Workshop No.1 — Systems Engineering Analysis

COVID-19 Global Forecasting
Kaggle Competition

Santiago Vargas Gomez - 20242020139
David Esteban Sanchez Torres - 20221020093
Dilan Guisseppe Triana Jimenez - 20221020100
Daniel Alejandro Castro - 20242020271
Course: Systems Analysis & Design
Professor: Carlos A. Sierra

September 27, 2025

1 Introduction

The Kaggle competition COVID-19 Global Forecasting addresses the challenge of modeling and predicting the evolution of the COVID-19 pandemic using historical time-series data. The dataset contains daily records of confirmed cases and fatalities reported across countries and provinces worldwide. By analyzing this data, participants are expected to design forecasting models capable of identifying patterns, trends, and trajectories of the pandemic. This initiative underscores the relevance of quantitative methods and predictive analytics as essential tools for understanding complex global health crises and supporting decision-making processes during emergencies.

2 Competition Overview

2.1 Goal

The primary objective of this competition is to forecast the cumulative daily number of confirmed COVID-19 cases and fatalities for each region. The task requires the application of time-series forecasting techniques and data-driven methodologies to generate reliable predictions that reflect the potential future course of the pandemic. These forecasts are intended not only to evaluate methodological approaches but also to emphasize the critical role of predictive modeling in informing public health strategies and global responses.

2.2 Dataset

The COVID-19 case and death prediction system can be understood as a set of interconnected elements that transform historical data into future forecasts. Each element fulfills a specific function, and the relationships between them allow information to flow from its source to become useful output for decision-making.

The starting point for the system is the dataset, which contains daily records organized by country and region. Each row contains variables such as date, geographic location, cumulative number of confirmed cases, and number of reported deaths. This dataset forms the basis of the system and is the essential input for all subsequent processing.

The competition provides three main files that form the foundation of the system. The **train.csv** file contains the training data, with the columns `Id`, `Province_State`, `Country_Region`, `Date`, `ConfirmedCases`, and `Fatalities`, which allow modeling the evolution of the

pandemic over time and across different regions. The **test.csv** file specifies the dates and locations for which predictions must be made, including the columns `ForecastId`, `Province_State`, `Country_Region`, and `Date`. Finally, the **submission.csv** file serves as a template for the correct submission format, where participants are expected to report, for each `ForecastId`, the estimated number of `ConfirmedCases` and `Fatalities`. Together, these files define the inputs, outputs, and basic rules of the information flow within the system.

The target of the system is the output variables that we want to predict: the number of confirmed cases and the number of deaths for future dates. These variables represent the ultimate goal and guide the model training process. The relationship between features and targets is what the system seeks to learn, so that when it receives input data, it can generate consistent forecasts.

2.3 Evaluation

The evaluation metric used in this competition is the Root Mean Squared Logarithmic Error (RMSLE). This measure calculates the squared difference between the logarithms of the predicted values (p_i) and the actual values (a_i), averaged over all observations and then square-rooted:

$$\text{RMSLE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\log(p_i+1) - \log(a_i+1)\right)^2}$$

The logarithmic transformation is a clever way to make sure that the metric focuses on relative error instead of absolute error. To put it simply, if there's an error of 10 cases when the actual count is 20, that's seen as a bigger deal than the same error when the actual count is 10,000. This characteristic makes RMSLE a great fit for modeling epidemics, especially since the number of cases can skyrocket over time.

This has a significant impact when we look at early-stage outbreaks. In these initial phases, the total number of cases is usually quite low, but any prediction errors are really important in relative terms. RMSLE makes sure to penalize these errors more heavily, which helps models stay attuned to early growth trends. On the flip side, when we reach later stages with a high number of cases, small absolute errors don't matter as much, and the metric adjusts accordingly to lessen their impact.

2.4 Rules and Constraints

- **Delivery format:** All predictions must be submitted in a CSV file that exactly follows the structure defined in `submission.csv`. This means that each record must contain the `ForecastId` field and the corresponding values for `ConfirmedCases` and `Fatalities`. Any deviation from the structure, such as additional columns, incorrect names, or changes in order, will result in automatic rejection of the submission.
- **Nature of the data:** Both confirmed cases and deaths are reported cumulatively, which means that their values can never decrease with respect to previous dates. This aspect must be respected when building models and generating predictions. In addition, some records may not include the `Province_State` column, as not all countries are subdivided into provinces; in such cases, the information is only identified at the national level. This requires the modeling system to be prepared to work with both detailed and aggregated data.
- **Time restrictions:** Predictions can only be made for dates explicitly included in the `test.csv` file. The training file (`train.csv`) and the test file (`test.csv`) have an initial overlap period that allows consistent comparisons and evaluation of the continuity of time series. Predictions for dates outside the defined range are not allowed, forcing models to focus exclusively on the provided time horizon.

