FlexPCP: A Clusterwise Predictive Method with Flexible Prototypes

XV Semana de Estatística da UFES

Marcelo R. P. Ferreira
marcelorpf@gmail.com
www.de.ufpb.br/~marcelo

Departamento de ESTATÍSTICA

SAPIENTIA ÆDIFICAT

Joint work with...

Dr. Marcelo Rodrigo Portela Ferreira (DE/UFPB)

Dr. Eufrásio de Andrade Lima Neto (De Montfort University/UK)

Wilter da Silva Dias (CI/UFPB)

José Nataniel Andrade de Sá (CIn/UFPE)

Outline

Introduction

- Two very common tasks in modern data analysis are clustering [Ingrassia et al., 2022, Sedghi et al., 2024, Diday and Simon, 1980, Jain et al., 1999, Xu and Wunsch, 2005, Jain, 2010] and regression [Speller et al., 2023, Kalogridis, 2024]
- Clustering aims to organize a set of observations (individuals, objects, images, pixels, etc.) into groups in such a way that observations belonging to the same group present a high degree of similarity. In contrast, observations belonging to different groups present a high degree of dissimilarity
- Regression methods aim to numerically estimate how a response variable (Y) and a set of explanatory variables ($X_1, \ldots, X_p$) are related through a mathematical equation

Introduction

- Clusterwise linear regression (CLR) [Späth, 1979, 1981, 1982] is a technique that simultaneously obtains a partition of a complete data set in a certain number of groups and estimates the regression coefficients for each group
- This work proposes a flexible clusterwise method to predict a response variable from a set of covariates assuming that the population under study is not homogeneous for the underlying model
- The proposed approach, called the Flexible Prototypes Clusterwise Predictive Method (FlexPCP), aims to segment the data into homogeneous clusters so that each cluster is represented by a predictive model

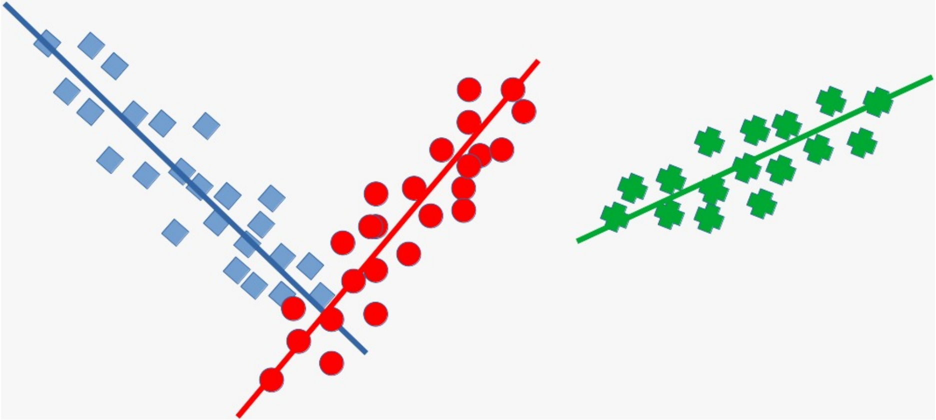
Introduction

- The predictive method that represents each cluster is chosen dynamically in a user-defined list of methods/models/algorithms
- The flexibility of the new method relies on the fact that any predictive statistical model or predictive machine learning algorithm can be considered
- This allow us to find the best relationship between the response variable and the covariates in different clusters (subsets) and improves the predictiveness of the response variable by the use of a mixture of models
- Experiments on real-world as well as synthetic data sets show that the proposed FlexPCP method outperforms its well-established counterpart, the clusterwise linear regression method, in a wide range of situations

