14/01/2022

IA Framework Project
Défi IA MeteoFrance
Lynda BEN BOUDAOUD
MS VALDOM

Project Report

Objective:
Predict the accumulated daily rainfall (on 24h) on ground stations which are distributed on the North-West quarter of France. These stations are equipped with different Meteo instruments to measure temperature, wind, rain, etc.

Data sources:
We have two types of data

- Measurements stations (X_station): measurements at date t (rainfall, wind, ...etc)
- Weather forecasts: made by weather forecast systems from METEO FRANCE to predict in the future the weather

Approaches:
Two approaches are suggested to use by MeteoFrance:

1) Learn from past errors
The Dataset will contain forecast data (METEO France) and measurements data to predict errors of MeteoFrance forecast model (overestimate the rainfall or underestimate)

2) Predict the future from the past measurements (time series prediction)
For a give station, from the measurements of the day D-1, predict the rainfall on the day D. Existing methods LSTM

The chosen Approach:
We choose to work on the first approach which consist of the Prediction of the error made by Meteo France model on cumulative rainfall prediction at station s on day D using station measurements of day D-1

Working environment:
We chosoe to use use in our working environment: python, pandas, numpy, tensorflow, docker, netcdf4

Preprocessing:
Brut data provided by Meteo France required many pre-processing steps before feeding into the model. This steps are described in the following:

1. Extract forecasts from netCDF files: Forecast data are stored in netCDF files. We have chosen to use Arome 2D and then we developed a python to extract total precipitation for

each point(latitude, longitude) for each day starting from 02/01/2016 to 31/12/2017, each day is saved in a separate CSV file (script:extract_forecasts.py)

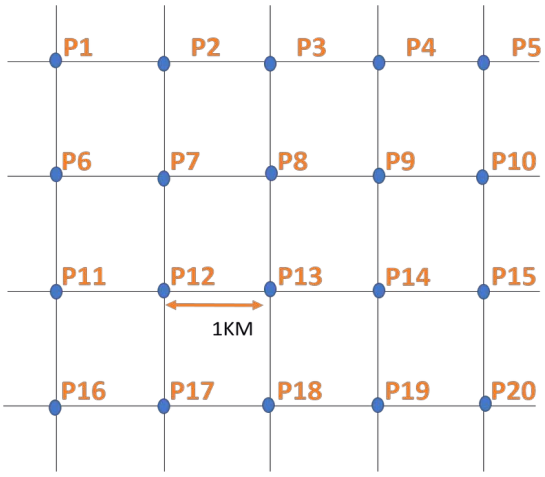
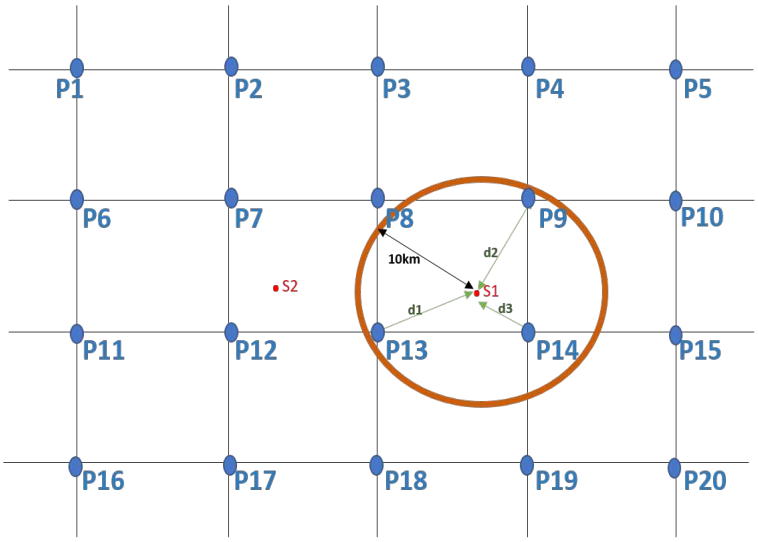


Figure 1 : AROME 2D grid.

2. Compute forecasts for stations (Y_forcast) according to the weighted average by distance and MeteoFrance previsions for each station, within a radius of 10 km as it depicted in figure 2.



