NOVA IMS Information Management School

MEGI

Mestrado em Estatística e Gestão de Informação

Master Program in Statistics and Information Management

Optimal level collapsing: *olc*

A tool for modelling in R

Acácio Luis Ramalho Mattos

Dissertation presented as partial requirement for obtaining the Master's degree in Statistics and Information Management

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação

Universidade Nova de Lisboa

NOVA Information Management School

Instituto Superior de Estatística e Gestão de Informação

Universidade Nova de Lisboa

OPTIMAL LEVEL COLLAPSING: OLC

by

Acácio Luis Ramalho Mattos

Dissertation presented as partial requirement for obtaining the Master's degree in Statistics and Information Management, with a specialization in Information Analysis and Management

Advisor : Jorge Mendes

June 2019

ABSTRACT

Using nominal variables with many levels as predictors in statistical modelling can be a difficult task. Ways to group the levels of such variables optimally becomes of special interest.

In this work we propose a procedure and a tool implemented in R language to deal with this type of situation, in order to create an optimally-recategorized variable to modelling or simply for descriptive purposes. An example with a real data in Insurance environment will be conducted to illustrate the performance, as well the advantages and disadvantages.

KEYWORDS

Optimal Level collapsing; level grouping; nominal variables; categorical variables; releveling; olc

INDEX

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

The process of statistical modelling is becoming more complex as large amount of data available. Dealing with all this information becomes an extremely complex task, especially when it comes to estimation of coefficients or calculation of metrics when there are many nominal variables, many of which can assume many levels.

A large insurance company, for example, has information about the city and / or car model of its customers, but this type of nominal variables becomes difficult to include in a model due to its high number of levels.

One option is to simply ignore this information. Another is to apply some type of cluster using other variables of a nature related to the categorical variable (not the dependent variable to be originally modeled), and to use the clustered variable as independent. This option is feasible, that is, the cluster will tend to be similar in relation to those attributes used in clustering, but does not take into account the dependent variable. This may make the clusters discordant in this sense, for instance, 1 cluster can have 10 car models very similar related to horse power, weight, cylinders etc... We can expect that they have the same behavior related to the dependent variable. But we cannot be sure about that.

Another possibly better option is to use a grouping/collapsing technique for nominal variables when there is no previous knowledge about the behavior of the variable in relation to the response.

1.1. BACKGROUND

The problem of grouping (or collapsing) levels of categorical variables is well known and discussed in the literature. It is usually the analysis done after Analysis of Variance (ANOVA), and aims to regroup the levels of the categorical variable (also called factor) according to the response variable. Common approaches are Tukey's multiple comparisons (Tukey, 1949), Fisher's Least Significant Squared Differences (Fisher, 1935), Bonferroni simultaneous confidence intervals (Montgomery, 2013), Scheffe's method for judging all contrasts in ANOVA (Scheffe, 1953), or False Discovery Rate (Bondell & Reich, 2009). A different approach for grouping the levels of categorical variables after the ANOVA is through a cluster analysis method (Scott & Knott, 1974).

In the last decades, estimators or estimation procedures have been proposed with the intention of dealing especially with this particularity, such as the "Pairwise Fused Lasso" (PFL) (Petry, Flexeder & Tutz, 2011) and "Collapsing and Shrinkage in ANOVA" (CAS-ANOVA) (Bondell & Reich, 2009). PFL use penalized likelihood estimates of coefficients, that is, the optimization process for finding the estimates that maximize the likelihood function has another term, a penalty term, that penalizes the difference between the pairs coefficients, collapsing them. CAS-ANOVA uses an additional constraint in the least square procedure, which uses the differences between the pairs of coefficients.

1.2. STUDY OBJECTIVE

In this work we propose the procedure and the implementation of the “optimal level collapsing” an auxiliary tool for statistical modeling, specifically for grouping levels of categorical variables optimally (in relation to a dependent variable).

2. PROBLEM IDENTIFICATION

Generalized Linear Models (GLM) (Nelder & Wedderburn, 1972) have been widely used since their formalization, both in industry and academy. The procedure for dealing with categorical variables involves some type of coding, which in turn involves the creation of auxiliary variables and finally has the potential to exponentiate the complexity of a model.

One way to treat this type of variable before inserting it into a final set of potential predictors for an GLM, so as to minimize the loss of information while drastically decreasing the number of parameters can be of great advantage when investigating a model with hundreds of potential variables that include categorical with still hundreds or more levels.

Suppose we want to use an important variable for a specific business in a GLM, but it has a very granular level of detail (car model in a claims model, most relevant words in comments on the page of a particular brand for a model of marketing campaigns, municipality or even neighborhood in a model of credit, etc.). In this type of scenario, a considerable amount of data is already expected to be available. If two or more independent variables with the same characteristics exist, it is almost impractical to interpret all the parameters of the model, not to mention the possibility of interaction effects.

Approaches such as LASSO and CAS-ANOVA attempt to identify coefficients (which may be related to dummy coded variables representing a single categorical variable or interactions) equal to 0 to group them simultaneously with other variables. The proposal of the procedure to be presented in this work is to preserve the interpretability that a categorical variable offers and at the same time treat it in an optimal way making it easier to be used in a later model, whether GLM or not.

3. DATA DESCRIPTION

The data that will be used in this paper to illustrate the proposed procedure contains information on the number of claims and car brand for 428512 passenger car policies of a large Portuguese insurer. The policies refer to private or private cars without a fleet. The oldest reported information policy dates from 2007 and the latest from 2018.

Initially, the database presented 922 unique Brand values. As most of data inserted values in the database was probably done manually, many inconsistencies were found, such as the same Brand or Model inserted differently, so it was necessary to clean the data. Using regular expression analysis (in *R*), it was possible to standardize most of the inconsistencies found, such, for instance, the Renault Brand, which had entries such as "REMAULT", "Renault", "RENAULT", "renault", "MEGANE "," CLIO "and etc. It was possible to reduce the number of unique values in the Brand variable from 922 to 212.

After the treatment of the Brand variable, it was defined that the minimum frequency of one level must be 3 observations, so that it was possible to obtain an estimate of the parameter related to the level. Thus, the Brand variable was recoded, so that levels with frequency less than 3 were grouped into a level called "Other". Only 163 observations have been grouped at this level, and the Brand variable that will be used (treated) has 81 levels.

Table 3.1 shows 10 random observations from the database.

Table 3.1 – Data base structure

Number of claims (Y)	**Brand**
0	ISUZU
0	RENAULT
1	DAIHATSU
0	RENAULT
0	BMW
1	MERCEDEZ
0	VOLKSWAGEN
0	VOLKSWAGEN
0	PEUGEOT
0	PEUGEOT

4. METHODOLOGY

Let $\boldsymbol{y}$ be a vector of a dependent variable with dimension n where n is the number of observations and $\boldsymbol{x}$ a vector of a categorical independent variable with k levels. We want to create $\boldsymbol{x}_2$, a variable with j levels ($j < k$) where these levels are the ones from $\boldsymbol{x}$, but collapsed in an optimal way, in relation to $\boldsymbol{y}$. If k is lower than 30, this could not sound like an issue, but in problems where we have a large number of variables with some of them nominal assuming a lot of levels, this could be very problematic.

This section will provide some basic concepts about the methods and techniques that will be used in the procedure and algorithm proposed, like Generalized Linear Models (GLM), Analysis of Variance (ANOVA), Clustering, Self-Organizing Maps (SOM) and a description of the "Optimal level collapsing" procedure.

4.1. GENERALIZED LINEAR MODELS

Let **Y** be a single random response variable that follows some probability distribution. We say that the distribution belongs to the exponential family of distributions if its probability or density function can be written as:

$$f(y, \theta, \varphi) = exp\left\{\frac{y\theta - b(\theta)}{a(\varphi)} + c(y, \varphi)\right\} \quad (1)$$

where θ is our parameter of interest, a(.), b(.) and c(.) are functions with different forms depending on the distribution and φ is called a nuisance parameter, only relevant in some distributions (Muller, 2012).

The Generalized linear model is determined by two main components: The probability distribution of **Y** (also called as a random component), and the link function. There is a third component called the systematic component, known as η, which connects the expected value of Y, $E[Y] = \mu$, to a linear combination of variables $\boldsymbol{X}^T\beta$.

4.1.1. The probability distribution

The probability distribution can be a Gaussian, also called Normal distribution, with probability density function. The notation for "Y follows a normal distribution" is $Y \sim N(\mu\,; \sigma^2)$, where μ and σ^2 corresponds to the parameters mean and variance. The probability density function of Y is given by:

$$f(y \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left\{\frac{(y-\mu)^2}{2\sigma^2}\right\}. \quad (2)$$

Note that we can rewrite (2) as (1) setting $\mu = \theta$, $\sigma = \varphi$, $a(\varphi) = \varphi^2$, $b(\theta) = \frac{\theta^2}{2}$ and $c(y, \varphi) = -\frac{y^2}{2\varphi^2} - \log(\sqrt{2\pi\varphi})$.

4.1.2. The link function

The link function is determined after identifying a possible probability distribution of **Y**, so it is the second component of the GLM, known as the G function. This function links the expected value of Y, $E[Y]$, giving some probability distribution, to the θ parameter of the general exponential family form

of distributions. For the normal example, as we have $\mu = \theta \leftrightarrow \mu(\theta) = 1\theta$, G in that case can be the identity function (Muller, 2012). For the Bernoulli case, we can have $\mu = \frac{\exp(\theta)}{1+\exp(\theta)}$ (also known as the *Logit* or *Logistic* link function). In Poisson case, we have that $\mu = \log(\theta)$.