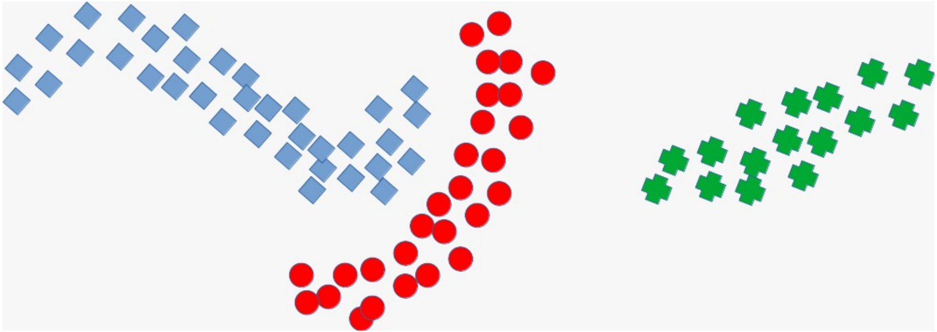
Introduction

- So far, the methods in the literature consider the same model (e.g. linear regression model and its variants or the non-linear regression model or SVMs, etc.) to describe the relationship between the response variable and a set of predictor variables in all clusters
- Despite its wide applicability, the CLR method is not able to properly handle non-linear data
- At the same time, it is challenging to define which non-linear function or machine learning algorithm describes the relationship between the response variable and a set of predictor variables for all clusters
- Moreover, it is too restrictive to assume the same regression method or ML algorithm in all the clusters, assuming a common type of relationship, e.g., SVMs

Introduction



Introduction



Clusterwise Linear Regression

- The CLR method is a combination of the dynamic clustering algorithm [Diday and Simon, 1980] and the linear regression method
- It delivers a partition $P_1, \ldots, P_K$ of a set of examples E into a specified number of clusters K along with K prototypes represented by linear regression models
- For each cluster P_k, let $n_k = |P_k|$ and let $e_{i_l} \in P_k$ $(1 \leq l \leq n_k)$ be described by the response variables Y_k and the respective set of covariates $X_{1(k)}, X_{2(k)}, \ldots, X_{p(k)}$ $(k = 1, \ldots, K)$
- It is assumed that the relationship of the response variable Y with the covariates X_j $(1 \leq j \leq p)$ is expressed as:

$$y_{i_l(k)} = \beta_{0(k)} + \sum_{j=1}^{p} \beta_{j(k)} x_{i_l j} + \epsilon_{i_l(k)}. \quad (1)$$

Clusterwise Linear Regression

- The local optimization of a suitable objective function delivers the clusters $P_1, P_2, \ldots, P_K$ and the respective K prototypes (K linear regression models)
- The total within-cluster sum-of-squares deviations is computed by the following objective function:

$$S_{CLR} = \sum_{k=1}^{K} \sum_{e_{i_l} \in P_k} (\epsilon_{i_l(k)})^2 = \sum_{k=1}^{K} \sum_{e_{i_l} \in P_k} \left[y_{i_l(k)} - \left(\beta_{0(k)} + \sum_{j=1}^{p} \beta_{j(k)} x_{i_l j} \right) \right]^2$$
$$= \sum_{k=1}^{K} \left[\left(\mathbf{y}_{(k)} - \mathbf{X}_{(k)} \boldsymbol{\beta}_{(k)} \right)^\top \left(\mathbf{y}_{(k)} - \mathbf{X}_{(k)} \boldsymbol{\beta}_{(k)} \right) \right], \quad (2)$$

Clusterwise Linear Regression

- where:

$$\underset{(n_k \times (p+1))}{\mathbf{X}_{(k)}} = \begin{pmatrix} 1 & x_{i_1 1} & \dots & x_{i_1 p} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{i_{n_k} 1} & \dots & x_{i_{n_k} p} \end{pmatrix},$$

$$\underset{((p+1)\times 1)}{\boldsymbol{\beta}_{(k)}} = \begin{pmatrix} \beta_{0(k)} \\ \vdots \\ \beta_{p(k)} \end{pmatrix}, \quad \underset{(n_k \times 1)}{\mathbf{y}_{(k)}} = \begin{pmatrix} y_{i_1} \\ \vdots \\ y_{i_{n_k}} \end{pmatrix}.$$

- Starting from an initial solution (either random or user-provided), the algorithm alternates between two steps: the fitting step, which determines the cluster prototypes (K linear regression models) and the assignment step, which produces the partition. This process continues until the convergence is achieved, meaning there are no further changes in the partition $P_1, \dots, P_K$.

