OpenClassRooms

Data Scientist

P5 Customer clustering for an e-commerce website

Developped on a Jupyter Colaboratory Notebook

Pictures used for educational purpose only

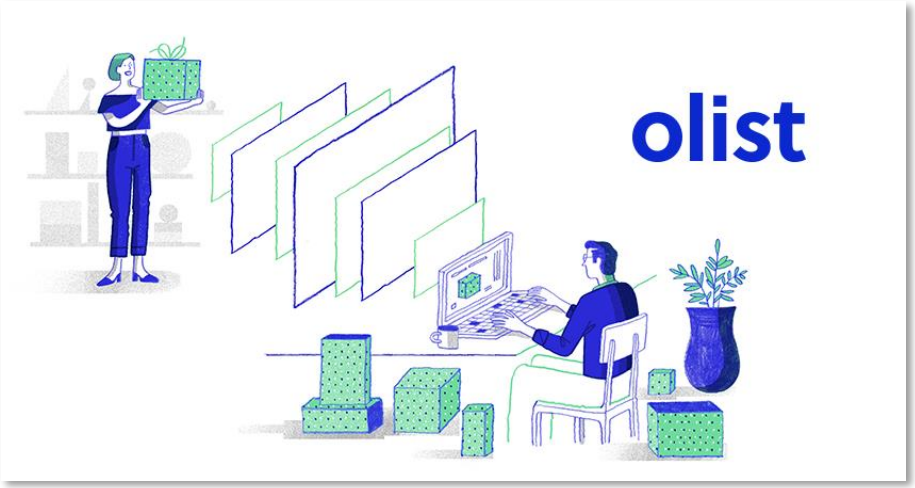
Benoît DELORME
Creation : 19/04/2021
Update : 18/05/2021

Summary

I. Introduction

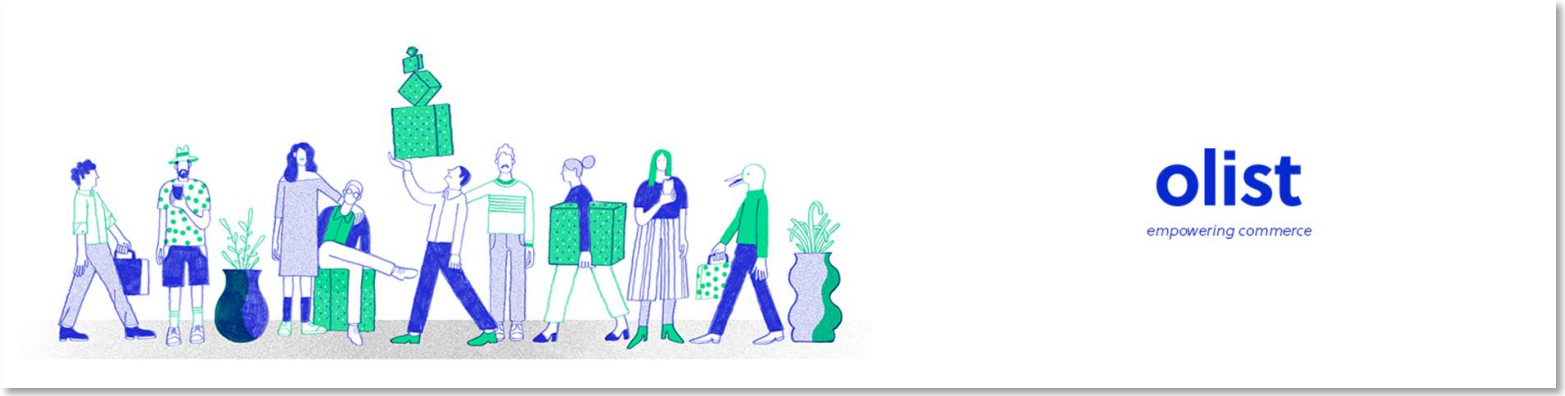
1. Olist: activity and needs

Created in 2015



E-commerce

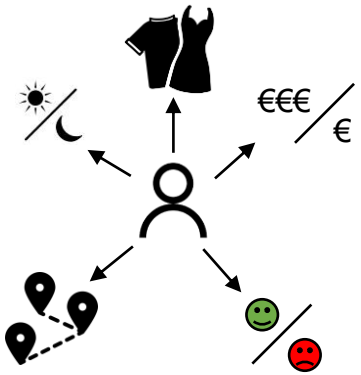
Main activity in Brazil



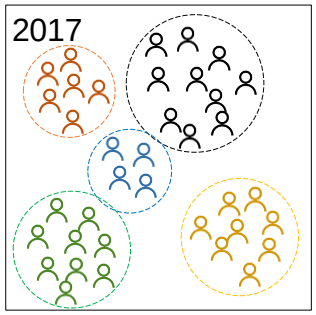
→ Needs to identify different types of customers, in order to optimize marketing campaigns

I. Introduction

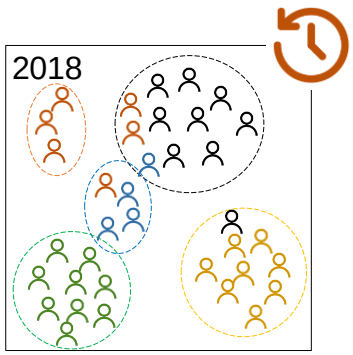
2. Specifications



1) Distinguish different types of customers
   a) using their purchase behaviour
   b) using the available data



2) Use non-supervised methods in order to gather similar customers.



3) Propose maintenance recommendations, based on an analysis of groups stability over time.

I. Introduction

3. The deliverable

Customers clustering

Client that purchase on week-ends, important budgets, …

Client that purchase decoration products, ordering preferably in the morning, …

Client unsatisfied, suffering delivery delays, …

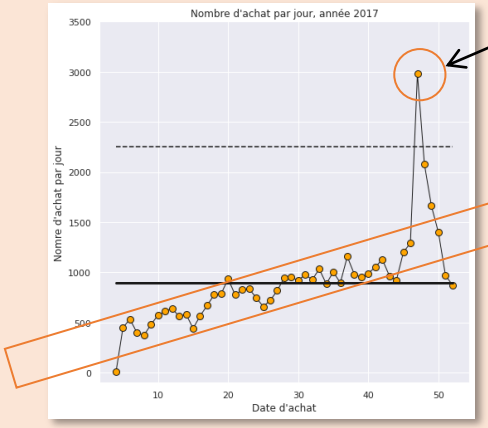
Client geographically isolated, low budgets, …

…

→ Final goal: focus marketing campaigns according to the client types

Update frequency recommendation

Consider **punctual events** (holidays, brazilian bank holidays, …)



Consider **general trends** (growing access to the Internet, Olist own growth, …)

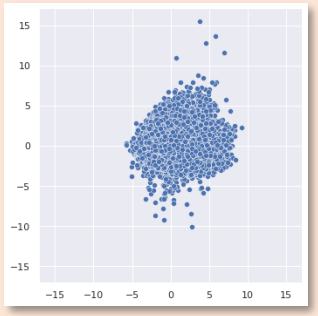
→ Final goal: assure clustering liability with a relevant update frequency.

