NEURAL NETWORKS FOR HIGH CARDINALITY CATEGORICAL DATA

Agostino Di Ciaccio

Department of Statistics, University of Rome "La Sapienza",
(e-mail: agostino.diciaccio@uniroma1.it)

ABSTRACT: If we want to apply neural networks to categorical data, we must necessarily adopt a coding strategy. This is a common problem for many multivariate techniques and several approaches have been suggested. In this paper, a method is proposed to analyze categorical variables with high cardinality. An application to simulated data illustrates the interest of the proposal.

KEYWORDS: encoding categorical data, neural networks, high cardinality attributes.

1 Introduction

Several machine learning algorithms cannot handle directly categorical variables and, in any case, categorical data can pose a serious problem if they have too many categories. Postal code is a good example of a categorical variable with high cardinality. This paper starts with some considerations on the currently used approaches, then an efficient encoding method is proposed for supervised neural networks when categorical variables with high cardinality need to be analyzed.

2 Approaches to quantify categorical features

Several methods have been proposed to encode categorical variables (a recent review is Hancock et al. 2020). From our point of view, they can be classified as:

1- Methods that do not use the target variable. In this category we find rather crude methods, such as the *Label Encoder* or the *Hashing Encoder*. The quantifications obtained are essentially arbitrary.
2- Methods that use only the target variable. The *Target Encoder* (TE) replaces the categorical variable with the conditional means of the target variable. This method often produces data leakage, to limit this inconvenience the *Leave one out Encoder* or the *Catboost Encoder* have been proposed.
3- Methods based on *One Hot Encoding* (OHE). In this approach a new binary variable is introduced for each category, indicating the presence or absence of that category. The eventual exclusion of one category is due to the

multicollinearity problem (the dummy variable trap), but applying machine learning models, as the neural networks, it is necessary to include all the categories, otherwise we would never consider the omitted category.

3 Single and multiple quantifications by OHE

One Hot Encoding is the most used method. The coding in dummies does not depend directly on the target. Despite its great use, some drawbacks of OHE are well known: the tendency of dummy variables to cause overfitting; the introduction of many new orthogonal variables, which can slow down or affect learning; memory problems.

The encoding of categorical variables has been extensively studied in the approach based on Optimal Scaling (OS, Gifi 1990) where the *embedding* of the categories in a *p*-dimensional space was proposed. Given a categorical variable X which can assume the values $[a_1, a_2, ..., a_k]$, with k the number of categories, n the number of observations, then $G = [g_1, g_2, ..., g_k]$ is the indicator matrix with dimension $n \times k$. Let **c** a vector of k real values, the quantification of X is the vector:

$$\mathbf{x} = \mathbf{G}\mathbf{c} = \sum_{h=1}^{k} c_h \mathbf{g}_h \tag{1}$$

The values of **c** are the quantifications of the k categories and have to be estimated. The vector of the quantified data **x** is a linear combination of the indicator variables, which are an orthogonal base of R^k, then is defined in a subspace of R^k. To obtain ordered quantifications in the OS, the order indicator matrices, with non-negativity constraints on the coefficients, can be used (Gifi 1990).

In expression (1) we considered a single quantification for a categorical variable. There are several reasons that may lead to consider two or more quantifications of the same variable (Di Ciaccio 2020). Considering a regressive problem, in OS (MORALS, Young et al. 1976) it is possible to obtain a multiple quantification by means of copies of the variables (Gifi 1990). After choosing the number p of quantifications, we can extend (1) as:

$$\underset{n\times p}{\mathbf{X}} = \underset{n\times k}{\mathbf{G}}\ \underset{k\times p}{\mathbf{C}} = \sum_{h=1}^{k} \underset{n\times 1}{\mathbf{g}_h} \cdot \underset{1\times p}{\mathbf{c}_h} \tag{2}$$

In neural network applications, fixing a low p, equal to 2 or 3, is usually enough for a good quantification of categorical variables even with high cardinality.

To introduce quantification (2) in a neural network it is necessary to define, for each categorical variable, a distinct input and a dense layer with p neurons without bias and with linear activation function. In the next layer the outputs, coming from all the variables, must be concatenated. For example, given 3 input categorical variables, each with 100 categories, and one hidden layer containing 512 neurons, using this approach we must estimate (considering a regression problem and p=2) 4.697 weights. Given t=512, p=2, m=3, k_j=100 for each j, the Neural Network can be written:

$$\hat{\mathbf{y}} = \beta_0 + \sum_{s=1}^{t} \beta_s \phi \left(\sum_{j=1}^{m} \sum_{r=1}^{p} \mathbf{G}_j \mathbf{c}_j^r w_{jrs} + w_{0s} \right) \tag{3}$$

where $\phi(.)$ is the activation function of the hidden layer, $\mathbf{c}_j^r$ is the quantification of the *j*-th variable on the *r*-th dimension. Conversely, in the classical OHE encoding:

$$\hat{\mathbf{y}} = \beta_0 + \sum_{s=1}^{t} \beta_s \phi\left(\sum_{j=1}^{m} \sum_{r=1}^{k_j} \mathbf{G}_j w_{jrs} + w_{0s} \right) \tag{4}$$

obtaining 154.625 weights to estimate.

