Imputing Missing Data and Feature Selection via Penalized Matrix Decomposition

Jennifer Starling & Jesse Miller
Department of Statistics & Data Science
University of Texas at Austin

Abstract

We compare several methods of imputing values for missing data with continuous and categorical variables in the logistic regression setting. We apply penalized matrix decomposition to impute missing continuous values and compare the results to two other methods. We compare two methods for imputing missing categorical values. We finish by performing variable selection to compare several different models.

1 Introduction

Regression in the presence of missing data values is a well-studied but still relevant problem. Many simple proposed solutions exist, including exclusion of observations with one or more missing predictors from the data set, imputing missing values with each predictor's mean value, and others.

The setting for this project is prediction of whether an undergraduate student will pass an introductory Calculus course. The data is from the fall 2014 and fall 2015 semesters at the University of Texas at Austin (UT), and contains observations for all students who took an in-residence introductory Calculus class (M408C/K/N/R). Given how many students are required to take Calculus as part of their degree plans, it is important to be able to predict which students are likely to pass. Among other things, accurate predictions would allow UT to direct students who are likely to fail to remedial material that would help them prepare for Calculus.

Prediction of the binary response Pass/Fail could be handled in many ways (logistic regression, K-Means, random forest, etc.), which all perform optimally in the absence of missing values. We use logistic regression in our analyses.

Predictors are both categorical and quantitative; this project focuses on imputing missing values for continuous predictors. Continuous predictors include SAT and ACT scores, predicted GPAs, AP test scores, scores from ALEKS (an online system formerly used for placement purposes), and scores from the University of Texas Math Assessment (UTMA), which is the current exam used by the UT Math Department for placement into M408C/K/N/R. Categorical variables include the schools in which students are enrolled, parental income, parental education levels, first-generation-college status, and what, if any, high school Calculus students have had.

Data in this setting typically contain many missing values. For example, students commonly take either the SAT or ACT, but usually not both. Students also rarely take all AP tests. The data set contains 6,266 observations, 28 continuous predictors, and has about 11% of the data missing (19,536 missing values out of 175,448 total values).

30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

2 Related Work

As described in the Introduction, there are many simple solutions for handling missing continuous data, such as imputation with the mean and excluding observations. More complex solutions include those proposed by Hastie et al. in [3]. That paper proposes imputation of missing data via three methods: Singular Value Decomposition, Nearest-Neighbor Imputation, and Imputation Using Regression, which is an Expectation-Maximization-based approach.

A relatively new solution uses properties of penalized matrix decomposition to impute missing values. This solution is proposed in [7]. This paper describes an algorithm for recovering the rank-K penalized matrix decomposition.

3 Proposed Work

We will first summarize the methodology from [7]. Second, we will present two simulated examples to illustrate how the method performs both missing data imputation and variable selection via sparsity. Third, we impute the missing data values using the penalized matrix decomposition methodology and discuss improvements in a logistic regression model, including comparison to two other methods. Fourth, we will illustrate how the penalized matrix decomposition method can be used for feature selection. Fifth, we will impute missing categorical values by imputing the mode and compare this to imputing an explicit "missing" value. Last, we will use the best of these methods to build several models which include both categorical and continuous predictors and compare them.

4 Experiment

4.1 Method Survey

We begin by providing a brief survey of the penalized matrix decomposition method.

Parameter Selections. The method allows the user to specify a rank (from 1 to the number of predictors) for the decomposition. For purposes of data imputation, a full rank decomposition is used. For variable selection, a rank 1 decomposition allows the number of interesting selected variables to decrease towards one.

The method allows the user two specify two penalty terms, λ_U and λ_V. These penalty terms dictate the sparsity of the U and V matrices in the penalized decomposition. In the data imputation case, large values of λ prove useful, as we are not interested in introducing sparsity in this scenario. For variable selection, decreasing λ progressively results in smaller subsets of variables identified as interesting. This is discussed in more detail later.