I. Introduction

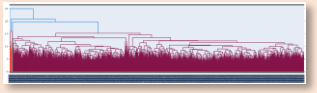
4. Available algorithm & metrics

Visualization algorithms

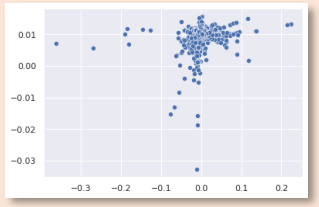
PCA



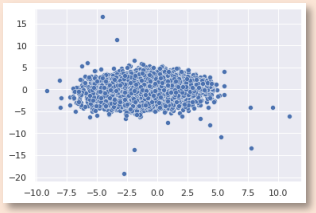
Dendrogram



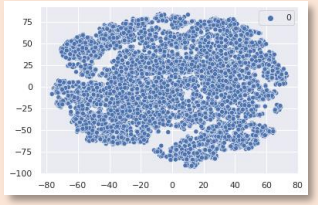
LLE



MDS



t-SNE

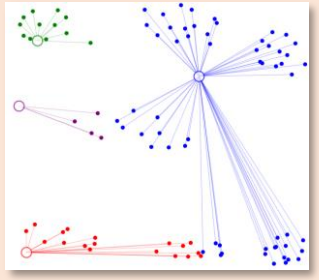


Evaluation metrics

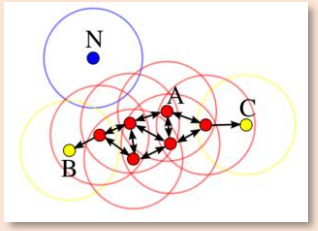
- Inertia (k-means)
- Silhouette coefficient
- Calinski-Harabasz coefficient
- Davies-Bouldin coefficient
- Noise proportion (DBSCAN-specific)

Clustering algorithms

k-means / k-prototypes

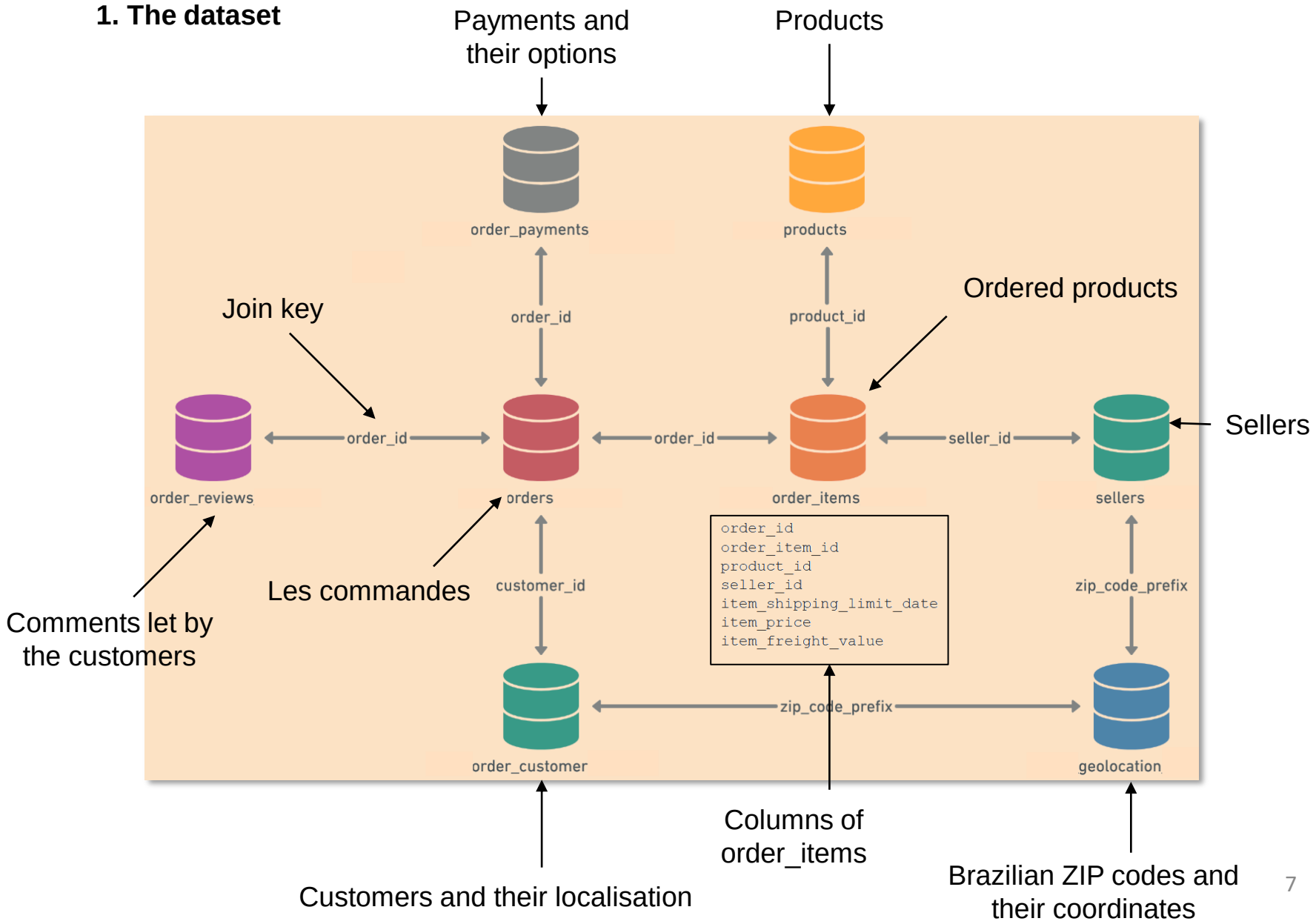


DBSCAN



II. Exploratory analysis and feature engineering

1. The dataset

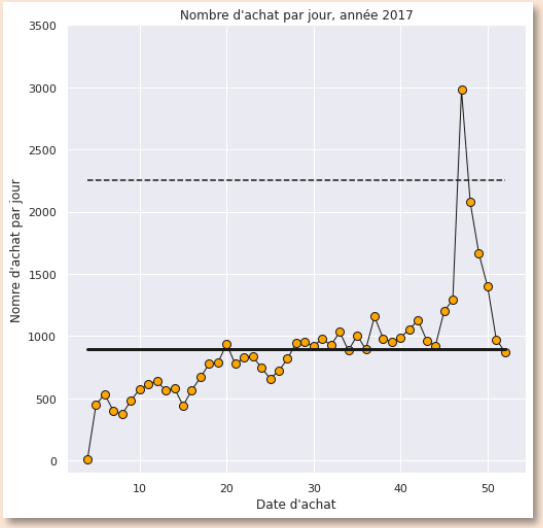


II. Exploratory analysis and feature engineering

2. User behaviour

« RFM » analysis

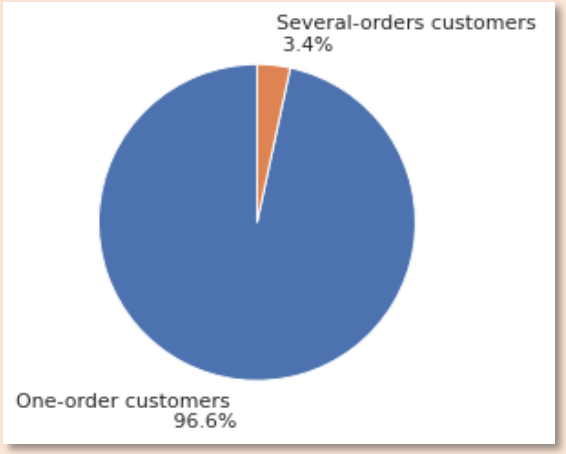
Recency
How recently did the customer purchase?



