A clusterwise supervised learning procedure based on aggregation of distances

Sothea Has, Aurélie Fisher, Mathilde Mougeot

A clusterwise supervised learning procedure based on aggregation of distances

S. Has[1], A. Fischer[1] & M. Mougeot[1,2]

[1]LPSM, Université de Paris, France
[2]CMLA, Ecole Normale Supérieure de Cachan & ENSIIE, France

aurelie.fischer@lpsm.paris
sothea.has@lpsm.paris
mathilde.mougeot@ensiie.fr

Abstract

Nowadays, many machine learning procedures are available on the shelve and may be used easily to calibrate predictive models on supervised data. However, when the input data consists of more than one unknown cluster, and when different underlying predictive models exist, fitting a model is a more challenging task. We propose, in this paper, a procedure in three steps to automatically solve this problem. The KFC procedure aggregates different models adaptively on data. The first step of the procedure aims at catching the clustering structure of the input data, which may be characterized by several statistical distributions. It provides several partitions, given the assumptions on the distributions. For each partition, the second step fits a specific predictive model based on the data in each cluster. The overall model is computed by a consensual aggregation of the models corresponding to the different partitions. A comparison of the performances on different simulated and real data assesses the excellent performance of our method in a large variety of prediction problems.

Keywords: Clustering, Bregman divergences, Aggregation, Classification, Regression, Kernel.

2010 Mathematics Subject Classification: 62J99; 62P30; 68T05; 68U99

1 Introduction

Machine learning tools and especially predictive models are today involved in a large variety of applications for the automated decision-making process

such as face recognition, anomaly detection... The final performance of a supervised learning model depends, of course, not only on the choice of the model but also on the quality of the dataset used to estimate the parameters of the model. It is difficult to build an accurate model when some information is missing: the frequent expression "garbage in, garbage out (GIGO)" highlights that nonsense or incomplete input data produces nonsense output.

For some reasons, several fields useful for processing or understanding data may be missing. For instance, in hiring processes, the use of information about individuals, such as gender, ethnicity, place of residence, is not allowed for ethic reasons and to avoid discrimination. Similarly, when high school students apply for further studies in higher education, not every information can be considered for selection. Besides, the General Data Protection Regulation (GDPR) text regulates data processing in the European Union since May 2018. It strengthens the French Data Protection Act, establishing rules on the collection and use of data on French territory Tikkinen-Piri et al. (2018). As a result, contextual data that could characterize individuals a little too precisely is often missing in available databases. Moreover, in an industrial context, not all recorded fields are made available for data processing for confidentiality reasons. For example, in the automotive industry, GPS data could be a valuable tool to provide services such as predictive vehicle maintenance. However, it is difficult to use such data as they are extremely sensitive. To sum up, in various areas, databases containing individual information have to respect anonymization rules before being analyzed.

Mining such databases can then be a particularly complex task as some critical fields are missing. In this context, the modalities of a missing qualitative variable correspond to several underlying groups of observations, which are a priori unknown but should be meaningful for designing a predictive model. In this case, the most common approach consists of using a two-step procedure: the clusters are computed in the first step and, in the second step, a predictive model is fit for each cluster. This two-step procedure has already been used to approximate time evolution curves in the context of nuclear industry by Auder and Fischer (2012), to forecast electricity consumption using high-dimensional regression mixture models by Devijver et al. (2015), or to cluster multi blocks before PLS regression by Keita et al. (2015). In a two-step procedure, the final performance of the model strongly depends on the first step. Different configurations of clusters may bring various performances, and finding an appropriate configuration of clusters is not an easy

task which often requires a deep data investigation and/or human expertise.

To build accurate predictive models in situations where the contextual data are missing, and to eliminate an unfortunate choice of clusters, we propose, in this work, to aggregate several instances of the two-step procedures where each instance corresponds to a particular clustering. Our strategy is characterized by three steps, each is based on a quite simple procedure. The first step aims to cluster the input data into several groups and is based on the well-known K-means algorithm. As the underlying group structures are unknown and may be complex, a given Bregman divergence is used as a distortion measure in the K-means algorithm. In the second step, for each divergence, a very simple predictive model is fit per cluster. The final step provides an adaptive global predictive model by aggregating, thanks to a consensus idea introduced by Mojirsheibani (1999), several models built for the different instances, corresponding to the different Bregman divergences (see also Mojirsheibani (2000); Balakrishnan and Mojirsheibani (2015); Biau et al. (2016); Fischer and Mougeot (2019)). We name this procedure the *KFC* procedure for K-means/Fit/Consensus.

This paper is organized as follows. In Section 2, we recall some general definitions and notations about supervised learning. Section 3 is dedicated to Bregman divergences, their relationship with probability distributions of the exponential family, and K-means clustering with Bregman divergences. Section 4 presents the consensual aggregation methods considered, in classification and regression. The KFC procedure is detailed in Section 5. Finally, Sections 6 and 7 present several numerical results carried out on simulated and real data, showing the performance and the relevance of using our method. We also study the robustness of the procedure with respect to the number K of clusters.

2 Definitions and notations

We consider a general framework of supervised learning problems where the goal is to construct a predictive model using input data to predict the value of a variable of interest, also called response variable or output. Let (X, Y) denote a random vector taking its values in $\mathbb{R}^d \times \mathcal{Y}$, where the output space $\mathcal{Y}$ is either $\{0, 1\}$ (binary classification) or $\mathbb{R}$ (regression). Constructing a predictive model is finding a mapping $g : \mathcal{X} \to \mathcal{Y}$ such that the image

$g(X)$ is "close" in some sense to the corresponding output Y. The space $(\mathbb{R}^d, \|\cdot\|)$ is equipped with the standard Euclidean metric. Let $\langle\cdot,\cdot\rangle$ denotes the associated standard inner product. Throughout, we take the convention $0/0 = 0$.

In classification problems, the performance of a predictor or classifier g is usually measured using the misclassification error

$$\mathcal{R}_C(g) = \mathbb{P}(g(X) \neq Y).$$

Similarly, the performance of a regression estimator g is measured using the quadratic risk

$$\mathcal{R}_R(g) = \mathbb{E}\Big[\Big(g(X) - Y\Big)^2\Big].$$

In the sequel, $\mathcal{R}(g)$ describes the risk of a predictor g without specifying the classification or regression case. A predictor g^* is called optimal if

$$\mathcal{R}(g^*) = \inf_{g\in\mathcal{G}} \mathcal{R}(g)$$

where $\mathcal{G}$ is the class of all predictors $g : \mathbb{R}^d \to \mathcal{Y}$. In regression, the optimal predictor is the regression function defined by $\eta(x) = \mathbb{P}(Y = 1|X = x)$, whereas in binary classification the minimum is achieved by the Bayes classifier, given by

$$g_B(x) = \begin{cases} 1 & \text{if } \eta(x) > 1/2 \\ 0 & \text{otherwise.} \end{cases}$$

Note that η and, hence g_B, depend on the unknown distribution of (X, Y).

In a statistical learning context, we observe independent and identically distributed random pairs $\{(X_1, Y_1), (X_2, Y_2), ..., (X_n, Y_n)\}$ distributed as (X, Y). The goal is to estimate the regression function η, or mimic the classifier g_B, based on the sample D_n.

We consider, in this work, situations where the input data D_n may consist of several clusters and where there exist different underlying regression or classification models on these clusters.

3 Bregman divergences and K-means clustering

Among all unsupervised learning methods, a well-known and widely used algorithm is the seminal K-means algorithm, based on the Euclidean distance, see for example Steinhaus (1956), Lloyd (1982), Linder (2001) or Jain (2010). This algorithm may be extended to other distortion measures, namely the class of Bregman divergences, Banerjee et al. (2005b).

3.1 Bregman Divergences

Let $\phi : \mathcal{C} \to \mathbb{R}$ be a strictly convex and continuously differentiable function defined on a measurable convex subset $\mathcal{C} \subset \mathbb{R}^d$. Let $int(\mathcal{C})$ denote its relative interior. A Bregman divergence indexed by ϕ is a dissimilarity measure $d_\phi : \mathcal{C} \times int(\mathcal{C}) \to \mathbb{R}$ defined for any pair $(x, y) \in \mathcal{C} \times int(\mathcal{C})$ by,

$$d_\phi(x, y) = \phi(x) - \phi(y) - \langle x - y, \nabla\phi(y)\rangle \tag{1}$$

where $\nabla\phi(y)$ denotes the gradient of ϕ computed at a point $y \in int(\mathcal{C})$. A Bregman divergence is not necessarily a metric as it may not be symmetric and the triangular inequality might not be satisfied. However, it carries many interesting properties such as non-negativity, separability, convexity in the first argument, linearity in the indexed function, and the most important one is mean as minimizer property by Banerjee et al. (2005a).

