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

Here we apply penalized matrix decomposition to impute missing data values and perform variable selection for continuous predictor variables in the logistic regression setting.

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 a basic calculus course. 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.

Data in this setting typically contains many missing values. Predictors are both categorical and quantitative; this project focuses on imputing missing values for continuous predictors. Continuous predictors include SAT scores, ACT scores, high school GPA data, extrapolated undergraduate GPA (as estimated by the University of Texas at Austin), and various standardized testing scores. Students commonly select either the SAT or ACT, and not all students sit for all standardized 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).

2 Related Work

As described in the Introduction, there are many simple solutions for handling missing data, such as imputation with the mean and excluding observations. More complex solutions include those proposed by Hastie et al. in their 1999 paper, "Imputing Missing Data for Gene Expression Arrays." Haste et al. propose 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 the Witten, Hastie & Tibshirani (2009) paper, "A Penalized Matix Decomposition, With Applications to Sparse Principal Components and Canonical Correlation Analysis." This paper details an algorithm for recovering the rank-K penalized matrix decomposition, which has several benefits, to be discussed.

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

3 Proposed Work

We will first summarize the methodology from the Witten, Hastie and Tibshirani paper. We will then present two simulated examples to illustrate how the method performs both missing data imputation and variable selection via sparsity. Third, we will 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. Last, we will briefly present using the methodology for variable selection, with a discussion of penalty parameter selection.

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 dicatate the sparsity of the U and V matrices in the penalized decomposition. In the data imputation case, large values of lambda prove useful, as we are not interested in introducing sparsity in this scenario. For variable selection, decreasing lambda 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 Witten, Hastie & Tibshirani, 2009. The R function implementation is *sp.matrix.decomp.rank1()*.

The optimization problem is:

$$\underset{u\in R^N, v\in R^p}{\text{argmin}} = ||\mathbf{X} - \mathbf{duv^T}||^{\mathbf{2}}_{\mathbf{F}} \text{ subject to } ||u||_2^2 = 1, ||v||_2^2 = 1, ||u||_1 \leq \lambda_u, ||v||_1 \leq \lambda_v,$$

where F indicates the squared Frobenius norm of a matrix (sum of squared elements).

This problem is equivalent to:

$$\text{maximize}_{u,v}\mathbf{u^T Xv} \text{ subject to } ||u||_2^2 = 1, ||v||_2^2 = 1, ||u||_1 \leq \lambda_u, ||v||_1 \leq \lambda_v,$$

The equivalence proof is in the appendix of the Witten et al. paper.

The outline of the algorithm is as follows.

1. Initialize v to some random vector where the l2 norm equals 1.
2. Let $u = \frac{S_{\theta_u}(Xv)}{||S_{\theta_u}(Xv)||_2}$
3. Let $v = \frac{S_{\theta_v}(X^T u)}{||S_{\theta_v}(X^T u)||_2}$

Iterate these steps until convergence is reached. Convergence criteria: $sum(abs(v.old - v.new)^2 < \epsilon$

Technical details:
• $S_{\theta_v}(a) = sign(a)(|a| - \theta_v)_+$ 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:

Witten et al. outline 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 l2 norm equals 1.
2. Let $\mathbf{X^1} = \mathbf{X}$.
3. For $k \in 1, ..., K$:
   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:

Witten et al. note that the algorithm works even when there are missing observations; the missing elements of 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. C indicates the set of indices where X has nonmissing values.

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

4.2 Example: Illustrating Missing Data Imputation

This example illustrates using penalized matrix decomposition to impuate missing data. We generate a small (5x4) X matrix of random values. Then a percentage of the X values (50% in this case) are randomly set to NA. The first matrix is the original, and the second matrix is imputed using the penalized matrix decomposition. Lambda 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

4.3 Example: Illustrating Feature Selection

Decreasing the lambda 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 (5x4) X matrix, this time with no missing values. Witten et al. note that to obtain equal levels of sparisty in U and V, set a constant c, and let $\lambda_U = c\sqrt{n}$ where $n = nrow(X)$, and let $\lambda_V = c\sqrt{p}$ where $p = ncol(X)$. In this example, c was allowed to vary so $c = 1.0, 0.8, 0.6, 0.4, 0.2$.

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 lambda decreases, illustrating the variable selection effect of the lambda penalties.

Table 1: Variable Selection by Lambda Penalty

c	lambdaU	lambdaV	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

After imputing the continous 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.