Clusterwise Linear Regression

Step 1: best-fitting (prototypes update)
We have that the partition of E in K clusters is fixed. To estimate the coefficient vectors $\boldsymbol{\beta}_{(k)}(k = 1, \ldots, K)$ that minimize S_{CLR}, we differenciate the equation (2) with respect to the coefficient vector and set it equal to zero. If the model matrices $\mathbf{X}_{(k)}$ $(k = 1, \ldots, K)$ have full rank, the least squares estimator of $\boldsymbol{\beta}_{(k)}$ is the solution of the system with $(p + 1)$ normal equations, given by:

$$\hat{\boldsymbol{\beta}}_{(k)} = \left(\boldsymbol{X}_{(k)}^{\top}\boldsymbol{X}_{(k)}\right)^{-1}\boldsymbol{X}_{(k)}^{\top}\boldsymbol{y}_{(k)}. \tag{3}$$

Clusterwise Linear Regression

Step 2: best assignment

Now, the estimated coefficient vectors $\hat{\beta}_{\mathbf{(k)}}$ $(k = 1, \ldots, K)$ are kept fixed and the optimal clusters P_k which minimize the criterion S_{CLR}, are obtained according to the following assignment rule:

$$P_k = \left\{ e_i \in E : \left(\epsilon_{i(k)}\right)^2 = \min_{h=1}^{K} \left(\epsilon_{i(h)}\right)^2 \right\}. \tag{4}$$

Therefore, the example e_i is assigned to cluster P_k if the squared error is minimal for this cluster when compared to the other squared errors for the observation e_i when computed by the linear models of the other $K - 1$ clusters. That is, the observation e_i $(i = 1, \ldots, n)$ will be assigned to the cluster P_k that minimizes the squared error.

FlexPCP: A Clusterwise Predictive Method with Flexible Prototypes

- Let $\mathcal{H} = \{f_1, \ldots, f_{\mathcal{H}}\}$ be a set of machine learning algorithms and/or statistical models like Support Vector Regression (SVR), Generalized Linear Models (GLM), K-NN Regression, Robust Regression, Kernel Regression, Gradient Boosting, etc.
- Consider the partition $\mathcal{P} = (P_1, \ldots, P_K)$ of E into K clusters and, for each cluster P_k, $(k = 1, \ldots, K)$, let $n_k = |P_k|$ and let $e_{i_l} \in P_k$ $(1 \leq l \leq n_k)$ be described by a response variable Y_k and their respective set of explanatory variables $X_{1(k)}, X_{2(k)}, \ldots, X_{p(k)}$ $(k = 1, \ldots, K)$

FlexPCP: A Clusterwise Predictive Method with Flexible Prototypes

- We assume that each cluster k $(k = 1, \dots, K)$ presents the following relationship between the response variable Y and a set of covariates $X_1, X_2, \dots, X_p$:

$$y_{i_l(k)} = f_{(k)}\left(\mathbf{x}_{i_l}, \boldsymbol{\beta}_{(k)}\right) + \epsilon_{i_l(k)}, \tag{5}$$

where $f_{(k)} \in \mathcal{H}$ and $\boldsymbol{\beta}_{(k)}$ is the vector of coefficients if a parametric model is selected as the best one for a given cluster k. A non-parametric model $f_k(x)$ with M hyperparameters can be also considered for the cluster k without loss of generalization

FlexPCP: A Clusterwise Predictive Method with Flexible Prototypes

- The K clusters and their respective K models (flexible prototypes) are obtained by a local iterative optimization process of the cost function that represents the total within cluster sum-of-squares of errors, which is expressed as:

$$S_{FlexPCP} = \sum_{k=1}^{K} \sum_{e_{i_l} \in P_k} \epsilon^2_{i_l(k)} = \sum_{k=1}^{K} \sum_{e_{i_l} \in P_k} \left[y_{i_l(k)} - f_{(k)} \left(\mathbf{x}_{i_l}, \beta_{(k)} \right) \right]^2 . \quad (6)$$

