Evaluation for Data Science in Techno-Socio-Economic Systems

February 25, 2021

Abstract

Here we describe the details of the evaluation in the lecture course "Data Science in Techno-Socio-Economic Systems", course number 851-0585-38L. In particular, we provide an overview of the potential projects for the data science analysis task, and the quiz.

1 Course evaluation summary

There will be two tasks for the course evaluation:

- Quiz on Moodle – 20% of the total mark. The quiz will be based on the lectures. The students will be able to retake the quiz twice without penalties.
- Data challenge – 80% of the total mark. The evaluation of this task will be described at the beginning of the following section.

2 Data challenge

Students can choose to work on one of the following data-science challenges: 1) the epidemic forecasting challenge, 2) the air quality forecast challenge, and 3) any data challenge present on Kaggle or KDD-cup.

Regardless of the challenge students will decide to work on, this part of the course evaluation will be based on the following objectives:

1. Abstract submission on time – 5 points. Students will be asked to submit an abstract which describes their chosen project and will prepare the assessors for their presentation. Abstracts should be submitted here.
2. Problem formulation – 5 points
3. Difficulty of the data challenge task – 10 points
4. Technical quality of the project – 20 points
5. Results 20 points
6. Critical thinking – 10 points
7. QA – 10 points

Objectives 2-7 will be assessed based on the presentation of the student team, that will be delivered during one of the last 4 lectures. The points obtained from the data challenge will be scaled and in the end will contribute to 80% of the final mark.

For the Epidemic datathon, objectives 2,3,4,5 are replaced by the evaluation as outlined in 2.1.4.

For the Air Quality forecast challenge, objectives 2,3,4,5 are replaced by the evaluation as outlined here.

2.1 Epidemic Datathon 2021

2.1.1 Location

This year, Epidemic Datathon will be hosted on eval.ai, as an in-class challenge (`https://eval.ai/web/challenges/challenge-page/759/overview`). The predecessor of this year's Epidemic Datathon can be found **here**.

Variable	Type	Meaning
$N_c(t)$	ground truth	total number of cases on day t and location c
$D_c(t)$	ground truth	total number of deaths on day t and location c
$\hat{N}_c(t)$	mandatory	forecast of total number of cases on day t and location c
$\hat{D}_c(t)$	mandatory	forecast of total number of deaths on day t and location c

Table 1: Overview of variables.

2.1.2 Data

We use case data from Johns Hopkins University (`github.com/CSSEGISandData/COVID-19/tree/master/csse_covid_19_data`), that includes the daily numbers of infections and the number of daily deaths in all available countries.The input data is organized in two separate CSV files: one for the daily number of cases (N_c), and one for the daily number of deaths (D_c) in a certain jurisdiction c. These CSV files use the following data format: [location region ... date1 date2 date3 ...]. E.g. an entry date1 gives D_c or N_c in a certain location on day date1 for jurisdiction c (see Tab. 1 for variable definitions).

2.1.3 Tasks

The results should be provided in a single CSV file, for which each row is organized as

$$[c, t_{\text{current}}, t_{\text{target}}, \hat{N}(t_{\text{target}}), \hat{N}_{\text{low}}, \hat{N}_{\text{high}}, \hat{D}(t_{\text{target}}), \hat{D}_{\text{low}}, \hat{D}_{\text{high}}], \tag{1}$$

where

- c is location (Province/State and Country/Region)
- t_{current} is the current date
- t_{target} is the date that the prediction is made for
- $\hat{y}(t_{\text{target}})$ is the predicted value of the variable y at time t_{target}
- N for cases, D for deaths
- $\hat{y}_{\text{low}}, \hat{y}_{\text{high}}$ are 95% confidence intervals for $\hat{y}$

The task is to make a prediction of the future number of daily cases and deaths in different locations. More specifically, t_{current} should be the following dates: $1.5.2020, 1.8.2020, 1.11.2020, 1.2.2021$ (in DD.MM.YYYY format). For every t_{current} you will be able to use $t_{\text{data}} = \{t_{\text{current}} - \Delta_{\text{training}}\}$, where $\Delta_{\text{training}} \in [0, 59]$, as training data.

For each t_{current} you need to make predictions from t_{current} to Δ_{predict} days in the future, so $t_{\text{target}} = t_{\text{current}} + \Delta_{\text{predict}}$, where $\Delta_{\text{predict}} \in [0, 29]$, for each country. Please note that all timestamps should be UTC.

Scripts for baseline epidemic model can be found in `github.com/ninoaf/baseline_epi_predict`.