↑ Purchase per week, 2017 ↑

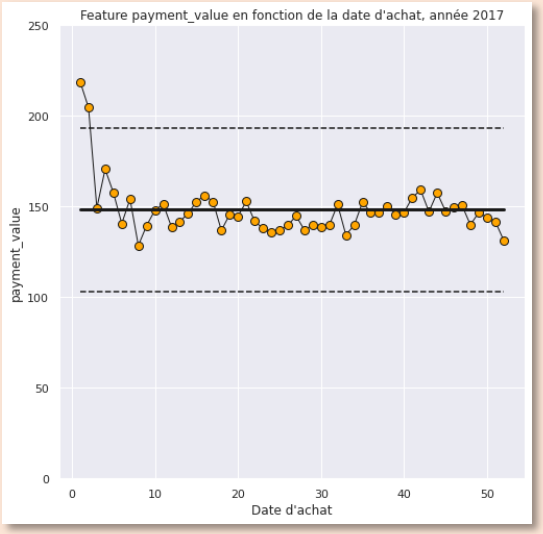
→ Date of purchase is present in the final dataset.

Frequency
How often do they purchase?



→ The majority of the customers ordered only once.
The purchase of frequency thus concerns only a small portion of customers: this criteria will not be kept for modelisation.

Monetary value
How much do they spend?



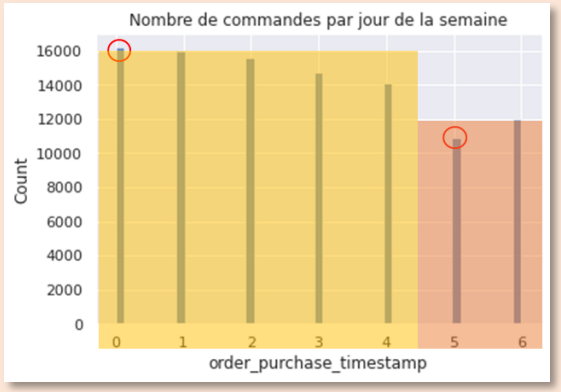
↑ Purchase amount per week, 2017 ↑

→ Payments amount is present in the final dataset.

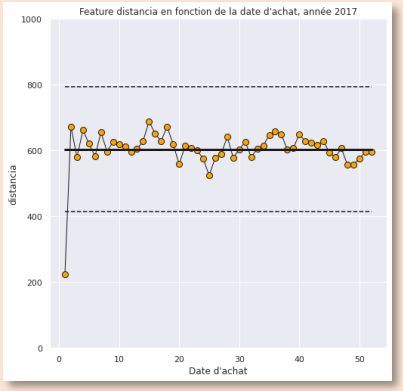
II. Exploratory analysis and feature engineering

3. Features created

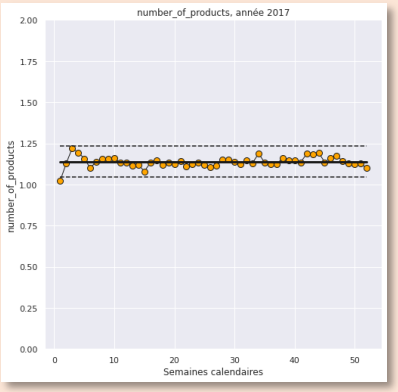
Week days



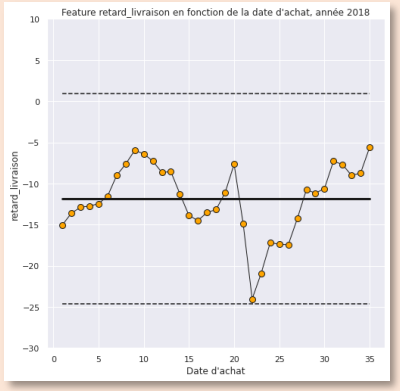
Distance



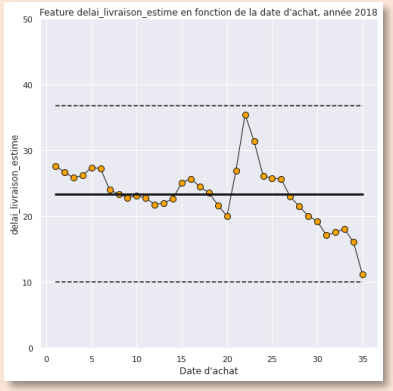
Product number per order



Delivery delay



Estimated delivery time



Deleted features

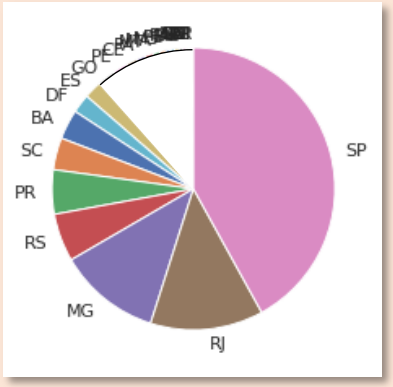
Date of comment,
Date of shipping,
Product dimensions,
…

→ Some features present important variations and/or imbalances.