Proposition 1 (Banerjee et al. (2005a)) *Suppose U is a random variable over an open subset $\mathcal{O} \subset \mathbb{R}^d$, then we have,*

$$\mathbb{E}[U] = \underset{x \in \mathcal{O}}{\operatorname{argmin}}\, \mathbb{E}[d_\phi(U, x)].$$

In this article, we will consider four Bregman divergences, presented in Table 1: Squared Euclidean distance (Euclid), General Kullback-Leibler (GKL), Logistic (Logit) and Itakura-Saito (Ita) divergences.

3.2 Bregman Divergences and Exponential family

An exponential family is a class of probability distributions enclosing, for instance, Geometric, Poisson, Multinomial distributions, for the discrete case,

BD	ϕ	d_ϕ	$\mathcal{C}$
Euclid	$\|x\|_2^2 = \sum_{i=1}^d x_i^2$	$\|x-y\|_2^2 = \sum_{i=1}^d (x_i - y_i)^2$	$\mathbb{R}^d$
GKL	$\sum_{i=1}^d x_i \ln(x_i)$	$\sum_{i=1}^d \left[x_i \ln(\frac{x_i}{y_i}) - (x_i - y_i) \right]$	$(0, +\infty)^d$
Logit	$\sum_{i=1}^d \left[x_i \ln(x_i) + (1-x_i)\ln(1-x_i) \right]$	$\sum_{i=1}^d \left[x_i \ln(\frac{x_i}{y_i}) + (1-x_i)\ln(\frac{1-x_i}{1-y_i}) \right]$	$(0, 1)^d$
Ita	$-\sum_{i=1}^d \ln(x_i)$	$\sum_{i=1}^d \left[\frac{x_i}{y_i} - \ln(\frac{x_i}{y_i}) - 1 \right]$	$(0, +\infty)^d$

Table 1: Some examples of Bregman divergences.

Exponential, Gaussian, Gamma distribution, for the continuous case. More formally, an Exponential family $\mathcal{E}_\psi$ is a collection of probability distributions dominated by a σ-finite measure μ with density with respect to μ taking the following form:

$$f_\theta(x) = \exp(\langle \theta, T(x) \rangle - \psi(\theta)), \theta \in \Theta, \tag{2}$$

where $\Theta = \{\theta \in \mathbb{R}^d : \psi(\theta) < +\infty\}$ is the parameter space of natural parameter θ, T is called sufficient statistics and ψ is called log-partition function. The equation (2) is said to be *minimal* if the sufficient statistics T is not redundant, that is, if there does not exist any parameter $\alpha \neq 0$, such that $\langle \alpha, T(x) \rangle$ equals a constant, $\forall x \in \mathbb{R}^d$. If the representation (2) is minimal and the parameter space Θ is open, then the family $\mathcal{E}_\psi$ is said to be *regular*. The relationship between a regular exponential family and Bregman divergence is given in the following theorem.

Theorem 1 (Banerjee et al. (2005b)) *Each member of a regular exponential family corresponds to a unique regular Bregman divergence. If the distribution of a random variable X is a member of a regular Exponential family $\mathcal{E}_\psi$ and if ϕ is the convex conjugate of ψ defined by*

$$\phi(x) = \sup_y \{\langle x, y \rangle - \psi(y)\},$$

then there exists a unique Bregman divergence d_ϕ such that the following representation holds:

$$f_\theta(x) = \exp(\langle \theta, T(x) \rangle - \psi(\theta)) = \exp(-d_\phi(T(x), \mathbb{E}[T(X)]) + \phi(T(x))).$$

Theorem 1 and Proposition 1 together provide a strong motivation for using K-means algorithm with Bregman divergences to cluster any sample distributed from the corresponding member of an exponential family.

We consider a set of n input observations $\{X_i\}_{i=1}^n$ distributed according to a law f_θ, organized in K clusters and d_ϕ is the associated Bregman divergence. Our goal is to find the centroids $\mathbf{c} = (\mathbf{c_1}, \ldots, \mathbf{c_K})$ of the clusters minimizing the function

$$W(f_\theta, \mathbf{c}) = \mathbb{E}\left[\min_{j=1,\ldots,K} d_\phi(X, c_j)\right].$$

We propose the following K-mean clustering algorithm with the Bregman divergence d_ϕ:

Algorithm 1

1. *Randomly initialize the centroids* $\{c_1^{(0)}, c_2^{(0)}, ..., c_K^{(0)}\}$ *among the data points.*
2. *At iteration* r*:* ***For*** $i = 1, 2, ..., n$*, assign* $X_i^{(r)}$ *to* k*-th cluster if*
$$d_\phi(X_i^{(r)}, c_k^{(r)}) = \min_{1 \le j \le K} d_\phi(X_i^{(r)}, c_j^{(r)})$$
3. *Denote by* $C_k^{(r)}$ *the set of points contained in the* k*-th cluster.*
For $k = 1, 2, ..., K$*, recomputes the new centroid by,*
$$c_k^{(r+1)} = \frac{1}{|C_k^{(r)}|} \sum_{x \in C_k^{(r)}} x$$

Repeat step 2 and 3 until a stopping criterion is met.

In practice, it is well-known that the algorithm might get stuck at a local minimum if it begins with a bad initialization. A simple way to overcome this problem is to perform the algorithm several times with different initialization each time and to keep the partition minimizing the empirical distortion. In our version, in the event of ties, they are broken arbitrarily and the associated empirical distortion is defined by

$$\widehat{W}(\mathbf{c}) = \frac{1}{n} \sum_{i=1}^{n} \min_{1 \le k \le K} d_\phi(X_i, c_k).$$

For example:

- Poisson distribution with parameter $\lambda > 0$: $X \sim \mathcal{P}(\lambda)$ has probability mass function: for any $k \in \{0, 1, ...\}, \mathbb{P}(X = k) = e^{-\lambda}\frac{\lambda^k}{k!}$, corresponding to the 1-dimensional General Kullback-Leibler divergence defined by,

$$d_\phi(x, y) = x \ln\left(\frac{x}{y}\right) - (x - y), \forall x, y > 0.$$

- Exponential distribution with parameter $\lambda > 0$: $X \sim \mathcal{E}(\lambda)$ has probability density function: for any $x > 0$, $f_\lambda(x) = \lambda e^{-\lambda x}$, corresponding to the 1-dimensional Itakura-Saito divergence defined by,

$$d_\phi(x, y) = \frac{x}{y} - \ln\left(\frac{x}{y}\right) - 1, \forall x, y > 0.$$

See Banerjee et al. (2005b) for more examples.

4 Consensual aggregation methods

In this section, we describe the aggregation methods, based on a consensus notion, which will be used in the next section to build our global predictive model. The original combination idea was introduced by Mojirsheibani (1999) for classification (see also Mojirsheibani (2000, 2006)) and adapted to the regression case by Biau et al. (2016). We will also consider, in both classification and regression, a modified version of the consensual aggregation method introduced recently by Fischer and Mougeot (2019).

4.1 The original consensual aggregation

Several methods of combining estimates in regression and classification have been already introduced and studied. LeBlanc and Tibshirani (1996) have proposed a procedure of combining estimates based on the linear combination of the estimated class of conditional probabilities, inspired on the "stacked regression" of Breiman (1996). Linear-type aggregation strategies, model selection and related problems have been also studied by Catoni (2004), Nemirovski (2000), Yang (2000), Yang et al. (2004), and Györfi et al. (2006). There are other related works by Wolpert (1992), and Xu et al. (1992).

In this paper, we will use a combining method introduced first in classification by Mojirsheibani (1999), based on an idea of consensus. For a new query point $x \in \mathbb{R}^d$, the purpose is to search for data items X_i, $i \in I$, such that all estimators to be combined predict the same label for X_i and x. The estimated label of x is then obtained by a majority vote among the corresponding labels Y_i, $i \in I$. More formally, for $x \in \mathbb{R}^d$, let $\mathbf{m}(x) = (m^{(1)}(x), ..., m^{(M)}(x))$ denote the vector of the predictions for x given by M estimators. The combined estimator is defined by:

$$Comb_1^C(x) = \begin{cases} 1 & \text{if } \sum_{i=1}^n \mathbf{1}_{\{\mathbf{m}(X_i)=\mathbf{m}(x)\}}\mathbf{1}_{\{Y_i=1\}} > \sum_{i=1}^n \mathbf{1}_{\{\mathbf{m}(X_i)=\mathbf{m}(x)\}}\mathbf{1}_{\{Y_i=0\}} \\ 0 & \text{otherwise.} \end{cases}$$

Under appropriate assumptions, the combined classifier is shown to asymptotically outperform the individual classifiers. It is also possible to allow a few disagreements among the initial estimators.

A regularized version, based on different kernels has been proposed in Mojirsheibani (2000) (see also Mojirsheibani and Kong (2016)). These smoother definitions are also a way not to require unanimity with respect to all the initial estimators, to lighten the effect of a possibly bad estimator in the list.