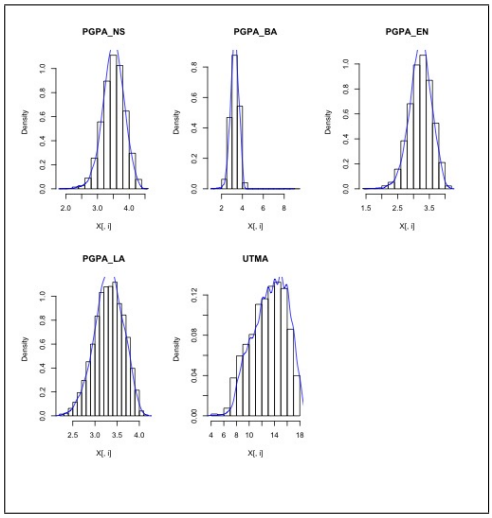

Figure 1: Histograms of imputed values for GPA-related predictors

All imputed variable ranges were checked for reasonableness as well. 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

4.5 Logistic Regression Results

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 .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 .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.

4.6 Variable Selection

As illustrated in Example 2, the penalized matrix decomposition can also be used for variable selection. (Recall: Decreasing the lambda 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 method was not as successful for variable selection as we had hoped. The test error rate for the most parsimonious was .20, improved from .21, but the chosen model 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.

References

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

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

[3] 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.

4.7 Style

Papers to be submitted to NIPS 2016 must be prepared according to the instructions presented here. Papers may only be up to eight pages long, including figures. Since 2009 an additional ninth page *containing only acknowledgments and/or cited references* is allowed. Papers that exceed nine pages will not be reviewed, or in any other way considered for presentation at the conference.

The margins in 2016 are the same as since 2007, which allow for $\sim 15\%$ more words in the paper compared to earlier years.

Authors are required to use the NIPS LaTeX style files obtainable at the NIPS website as indicated below. Please make sure you use the current files and not previous versions. Tweaking the style files may be grounds for rejection.

4.8 Retrieval of style files

The style files for NIPS and other conference information are available on the World Wide Web at

`http://www.nips.cc/`

The file `nips_2016.pdf` contains these instructions and illustrates the various formatting requirements your NIPS paper must satisfy.

The only supported style file for NIPS 2016 is `nips_2016.sty`, rewritten for LaTeX 2_ε. **Previous style files for LaTeX 2.09, Microsoft Word, and RTF are no longer supported!**

The new LaTeX style file contains two optional arguments: `final`, which creates a camera-ready copy, and `nonatbib`, which will not load the `natbib` package for you in case of package clash.

At submission time, please omit the `final` option. This will anonymize your submission and add line numbers to aid review. Please do *not* refer to these line numbers in your paper as they will be removed during generation of camera-ready copies.

The file `nips_2016.tex` may be used as a "shell" for writing your paper. All you have to do is replace the author, title, abstract, and text of the paper with your own.

The formatting instructions contained in these style files are summarized in Sections 5, 6, and 7 below.

5 General formatting instructions

The text must be confined within a rectangle 5.5 inches (33 picas) wide and 9 inches (54 picas) long. The left margin is 1.5 inch (9 picas). Use 10 point type with a vertical spacing (leading) of 11 points. Times New Roman is the preferred typeface throughout, and will be selected for you by default. Paragraphs are separated by ½ line space (5.5 points), with no indentation.

The paper title should be 17 point, initial caps/lower case, bold, centered between two horizontal rules. The top rule should be 4 points thick and the bottom rule should be 1 point thick. Allow ¼ inch space above and below the title to rules. All pages should start at 1 inch (6 picas) from the top of the page.

For the final version, authors' names are set in boldface, and each name is centered above the corresponding address. The lead author's name is to be listed first (left-most), and the co-authors' names (if different address) are set to follow. If there is only one co-author, list both author and co-author side by side.

Please pay special attention to the instructions in Section 7 regarding figures, tables, acknowledgments, and references.

6 Headings: first level

All headings should be lower case (except for first word and proper nouns), flush left, and bold.

First-level headings should be in 12-point type.

6.1 Headings: second level

Second-level headings should be in 10-point type.

6.1.1 Headings: third level

Third-level headings should be in 10-point type.

Paragraphs There is also a `\paragraph` command available, which sets the heading in bold, flush left, and inline with the text, with the heading followed by 1 em of space.

7 Citations, figures, tables, references

These instructions apply to everyone.

7.1 Citations within the text

The `natbib` package will be loaded for you by default. Citations may be author/year or numeric, as long as you maintain internal consistency. As to the format of the references themselves, any style is acceptable as long as it is used consistently.

The documentation for `natbib` may be found at

`http://mirrors.ctan.org/macros/latex/contrib/natbib/natnotes.pdf`

Of note is the command `\citet`, which produces citations appropriate for use in inline text. For example,

```
\citet{hasselmo} investigated\dots
```

produces

Hasselmo, et al. (1995) investigated. . .

