**ML**
**TD1 – feature engineering**

An online merchant asks you to make real-time predictions for the customers of his website: when they are ready to buy, tell them the date of receipt of their product, for the various possible transport services. (Chronopost ..).
To do this, you have 6 weeks of order history with 3 pieces of information
- Date and time of the customer's order
- Shipment date (to simplify considered as order shipment date)
- Transport service

In production
- You will have a daily update (every night) of the warehouse status (this same updated file)
- You will need to be able to deliver real-time on-the-fly prediction with each new online order. The speed of your predictive model will be as important as its accuracy

**You have 3 hours to build your project strategy and your model**
**How do you take the problem?**

Do you agree to make this prediction? if so, with which commitment (s)?
The question is do we have enough data to make a prediction? as a first approach .. no because we do not know anything about the state of the system for each order: how many staff? Machine operation? accumulated delay? what are the public holidays?
As a second approach .. the answer is "maybe" because the history allows us to reconstruct a lot of data on the state of the system: we just have to calculate them (delay, production capacity, ...). The warehouse is a black box but we know its entrances and exits
What commitment? none on performance because if we can reconstruct data nothing tells us that the most useful data will really be there. Our only commitment is to respond quickly (1 agile cycle to have a first performance assessment and to know if it is worth investing more.

**What will be your strategy**

- target: how to predict a date?

The prediction of a model is done in the universe of data known in the past. Orders for this week or next month are mechanically outside the dates known during learning.
The right target is not a date but a deadline: how long between the date of order and the date of shipment. This is the first value to calculate on our training game. This period will be in days because if we have an order time we only have a shipping date.

- classification or regression (we are looking for a discrete numerical value ...)

As it is often the case in ML ... we don't know and the only answer is we have to test. The low value range (99% between 0 and 5 days) is a good incentive to try a classification approach.
In both cases, it may be wise to limit the outliers> 5 days to 5 days: your model has too little data to look for cases that are by nature very specific.
Dans les deux cas, supprimer les valeurs aberrantes (délai négatif) est requis

- How to select the test set? -…

This point is major! the tradition is to select a random test set. This is relevant when the data is IID (independent and identically distributed). This is clearly not the case here. When the warehouse is late, all orders are impacted. It's always best to think of this as the model's use case: it will apply to the new order flow. Taking the last two weeks of the dataset into the test set is therefore more honest but will not allow all possible configurations to be validated. The available data window is reduced, your model will be too. Here again, a little humility is required: it must be announced that the model will continue to learn as the history is enriched.
Once your test set is validated it is essential to calculate the performance of a naïve model. For example on a random test game, the main delay mode is 2 with 35% orders.

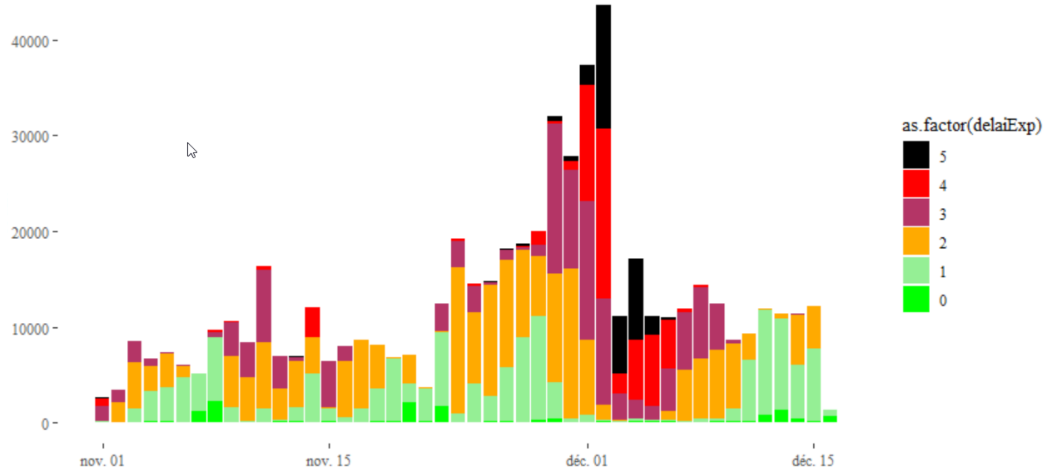
This serves as a benchmark to qualify the added value of the model: having 35% correct answers!