To simplify the notation, let K be a positive decreasing kernel defined either on $\mathbb{R}_+$ or $\mathbb{R}^d$ to $\mathbb{R}_+$ then the kernel-based combined classifier is defined as follows:

$$Comb_2^C(x) = \begin{cases} 1 & \text{if } \sum_{i=1}^n (2Y_i - 1)K_h\Big(d_{\mathcal{H}}(\mathbf{m}(X_i), \mathbf{m}(x))\Big) > 0 \\ 0 & \text{otherwise,} \end{cases}$$

where $d_{\mathcal{H}}$ stands for the Hamming distance (the number of disagreements between the components of $\mathbf{m}(X_i)$ and $\mathbf{m}(x)$), and $K_h(x) = K(x/h)$. We will consider the following kernels:

1. Gaussian kernel: for a given $\sigma > 0$ and for all $x \in \mathbb{R}^d$,

$$K(x) = e^{-\frac{\|x\|_2^2}{2\sigma^2}}.$$

2. Triangular kernel: for all $x \in \mathbb{R}^d$,

$$K(x) = (1 - \|x\|_1)\mathbb{1}_{\{\|x\|_1 \leq 1\}}.$$

where $\|.\|_1$ is the ℓ_1-norm and is defined by: $\|x\|_1 = \sum_{i=1}^d |X_i|$

3. Epanechnikov kernel: for all $x \in \mathbb{R}^d$,

$$K(x) = (1 - \|x\|_2^2)\mathbb{1}_{\{\|x\|_2 \leq 1\}}.$$

where $\|.\|_2$ is the ℓ_2-norm and is defined by: $\|x\|_2 = \left(\sum_{i=1}^d X_i^2\right)^{1/2}$

4. Bi-weight kernel: for all $x \in \mathbb{R}^d$,

$$K(x) = (1 - \|x\|_2^2)^2\mathbb{1}_{\{\|x\|_2 \leq 1\}}.$$

5. Tri-weight kernel: for all $x \in \mathbb{R}^d$,

$$K(x) = (1 - \|x\|_2^2)^3\mathbb{1}_{\{\|x\|_2 \leq 1\}}.$$

These kernels are plotted in dimension 1 in Figure 1, together with the uniform kernel corresponding to $Comb_1^C$.

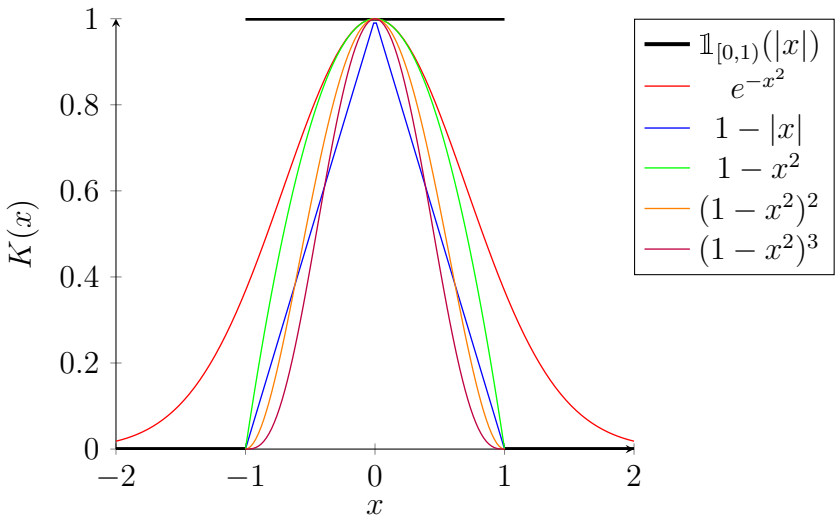

Figure 1: The shapes of all kernels.

In the regression case, mimicking the rule introduced in classification, the predictions will be required to be close to each other, in the sense of some

threshold, with the predicted value obtained as a weighted average of the outputs of the selected data. The combined regression estimator, proposed in Biau et al. (2016) is given, for $x \in \mathbb{R}^d$, by

$$Comb_1^R(x) = \frac{1}{n}\sum_{i=1}^{n} W_{n,i}(x)Y_i, \quad W_{n,i}(x) = \frac{\prod_{\ell=1}^{M} \mathbf{1}_{\{|m^{(\ell)}(X_i)-m^{(\ell)}(x)|<\varepsilon\}}}{\sum_{j=1}^{n}\prod_{\ell=1}^{M} \mathbf{1}_{\{|m^{(\ell)}(X_j)-m^{(\ell)}(x)|<\varepsilon\}}}.$$

Once again, unanimity may be relaxed, for instance, if the distance condition is only required to be satisfied by a fraction α of the individual estimators:

$$W_{n,i}(x) = \frac{\mathbf{1}_{\left\{\sum_{\ell=1}^{M} \mathbf{1}_{\{|m^{(\ell)}(X_i)-m^{(\ell)}(x)|<\varepsilon\}} \geq M\alpha\right\}}}{\sum_{j=1}^{n} \mathbf{1}_{\left\{\sum_{\ell=1}^{M} \mathbf{1}_{\{|m^{(\ell)}(X_j)-m^{(\ell)}(x)|<\varepsilon\}} \geq M\alpha\right\}}}.$$

It is shown that, when $\alpha \to 1$, the combined estimator asymptotically outperforms the different individual estimators. Here, we propose also to use a kernel version $Comb_2^R$, by setting:

$$W_{n,i}(x) = \frac{K_h(\mathbf{m}(X_i) - \mathbf{m}(x))}{\sum_{j=1}^{n} K_h(\mathbf{m}(X_j) - \mathbf{m}(x))}.$$

4.2 Consensual aggregation combined to input distance

An alternative definition of combined estimator suggests mixing the consensus idea with information about distances between inputs (Fischer and Mougeot (2019)). This is a way to limit the influence, if any, of a bad estimator; using at the same time information on the geometry of the inputs. In regression, the estimator is defined, for $x \in \mathbb{R}^d$, by

$$Comb_3^R(x) = \frac{1}{n}\sum_{i=1}^{n} W_{n,i}(x)Y_i, \quad W_{n,i}(x) = \frac{K\left(\frac{X_i-x}{\alpha}, \frac{\mathbf{m}(X_i)-\mathbf{m}(x)}{\beta}\right)}{\sum_{j=1}^{n} K\left(\frac{X_j-x}{\alpha}, \frac{\mathbf{m}(X_j)-\mathbf{m}(x)}{\beta}\right)}.$$

In classification, by plug-in, we set

$$Comb_3^C(x) = \begin{cases} 1, & \text{if } \displaystyle\sum_{i=1}^{n}(2Y_i-1)K\left(\frac{X_i - X}{\alpha}, \frac{\mathbf{m}(X_i) - \mathbf{m}(x)}{\beta}\right) > 0 \\ 0 & \text{otherwise.} \end{cases}$$

5 The KFC procedure

We recall hereafter the three steps of the KFC strategy and specify the parameters chosen at each step.

1. *K-means.* The input data X are first clustered using the K-means clustering algorithm with a chosen Bregman divergence. The choice of the number K of clusters is discussed in the next Section where the numerical results on several examples are presented. In this work, $M = 4$ divergences are considered: Squared Euclidean distance (Euclid), General Kullback-Leibler (GKL), Logistic (Logit) and Itakura-Saito (Ita) divergences, as already defined in Section 3.

2. *Fit.* For each Bregman divergence m and for each cluster k, a dedicated predictive model, $\mathcal{M}_{m,k}$, is fit using the available observations, $1 \leq m \leq M$ and , $1 \leq k \leq K$.

   In the numerical applications, we simply choose for regression models linear regression, whereas for the classification models, we choose logistic regression. Much more complex models can be of course considered, but one of the main ideas of this paper, based on our modeling experience gained over several real-life projects, is that if the initial data are initially clustered «in an appropriate way» then the fit of the target variable can often be successfully computed with quite simple models in each group.

3. *Consensus.* As neither the distribution nor the clustering structure of the input data is known, it is not clear in advance which divergence will be the most efficient. Thus, we propose to combine all the previous estimators, in order to take the best advantage of the clustering step. For the combination task, we use the different consensus-based procedures already described. Practically, the different kernel bandwidths appearing in the combining methods are optimized on a grid, using cross-validation.

Once the candidate model, which is the collection of all the local models constructed on the corresponding clusters, is fitted, in order to make a prediction for a new observation x, we first affect x to the closest cluster for each divergence, which yields one prediction per divergence, and then, performs the aggregation. The procedure is illustrated in Figure 2 below.

