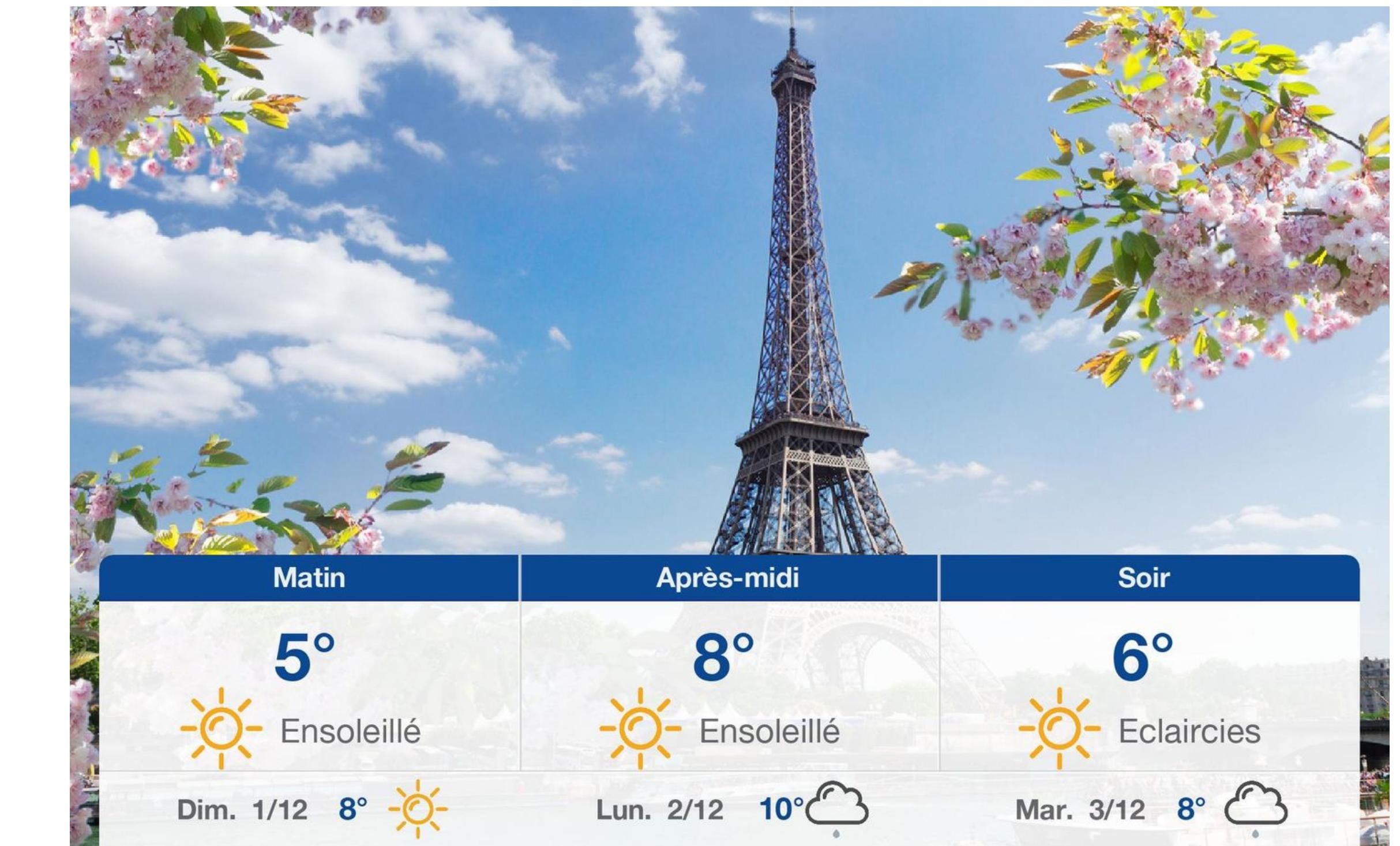
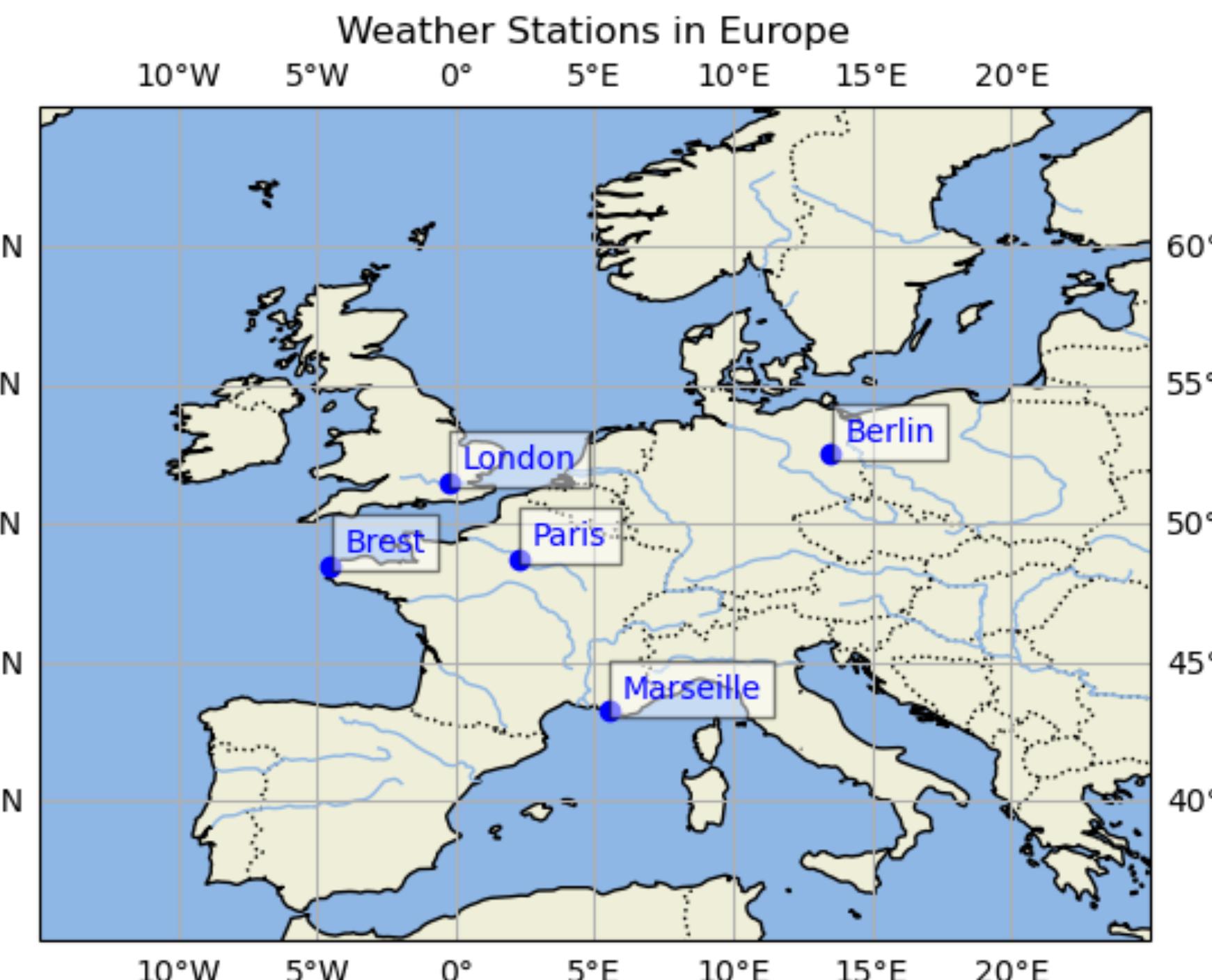