Problem: Find the meteo France forecast on the stations

$$ps_1 = \frac{p_{13} * d_1 + p_9 * d_2 + p_{14} * d_3}{d_1 + d_2 + d_3}$$

Figure 2 : stations forecast.

3. Handle vertical and horizontal NANs

The CSV file X_station_train contains the collected measurements for each station, each day, for each hour.

Encountered problems:

- There are only data for some hours, rather than for 24h. In fact, our model takes as input D-1/24h.

  Solution:

  Clean the dataset by deleting data for stations in a day that we don't have the 24h.

- There are missing data in some rows (some measures missing)

  Solution:

  We proceed to replace missing data by the mean of the column

4. Prepare the dataset for training: In this step we delete Id , number_station and date columns and create months as a new column, while transforming X_train dimension to a 3D dimension, where each 2D slice correspond to a 24 hours and measurements varying them according to a third dimension which is stations/days. The following figure depict the space transformation.

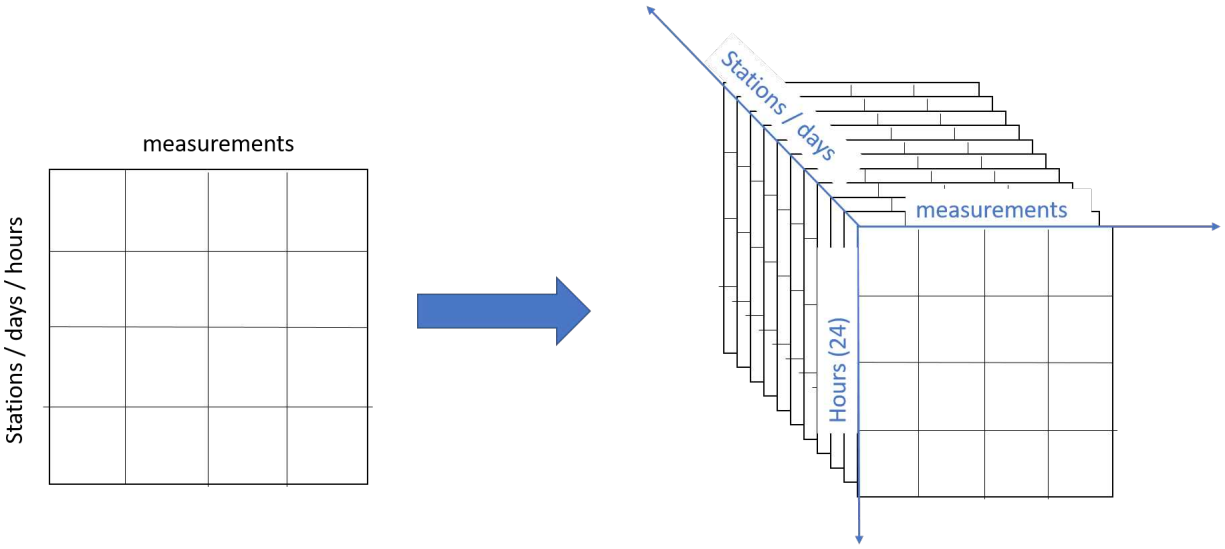


Figure 3: data transformation

In order to complete the data preparation we compute Y_train which is the error of MeteoFrance model according this formula below:

Y_train = Y_forecast_stations – Y_ground_truth (of the accumulated rainfall)

Then For each day, each station, we have to find the corresponding error in Y_train, for this, we sort Y_train according to axe Stations/days, in fact, **Y_train (D+1) on station s corresponds to X_train (D) of station s.**