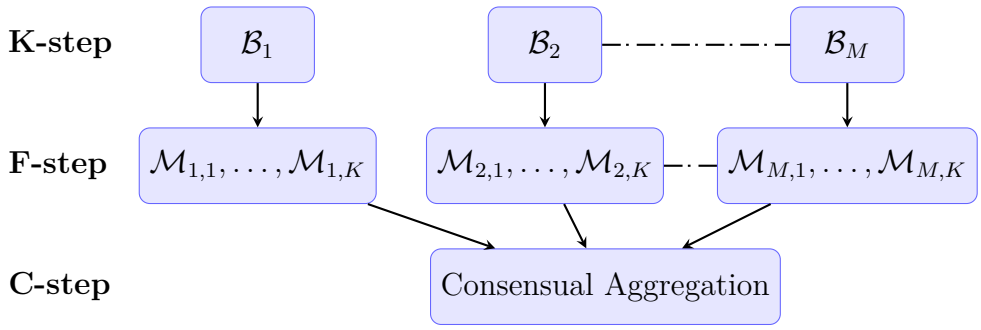

Figure 2: The main steps of the model construction: for each Bregman divergence $\mathcal{B}_m$, one model $\mathcal{M}_{m,k}$ is fit per cluster k, then the models corresponding to the different divergences are combined.

6 Simulated data

In this section, we analyze the behavior of the strategy on several simulated datasets in both classification and regression problems.

6.1 Description

In both cases of classification and regression problems, we simulate 5 different kinds of datasets. We consider 2-dimensional datasets where the two predictors (X_1, X_2) are simulated according to Exponential, Poisson, Geometric and Gaussian distribution respectively. The remaining dataset is 3-dimensional, with predictors (X_1, X_2, X_3), distributed according to Gaussian distribution. Each simulated training and testing dataset contains respectively 1500 and 450 data points. Each dataset consists of $K = 3$ balanced clusters; each cluster contains 500 observations for training and 150 for testing. Note that this choice of $K = 3$ clusters is to illustrate the procedure and performance of our algorithm. Various complementary studies with different number of clusters showed that similar results held.

The different distribution parameters used in the simulations are listed in Table 2. Each cell of the table contains the parameters of each distribution at the corresponding cluster for the input variables (X_1, X_2) or (X_1, X_2, X_3).

For the regression cases, the target observation Y_i belonging to cluster k, is computed by $Y_i^k = \beta_0^k + \sum \beta_j^k X_i^k + \epsilon_i$ where $X_i^k = (X_{i,j}^k)_{j=1,\ldots,d}$ is the

Distribution	**Cluster 1**	**Cluster 2**	**Cluster 3**
Exponential: λ	$0.05; 0.5$	$0.5; 0.05$	$0.1; 0.1$
Poisson: λ	$3; 11$	$10; 2$	$13; 12$
Geometric: p	$0.07; 0.35$	$0.55; 0.07$	$0.15; 0.15$
2D Normal: $\begin{cases}\mu\\ \sigma\end{cases}$	$\begin{cases}4; 12\\ 1; 1\end{cases}$	$\begin{cases}22; 9\\ 2; 1\end{cases}$	$\begin{cases}10; 5\\ 2; 2\end{cases}$
3D Normal $\begin{cases}\mu\\ \sigma\end{cases}$	$\begin{cases}6; 14; 6\\ 1; 2; 1\end{cases}$	$\begin{cases}5; 10; 15\\ 2; 1; 2\end{cases}$	$\begin{cases}8; 6; 14\\ 1; 1; 2\end{cases}$

Table 2: Parameters of the simulated data.

input observation of dimension d, $\beta^k = (\beta_j^k)_{j=1,\dots,d}$ the parameters of cluster k, $1 \leq k \leq K$, $d = 2$ or $d = 3$ and $\epsilon_i \sim \mathcal{N}(0, 10)$.

For classification cases, the target observation belonging to cluster k, is computed by $Y_i^k = 0$ if $\frac{1-e^{\beta_0^k + \sum \beta_j^k X_i^k + \epsilon_i}}{1+e^{\beta_0^k + \sum \beta_j^k X_i^k + \epsilon_i}} \leq 0$ and $\epsilon_i \sim \mathcal{N}(0, 10)$.

	Cluster 1	**Cluster 2**	**Cluster 3**
	$(k = 1)$	$(k = 2)$	$(k = 3)$
2D (β_1^k, β_2^k)	$(-8, 3)$	$(-6, -5)$	$(5, -7)$
3D $(\beta_1^k, \beta_2^k, \beta_3^k)$	$(-10, 3, 7)$	$(7, 5, -12)$	$(6, -11, 10)$

Table 3: The coefficients of the simulated models.

In regression problems, we choose the intercepts $(\beta_0^1, \beta_0^2, \beta_0^3) = (-15, 25, -10)$ for the 3 clusters. For classification, we study cases where each cluster has the same number of observations form the 2 labels. In order to balance the positive and negative points in classification cases, we choose intercepts so that the hyperplane defined by the input data within each cluster is centered at zero. Therefore, after applying the sigmoid transformation, we would have a balance between the two classes within each cluster. This can be done as follows.

- Compute α_j^k: the conditional average of the j-th input variable falling

into the k-th cluster which is defined by

$$\alpha_j^k = \frac{1}{|C_j^k|} \sum_{x \in C_j^k} x$$

where $C_j^k \subset X_j$ is the subset of the j-th input variable that are contained in the k-th cluster.

- The intercept of the k-th cluster for $k \in \{1, 2, 3\}$ is given by,

$$\beta_0^k = -\langle \beta^k, \alpha^k \rangle = \sum_{j=1}^{d} \alpha_j^k \beta_j^k, \text{ for } d = 2 \text{ or } d = 3$$

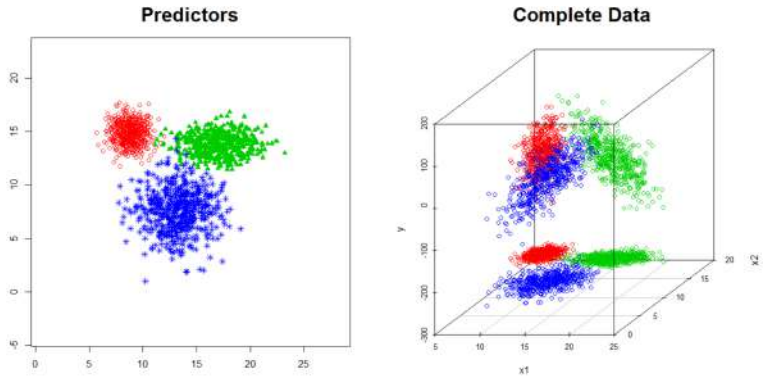

Figure 3: An example of simulated data in regression problem with Gaussian predictors.

Remark 1 *Note that in our simulations, the simulated samples might fall outside the domain $\mathcal{C}$ for some Bregman divergences for instance, the logistic one which can handle only data points in $(0, 1)^d$. In practice, we can solve this problem by normalizing our original samples using the ℓ_1-norm $\|.\|_1$, i.e., $X_i \to \tilde{X}_i = X_i / \|X_i\|_1$. Moreover, we ignored those negative data points or added a suitable constant in order to avoid negativity.*

Each performance is computed over 20 replications of the corresponding dataset.

6.2 Normalized Mutual Information

Before analyzing the performances of our combined estimators, it is interesting to take a look at the performances of the clustering algorithm itself with different Bregman divergences. Even though this is not possible in practice, the clustering structure is here available in our simulations. We use a correlation coefficient between partitions proposed by Strehl and Ghosh (2002) known as Normalized Mutual Information (NMI). Let $S = \{S_j\}_{j=1}^K$ and $S' = \{S'_\ell\}_{\ell=1}^K$ be two partitions of n-point observations. Let n_j, n'_ℓ and $n_{j,\ell}$ denote the number of observations in $S_j \in S, S'_\ell \in S'$ and $S_j \cap S'_\ell$ respectively. Then, the NMI of the two partitions S and S' is given by

$$\rho(S, S') = \frac{\sum_{j=1}^K \sum_{\ell=1}^K n_{j,\ell} \log\left(\frac{n.n_{j,\ell}}{n_j n'_\ell}\right)}{\sqrt{\left(\sum_{j=1}^K n_j \log\left(\frac{n_j}{n}\right)\right)\left(\sum_{\ell=1}^K n'_\ell \log\left(\frac{n'_\ell}{n}\right)\right)}}.$$

This criterion allows us to compare the observed partition given by the clustering algorithm to the expected (true) one. We have $0 \leq \rho(S, S') \leq 1$ for any partitions S and S'. The closer coefficient to 1, the better the result of the clustering algorithm.

Distributions	Euclidean	GKL	Logistic	Itakura-Saito
Exponential	17.77 (1.53)	24.79 (2.26)	60.42 (1.35)	**76.61** (1.82)
Poisson	88.26 (1.16)	**92.24** (1.41)	68.19 (1.47)	83.53 (9.85)
Geometric	53.61 (1.86)	86.06 (10.04)	**87.31** (0.82)	81.16 (1.56)
2D Normal	**97.89** (0.89)	97.46 (0.99)	69.56 (1.41)	94.81 (1.29)
3D Normal	**91.55** (1.31)	91.19 (1.22)	89.22 (1.57)	89.95 (1.66)

Table 4: Average Normalized Mutual Information (1 unit = 10^{-2}).