- From an initial (random or user-provided) solution, the algorithm alternates between the fitting step, which delivers the best K models (the flexible prototypes), and the assignment step, which provides the partition $P_1, \ldots, P_K$, until convergence, when there are no more assignment changes of objects into clusters

FlexPCP: A Clusterwise Predictive Method with Flexible Prototypes

Step 1: best-fitting (prototypes update)

The partition of E in K clusters is kept fixed. Then, the algorithm finds the best set of K models $f_{(k)} \in \mathcal{H}$ that minimizes the objective function $S_{FlexPCP}$:

$$f_{(k)h} = \sum_{k=1}^{K} \min_{1 \leq h \leq H} f_h, \quad \text{wehre } f_h = \sum_{e_{i_l} \in P_k} \left[y_{i_l(k)} - f_{(k)h} \left(\mathbf{x}_{i_l}, \hat{\beta}_{(k)} \right) \right]^2 . \qquad (7)$$

As the objective function is additive in K, the solution of the expression (7) represents a local optimization of each cluster by applying the set of models belongs to $\mathcal{H} = \{f_1, \ldots, f_{\mathcal{H}}\}$ and selecting the model $f_{(k)}$ that presented the minimal sum-of-squares of errors within the cluster k $(k = 1, \ldots, K)$

FlexPCP: A Clusterwise Predictive Method with Flexible Prototypes

Step 2: best assignment

Now, the models $f_{(k)}(k = 1, \ldots, K)$ are kept fixed. The optimal clusters P_k which minimize the criterion $S_{FlexPCP}$, are obtained according to the following assignment rule:

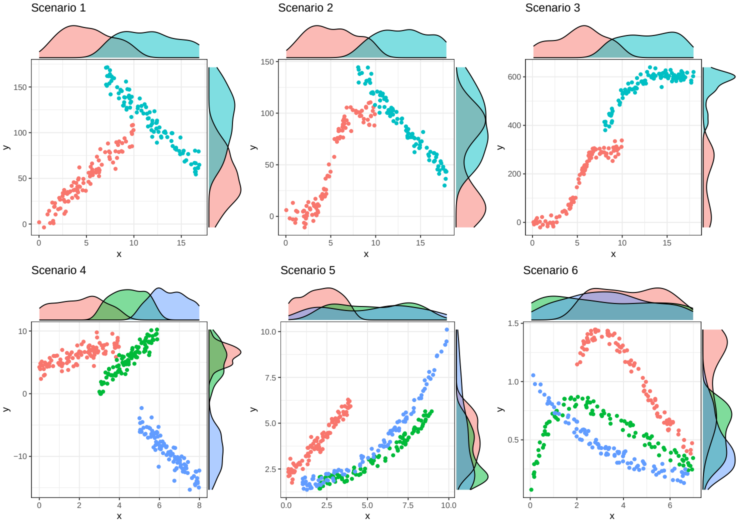
$$P_k = \left\{ e_i \in E : \left(\epsilon_{i(k)}\right)^2 = \min_{h=1}^{K} \left(\epsilon_{i(h)}\right)^2 \right\}. \tag{8}$$

Thus, the example e_i will be allocated to the cluster P_k if the squared error is minimal for P_k, in comparison with the squared errors obtained by the prototype models of the remaining $K-1$ clusters for that same example e_i

Numerical Experiments

- Simulated as well as real data sets
- The methods were evaluated in terms of the root mean square error (RMSE)
- For the simulated scenarios, a Wilcoxon non-parametric test was used to compare the approaches after a Monte Carlo simulation with 500 replicates
- We considered 20 different models into the set $\mathcal{H}$ of available models based on 6 different techniques: Generalized Linear Models (GLM), Support Vector Regression (SVR), Generalized Additive Models (GAM), K-NN Regression, Conditional Inference Trees and Robust Regression
- The flexibility of the FlexPCP method allows considering different techniques for the problem (parametric, nonparametric, robust, semiparametric and machine learning methods)

