Handling Missing Data in Data Science

Simulating the effects of missing data methods and how to present the results in an interactive plot with Github Pages

Rianne Schouten[1,2]

[1]University of Utrecht, Department of Methodology and Statistics, Sjoerd Groenman building, Padualaan 14, 3584 CH Utrecht, The Netherlands
[2]Samen Veilig Midden-Nederland, Information Management Team, Tiberdreef 8, 3561 GG Utrecht, The Netherlands

Keywords: Commit2Data, Missing Data Methodology, Simulation, Github Pages

Abstract ICT.OPEN2018

Purpose

When you work with data, it is most likely you will encounter missing data. Especially in application-oriented fields, available datasets are almost always incomplete. As most analyses require complete data, what to do with the missing values?

An often used and easy solution to handle missing data is deleting incomplete rows and/or columns from the dataset. Is this method the most appropriate one, though? What other methods are available and what is their influence on the outcome of an analysis?

To gain more insights into the effects of missing data methods on model evaluation error metrics such as MSE and R^2, we performed an extensive simulation study. We applied several missing data methods under various circumstances and calculated the evaluation error metrics as one would do in a real data science application.

Because of the large number of paramters, the simulation resulted in a huge amount of outcomes. In order to make all these results available to everyone, we built an interactive plot with Highcharts. Github Pages turned out to be the perfect medium to make the plot available to everyone.

Methods

Simulation

We repeated the following procedure for every combination of parameters:

1. Split a given dataset into 60% training data and 40% testset

2. Generate missing values according one of five mechanisms (MCAR, MARZ, MNARZ, MARX, MNARX) and with several missingness percentages (5%, 10%, ..., 55%)

3. Apply a missing data method on the training data. In this simulation, we tested six methods: listwise deletion, mean imputation, median imputation, random imputation, regression imputation and stochastic regression imputation.

4. Fit a linear regression model on the completed training data.

5. Apply same missing data method as in step 3 on the test data. Then, apply the fitted regression model on the completed testset.

6. Evaluate the performance of the regression model by calculating MSE, RMSE, MAE, R^2, and the MSE difference between training and test data.

We saved and reported the average and IQR of the evaluation error metrics as calculated in step 6. For four real datasets, we used 1000 replications of the procedure described above. For four simulated datasets, 20 replications were enough.

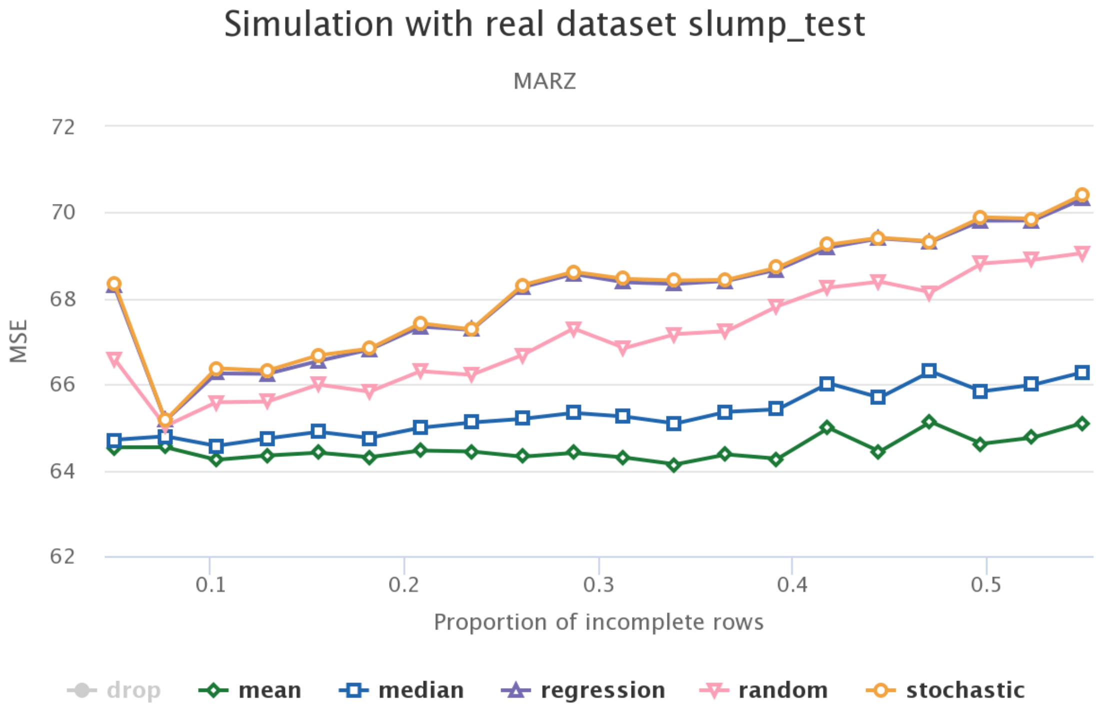
Presentation

We use Highcharts to present the simulation results in an interactive plot. By means of javascript and Highcharts we are able to select the required columns and rows for a particular setting of the interactive plot. An HTML file contains the interactive buttons and loads the Highcharts libraries and javascript documents.

Running the HTML file is done with Github Pages. To make this work, one creates a repository with a folder called /docs. In this folder, all javascript, css, text and other files are stored in another folder. The HTML file is saved as index.html and is the only file directly available. In the settings of the repository, one activates Github Pages and the interactive plot is available!

Results

Click on the figure below to load the interactive plot.


Simulation with real dataset slump_test
MARZ
MSE
Proportion of incomplete rows
drop
mean
median
regression
random
stochastic