Table 4 above contains the average NMI over 20 runs of K-means clustering algorithm performed on each simulated dataset. The associated standard

deviations are provided in brackets. The out-performance of each case is highlighted in blue. Note that the results in the Table 4 recover the expected relation between distributions and Bregman divergences as discussed in Section 3.2. Figure 4 illustrates the computed partitions for one run simulation using K-means algorithm with Bregman divergences.

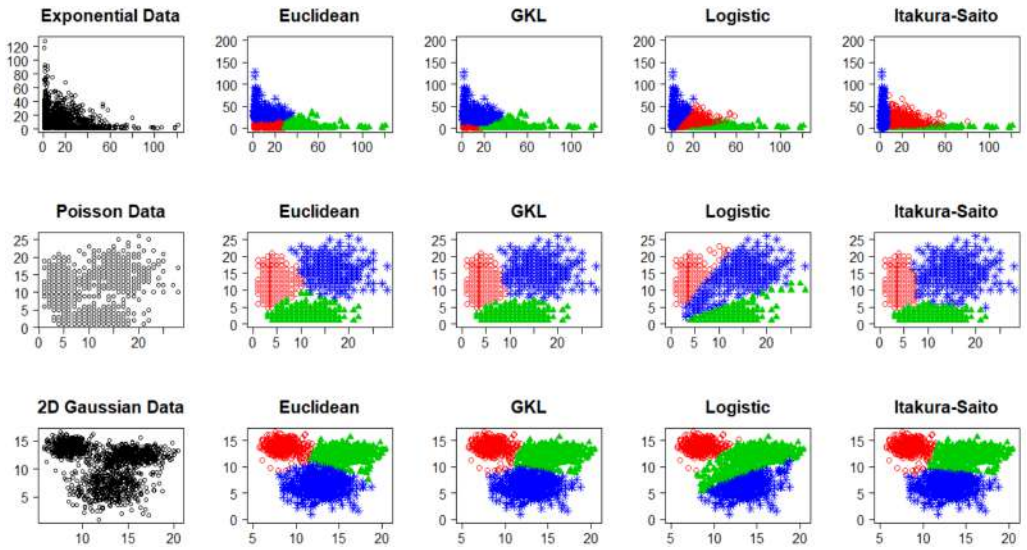

Figure 4: Partitions obtained via K-means with Bregman divergences.

6.3 Numerical results

This section analyzes the ability of the KFC procedure for classification or regression on the five simulated examples described in section 6. Each example is simulated 20 times. For each run, the error obtained using the KFC procedure is computed on the Test dataset; the classification error is evaluated using the misclassification rate and the regression error the Root Mean Square Error. The average and the standard deviation (in bracket) of the errors computed over the 20 runs are provided in the result tables. In order to compare the benefit of the consensual aggregation of KFC procedure, we evaluate the performance of the model on the test data in different situations. First, without any preliminary clustering (i.e. considering only one cluster), the corresponding errors are reported in the column block named "Single" in the different graphs or tables. Second, considering a preliminary

clustering using one given divergence. In this case, the corresponding errors are reported in the column block named "Bregman divergence" in different tables. The four columns named Euclid, GKL, Logistic and Ita contain the results of the 4 individual estimators corresponding to the 4 chosen Bregman divergences. Last, the errors computed with the KFC procedure are presented with several kernels in the block named "Kernel" which consists of six columns named Unif, Epan, Gaus, Triang, Bi-wgt and Tri-wgt standing for Uniform, Epanechnikov, Gaussian, Triangular, Bi-weight and Tri-weight kernel (procedures $Comb_1, Comb_2$). The KFC procedure is also evaluated taking into account the inputs ($Comb_3$), and the corresponding results are provided in the second row of each distribution.

For each table, the first column of each row mentions the names of the simulated datasets where Exp, Pois, Geom, 2D Gauss, and 3D Gauss stand for Exponential, Poisson, Geometric, 2-dimensional and 3-dimensional Gaussian datasets respectively.

For each distribution, we highlight the out-performance of the individual estimators in bold font and the two kinds of combining methods in boldfaced blue $Comb_1, Comb_2$) and red ($Comb_3$) respectively. In each simulation, we consider 300 values of smoothing parameter h or ε on the grid $\{10^{-300}, ..., 5\}$ for $Comb_1$ and $Comb_2$, and consider 50×50 values of parameters $(\alpha, \beta) \in \{10^{-300}, ..., 10\}^2$ for $Comb_3$.

6.3.1 Classification

Table 5 below contains the results of misclassification errors computed on the different kinds of simulated datasets. We observe that the results of all individual estimators in the second block seem to agree with the results of NMI provided in Table 4. Of course, all models built after a clustering step outperform the simple model of the first block. The combined classification methods perform generally better than or similarly to the best individual estimator. The results of $Comb_3^C$, in the second row, seem to be better compared to the ones of $Comb_2^C$, in the first row, with remarkably smaller variances. We also note that the Gaussian kernel seems to be the most outstanding one among all kernel-based methods. Figure 5 and Figure 6 represent the boxplots of the associated average misclassification errors for $Comb_2^C$ and $Comb_3^C$ respectively (the results of the Table 5)

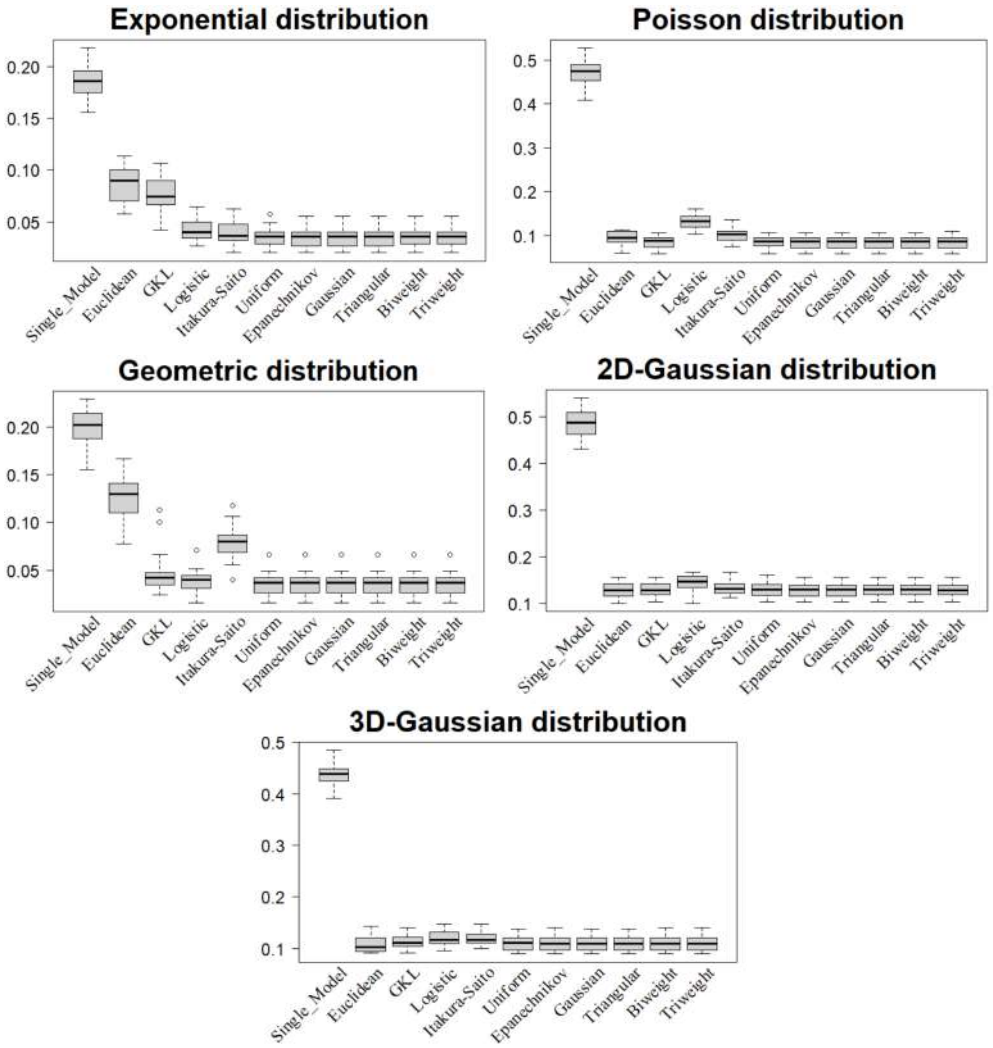

Figure 5: Boxplots of misclassification error of $Comb_2^C$.