II. Exploratory analysis and feature engineering

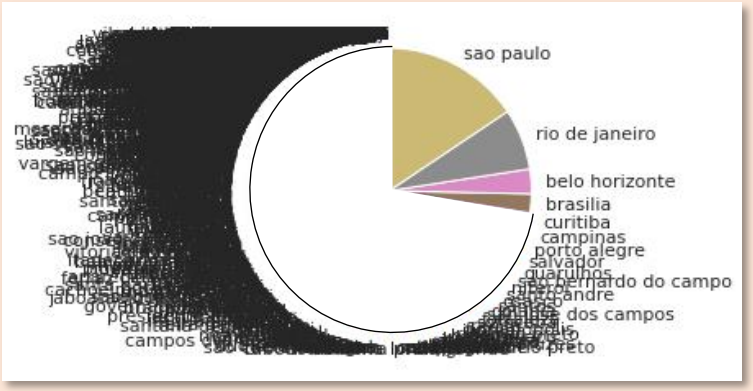
4. Encoded features

States : > 2% of total



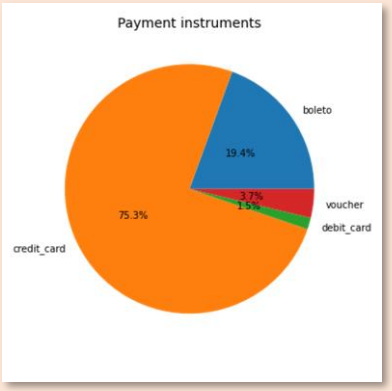
+ 9 features

Cities : > 2% of total



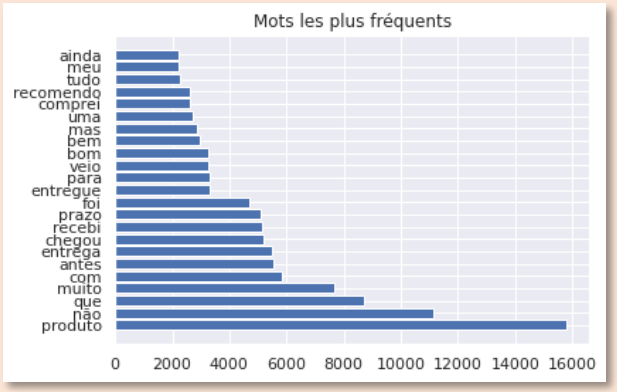
+ 3 features

Payments instruments



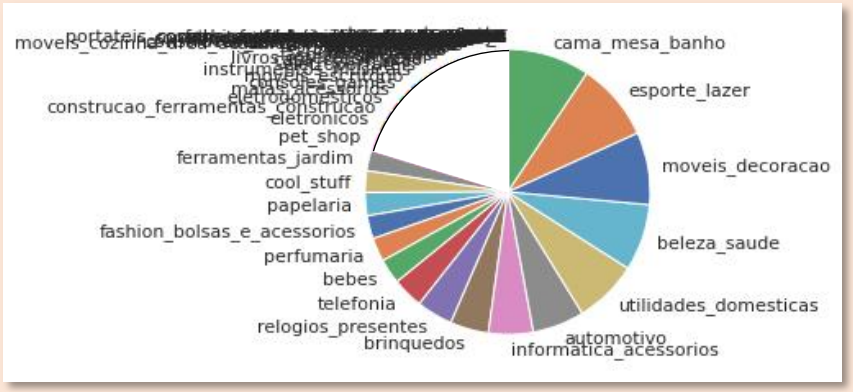
+ 3 features

Comments: case by case



+ 11 features

Product category : > 2% of total

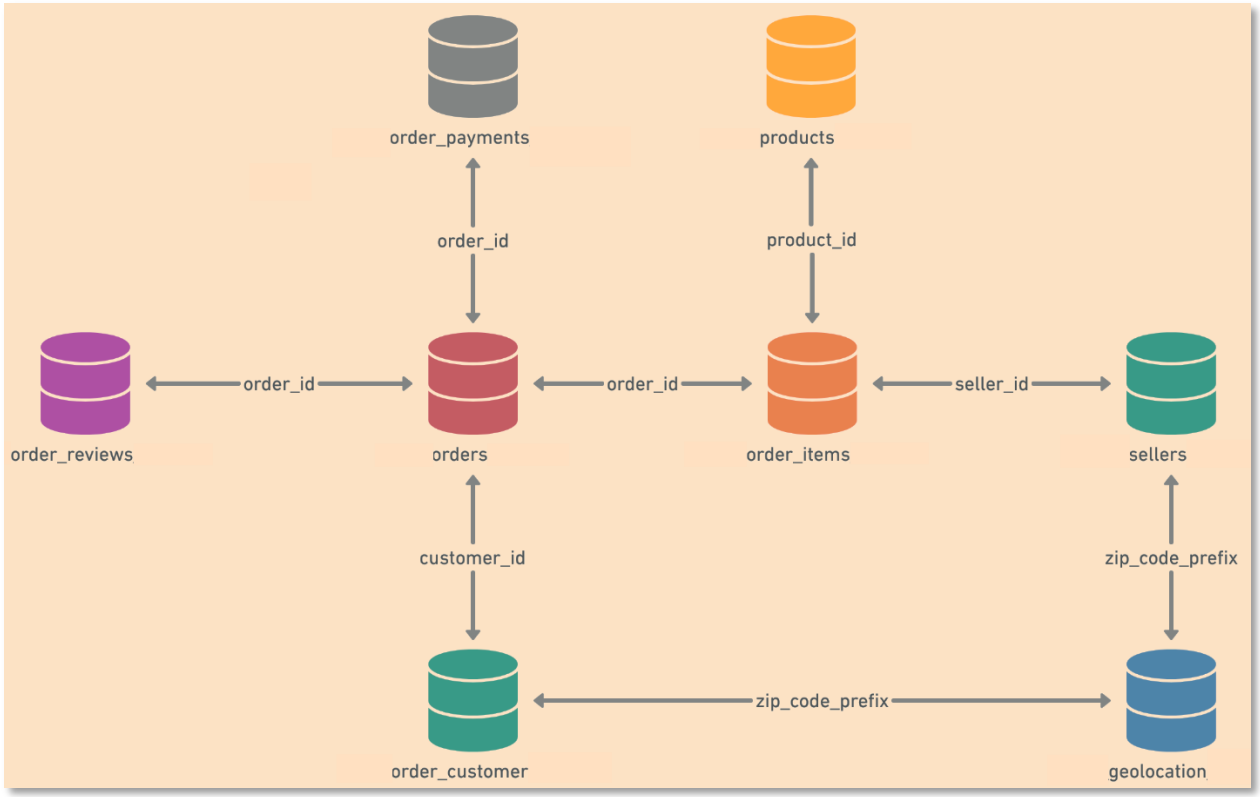


+ 15 features

→ Encoding assessment: 41 features created

II. Exploratory analysis and feature engineering

5. Merge



Method

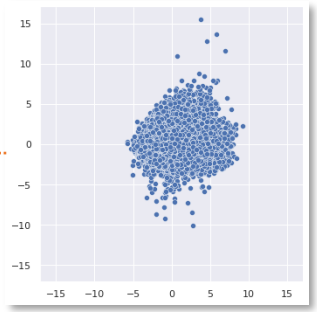
1. One order, one line
2. For each order, let's keep some features raw:
   a. payment instrument;
   b. product category;
   c. comment words.
3. For other features, we use :
   a. volume;
   b. weight;
   c. number of images.

	order_id	item_price	number_of_products	product_weight_g	product_volume_cm3	item_freight_value
0	e481f51cbdc54678b7cc49136f2d6af7	29.99	1	500.0	1976.0	8.72
1	128e10d95713541c87cd1a2e48201934	29.99	1	500.0	1976.0	7.78
2	0e7e841ddf8f8f2de2bad69267ecfbcf	29.99	1	500.0	1976.0	7.78
3	bfc39df4f36c3693ff3b63fcbea9e90a	29.99	1	500.0	1976.0	14.10
4	8736140c61ea584cb4250074756d8f3b	75.90	1	238.0	3000.0	7.79

III. Modelisations tries-out

1. Steps

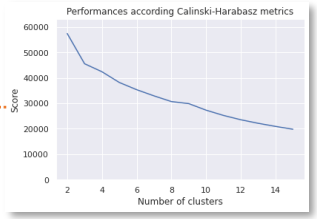
Visualize



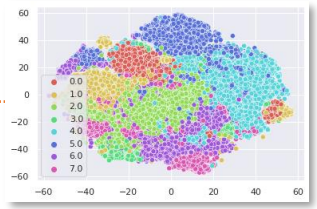
Modelise

```
k_means(
    my_data=temp_df,
    n_clust=8
    )
```

