

Clustering de courbes de charge EDF

Benjamin Auder ¹

Jairo Cugliari ²

¹CNRS Orsay / Université Paris-Sud

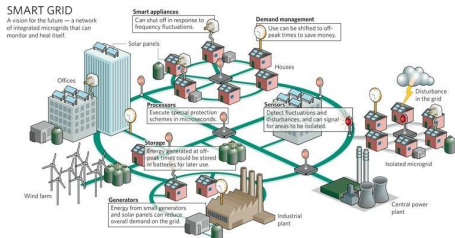
²Laboratoire ERIC / Université Lumière Lyon 2

Contexte industriel

Smartgrid & Smart meters : 35M compteurs individuels donnant de l'information en temps réel.

⇒ **Beaucoup** de données.

Comment les traiter ?



Des données variées, à différentes échelles

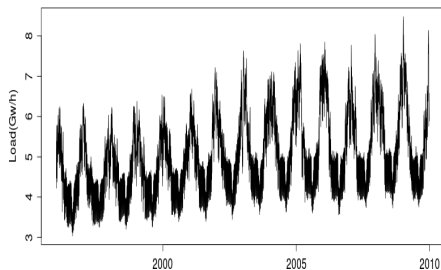


FIGURE: Tendence à long terme

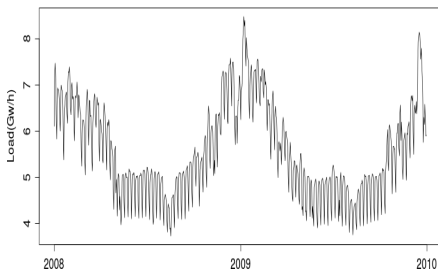


FIGURE: Cyclicité semaine

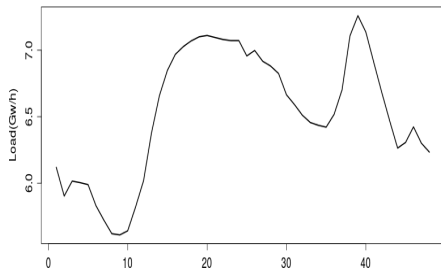


FIGURE: Moyenne journalière

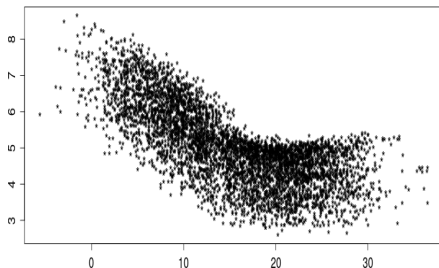
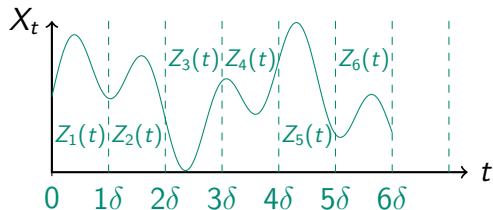


FIGURE: Conso. vs. température

Découpage en tranches non stationnaires

Si $\exists \delta \ll D$, tel que les séries δ -agrégées soient stationnaires, on les agrège et les traite comme des processus stationnaires.



$$Z_k(t) = X(t + (k - 1)\delta)$$

$$k \in \mathbb{N} \quad \forall t \in [0, \delta)$$

Mais... Une série temporelle représentant un phénomène complexe est en général clairement non stationnaire.

\Rightarrow On décide de tenir compte de chaque point de discrétisation.

Réduction de dimension

Données enregistrées toutes les 30 minutes pendant un an :
 $48 \times 365 = 17520$ points de discrétisation.