The following illustrations are based on a random test set. But keep in mind the previous remark

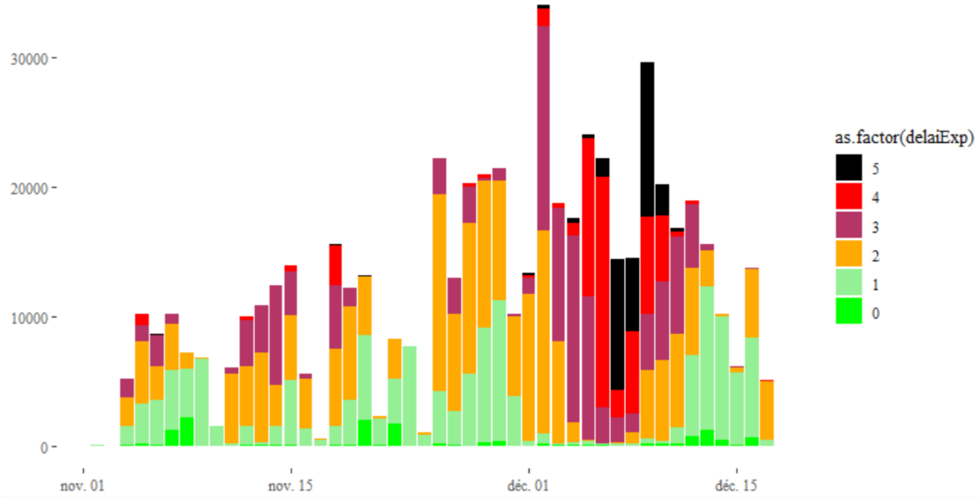
**Data Discovery:**

- Visualize the flows,

Here is an example: order volume per day with colors indicating the shipping time



The same by date of shipment



- Make observations

Over the period we observe

- First graph:
  - An order spike during Black Friday
  - Rare same day shipments
  - High order variability, even before Black Friday (variations from single to double)
  - o …
- Second graph
  - An activity 7 days a week even if at the beginning of the period it is very reduced on Sundays
  - A disturbance effect of order peaks that spans several days
  - No FIFO / LIFO management: each day seems to distribute its activity over orders of the day and orders that are already old. There is a management rule for the activity to be sought .. may be inaccessible with the limited data available.
  - With two exceptions, the warehouse is always operating at full capacity (there is almost always 2-day stock left)

It is clear that these patterns can change over time. Regular (daily?) Relearning is necessary to detect organizational changes

Many other visualizations would be useful to deepen the understanding of the flow and imagine interesting features: this work is never finished

**Feature engineering**

- How to enrich the model with new features? Knowing that no other features can be given to you

As a reminder, the target is the first data to calculate. The shipment date column can then be deleted.

Of course the dalai must be calculated with an order date and not an order date / time ...

Secondly, we could reconsider this date and run a model, for example to predict / understand, the level of activity on Sunday for example, and deduce a specific feature that allows this information to be learned in the model.

Beyond the target, some features are obvious

- Weekday of the order
- Order time
- The service provider id is to be passed in string / factor: it is an identifier not a measure (in some cases an identifier is interesting as a numerical value .. but not here)

Others are to be imagined, for example

- Warehouse stock: any order received and not shipped
- Part of the stock which is 2 days (or 4 days, or ..) or more late
- Order volume from the day before after 6 p.m. (or 7 p.m. ...)
- Warehouse production capacity: shipment volume of the day before (or max day before the day before because Sunday is clearly not representative of a nominal capacity)
- Level of delay, expressed in stock / daily processing capacity

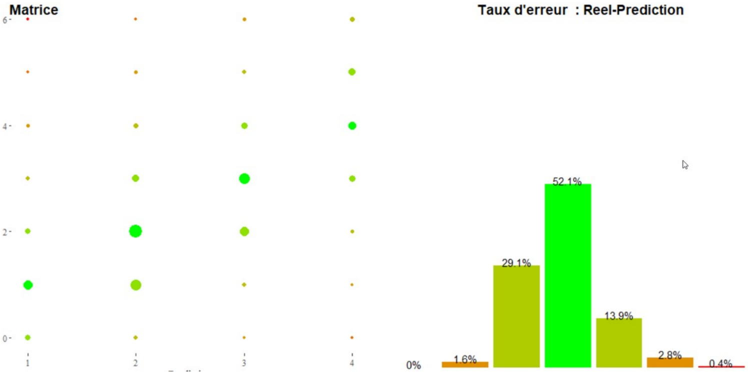
The construction of new features is done by analyzing the data, by analyzing the important features of the first test models (where it may be useful to refine), but especially with operational staff: we will not practice this business survey. in progress but this is the essence of your job!

