Load Forecasting during the Covid Period

Halvard Bariller, Angel Reyero

M1 Mathematics & Artificial Intelligence
Institut de Mathématiques d'Orsay (IMO)
Université Paris-Saclay

{*halvard.bariller, angel.reyero-lobo*}*@universite-paris-saclay.fr*

March 20, 2023

Overview

Challenge

- **Objective** : French electricity load forecasting during COVID period (15/04/2020 - 01/01/2021)
- **Training data** : daily observations from 01/01/2012 to 15/04/2020
- **Target** : Load (in MW)
- **Features** : 14 explanative variables, mix of categorical (Bank holidays, daylight savings, summer and Christmas breaks, weekdays...) and continuous (date, temperature, smoothed temperature, load with lags...)

Electricity Load

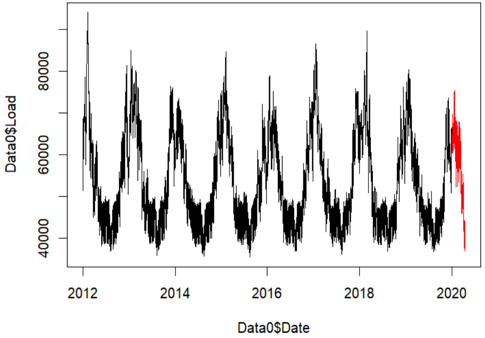


Figure: French electricity load between 2012-01-01 and 2020-04-15. Our objective is to forecast from ”2020-04-16” to ”2021-01-15”.
In black lines we have our training data (prior to 2020), and in red the cross-validation set data (from 2020-01-01 to 2020-04-15) we used to test our model and train the COVID corrections.

Our objective
Diversity

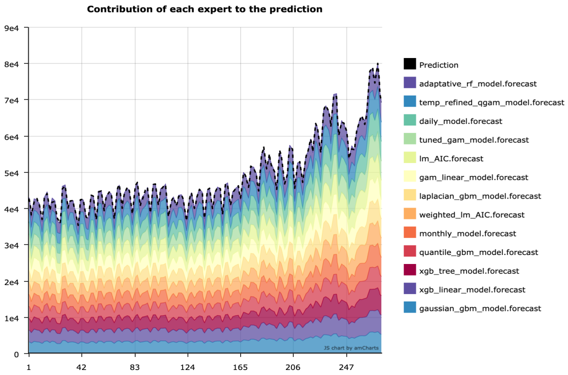


Figure: Our Online Expert Aggregation. Each model is able to capture some information at a specific period in time. We have sought for diversification in models, training data and loss functions.

Adapting the models

Learning from the new information

First Model: $\underbrace{d_1 \ldots d_n}$

Second Model: $\underbrace{d_1 \ldots d_n \ldots d_{n+uf}}$

Third Model: $\underbrace{d_1 \ldots d_n \ldots d_{n+uf} \ldots d_{n+2\times uf}}$

The first model is trained over $[d_1; d_n]$ (with n the initial window size).
Once we have observed the data until the $n + uf^{th}$ day (with uf the update frequency in days), we retrain our model over $[d_1; d_{n+uf}]$ and then we predict the uf following days.

Adapting the models

Problems

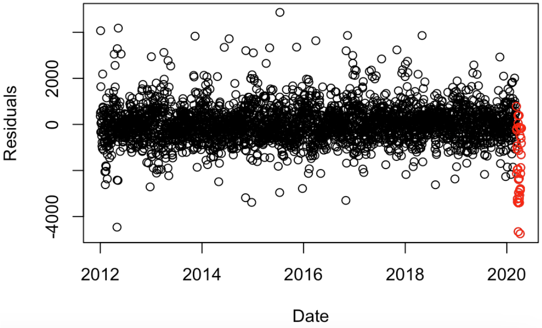


Figure: Unluckily, if we use all the training data, as the incorporated COVID data represents only a tiny part in the whole dataset, it is treated as outliers : we are not able to learn from it.