Optimize

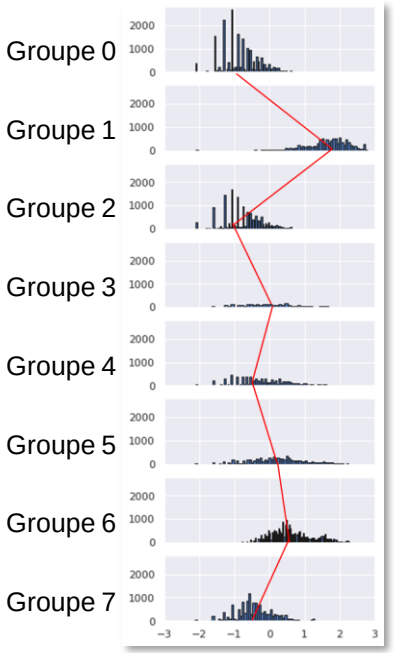


Represent



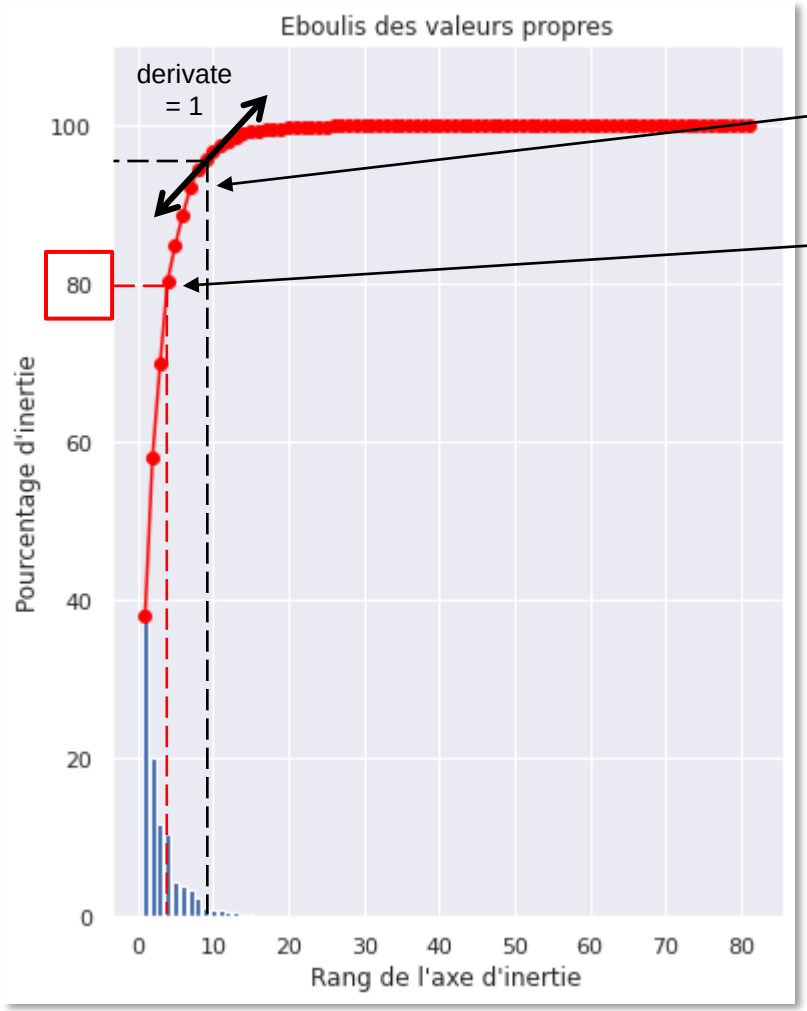
Analyse

Example : orders total weight



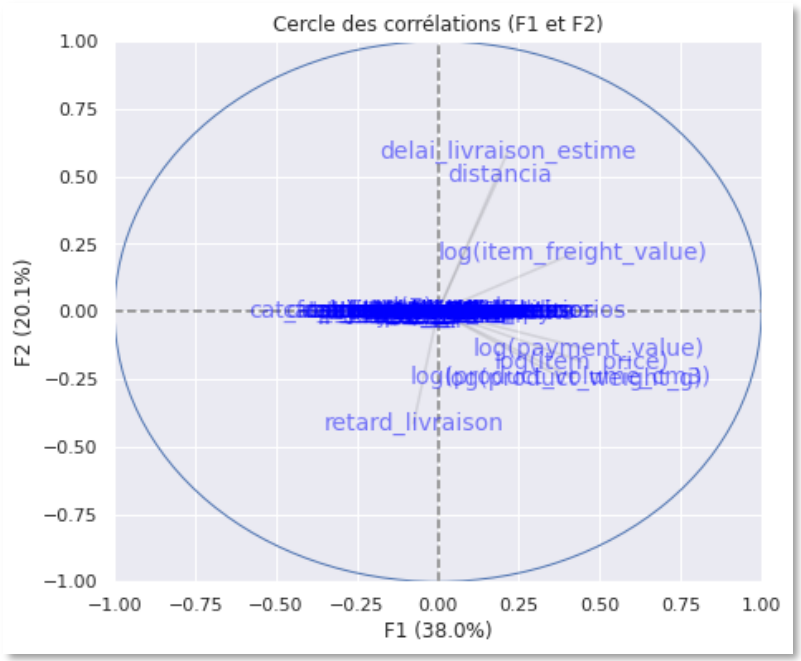
III. Modelisations tries-out

2. Principal components analysis



Learning rythm: ≈ 10 features are enough

Learning level: ≈ 4 features are enough



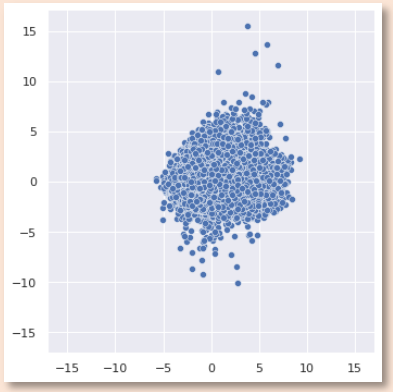
→ A new dataset is created, using the 10 most important principal components.
Modelisations will be done on this dataset, and in parallel on the original one.