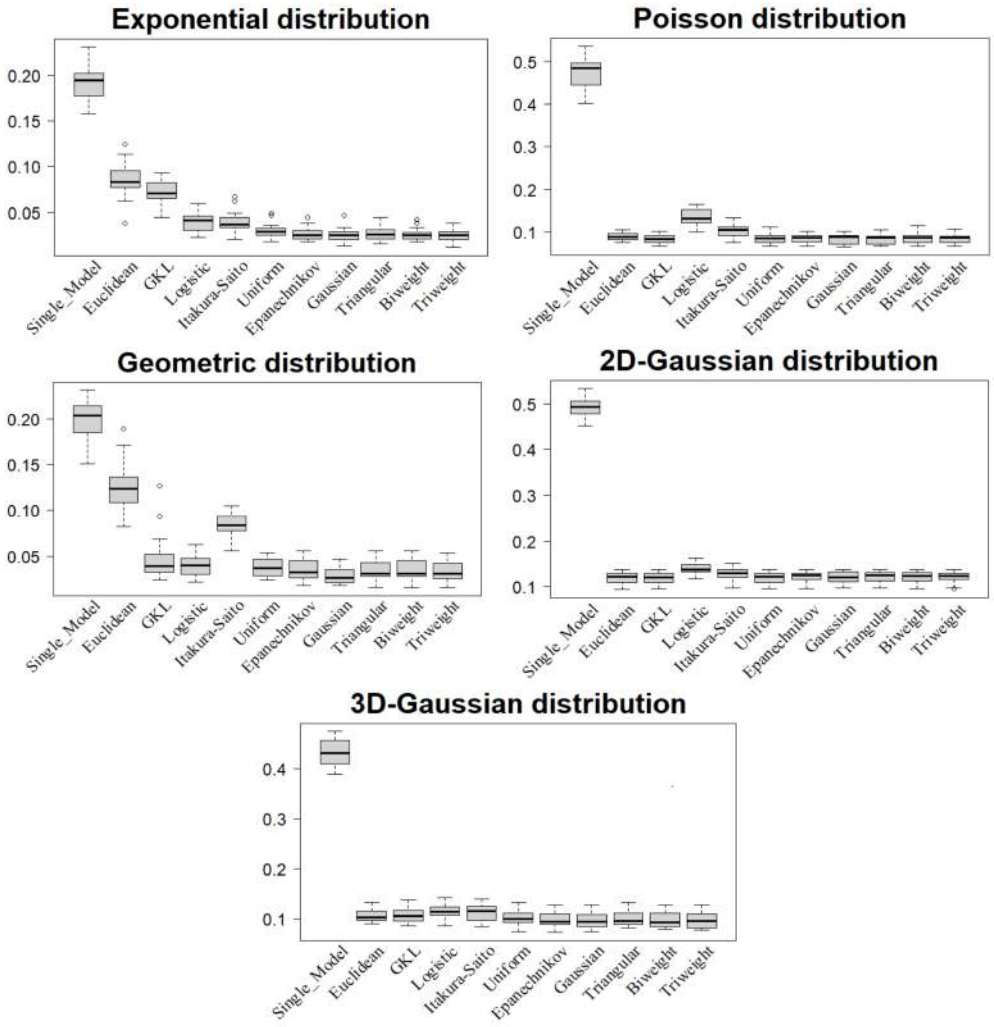

Figure 6: Boxplots of misclassification error of $Comb_3^C$.

Distribution	**Single**	**Bregman divergence**				**Kernel**					
		Euclid	GKL	Logit	Ita	Unif	Epan	Gaus	Triang	Bi-wgt	Tri-wgt
Exp	18.86 (1.70)	8.58 (1.77)	7.42 (1.55)	4.09 (1.08)	**3.92** (1.15)	3.49 (0.89)	3.51 (0.94)	**3.46** (0.88)	3.51 (0.94)	3.56 (0.91)	3.56 (0.91)
						2.91 (0.81)	2.63 (0.70)	2.49 (0.74)	2.70 (0.75)	2.56 (0.63)	**2.46** (0.66)
Pois	46.93 (3.35)	9.19 (1.27)	**8.45** (1.24)	13.33 (1.84)	10.15 (1.47)	8.59 (1.37)	**8.51** (1.46)	**8.51** (1.47)	**8.51** (1.46)	8.52 (1.47)	8.52 (1.49)
						8.51 (1.28)	8.46 (1.11)	8.44 (1.17)	**8.42** (1.15)	8.57 (1.28)	8.44 (1.13)
Geom	19.90 (2.07)	12.57 (2.39)	4.71 (2.37)	**3.94** (1.15)	8.12 (1.57)	3.61 (1.15)	**3.60** (1.16)	**3.60** (1.16)	3.61 (1.15)	**3.60** (1.16)	**3.60** (1.16)
						3.76 (0.92)	3.52 (1.11)	**2.94** (0.93)	3.48 (1.09)	3.47 (1.11)	3.40 (1.06)
2D Gaus	49.00 (2.52)	**12.37** (1.55)	12.40 (1.50)	14.14 (1.44)	13.05 (1.61)	12.87 (1.60)	12.82 (1.59)	**12.80** (1.56)	12.84 (1.57)	12.84 (1.57)	12.87 (1.60)
						12.02 (1.30)	12.11 (1.24)	12.06 (1.35)	12.11 (1.27)	12.09 (1.23)	12.10 (1.22)
3D Gaus	43.39 (2.52)	**10.77** (1.40)	10.99 (1.44)	11.74 (1.45)	11.56 (1.51)	11.08 (1.58)	11.01 (1.52)	**11.00** (1.50)	**11.00** (1.50)	11.04 (1.57)	11.03 (1.55)
						10.23 (1.40)	9.93 (1.47)	**9.76** (1.53)	10.04 (1.47)	9.83 (1.61)	9.84 (1.61)

Table 5: Misclassification errors of $Comb_2^C$ and $Comb_3^C$ computed over 20 runs of all simulated data (1 unit $= 10^{-2}$).

6.3.2 Regression

In the regression case, the results in the Table 6 again agree with the NMI results given in Table 4, except for Geometric distribution, where the estimator based on Generalized Kullback-Leibler Divergence outperforms the estimator built after clustering with Logistic divergence. Again, the performance of the estimators is globally improved by combining. It is clear that Gaussian kernel does the best job, and $Comb_2^R$ and $Comb_3^R$ alternatively outperform each other.

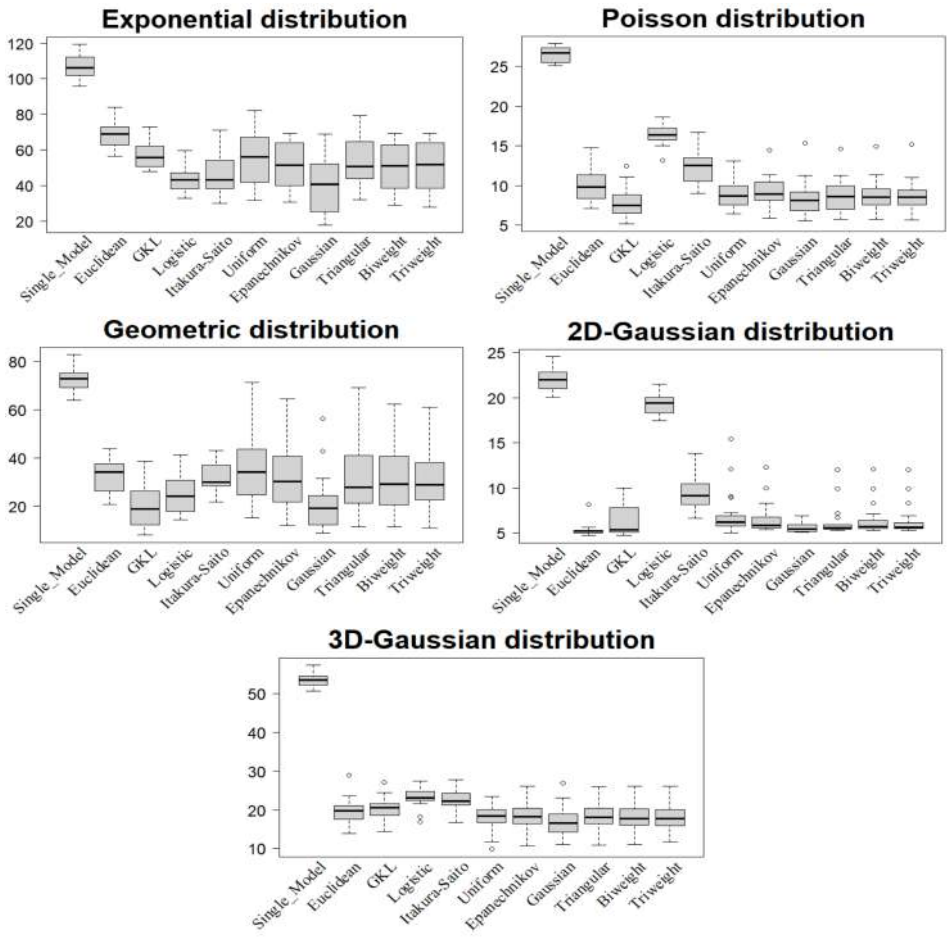

Figure 7: Boxplots of RMSE of $Comb_2^R$.