- **Competition rules:** Participants are allowed to enrich their models with external datasets, provided that these are publicly available and accessible within the Kaggle platform. However, the final submission must strictly cover all combinations of `ForecastId` specified in the test file, ensuring that there are no omissions or duplicates.
- **Internal actors:** Competitors are responsible for designing, training, and adjusting prediction models, applying different statistical or machine learning methodologies. For its part, the Kaggle platform acts as a validator, verifying both the structure of the files submitted and the compliance with established restrictions. In addition, Kaggle manages the updating of the scoreboards and ensures that all participants are evaluated under the same conditions.

3 System Analysis

3.1 Elements

The system consists of several fundamental elements that work together to achieve the goal of forecasting the spread of COVID-19. These include the inputs that feed the system, the processes that transform the information, the outputs that reflect the results, and the actors that interact with the system.

- **Inputs:** The system receives data mainly from the **train.csv** and **test.csv** files, which contain historical information and the periods to be predicted. Additionally, external datasets can be used to supplement the information base.
- **Processes:** The operation of the system involves different stages such as data preprocessing, statistical analysis, time series modeling, and the use of machine learning algorithms to generate predictions.
- **Outputs:** The main result of the system is the prediction file (**submission.csv**), which must strictly follow the format required by the competition. This file contains the cumulative confirmed cases and expected deaths.
- **Actors:** Both the participants and the Kaggle platform are involved in the system. Participants design, train, and submit predictive models, while Kaggle validates the structure of the submissions, enforces the rules, and manages the leaderboards.

The inputs are critical because they determine the scope of the forecasting task. The training file (**train.csv**) provides historical records including `Province_State`, `Country_Region`, `Date`, confirmed cases, and deaths. The test file (**test.csv**) specifies the regions and dates for which predictions are required, ensuring consistency among all competitors. In addition, participants can incorporate external datasets to enrich their models, for example, with demographic, mobility, or health-related variables.

The processes represent the core of the system. They begin with data cleaning and preprocessing to handle missing or inconsistent records. Feature engineering techniques are then applied to extract relevant variables, followed by statistical or machine learning models that project future cases and deaths. These processes can be simple, such as exponential smoothing, or advanced, such as gradient boosting or recurrent neural networks. It is important that all predictions respect the cumulative nature of the data, meaning that values can never decrease over time.

Outputs are standardized to ensure comparability. Predictions must be submitted in a single file (**submission.csv**) with the exact structure required by the competition. This ensures that all ForecastIds defined in the test set are covered and that the reported values correspond to

confirmed cases and cumulative deaths. By requiring this strict format, the system allows for fair and automatic evaluation of all participants' models.

Finally, the actors are key to the functioning of the system. On the one hand, participants are responsible for developing accurate models using different statistical or computational approaches. On the other hand, Kaggle acts as a validator, ensuring compliance with the rules, verifying delivery formats, updating scoreboards, and guaranteeing the transparency of the competition. The interaction between these two types of actors ensures the reliability and fairness of the entire forecasting system.