Numerical Experiments: simulated data



Numerical Experiments: simulated data

Table: Comparison between the methods FlexPCP and CLR, by scenario. Mean and standard deviation (in brackets) for the RMSE. *p*-value for the Wilcoxon nonparametric test

Method	Scenario					
	1	2	3	4	5	6
FlexPCP	5.0242 (0.4584)	3.5390 (0.3021)	10.0416 (1.0839)	0.5605 (0.0968)	0.1762 (0.0411)	0.0480 (0.0060)
CLR	7.7166 (0.3880)	10.8390 (0.5257)	36.2151 (1.3307)	0.9350 (0.0455)	0.3176 (0.0383)	0.0949 (0.0055)
p-value	< 0.0001	< 0.0001	< 0.0001	< 0.0001	< 0.0001	< 0.0001

Numerical Experiments: real data

- The FlexPCP method was compared to the CLR method as well as to the independent run of the K-means algorithm followed by OLS fit for each cluster (OLS-KM) and the K-means method followed by the fit of the best one among the $\mathcal{H}$ models (BF-KM)
- For the sake of simplicity, the prior information about K is obtained using the K-means method. We consider a grid of values between 2 and $\sqrt{n}$ and the elbow's method was used to define the number of clusters K for each data set
- The methods were evaluated based on the predictive performance for unseen instances, according to the root mean square error (RMSE) in independent test data sets, under a 10-fold cross-validation scheme.

Numerical Experiments: real data

Table: Description of the real data sets according to sample size and number of explanatory variables (p)

Data Set	Sample size	p	Description
Energy Efficiency	768	7	Heating/cooling load and requirements of buildings as a function of building parameters
Auto MPG	398	6	Predict the fuel-efficiency base on car features
Liver Disorders	345	5	Predict the amount of alcoholic beverages drunk per day based on blood test information
Real Estate Valuation	414	6	The aim is to predict the household price based on some features

Numerical Experiments: real data

Table: Description of the real data sets according to sample size and number of explanatory variables (p)

Data Set	Sample size	p	Description
Blood Transfusion Service	748	4	Predict the time to return to the blood transfusion service centre
Concrete	1030	8	Predict the concrete compressive strength for a given specification mixture
Bike Sharing	731	7	Predict the daily number of bike rentals based on weather conditions
Abalone	4177	6	Predict the weight based on physical measurements and the number of rings

Numerical Experiments: real data

Table: Comparative performance between the methods by real data sets. Values of the objective function (S) and the best-fitted models, by approach

Data set	Method	S	Best model (per cluster)
Energy Efficiency ($K = 3$)	OLS-KM	2.917	3 Linear models
	BF-KM	0.881	$\mathrm{SVM}^{(a,\nu)}$, GAM, GAM
	CLR	0.319	3 Linear models
	FlexPCP	0.205	GAM, GAM, GAM
Auto MPG ($K = 4$)	OLS-KM	408,252.00	4 Linear models
	BF-KM	240,561.50	$\mathrm{SVM}^{(a,\nu)}$, $\mathrm{SVM}^{(a,\varepsilon)}$, $\mathrm{SVM}^{(a,\nu)}$, GAM
	CLR	41,684.65	4 Linear models
	FlexPCP	31,807.53	$\mathrm{SVM}^{(a,\nu)}$, GAM, $\mathrm{SVM}^{(a,\nu)}$, RR

Numerical Experiments: real data

Table: Comparative performance between the methods by real data sets. Values of the objective function (S) and the best-fitted models, by approach