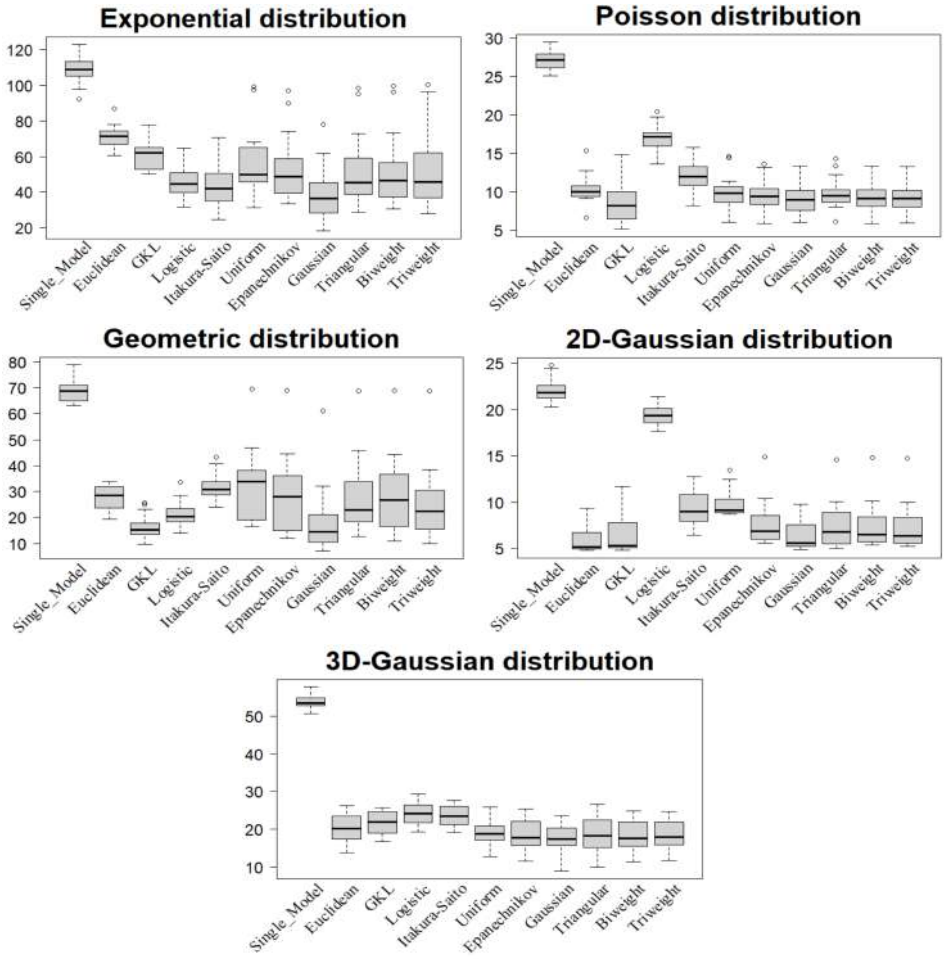

Figure 8: Boxplots of RMSE of $Comb_3^R$.

Figure 7 and Figure 8 above represent the associated boxplots of root mean square errors for $Comb_2^R$ and $Comb_3^R$ respectively (the results of the Table 6).

The numerical results are quite satisfactory, and this is a piece of evidence showing that KFC procedure is an interesting method for building predictive models, especially when the number of existing groups of the input data is available. It is even more interesting in the next section where the procedure is implemented on a real dataset of Air compressor machine for which the number of clustering is not available.

Distribution	**Single**	**Bregman divergence**				**Kernel**					
		Euclid	GKL	Logit	Ita	Unif	Epan	Gaus	Triang	Bi-wgt	Tri-wgt
Exp	106.58 (7.13)	68.74 (6.84)	57.06 (7.37)	44.54 (7.37)	**44.46** (10.96)	55.11 (15.85)	51.14 (13.31)	**40.21** (14.40)	52.99 (13.12)	50.24 (13.74)	50.64 (14.41)
						56.34 (17.48)	52.62 (17.82)	**39.12** (14.98)	51.31 (19.55)	51.20 (19.69)	51.98 (20.12)
Pois	26.76 (1.11)	10.16 (1.91)	**8.22** (2.25)	16.72 (1.61)	12.15 (1.86)	8.88 (1.65)	9.18 (1.98)	**8.43** (2.18)	8.85 (2.06)	8.84 (2.03)	8.76 (2.03)
						9.73 (2.25)	9.61 (1.86)	**9.13** (1.92)	9.64 (1.91)	9.40 (1.86)	9.43 (1.93)
Geom	70.45 (4.52)	29.99 (5.95)	**18.33** (7.34)	22.94 (6.21)	31.94 (5.19)	36.39 (13.81)	32.49 (13.49)	**21.51** (11.79)	31.48 (14.31)	31.44 (13.51)	30.89 (12.21)
						31.83 (12.88)	27.90 (14.20)	**17.82** (12.58)	26.82 (13.28)	28.45 (14.02)	24.58 (13.21)
2D Gaus	21.98 (1.20)	**5.63** (1.26)	6.46 (1.81)	19.36 (1.11)	9.38 (1.86)	7.09 (2.55)	6.57 (1.78)	**5.57** (0.49)	6.20 (1.72)	6.41 (1.76)	6.33 (1.75)
						9.75 (1.30)	7.70 (2.24)	**6.42** (1.49)	7.45 (2.42)	7.47 (2.28)	7.34 (2.31)
3D Gaus	53.55 (1.74)	**19.89** (3.49)	20.93 (2.97)	23.71 (2.70)	22.96 (2.74)	18.16 3.42)	18.20 (3.45)	**16.94** (4.06)	18.25 (3.41)	18.05 (3.50)	18.00 (3.49)
						19.24 (3.54)	18.52 (4.02)	**17.51** (3.64)	18.64 (4.37)	18.19 (3.91)	18.42 (3.68)

Table 6: RMSE of $Comb_2^R$ and $Comb_3^R$ computed over 20 runs of all simulated data.

Throughout the simulation, we could see that the procedure is time-consuming, especially when the implementation is done with more options of Bregman divergences. However, it should be pointed out that the structure of KFC procedure is parallel in a sense that the K and F steps (K-*means* and *Fit* step) of the procedure can be implemented in parallel independently, and only the predictions given by all of those independently constructed estimators are required in the consensual aggregation step.

7 Application

In this section, we study the performance of the KFC procedure on real data. The goal of the application here is to model the power consumption of an air compressor equipment Cadet et al. (2005). The target is the electrical power of the machine, and 6 explanatory variables are available: air temperature, input pressure, output pressure, flow, water temperature. The dataset contains $N = 2000$ hourly observations of a working air compressor. We run the algorithms over 20 random partitions of 80% training sample. The root mean square error (RMSE) computed on the testing sets as well as the associated standard errors are summarized in Table 7. As the number of clusters is unknown, we perform the KFC algorithm with different values of the number of clusters $K \in \{1, 2, ..., 8\}$. For the consensual aggregation step, we use a Gaussian kernel which showed to be the best one in the simulations with synthetic data. Note that for the simple linear model with only one cluster on the whole dataset ($K = 1$), we obtain the average RMSE of 178.67 with the associated standard error of 5.47.

The associated boxplots are given in Figure 9 below. We observe that the performance of the individual estimators improve as the number K of clusters increases. Note that $Comb_3^R$ outperforms $Comb_2^R$ with much lower error (reduced more than 20% of error given by $Comb_2^R$) and also a smaller variance. Regardless of the number of clusters, the combination step allows to reduce the RMSE in each case to approximately the same level. Hence, our strategy may be interesting even without the knowledge of the number of clusters.