4.1.3. The systematic component

The systematic component is a linear predictor $\eta = \boldsymbol{X}^T\beta$, and its connected to the expected value of Y, $E[Y] = \mu$, by the link function G. Thus $G(\mu) = \eta = \boldsymbol{X}^T\beta$. Recall that for the Normal case the G function can be the identity, so $\mu = \boldsymbol{X}^T\beta$. This is the classical general linear model, one of the most known and used models. When the parameter $\theta = \eta$, then G is called a canonical link function. So identity and logit functions are canonical link functions for the Normal and Bernoulli distribution, respectively.

4.1.3.1. Dummy coding

Consider $\boldsymbol{x}$ above and that we want to model $\boldsymbol{y}$ trough a GLM. As $\boldsymbol{x}$ is a nominal variable, we need to use some type of coding. Considering dummy coding, we will have:

$$\boldsymbol{X} = \begin{bmatrix} 1 & f(x_1)_1 & \dots & f(x_{k-1})_1 \\ 1 & f(x_1)_2 & \dots & f(x_{k-1})_2 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & f(x_1)_n & \dots & f(x_{k-1})_n \end{bmatrix}, \text{ where } f(x_j)_i = \begin{cases} 1, & if\ individual\ i\ assume\ level\ j\ of\ \boldsymbol{x} \\ 0, & otheriwse \end{cases}$$

And then we use $\boldsymbol{X}$ to model $\boldsymbol{y}$:

$$G(\boldsymbol{y}) = \boldsymbol{X}^T\beta + \boldsymbol{\varepsilon}$$

After the estimation process, we will have $\widehat{\boldsymbol{\beta}}$, which have the coefficients related to which one of k-1 levels. These coefficients, if the estimation process was well conducted, carry the information of those levels in relation to the dependent variable $\boldsymbol{y}$, taking one level as reference. That is, the impact that level j has in $\boldsymbol{y}$ taking k as reference.

4.1.4. Estimatior properties

The estimator $\hat{\beta}$, has an asymptotic normal distribution, that is:

$$\widehat{\boldsymbol{\beta}} \sim N_p(\boldsymbol{\beta}, \boldsymbol{\Sigma}^{-1}) \text{ when } n \to \infty$$

Where $\boldsymbol{\Sigma}^{-1}$ is the variance-covariance matrix of $\widehat{\boldsymbol{\beta}}$ given by

$$\boldsymbol{\Sigma}^{-1} = Cov(\widehat{\boldsymbol{\beta}}) = E\left[(\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta})(\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta})^T\right] = \boldsymbol{\Sigma}^{-1}E[\boldsymbol{U}\boldsymbol{U}^T]\boldsymbol{\Sigma}^{-1} = \boldsymbol{\Sigma}^{-1}\boldsymbol{\Sigma}\boldsymbol{\Sigma}^{-1} = \boldsymbol{\Sigma}^{-1}$$

, $\boldsymbol{U}(\widehat{\boldsymbol{\beta}}) = \boldsymbol{U}(\boldsymbol{\beta}) - \boldsymbol{\Sigma}(\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}) = \boldsymbol{0}$ is the score vector in relation to $\boldsymbol{\beta}$ and $\boldsymbol{\Sigma}$ is the information matrix of Fisher.

With the vector of coefficients $\widehat{\boldsymbol{\beta}}$, we want to use its properties to somehow group the $\hat{\beta}_j$ that are similar to each other, through some method or technique. In this work we will try a lot of different grouping methods, with different approaches.
For instance, we can simply apply some univariate clustering technique over $\widehat{\boldsymbol{\beta}}$. This way we hope that each final cluster have $\hat{\beta}_j$ very close inside a cluster and distant to others $\hat{\beta}_j$ from other clusters.

Another option would be calculating the upper and lower bounds of confidence intervals to each $\hat{\beta}_j$, since we know its asymptotic distribution and its variance-covariance matrix, this way we will have a $n \times 2$ vector $\widehat{\boldsymbol{\beta}}_{\boldsymbol{2}}$, which have the lower bound of confidence intervals in the first column and the upper bound of confidence intervals in second column. Now we can apply some clustering method over $\widehat{\boldsymbol{\beta}}_{\boldsymbol{2}}$.

Still, since $\widehat{\boldsymbol{\beta}}$ is a mean vector, or a vector of expected values of the parameter vector $\boldsymbol{\beta}$, we can use some multiple comparisons approach such as those mentioned in Section 2, to group the means.

Note that $\widehat{\boldsymbol{\beta}}$ contains information about $\boldsymbol{x}$ in relation to $\boldsymbol{y}$, so with any of this grouping techniques, the resulted groups should have levels of $\boldsymbol{x}$ that shows similar behavior between themselves, related to $\boldsymbol{y}$.

4.2. ZERO INFLATED MODELS

When zero counts are very frequent in the response variable, Poisson distribution GLM models may not be well suited to this type of structure. Alternatives in the literature suggest the use of the Negative Binomial distribution, as it allows the dispersion parameter to be estimated independently of the location parameter. Still, if the mass point at "zero" is too high, models that incorporate a component to control this structure emerge as a better option.

Zero-inflated models for count data are two component mixture models combining a point mass at zero with a proper count distribution, like Poisson or Negative Binomial. The point of mass at zero is modeled by a binary component that captures the probability of zero in the response variable (Zeileis, Kleiber & Jackman, 2008). In the simplest case, which will be the one applied in this work, this component can have just the intercept.

The Zero inflated Poisson model is given by:

$$P(y_i|\boldsymbol{x}_i) = \begin{cases} \pi_i + (1-\pi_i) \times e^{-\theta_i} & , \quad y_i = 0 \\ (1-\pi_i) \times \dfrac{e^{-\theta_i} \times {\theta_i}^{y_i}}{y_i!} & , \quad y_i > 0 \end{cases}$$

Where π_i is possibility of existence of extra zeros and $\theta_i = e^{\boldsymbol{x}_i'\boldsymbol{\beta}}$ is the Poisson parameter. The expression for Negative Binomial zero inflated models can be found in Kim & Jun (2016). Zero inflated models are implemented in **R** in the package "zeroinfl", which will be used in the procedure proposed in this work.

4.3. MULTIPLE COMPARISONS

4.3.1. Scheffé Method

The Scheffé method was proposed by Henry Scheffé in 1953 to judge all contrasts in the Analysis of Variance. The problem of making further inferences about the contrasts arise when we reject the global F-test in a linear regression model, pointing that we have at least one of the means statistically different from the others, and it was discussed by various authors including Fisher (1935), Newman (1939) and Tukey (1951). One advantage of this method is that it does not require that the underlying populations of each group have same variance.

This method is implemented in **R** in the package "agricolae" and will be used in this work to compare it's grouping performance with the other ones that will be used. More details about Scheffé method can be found in Scheffé (1953).

4.3.2. Student-Newman-Keuls Method

The Student-Newman-Keuls (SNK) method is a sequential test designed to have more power than the Tukey method (Abdi & Williams, 2010). Both Tukey and Newman-Keuls tests uses a sampling distribution called Studentized Range, which is similar to the *t-student* distribution. The details of the method and how the tests are conducted can be found in Abdi & Williams (2010). One difference between this method and the Scheffé is that the SNK method tests sequentially just the adjacent means, while Scheffé method compare all pairwise of means.

This method is also implemented in R for linear models in package "agricolae", which will be used to be compared with the other methods that will be applied. The package "multcomp", which provides multiple comparisons for generalized linear models, can only handle number of factors that generates the maximum of 1000 multiple comparisons tests.

4.3.3. Clustering method

Scott, & Knott (1974) introduce their method by quoting Tukey: "At a low and practical level, what do we want to do? We wish to separate the varieties into distinguishable groups as often as we can without too frequently separating varieties which should stay together".

In some purposes, dividing the averages between roughly homogeneous groups of elements is enough for the analysis. Plackett suggested using cluster analysis to accomplish this goal. Scott & Knott, (1974), demonstrate that there is a relationship between the likelihood ratio test and the cluster analysis method, and use hierarchical univariate clusters to group means. The procedure proposed in this paper will use a hierarchical cluster to find the initial centroids to be used in a partitioned cluster method, the *k-means*. Details about the clustering method for multiple comparisons can be found in Scott & Knott, (1974).

4.4. CLUSTERING

Everitt & Hothorn (2011) define cluster analysis in their book as generic term of a wide range of numerical methods with the objective of split elements in separated groups that have homogeneous elements within it. There three types of clustering methods: *agglomerative hierarchical clustering, k-means clustering and model-based clustering*. In this work, we will use just the two first mentioned.

4.4.1. Agglomerative hierarchical clustering

In hierarchical clustering, Data is not split into a particular number of classes or groups at a single step, instead it consists in sequentially fusion of observations, from one cluster for each observation, to a single cluster with all of them. This imply that when the process fusion two observations, they cannot be found in different clusters in the following iterations. As many clustering methods, it is based on the distance between observations and the mostly used is the Euclidean distance, given by:

$$d_{ij} = \sqrt{\sum_{k=1}^{q} (x_{ik} - x_{jk})^2},$$

Where d_{ij} is the Euclidean distance between individual *i* with variables values $x_{i1}, x_{i2}, \dots, x_{iq}$ and individual j with variables values $x_{j1}, x_{j2}, \dots, x_{jq}$.

