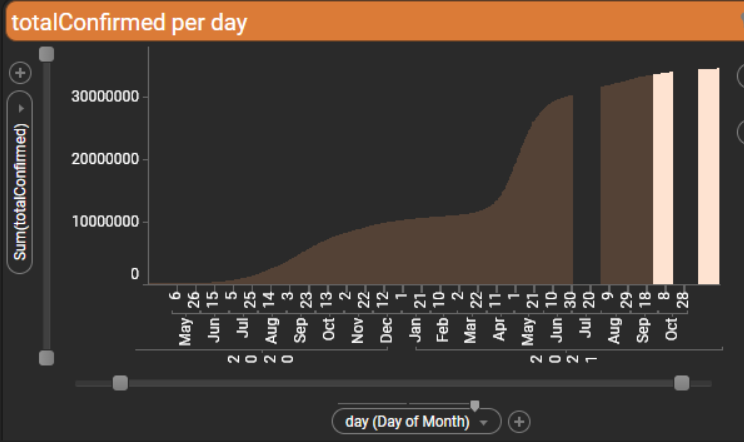
* Created using Spotfire’s **Data Canvas** and **relationships**.
* Joined on common keys:
  + Covid19\_India and Geo Code on state name for Geo data
  + Location and weather on lat, lon for state, city and country information.
  + Covid19\_USA with state table on state code to get state name
* Appended Covid19\_India and Covid19\_US\_data
* Relationship types:
  + Location ↔ Weather: **One-to-Many**
  + Location ↔ Covid19: **One-to-Many**
  + Location ↔ US Census: **One-to-Many**
  + Covid19 ↔ World State: **Many-to-One**
  + Covid19 ↔ Geo\_India: **Many-to-One**
  + Covid19\_India ↔ Covid19: **Many-to-Many**
  + Covid19\_USA ↔ Covid19: **Many-to-Many**

**Data Cleaning and Transformation**

* **Handling Missing Data**: Used Spotfire's Replace Nulls and Fill Down features.
* **String Parsing**: Removed special characters, extracted city names using calculated columns.
  + **Temp\_Cel and feels\_like**: Checked for null values and temperature unit and converted it into Celsius.
  + **Date, Day and Time**: Calculated from time stamp.
  + **Name:** Concatenatedthe city, country and region.
  + **State:** Removed special characters (\*, #) from the column and corrected the spelling mistake for the standardized state names for column matching.
* **Imputation**: In case of no records of precipitation in weather API,
  + imputed null values to 0 using python.
  + Predicted confirmed, discharged and death cases for the whole month for Covid19\_India table.
* **Date Normalization**: Converted UNIX timestamps with time zone offsets as per the region, standardized to YYYY-MM-DD before the data is loaded in python.
  + Sunrise = datetime.datetime.utcfromtimestamp(data['city']['sunrise']+timezone\_offset)
* **Derived Columns**:
  + Covid daily recovered, discharged and death cases from cumulative data.
  + Temperature in Celsius
  + Day, Date, Year from timestamp
  + Concatenated state and city
  + GDP

**Exploratory Data Analysis**

Performed EDA to detect outliers using scatter plots and analyzed mean values of temperature, humidity, wind speed, and COVID-19 daily cases to identify trends and anomalies across different cities and dates.

* **Missing Data**: COVID-19 data for June 2021 and November 2021 was missing in the Covid19\_India table. These gaps were addressed through prediction using regression models during the modeling phase
* **Inconsistent State Names**: The State column in the Covid19\_India table contained special characters and incorrect spellings. String cleaning and standardization were applied during the preprocessing stage.



* **Temperature Outliers**: Unusual temperature values were initially detected as outliers. Upon inspection, these were found to be in Kelvin. The values were converted to Celsius during preprocessing after unit verification.