3.2 Relationships and Data Flow

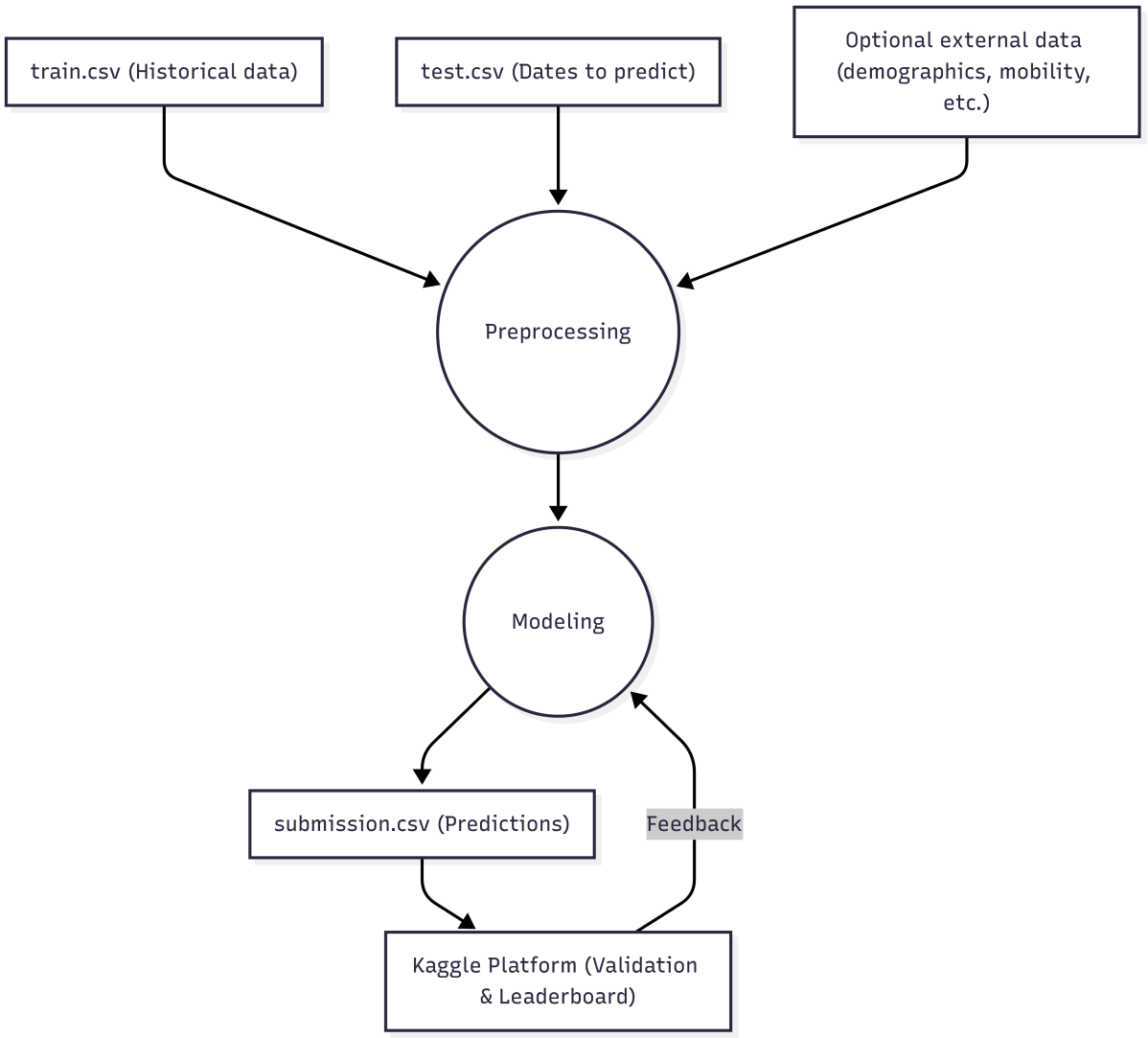

Figure 1: Data Flow

The system's data flow follows a well-defined sequence that allows raw information to be transformed into validated predictions within the competition. It all starts with the integration of the files provided: **train.csv**, which contains historical records of confirmed cases and deaths, and **test.csv**, which explicitly establishes the dates on which predictions must be generated.

Based on these inputs, the system goes through the preprocessing phase, where the data is cleaned, transformed, and organized. At this stage, problems such as missing values and differences in geographic granularity (countries with and without provincial subdivisions) are

resolved, and new variables are generated to facilitate model learning. The main objective of this phase is to ensure that all information meets the necessary conditions to be processed uniformly and efficiently.

Subsequently, the processed data feeds into the modeling stage, in which statistical or machine learning techniques are applied to estimate the evolution of confirmed cases and deaths. This process is not linear, as different approaches can be implemented and results compared, adjusting hyperparameters and methodologies based on the performance obtained.

The results are stored in a **submission.csv** file, which must strictly comply with the format defined by the competition. This file is sent to the Kaggle platform, where the structure is validated, the scores on the leaderboard are updated, and feedback is provided indicating the quality of the model against the actual data. This feedback becomes a key input for participants, as it allows them to identify possible improvements, refine preprocessing, or rethink modeling strategies, thus generating an iterative cycle of continuous improvement.