If you wish to load the `natbib` package with options, you may add the following before loading the `nips_2016` package:

```
\PassOptionsToPackage{options}{natbib}
```

If `natbib` clashes with another package you load, you can add the optional argument `nonatbib` when loading the style file:

```
\usepackage[nonatbib]{nips_2016}
```

As submission is double blind, refer to your own published work in the third person. That is, use "In the previous work of Jones et al. [4]," not "In our previous work [4]." If you cite your other papers that are not widely available (e.g., a journal paper under review), use anonymous author names in the citation, e.g., an author of the form "A. Anonymous."

7.2 Footnotes

Footnotes should be used sparingly. If you do require a footnote, indicate footnotes with a number[1] in the text. Place the footnotes at the bottom of the page on which they appear. Precede the footnote with a horizontal rule of 2 inches (12 picas).

Note that footnotes are properly typeset *after* punctuation marks.[2]

7.3 Figures

All artwork must be neat, clean, and legible. Lines should be dark enough for purposes of reproduction. The figure number and caption always appear after the figure. Place one line space before the figure caption and one line space after the figure. The figure caption should be lower case (except for first word and proper nouns); figures are numbered consecutively.

You may use color figures. However, it is best for the figure captions and the paper body to be legible if the paper is printed in either black/white or in color.

[1] Sample of the first footnote.

[2] As in this example.

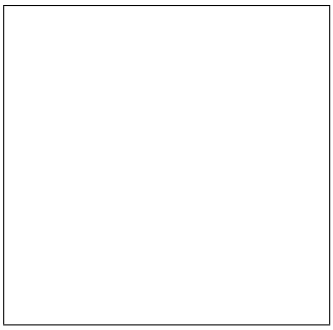

Figure 2: Sample figure caption.

Table 3: Sample table title

	Part	
Name	Description	Size (μm)
Dendrite	Input terminal	$\sim$100
Axon	Output terminal	$\sim$10
Soma	Cell body	up to 10^6

7.4 Tables

All tables must be centered, neat, clean and legible. The table number and title always appear before the table. See Table 3.

Place one line space before the table title, one line space after the table title, and one line space after the table. The table title must be lower case (except for first word and proper nouns); tables are numbered consecutively.

Note that publication-quality tables *do not contain vertical rules.* We strongly suggest the use of the `booktabs` package, which allows for typesetting high-quality, professional tables:

`https://www.ctan.org/pkg/booktabs`

This package was used to typeset Table 3.

8 Final instructions

Do not change any aspects of the formatting parameters in the style files. In particular, do not modify the width or length of the rectangle the text should fit into, and do not change font sizes (except perhaps in the **References** section; see below). Please note that pages should be numbered.

9 Preparing PDF files

Please prepare submission files with paper size "US Letter," and not, for example, "A4."

Fonts were the main cause of problems in the past years. Your PDF file must only contain Type 1 or Embedded TrueType fonts. Here are a few instructions to achieve this.

- You should directly generate PDF files using `pdflatex`.
- You can check which fonts a PDF files uses. In Acrobat Reader, select the menu Files>Document Properties>Fonts and select Show All Fonts. You can also use the program `pdffonts` which comes with `xpdf` and is available out-of-the-box on most Linux machines.
- The IEEE has recommendations for generating PDF files whose fonts are also acceptable for NIPS. Please see `http://www.emfield.org/icuwb2010/downloads/IEEE-PDF-SpecV32.pdf`
- `xfig` "patterned" shapes are implemented with bitmap fonts. Use "solid" shapes instead.
- The `\bbold` package almost always uses bitmap fonts. You should use the equivalent AMS Fonts:

```
\usepackage{amsfonts}
```

followed by, e.g., `\mathbb{R}`, `\mathbb{N}`, or `\mathbb{C}` for $\mathbb{R}$, $\mathbb{N}$ or $\mathbb{C}$. You can also use the following workaround for reals, natural and complex:

```
\newcommand{\RR}{I\!\!R} %real numbers
\newcommand{\Nat}{I\!\!N} %natural numbers
\newcommand{\CC}{I\!\!\!\!C} %complex numbers
```

Note that `amsfonts` is automatically loaded by the `amssymb` package.

If your file contains type 3 fonts or non embedded TrueType fonts, we will ask you to fix it.

9.1 Margins in LaTeX

Most of the margin problems come from figures positioned by hand using `\special` or other commands. We suggest using the command `\includegraphics` from the `graphicx` package. Always specify the figure width as a multiple of the line width as in the example below:

```
\usepackage[pdftex]{graphicx} ...
\includegraphics[width=0.8\linewidth]{myfile.pdf}
```

See Section 4.4 in the graphics bundle documentation (`http://mirrors.ctan.org/macros/latex/required/graphics/grfguide.pdf`)

A number of width problems arise when LaTeX cannot properly hyphenate a line. Please give LaTeX hyphenation hints using the `\-` command when necessary.

Acknowledgments

Use unnumbered third level headings for the acknowledgments. All acknowledgments go at the end of the paper. Do not include acknowledgments in the anonymized submission, only in the final paper.