**Custom R/ Python Data Functions**

1. **Data Loading (Python)**
   1. Location API: Utilized an API to fetch geographic metadata such as state name, country, latitude, and longitude based on user-input city names.
   2. TemperatureAPI: Takes latitude and longitude from Location API as input and retrieves 5-day/3-hour interval meteorological data (e.g., temperature, humidity, wind speed) for the user-specified cities.
   3. Covid19\_India: Retrieves cumulative daily state-wise COVID-19 data for India from 2020 to 2021, including confirmed, recovered, and death cases.
   4. Covid19\_USA: Provides daily cumulative COVID-19 statistics for each U.S. state from 2020 to 2021, covering confirmed cases, recoveries, and deaths.
   5. US Sensus: Retrieves annual state-level demographic and socio-economic indicators from 2018 to 2023, including population, median income, age, housing values, race distribution, and unemployment rates using the U.S. Census Bureau ACS API.
2. **Time Series Forecasting (Python)**
   1. Library: Scikit-learn
   2. Algorithm: Linear Regression
   3. Target: Predicting daily Confirmed, Discharged and Death cases
   4. Output: Projected values for the 1 month
   5. Description: Used **simple time-based linear regression** to **predict missing daily COVID-19 statistics** (dailyConfirmed, dailyDischarged, and dailyDeaths) **per state** in India for July and Nov 2021 using June and Oct 2021 data as the training window.
3. **Clustering (Python)**
   1. Library: Scikit-learn
   2. Algorithm: K-Means Clustering
   3. Target: Grouping similar records based on numerical features
   4. Output: Cluster labels assigned to each input record
   5. Description: Applied unsupervised **K-Means clustering** to segment the dataset into n\_clusters groups using numerical features. The model parameters like **init, n\_init, max\_iter, and tol** were tuned for **optimal convergence**, and cluster labels were added to the dataset for downstream analysis or visualization.

**Dashboard Features**

* **Performance Optimization**
  + Filtered and fetched only the cities provided by the user as input, reducing unnecessary API calls and data load, resulting in improved efficiency and faster processing.
* **Dynamic Filters**:
  + Drop Down (Cities)
  + Location (State/UT)
  + Date Range
  + Case Type (Confirmed / Deaths / Recovered)
* **Custom Header & Layout**:
  + HTML/CSS injected using Text Areas
  + JavaScript Components Used:
    - Header- Displays the main heading of the COVID-19 analysis page.
    - Card- Summarizes key weather metrics such as latest temperature, humidity, "feels like" temperature, wind speed, etc.
    - Days- Provides a 3-hour interval temperature and weather condition forecast (5-day overview).
    - Search card- Allows users to input and search city names with dropdown selection for user-friendly navigation.
    - Sun card – Displays sunrise and sunset timings in a visually styled layout.
* **Drilldown Capabilities**:
  + Select the city → Filter state of that city on second page
  + Click on state → filters visuals
* **Geospatial Layers**:
  + Choropleth + Markers based on severity

**Visualizations**

**How It Works:**

* **Step 1**: Enter **all locations** (cities) you want to load data for and click **Search button**.
* **Step 2**: It is required to use the **dropdown** to select **one specific location** and **click GO**. This selected location will populate the data on the overview and drilldown pages.
* The pages — **Weather**, **Covid19 Drilldown**, and **US Census Drilldown** are **interconnected**.
* All three pages will display data based on the **selected location**.
* The **Covid19** and **US Census** pages show **full datasets** for deeper analysis of the selected city/state.

