Optimized Linear Imputation

Yehezkel S. Resheff[1,2] and Daphna Weinshall[1]
[1]*School of Computer Science and Engineering, The Hebrew University, Jerusalem, Israel*
[2]*Edmond and Lily Safra Center for Brain Sciences, The Hebrew University, Jerusalem, Israel*
{*heziresheff, daphna*}*@cs.huji.ac.il*

Keywords: Imputation.

Abstract: Often in real-world datasets, especially in high dimensional data, some feature values are missing. Since most data analysis and statistical methods do not handle gracefully missing values, the first step in the analysis requires the imputation of missing values. Indeed, there has been a long standing interest in methods for the imputation of missing values as a pre-processing step. One recent and effective approach, the IRMI stepwise regression imputation method, uses a linear regression model for each real-valued feature on the basis of all other features in the dataset. However, the proposed iterative formulation lacks convergence guarantee. Here we propose a closely related method, stated as a single optimization problem and a block coordinate-descent solution which is guaranteed to converge to a local minimum. Experiments show results on both synthetic and benchmark datasets, which are comparable to the results of the IRMI method whenever it converges. However, while in the set of experiments described here IRMI often diverges, the performance of our methods is shown to be markedly superior in comparison with other methods.

1 INTRODUCTION

Missing data imputation is an important part of data preprocessing and cleansing (Horton and Kleinman, 2007; Pigott, 2001), since the vast majority of commonly applied supervised machine learning and statistical methods for classification rely on complete data (García-Laencina et al., 2010). The most common option for many applications is to discard complete records in which there are any missing values. This approach is insufficient for several reasons: first, when missing values are not missing at random (Little, 1988; Heitjan and Basu, 1996), discarding these records may bias the resulting analysis (Little and Rubin, 2014). Other limitations include the loss of information when discarding the entire record. Furthermore, when dealing with datasets with either a small number of records or a large number of features, omitting complete records when any feature value is missing may result in insufficient data for the required analysis.

Early methods for data imputation include methods for replacing a missing value by the mean or median of the feature value across records (Engels and Diehr, 2003; Donders et al., 2006). While these values may indeed provide a “good guess” when there is no information present, this is often not case. Namely, for each missing feature value there are other non-missing values in the same record. It is likely therefore (or indeed, we assume) that other features contain information regarding the missing feature, and imputation should therefore take into account known feature values in the same record. This is done by subsequent methods.

Multiple imputation (see (Rubin, 1996) for a detailed review) imputes several sets of missing values, drawn from the posterior distribution of the missing values under a given model, given the data. Subsequent processing is then to be performed on each version of the imputed data, and the resulting multiple sets of model parameters are combined to produce a single result. While extremely useful in traditional statistical analysis and public survey data, it may not be feasible in a machine learning setting. First, the run-time cost of performing the analysis on several copies of the full-data may be prohibitive. Second, being a model-based approach it depends heavily on the type and nature of the data, and can’t be used as an out-of-the-box pre-processing step. More importantly though, while traditional model parameters may be combined between versions of the imputed data (regression coefficients for instance), many modern machine learning methods do not produce a representation that is straightforward to combine (consider the

Resheff, Y. and Weinshal, D.
Optimized Linear Imputation.
DOI: 10.5220/0006092900170025
In *Proceedings of the 6th International Conference on Pattern Recognition Applications and Methods (ICPRAM 2017)*, pages 17-25
ISBN: 978-989-758-222-6

parameters of an Artificial Neural Network or a Random Forest for example [1]).

In (Raghunathan et al., 2001), a method for imputation on the basis of a sequence of regression models is introduced. This method, popularized under the acronym MICE (Buuren and Groothuis-Oudshoorn, 2011; Van Buuren and Oudshoorn, 1999), uses a non-empty set of complete features which are known in all the records as its base, and iteratively imputes one feature at a time on the basis of the completed features up to that point. Since each step produces a single complete feature, the number of iterations needed is exactly the number of features that have a missing value in at least one record. The drawbacks of this method are twofold. First, there must be at least one complete feature to be used as the base. More importantly though, the values imputed at the $i-th$ step can only use a regression model that includes the features which were originally full or those imputed in the $i-1$ first steps. Ideally, the regression model for each feature should be able to use all other feature values.

