Categorical Encoding for Machine Learning

Quantificazione delle variabili qualitative per il Machine Learning

Agostino Di Ciaccio

Abstract In recent years, interest has grown in addressing the problem of encoding categorical variables, especially in deep learning applied to big-data. However, the current proposals are not entirely satisfactory. The aim of this work is to show the logic and advantages of a new encoding method that takes its cue from the recent word embedding proposals and which we have called Categorical Embedding. Both a supervised and an unsupervised approach will be considered.

Abstract *Negli ultimi anni è cresciuto l'interesse nell'affrontare il problema della quantificazione delle variabili qualitative soprattutto nel deep learning applicato a grandi insiemi di dati. Le soluzioni proposte non sono però del tutto soddisfacenti. Obiettivo di questo lavoro è mostrare la logica e i vantaggi di un nuovo metodo di codifica che, prendendo spunto dalle recenti proposte di word embedding, abbiamo chiamato Categorical Embedding. Sarà considerato sia un approccio supervisionato che non-supervisionato.*

Key words: categorical encoding, deep learning, word embedding

1 Introduction

Usually, Big-Data include tens or hundreds of variables, which have mixed measurement levels with many categorical variables, sometimes with high cardinality. The treatment of many categorical variables, especially when combined with quantitative variables, is a complex topic that has no easy solutions. The problem has been considered in many areas of classical statistics (see for example Azzalini 2001)

This theme is particularly relevant in machine learning applied to large datasets. In fact, the only method that can naturally handle many variables with a mixed

[1] Agostino Di Ciaccio, University of Rome "La Sapienza"; agostino.diciaccio@uniroma1.it

measurement level is the decision tree (although software often does not take this potential into account).

The purpose of this work is to show the logic and the advantages of a new encoding method that takes its cue from the recent proposals of *word embedding* in Natural Language Processing (NLP) (Bengio et al. 2003) and which we call *categorical embedding*. Some applications to real datasets will show the interest of our proposal.

2 Encoding Categorical variables

Applying neural networks to categorical data requires some form of encoding. Perhaps the most used method is *one-hot* encoding, i.e., for each category, adding a new binary feature indicating it. However, in the case of high cardinality variables, such technique leads to a large number of new features. Moreover, the new variables are perfectly independent, and this is unrealistic. The categories can have relationships and similarities that could be extracted from the context. An example is the variable "*day of the week*" which is a cyclic ordinal feature. If we represent the days of the week through 7 one-hot vectors (or at least 6, but in Machine Learning it is not necessary to drop one dummy) we obtain a spatial representation of the variable in 7 dimensions in which every pair of categories is at Euclidean distance $\sqrt{2}$ from each other. This representation is not meaningful: it could be more coherent to represent the days of the week on a circumference in a smaller two-dimensional space.

On the other hand, even a 'circular' representation does not take into account that the days of the week can be distinguished between 'working' and 'non-working' and this distinction is often more relevant for the analysis. In table 1 we have reported the distances between the days of the week obtained by considering the encodings given by the Glove's word-vectors (Pennington et al. 2014) obtained by analysing millions of documents by an NLP model. If our analysis concerns, e.g., the sales forecast of a supermarket or the level of particulates in the air of a great city, this is certainly a coherent coding.

Table 1: Distances between the days of the week in the 50-dimensional Glove word-vectors

	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Monday	0.002	0.004	0.002	0.007	**0.086**	**0.074**
Tuesday		0.002	0.002	0.009	**0.087**	**0.077**
Wednesday			0.003	0.008	**0.079**	**0.069**
Thursday				0.007	**0.085**	**0.074**
Friday					**0.072**	**0.062**
Saturday						0.009

It is therefore not obvious that the 'natural' order of the categories should be kept in our coding. If we want to maintain the natural order, we could use the matrix of "*hot-vectors of order*" as suggested in the Optimal Scaling approach (Gifi 1981, Di Ciaccio 1988).

In general, also categorical variables with high cardinality can have a satisfactory representation in a small space, correctly representing their relationship. Consider, for

example, the 102 Italian provinces: the one-hot encoding would provide a representation in a big dimensional space while we know that a representation in a two-dimensional space, using for example the geographical position of the principal city, could be sufficient for our analysis. In general, there is no encoding which is optimal independently of the objective of the analysis and the model applied. The biggest distinction between the encoding methods is based on the approach that can be supervised or unsupervised.

An old but interesting proposal is Optimal Scaling (Gifi 1981) which can generate quantifications both in a supervised and unsupervised approach. In an unsupervised approach, two well-known equivalent methods can be derived: Homogeneity Analysis (HA, Gifi 1981) and Multiple Correspondence Analysis (MCA, Benzecri 1973). By these methods, the categories are 'optimally' encoded maximizing the eigenvalues of the correlation matrix. In the French approach the problem is solved analytically, while in the Gifi approach the problem is solved numerically. This numerical variant offers a large amount of flexibility.