**Page 1:**

**Location Filter**

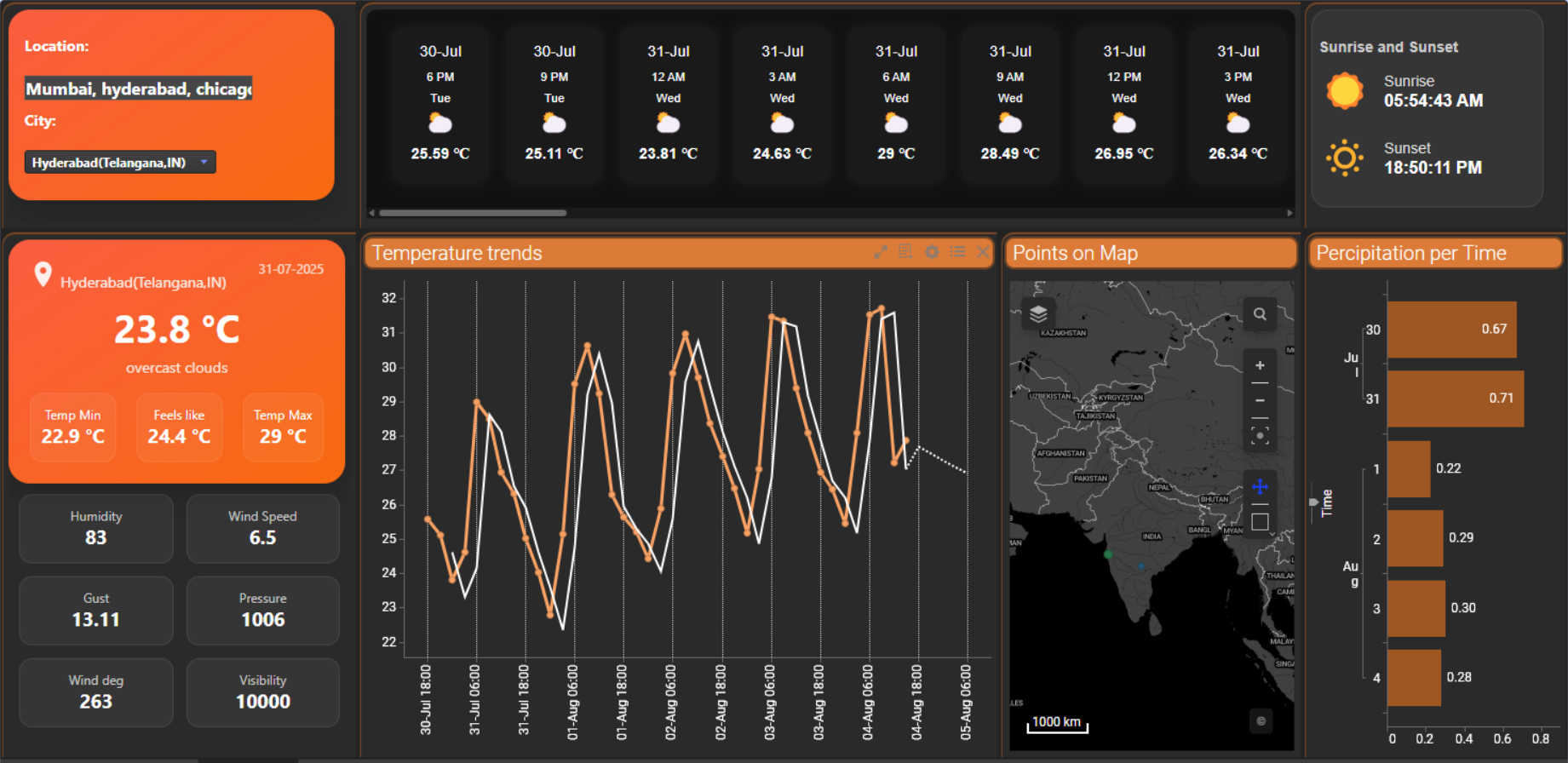
* Let’s users input and select multiple cities (e.g., Mumbai, Hyderabad, Chicago).
* Enables filtering and dynamic update of weather details and charts based on selected locations.

**Weather Forecast Cards**

* Shows 3-hour interval forecasts for temperature and weather conditions (icons) from 30th July to 31st July.
* Provides a short-term outlook to help users visualize upcoming temperature fluctuations and sky conditions.

**Sunrise and Sunset**

* Displays the local sunrise and sunset time for the selected city.
* Useful for understanding daylight hours, which impacts weather analysis and planning for outdoor activities.



**Current Weather Snapshot**

* Shows current weather conditions like temperature, humidity, wind speed/direction, pressure, visibility, etc., for the selected city.
* Offers a complete snapshot of real-time atmospheric parameters useful for immediate weather monitoring.

**Temperature Trends (Line Chart)**

* Line chart displaying variation in temperature over time for the selected cities.
* Helps identify patterns, peaks, and drops in daily temperatures to assess consistency or volatility.

**Points on Map**

* Plots the selected cities on a map using latitude and longitude.
* Geographical visualization helps in spatial analysis and better understanding of location-specific weather patterns.

**Precipitation per Time (Bar Chart)**

* Displays the predicted precipitation in millimeters for each day.
* Helps track rain trends and identify wet or dry periods, useful for planning and environmental monitoring.

**Page 2:**

**Total Confirmed Gauge**

* Displays the total number of confirmed COVID-19 cases
* Helps provide a quick snapshot of the pandemic's overall impact across selected regions.