Data set	Method	S	Best model (per cluster)
Liver Disorders ($K = 4$)	OLS-KM	533.379	4 Linear models
	BF-KM	348.076	$\mathrm{SVM}^{(a,\nu)}$, $\mathrm{SVM}^{(a,\varepsilon)}$, Ctree, $\mathrm{SVM}^{(a,\nu)}$
	CLR	253.850	4 Linear models
	FlexPCP	236.804	$\mathrm{SVM}^{(b,\varepsilon)}$, $\mathrm{SVM}^{(b,\varepsilon)}$,$\mathrm{SVM}^{(a,\varepsilon)}$, $\mathrm{SVM}^{(a,\varepsilon)}$
Real Estate Valuation ($K = 4$)	OLS-KM	8,448.040	4 Linear models
	BF-KM	4,645.808	$\mathrm{SVM}^{(b,\nu)}$, $\mathrm{SVM}^{(a,\varepsilon)}$, $\mathrm{SVM}^{(a,\varepsilon)}$, $\mathrm{SVM}^{(a,\varepsilon)}$
	CLR	3,745.922	4 Linear models
	FlexPCP	2,033.518	GAM, CTREE, $\mathrm{SVM}^{(a,\nu)}$, GAM

Numerical Experiments: real data

Table: Comparative performance between the methods by real data sets. Values of the objective function (S) and the best-fitted models, by approach

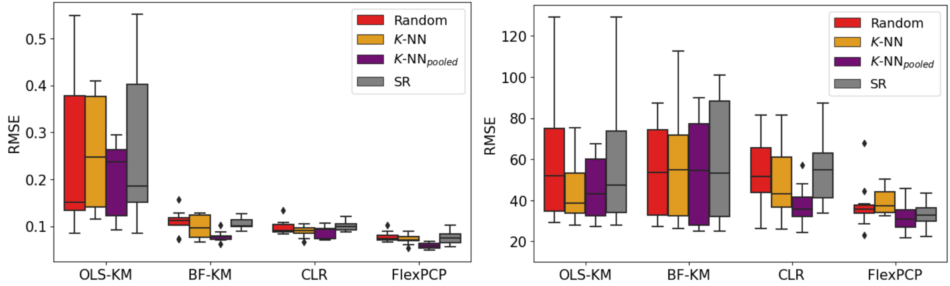
Data set	Method	S	Best model (per cluster)
Blood Transfusion Service ($K = 4$)	OLS-KM	7,366.938	4 Linear models
	BF-KM	2,408.565	4 GAM
	CLR	1,712.776	4 Linear models
	FlexPCP	1,578.459	4 GAM
Concrete ($K = 4$)	OLS-KM	20,838.700	4 Linear models
	BF-KM	11,265.06	GAM, $\mathrm{SVM}^{(a,\varepsilon)}$, $\mathrm{SVM}^{(a,\nu)}$, GAM
	CLR	8,306.293	4 Linear models
	FlexPCP	5,953.834	4 $\mathrm{SVM}^{(a,\nu)}$

Numerical Experiments: real data

Table: Comparative performance between the methods by real data sets. Values of the objective function (S) and the best-fitted models, by approach

Data set	Method	S	Best model (per cluster)
Bike Sharing ($K = 4$)	OLS-KM	161,852,597	4 Linear models
	BF-KM	116,318,303	$\mathrm{SVM}^{(a,\varepsilon)}$, $\mathrm{SVM}^{(a,\varepsilon)}$, $\mathrm{SVM}^{(a,\varepsilon)}$, GAM
	CLR	71,095,420	4 Linear models
	FlexPCP	35,886,294	4 $\mathrm{SVM}^{(a,\nu)}$
Abalone ($K = 4$)	OLS-KM	8.720	4 Linear models
	BF-KM	7.349	$\mathrm{SVM}^{(a,\varepsilon)}$, $\mathrm{SVM}^{(a,\nu)}$, $\mathrm{SVM}^{(a,\varepsilon)}$, RR
	CLR	1.467	4 Linear models
	FlexPCP	1.497	$\mathrm{SVM}^{(a,\nu)}$, GAM, GAM, $\mathrm{SVM}^{(a,\nu)}$