First step is defining a symmetric matrix of all distances between of pair of observations, and then start to fusion them with some criteria. After the first fusion, the concept of distance between the formed groups and the remaining observations or groups arises and it must be calculated. There are different techniques to do that and in this work, we will use the Ward's method, which uses the *group average* distance, defined by:

$$d_{AB} = \frac{1}{n_A n_B} \sum_{i \in A} \sum_{j \in B} d_{ij},$$

Where d_{AB} is the distance between clusters A and B, $n_A \; and \; n_B$ are the numbers of elements in A and B, respectively. Other distances and hierarchical clustering procedures can be found in Everitt & Hothorn (2011). Hierarchical clustering is implemented in base **R** and the *hclust* function will be used.

4.4.2. K-Means Clustering

K-means method of clustering seeks to split the data into predefined *k* number of clusters with predefined or random centroids. The *optimal level collapsing* proposed in this work will use the centroids defined by the hierarchical clustering described in previous section as starting centroids. The most commonly used criteria to find this groups are one that tries to find a partition that minimizes the *within-group sum of squares* (WGSS) over the variables, (Everitt & Hothorn, 2011). The WGSS is defined by:

$$WGSS = \sum_{j=1}^{q} \sum_{l=1}^{k} \sum_{i \in G_l} (x_{ij} - \bar{x}_j^{(l)})^2,$$

Where $\bar{x}_j^{(l)} = \frac{1}{n_i} \sum_{i \in G_i} x_{ij}$ is the mean of the individuals in group G_l on variable *j*.

The problem of find the best *k* is directly related with the objective of the procedure proposed in this work, which is to find a way to determine the best number of groups that resume a nominal variable and its information that it carries about a response variable.

4.5. MULTIDIMENSIONAL SCALING

Multidimensional scaling is a technique that aims to map a set of observations in relation to their variables based in some distance, commonly the Euclidean distance already presented in previous sub-section. With a matrix of distances between observations, it finds a set of *n* m-dimensional coordinates, each one representing an observation of the data. The coordinates are found minimizing some measure of fit between the distances, as WGSS in the k-means method.

The set of coordinates is found by the spectral decomposition of the matrix of distances D:

$$D = V\Lambda V'$$

Where $\Lambda = (\Lambda_1, \Lambda_2, \dots, \Lambda_n)$ is the diagonal matrix of eigenvalues of D and $V = (V_1, V_n, \dots, V_n)$ the corresponding matrix of eigenvectors, normalized so that the sum of squares of their elements is unity, that is, $V_i V_i' = 1$. The best two-dimensional solution, which will be the one used in this work, arises from the two highest eigenvalues as described in Everitt & Hothorn (2011).

4.6. REGULARIZATION APPROACHES FOR GENERALIZED LINEAR MODELS

Regularized estimation of regression parameters has been investigated thoroughly within the last decade. Methods for sparse modeling in the high-predictor case became available after the LASSO, proposed by Tibshirani (1996). Elastic-net, proposed by Zou & Hastie (2005), combines *ridge* regression with LASSO, doing variable selection.

Let $l(\boldsymbol{\beta})$ be the negative log-likelihood function, associated with the density function a distribution in its exponential family form, corresponding to the one presented in expression (1) in subsection 4.1. The penalized likelihood estimate has the general form:

$$\widehat{\boldsymbol{\beta}} = argmin_{\boldsymbol{\beta}}\{l(\boldsymbol{\beta}) + P_\gamma(\boldsymbol{\beta})\},$$

Where $P_\gamma(\boldsymbol{\beta})$ is the penalty term that regularizes the estimates. The ridge regression, proposed by Hoerl & Kennard in 1970, uses

$$P_\gamma^R(\boldsymbol{\beta}) = \gamma \sum_{j=1}^{P} \beta_j^2$$

As penalty term. This penalty usually has smaller prediction error than those found by the ordinary maximum likelihood (ML) estimates. Lasso uses

$$P_\gamma^L(\boldsymbol{\beta}) = \gamma \sum_{j=1}^{P} |\beta_j|.$$

And its penalty shrunk to zero the coefficients that's shows low influence on the response variable, so it can be used to do variable selection. Elastic-net penalty is given by

$$P_\gamma^{EN}(\boldsymbol{\beta}) = P_\gamma^R(\boldsymbol{\beta}) + P_\gamma^L(\boldsymbol{\beta}).$$

EN avoids the Lasso problem of not grouping predictors, as pointed by Zou & Hastie (2005).

More recently, alternatives to enforce the grouping properties of estimators have been proposed, like the Fused Lasso (FL), proposed by Tibshirani, Saunders, Rosset, Zhu & Knight (2005). An attractive feature of FL is that it penalizes the differences between the coefficients of adjacent predictors β_j and β_{j-1}. So, with the proper choice of the tuning parameter γ, adjacent predictors are grouped. The FL penalty combines the Lasso penalty, for variable selection, and the adjacent coefficients penalization term. So, it is given by:

$$P^{FL}_{\gamma_1,\gamma_2}(\boldsymbol{\beta}) = P^{L}_{\gamma_1}(\boldsymbol{\beta}) + \gamma_2 \sum_{j=2}^{k} |\beta_j - \beta_{j-1}|, \qquad \gamma_1, \gamma_2 \leq 0$$

In this work, the group property of the FL will be compared to the groups find by *optimal level collapsing* procedure. The FL penalty are implemented in **R** in package "lqa", which also provides the other penalties mentioned here and others. To compare its grouping property with the one proposed with *optimal level collapsing*, a simulated data will be used since the machine used to do the calculations have memory limitations in fitting the model.

4.7. OPTIMAL LEVEL COLLAPSING

The *optimal level collapsing* will consist in find the "best" number of groups that resume a nominal variable impact in a *generalized linear model*. It must be viewed as a transformation technique that uses the concepts of multiple comparisons and clustering approaches. To compare the performances of the different methods proposed, we will use different metrics for each type of model, accordingly with the nature of the distribution, like the Deviance, Adjusted R-Squared, AIC, AUC and the p-values from the Likelihood-Ratio comparing the model with all the original levels, and the model at iteration j=k-1 (since the first iteration will considerer *k*=2 groups) with k collapsed groups.

All the methods that will be described below follow the same logic, but the variables inputs to be used in clustering will be different. First, the model of any distribution (proper to the response variable, chosen by the user) is fitted with all the original levels. Then, we will take the vectors of its coefficients and standard errors and apply some clustering technique to try to regroup the levels in a way that it keeps in the same groups the levels with approximated same impact in the model. To evaluate where should be the best number of groups, an intensive method is applied fitting the model with a regrouped variable with *k* levels and storing the metrics of quality. With that we can compare all of the methods for a fixed *k*. Scatter plots will be used to illustrate their performance.

The clustering procedures will be done with the following steps:

- At iteration *k* a hierarchical cluster is applied in the vector that carry some information about the levels in the model and we cut the dendogram at *k*.
- The centroids provided by the hierarchical clustering is taken and passed to the *k-means* method.

We will try four different methods that will have different matrices that carry information about the variable levels to be clustered. It will be denominated as $\mathbf{O}_i$, with $i = 1,2,3,4$.

Each one of the methods and different matrices forms is described below.

4.7.1. Method 1

Method 1 is the simplest one. It is a vector just with the value of the estimated coefficients. So, the clustering procedure will be univariate. It has the form $\mathbf{O}_1' = (\hat{\beta}_1, \hat{\beta}_2, \dots, \hat{\beta}_q)$, where $\hat{\beta}_j$ is the value of the estimation for the coefficient associated with level *j*.

4.7.2. Method 2

Method 2 carry also the information about the standard error of the estimate, to try to control the levels that has high variability. It has the form $\mathbf{O}_2' = ((\hat{\beta}_1; \hat{\sigma}_1), (\hat{\beta}_2; \hat{\sigma}_2), \dots, (\hat{\beta}_q; \hat{\sigma}_q))$.

4.7.3. Method 3

Method 3 will use the variables X_1 and Y_2, provided from the multidimensional scaling. So, its vector will have the form $\mathbf{O}_3' = ((x_1, y_1), (x_2, y_2), \dots, (x_q, y_q))$.

4.7.4. Method 4

Method 4 uses the 95% confidence lower and upper bounds of the coefficients, but with the condition that if the p-value associated with the level is higher than 0,05, it is shrunken to 0. This condition is to avoid the problem with coefficients with higher standard errors, which will imply in extremely large intervals, rioting the clustering procedure. This also imply that, for method 4, basically a cluster is already settle in the start. One with values for lower and upper confidence bounds with value 0 within it. Vector for method 4 can be written as $\mathbf{O}_4' = ((L_1, U_1), (L_2, U_2), \dots, (L_q, U_q))$.

4.7.5. Method 5

Method 5 will use the coordinates variables from the multidimensional scaling of the values of the vector of method 2. Its vector is $\mathbf{O}_5' = ((x_1, y_1), (x_2, y_2), \dots, (x_q, y_q))$.

5. RESULTS

5.1. SIMULATION STUDY

To determine which of the methods has the best performance in grouping the levels of a nominal variable, they will be applied in a simulated database. The distribution chosen was Poisson, as it is the distribution used to model the response variable in the real database. The observations were generated in a similar way carried out in Smith *et al.,* 2011.

A nominal variable with 100 levels was created, with groups of 10 levels having the same parameter. We applied the *optimal level collapsing* with all the method described in section 4.7, the two multiple comparisons methods and the FL, and their concordance rate will be compared to determine which of the methods proposed for *optimal level collapsing* are the best, if it collapse correctly the levels of the variable and how good are the collapsing comparing with the multiple comparisons techniques and the FL.