**Discharged Gauge**

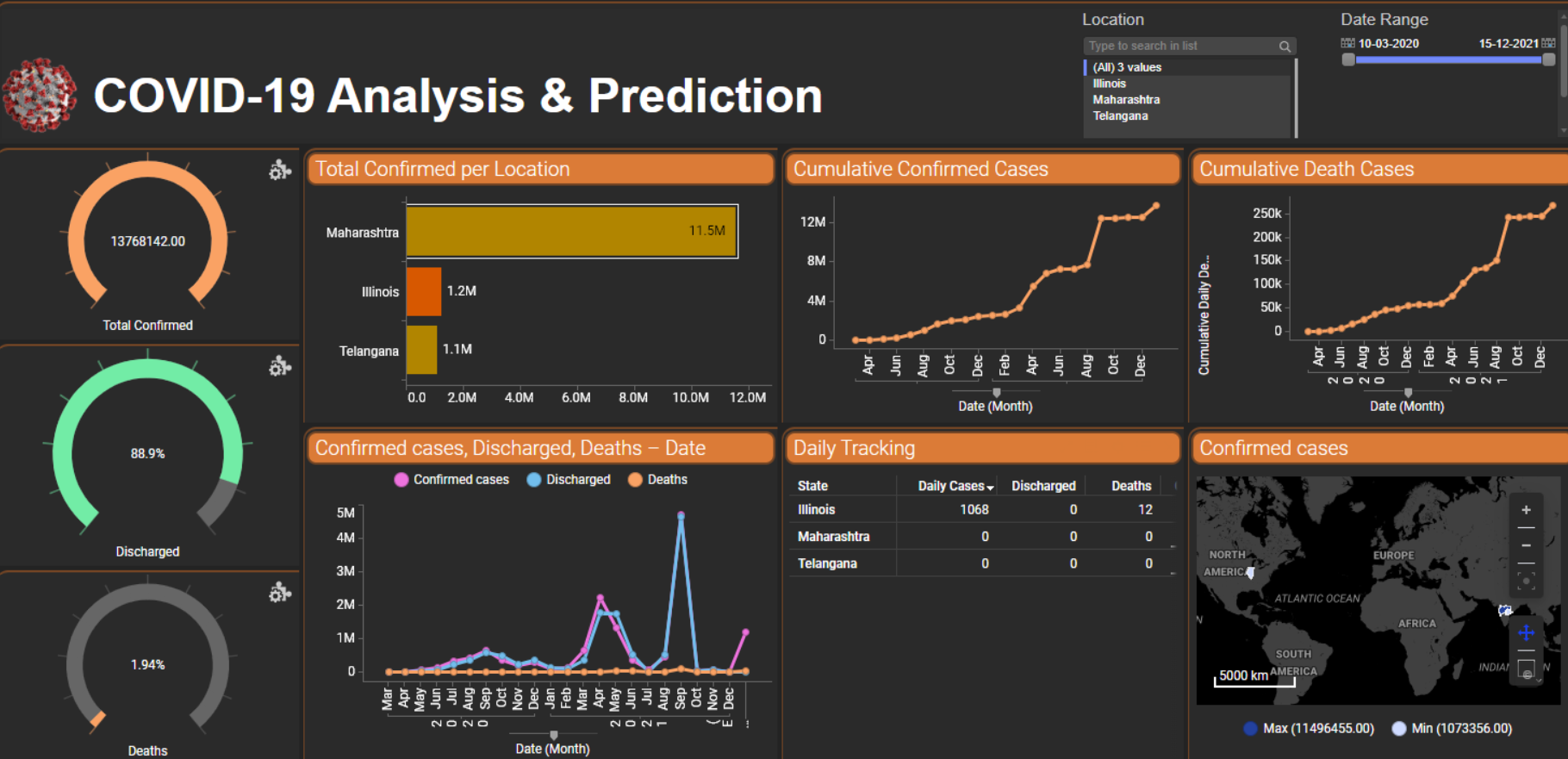
* Shows the percentage of recovered/discharged patients from the total confirmed cases.
* Indicates recovery rate effectiveness and gives insights into healthcare response.

**Deaths Gauge**

* Displays the death rate as a percentage of total confirmed cases.
* Highlights the fatality rate, essential for assessing pandemic severity and healthcare strain.