Let m the number of categorical variables and j the index of the generic variable ($j=1,2,\ldots,m$), k_j the number of categories of the j-*th* variable, $\mathbf{G}_j = \left[\mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_{k_j}\right]$ the indicator matrix of dimension $n \times k_j$. Let p be the number of dimensions, which needs to be fixed a priori, and r the generic dimension ($r=1,2,\ldots,p$). Each variable can be associated with a matrix $\mathbf{C}_j$ of dimension $k_j \times p$ containing the *category quantifications*. The quantification for the j-*th* variable on the r-*th* dimension is given by

$$\mathbf{v}_{jr} = \mathbf{G}_j \mathbf{c}_{jr} = \sum_{h=1}^{k_j} c_{jhr} \mathbf{g}_{jh} \tag{1}$$

Hence, the vector of the quantified data is a linear combination of the indicator variables and the set of possible quantifications defines a subspace R^{k_j}. In fact, the quantification $\mathbf{v}_{jr}$ is a linear combination of an orthogonal base of R^{k_j}. If the variable is ordered, the constraints on the quantifications define a polyhedric convex cone (Gifi 1981) and a similar approach based on b-splines can manage also quantitative data (Di Ciaccio 1990). Define the matrix $\mathbf{X}$ containing the so-called *object scores* with dimension $n \times p$, to obtain the scores and the quantifications, we can minimize the following equation by an *alternating least squares* algorithm:

$$\sigma(\mathbf{X}, \mathbf{C}_1, \ldots, \mathbf{C}_m) = \sum_{j=1}^{m} tr(\mathbf{X} - \mathbf{G}_j\mathbf{C}_j)'(\mathbf{X} - \mathbf{G}_j\mathbf{C}_j) = \sum_{j=1}^{m} \left\|\mathbf{X} - \mathbf{G}_j\mathbf{C}_j\right\|^2 \tag{2}$$

subject to the normalization constraints $\mathbf{X}'\mathbf{X}=n\mathbf{I}$ and $\mathbf{J}\mathbf{X}=\mathbf{X}$, where $\mathbf{J}$ is the projector on the subspace orthogonal to the unit vector.

In a classical, supervised, regression model we want to predict a quantitative response variable Y using m predictor variables. If the predictor variables are categorical, we can use the expression (1), with $p=1$, to quantify the variables obtaining the loss function:

$$\left\|\mathbf{y} - \sum_{j=1}^{m} \boldsymbol{\beta}_j \mathbf{G}_j \mathbf{c}_j\right\|^2 = min \tag{3}$$

Assuming $\varphi(\mathbf{y}) = \mathbf{y}$, with the appropriate constraints, this loss function corresponds to the Morals model (Gifi 1981) and the solution can be identified through an alternating least square procedure, which is sometimes called backfitting.

Morals calculates a single quantification of each categorical variable. A way to extend (3) to consider p quantifications is:

$$\left\|\mathbf{y} - \sum_{r=1}^{p}\sum_{j=1}^{m}\boldsymbol{\beta}_{rj}\mathbf{G}_j\mathbf{c}_{rj}\right\|^2 = min \quad (4)$$

Other encoding methods for ML have been proposed in the literature (see for example Potdar et al. 2017) mainly for the supervised case. The *Scikit-learn* software library allows to apply 15 different methods.

3 Categorical encoding

In this paper we propose a method to encode categorical variables that uses dense matrices of reduced size through an '*embedding*' of the categories in a low dimension space. A well-known form of embedding is *word-embedding* (Bengio et al. 2003). Embedding in NLP is a vector representation of the words in such a way that the words that frequently appear in similar contexts are close to each other.

With the most used software libraries for Neural Networks (e.g. Tensorflow, www.tensorflow.org) it is now available the '*Embedding layer*' for NLP. This layer transforms a sequence of words into their vectorial representation introducing an array of quantification parameters. This operation can be defined as in (1) in which the one-hot vectors identify the different words in a vocabulary and the vectors of parameters $\mathbf{c}_r$ are the corresponding vectorial representations in a p-dimensional space. The *Embedding layer* is the first layer in a predictive model and the parameters are determined by backpropagation, trying to maximize the fit of the model to the target.

This layer performs the same kind of transformation as the Optimal Scaling. The difference lies in the objective function and in the method of estimating the parameters given by the *gradient descent*. Let $t = \sum_{j=1}^{m} k_j$, using the approach of (4), it is possible to search for a vectorial representation of the t categories inside a p-dimensional space to optimize the prediction of a target Y. The embedding of categories was considered also by two previous works (Guo et al. 2016, Stefanini 2020) in which the authors proposed the '*entities embedding'*. This approach consists of a distinct embedding phase for each categorical variable where the encoding is defined as in (1), each feature embedding is optimized independently and could have a different dimension.

From a computational point of view, in our approach there is no need to create the inefficient indicator super matrix G = (G_1| . . . |G_m) with dimensions $n \times t$ and sparsity equal to 1-m/t. It is sufficient to create the t-dimensional '*vocabulary*' of the categories and index it. Then all the categories in the data will be substituted by the corresponding numerical index. At this point we can introduce the array $\mathbf{C}$ = (C_1| . . . |C_m) of parameters with dimension $t \times p$ containing the quantification of the categories in the *vocabulary*. The value of p can be small ($\leq$10) also with high cardinality variables. It is an hyperparameter in the fitting of the model.