For the observations generation, it was used the *rpois()* function from the R base package. It was chosen the following true parameter vector:

$$\boldsymbol{\beta}_{true} = (\underbrace{0.01,\dots,0.01}_{10},\underbrace{0.5,\dots,0.5}_{10},\underbrace{1,\dots,1}_{10},\underbrace{1.5,\dots,1.5}_{10},\underbrace{2,\dots,2}_{10},\underbrace{2.5,\dots,2.5}_{10},\underbrace{3,\dots,3}_{10},\underbrace{3.5,\dots,3.5}_{10},\underbrace{4,\dots,4}_{10},\underbrace{4.5,\dots,4.5}_{10})'$$

Based on $\boldsymbol{\beta}_{true}$, 200 observations were generated for each group β_j, with the associated probability function:

$$P(Y = y) = e^{(\beta_j \boldsymbol{x})^y} \times \frac{e^{-e^{(\beta_j \boldsymbol{x})}}}{y!}, \qquad y \geq 0$$

The seed used was 1235 and the example presented here can be replicated with the code that will be provided in the appendix. For the FL, first it was necessary to find the optimal tuning parameters. They were found via cross-validation and are 0.05 for the lasso penalty component and 0.07 for the grouping penalty component. The FL model found 11 unique coefficients in the estimation process. The two closest coefficients were grouped via clustering to have a number of groups equal to the actual pre-determined number of groups. Both Scheffé and SNK multiple comparisons methods found 10 groups in the linear model for the data generated and the Figure 5.1 shows how the methods from the *optimal level collapsing* behaved. Methods 1, 2, 3 and 5 had exactly the same performance in relation to the AIC reduction (respective to the model full model - with one coefficient per level). Method 4 had best performance for a low number of groups, but from 7 groups or more, the other methods were better. For the LR-test, method 4 kept different from the full model even with 10 groups, while the other methods get statistical equal to the full model with 9 groups, as it can be seen in Figure 5.2.

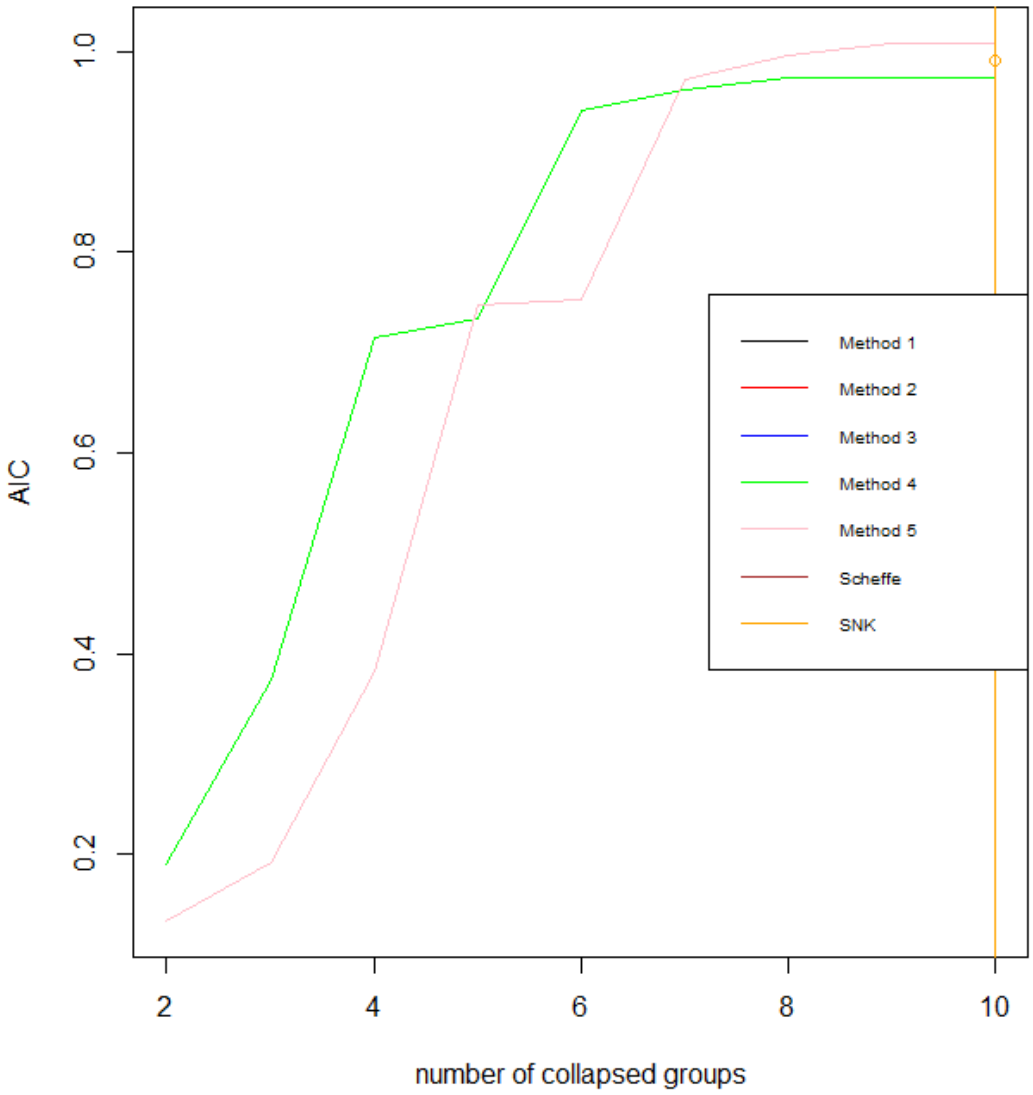

Figure 5.1 – Optimal level collapsing AIC evaluation for simulated dataset.

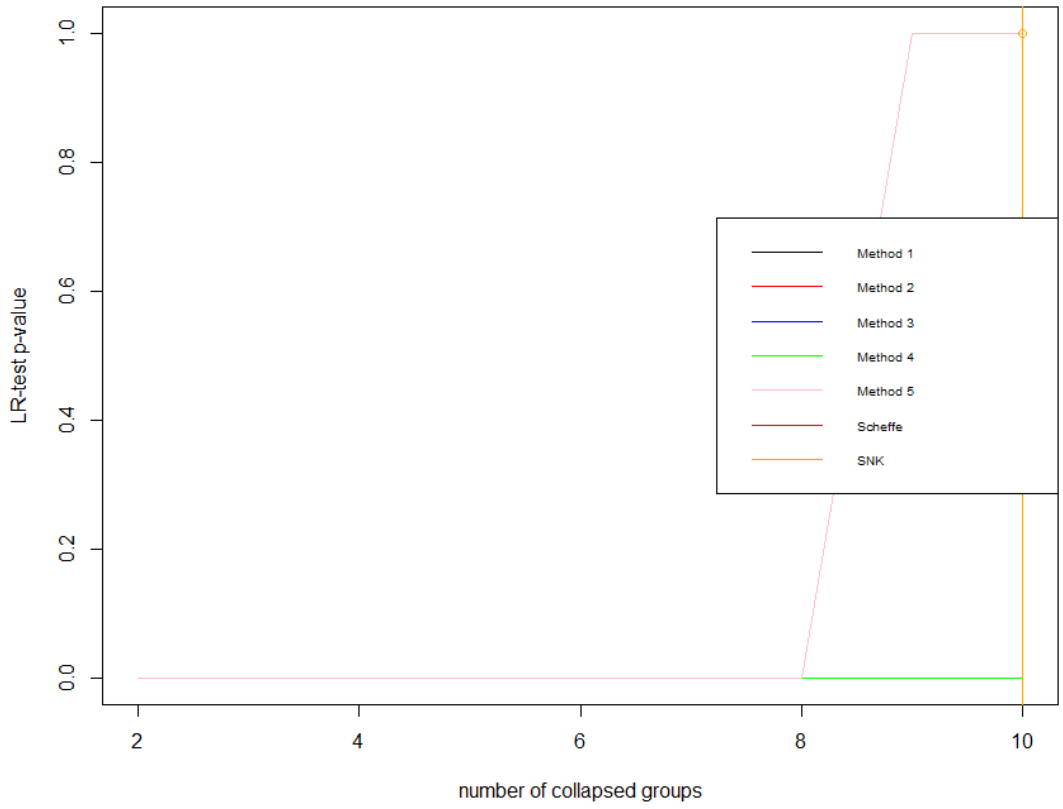

Figure 5.2 – Optimal level collapsing Likelihood Ratio Test evaluation for simulated dataset.

Table 5.1 shows the concordance rate among all grouping approaches and the real groups defined previously. Concordance rate for method 4 was not calculated because it found just 9 different groups, so we did not get a quadratic matrix. As it showed best performance than the other methods for low number of groups, we will keep it as one method offered for the *optimal level collapsing* procedure function, so the user can choose which is better for the case. The perfect concordance of the other methods shows that the *optimal level collapsing* in fact has power to group correctly the level of a nominal variable, and can be very useful as a easy way to transform nominal variables before use it in a modeling process, saving a lot of computational resources and degrees of freedom.

Table 5.1 – Concordance rate for the different grouping approaches

Method	Concordance rate
Olc – 1	100%
Olc – 2	100%
Olc – 3	100%
Olc – 4	NA
Olc – 5	100%
Scheffé	100%
SNK	100%

FL	100%

With that, *optimal level collapsing* procedure will offer 3 methods for grouping the levels of a nominal variable. The method 1, method 4 and the Scheffé method. User will be free to choose which one suits better in your problem.