**Total Confirmed per Location (Bar Chart)**

* Compares the total number of confirmed cases across different selected states.
* Useful to identify hotspots and understand which regions are most impacted.

****

**Cumulative Confirmed Cases (Line Chart**)

* Shows the growth trend of cumulative confirmed cases over time.
* Helps track the progression of the outbreak and assess peaks/waves.

**Cumulative Death Cases (Line Chart)**

* Displays the accumulation of death cases over time.
* Reasoning: Allows monitoring of mortality trends and evaluating the effectiveness of interventions.

**Confirmed, Discharged, Deaths Over Time (Multi-line Chart)**

* Plots confirmed, recovered, and death cases together over time.
* Reasoning: Offers a comparative view of how these key metrics change in relation to one another.

**Daily Tracking Table**

* Tabular representation of daily confirmed, discharged, and death cases per state.
* Enables detailed tracking and supports day-to-day monitoring of new cases and trends.

**Confirmed Cases Map (Geographical Visualization)**

* A map highlighting confirmed case concentrations by location.
* Provides spatial understanding of virus spread and regional intensity.

**Date Range & Location Filter Panel**

* Allows selection of specific states and date ranges for dynamic filtering.
* Enables targeted analysis by location and time period for better decision-making.

**Page 3:**

### **LEFT PANEL: Filters**

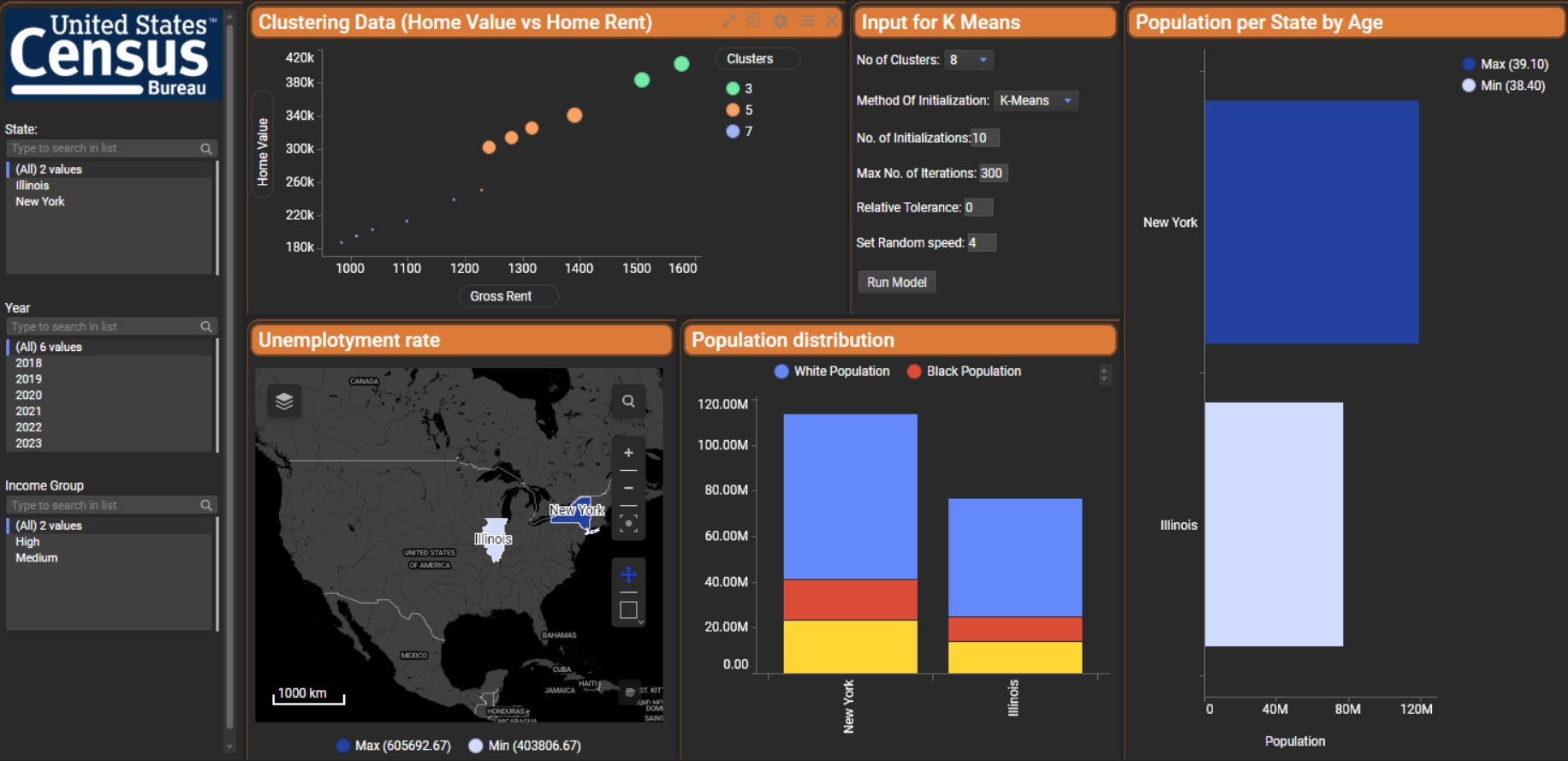
* **State, Year, Income Group**
  + User input controls to filter all visualizations across the dashboard
  + Interactivity ensures comparisons (like New York vs. Illinois) are based on user-defined selections and subsets of ACS data.