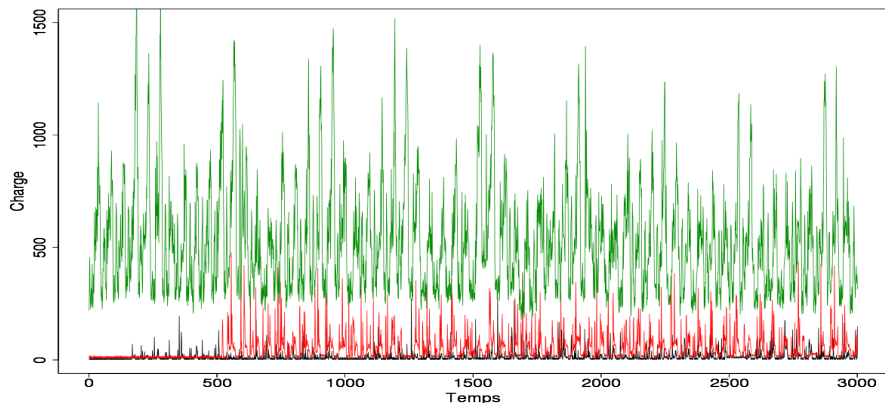
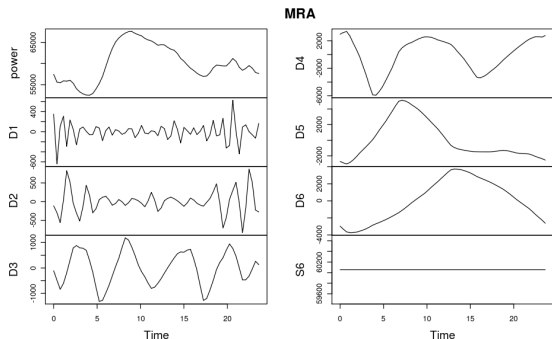


FIGURE: Trois types de courbes de charge (*données irlandaises*)

⇒ Il faut déterminer une représentation parcimonieuse, capturant bien les variations localisées. On choisit une base d'ondelettes.

Wavelets to cope with FD



- domain-transform technique for hierarchical decomposing finite energy signals
- description in terms of a broad trend (approximation part), plus a set of localized changes kept in the details parts.

Discrete Wavelet Transform

If $z \in L_2([0, 1])$ we can write it as

$$z(t) = \sum_{k=0}^{2^{j_0}-1} c_{j_0,k} \phi_{j_0,k}(t) + \sum_{j=j_0}^{\infty} \sum_{k=0}^{2^j-1} d_{j,k} \psi_{j,k}(t),$$

where $c_{j,k} = \langle g, \phi_{j,k} \rangle$, $d_{j,k} = \langle g, \psi_{j,k} \rangle$ are the scale coefficients and wavelet coefficients respectively, and the functions ϕ et ψ are associated to a orthogonal MRA of $L_2([0, 1])$.

Energy decomposition of the DWT

- Energy conservation of the signal

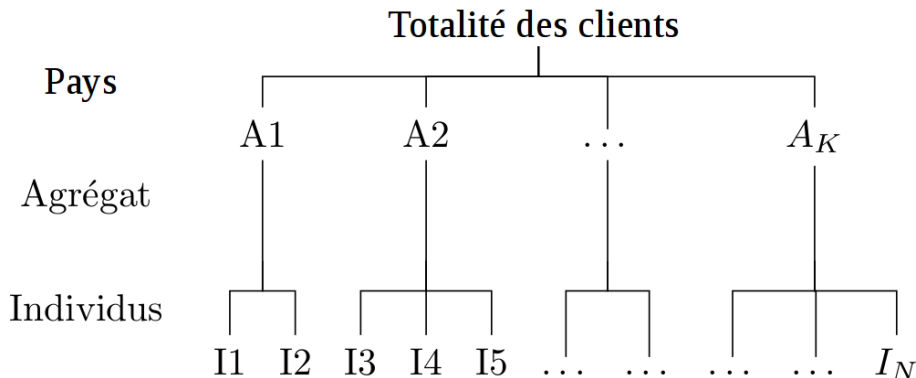
$$\|z\|_H^2 \approx \|\tilde{z}_J\|_2^2 = c_{0,0}^2 + \sum_{j=0}^{J-1} \sum_{k=0}^{2^j-1} d_{j,k}^2 = c_{0,0}^2 + \sum_{j=0}^{J-1} \|\mathbf{d}_j\|_2^2.$$

- For each $j = 0, 1, \dots, J-1$, we compute the absolute and relative contribution representations by

$$\underbrace{\text{cont}_j = \|\mathbf{d}_j\|_2^2}_{\text{AC}} \quad \text{and} \quad \underbrace{\text{rel}_j = \frac{\|\mathbf{d}_j\|_2^2}{\sum_j \|\mathbf{d}_j\|_2^2}}_{\text{RC}}.$$

- They quantify the relative importance of the scales to the global dynamic.
- RC normalizes the energy of each signal to 1.

Objectif



Regroupement par tarifs, zones géographiques, types de clients ...

⇒ **Idée** : clustering pour déterminer ces groupes.