- **Model selection**
- - test different algorithms

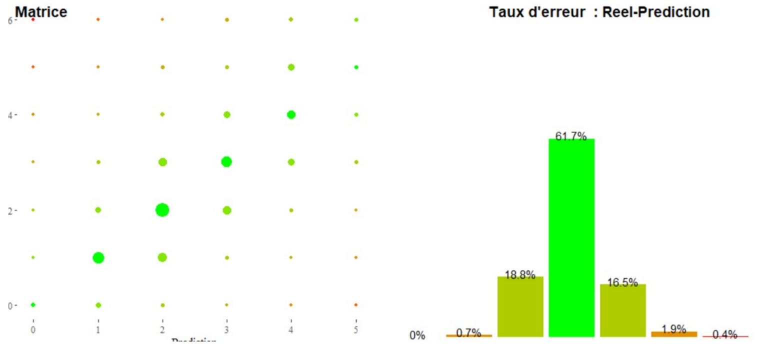
Every model needs to be tuned: don't just settle for the default settings. This step takes time, remember to parallelize your tests on different clusters of your PC. For example in R doparallel package

- visualize your results

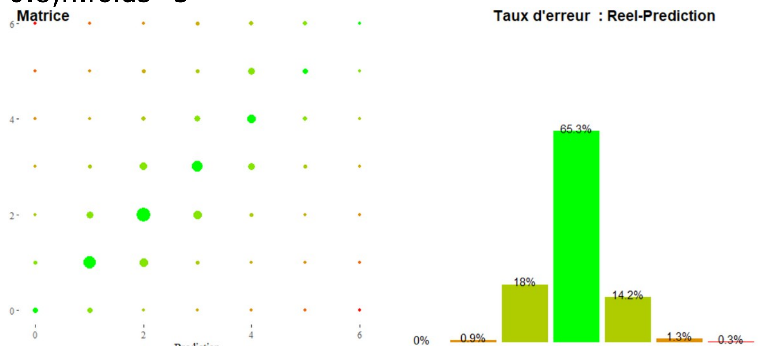
Decision Tree : Confusion Matrix and rate or error



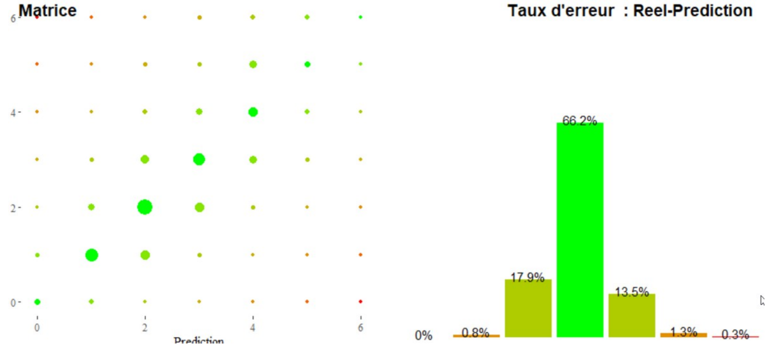
Random Forest – réglage mtry=2



Gradient boosting – réglage tree.complexity = 6,learning.rate = 0.02, bag.fraction = 0.8,n.folds=5

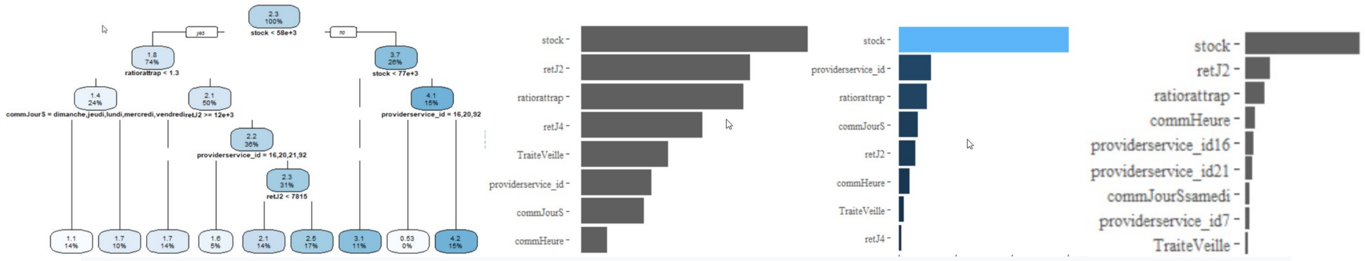


XGboost -réglage



- analyze your important geatures

Tree Random Forest GBM XGBoost



- Deduce in new features?

Calculate stocks by service provider and number of days late for example?

To go further: cross-importance analysis (combinations of important features, visualization of features and important combinations of features (identify specific thresholds or patterns

**How do you go to production**

- You receive in production only a date and time (real order time) and not from the transport service (not yet chosen by the customer): with this only information and the update of the orders shipments from the warehouse updated during at night you must predict the calendar dates associated with each transport service (the customer will choose)

- How to reduce the response time?

The model takes little input information: A date and time, a service provider
And concretely the prediction is the same between 3:03 pm and 3:47 pm .. the model can turn every hour before even receiving new orders.
It is therefore a matter of predicting as many delays as there is service provider (because this data cannot be anticipated and the associated delays recorded. When placing a new order ... the predictions are already ready to send!

-