Dealing with categorical missing data using CleanerR

Rafael S. Pereira[1], **Fabio Porto**[1]

[1]Laboratório Nacional de Computação Cientifica (LNCC), Data Extreme Lab (DEXL)
CEP: 22651-075 – Petrópolis – RJ – Brazil

{rpereira,fporto}@lncc.br

Abstract. *Missing data is a common problem in the world of data analysis. They appear in datasets due to a multitude of reasons, from data integration to poor data input. When faced with the problem, the analyst must decide what to do with the missing data since its not always advisable to discard these values from your analysis. On this paper we shall discuss a method that takes into account information theory and functional dependencies to best imput missing values.*

1. Introduction

A common problem everyone faces when doing data analytics is that your data may not have all its attributes complete. This may be due to many different reasons. For instance, data can be missing because of miscommunication, human error, equipment error or even data could not be gathered. In all but the last case data could then be inputted. One reason the analyst might not want to discard the data could be that he/she wants to train a machine learning model on it, and in this case there are two things that are required: the data should be mostly correct, so you don't train a model on bad data; and that there is enough data for the model to be able to generalize.

In this paper we shall discuss how a method that uses the concept of minimizing entropy by searching for almost functional dependencies, can give high confidence in the accuracy of the inputted values. Next we will present a implemented package on the R language [R Core Team 2014].

Our contributions on this paper are the test of the efficacy of the method for categorical data, as well as the implementation of the cleanerR package [Pereira 2019] that is already on CRAN.

The remaining of this paper is structured as follows. In section 2, we discuss previous works related to dealing with missing values. Next, in section 3, we discuss the methodology applied to solve the problem discussed in this paper. Section 4 presents some results obtained on a Kickstarter dataset. Finally, section 5 concludes.

2. Related Work

There are many different inputation techniques to input missing data. Some are non model based like inputting by the mean/ mode of a distribution, while others are more model based like Weighted K Nearest Neighbors[Troyanskaya O1 2001], Sequential Regression Trees[Burgette LF 2010] and auto associative neural networks [Abdella 2005]. There are also some tools to help dealing with missing data. For example, MICE[van Buuren and Groothuis-Oudshoorn 2011] uses Multivariate imputation using chained equations, While amellia [Honaker et al. 2011] uses a Expectation-Maximization

with Bootstrapping algorithm and Hmisc [Harrell Jr et al. 2019] uses additive regression, bootstrapping, and predictive mean matching as well as Fisher's optimum scoring method for categorical variables.

3. Methodology

When predicting possible values for a given attribute, the scientist must make some hyphotesis. For example, one may assume that the data provided is representative of the complete data domain, or in the case of categorical data, that all possible values appeared at least once. If then one wished to predict new values based solely on the data original distribution and the assumed hypothesis hold true, then we could define the statistical accuracy, which is the expected accuracy given that a chosen value is based on how frequent it already appeared on the original dataset. This is defined in the following formula:

For discrete values, let $P_1, P_2, ..P_n$ be probabilities so that

$$\sum_{i=1}^{n} P_i = 1 \tag{1}$$

Then the statistical accuracy will be defined by:

$$\overline{A} = \sum_{i=1}^{n} P_i^2 \tag{2}$$

And for continuous variables if $F(x)$ is a PDF

$$\overline{A} = \int_{-\infty}^{\infty} F(x)F(x)dx \tag{3}$$

Another way to look at these expressions is to say that the probability a certain value i would be choosen among all possible values of i is proportional to how often i has appeared on the dataset compared to other possible values.

It can be expressed that considering only these factors both in the discrete and continuos expression we could say:

$$\sum_{i=1}^{n} P_i^2 \geq \sum_{i=1}^{n} P_i P_j \tag{4}$$

$$\int_{-\infty}^{\infty} F(x)F(x)dx \geq \int_{-\infty}^{\infty} F(x)G(x)dx \tag{5}$$

Where both F and G are Probability Density Functions.