$\mathbf{G}_j$ can be very big sparse matrices (sparsity equal to $1-1/k_j$), but we can avoid building such an inefficient coding estimating the dense matrix of quantifications $\mathbf{C}_j$ of expression (3) without building the sparse matrix $\mathbf{G}_j$.

In the first step, for a categorical variable *X*, the *k*-dimensional 'vocabulary' **V** of the categories have to be created and indexed. Then all the categories in the data will be substituted by the corresponding numerical index in the vocabulary, in a similar way to what the Label Encoder does. Call a_i the modality assumed by the categorical variable, and $\mathbf{v}[a_i]$ the index in the vocabulary corresponding to this modality. The *i*-th row of the $(n \times p)$ matrix of the quantified variable *X* can be expressed as:

$$\mathbf{x}_i = \mathbf{C}[\mathbf{v}[a_i]] \tag{5}$$

Each line of the quantification matrix **C** can be seen as the *p*-dimensional representation of one category. Inspired by Natural Language Processing, Guo & Berkhahn's (2016) *entity embedding* technique takes a similar approach. To obtain the estimate of **C** in a supervised neural network, the gradient descent and the backpropagation can be used, where the matrix **C** is initialized with random values taken from a standardized normal and subsequently updated through an iterative procedure to minimize the loss function, which in the case of regression is the classic Sum of Square Error. We call this technique LEE, Low Embedding Encoder, and to illustrate the proposed approach, a small simulation for a regression problem was build. Given three qualitative variables X_1, X_2, X_3 with 200 categories each (coded as the integers between 1 and 200), for each variable 20,000 observations were extracted randomly from a uniform distribution, then *Y* was computed by the rules:

$(X_1 > X_2 \text{ and } X_3 < 100) \rightarrow Y \sim N(20,1.5)$
$(X_1 \leq X_2 \text{ and } X_3 < 100) \rightarrow Y \sim N(10,1.5)$ *else* $Y \sim N(1,1.5)$

There are only 3 expected values $E(Y \mid x_1, x_2, x_3)$, i.e. (1, 10, 20), so an optimal regressive model should predict these values. Note that the expected value of *Y* depends on the interaction of the three categorical variables and that the three conditional distributions of *Y* overlap in the tails. The dataset was then splitted as training-set (50%) and test-set (50%). Regression algorithms such as MORALS or Regression Tree cannot make a satisfactory prediction on this data unless introducing explicitly the interaction terms into the model, producing thousands of dummy variables. On the contrary, neural networks are able to autonomously detect the interactions, then a small neural network was chosen to predict the target *Y* in our simulation. The network includes an input layer, two hidden layers with 8 and 3 neurons (*elu* activation function), and 1 output neuron with linear activation function. With the LEE approach, each categorical variable is considered a separate input and one dense layer with 2 neurons ($p = 2$) and no bias, for each categorical variable, is added to the input. If we want to avoid sparse matrices, an *embedding* layer can be

added, for each original categorical variable, using (5). It was also checked that the results obtained did not improve, on the test-set, by changing the size of the network or the number of iterations. Although the *Target Encoder* was applied also with a bigger neural network, with 32 neurons in each hidden layer, the result is very poor even on the training-set, as this encoding prevents interactions from being identified.

Table 1. *Comparison between three approaches*

	MSE - train	*MSE - test*	*n. parameters*
OHE	2.11	6.18	4839
LEE	2.55	4.82	1287
Target Encoder	61.47	61.48	1217

Figure 1. *OHE on the test-set*

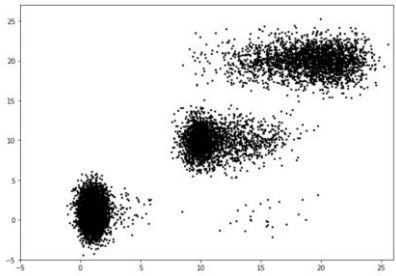
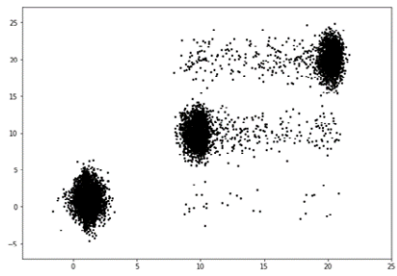


Figure 2. *LEE on the test-set*



4 Conclusions

The proposed method LEE allows to apply neural networks to categorical variables with high cardinality, reducing the number of parameters and memory resources. The results obtained show an increased predictive capacity of the neural network thanks to the more efficient architecture.

References

DI CIACCIO, A. 2020. Categorical Encoding for Machine Learning. *Book of short papers SIS2020*, A. Pollice et al. eds., ISBN 9788891910776, Pearson Italia.

GIFI, A. 1990. *Nonlinear Multivariate Analysis*. John Wiley & Sons, New York.

GUO, C., & BERKHAHN, F. 2016. Entity embeddings of categorical variables. *arXiv*:1604.06737.

HANCOCK, J.T., & KHOSHGOFTAAR, T.M. 2020. Survey on categorical data for neural networks. *Journal of Big Data*,**7**,28, https://doi.org/10.1186/s40537-020-00305-w

YOUNG, F.W., DE LEEUW, J., TAKANE, Y. 1976. Regression with qualitative and quantitative variables: an alternating least squares method with optimal scaling features. *Psychometrika*, v. **41**, n. 4.