In an unsupervised approach, the categorical variables can be quantified by other NLP algorithms. The best-known method is *word2vec* (Mikolov et al. 2013) but also more recent proposals are available. Without a target variable, it is possible to estimate the array $\mathbf{C}$ by defining a Neural Network model that, for each unit, predict the category of a variable by knowing the categories of the other variables. The obtained

quantifications can be then used in a supervised model. This approach, which has allowed great progress in NLP, can manage big-data and categorical variables with high cardinality. One advantage of the unsupervised approach is that it does not necessarily require the analysis of many data, since the encodings can be obtained indirectly also on other data sets. Furthermore, the encodings do not use the target and therefore there is not risk of overfitting, but, in predictive models, these encodings are not optimized for the specific target that is analysed.

4 A comparison

Some of the most used encoding techniques have been compared with our proposal. Each of these methods has important drawbacks. Target encoder has a risk of '*target leakage*', in fact, it uses some information from target to predict the target itself, increasing the chance of overfitting on the training data. One-Hot Encoder, in the case of high cardinality leads to a large number of orthogonal features. Furthermore, this method alters the relationship between categorical and quantitative variables and can create memory problems. The Helmert contrast encoder can lead to overfitting and requires that the levels of the categorical variable are ordered and target is quantitative. In computational terms, we can remark that target-based methods are the fastest, while one-hot and categorical encoding take longer to calculate, depending on the cardinality of the categorical variables.

We considered the application to two different well-known datasets. The *Human Resources dataset* consists of 54,808 examples, 14 categorical and quantitative features and one binary target. The *Allstate Claims Severity* dataset is composed of 131 features, including 116 categorical variables and one continuous target observed on 188,318 examples.

In tab. 1 we have compared some encoders in a supervised approach. Considering a predictive aim, in each analysis we added to the encoder the same simple neural network with 4 internal layers. To regularize the network, dropout layers have been introduced.

Table 1: Comparison of encoding methods in two supervised analysis

Allstate Claim Severity data	MSE on Train (70%)	MSE on Test (30%)
One-Hot encoder	0.6584	0.6581
Helmert encoder	0.6583	0.6580
Target encoder	0.6582	0.6581
Categorical Encoder	**0.2688**	**0.3199**

Human Resources data	AUC on Train (70%)	AUC on Test (30%)
One-Hot encoder	0.9181	0.8991
Helmert encoder	0.9345	0.8929
Target encoder	0.8053	0.8070
Categorical Encoder	**0.9085**	**0.9033**

The usefulness of each encoder can therefore be assessed based on the predictive capacity of the model obtained. We can see that the categorical encoding (with p=10) is more effective in both cases, but mostly on the dataset with many categorical variables and high cardinality.

5 Conclusions

In Neural Networks, the One-Hot is the most common encoding technique for categorical data and, often, it leads to good results. On the other hand, this encoding has many drawbacks that can degrade the predictive performance of the Neural Networks model. Also, the alternative methods proposed in the literature have not proved particularly effective. The approach we have shown in this paper seems to bring clear advantages when there are many categorical variables with high cardinality. Using this approach, we can also build unsupervised models to analyse the relationship between categorical variables as in MCA. Some early applications in this regard show encouraging results.

References

1. Azzalini, A.: Inferenza statistica - Una presentazione basata sul concetto di verosimiglianza. Springer (2001)
2. Bengio Y., Ducharme R., Vincent P., Janvin C.: A neural probabilistic language model, The Journal of Machine Learning Research 3, 1137-1155 (2003).
3. Benzécri J.P.: L'Analyse Des Données: Tome II: L'Analyse Des Correspondances. Dunod, Paris (1973).
4. Di Ciaccio A.: Some considerations on the quantification of categorical data, Research-Report, Department of Data Theory, University of Leiden, (1988).
5. Di Ciaccio A.: Analisi simultanea dei caratteri qualitativi e quantitativi, Metron, vol.XLVIII, n. 1-4, (1990).
6. Gifi A.: Nonlinear Multivariate Analysis. Technical report, University of Leiden, Department of Data Theory, Leiden (1981).
7. Guo C., Berkhahn F.: Entity Embeddings of Categorical Variables, available: https://arxiv.org/abs/1604.06737 (2016)
8. Mikolov T., Chen K, Corrado G., Dean J.: Efficient Estimation of Word Representations in Vector Space. arXiv:1301.3781 (2013).
9. Pargent F.: A Benchmark Experiment on How to Encode Categorical Features in Predictive Modeling. Master Thesis in Statistics at LMU Munich (2019)
10. Pennington, J., Socher, R., Manning, C.D.: GloVe: Global Vectors for Word Representation. Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP) (2014), doi: 10.3115/v1/D14-1162
11. Potdar, K., Pardawala, T.S., Pai, C.D.: A comparative study of categorical variable encoding techniques for neural network classifiers. International Journal of Computer Applications 175, 4 (2017).
12. Stefanini, E.: Deep Embedding per variabili categoriche. Master's thesis, dept. of Statistics, university of Rome, La Sapienza (2020).