This expression has an entropy as defined by Shannon:

$$S = -\sum_{i} P_i log(P_i) \tag{6}$$

Then we can ask ourselves, can we use another source of information to make this entropy smaller?
Let Q_j be a set of elements that will be used to help predict the correct P_i, this way we can redefine the statistical accuracy equation for a more accurate method which is:

$$\overline{A_j} = \sum_{i=1}^{n} (P_i|Q_j)^2 \tag{7}$$

Where our choices of Q_j reduces the entropy from the original equation.

The algorithm cleanerR is based on searches for approximate functional dependencies which are described in databases theory to find the optimal Q_j to maximize accuracy.

3.1. Approximate Functional Dependency Search

To search for approximate functional dependencies we shall use the following algorithm:

- Consider a data set $D = \{A, G\}$.
- Define a goal column to predict, e.g. G.
- Define S as the unique values of the goal column.
- Define Q as all possible combinations of attributes from A, of max size L.
- Then, for every Q_i possible we calculate the entropy of $P_S|Q_i$
- The set of Q_i with minimum entropy represents approximate functional dependencies, in some cases even functional dependencies.

3.2. Values Imputation

For the value imputation the following algorithm is used:

Consider again a data set $D = \{A, G\}$ for which we want to predict values of a goal column G.

- This goal column has S distinct values;
- Each value has a relative frequency, that will be treated as its probability of P_i;
- Using the Functional Dependency Search algorithm, we determine the best Q set;
- Then for all missing values $P_i|Q_j$ is calculated;
- A Good Q will set $P_i = 0$ for most i;
- If all P_i but one equals to 0 for every unique Q_j, then Q is a functional dependency to P;
- A discrete PDF is built from the $P_i|Q_j$ and a random number is calculated between 0 and 1;
- This number will determine the inputted value, more frequent elements in $P_i|Q_j$ have a higher probability of being chosen;
- Process is done for every missing value in the goal collumn;
- Inputted values must be contained in S.

4. Experiments

In this section we define the context of our experiments, including the description of the experimental dataset. Next, we describe the application of our method to some of the columns of these datasets, when they are chosen as targets.

4.1. Dataset

This dataset was generated from the kickstarter website [1], and to scrape the data we used a robot tool [2] .
The dataset that was used to calibrate the method is a kickstarter dataset that stores information about all projects initiated in kickstarter in a certain period of time. The dataset has the following attributes:

- backersCount: number of backers on this project;
- blurb: description of the project
- category: category of the project
- converted: pledgedAmount Amount of dollars obtained
- country: country of origin of project creator;
- createdAt: date the project was created;
- creator: project creator
- currency: currency of the country of origin
- currencySimbol: symbol of the respective currency

And many more. The dataset has a total of 37 attributes and 205091 lines. Attributes of interest, could be: *country*, which tells us the country of the project, with 22 different countries; *state*, which tells us the state of the project, for example {successful, failed, canceled, live, suspended}; *currency*, which has 14 different values, and *spotlight*, which has only two values TRUE or FALSE. In the experiment section we shall discuss our efficacy when predicting the values for these attributes.

4.2. Experiment set-up

For the experiment we used the following technologies:

- R version 3.4.4
- cleanerR 0.1.1

Since our main goal is to see how well can we predict categorical missing values for the following experiments we will show the accuracy of prediction when the dataset has 5% missing data, 10%, 25% and 50% missing data in the attribute we wish to predict. This process will be done 10 times for spotlight, currency, country and state, and we shall discuss the statistical accuracy vs the method accuracy.

4.3. Experimental Results

In this section we discuss the results obtained in this work. To define the statistical accuracy, we consider that each value has the probability of being chosen as frequent as it appears, for example if TRUE appears 60 percent of the time, then it has probability 0.6 of being chosen in the prediction.