Méthode : paralléliser un algorithme classique.

Fonction objectif

On cherche à minimiser la distorsion

$$\Delta = \sum_{i=1}^n \min_{k=1..K} \|x_i - c_k\|_2$$

avec pour variables les $\{c_1, \dots, c_K\} \subset \{x_1, \dots, x_n\}$, $c_i \neq c_j \forall i \neq j$.

C'est un problème NP-dur (O. Kariv & S. L. Hakimi, *An Algorithmic Approach to Network Location Problems. II : The p-Medians*).

Pire : garantir un facteur $(1 + \varepsilon)$ de l'optimum est NP-dur (J-H. Lin & J. S. Vitter ε -Approximations with Minimum Packing Constraint Violation).

NP : "Non-deterministic Polynomial-time algorithms"

Exécution en temps polynomial sur une machine de Turing non déterministe.

NP-dur

"Au moins aussi dur que le plus complexe des problèmes NP"

Algorithme PAM

- 0 Initialize : randomly select (without replacement) K of the n data points as the medoids.
- 1 Associate each data point to the closest medoid. (“closest” here is defined using any valid distance metric, most commonly Euclidean distance, Manhattan distance or Minkowski distance).
- 2 For each medoid m
For each non-medoid data point o *in the same cluster*
Swap m and o and compute the total cost.
- 3 Select the configuration with the lowest cost.
If any change occurred in the medoids, go to step 1.

Réduire le coût des étapes 2 et 3 ?

- Dans R, `pam(do.swap=FALSE)` supprime les étapes 2 et 3.
- A. P. Reynolds et al. (2006) : quelques astuces algorithmiques.

Parallélisation

Deux approches (entre autres)

- Découpage de l'espace en $Z < K$ zones, et recherche de K/Z clusters dans chaque zone.
- Partition des données P_1, \dots, P_Z puis clustering à K groupes dans chaque P_k . (Puis "fusion" des médoïdes).

Choix de la seconde alternative et implémentation avec OpenMPI :

- 0 Le processus "maître" a pour numéro 0. Il divise les données en sous-ensembles de cardinal au plus C ($C = 5000$ par exemple). Il envoie ensuite une tâche de clustering par sous-ensemble, et attend les résultats.
- 1 Chaque processus "esclave" (numérotés de 1 à $p - 1$) reçoit une liste de (références de) courbes, qu'il récupère et classe via l'algorithme PAM. Il retourne les centres au processus 0.
- 2 Si on obtient plus de C médoïdes, on recommence depuis l'étape 1. Sinon, on applique une dernière fois l'algorithme PAM (sur les médoïdes).

Exécution du programme

```
auder@wildrose:~/code/build - LilyTerm
auder ~/code/build $ mpirun -np 8 ./ppam.exe cluster ~/data/EDF/2009.bin 5000 10 0 2
0 / Send work /home/auder/data/EDF/2009.bin to rank=1 / 5002
1 / Slave pid=4860 work on /home/auder/data/EDF/2009.bin
0 / Send work /home/auder/data/EDF/2009.bin to rank=2 / 10004
0 / Send work /home/auder/data/EDF/2009.bin to rank=3 / 15006
2 / Slave pid=4861 work on /home/auder/data/EDF/2009.bin
3 / Slave pid=4862 work on /home/auder/data/EDF/2009.bin
0 / Send work /home/auder/data/EDF/2009.bin to rank=4 / 20008
4 / Slave pid=4863 work on /home/auder/data/EDF/2009.bin
0 / Send work /home/auder/data/EDF/2009.bin to rank=5 / 25011
5 / Slave pid=4864 work on /home/auder/data/EDF/2009.bin
0 / Receive result from rank=4 on /home/auder/data/EDF/2009.bin
0 / Receive result from rank=3 on /home/auder/data/EDF/2009.bin
0 / Receive result from rank=2 on /home/auder/data/EDF/2009.bin
0 / Receive result from rank=5 on /home/auder/data/EDF/2009.bin
0 / Receive result from rank=1 on /home/auder/data/EDF/2009.bin
0 / Send final work .tmp/8040267 to rank=6 / 50
6 / Slave pid=4865 work on .tmp/8040267
0 / Receive final result from rank=6 on .tmp/8040267
auder ~/code/build $ █

auder@wildrose:~/code/build - LilyTerm
auder ~/code/build $ cat ppamResult.xml
<medoids>

<file>ppamFinalSeries.bin</file>

<p_for_dissims>2</p_for_dissims>

<IDs>
  <ID>18</ID>
  <ID>18</ID>
  <ID>18</ID>
  <ID>19</ID>
  <ID>18</ID>
  <ID>18</ID>
  <ID>19</ID>
  <ID>19</ID>
  <ID>18</ID>
  <ID>19</ID>
</IDs>

<ranks>
  <rank>2</rank>
  <rank>29</rank>
  <rank>50</rank>
  <rank>12</rank>
  <rank>41</rank>
  <rank>36</rank>
  <rank>28</rank>
  <rank>48</rank>
  <rank>39</rank>
  <rank>4</rank>
</ranks>
</medoids>
```