In this way, the flow not only describes a linear data transition, but also a dynamic feedback system, in which each iteration strengthens the model and contributes to more robust and reliable predictions.

3.3 System Boundaries

The prediction system defined by the competition has clear boundaries that allow us to distinguish which elements form part of its core and which act as external factors.

Within the limits of the system are the processes directly related to the manipulation and use of data: the incorporation of official files (**train.csv**, **test.csv**, and **submission.csv**), the preprocessing of information, the construction of statistical or machine learning models, and the generation of predictions in the format established by the competition. These components constitute the heart of the system, since without them it would not be possible to fulfill the main objective: to estimate the evolution of confirmed cases and deaths from COVID-19.

Outside the boundaries are external factors that, although they influence the development of the models, are not part of the system itself. These include the social, political, and economic dynamics that affect the spread of the pandemic, external databases that participants may decide to integrate, and the specific methodological decisions of each competitor. Likewise, the Kaggle platform infrastructure acts as a validation and feedback environment, but is not part of the internal prediction system, as its function is to ensure compliance with the rules and provide comparative results on the leaderboard.

In this way, the limits of the system are defined by the tasks strictly necessary to process official data and generate valid predictions, while the external context provides additional conditions and resources that may influence the quality of the model, but without forming a central part of the process.

3.4 Complexity & Sensitivity

Simplified Constraints

1. **Data Quality Issues:** Missing values, inconsistent reporting, and noisy daily updates.
2. **Inconsistent Granularity:** The inconsistencies with the granularity, because some countries reports are at the province/state level, while the others are only at the national level.
3. **Overfitting Risk:** The model may capture short-term noise instead of long-term trends leading into a possible overfitting behaviors.
4. **Temporal Dependencies:** The information constantly requires chronological validation.

5. **Unmodeled External Factors:** There are external factors like the government interventions, which are not included in the dataset but strongly affect the outcomes.

Based on the foregoing, the sensitivity of the system is high because all the variations related to the inputs end in significant changes in the outcomes like the ones related to the growth rates, which tend to lead towards inflated predictions associated with confirmed cases or fatalities. The provided dataset includes missing values related to the specifying of information related to the provinces (54% is defined as null). Furthermore, the data changes between certain adjacent dates are abrupt due to the delayed updates in the information provided by John Hopkins CSSE.

3.5 Chaos & Randomness

The prediction system is highly exposed to chaotic dynamics and random fluctuations from both the pandemic's nature and data structure. A clear example is when a country reports several days late and then suddenly adds several hundred or thousands of cases. This sudden update changes growth curves and causes dramatic variation in models. A database error (e.g. creating a negative or duplicating a value) can produce small perturbations which propagate over time and magnify differences in predictive ability.

Randomness can also be observed in contagion dynamics. The sudden emergence of a "super spreader", or an unexpected localized outbreak in a city could suddenly change the expected trend. None of these events can be predicted simply on historical data alone since they are influenced by context or external factors. In the same way, government actions (like lockdowns or removing restrictions) can cause the rate of spread to change suddenly leading to data jumps or changes in the modelling framework that may not account for this shift.

Social behavior also has another chaotic element. Small changes in mobility or compliance with biosecurity protocols can lead to sudden increases or decreases in cases, leading to yet another layer of uncertainty. Even local decisions, such as whether or not to reopen schools or hold mass events, although they may look small, can produce chains of contagion, observable in global data even weeks later.

Consequently, the second part of the COVID-19 modeling challenge is technical, but also systemic: predictions must occur in situations where chaos and randomness are not merely temporary deficiencies, but are structural features of the system.

4 Conclusion

The analysis of this competition led us to the comprehension of the intrinsic complexity related to the modeling of a system capable of forecasting specific scenarios related to the COVID-19 pandemic. By taking into account the model choice, we identified the points of variability of the system, like the ones related to the sensitivity of small input changes. Due to the high complexity related to the data and its high variability, the minor perturbations can cascade into large deviations in predictions. While using a predictive model can provide useful insights, its effectiveness depends on rigorous preprocessing, sensitivity analysis, and awareness of systemic constraints that reveal the chaotic tendency of the system.

5 References

- Kaggle — COVID-19 Global Forecasting (Week 1). `https://www.kaggle.com/competitions/covid19-global-forecasting-week-1`