III. Modelisations tries-out

3. Visualizations

Projection in principal plan

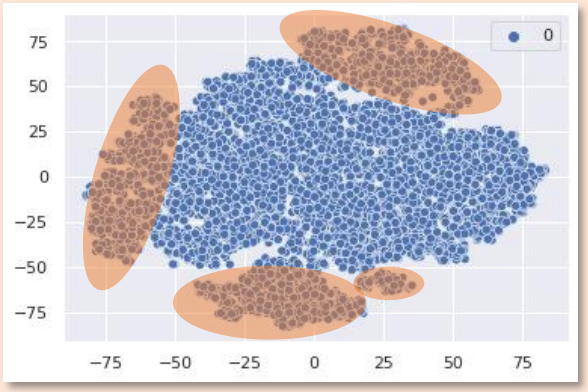
PCA



Interpretation : pas de segmentation discernable

Complete dataset

t-SNE

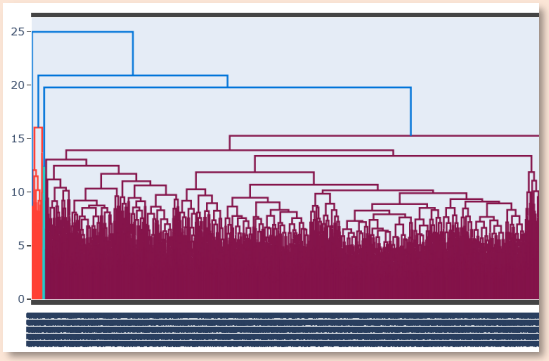


Interpretation : 5 groups, maybe more

10% of dataset

Ascending clustering

Dendrogram



Interpretation : 4 groups, maybe more

10% of dataset

→ Expecting between 5 and 10 groups on the complete dataset

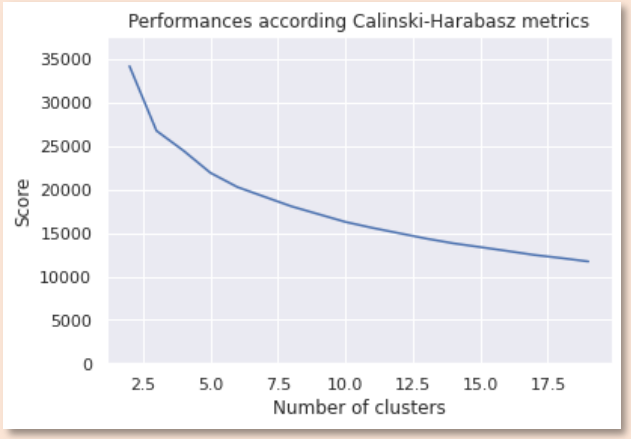
III. Modelisations tries-out

4. k-means

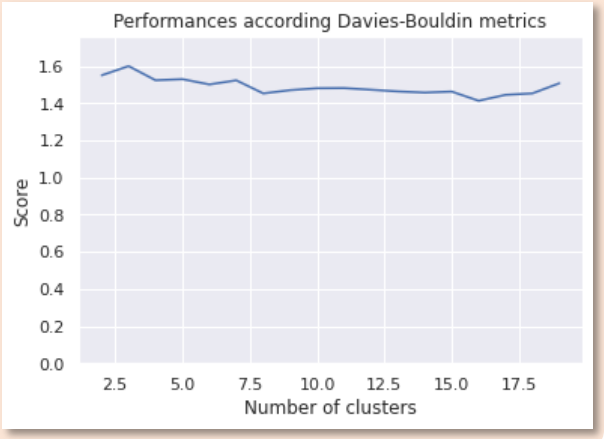
Metrics

Inertia

Calinski-Harabasz



Davies-Bouldin



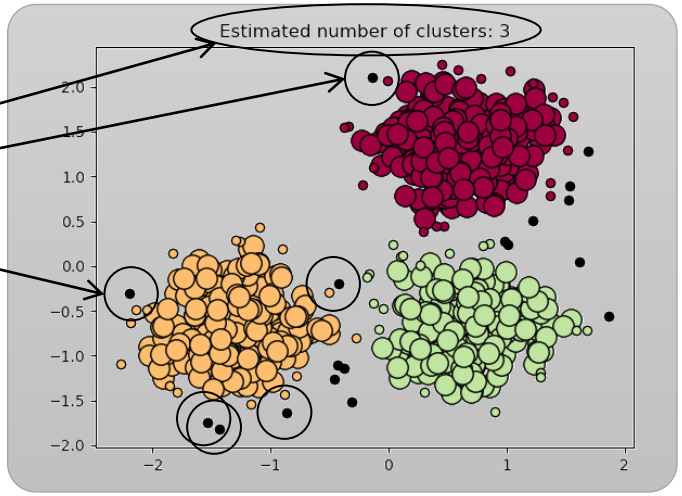
(Long computation time for Silhouette coefficient)

Visualization t-SNE on 10% of data

III. Modelisations tries-out

5. DBSCAN

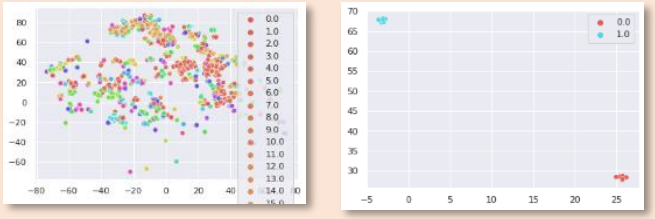
DBSCAN algorithm naturally provides with:
- the number of identified groups
- the number of noise points



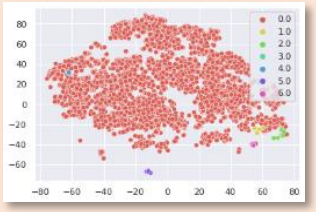
Disadvantages

DBSCAN is very long, and cannot finds a just balance:

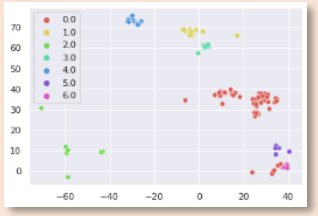
Too many (360) or too few (2) groups



Imbalanced group sizes



Too many noise points (≈ 50.000)



Poor metrics

```
Calinski-Harabasz coefficient: 814.382
Davies-Bouldin coefficient: 1.913
```

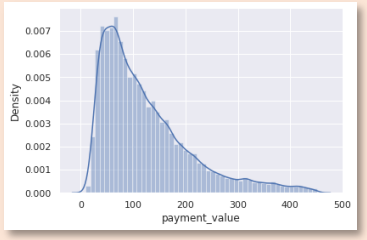
III. Modelisations tries-out

6. k-prototype

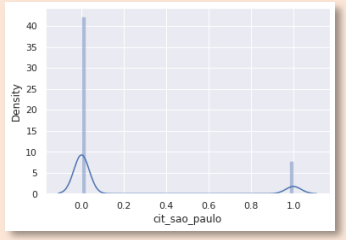
Problematic

k-means computes distances between points. But when there is a lot of data encoded in 0 & 1, the distance is less relevant.

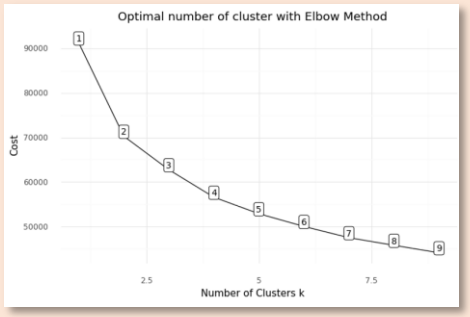
Quantitative distribution



Binary distribution (encoded data)



Solution: k-prototype algorithm



Disadvantage: very long, almost 25 minutes for one modelisation only.

Alternative: transform the encoded data and give them a quasi-quantitative distribution.

payment_value
38.71
37.77
37.77
44.09
83.69

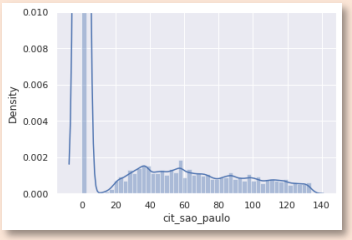
✖

sta_SP
1.0
1.0
1.0
0.0
1.0

=

sta_SP
38.71
37.77
37.77
0.00
83.69

Modified distribution



IV. Model optimization

1. Design of experiment

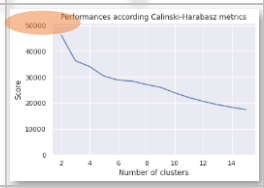
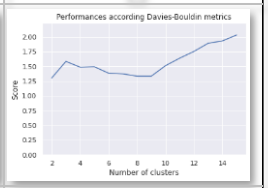
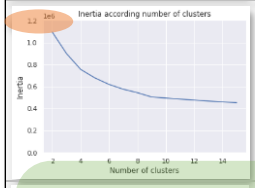
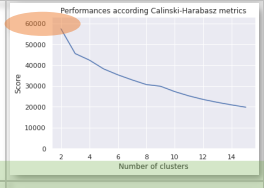
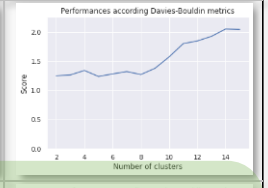
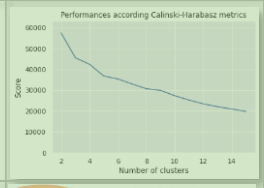
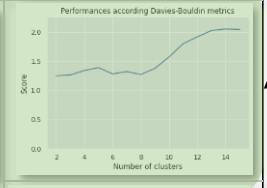
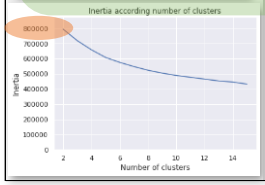
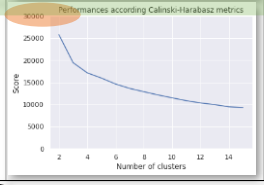
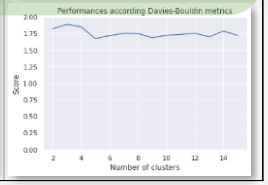
Hyperparameters

Combinations

Sup. values / Inf. values

Inputs		
n_init	max_iter	tol
12	360	0,0002
12	360	0,00005
8	360	0,00005
8	240	0,0002

Outputs

Inertia	Silhouette Coefficient	Calinski-Harabasz Index	Davies-Bouldin Index
			
			
			
			