Now, let's apply the *optimal level collapsing* in the real insurance dataset. We will evaluate the performance based on the comparison of the full model – with one coefficient for each level – and the model with some levels collapsed.

5.2. DESCRIPTIVE ANALYSIS

Descriptive analysis identified an skewed distribution on the right for the claim count, as expected. In Figure 5.3 this behavior can be verified. Only 12,64% of the analyzed policies presented some claim in the analyzed time period. The mean observed is of 0,1497 and the maximum number of claims is 99, which can be considered an outlier.

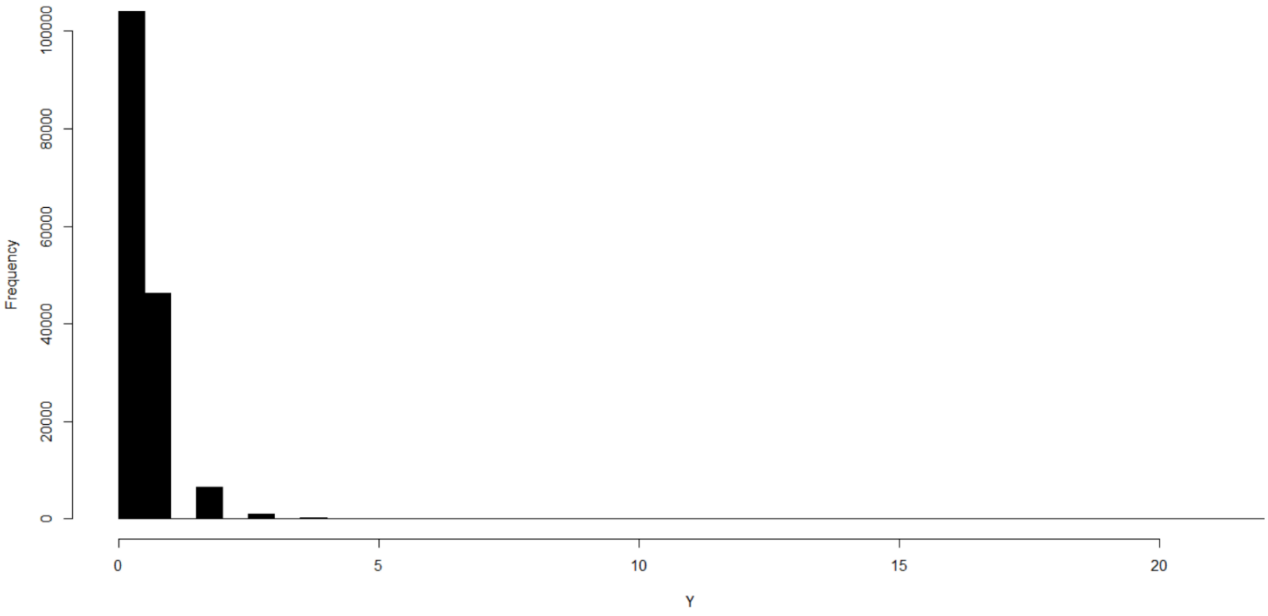

Figure 5.3 – Histogram of claim counts

In Figure 5.4, it is noted that it is extremely difficult to draw any conclusions regarding similar levels of Brand with regard to claim counting, first because all distributions are remarkably asymptotic on the right, second because the number of levels in each of the variables makes visualization difficult and third because the frequency of zeros is relatively high between levels, causing boxplots to report observations with counts greater than 0 as outliers on almost every level.

Tables 5.2 summarize information for the 10 highest and lowest average levels in the response variable (claim count) by Brand, as the number of observations at each level, in addition to the average, median, minimum, maximum, 1st and 3rd quartile of claims count. In the first, we can observe that both the highest and the lowest averages occur, in general, at levels that have a low number of observations. The outlier identified in the claim count analysis (Y = 99) belongs to the "Unidentified" level.

Table 5.2 – Table of descriptive statistics of variable Y by Brand (10 highest and lowest means)

Brand	Y_min	Y_1q	Y_mean	Y_median	Y_3q	Y_max	Y_sd	N
No identified	0	0	0.825175	0	0	99	8.278017	143
DACIA	0	0	0.4	0	0.75	2	0.699206	10
FABIA	0	0	0.333333	0	0.5	1	0.57735	3
GEO	0	0	0.333333	0	0.5	1	0.57735	3
AUTOBIANCHI	0	0	0.285714	0	0	2	0.61125	14
CADILLAC	0	0	0.25	0	0.25	1	0.5	4
HONDA	0	0	0.178705	0	0	9	0.491165	10397
MAZDA	0	0	0.174856	0	0	7	0.474726	3826
DODGE	0	0	0.172932	0	0	3	0.452456	133
PIAGGIO	0	0	0	0	0	0	0	5
PONTIAC	0	0	0	0	0	0	0	8
PORTARO	0	0	0	0	0	0	0	4
ROLLS-ROYCE	0	0	0	0	0	0	0	3
SANTANA	0	0	0	0	0	0	0	12
TALBOT	0	0	0	0	0	0	0	6
TATA	0	0	0	0	0	0	0	18
VAUXHALL	0	0	0	0	0	0	0	6
YARIS	0	0	0	0	0	0	0	4

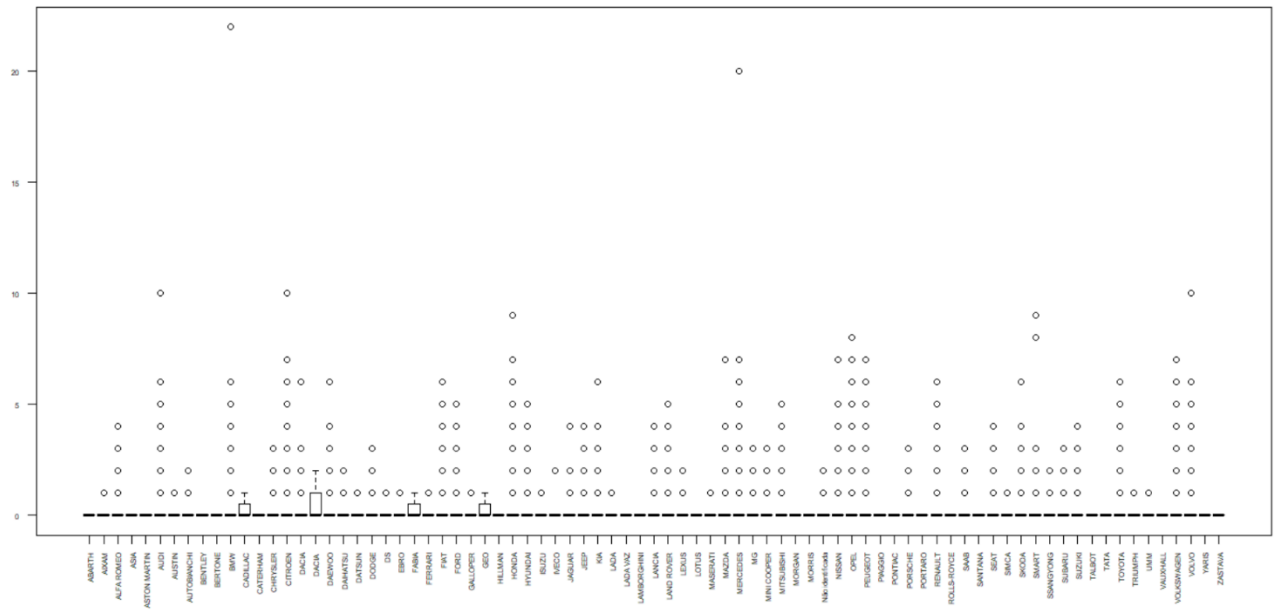

Figure 5.4 – Box-plots of claim counts by Brand

5.3. MODELING

The procedure proposed in this work will be illustrated with models of two different distributions that fit the response variable: Poisson and Negative Binomial. All these models are part of the exponential family, which enables us to use all of its properties mentioned in Chapter 3. We will first see if we can collapse the levels of Brand variable analyzing the AIC and LR-test p-value, as did in section 5.1. Then, we will fit a FL model using the same distribution mentioned, collapse the levels using Clustering and compare the concordance between the groups found. To make this comparison, we will use a sample of 25000 observations of the database, because it is the size that the *Iqa()* function can handle.

To exempt us from the problem of how to choose which level of variables to refer to (taken as 0), the models employed will not consider the intercept, so that all levels have an associated estimated parameter.

For the collapsing of coefficients, hierarchical clustering technique were used to find the initial points, specifically the "Ward" method, and then passed to the K-means clustering.

So first, we estimate two models, both using only Brand as covariate, changing the probability distribution. We have the two following models:

$Model\ 1\colon\quad Y \sim Brand + \varepsilon$; with Poisson distribution

$Model\ 2\colon\quad Y \sim Brand + \varepsilon$; with Negative Binomial distribution

With Table 5.3 it is possible to see that both variables have significant levels, ie, by the Wald test, the estimated parameters are statistically different from 0. The significance of each of the variables can also be observed in the reduction of the deviance in relation to the model only with the intercept (Null deviance of 760092). The estimated coefficients for Model 1 using Poisson distribution range from approximately -14 to -0.19. For Model 2, they range from approximately -4 to -0.32.

Table 5.3 – Coefficients table output of Model 1 with Poission distribution.