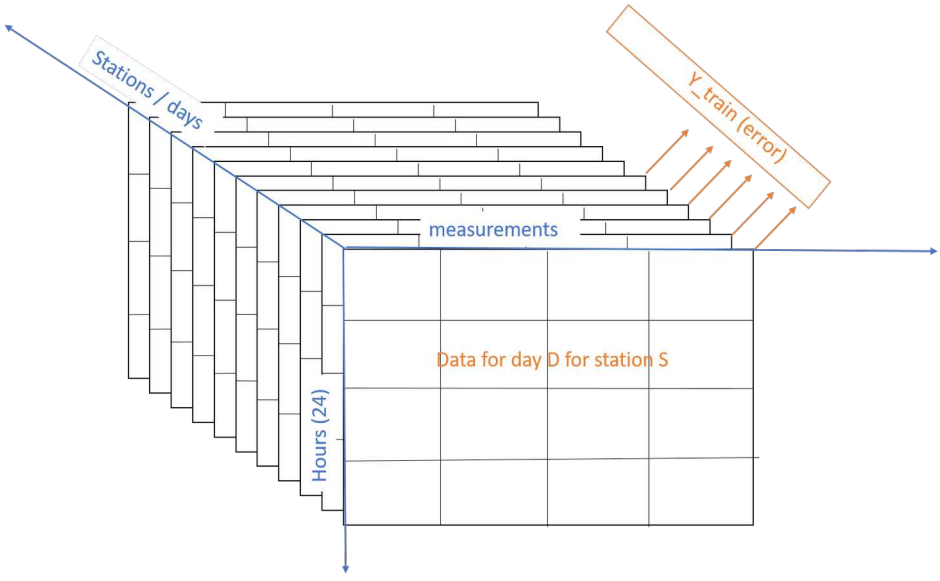


Figure 4 : The final dataset format.

The model:
The data was prepared. Now, it is time to feed it to a model and train it in order to predict the errors.

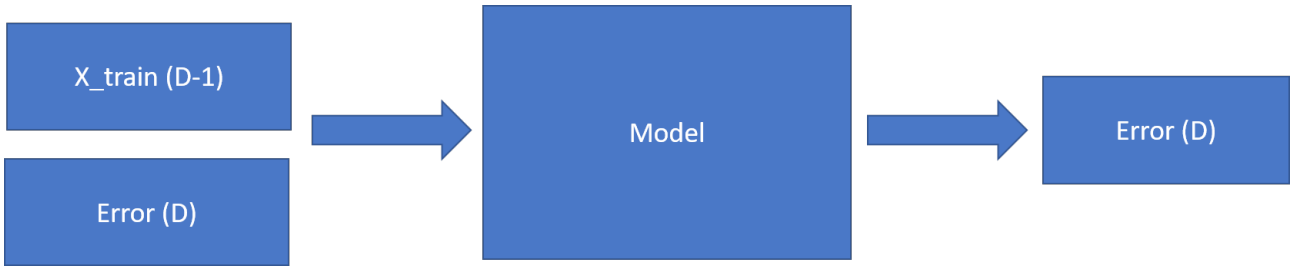


Figure 5: model input/output

In this project, we are confronting to a time series problem. A time series is a sequence of observations taken sequentially in time. Thereafter, we chose to use a LSTM deep learning based model.

An LSTM (Long short Term Memory) is a variant of artificial recurrent neural networks (RNN) architectures, capable of learning long term dependencies. A common LSTM unit is composed of a **cell**, an **input gate**, an **output gate** and a **forget gate**. The cell remembers values over arbitrary time intervals and the three *gates* regulate the flow of information into and out of the cell.

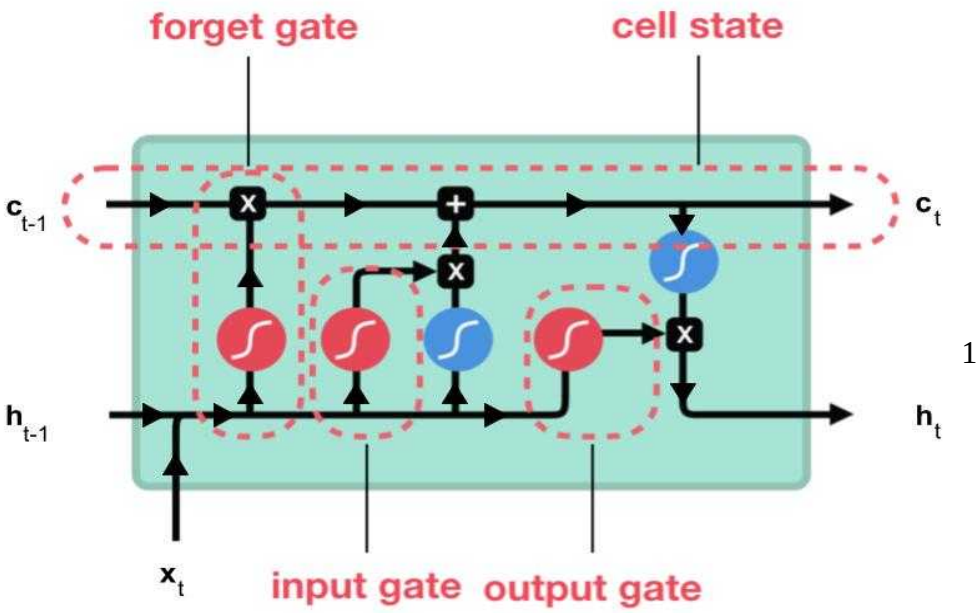


Figure 6: A typical architecture of an LSTM cell.