Adapting the models

The rolling window

First Model: $\underbrace{d_1 \ldots d_n}$

Second Model:$d_1 \ldots d_{uf}\ \underbrace{d_{uf+1} \ldots d_{n+uf}}$

Third Model:$d_1 \ldots d_{2\times uf}\ \underbrace{d_{2\times uf+1} \ldots d_{2\times n+uf}}$

As we want to learn from the COVID data, instead of increasing the original window size for training, we will shift the window on a regular time-basis defined by the *update_frequency* parameter.

Linear Model with a quadratic combination of the explanatory variables

AIC Criterion for Variable selection

```r
# lm_AIC <- linear_model_AIC(Data0[sel_a,])
lm_AIC <- lm(Load ~ Load.1 + Load.7 + Temp + Temp_s95 + Temp_s95_max + Temp_s99_max +
                       toy + WeekDays + BH + DLS + Summer_break + Christmas_break +
                       Time + Load.1:Temp + Load.1:Temp_s95 + Load.1:Temp_s95_max +
                       Load.1:Temp_s99_max + Load.1:toy + Load.1:WeekDays + Load.1:DLS +
                       Load.1:Summer_break + Load.1:Christmas_break + Load.7:Temp_s95_max +
                       Load.7:Temp_s99_max + Load.7:DLS + Load.7:Summer_break +
                       Load.7:Christmas_break + Temp:Temp_s95 + Temp:Temp_s99_max +
                       Temp:WeekDays + Temp:DLS + Temp:Christmas_break +
                       Temp_s95:Temp_s95_max + Temp_s95:Temp_s99_max +
                       Temp_s95:toy + Temp_s95:WeekDays +
                       Temp_s95:DLS + Temp_s95:Summer_break + Temp_s95:Christmas_break +
                       Temp_s95:Time + Temp_s95_max:Temp_s99_max + Temp_s95_max:toy +
                       Temp_s95_max:Christmas_break + Temp_s95_max:Time +
                       Temp_s99_max:toy +Temp_s99_max:WeekDays + Temp_s99_max:BH +
                       Temp_s99_max:DLS + toy:WeekDays + toy:BH + toy:DLS +
                       toy:Summer_break + toy:Christmas_break +
                       WeekDays:BH + WeekDays:DLS + WeekDays:Summer_break +
                       WeekDays:Christmas_break +
                       WeekDays:Time + BH:DLS + BH:Summer_break + BH:Christmas_break +
                       BH:Time + Summer_break:Time, data= Data0[sel_a,])
```

Figure: We have multiplied all the explanatory variables to get more flexibility. Then, we have selected the explanatory variables using the AIC criterion with bidirectional stepwise selection.

Linear Model Hypothesis violation

Residuals Homoscedasticity: QQplot

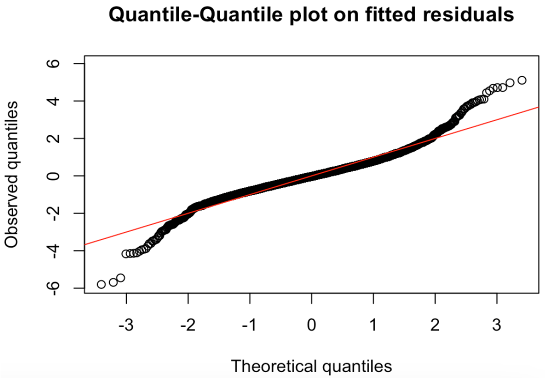


Figure: Graphically, the residuals are not distributed following a $\mathcal{N}(0, \sigma^2)$-distribution

Linear Model Hypothesis violation

Residuals Homoscedasticity: Breusch Pagan test

$\mathcal{H}_0 = $ *Residuals Homoscedasticity*
$\mathcal{H}_1 = $ *Residuals Heteroscedasticity*

```r
library(lmtest)
#perform Breusch-Pagan test
bptest(lm_AIC)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  lm_AIC
## BP = 570.22, df = 118, p-value < 2.2e-16
```

Figure: We reject the null hypothesis.