[1] https://www.kickstarter.com/
[2] https://webrobots.io/kickstarter-datasets/

To calculate the statitical accuracy we do the following:

- Obtain the frequency distribution of every unique value in a variable;
- Divide by the total number of elements in this variable obtaining probability of each variable;
- Having this probability vector we then calculate the dot product of this vector by itself.
- This result is the statitical accuracy, where a value is choosen based on how frequent it is.

4.3.1. Currency

Currency: Statistical Accuracy:0.6307156
Applying the same analysis using *cleanerR*, the tool indicates to use *country* to determine *currency*. As our experiments highlighted, this is indeed a case of a functional dependency, see Table 1.

Table 1. Country / Currency Data dependency

Percentage	Mean Accuracy	Standard Deviation of Accuracy
5	1	0
10	1	0
25	1	0
50	1	0

This table shows us the Mean Accuracy of predicting currency values when 5 to 50 percent of the original data was missing, we can see in this case that currency is fully determined by country.

4.3.2. Country

Country: Statistical accuracy: 0.6291397
In the case of country, we have an example where we do not find a functional dependency. Applying the analysis with *cleanerR BestVector*, which identifies the attribute with highest correlation with the target attribute, tells us to use currency, but in this case we get the results as in Table 2.

Table 2. Country with No Data dependency

Percentage	Mean Accuracy	Standard Deviation of Accuracy
5	0.9629315	0.001878949
10	0.9634112	0.001594487
25	0.9642183	0.0005740445
50	0.963773	0.0004847534

4.3.3. State

State: Statistical accuracy:0.4462562
In the case of *state*, *cleanerR* tells us to use both *disableCommunication* and *spotlight* atributes to best predict it. Table 3 shows the results:

Table 3. State Data dependency

Percentage	Mean Accuracy	Standart Deviation of Accuracy
5	0.8711137	0.0.00220583
10	0.8700375	0.001589842
25	0.8706741	0.0008510364
50	0.8702336	0.0007285919

for a more in-depth analysis or to do their own, the reader can check the jupyter notebook on github: https://github.com/R-S-P-MODELS/cleanerR-article

As we then seen in table 3 and 1 while some variables can be totally described by another, some require more variables and even so can not be predicted with 100 % accuracy. Even so we can see how using cleanerR greatly improved our accuracy during prediction

5. Conclusion

In this paper we discussed how a method that can find good attributes as features to predict a certain attribute can increase our accuracy by using conditional probabilities. It was also discussed how a method that searches for almost functional dependencies can minimize entropy, giving a set of features that help us predict well the data. Finally, we presented an evaluation using the *cleanerR* package, which helps the user to work with categorical missing data, giving tools for a good prediction.

6. Acknowledgements

The authors would like to thank the support of Conselho Nacional de Desenvolvimento Científico e Tecnológico(CNPQ). This study was financed in part by the Conselho Nacional de Desenvolvimento Científico e Tecnológico(CAPES) -Finance Code 001.

References

Abdella, M. Marwala, T. (2005). Treatment of missing data using neural networks and genetic algorithms. *Proceedings of the 2005 IEEE International Joint Conference on Neural Networks*, 1:598–603.

Burgette LF, R. J. (2010). Multiple imputation for missing data via sequential regression trees. *American Journal of Epidemiology*, 172:1070–1076.

Harrell Jr, F. E., with contributions from Charles Dupont, and many others. (2019). *Hmisc: Harrell Miscellaneous*. R package version 4.2-0.

Honaker, J., King, G., and Blackwell, M. (2011). Amelia II: A program for missing data. *Journal of Statistical Software*, 45(7):1–47.

Pereira, R. S. (2019). *cleanerR: How to Handle your Missing Data*. R package version 0.1.1.

R Core Team (2014). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Troyanskaya O1, Cantor M, S. G. B. P. H. T. T. R. B. D. A. R. (2001). Missing value estimation methods for dna microarrays. *Bioinformatics*, 17(06):520–525.

van Buuren, S. and Groothuis-Oudshoorn, K. (2011). mice: Multivariate imputation by chained equations in r. *Journal of Statistical Software*, 45(3):1–67.