Rank 1 Penalized Matrix Decomposition. We first implement the rank 1 sparse matrix factorization algorithm detailed on pages 519–520 of [7]. The R function implementation is `sp.matrix.decomp.rank1()`.

The optimization problem is:

$$\underset{u\in R^N, v\in R^p}{\operatorname{argmin}} = ||\mathbf{X} - \mathbf{duv^T}||_{\mathbf{F}}^{\mathbf{2}} \text{ subject to } ||\mathbf{u}||_{\mathbf{2}}^{\mathbf{2}} = \mathbf{1},\ ||\mathbf{v}||_{\mathbf{2}}^{\mathbf{2}} = \mathbf{1},\ ||\mathbf{u}||_{\mathbf{1}} \leq \lambda_{\mathbf{u}},\ ||\mathbf{v}||_{\mathbf{1}} \leq \lambda_{\mathbf{v}},$$

where F indicates the squared Frobenius norm of a matrix (sum of squared elements). This problem is equivalent to:

$$\text{maximize}_{u,v}\mathbf{u^T Xv} \text{ subject to } ||\mathbf{u}||_{\mathbf{2}}^{\mathbf{2}} = \mathbf{1},\ ||\mathbf{v}||_{\mathbf{2}}^{\mathbf{2}} = \mathbf{1},\ ||\mathbf{u}||_{\mathbf{1}} \leq \lambda_{\mathbf{u}},\ ||\mathbf{v}||_{\mathbf{1}} \leq \lambda_{\mathbf{v}}.$$

The equivalence is proved in the appendix of [7]. The outline of the algorithm is as follows.

1. Initialize v to some random vector whose l_2 norm equals 1.
2. Let $u = \frac{S_{\theta_u}(Xv)}{||S_{\theta_u}(Xv)||_2}$
3. Let $v = \frac{S_{\theta_v}(X^T u)}{||S_{\theta_v}(X^T u)||_2}$

4. Repeat until convergence is reached.

The convergence criteria is $sum(abs(v.old - v.new)^2 < \epsilon$. Two technical details:

- $S_{\theta_v}(a) = \text{sign}(a)(|a| - \theta_v)_+$ is the same soft-thresholding operator used in the LASSO solution.
- θ_u and θ_v are found using binary searches within each iteration.

Rank K Penalized Matrix Decomposition. There is an outline in [7] of a recursive strategy to derive the rank K penalized matrix decomposition. This recursion relies on repeated use of the rank 1 algorithm. The R function implementation is `sparse.matrix.decomp.rankk()`.

The outline of the algorithm is as follows.

1. Initialize v to some random vector where the l_2 norm equals 1.
2. Let $\mathbf{X^1} = \mathbf{X}$.
3. For $k \in 1, \ldots, K$:
   (a) Find $\mathbf{u_k}, \mathbf{v_k}$ and d_k by using the rank 1 algorithm with input $\mathbf{X^K}$.
   (b) $\mathbf{X^{K+1}} = \mathbf{X^K} - \mathbf{d_k u_k x_k^T}$

Missing Data Method. It is noted in [7] that the algorithm works even when there are missing observations; the missing elements of a matrix X can be excluded from all computations, and so the algorithm can be used to impute missing values. The algorithm is essentially identical to the rank K method above.

The algorithm is as follows. The set of indices where X has nonmissing values is denoted by C.

$$\text{maximize}_{u,v} \sum_{(i,j)\in C} X_{ij} u_i v_j \quad \text{subject to} \quad ||u||_2^2 = 1,\ ||v||_2^2 = 1,\ ||u||_1 \leq \lambda_u,\ ||v||_1 \leq \lambda_v.$$

4.2 Example: Illustrating Missing Data Imputation

This example illustrates using penalized matrix decomposition to impute missing data. We generate a small (5×4) X matrix of random values. Then 50% of these values are randomly set to NA. The first matrix is the original, and the second matrix is imputed using the penalized matrix decomposition. The λ penalties were set to a large value (ten) in this case, as sparsity is not a desired feature.