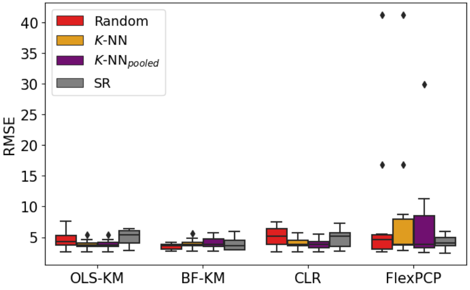
Numerical Experiments: real data



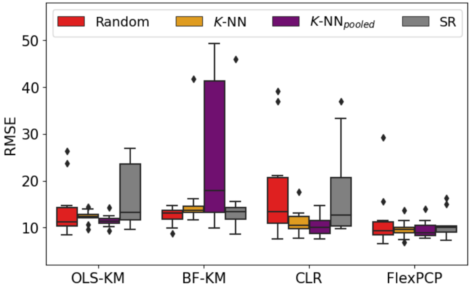
(a) Energy Efficiency (b) Auto MPG

Figure: Boxplots of RMSE across the 10 folds of cross-validation on each data set

Numerical Experiments: real data



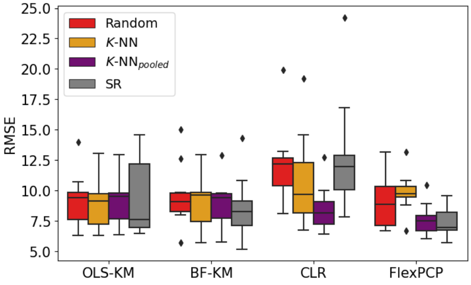
(c) Liver Disorders



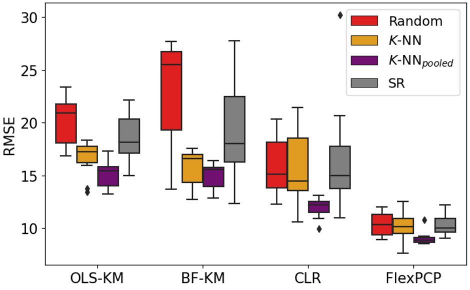
(d) Real Estate Valuation

Figure: Boxplots of RMSE across the 10 folds of cross-validation on each data set

Numerical Experiments: real data



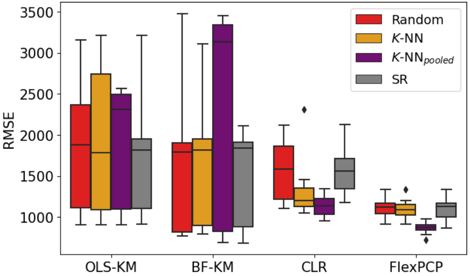
(e) Blood Transfusion Service



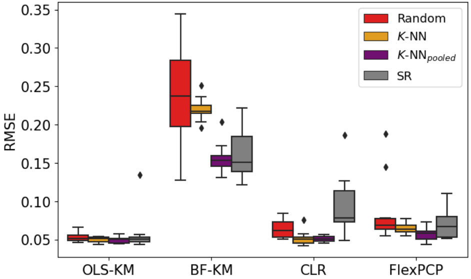
(f) Concrete

Figure: Boxplots of RMSE across the 10 folds of cross-validation on each data set

Numerical Experiments: real data



(g) Bike Sharing



(h) Abalone

Figure: Boxplots of RMSE across the 10 folds of cross-validation on each data set

Numerical Experiments: real data

Table: Predictive performance overall rank for the clusterwise methods and combined approaches in the real data sets

Method	Data set								Cumulative ranking
	Energy Efficiency	Auto MPG	Liver Disorders	Real Estate Valuation	Blood Transfusion	Concrete	Bike Sharing	Abalone	
OLS-KM	4	3	2	3	4	4	4	1	25
BF-KM	2	4	1	4	3	3	3	4	24
CLR	3	2	3	2	2	2	2	2	18
FlexPCP	1	1	4	1	1	1	1	3	13