Performances

Best performances

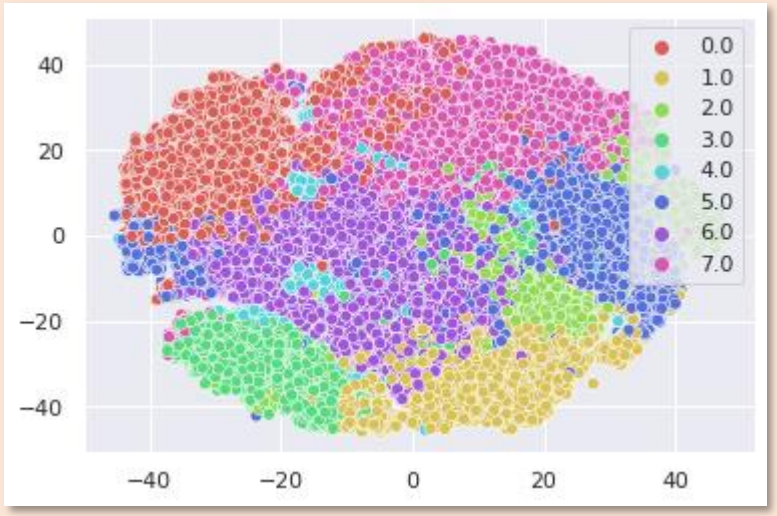
(Silhouette very long)

IV. Model optimization

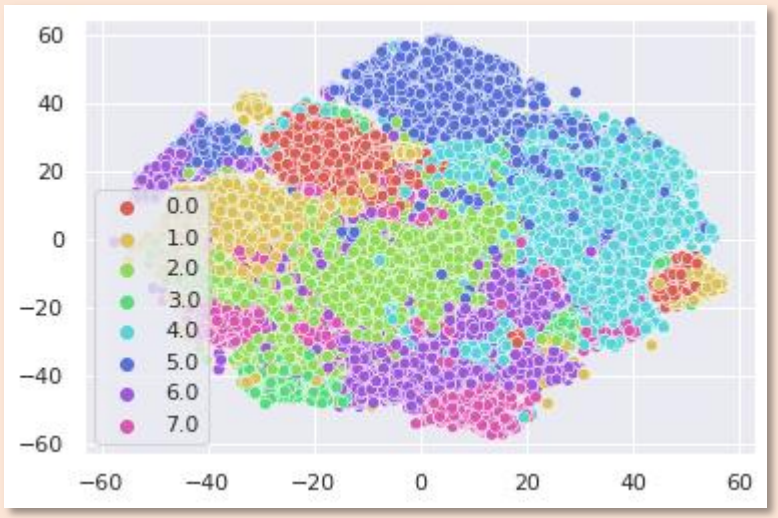
2. Groups visualization

Projection in t-SNE plan
Colors correspond to the groups found by k-mean

PCA applied (10 components)



No PCA applied (71 features)



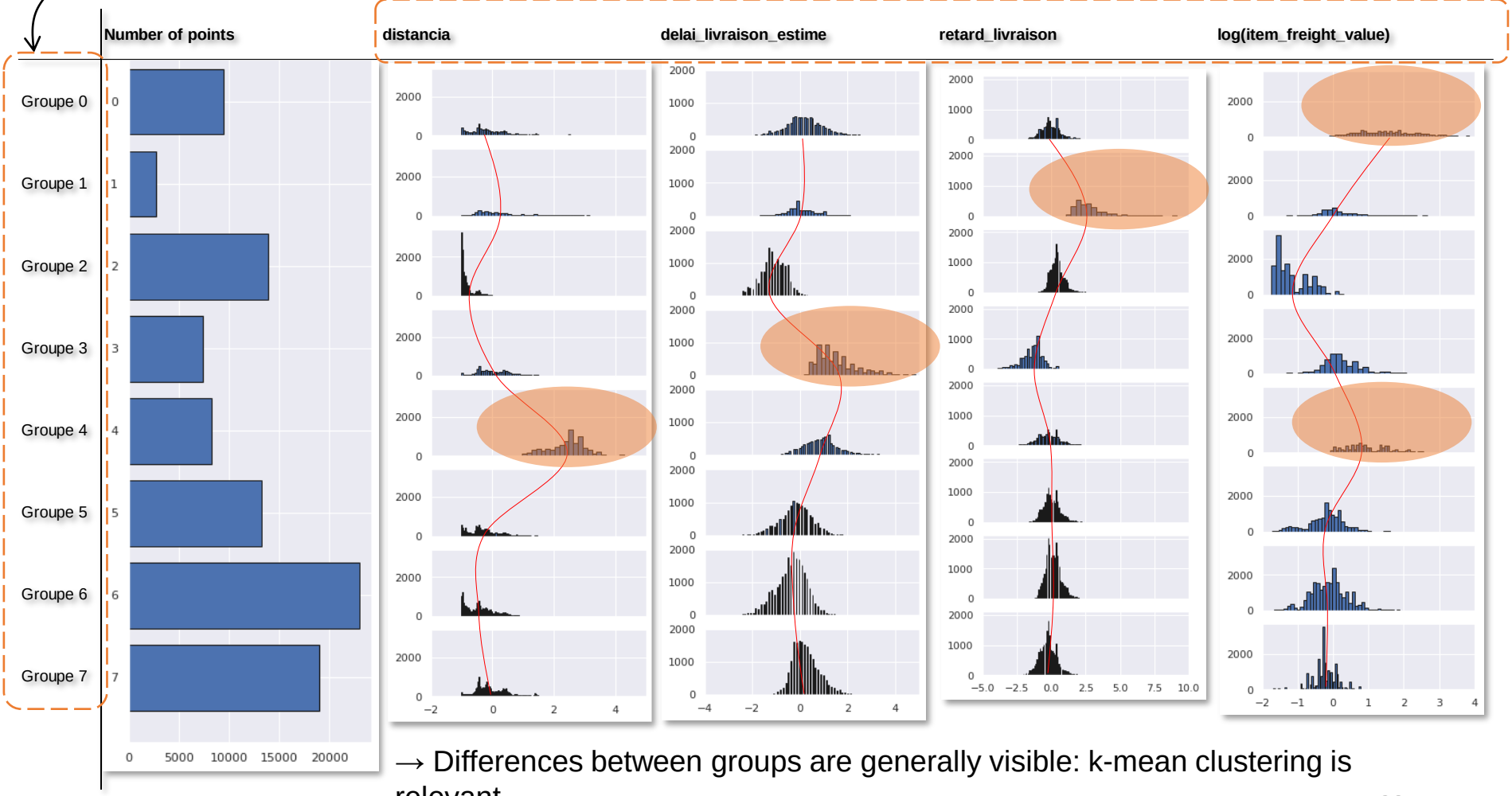
→ Consistency between **visualization** algorithm t-SNE &
clustering algorithm k-mean for this dataset

IV. Model optimization

3. Manual check of differences between groups

Groups identified by k-mean

Some features

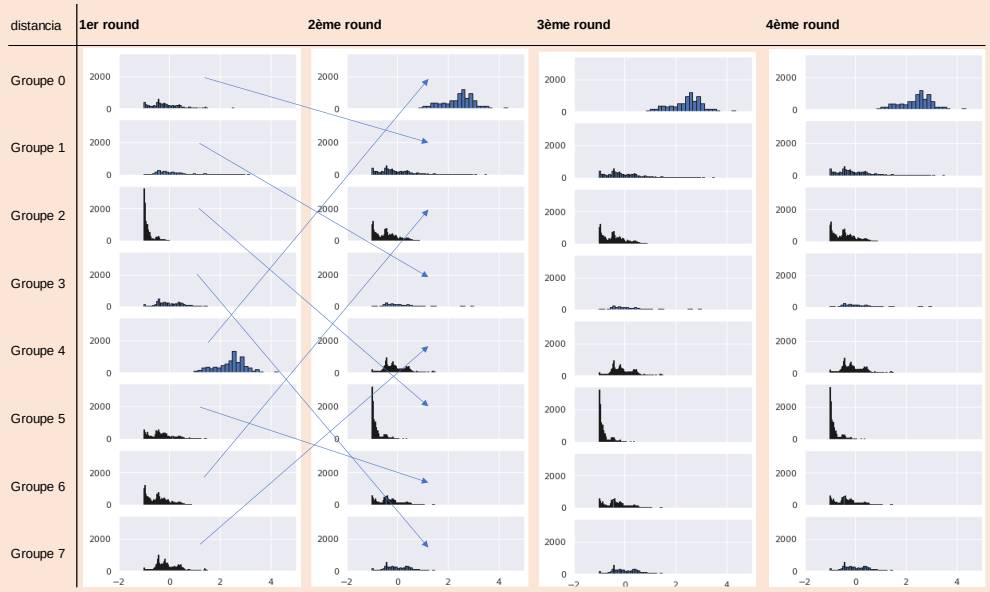


→ Differences between groups are generally visible: k-mean clustering is relevant.