The IRMI method (Templ et al., 2011) goes one step further by building a sequence of regression models for each feature that can use all other feature values as needed. This iterative method initially uses a simple imputation method such as median imputation. In each iteration it computes for each feature the linear regression model based on all other feature values, and then re-imputes the missing values based on these regression models. The process is terminated upon convergence or after a per-determined number of iterations (Algorithm 1). The authors state that although they do not have a proof of convergence, experiments show fast convergence in most cases.

In Section 2 we present a novel method of Optimized Linear Imputation (OLI). The OLI method is related in spirit to IRMI in that it performs a linear regression imputation for the missing values of each feature, on the basis of all other features. Our method is defined by a single optimization objective which we then solve using a block coordinate-descent method. Thus our method is guaranteed to converge, which is its most important advantage over IRMI. We further show that our algorithm may be easily extended to use any form of regularized linear regression.

In Section 3 we ompare the OLI method to the IRMI, MICE and Median Imputation (MI) methods. Using the same simulation studies as in the original IRMI paper, we show that the results of OLI are rather similar to the results of IRMI. With real datasets we show that our method usually outperforms the alternatives MI and MICE in accuracy, while providing comparable results to IRMI. However, IRMI did not converge in many of these experimentsm while our method always provided good results.

2 OLI METHOD

2.1 Notation

We start by listing the notation used throughout the paper.

N Number of samples

d Number of features

$x_{i,j}$ The value of the $j-th$ feature in the $i-th$ sample

$m_{i,j}$ Missing value indicators:

$$m_{i,j} = \begin{cases} 1 & x_{i,j} \ is \ missing \\ 0 & otherwise \end{cases}$$

m_i Indicator vector of missing values for for the $i-th$ feature

The following notation is used in the algorithms' pseudo-code:

$A[m]$ The rows of a matrix (or column vector) A where the boolean mask vector m is $True$

$A[!m]$ The rows of a matrix (or column vector) A where the boolean mask vector m is $False$

linear_regression(X, y) A linear regression from the columns of the matrix X to the target vector y, having the following fields:

.parameters: parameters of the fitted model.

.predict(X): the target column y as predicted by the fitted model.

2.2 Optimization Problem

We formulate the linear imputation as a single optimization problem. First we construct a design matrix:

$$X = \left[\begin{array}{cc} & 1 \\ [x_{i,j}(1-m_{i,j})] & \vdots \\ & 1 \end{array} \right] \qquad (1)$$

[1] In this case it would be perhaps more natural to train the model using data pooled over the various copies of the completed data rather than train separate models and average the resulting parameters and structure. This is indeed done artificially in methods such as denoinsing neural nets (Vincent et al., 2010), and has been known to be useful for data imputation (Duan et al., 2014).

Algorithm 1: The IRMI method for imputation of real-valued features (see (Templ et al., 2011) for more details).

input:

- X - data matrix of size $N \times (d+1)$ containing N samples and d features
- m - missing data mask
- max_iter - maximal number of iterations

output:

- Imputation values

```
$\tilde{X} := median\_impute(X)$ {assigns each missing value the median of its column}
while not converged and under $max\_iter$ iterations do
    for i := 1...d do
        regression = linear_regression($\tilde{X}_{-i}[!m_i]$,$\tilde{X}_i[!m_i]$)
        $\tilde{X}_i[m_i]$ = regression.predict($\tilde{X}_{-i}[m_i]$)
    end for
end while
return $\tilde{X} - X$
```

where the constant-1 rightmost column is used for the intercept terms in the subsequent regression models. Multiplying the data values $x_{i,j}$ by $(1 - m_{i,j})$ simply sets all missing values to zero, keeping non-missing values as they are.