Coefficients	Estimate	Std. Error	Z-value	P-value
ABARTH	-13.30	234.66	-0.05	0.95
AIXAM	-2.07	1.00	-2.07	0.03
ALFA ROMEO	-1.96	0.05	-37.03	< 0.00
ASIA	-13.30	270.96	-0.04	0.96
ASTON MARTIN	-13.30	102.41	-0.13	0.89
AUDI	-1.90	0.01	-101.62	< 0.00
AUSTIN	-3.55	0.70	-5.02	< 0.00
AUTOBIANCHI	-1.25	0.50	-2.50	0.01
BENTLEY	-13.30	92.04	-0.14	0.88
BERTOME	-13.30	270.96	-0.04	0.96
BMW	-1.97	0.01	-107.56	< 0.00
CADILLAC	-1.38	1.00	-1.38	0.16
CATERHAM	-13.30	209.88	-0.06	0.94
CHRYSLER	-1.98	0.11	-17.41	< 0.00
CITROEN	-1.82	0.01	-111.53	< 0.00
DACIA	-1.88	0.07	-25.13	< 0.00
DAEWOO	-0.91	0.50	-1.83	0.06
DAIHATSU	-1.95	0.07	-25.05	< 0.00

Looking more closely at the estimated coefficients of the Brand variable, we have 10 estimated unique values (using 5 decimal places), that is, we can immediately conclude that some of the levels of this variable can be grouped, as they are literally equal. As we will use hierarchical clustering methods, it is not necessary to pre-select the number of clusters to be employed, allowing us to explore and analyze the impact of each possible number of clusters on the model. The automatic choice of the number of clusters to be implemented in the proposed procedure may use metrics such as deviance, AIC, adjusted R-square, ROC curve AUC and p-values from the Likelihood-Ratio test.

For the illustration of the procedure in this work, only the deviance metric was used, which clearly allowed an indication of the "optimal" number of collapsed levels.

Figure 5.5 presents the AIC for the Poisson model using different numbers of grouped levels for the Brand variable, ranging from 2 to the number of unique estimated coefficients (10). It can be said that one optimal number of clusters is identified by the curve elbow in the graph, which is between 6 clusters. The 2 groups formed by the Scheffé and SNK multiple comparisons had worst performance then method 1 (for a fixed number of 2 groups). For the p-values of the Likelihood ratio-test, the method 1 also shows better results. It achieves no statistically difference with the full model way faster than method 2. With 6 groups method 1 has a p-value = 1, suggesting that is a good number to regroup the original levels, like what was observed on Figure 5.5.

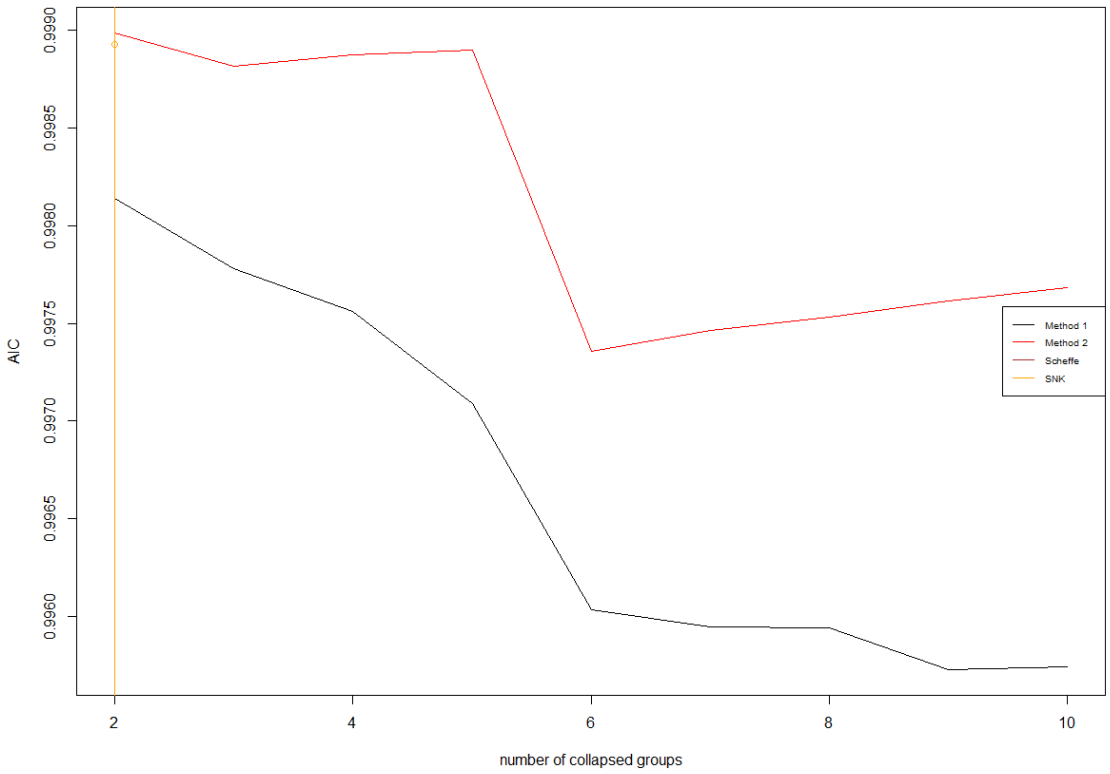

Figure 5.5 – AIC of Model 1 with Poisson distribution by number of collapsed levels.

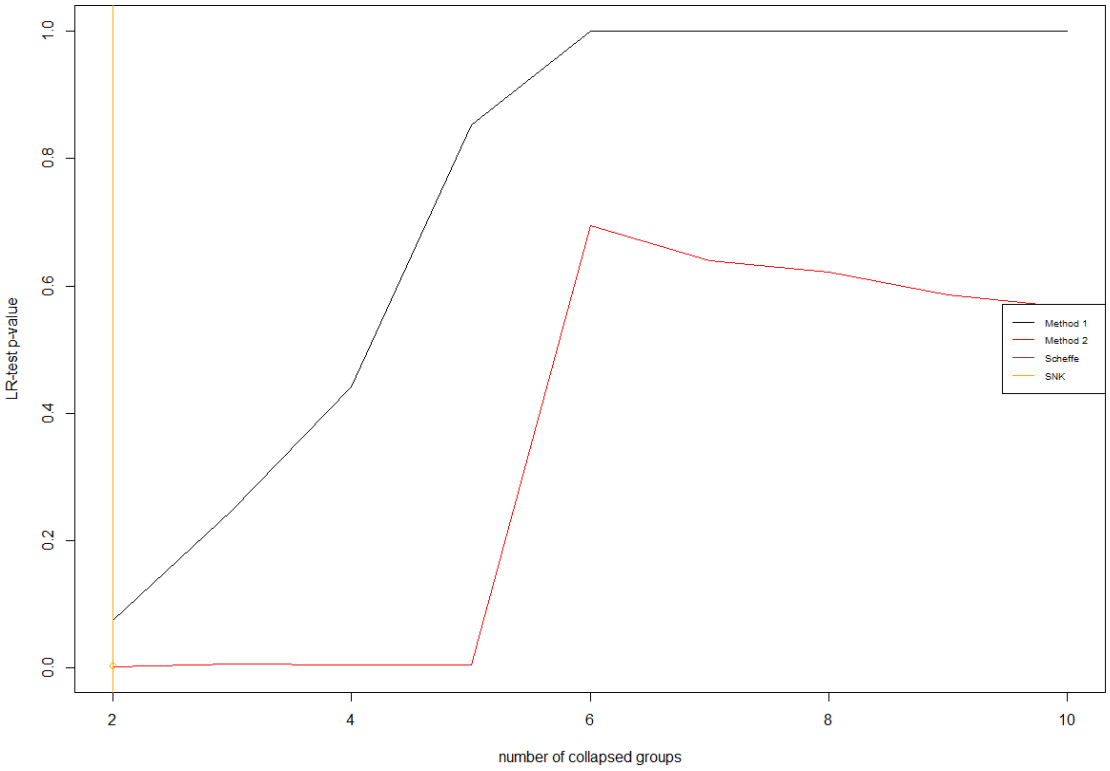

Figure 5.6 – LR-test p-values comparing Model 1 with Models by number of collapsed levels.

Table 5.4 presents some of the estimated coefficients for Model 1 using the Negative Binomial distribution. Once again there were levels with statistically associated coefficients other than 0. The distribution of the residuals indicates that this model is possibly more appropriate for this data. Following the same reasoning used for Poisson distribution models, we will determine the optimal number of groupings for the Brand variable based on its impact on model quality metrics. For the Negative Binomial distribution, AIC (Akaike Information Criteria) was chosen.

Table 5.4 – Coefficients table output of Model 2 with Negative Binomial distribution.

Coefficients	Estimate	Std. Error	Z-value	P-value
ABARTH	-18.30	2858.76	-0.00	0.99
AIXAM	-2.07	1.09	-1.90	0.05
ALFA ROMEO	-1.96	0.05	-33.62	< 0.00
ASIA	-18.30	3301.01	-0.04	0.99
ASTON MARTIN	-18.30	1247.66	-0.01	0.98
AUDI	-1.90	0.02	-91.76	< 0.00
AUSTIN	-3.55	0.72	-4.92	< 0.00
AUTOBIANCHI	-1.25	0.59	-2.09	0.03
BENTLEY	-18.30	1121.30	-0.01	0.98
BERTOME	-18.30	3301.01	-0.00	0.99

BMW	-1.97	0.01	-97.71	< 0.00
CADILLAC	-1.38	1.00	-1.18	0.23
CATERHAM	-18.30	209.88	-0.00	0.99

The graph of the AIC's in Figure 5.7 by the olc using the Negative Binomial model have similar results when comparing to Model 1. Methods 1 and 2 shown global optimal around 5 and 6 number of groups. As well, in the plot with the p-values for the Likelihood-Ratio test, in Figure 5.8, we also observe that method 1 was faster to achieve no statistically difference with the full model.