AIC penalty on polynomial combinations for Weighted Linear Model

Weighted Linear Regression model :

$$r_i(\beta) = y_i - f(x_i, \beta)$$

$$S = \sum_{i=1}^{n} W_{ii} {r_i}^2, \qquad W_{ii} = \frac{1}{{\sigma_i}^2}$$

It distributes weights on the observations such that those with small error variance are given more weight since they contain more information compared to observations with larger error variance.

Residuals Correction

Finding the trend and the seasonality

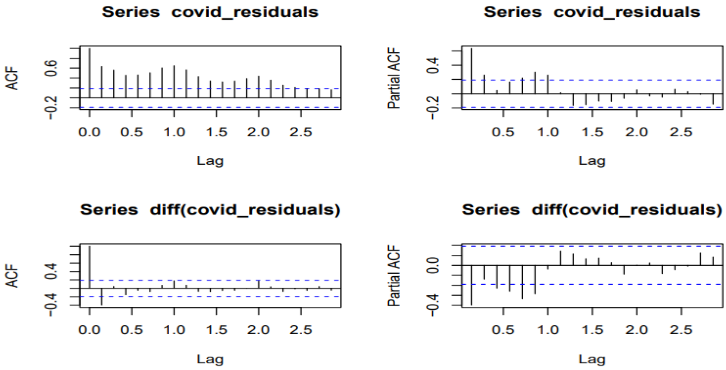
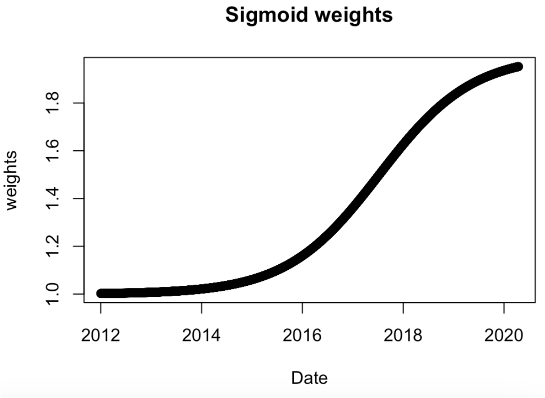


Figure: Residuals auto-correlograms (top) : we represent the empirical autocorrelations between the residuals (left) and the partial empirical residuals autocorrelations (right).
We do the same after applying a differentiation (bottom).

Weighting the Loss Function

Sigmoid Forgetting Factor

The objective is to give more importance to the most recent observations as we are dealing with time series. Hence, with these weights, the latest observations will be counted twice as much as the oldest observations.



Adaptive Weighted Linear Model with AIC criterion

Pseudo-code

We want to adapt the model with an update frequency uf. To simplify notations, we suppose that the first day of the test set is d_n, with n the window size.

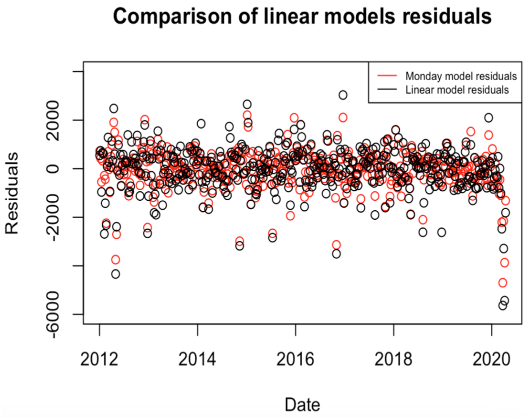
Algorithm Adaptative WLM

Result: Load Forecast from d_n to d_{end}

```
model_variables=stepAIC in [d_0; d_n − 1]
model=LinearModel with model_variables and sigmoid weight
for i ← 1 to end − n do
    if i%uf == 0 then
        model_variables=stepAIC in [d_{n+i−uf}; d_{n+i−1}]
        model=LinearModel with model_variables and sigmoid weight
    end
    We use the model to compute the forecast of d_{n+i}
end
```

Specialized LM with ARIMA corrections

Daily LM improvement



	Linear model	Monday model
Mean	-49,54	-43,69
Standard deviation	964,27	801,3

Figure: We notice how an specific LM with ARIMA improves the general LM with ARIMA that we have trained before over its period of action.