Listing 1: Original Matrix

−0.371	−0.869	NA	−0.262
−0.566	−0.317	NA	NA
NA	0.022	NA	NA
0.245	−1.372	NA	0.562
−0.409	NA	NA	NA

Listing 2: Imputed Matrix

−0.371	−0.869	−0.244	−0.262
−0.566	−0.317	−1.326	−0.141
−0.093	0.022	0.937	−0.098
0.245	−1.372	1.131	0.562
−0.409	−0.145	−0.907	0.973

The penalized matrix decomposition method imputes the missing values, while preserving the non-missing values.

4.3 Example: Illustrating Feature Selection

Decreasing the λ penalties increases sparsity, effectively selecting a limited number of non-zero elements of U and V, leading to a reconstructed X matrix, where $X = U * D * t(V)$, with a limited number of columns containing non-zero elements.

This example uses another toy (5×4) X matrix, this time with no missing values. It is noted in [7] that to obtain equal levels of sparsity in U and V, set a constant c, and let $\lambda_U = c\sqrt{n}$ where $n = \text{nrow}(X)$, and let $\lambda_V = c\sqrt{p}$ where $p = \text{ncol}(X)$. In this example, we used values of c in $\{1.0, 0.8, 0.6, 0.4, 0.2\}$.

This table shows the number of columns in the reconstructed $X = U * D * t(V)$ matrix containing non-zero values. The number of columns containing at least one non-zero value decreases as λ decreases, illustrating the variable selection effect of the λ penalties.

Table 1: Variable Selection by Lambda Penalty

c	λ_U	λ_V	Nonzero X cols
1.0	2.236	2.0	4
0.8	1.789	1.6	4
0.6	1.342	1.2	2
0.4	0.894	0.8	1
0.2	0.447	0.4	1

4.4 Missing Data Imputation for Continuous Predictors

The penalized matrix decomposition method was performed on the scaled continuous predictor data, with large lambda penalty values as to avoid inducing sparsity. After imputing the continuous predictor data using penalized matrix factorization, a few checks were performed to verify reasonableness.

First, we generated histograms of the raw data for each continuous predictor, with the kernel density estimate of the predictor including imputed data overlaid in blue. This is a check to ensure that imputing the data is not significantly changing the shape of the distribution of each predictor. The histograms verify that the imputed data is not significantly altering the distribution of any of the predictors. A sample of the histograms are shown below.

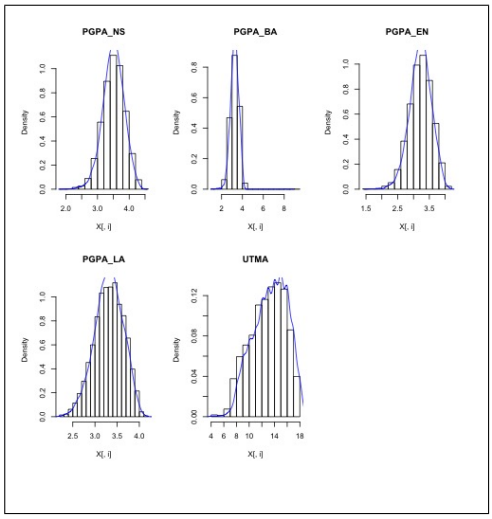

Figure 1: Histograms of imputed values for GPA-related predictors

All imputed variable ranges were checked for reasonableness as well, since the continuous predictors represent variables such as GPA, SAT and ACT test scores, where imputed values must be within a specific range of allowable values in order to be plausible. Below is an example of checking the ranges of imputed predictors for reasonableness for the SAT-related predictors.

Table 2: Imputed SAT Range Checks

Original			Imputed		
SATQ	SATV	SATTot	SATQ	SATV	SATTot
370	310	770	369.88	319.83	809.77
800	800	1600	799.75	799.82	1599.54

Let us now use a logistic regression for the continuous predictors only, to compare the results of the penalized matrix decomposition for missing value imputation versus the raw continuous predictor data and two other methods of imputation. We split the data into a training set (70%) and a test set (30%). This split was performed for both the original data and the imputed data, yielding two training sets and two test sets.

