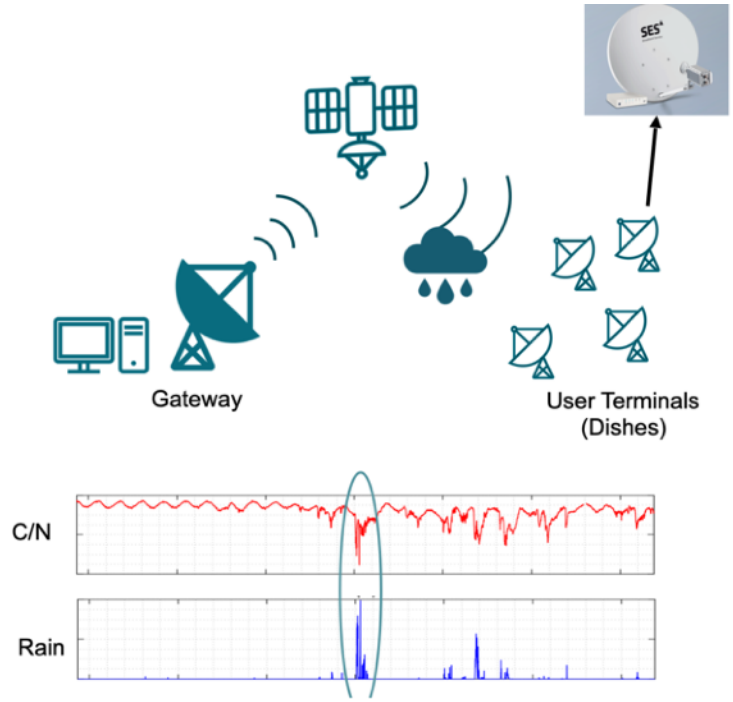
Databourg DS Interview Task

Real-Time Rain Monitoring using Communication Satellites

In communication satellite systems, satellite operators monitor the connection to each of their satellite dishes (internet users everywhere on Earth) in real-time. This parameter is the carrier-to-noise (C/N) ratio which can be understood as the signal-to-noise ratio and is measured in dB. In case of rain above the terminal, the signal drops and there is a clear correlation between the amount of signal drops and the rain intensity. As the signal is almost only impacted by rain it can be used as a virtual rain sensor. More precisely, it can be seen as a point measurement of rain at the location of the satellite dish. Combining many of these rain point measurement allows for creating rain maps for a full region with e.g. 1km x 1km spatial resolution.



Satellite communication network and the impact of rain.

Tasks

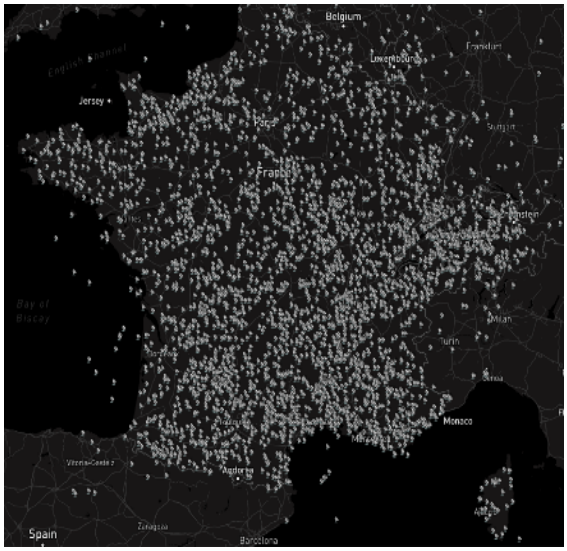
Attached you can find 2 C/N data samples from one satellite dish. For several months and for every 5 minutes you can find the time stamp, the C/N value (“FWD C/N”) and the actual rain intensity in mm/h (“rain_intensity_rg”) as measured by a rain gauge located in close proximity to the satellite dish. If the rain is very strong, the satellite dish sometimes experiences an outage in which case there is no C/N reported.

timestamp_utc	FWD (C/N)	rain_intensity_rg
2020-11-01 00:00:00+00:00	5.9	0.0
2020-11-01 00:05:00+00:00	5.9	0.0
2020-11-01 00:10:00+00:00	5.9	0.0
2020-11-01 00:15:00+00:00	5.8	0.0

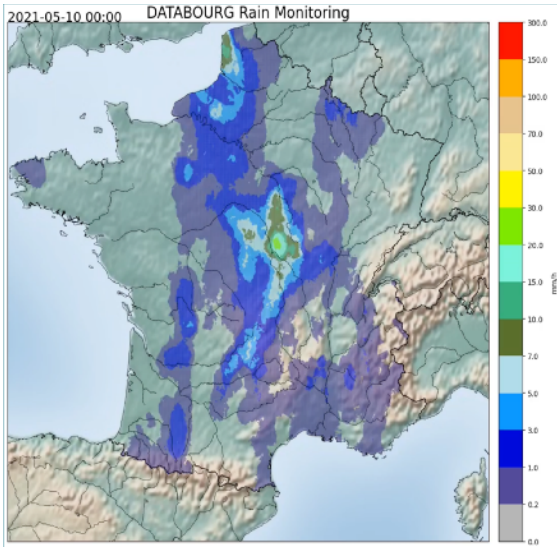
Data sample including time stamp, C/N data and rain intensity from near-by rain gauge.

The task is to develop an algorithm that extracts rain intensity information from the C/N signal. In order to estimate the rain at any time stamp, the 2 hour history of C/N values can be used. But as it is a real-time product no future C/N values can be used.

1) Which machine learning algorithms would you consider for this task. Mention at least 3 and discuss pros and cons.
2) Do you think it makes sense to split the task into 2 subtasks: first, classify a timestamp into rain/dry event and second, estimate the rain intensity for only the rainy time stamps?
3) Before any experiments, which steps of data preprocessing and cleaning are necessary? Which ways for handling missing data do you propose given this kind of data?
4) Develop a supervised deep learning algorithm for the problem using the first dataset ("data1.csv") as training and the second as test data set ("data1.csv"). Keep in mind, the algorithm needs to run efficiently thousands of times for every satellite dish every five minutes. Which metrics do you propose for measuring the quality of the rain estimation?
5) Given a large number of satellite dishes (several 1000s), which algorithms are your first choice to create interpolated rain maps such as the one below?



Distribution of Satellite Terminals



Interpolated Rain Map

6) Imagine a satellite operator pushes a csv file with new C/N data on your Linux data server every five minutes. How would you construct an efficient real-time data pipeline that includes above mentioned algorithm 4), an interpolation strategy 5) and provides a customer API for accessing rain map results? Which components are necessary? Which tools would you use?
7) Going beyond a single terminal algorithm for rain extraction, there is also the possibility to use all terminals or neighbouring terminals in order to improve the estimated rain accuracy. How would you construct such a multi-terminal algorithm? What are advantages and disadvantages compared to the single terminal algorithm?

Hints:
Except task 4), the tasks do not require coding and it is sufficient to write down your thoughts and proposals. In case you are short of time, it is also acceptable to provide a detailed plan for how you would solve task 4) instead of code.