Weather station 2024-2025



Mentor: Julio Cardenas

Students: Dimitri Iratchet - Fabien Lagnieu - Tristan Waddington

I. Project presentation

1.1 General presentation

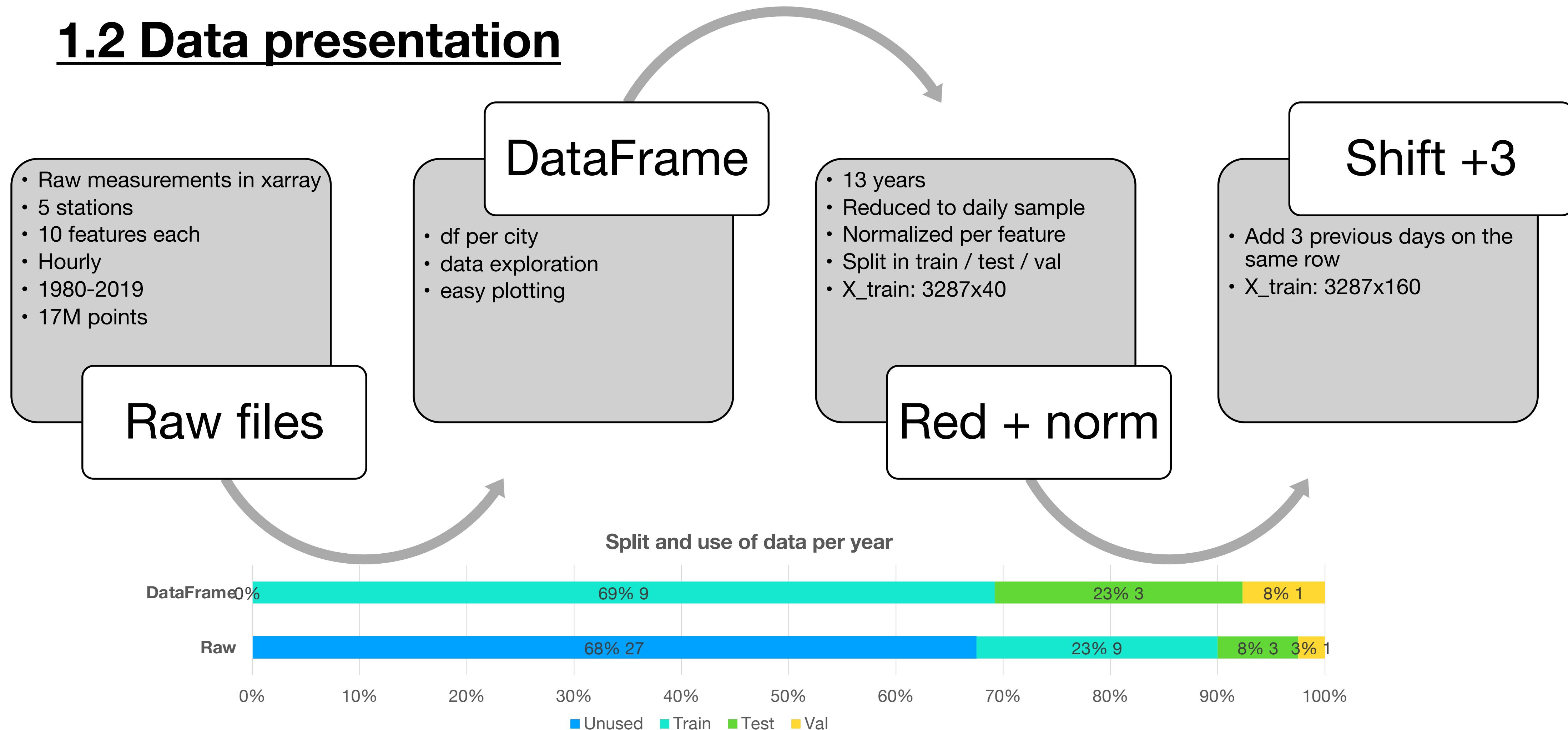
Aim of the project :

Paris weather station was cyber-hacked !
It is therefore unable to share its weather measures since the beginning of 2019.

Our task is to predict this missing data from other European weather stations measures.