A logistic regression was performed on the training data with missing values and fit to the test data with missing values. The test error was 0.31. A second logistic regression was performed, this time using the training date with imputed values and fit to the test data with imputed values. The test error was 0.21.

Both of these models were fit using no other model selection techniques; both were fit to all continuous predictors. The imputed data results in a drastic improvement in test error, and the model is now ready to be improved in other ways such as variable selection.

Two other methods of imputing missing data were also compared. First, imputing missing values using the mean for each predictor resulted in a test error of 0.23. Second, imputing missing values using singular value decomposition also resulted in a test error of 0.23. Both methods improved on the original missing data results, but did not yield as much improvement as the penalized matrix decomposition method.

In the remaining sections, we use continuous variables with values imputed using the penalized matrix decomposition method.

4.5 Missing Data Imputation for Categorical Predictors

Two methods of imputation of missing categorical values were considered: imputing the mode of each variable, and imputing an explicit "missing" value to create an additional category for each variable. We again used a 70/30 split of the two imputed data sets into two pairs of training and testing sets. Averaging over 10 splits, we obtained the following test errors for logistic regression models using only categorical variables and models using all variables, with missing continuous values imputed using the matrix method.

Table 3: Error Rates for Mode Imputation versus "Missing" Imputation

	Mode	Missing
categorical variables	0.257	0.247
all variables	0.216	0.215

In both cases, we can see a slight improvement by adding a "missing" category over imputing the mode, although the improvement nearly disappears in the presence of the continuous variables. In the remaining section, we use categorical variables with a "missing" value imputed.

4.6 Variable Selection

Variable Selection via Penalized Matrix Decomposition:

First, the λ penalty parameters of the matrix decomposition method were adjusted to explore use of this method as a feature selection tool. The following λ combinations were tested.

Table 4: Variable Selection via Penalized Matrix Decomposition

c	λ_U	λ_V	Predictors Included	Test Error
0.30	23.747	1.775	35	0.21
0.20	15.832	1.183	34	0.23
0.10	7.916	0.592	25	0.23
0.05	3.958	0.296	16	0.23
0.01	0.792	0.059	6	0.25

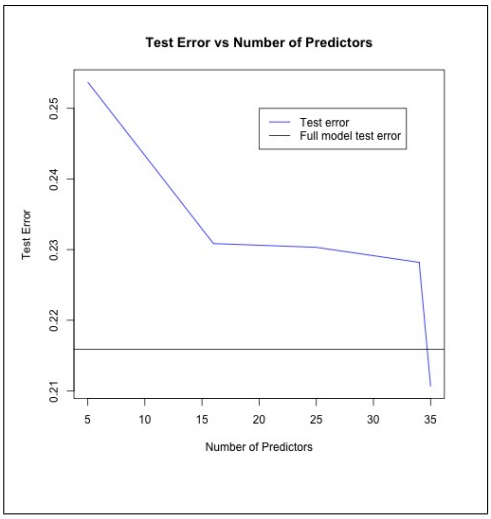

Figure 2: Test Error

This method was not as successful for variable selection as we had hoped. The test error rate for the most parsimonious was 0.21, which matched the test error rate when including all continuous predictors. Additionally, the model with the best test error only eliminated one predictor. With a full set of 36 predictors, other methods would be perhaps more effective in producing a simpler but more effective model.

The values for c, and subsequently the λ values, were chosen as described in the method section to impose equal levels of sparsity on the U and V components of the penalized matrix decomposition. A further area for investigation could be the effectiveness of testing differing levels of sparsity when performing variable selection using the penalties.

After ruling out adjustment of the penalty values as an effective method of variable selection, we proceeded to perform various variable selection techniques on the entire data set, including categorical and continuous predictors together.