2.1.4 Evaluation

To appropriately evaluate and compare different submissions, we begin with an overview of desired properties (1-9) of our evaluation metric:

1. **Interpretability**: We prefer "mathematically simple" measures as we want that the best models/submissions eventually become useful for policy makers.
2. **Mathematically well-defined**: Our evaluation metric should be connected to existing evaluation metrics.
3. **"Unbiased"**: Our evaluation measure shall not systematically prefer methods whose forecasts are too low or applied only to "predictable" cases.

4. **Comparability across time**: The actual case numbers increase according to epidemic spreading dynamics, so we have to be able to compare submissions for different epidemic growth regimes (i.e., different total case numbers and rates of change).
5. **Comparability across countries**: The outbreak dynamics will be different in every country/region, so we have to be able to compare different local epidemic growth regimes.
6. **Uncertainty evaluation**: If participants decide to include confidence intervals in their submissions, our evaluation measure should take this information into account.
7. **Comparable and additive with optional predictions**: We want to encourage participants to prepare submissions for many different countries/regions and optional variables.
8. **Well-defined if no prediction was submitted**

Clearly, some of the properties are in contradiction and we will have to make compromises.

Global leader board We use the Absolute Logarithm Error (AbsLogE) over any set of points $\{\hat{y}(1), \ldots, \hat{y}(n)\}$ with associated ground truth $\{y(1), \ldots, y(n)\}$:

$$\text{AbsLogE}(\hat{y}(1), \ldots, \hat{y}(n), y(1), \ldots, y(n)) = \sum_{i=1}^{n} |\log(\hat{y}(i)+1) - \log(y(i)+1)|. \tag{2}$$

For each country or location c and evaluation day d, we define the global score as

$$\text{SCORE} = \sum_{i=1}^{\Delta} \sum_{d} \sum_{c=1}^{C} \frac{1}{|\log(\hat{N}_c(i)+1) - \log(N_c(i)+1)| + |\log(\hat{D}_c(i)+1) - \log(D_c(i)+1)| + 1}. \tag{3}$$

Here d is each of the days that students have to make predictions for. If the prediction matches the actual ground truth (i.e., if $\hat{N}(i) = N(i)$ and $\hat{D}(i) = D(i)$), the corresponding contribution to SCORE is 1. Correctly guessing all outcomes in all countries during one-week yields the upper bound $7C$ of SCORE. If no submissions are provided for certain days or countries, we set the corresponding contribution to zero (the equivalent of having an infinite error). This measure satisfies properties 2,4,5,7,8. It has a small bias towards having more global (all countries) solutions, it does not take uncertainty into the account. In order to facilitate readability, we normalize SCORE such that a flawless submission corresponds to a SCORE of 100.

In addition to SCORE the leader board will provide the AbsLogE computed solely over the predictions of N or D respectively. This metric can be used as additional feedback on model performance.

Uncertainty evaluation For the optional confidence intervals $\{y_{\text{low}}(i) < \hat{y}(i) < y_{\text{high}}(i)\}_{i=1}^{n}$ and corresponding ground truth variables $\{y(1), ...y(1)\}$, we define the mean coverage error (CE) as

$$\text{CE} = \frac{1}{n} \sum_{i=1}^{n} \frac{y_{\text{high}}(i) - y_{\text{low}}(i)}{\mathbf{1}[y_{\text{low}}(i) \leq y(i) \leq y_{\text{high}}(i)] + \epsilon}, \tag{4}$$

where n is the number of datapoints in the testing set, and $\mathbf{1}[y_{\text{low}}(i) \leq y(i) \leq y_{\text{high}}(i)]$ is the binary counting function which is equal to 1 if the ground truth lies within the confidence interval, and 0 otherwise. For simplicity, we again set $\epsilon = 1$. Note that the binary counting function measures how good your confidence intervals are. In addition, it takes the confidence-interval width $y_{\text{high}}(i) - y_{\text{low}}(i)$ into account. If forecasts intervals are too "wide", the difference $y_{\text{high}}(i) - y_{\text{low}}(i)$ will increase; if the intervals are too "narrow", coverage will be low and the denominator of CE approaches ϵ. The interval bounds y_{high} and $y_{\text{low}}(i)$ have to satisfy the condition: $y_{\text{high}} - y_{\text{low}}(i) \geq 2$.

In the case where students choose not to calculate the confidence intervals, the corresponding columns $[\hat{y}_{\text{low}}, \hat{y}_{\text{high}}]$ should be left out of the submission CSV file.

2.2 Air Quality Forecast Challenge 2021

Air Quality Forecast Challenge 2021 is outlined here and supervised by Dr. S. Mahajan.

2.3 Alternative projects

Analysis of any dataset on Kaggle or KDD-cup. In theory you could use any dataset that is available, however, ensure that the dataset you choose is well-documented. It should also have an outlined task, or you should be able to create your own task.