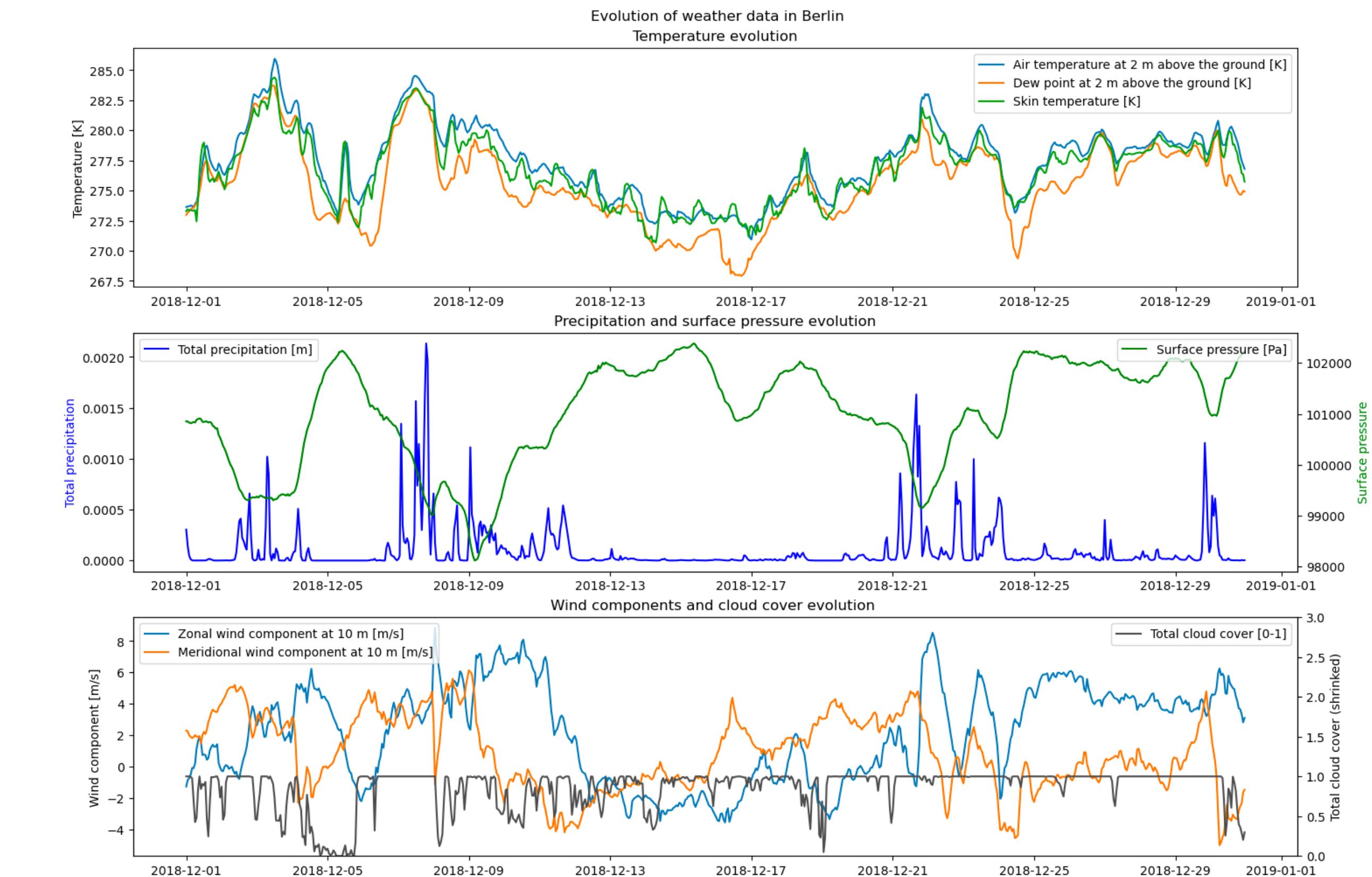
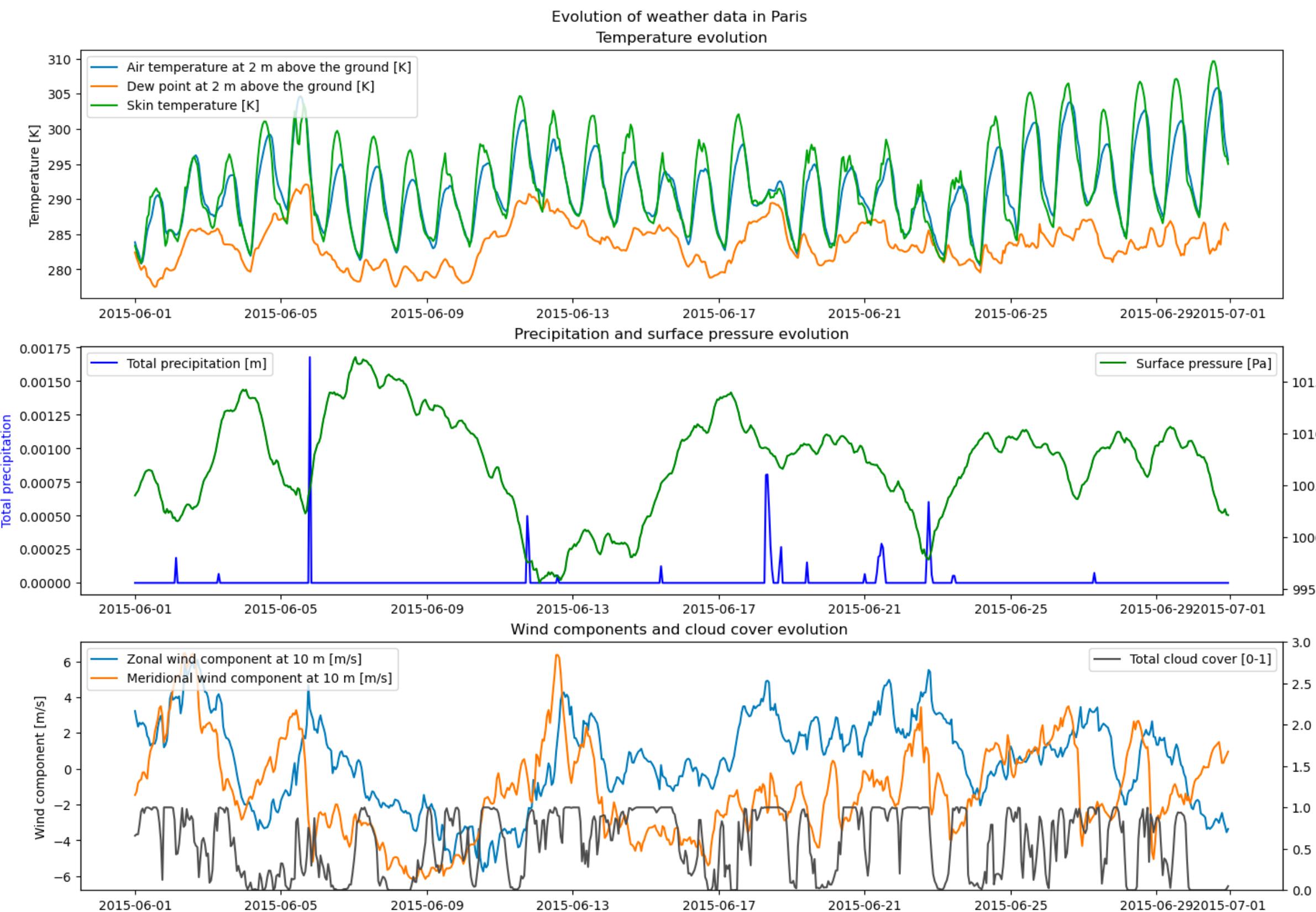
I. Project presentation