Variable selection with Random Forests

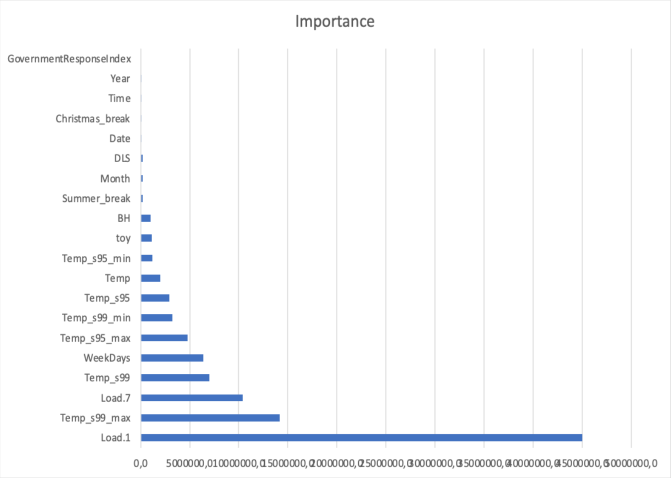


Figure: To have an initial idea of the variables to choose for our GAM, we have built a RF and studied the variable importance.

Variable selection with the LM AIC

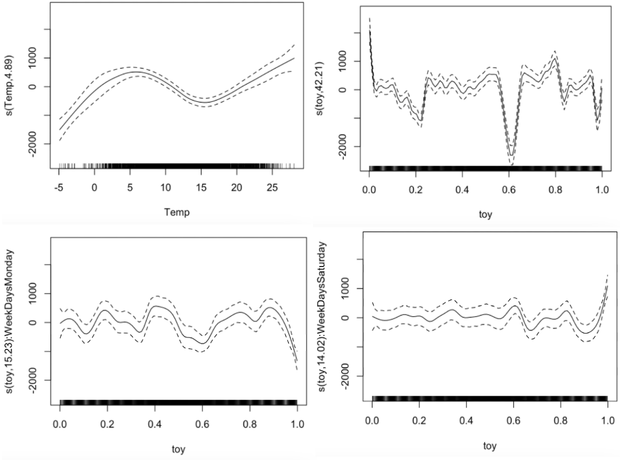


Figure: This GAM is able to capture smoothing effects that our linear models did not capture.

Selecting the quantile

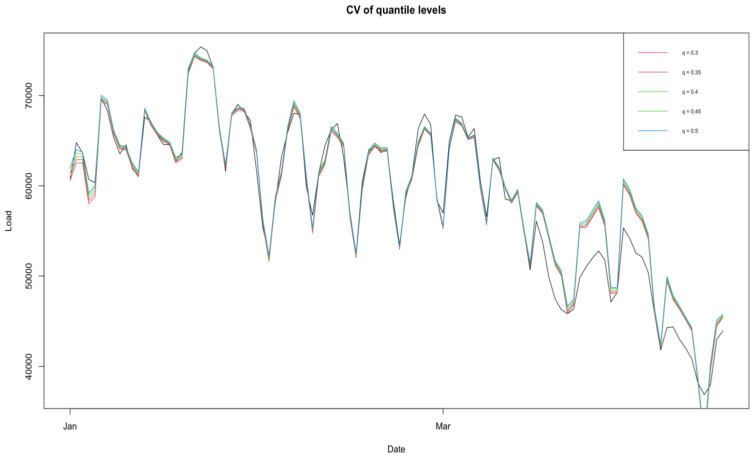


Figure: We have noticed that with the Covid period, the Load was sharply reduced and our predictions were going to overestimate the Load. To model this effect, we have tested quantiles under the median.

