# A Data-Based Approach to Flattening the Curve

By: Oluwaseun Ademiloye, Tye Robison, Paul Lee, and Vijay Fisch

**1. The Scope** **1**
**2. Data Exploration Steps** **1**
**3. Defending Our Statistics** **2**
**4. Implementation of Statistics** **2**
**5. Policy Recommendations** **2**
**6. Auxiliary Data and Further Research** **3**
**7. Appendix** **4**
a. Executive Summary 4

## 1. The Scope

The challenge was to create comprehensive policy plans for mitigating and preventing the next wave of COVID-19 in Caladan - a midsize commonwealth with a total population of 3.2 million. We have addressed the future challenges posed by COVID-19 within the commonwealth of Caladan by evaluating the efficacy of a diverse array of COVID-19 policy options within a 10 country dataset. By evaluating established global policies, the ultimate objective is to determine "the most unrestrictive policies they [Caladan] can implement to keep the growth rate of deaths below 1% and the growth rate of new cases below 3% on a 30-day rolling average." Below, we explain our data-driven strategy that utilized PowerBI and machine learning tools in python to determine a suitable suite of policies for Caladan.

## 2. Data Exploration Steps

We've imported three datasets—"Recoveries," "Deaths," and "Cases"—into the VM metrics within our data lake. Before starting the analysis, we conducted the data exploration process, prioritizing data cleaning. This involved addressing unnecessary overlapping columns, such as country latitude and longitude redundancies in the geography data. In the "Deaths" dataset, we eliminated five columns: country_region, ISO2, latitude, longitude, and load time. Similarly, for "Recovery" and "Cases" datasets, we removed the same five columns via the dataflow.

Additionally, we relabeled ISO3 from deaths to CountryCode to make it easier on ourselves later during the processes. Furthermore, we identified null columns in the policy data sources, such as Region Name and Region Code, which were subsequently removed. In order to move on to the next process we created a new column with country code and date to make connections in our schema. Via this exploration, we determined the key columns we intended to use within our schema and the best way to create one-to-many relationships between columns.

### 3. Defending Our Statistics

We decided to analyze a specific statistical metric into our analysis, calculating the average percent decrease in growth for cases and deaths for each policy using PowerBI. This visual highlights the factors that exert the most significant influence within the metric but also offers a nuanced understanding of their relative importance. By calculating the average percent decrease across all possible values for each “Theme” (all the selected factors within the range), except for Usability – which is the selected influencer, we were able to gain a comprehensive perspective on the efficacy of various policies and rank them in terms of influence.

From a business understanding perspective, these statistics are essential to grasp the fundamental objectives and solution for the organization, Caladan, for this project. By aligning our ranked average percent decrease rate with the strategic goals of the organization, we can gain insights to address key issues and opportunities effectively. Through this methodology, we successfully chose the policies with the highest effectiveness and least restrictiveness as recommendations for Caladan’s covid policymaking.

### 4. Implementation of Statistics

We were able to implement these statistics by cross referencing the policies we had recognized to be the most effective via analyzing the growth rates of cases and deaths over the 30-day rolling average in PowerBI. We implemented all of the policies into the key-influencers graph and compared this with our previous findings to determine a majority of the data agreed with the analysis.

For example, we used PowerBI slicing and comparisons between the dates in which these policies were implemented, alongside the growth rates of deaths and cases. This led to the assumption that due to growth and death rates demonstrating decreasing values after the implementation of the H6_facial_coverings policy in multiple countries and overall, this policy was extremely effective. On that note, when countries introduced the H6_facial_coverings, we noted that the average of death and cases growth rates decreased by 2.61% and 2.01% respectively. The key indicators feature within PowerBI was able to bolster this assumption, not only for the sum over all countries, but also by displaying significant percentage decreases for individual nations which enacted this policy at varying degrees. We continued this process for the rest of our chosen policies.