1.2 Data presentation



II. Data Loading / preparation / study

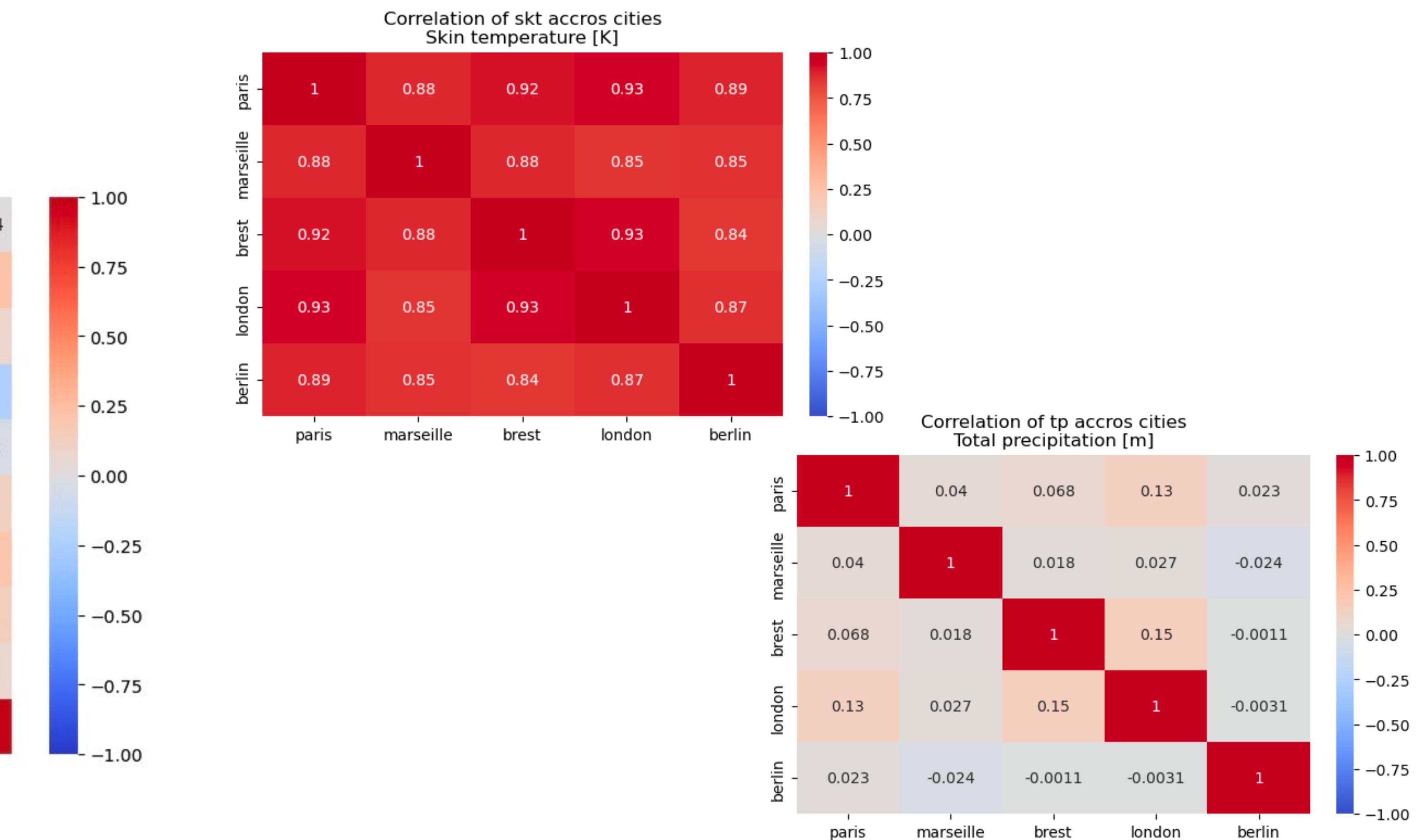
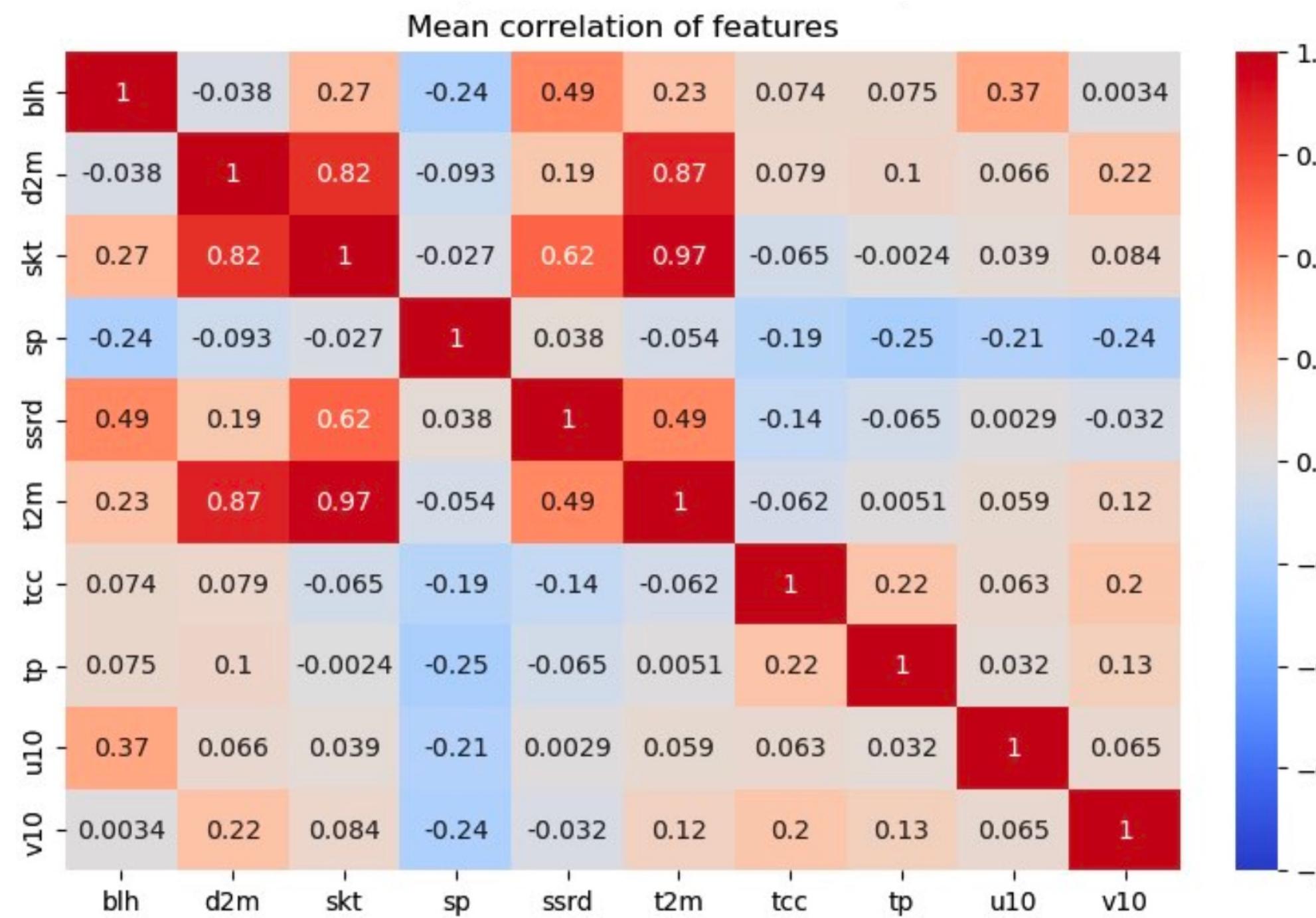
2.1 Data exploration