Adaptive Random Forest

OOB evolution

Using the rolling window idea, we have trained an Adaptive Random Forest

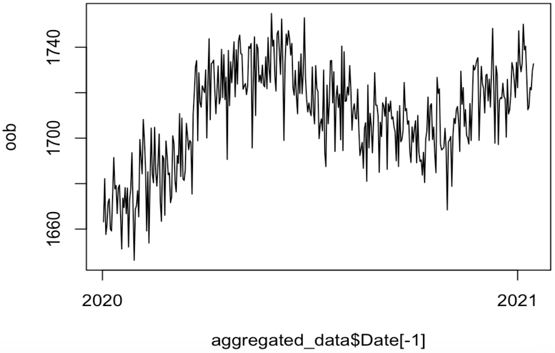


Figure: We notice that with the beginning of Covid period, we suffer from an increase in the OOB error. Nevertheless, as long as we get more Covid data, the OOB error improves, before degrading again on the period corresponding to an increase in activity.

Generalized Boosted Models (GBM)

We have tested the GBM with three losses: Squared Error, Absolute Loss and Quantile Loss(at 0.3). We have weighted the method with the Government Response Index variable.

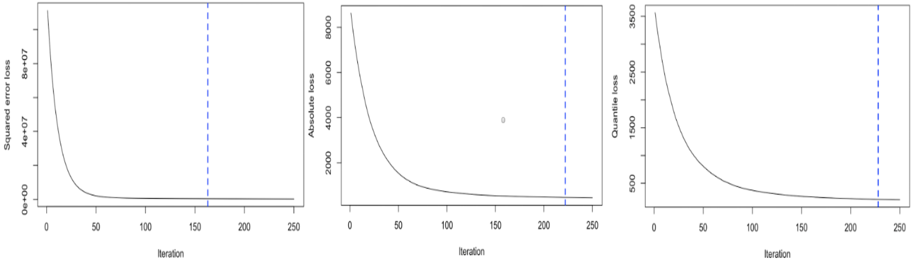


Figure: We have selected the number of iterations with the OOB error.

eXtreme Gradient Boosting(XGBoost)

We have used the CART and Linear Model as weak-learners.

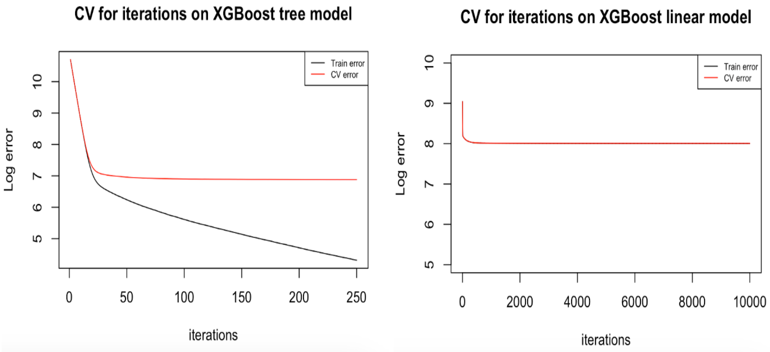
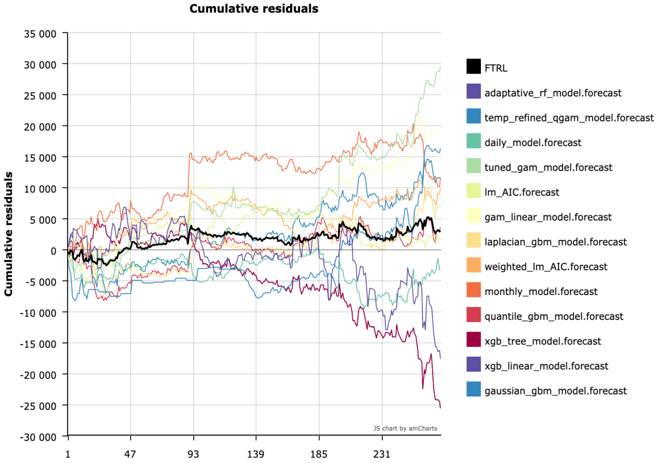


Figure: Using the CV, we were able to early stop the number of iterations of the algorithm if we did not get improvements.

Final Model

FTRL

We have finally mixed the predictions from our models to leverage their diversity.



Thank you for your attention

Questions?

{halvard.bariller, angel.reyero-lobo}@universite-paris-saclay.fr