General Variable Selection with Continuous and Categorical Predictors:

The following table shows the error rate for several different models, listed in descending order of error rates. One model was obtained using stepwise selection to determine the variables. Backward selection was also done, but this resulted in the same model as stepwise selection. Three others were obtained using penalized matrix decomposition (PMD) to choose from among the continuous variables, as described above. (The categorical variables in those cases were not considered.) Several context-important models are also included. The UTMA is the current placement exam for introductory Calculus at UT, so its performance on its own was tested. The SAT and ACT tests are very common standardized tests and are often used for admission criteria and placement purposes, so we tested a model that includes the SAT quantitative score and the ACT math score. The benchmark error rate is the rate using the trivial model that predicts that everyone will pass. The culled full model includes all predictors except two that are each sums of two other predictors, two with the same entry for all students, and one with no data prior to imputation.

Table 5: Model Testing, in Descending Order of Error Rates

Model	# of Continuous Predictors	# of Categorical Predictors	Error Rate
benchmark	0	0	0.2686170
PMD ($\lambda = 0.01$)	5	0	0.2537234
SATQ & ACTMath	2	0	0.2430851
UTMA	1	0	0.2345745
PMD ($\lambda = 0.05$)	16	0	0.2308511
PMD ($\lambda = 0.1$)	25	0	0.2303191
stepwise selection	20	2	0.2148936
culled full model	31	7	0.2143617

5 Conclusions and Further Work

Imputation of continuous variables using the matrix method led to a more accurate logistic regression model than either the use of the original data or the use of imputed mean values. Imputation of categorical variables using a "missing" value led to a slightly more accurate model than did imputation of modes. Based on what we have seen so far with this data set, though, it will be difficult to lower the error rate below 20%.

From the perspective of creating a predictive model, this is somewhat disheartening. From the perspective of a University, though, perhaps this is good: we are not offering a course whose outcome could have been predicted based on pre-college data alone. We are offering students something new.

That being said, there are still avenues to explore for model improvement. There are other methods of imputing values for missing continuous variables. There are also much more sophisticated methods available for dealing with missing categorical data: [2] and [6] discuss the use of latent class models for imputation, [4] discusses a Bayesian method for imputation, and [1] discusses bias correction. We can also explore classification methods other than logistic regression.

6 Source Code

All project documentation and source code is available in the following github repository.

`https://github.com/jstarling1/penalized-matrix-decomp`

References

[1] Das, U, Maiti, T, and Pradhan, V. (2010) *Bias correction in logistic regression with missing categorical covariates.* Journal of Statistical Planning and Inference, **140**: pp 2478—2485.

[2] Gebregziabher, M. and DeSantis, S. (2010) *Latent class based multiple imputation approach for missing categorical data.* Journal of Statistical Planning and Inference, **140**: pp 3252—3262.

[3] Hastie, T, Tibshirani, R, Sherlock, G, Eisen, M, Brown, P, and Botstein, D. (1999) *Imputing Missing Data for Gene Expression Arrays.* Technical Report, Division of Biostatistics, Stanford University.

[4] Li, X. (2009) *A Bayesian Approach for Estimating and Replacing Missing Categorical Data.* ACM Journal of Data and Information Quality, **1** (1): Article No. 3.

[5] Troyanskaya, O, Cantor, M, Sherlock, G, Brown, P, Hastie, T, Tibshirani, R, Botstein, D, and Altman, R.B. (2001). *Missing value estimation methods for DNA microarrays.* Bioinformatics, **17** (6), pp 520-–525.

[6] Vidotto, D, Vermunt, J, and Kaptein, M. (2015) *Multiple Imputation of Missing Categorical Data using Latent Class Models: State of the Art.* Psychological Test and Assessment Modeling, **57** (4): pp 542–576.

[7] Witten, D, Tibshirani, R, and Hastie, T. (2009) *A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis.* Biostatistics, **10** (3): pp 515–534.