Additionally, we sought for the time periods where death and case growth were at their peak and minimum. Around November 2020, deaths and case numbers peaked in most countries, but decreased significantly over the next several months. It was at these times where we analyzed what policies were enforced and at what stringency levels to understand which policies were the most effective at minimizing death and case growth rate.

### 5. Policy Recommendations

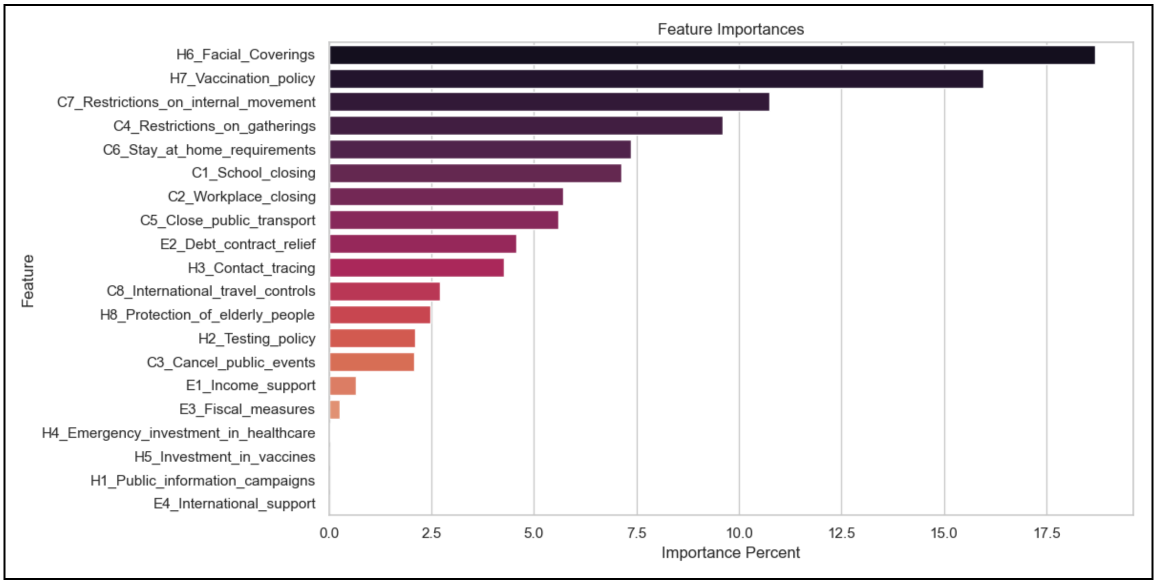
Based on the analysis of 10 countries' (United Kingdom, France, New Zealand, Russia, South Korea, Sweden, Canada, Japan, Italy, and Germany) policies, we have compared stringencies of two countries that had the strongest restrictive policies (Sweden and Great Britain) to two countries that had the weakest restrictive policies (France and Russia) within the time period given. Based on the analysis, the recommended policies we have chosen are H6_facial_coverings and H7_vaccination_policy. These 2 policies showed the most dramatic impact on keeping the growth rate of deaths below 1% and the growth rate of new cases below 3% on a 30-day rolling average. On a level of restrictiveness, the policy regarding H7_vaccination_policy showed the lowest degree of restrictiveness, so we recommend using this policy in Caladan as a first response. While the policy  H6_facial_coverings demonstrated a high level of effectiveness, we determined that this was not the top recommendation due to it being the highest level of restrictiveness in comparison to the other recommended policies. We also chose to note that C6_Stay_at_home_requirements was quite effective, however, it is the most restrictive overall by a much larger margin, so is not within our top recommendations.

## 6. Auxiliary Data and Further Research

To supplement our PowerBI analysis, we uploaded the cases and policy data into a Python JupyterNotebook to do a machine learning analysis utilizing the Scikit-Learn Random Forest Regressor. The regressor fits a chosen number of decision trees with your variables and target data, with a variety of hyperparameters available to fine-tune. The optimal solution is determined by taking the average of every tree's prediction. The main limitation of the regressor is that trees can grow too large and overfit data absent controls on the max_depth and min_samples_leaf attributes. Utilizing *train, test, split*, and tuning these parameters helps avoid overfitting.