II. Data Loading / preparation / study

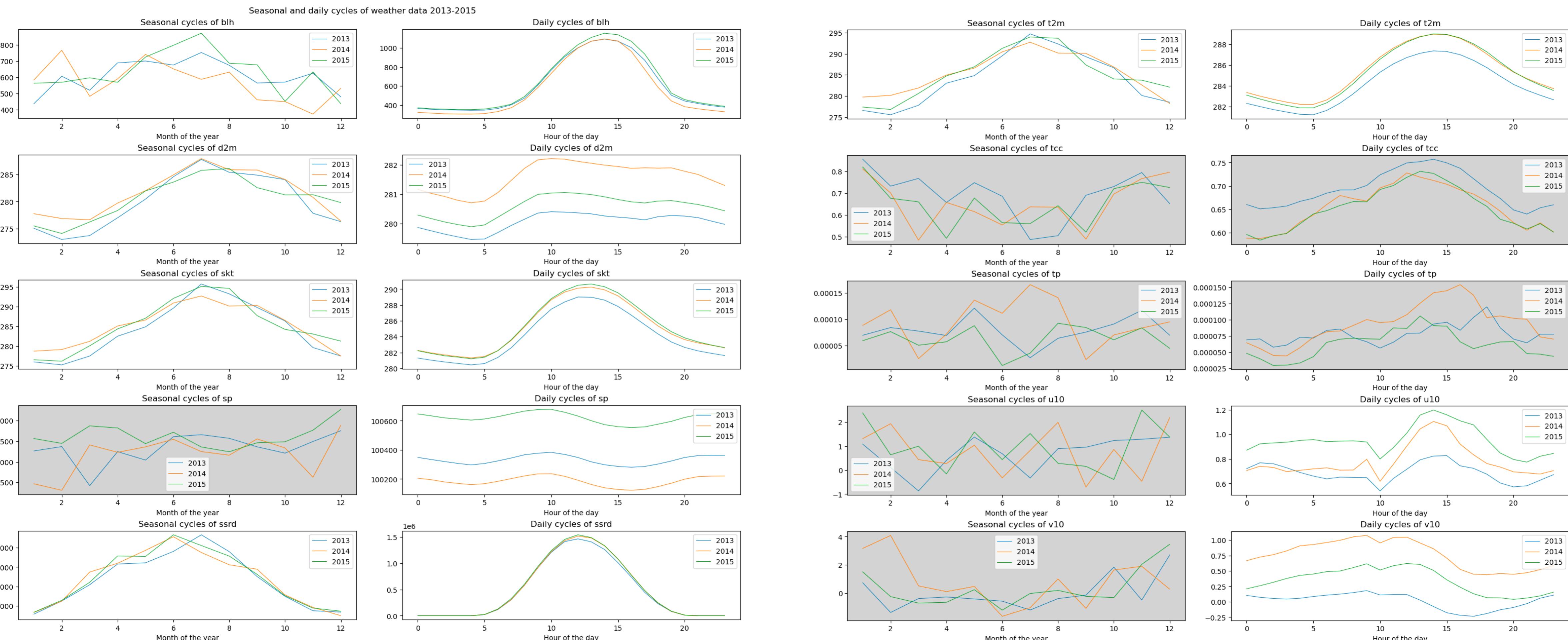
2.2 Covariance study

→ Between Features



II. Data Loading / preparation / study

2.3 Cycles in features



III. Prediction of Paris Weather

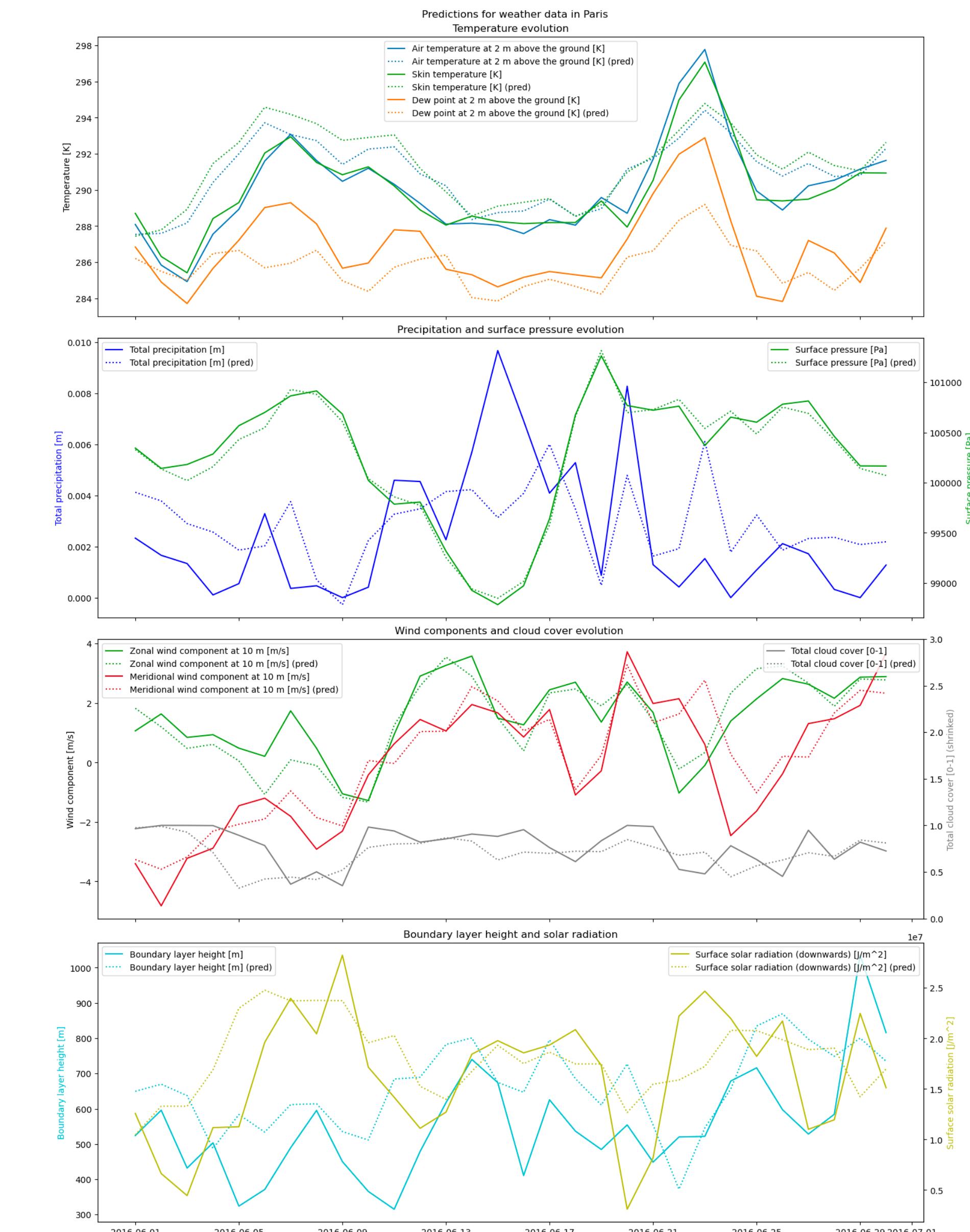
3.0 Baseline : Linear regressor

Do we have different result when focusing only on one feature or are we more accurate when using all available features of the dataset?

In the following colored-gradients, we are expecting to have the most dark green / low value cells.

MSE per variable for the baseline model per variables: LinearRegression()										
	paris_blh	paris_d2m	paris_skt	paris_sp	paris_ssrd	paris_t2m	paris_tcc	paris_tp	paris_u10	paris_v10
2016	0.166761	0.062701	0.039654	0.009005	0.085802	0.046109	0.248094	3.223900	0.046913	0.140415
2017	0.193160	0.051314	0.033328	0.009482	0.071674	0.043901	0.210177	3.543318	0.045133	0.129609