**Clustering Data (Home Value vs Home Rent)**

* A scatter plot where each point represents a location (state, county, or tract), showing the relationship between **gross rent** and **home value**.
* **Color by**: Cluster number (e.g. 3, 5, 7)
* It reveals how different regions cluster economically — areas with high home values also tend to have high rents, but clustering helps detect **outliers**, **affordability gaps**, and **distinct economic segments**.

## **Input for K-Means**

* An interactive form to define K-means clustering model parameters:
  + Number of clusters
  + Initialization method
  + Iteration limits
  + Random seed, etc.
* It allows users to dynamically explore how changing parameters affects the clustering logic used in the **scatter plot above**.

****

## **Population per State by Age**

* Vertical bar chart comparing total **population by state**, segmented by **age bracket or average age** (based on the color legend showing Max and Min).
* This helps compare population sizes and aging trends. A higher average age could indicate an older population, influencing policy needs like healthcare, retirement planning, or labor force shifts.

## **Unemployment Rate**

* A **choropleth** or shaded map showing unemployment rates by state (e.g., Illinois and New York are highlighted).
* Offers geographic insight into which states are facing higher unemployment, supporting **labor market comparisons** or **policy impact evaluations**.

## **Population Distribution**

* A **stacked bar chart** showing the **racial breakdown** of population by state. Each bar is split between:
  + 🔵 White Population
  + 🔴Black Population

(Possibly more races if dataset includes them)

**Insights and Analysis**

**Weather**

* **Forecast Trends:** Hourly and 4-day temperature forecasts reveal **cyclical weather patterns**.
* **Sunrise/Sunset:** Daily light hours could relate to **public activity patterns or solar planning**.
* **high humidity + temperature** might correlate with **disease patterns** or **energy demands**.

**Covid 19**

* A high discharge rate with low death rate indicates effective clinical management.
* **Death Rate:** Very low mortality (~1.94%), suggesting manageable fatality levels despite high case volume.
* Trend charts can be used for **predictive modeling** — estimating future outbreaks or resource needs.

**US Census**

* **Clustering (Rent vs Home Value):** Strong correlation; higher rent aligns with higher home values. Clustering reveals economic segmentation.
* **Unemployment:** New York and Illinois show different rates, highlighting **regional economic differences**.
* **Population Distribution:** New York has a **higher population overall**, with more racial diversity (larger Black and White populations).
* **Age Analysis:** New York has a higher average age — could impact workforce trends or aging-related services.

**Github Repository**

* **Link:** [**https://github.com/sagar221996/AIQ\_Assignment.git**](https://github.com/sagar221996/AIQ_Assignment.git)
* **Contents**:
  + [Assignment\_Dashboard.dxp](https://spotfire-next.cloud.tibco.com/spotfire/wp/OpenAnalysis?file=7c47867f-ac86-46c2-923f-77b2e9e0a54d)
  + Solution.doc
  + scripts/ (R, Python)
    - K\_Means\_Cluserting
    - India Predicted values