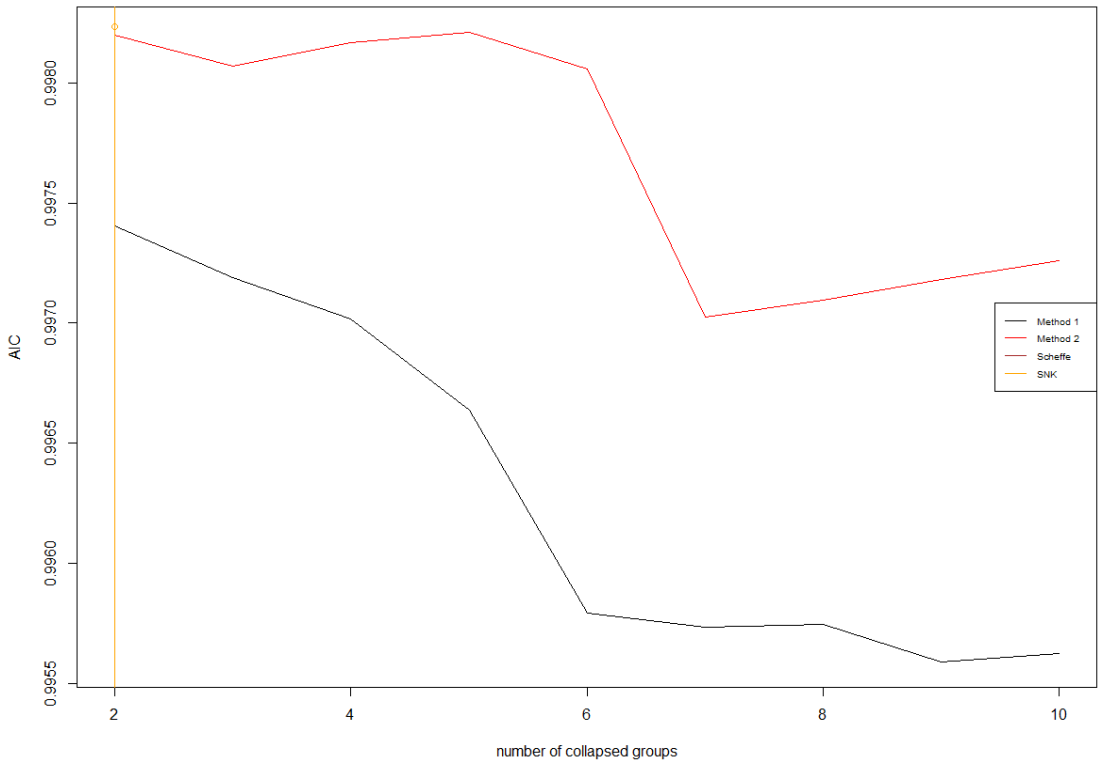

Figure 5.7 – AIC for Model 2 by number of collapsed levels

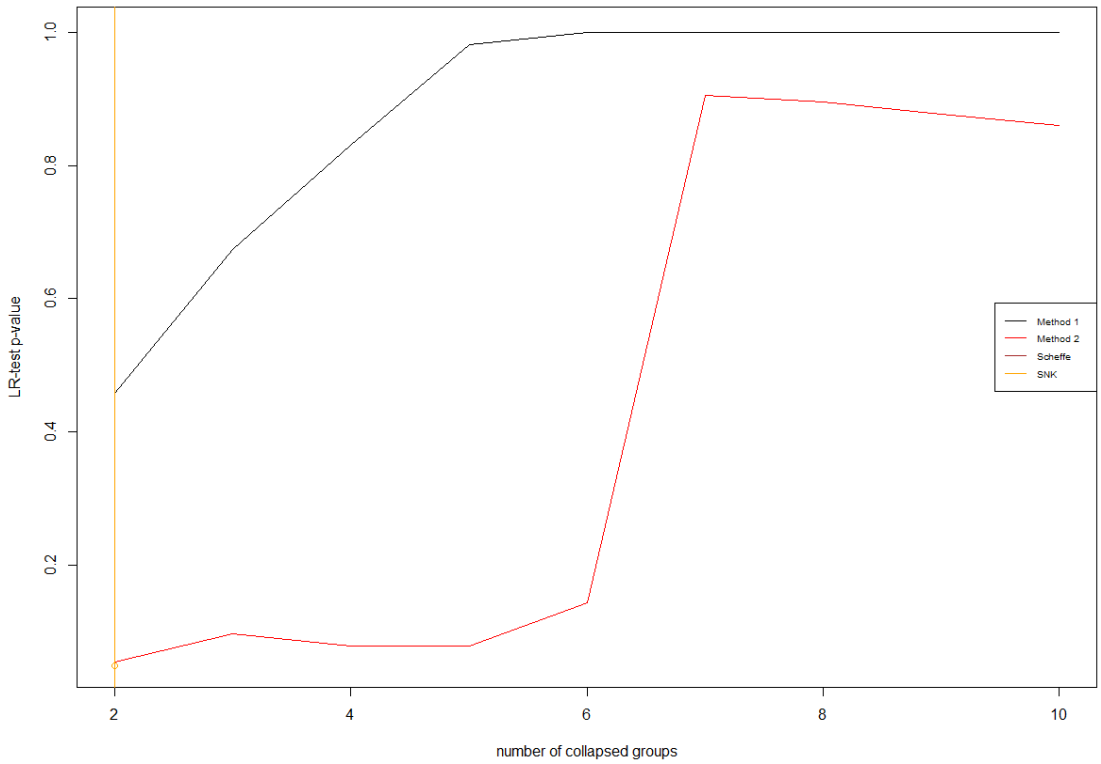

Figure 5.8 – LR-test p-values for Model 2 by number of collapsed levels

Now, to compare the groups found between the olc procedure and the FL model, we will fix a number of 6 groups and calculate the concordance rate in the matrix, similar to what was did in simulation study in section 5.1, regarding that we don't know the true parameters.

Tables 5.5 and 5.6 shows that method 1 presented the highest concordance rate with the FL model, and a concordance rate that can be seen as good, since the computation effort needed in olc procedure is way lower than the FL model (I was able to use the whole dataset in olc procedure, while the FL model using *lqa()* function required more than 10x the RAM that my machine have. The most important thing in olc, besides the lower computational effort, is that it can collapse levels of a categorical variable in a way making sense and the job of working with categorical variables easier.

Table 5.5 – Concordance rate between groups found in the FL model and olc methods with Poisson distribution.

Method	Concordance rate
1	72,13%
2	57,37%

Table 5.6 – Concordance rate between groups found in the FL model and olc methods with Negative-Binomial distribution

Method	Concordance rate
1	65,57%
2	47,54%

6. CONCLUSIONS

So far, *optimal level collapsing* show that it can collapse levels of a nominal variable properly, and the behavior of the quality metrics used to evaluate the impact of the different number of groups show convergence among the models used in relation to the global optimal as well the performance of the different coefficients clustering methods proposed.

Through the study with simulated data presented in section 5.1, it was possible to observe that methods 1 and 3 performed better in the process of finding groups with equal parameters, both evaluating the AIC of the models and the p-value of the LR-test. In addition, except for method 4, in which it was not possible to define 10 different groups through clustering, all other proposed methods showed 100% agreement with the real groups (defined at the time of generating the data), together with the FL, Scheffé and SNK. With the results obtained in section 5.1, it was defined that the functions developed from the "olc" package will only include methods 1 and 4 among those presented in section 4.7, since methods 1 and 3 showed identical results and method 4 by the possibility of bringing better results on some occasions. In addition to these two methods, which use the clustering procedure to find groups, the package also includes the multiple comparison methods Scheffé and SNK. With this, the methods originally named 1 and 4 in section 4.7 are now referred to as methods 1 and 2, respectively, in the package functions.

In the application with real data presented in section 5.3, it was possible to observe that on both metrics used to evaluate the regrouping, AIC and p-value of the LR-test, method 1 shows better performance. While method 1 presented mostly lower AIC in the groups found than method 2, it also achieves none statistically difference with the full model way faster. When comparing the groups found by these methods with the groups found using the penalized regression technique FL, we obtained a greater agreement using method 1 both in the Poission regression model and in the Negative Binomial regression model. As we do not know the actual parameters associated with each of the categories of the analyzed covariate, it is not possible to say which groups are closer to the optimum.

As mentioned in section 5.3, we had to use a sample of the dataset because the machine used in this work does not have the memory RAM required to fit the FL model, giving a possible situation of why you could use olc.

The *olc* package was developed with the functions *olc(), olc.eval(), olc_AIC_plot* and *olc_LR_plot,* and it can be found in Github platform, in the following address: https://github.com/acaciomattos/olc/. The package is available for everyone to use it and improve it.

7. REFERENCES

Abdi, H., & Williams, L. J. (2010). Newman-Keuls test and Tukey test. *Encyclopedia of research design*, 1-11.

Bondell, H. D., & Reich, B. J. (2009). Simultaneous factor selection and collapsing levels in ANOVA. *Biometrics, 65*(1), 169-177.