Below are the steps for this part of the analysis:

1. In Jupyter Notebook, Connected *Cases* and *Policy* on UniqueID into one dataframe
    a. Eliminated duplicate, null rows
2. Imported world population data from UN 2022 Revision of World Population Prospects into *pop_df* dataframe
    a. Divided “confirmed” in merged cases/policy data frame by “population” in *pop_df* to make it proportional
3. Isolated columns for machine learning analysis (variables are policies, label/target is confirmed_per_capita)
4. Train, test, split (test size 80%)
5. Fit RandomForestRegressor on training data with target as label data, utilized hyperparameter tuning:
    a. *max_depth=10:* controls maximum depth of a tree, limiting overfitting. 10 is also not too shallow to underfit the data given the number of policies
    b. *n_estimators=50:* sets number of trees in the forest to 50, balancing model performance and computational efficiency
6. Evaluate efficacy on on test data, calculating regressor score (~68%)
7. Run feature importances, graph using SeaBorn



## 7. **Appendix**

### a. Executive Summary

The assumed roles prior to the project are as follows: Oluwaseun Ademiloye as Project Manager, Vijay Fisch as Data Architect, and Tye Robison & Paul Lee as Data Engineers. Although this organization structure was our initial plan for the distribution of work, it started off a little disconnected due to our differences in ability to meet up. Vijay handled the CosmosDB, Paul covered the Azure SQL data, and Oluwasseun took over gaining the data from the VM with the help of Tye who did research for how to apply the SHIP status to the VM.

As challenge 2 commenced, Olu and Tye struggled together to get the initial steps of challenge 2 completed, but they succeeded. In order to formulate the rest of the data flow, Vijay saved the day with his complex understanding of azure data factory, alongside Tye, they were able to construct the final data flow which allowed for 6 parquet files to all be evenly distributed into the ODS and ready for future processing. This is where Vijay's schema comes in. Olu and Tye sat down for an extended period of time going through each and every file in the data flow and renaming or deleting columns in order to preserve one-to-many relationships and remove many-to-many connections. After this dissection, we determined that the best way to connect the metrics data to the policy data (which was decided by Vijay to be our fact table) was to create a derived column with a UniqueID to be inputted into the Policies and all the metrics data sets. Vijay and Tye worked to use a DataFlow to concatenate the dates and the country codes into a single "UniqueID column in each dataset, ensuring that we could create a maintain many-to-one relationship between the policy set (fact table) and the metrics (dimension tables - deaths, recoveries, and cases).

For challenge 3, since we had already completed the schema and organized the data, Tye was able to quickly create the external tables. Later, Tye and Olu were able to upload all of this data to Power BI and they could begin challenge 4.

As challenge 4 commenced, all members began taking different roles and working on various tasks in order to ensure all the deliverables were gonna be completed on time. Tye created the data architecture and added greater detail to the data flow. She also is writing this section of the writeup and assisted Paul in creating the presentation slides. Paul worked tirelessly to create the slides presentation, complete the first half of the writeup, and create the code for the growth rates of cases and deaths in PowerBI. Oluwaseun worked alongside Paul in creating the Power BI graphs and visualizations which were later used to collect our findings. Additionally, to create the 30 day rolling average Tye and Vijay sat alongside Emma, David, and Josh for a few hours and eventually came to a successful solution. After this, the team worked together to write the code within a new measure to calculate the growth rates for both deaths and cases. Vijay had another task at hand, working tirelessly to feed our data into a Python machine learning model and supplement the PowerBI portion of the project–(hopefully) earning some creativity points and assisting the team with additional insights utilizing external data (UN population data).

We worked very effectively as a team and completed the presentation and deliverables days before the deadline.