2018	0.178478	0.069938	0.044037	0.020020	0.074057	0.048910	0.232721	3.370375	0.054735	0.157305
all	0.179466	0.061318	0.039006	0.012836	0.077178	0.046307	0.230331	3.379198	0.048927	0.142443

MSE per variable for the baseline model global: LinearRegression()										
	paris_blh	paris_d2m	paris_skt	paris_sp	paris_ssrd	paris_t2m	paris_tcc	paris_tp	paris_u10	paris_v10
2016	0.142424	0.045776	0.036308	0.001665	0.069907	0.040994	0.206956	2.554993	0.017333	0.042979
2017	0.142414	0.041109	0.024748	0.001933	0.059055	0.030950	0.177964	2.966685	0.017705	0.043503
2018	0.138695	0.059709	0.035873	0.002542	0.061844	0.044294	0.204308	2.782853	0.063525	0.133017
all	0.141178	0.048865	0.032310	0.002046	0.063602	0.038746	0.196409	2.768177	0.032854	0.073167



III. Prediction of Paris Weather

3.1 Testing models

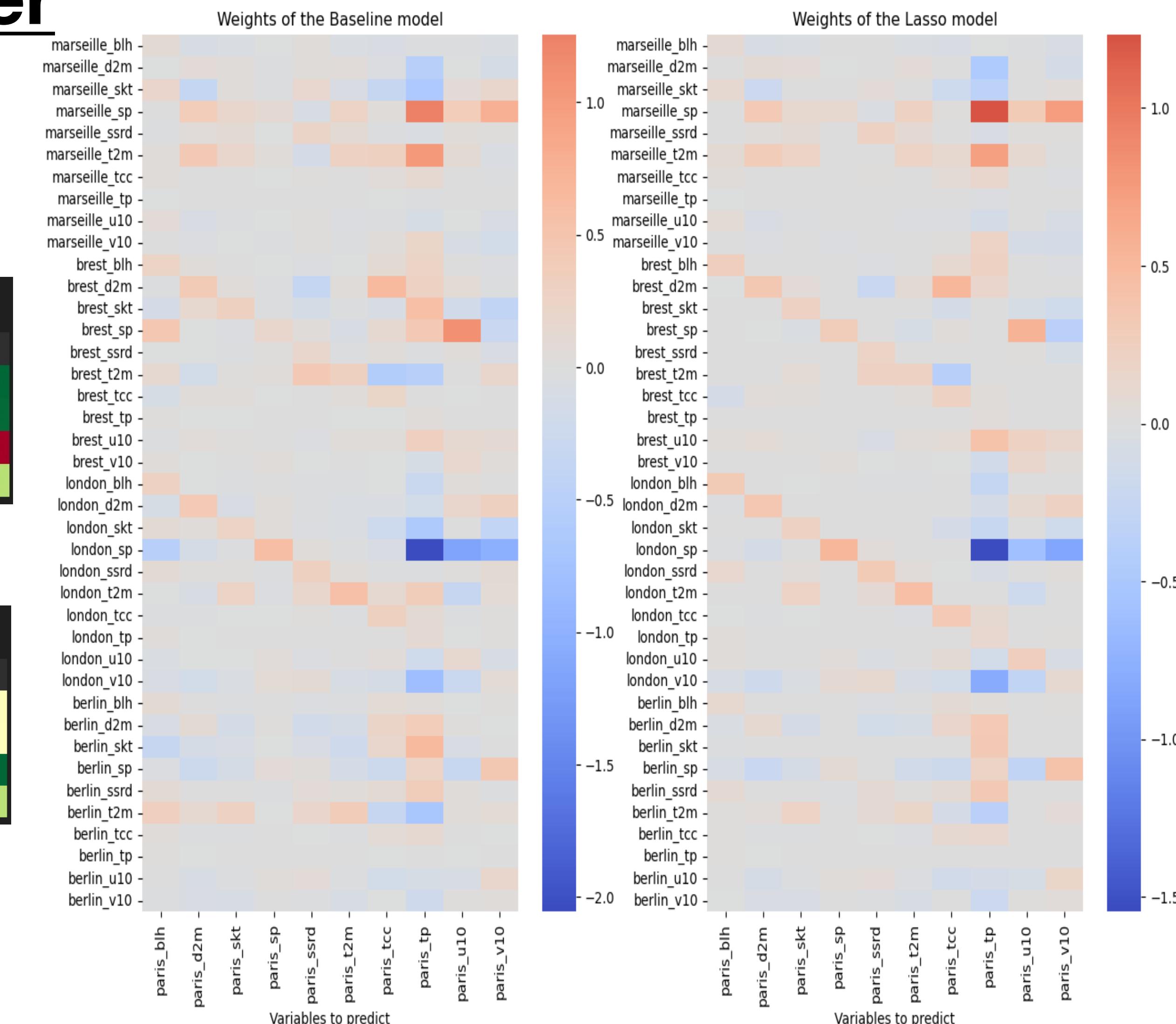
- **Linear Lasso**

Mean Squared Error (MSE) for Lasso Regression:

	paris_bh	paris_d2m	paris_skt	paris_sp	paris_ssr	paris_t2m	paris_tcc	paris_tp	paris_u10	paris_v10
2016	0.143073	0.044610	0.036376	0.001690	0.071310	0.041248	0.209331	2.565525	0.017508	0.043377
2017	0.141890	0.040176	0.025366	0.002042	0.059067	0.031453	0.178193	2.963445	0.018364	0.043695
2018	0.135104	0.060135	0.033120	0.003115	0.061730	0.045443	0.201271	2.791260	0.046535	0.114168
all	0.140022	0.048307	0.031621	0.002282	0.064036	0.039381	0.196265	2.773410	0.027469	0.067080

Baseline vs Lasso model, reduction of MSE per variable

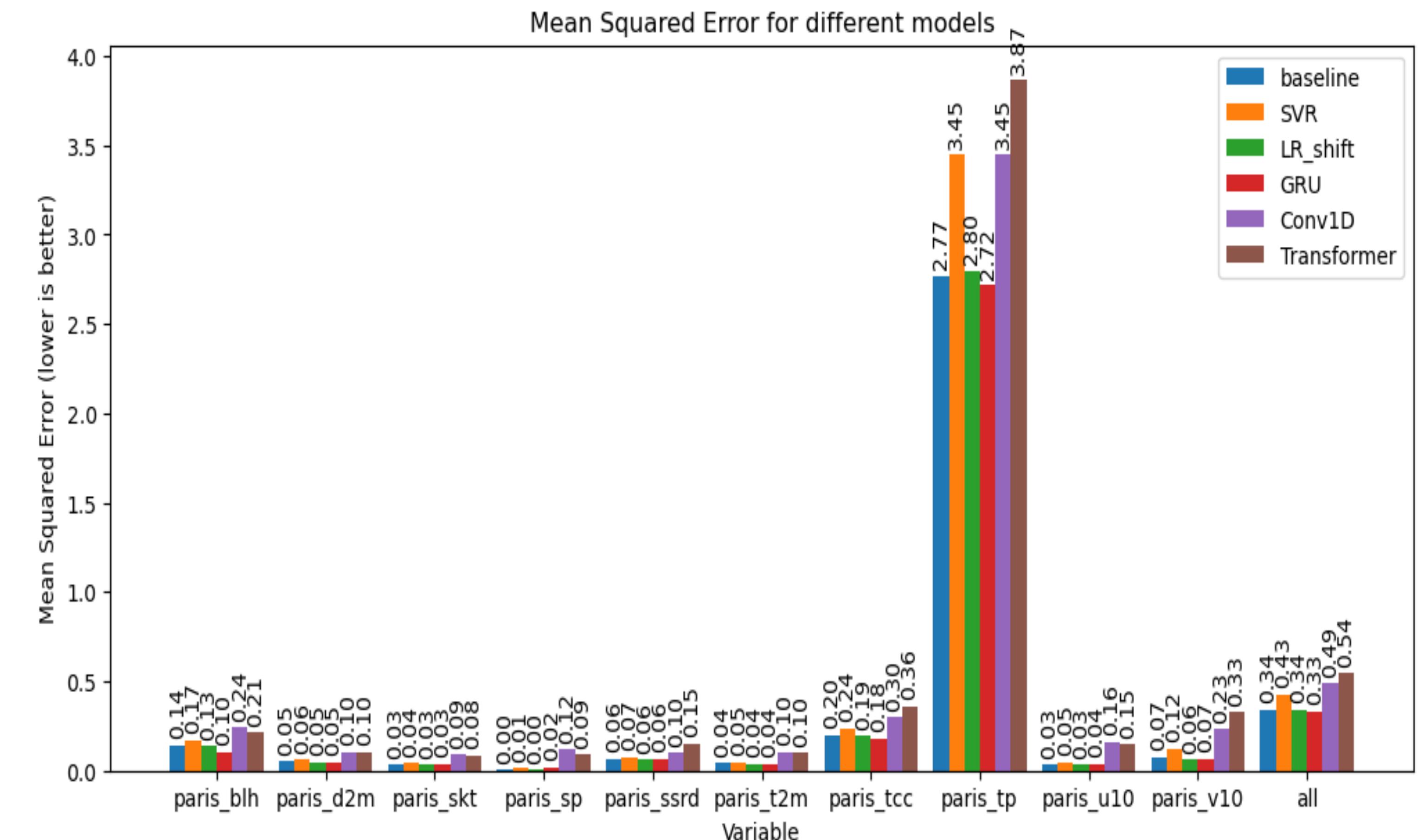
	paris_bh	paris_d2m	paris_skt	paris_sp	paris_ssr	paris_t2m	paris_tcc	paris_tp	paris_u10	paris_v10
2016	-0.000650	0.001166	-0.000068	-0.000025	-0.001402	-0.000254	-0.002376	-0.010532	-0.000175	-0.000398
2017	0.000524	0.000933	-0.000618	-0.000110	-0.000012	-0.000502	-0.000230	0.003240	-0.000658	-0.000191
2018	0.003591	-0.000425	0.002753	-0.000573	0.000114	-0.001149	0.003036	-0.008407	0.016990	0.018849
all	0.001155	0.000558	0.000689	-0.000236	-0.000434	-0.000635	0.000144	-0.005233	0.005385	0.006087



III. Prediction of Paris Weather

3.1 Testing models

- **Tree-based models :**
 - **RandomForestRegressor**
 - **XGBoost**
- **SVR**
- **Deep learning :**
 - **GRU**
 - **CONV1D**
 - **Transformer**