Everitt, B., & Hothorn, T. (2011). An introduction to applied multivariate analysis with R. *Springer Science & Business Media*.

Fisher, R. A. (1935). *The design of experiments*, Oliver and Boyd, London, 244 pp.

Kim, J. M., & Jun, S. (2016). Zero-inflated poisson and negative binomial regressions for technology analysis. *International Journal of Software Engineering and Its Applications*, 10(12), 431-448.

Montgomery, D. C. (2013). Design and analysis of experiments. John wiley & sons.

Müller, M. (2012). Generalized linear models. *In Handbook of Computational Statistics* (pp. 681-709). Springer, Berlin, Heidelberg.

Nelder, J. A., & Wedderburn, R. W. (1972). Generalized linear models. *Journal of the Royal Statistical Society: Series A (General), 135*(3), 370-384.

Newman, D. (1939). The distribution of range in samples from a normal population, expressed in terms of an independent estimate of standard deviation. *Biometrika*, *31*(1/2), 20-30.

Petry, S., Flexeder, C., & Tutz, G. (2011). Parwise Fused Lasso. Department of Statistics, University of Munich: *Technical Reports*, No. 102, https://doi.org/10.5282/ubm/epub.12164

Scheffé, H. (1953). A method for judging all contrasts in the analysis of variance. *Biometrika, 40*(1-2), 87-110.

Scott, A. J., & Knott, M. (1974). A cluster analysis method for grouping means in the analysis of variance. *Biometrics*, 507-512.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, *58*(1), 267-288.

Tibshirani, R., Saunders, M., Rosset, S., Zhu, J., & Knight, K. (2005). Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *67*(1), 91-108.

Tukey, J. W. (1949). Comparing individual means in the analysis of variance. *Biometrics, 5*(2), 99-114.

Tukey, J. W. (1951). Quick and dirty methods in statistics. Part II. Simple analyses for standard designs. *American Society for Quality Control*, 189-197.

Zeileis, A., Kleiber, C., & Jackman, S. (2008). Regression models for count data in R. *Journal of statistical software*, *27*(8), 1-25.

Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, *67*(2), 301-320.

8. APPENDIX

8.1. SIMULATION STUDY CODE

```
######## Simulation Study

### Creating vector with simulated levels

x <- rep(1:100,200)

x <- as.factor(x)

### Vector of unique parameters

betas <- c(0.01,0.5,1,1.5,2,2.5,3,3.5,4,4.5)

### Vector of unique lambdas

lambdas <- exp(betas)

### Vector with response variable based on a Poisson distribution with parameters previously defined

set.seed(1235)

y <- ifelse(x%in%as.character(1:10),rpois(1,lambdas[1]),
        ifelse(x%in%as.character(11:20),rpois(1,lambdas[2]),
            ifelse(x%in%as.character(21:30),rpois(1,lambdas[3]),
                ifelse(x%in%as.character(31:40),rpois(1,lambdas[4]),
                    ifelse(x%in%as.character(41:50),rpois(1,lambdas[5]),
                        ifelse(x%in%as.character(51:60),rpois(1,lambdas[6]),
                            ifelse(x%in%as.character(61:70),rpois(1,lambdas[7]),
                                ifelse(x%in%as.character(71:80),rpois(1,lambdas[8]),

ifelse(x%in%as.character(81:90),rpois(1,lambdas[9]),rpois(1,lambdas[10]))))))))))

### Creating the regression matrix

XX <- model.matrix(y~-1+x)

### Fitting fused lasso with Poission distribution
```

```
mod_fused.lasso <- lqa(y ~ -1+XX, family = poisson (),
                       penalty = fused.lasso (c (0.05, 0.07)))
### Taking groups of 10 levels by fused lasso
betas_FL <- round(mod_fused.lasso$coefficients,3)
length(unique(betas_FL))
clust_FL  <- hclust(dist(betas_FL),method="ward.D2")
groups_FL <- cutree((clust_FL),k=10)
### Using olc to fund best number of collapsed levels with maximum number of groups = 10
teste <- olc.eval(YY=y,Levels = x, model="poisson",k.max=10)
olc_AIC_plot(teste)
olc_LR_plot(teste)
### Creating new levels with k = 10
teste2 <- olc(YY=y,Levels = x, model="poisson",k=10)
teste2$variable_aux$true <- sort(rep(1:10,10))
### Comparing the groups between fused lasso, olc method 1, 2 and SNK multiple comparisons
groups
table(teste2$variable_aux$olc.met1.10.Levels,teste2$variable_aux$true)
table(teste2$variable_aux$olc.met2.10.Levels,teste2$variable_aux$true)
table(teste$snk$Groups$Grupo_snk,teste2$variable_aux$true)
```

8.2. APPLICATION CODE

```
### Installing olc package and loading other packages needed
require(devtools)
install_github("https://github.com/acaciomattos/olc")
require(olc)
require(data.table)
require(lqa)
## Loading dataset
load("C:\\Users\\User\\Desktop\\Mestrado Acacio\\Dados_tratados_vs2.RData")
### Avoiding scientific notation
options(scipen=999)
#### Descriptive analysis ####
summary(Dados_tese$Y)
table(Dados_tese$Y>0)/nrow(Dados_tese)
Marcas <- Dados_tese %>% group_by(Marca) %>%
  summarise(Y_min=min(Y),
            Y_1q=quantile(Y,0.25),
            Y_mean=mean(Y),
            Y_median=median(Y),
            Y_3q=quantile(Y,0.75),
            Y_max=max(Y),
            Y_sd=sd(Y),
            N=n()) %>%
  arrange(desc(Y_mean))
export(Marcas,"tabela_marcas.xlsx")
```

```
hist(Dados_tese$Y[Dados_tese$Y<90],breaks = 35,ylim=c(0,100000),col="black",xlab="Y",
    main="Histogram of Y (count of claims by policy)")
boxplot(Dados_tese$Y[Dados_tese$Y<90]~Dados_tese$Marca[Dados_tese$Y<90],
      main="Box-plots of Y (count of claims by policy) by Brand",cex.axis=0.5,las=2)
# Taking a sample
set.seed(123)
Base <- copy(data.table(Dados_tese)[sample(1:.N, 25000, replace = F)])
### Fitting fused lasso model with Poisson distribution
mod_fused.lasso_poi <- lqa(Y ~ -1+Marca, family = poisson(), data = Base,
                           penalty = fused.lasso (c (0.05, 0.07)))
### Creating groups of fused lasso levels with k = 6
betas_FL.poi <- round(mod_fused.lasso_poi$coefficients,3)
length(unique(betas_FL.poi))
clust_FL.poi  <- hclust(dist(betas_FL.poi),method="ward.D2")
groups_FL.poi <- cutree((clust_FL.poi),k=6)
### Using olc to find groups of levels with Poission distribution and maximum groups of 10.
olc.eval.poi <- olc.eval(YY=Base$Y,Levels = Base$Marca, model="poisson",k.max=10)
olc_AIC_plot(olc.eval.poi)
olc_LR_plot(olc.eval.poi)
### Create new callapsed levels with k = 6
olc.poi <- olc(YY=Base$Y,Levels = Base$Marca, model="poisson",k=6)
### Taking the groups by olc
groups_olc.poi1 <- olc.poi$variable_aux$olc.met1.6.Levels
groups_olc.poi2 <- olc.poi$variable_aux$olc.met2.6.Levels
### Comparing groups by olc with groups by fused lasso
table(groups_FL.poi,groups_olc.poi1)
table(groups_FL.poi,groups_olc.poi2)
```

```
### Calculating concordance rate between groups by fused lasso and groups by olc method 1 and 2
sum(diag(table(groups_FL.poi,groups_olc.poi1)[c(6,4,3,2,1,5), c(1,4,2,5,6,3)]))/61 # 72.13%
sum(diag(table(groups_FL.poi,groups_olc.poi2)[c(6,5,4,3,2,1), c(6,3,4,1,5,2)]))/61 # 57.37%%
### Fitting fused lasso using Negative Binomial distribution
mod_fused.lasso_nb <- lqa(Y ~ -1+Marca, family = negative.binomial(theta = 1), data = Base,
                          penalty = fused.lasso (c (0.05, 0.07)))
### Taking groups of fused lasso with number of groups = 6
betas_FL.nb <- round(mod_fused.lasso_nb$coefficients,3)
length(unique(betas_FL.nb))
clust_FL.nb  <- hclust(dist(betas_FL.nb),method="ward.D2")
groups_FL.nb <- cutree((clust_FL.nb),k=6)
### Using olc to find groups of levels with Negative Binomial distribution and maximum groups of 10.
olc.eval.nb <- olc.eval(YY=Base$Y,Levels = Base$Marca, model="nb",k.max=10)
olc_AIC_plot(olc.eval.nb)
olc_LR_plot(olc.eval.nb)
### Creating new collapsed levels with k = 6
olc.nb <- olc(YY=Base$Y,Levels = Base$Marca, model="nb",k=6)
### Taking the groups
groups_olc.nb1 <- olc.nb$variable_aux$olc.met1.6.Levels
groups_olc.nb2 <- olc.nb$variable_aux$olc.met2.6.Levels
### Comparing groups by fused lasso and olc method 1 and method 2 and calculating concordance
rate
table(groups_FL.nb,groups_olc.nb1)
(10+1+3+16+7+3)/61 # 65.57%
```

```
table(groups_FL.nb,groups_olc.nb2)

(10+13+2+3+1)/61 # 47.54 %
```