Application I : Electricity Smart Meter CBT

- 4621 Irish households smart meter data (ISSDA)
- About 25K discretization points
- We test with $K = 3$ or 5 classes
- We compare sequential and parallel versions

	Distortion	(Internal) adequacy
3 clusters sequential	1.90e7	0.90
3 clusters parallel	2.15e7	0.90
5 clusters sequential	1.61e7	0.89
5 clusters parallel	1.84e7	0.89

Adequacy : given $P_1 = (i_1, \dots, i_n)$ and $P_2 = (j_1, \dots, j_n)$,
find a matching which maximize $S = \sum_{k=1}^n 1_{i_k=j_k}$
(hungarian algorithm), and then return S/n .

Application II : Starlight curves

- Data from **UCR Time Series Classification/Clustering**
- 1000 curves learning set + 8236 validation set ($d = 1024$)

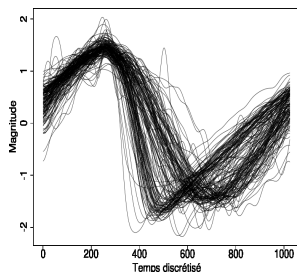


FIGURE: Groupe 1

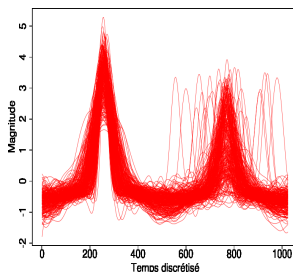


FIGURE: Groupe 2

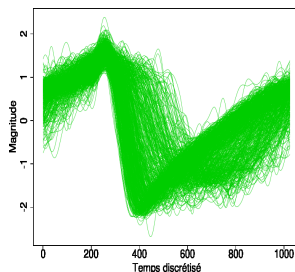


FIGURE: Groupe 3

		Adequacy	
	Distortion	Internal	External
Training (sequential)	1.31e4	0.79	0.77
Training (parallel)	1.40e4	0.79	0.68
Test (sequential)	1.09e5	0.78	0.76
Test (parallel)	1.15e5	0.78	0.69

Conclusion





Résumé

- Les smartmètres mesurent la charge électrique pour chaque client, en temps réel \Rightarrow données fonctionnelles.
- Les ondelettes fournissent des représentations parcimonieuses tout en préservant la nature fonctionnelle des données.
- L'analyse de ces représentations à l'aide de l'algorithme PAM permet d'identifier des groupes de clients.
- L'algorithme PAM est appliqué en parallèle sur des jeux de données de tailles raisonnables.

Perspectives

- L'étude des groupes de clients peut donner lieu à l'élaboration de K modèles prédictifs spécialisés.
- La méthode de clustering parallèle proposée peut être adaptée pour traiter les 35M séries (sur un supercalculateur?).

Références

-  A. Antoniadis, X. Brossat, J. Cugliari, J.-M. Poggi (2013), *Clustering Functional Data Using Wavelets*, *Wavelets, Multiresolution and Information Processing*, 11(1), 35–64
-  A. Arbelaez, L. Quesada (2013), *Parallelising the k-Medoids Clustering Problem Using Space-Partitioning*, *Symposium on Combinatorial Search*, AAAI Publications
-  R. Bekkerman, M. Bilenko, J. Langford - éditeurs (2011), *Scaling up Machine Learning : Parallel and Distributed Approaches*, Cambridge University Press
-  A. P. Reynolds, G. Richards, B. de la Iglesia, V. J. Rayward-Smith (2006), *Clustering Rules : A Comparison of Partitioning and Hierarchical Clustering Algorithms*, *Mathematical Modelling and Algorithms*, 5(4), 475–504