Numerical Experiments: real data

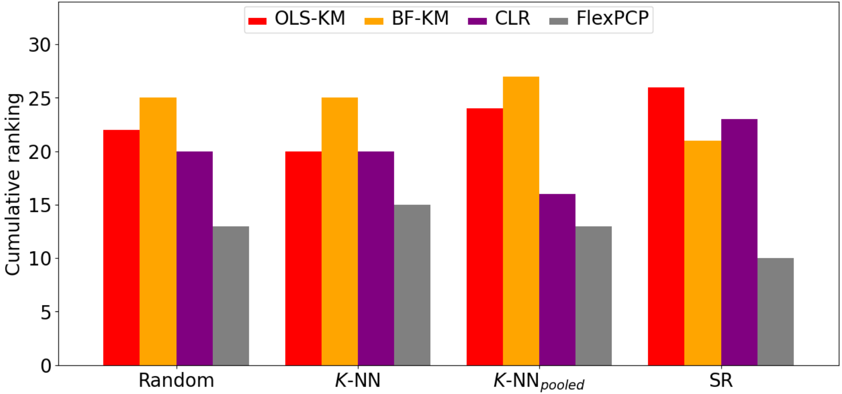


Figure: Cumulative ranking of the models according to the assignment strategy

Conclusion

- We proposed FlexPCP: a new clusterwise method allowing to consider a set of different statistical models and/or machine learning algorithms into a set of K homogeneous groups
- The FlecPCP method, with 20 different models representing 6 different techniques, outperformed the CLR method in all considered simulated scenarios
- The results on real data demonstrated that the FlexPCP method achieved better performance in comparison with the CLR method and two naive approaches according to the value of the objective function
- Regarding the predictive performance, the FlexPCP method showed better results compared to the CLR method and the OLS-KM and BF-KM approaches

Main References I

E. Diday and J.C. Simon. Clustering analysis. In K.S. Fu, editor, *Digital Pattern Recognition*, pages 47–94. Springer, 1980.

Salvatore Ingrassia, Julien Jacques, and Weixin Yao. Special issue on “models and learning for clustering and classification”. *Advances in Data Analysis and Classification*, 16(2):231–234, 2022.

A.K. Jain. Data clustering: 50 years beyond k-means. *Pattern Recogn. Lett.*, 31(8): 651–666, June 2010. ISSN 0167-8655. doi: 10.1016/j.patrec.2009.09.011. URL `https://doi.org/10.1016/j.patrec.2009.09.011`.

A.K. Jain, M.N. Murty, and P.J. Flynn. Data clustering: a review. *ACM Comput. Surv.*, 31(3):264–323, 1999. ISSN 0360-0300. doi: http://doi.acm.org/10.1145/331499.331504.

Ioannis Kalogridis. Robust and adaptive functional logistic regression. *Computational Statistics & Data Analysis*, 192:107905, 2024.

Main References II

Mohaddeseh Sedghi, Ebrahim Akbari, Homayun Motameni, and Touraj Banirostam. Clustering ensemble extraction: a knowledge reuse framework. *Advances in Data Analysis and Classification*, pages 1–28, 2024.

H. Späth. Algorithm 39 clusterwise linear regression. *Computing*, 22(4):367–373, 1979.

H. Späth. Correction to algorithm 39 clusterwise linear regression. *Computing*, 26: 275–275, 09 1981. doi: 10.1007/BF02243486.

H. Späth. A fast algorithm for clusterwise linear regression. *Computing*, 29(2): 175–181, 1982.

Jan Speller, Christian Staerk, Francisco Gude, and Andreas Mayr. Robust gradient boosting for generalized additive models for location, scale and shape. *Advances in Data Analysis and Classification*, pages 1–20, 2023.

R. Xu and D. Wunsch. Survey of clustering algorithms. *IEEE Transactions on neural networks*, 16(3):645–678, 2005.

FlexPCP: A Clusterwise Predictive Method with Flexible Prototypes

XV Semana de Estatística da UFES

Marcelo R. P. Ferreira
marcelorpf@gmail.com
www.de.ufpb.br/~marcelo

Departamento de ESTATÍSTICA

SAPIENTIA ÆDIFICAT