Architecture of the used model:

We used an LSTM model composed of 2 layers: one LSTM layer, and one dense layer, as it is depicted in the following figures. This model is suggested by keras in their documentation and already experimented for weather prediction.

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_7 (InputLayer)         [(None, 24, 7)]           0
_________________________________________________________________
lstm_6 (LSTM)                (None, 24)                3072
_________________________________________________________________
dense_6 (Dense)              (None, 1)                 25
=================================================================
Total params: 3,097
Trainable params: 3,097
Non-trainable params: 0
_________________________________________________________________
```

Figure 7: LSTM based model parameters.

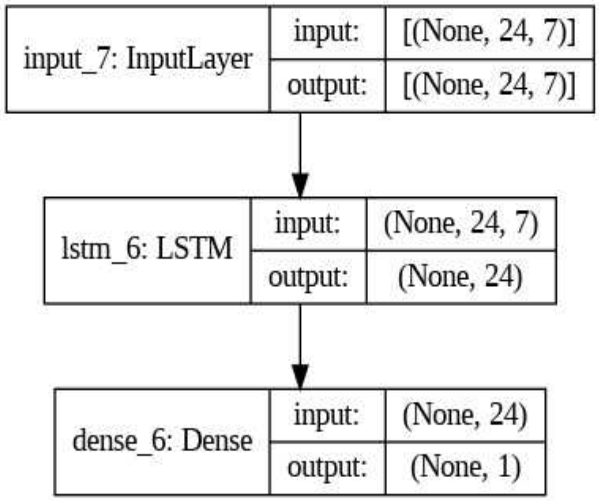


Figure 8: LSTM based model architecture.

Results:

The figure below shows the result of the loss (training vs validation).

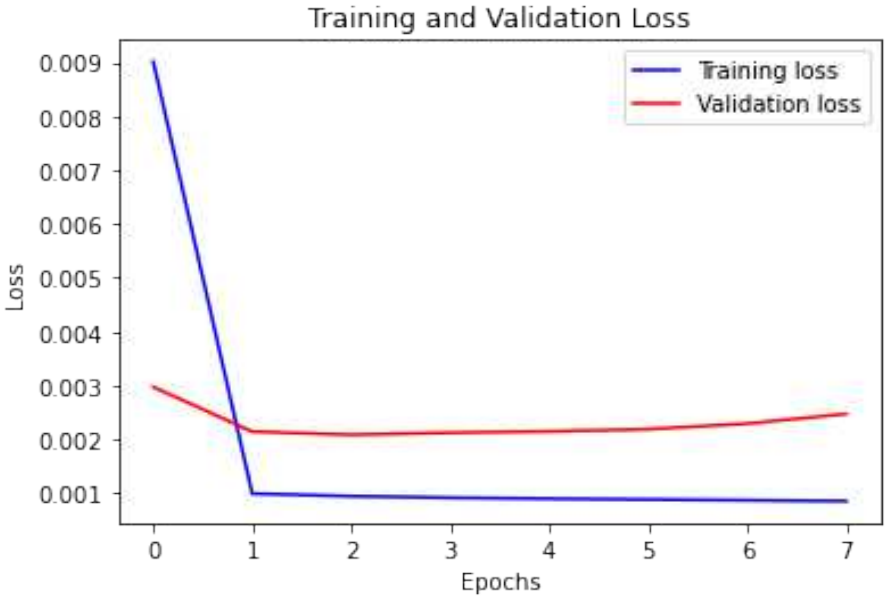


Figure 9: Training and validation loss.

Conclusions:

This is one of the way to forecast which is not the best one, in fact, model, and parameters can be improved and optimized to get better results. Moreover, we can also improve the methods used in the pre-processing part to compute missing data.

Preprocessing data in this projet was the part that has taken the more time, and the more difficult than the LSTM modeling part.