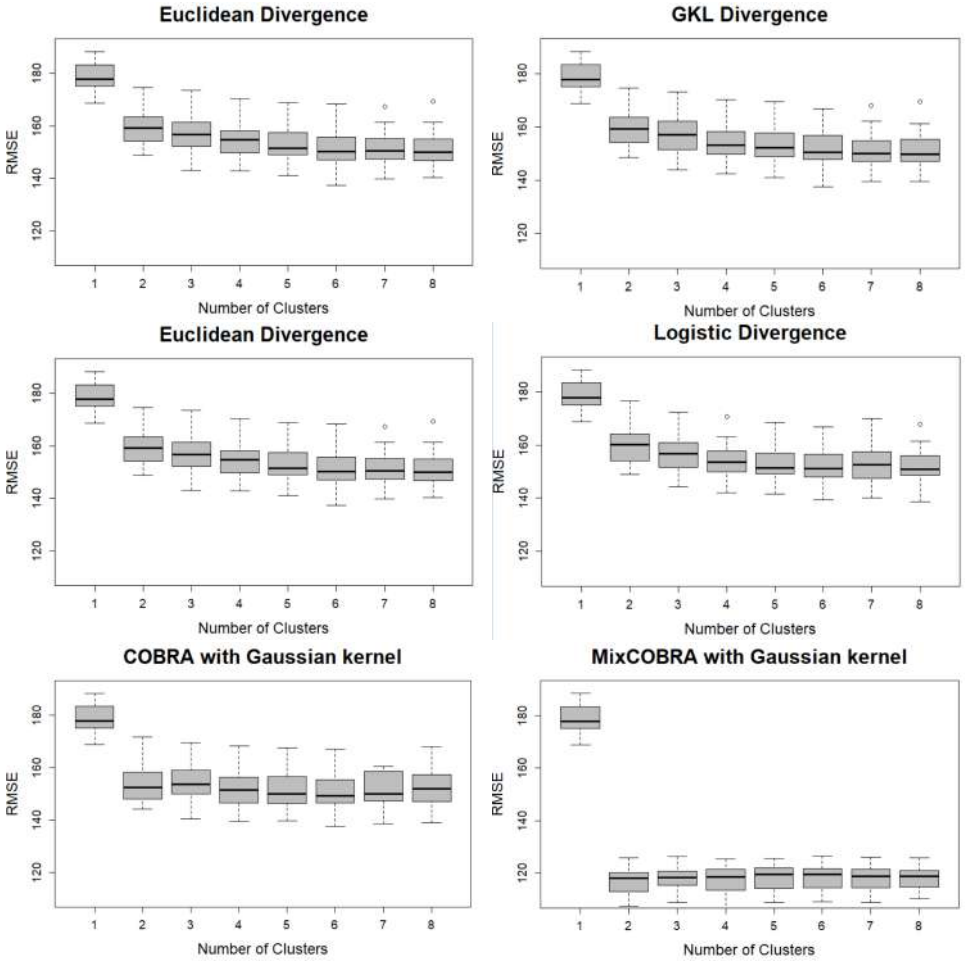

Figure 9: Boxplots of RMSE of $Comb_2^R$ and $Comb_3^R$ evaluated on Air Compressor data.

K	Euclid	GKL	Logistic	Ita	$Comb_2^R$	$Comb_3^R$
2	158.85	158.90	159.35	158.96	153.34	**116.69**
	(6.42)	(6.48)	(6.71)	(6.41)	(6.72)	(5.86)
3	157.38	157.24	156.99	157.24	153.69	**117.45**
	(6.95)	(6.84)	(6.65)	(6.85)	(6.64)	(5.55)
4	154.33	153.96	153.99	154.07	152.09	**117.16**
	(6.69)	(6.74)	(6.45)	(7.01)	(6.58)	(5.99)
5	153.18	153.19	152.95	152.25	151.05	**117.55**
	(6.91)	(6.77)	(6.57)	(6.70)	(6.76)	(5.90)
6	151.16	151.67	151.89	151.75	150.27	**117.74**
	(6.91)	(6.96)	(6.62)	(6.57)	(6.82)	(5.86)
7	151.08	150.99	152.81	151.85	150.46	**117.58**
	(6.77)	(6.84)	(7.11)	(6.61)	(6.87)	(6.15)
8	151.27	151.09	152.07	150.90	150.21	**117.91**
	(7.17)	(7.01)	(6.65)	(6.96)	(7.03)	(5.83)

Table 7: Average RMSE of each algorithm performed on Air Compressor data.

8 Conclusion

The KFC procedure aims to take advantage of the inner groups of input data to provide a consensual aggregation of a set of models fitted in each group built thanks to the K-means algorithm and several Bregman divergences. Simulations using synthetic dataset shown that, in practice, this approach is extremely relevant particularly when groups of unknown distributions belong to the data. The introduction of several Bregman divergences let automatically captures various shapes of groups. The KFC procedure brings also relevant improvements for modeling in real-life applications when missing information may induce inner groups. When the number of groups is unknown, which is often the case, cross-validation on the number of groups helps to find the best configurations.

References

B. Auder and A. Fischer. Projection-based curve clustering. *Journal of Statistical Computation and Simulation*, 82(8):1145–1168, 2012.

N. Balakrishnan and M. Mojirsheibani. A simple method for combining estimates to improve the overall error rates in classification. *Journal of Computational Statistics*, 30(4):1033–1049, December 2015. URL https://link.springer.com/article/10.1007%2Fs00180-015-0571-0.

A. Banerjee, X. Guo, and H. Wang. On the optimality of conditional expectation as a bregman predictor. *IEEE Transactions on Information Theory*, 51(7):2664–2669, 2005a.

A. Banerjee, S. Merugu, I.S. Dhillon, and J. Ghosh. Clustering with Bregman divergences. *Journal of Machine Learning Research*, 6:1705–1749, 2005b. URL http://www.jmlr.org/papers/volume6/banerjee05b/banerjee05b.pdf.

G. Biau, A. Fischer, B. Guedj, and J.D. Malley. COBRA: a combined regression strategy. *Journal of Multivariate Analysis*, 146:18–28, 2016. URL https://www.sciencedirect.com/science/article/pii/S0047259X15000950?via%3Dihub.

L. Breiman. Stacked regressions. *Machine learning*, 24(1):49–64, 1996.

O. Cadet, C. Harper, and M. Mougeot. Monitoring energy performance of compressors with an innovative auto-adaptive approach. In *Instrumentation System and Automation -ISA- Chicago*, 2005.

O. Catoni. *Statistical learning theory and stochastic optimization: Ecole d'Eté de Probabilités de Saint-Flour XXXI-2001*. Springer, 2004.

E. Devijver, Y. Goude, and J.M. Poggi. Clustering electricity consumers using high-dimensional regression mixture models. *arXiv preprint arXiv:1507.00167*, 2015.

A. Fischer and M. Mougeot. Aggregation using input-output trade-off. *Journal of Statistical Planning and Inference*, 200:1–19, May 2019. URL https://www.researchgate.net/publication/323655014_Aggregation_using_input-output_trade-off.

L. Györfi, M. Kohler, A. Krzyzak, and H. Walk. *A distribution-free theory of nonparametric regression*. Springer Science & Business Media, 2006.

A.K. Jain. Data clustering: 50 years beyond k-means. *Pattern recognition letters*, 31(8):651–666, 2010.

N. Keita, S. Bougeard, and G. Saporta. Clusterwise multiblock PLS regression. In *CFE-CMStatistics 2015*, page 195, Londres, Grande Bretagne, December 2015. 8th International Conference of the ERCIM (European Research Consortium for Informatics and Mathematics) Working Group on Computational and Methodological Statistics (CMStatistics 2015) ISBN 978-9963-2227-0-4.

M. LeBlanc and R. Tibshirani. Combining estimates in regression and classification. *Journal of the American Statistical Association*, 91(436):1641–1650, Dec 1996.

T. Linder. Learning-theoretic methods in vector quantization. In László Györfi, editor, *Principle of Nonparametric Learning*. Springer-Verlag, 2001. URL http://www.mast.queensu.ca/~linder/pdf/cism.pdf. Lecture Notes for the Advanced School on the Principles of Nonparametric Learning.

S. Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137, 1982.

M. Mojirsheibani. Combined classifiers via disretization. *Journal of the American Statistical Association*, 94(446):600–609, June 1999. URL http://www.jstor.org/stable/2670180.

M. Mojirsheibani. A kernel-based combined classification rule. *Journal of Statistics and Probability Letters*, 48(4):411–419, July 2000. URL https://www.sciencedirect.com/science/article/pii/S0167715200000249.

M. Mojirsheibani. A comparison study of some combined classifiers. *Communications in Statistics-Simulation and Computation*, 31:245–260, Aug 2006. URL https://www.tandfonline.com/doi/abs/10.1081/SAC-120003337.

M. Mojirsheibani and J. Kong. An asymptotically optimal kernel combined classifier. *Journal of Statistics and Probability Letters*, 119:91–100,

2016. URL https://www.sciencedirect.com/science/article/pii/S0167715216301304?via%3Dihub.

A. Nemirovski. Topics in non-parametric. *Ecole d'Eté de Probabilités de Saint-Flour*, 28:85, 2000.

H. Steinhaus. Sur la division des corp materiels en parties. *Bull. Acad. Polon. Sci*, 1(804):801, 1956.

A. Strehl and J. Ghosh. Cluster ensembles - a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research*, 2002. URL http://www.jmlr.org/papers/v3/strehl02a.html.

C. Tikkinen-Piri, A. Rohunen, and J. Markkula. Eu general data protection regulation: Changes and implications for personal data collecting companies. *Computer Law & Security Review*, 34(1):134–153, 2018.

D.H. Wolpert. Stacked generalization. *Neural Networks*, 5:241–259, 1992. URL https://www.sciencedirect.com/science/article/pii/S0893608005800231#!

L. Xu, A. Krzyzak, and C.Y. Suen. Methods of combining multiple classifiers and their applications to handwriting recognition. *IEEE Transactions on Systems, Man, and Cybernetics*, 22:418–435, 1992. URL https://ieeexplore.ieee.org/document/155943/.

Y. Yang. Combining different procedures for adaptive regression. *Journal of multivariate analysis*, 74(1):135–161, 2000.

Y. Yang et al. Aggregating regression procedures to improve performance. *Bernoulli*, 10(1):25–47, 2004.