Data Imputation Methods for Orthogonal Data Sets

Eilif B. Mikkelsen

1 Abstract

Through a factorial experiment, the shape of model degredation is measured in the presense of various levels of missing data, data relationship characteristics, and imputation methods.This paper proposes a series of reccomendations for data imputation on datasets that show little to no multicolinearity.

2 Introduction

Datasets often have missing data in some or all of the features of interest. Given a data set with independent regressors that can be described using ordinary least squares linear regression, what is the best way to fill in any missing data.

There is extensive study on how to leverage feature relationships to fill in missing data however there is little information on methods of independent data sets. The objective of this study is to quantify the impact of various data imputation methods on the model performance and provide recommendations on best practices for independent datasets.

3 State of Research

4 Methodology

The implementation of the methods explored in this paper were implemented using the Python/Numpy/Pandas stack with the Statsmodels package providing a robust implementation of Ordinary Least Squares linear regression.

4.1 Data Assumptions and Generation Details

The data set must be comprised of n i.i.d variables. The response y is computed to follow a linear model with all main effects and two factor interactions in accordance with equation 1. In practice, most of the research focused on the two variable case shown in equation 2.

$$y = \beta_0 + \beta_{X_1}X_1 + \ldots + \beta_{X_n}X_n + \beta_{X_1X_2}X_1X_2 + \ldots + \beta_{X_{n-1}X_n}X_{n-1}X_n + \epsilon \tag{1}$$

$$y = \beta_0 + \beta_{X_1}X_1 + \beta_{X_2}X_2 + \beta_{X_1X_2}X_1X_2 + \epsilon \tag{2}$$

To simplify data generation, all β values are proportional to β_{X_1}. For example, in a data set with three variables with β_{X_1} set to 10, the array $[0.1, 1.5]$ defines β_{X_2} and β_{X_3} to 1 and 15 respectively. The β values for interaction terms are set explicitly.

4.2 Simulating Missing Data

For a given observation, the probability that a given variable is missing is idependent from the probability that any other variable is missing. This randomness assumption is refered to as Missing Completely at Random (MCAR). For the implementation used in this study, a boolean mask was randomly created for each of the variables. The selection of data to eliminate can be written as replacement. The DataFrame method, `pandas.DataFrame.sample` is used for performance and simplicity reasons. Removed values were set to NULL. The number of datapoints to remove is controlled as the fraction of data missing. Given MCAR, it is possible for more than one variable to be missing. These observations are removed from the dataset before imputation begins. The response y is never nulled.

4.3 Imputation Methods

Dropping any observations with missing data, hereafter refered to as the Drop imputation method. Replacing missing values in a variable with the mean of the remain data, hereafter refered to as the Mean imputation method.

$$X_m = \begin{bmatrix} 1 & X_1 & X_2 & X_1X_2 \end{bmatrix} \tag{3}$$

The complete observations are given as.

$$X_c = \begin{bmatrix} X_m \neq NULL \end{bmatrix} \tag{4}$$

Compute the beta estimates for the data using the remaining complete observations.

$$\hat{\beta} = (X_c^T X_c)^{-1} X_c^T y \tag{5}$$

4.4 Experimental Design

Variable	Levels	Description
`sample_size`	50, 100, 500	Lots of stuff here
`noise_factor`	0.1, 0.3	Lots of stuff here
`noise_factor`	0.1, 0.3	Lots of stuff here

The objective of the experiment is to understand how