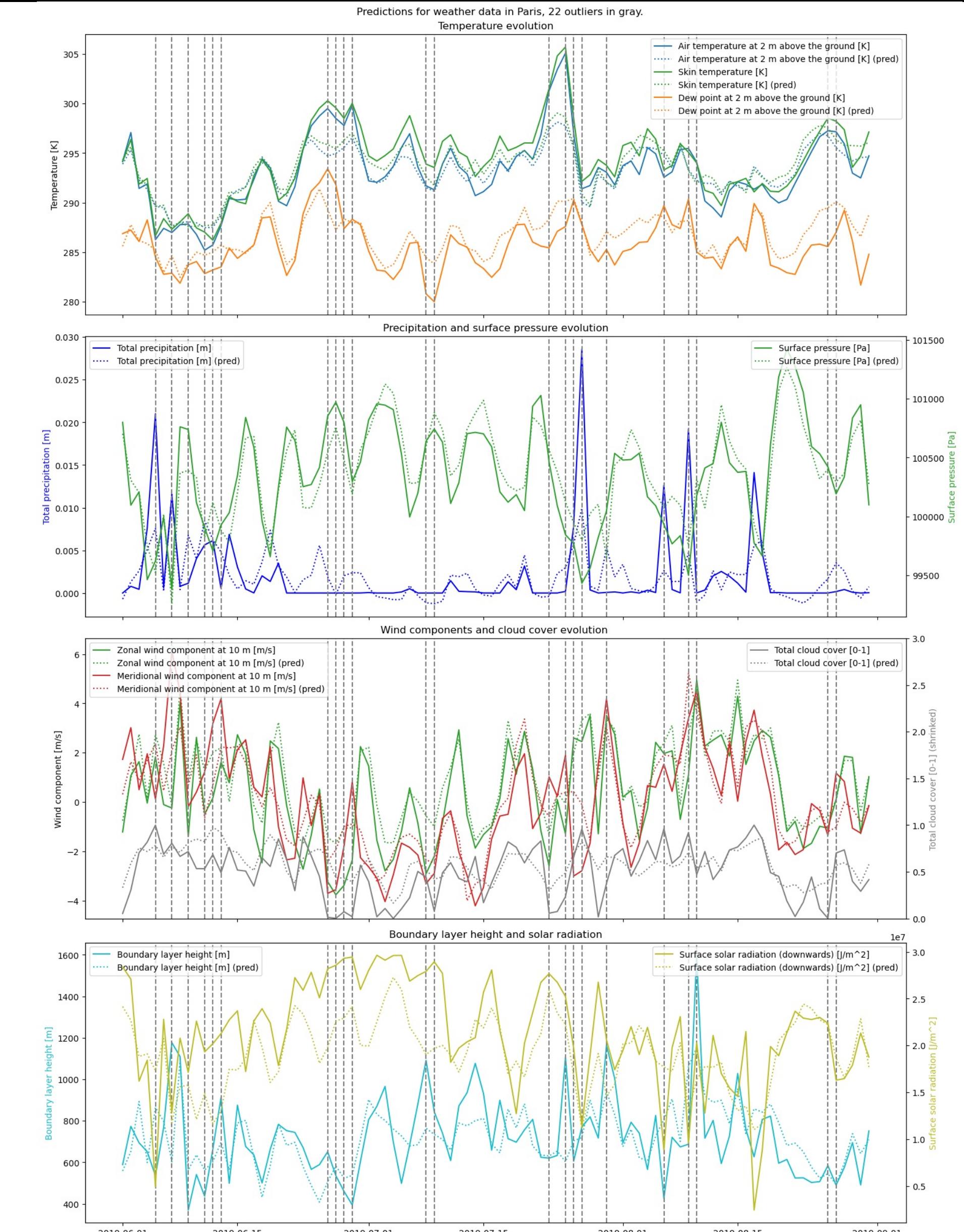
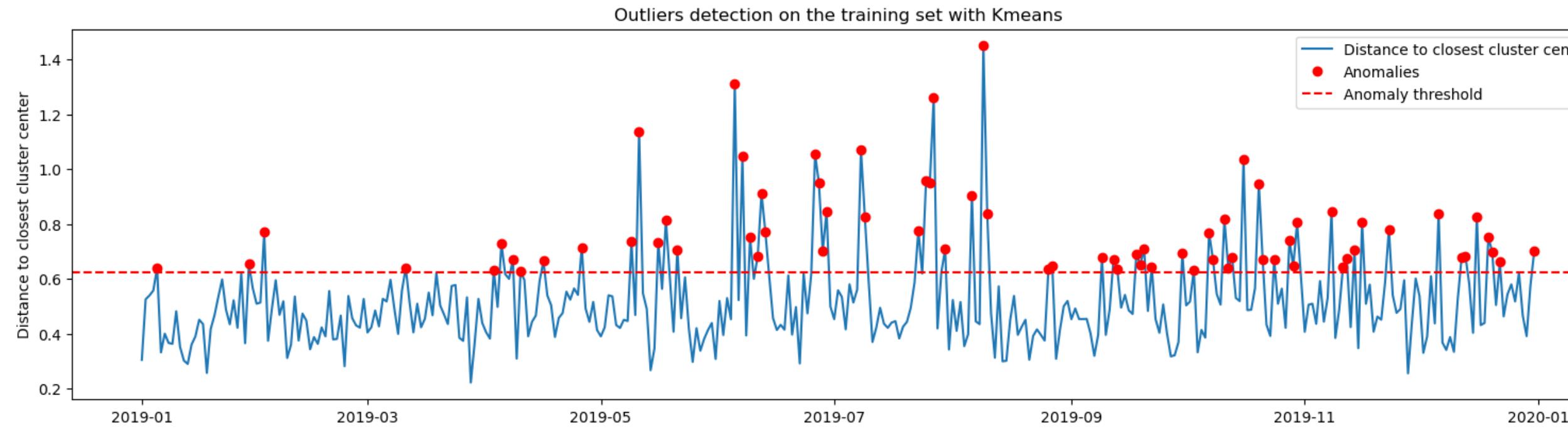
IV. Anomaly Detection

Asumption 2016-2018: 1 cyber-incident / year with Paris reports

Detect anomalies between predictions and reports:

- Best predictor= **average of the 3 best models**
- With **Kmeans** --> **72 anomalies in 2019**
- With **IsolationForest** --> **1 anomaly in 2019**

But represent extrema in weather reports ==> no cyber hack.



Conclusion

Ambitious project: Supervised Regression + Unsupervised Classification

==> we did it !

To go further :

- times series with hourly data
- simulate a datastream with modified data