IV. Model optimization

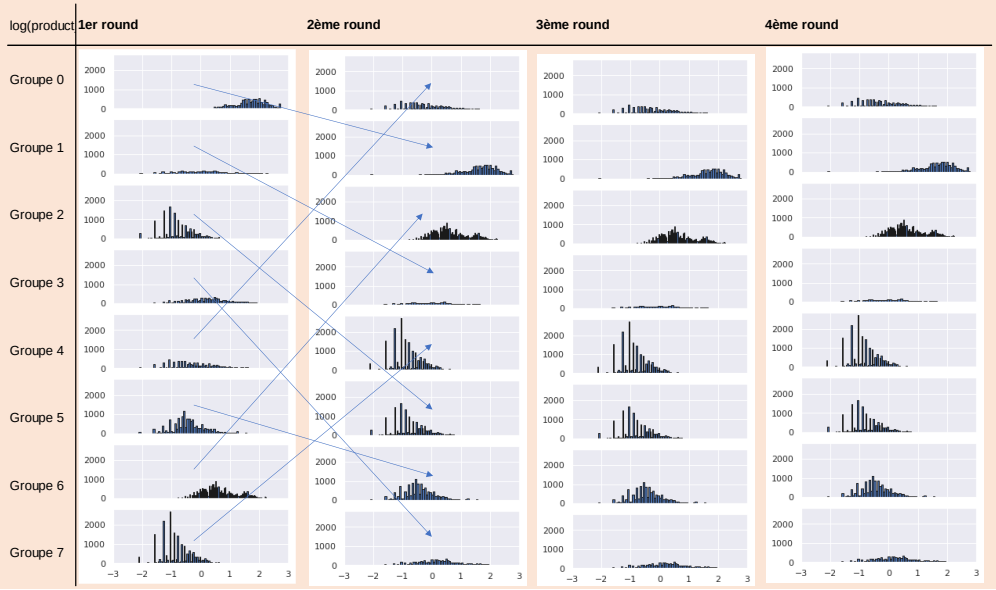
4. k-mean stability

Distance



→ Clustering stable

Weights



→ Clustering stable

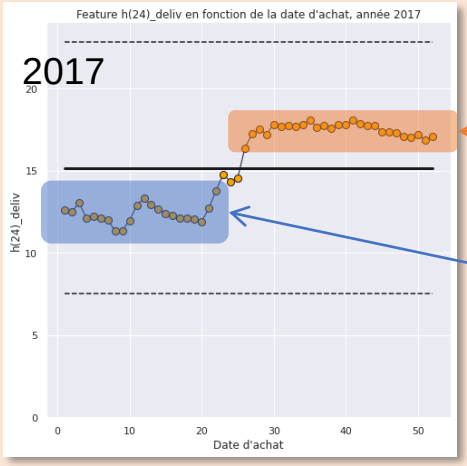
IV. Model optimization

5. Deliverable

IV. Model optimization

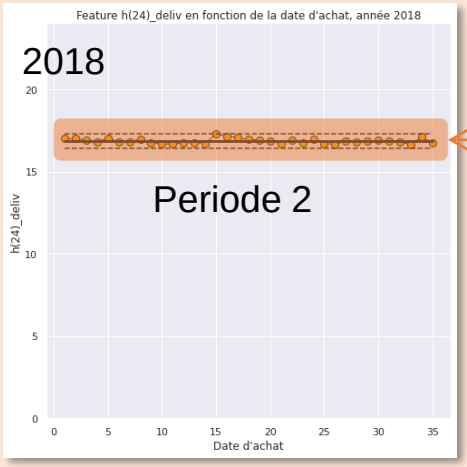
6. Update frequency

Delivery hour evolution
(→ does not need clustering step)



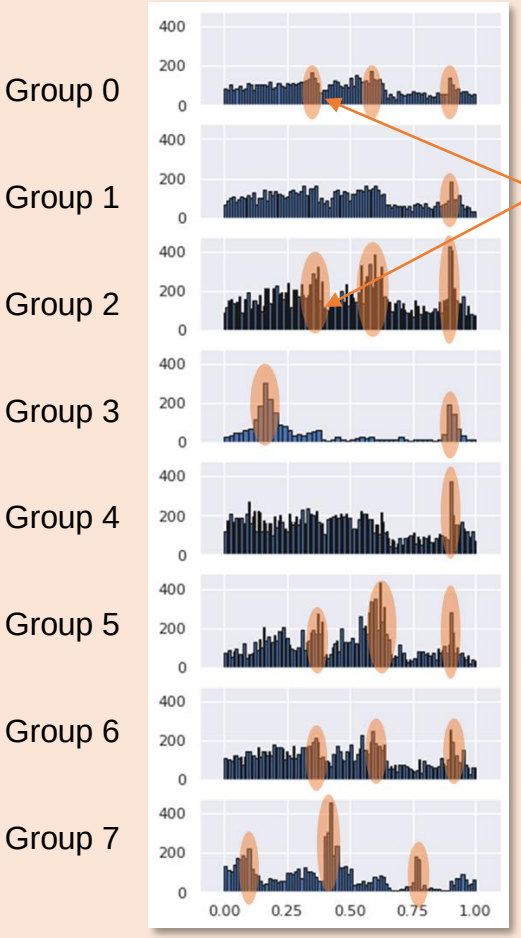
Period 2
16H30

Period 1
12H00 - 13H00



→ **punctual** trends
→ relaunch the algorithm according to the **sudden variations** of features.

Distribution of purchase dates in the different groups
(→ needs clustering step)



→ **general** trends
→ relaunch the algorithm according to the **identified periods**.

V. Conclusion

Dataset	1) Dataset can be used for the current problematic. 2) The final dataset contains more than 98.000 elements, for 100.000 originally.
Algorithms	The algorithms return similar results on clustering, excepting DBSCAN (not relavant ofr this dataset ?)
Clustering	1) The groups found by k-mean show particularities for each of them. 2) Clustering is stable on several launches.
Update frequency	1) Update the algorithm every quarter. 2) Nonetheless, let's keep in mind to monitor the sudden variations that may affect some features.
Efficiency check	Recommendation: use the algorithm on one half of the dataset only, and evaluate the impact of marketing campaign on the two populations.

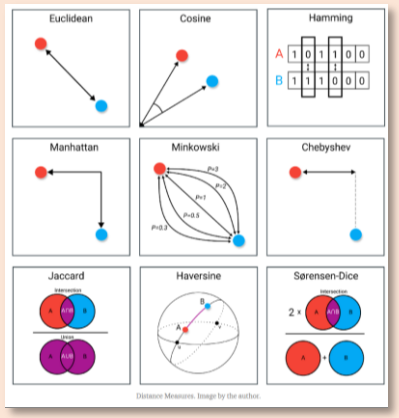
V. Conclusion

6. Outlooks

Study of the faithful customers

Textual analysis of comments

Use other types of distances



Use UMAP

Use unique customer identifiers

End of the presentation

Thank you for your attention