Our approach aims to find consistent missing value imputations and regression coefficients as a single optimization problem. By consistent we mean that (a) the imputations are the values obtained by the regression formulas, and (b) the regression coefficients are the values that would be computed after the imputations. We propose the following optimization formulation:

$$\begin{cases} \min\limits_{A,M} & ||(X+M)A-(X+M)||_F^2 \\ s.t. & m_{i,j}=0 \Rightarrow M_{i,j}=0 \\ & M_{i,d+1}=0\ \forall i \\ & A_{i,i}=0 \quad i=1...d \\ & A_{i,d+1}=\delta_{i,d+1} \quad \forall i \end{cases} \tag{2}$$

where $||.||_F$ is the Frobenius norm.

Intuitively, the objective that we minimize measures the square error of reconstruction of the imputed data $(X+M)$, where each feature (column) is approximated by a linear combination of all other features plus a constant (that is, linear regression of the remaining imputed data). The imputation process by which M is defined is guaranteed to leave the non-missing values in X intact, by the first and second constraints which make sure that only missing entries in X have a corresponding non-zero value in M. Therefore:

$$(X+M) = \begin{cases} M & for\ missing\ values \\ X & for\ non\ missing\ values \end{cases}$$

The regression for each feature is further constrained to use only **other** features, by setting the diagonal values of A to zero (the third constraint). The forth constraint makes sure that the constant-1 rightmost column of the design matrix is copied as-is and therefore does not impact the objective.

We note that all the constraints set variables to constant values, and therefore this can be seen as an unconstrained optimization problem on the remaining set of variables. This set includes the non-diagonal elements of A and the elements of M corresponding to missing values in X. We further note that this is not a convex problem in A, M since it contains the MA factor. In the next section we show a solution to this problem that is guaranteed to converge to a local minimum.

2.3 Block Coordinate Descent Solution

We now develop a coordinate descent solution for the proposed optimization problem. Coordinate descent (and more specifically alternating least squares; see for example (Hope and Shahaf, 2016)) algorithms are extremely common in machine learning and statistics, and while don't guarantee convergence to a global optimum, they often preform well in practise.

As stated above, our problem is an unconstrained optimization problem over the following set of variables:

$$\{A_{i,j} | i,j = 1,..,d; i \neq j\} \cup \{M_{i,j} | m_{i,j} = 1\}$$

Keeping this in mind, we use the following objective function:

$$L(A,M) = ||(X+M)A-(X+M)||_F^2 \tag{3}$$

$$= \sum_{i=1}^{d} ||(X+M)_{-i}\beta_i - (X+M)_i||_F^2 \tag{4}$$

Algorithm 2: Optimized Linear Imputation (OLI).

input:

- X_0 - data matrix of size $N \times d$ containing N samples and d features
- m- missing data mask

output:

- Imputation values

```
$X := median\_impute(X_0)$
$M := zeros(N,d)$
$A := zeros(d,d)$
while not converged do
    for $i := 1...d$ do
        $\beta := linear\_regression(X_{-i}, X_i).parameters$
        $A_i := [\beta_1, ..., \beta_{i-1}, 0, \beta_i, ..., \beta_d]^T$
    end for‘
    while not converged do
        $M := M - \alpha[(X+M)A - (X+M)](A-I)^T$
        $M[!m] := 0$
    end while
    $X := X + M$
end while
return $M$
```

where C_{-i} denotes the matrix C without its $i-th$ column, C_i the $i-th$ column, and β_i the $i-th$ column of A without the $i-th$ element (recall that the $i-th$ element of the $i-th$ column of A is always zero). The term $(X+M)_{-i}\beta_i$ is therefore a linear combination of all but the $i-th$ column of the matrix $(X+M)$. The sum in (4) is over the first d columns only, since the term added by the rightmost column is zero (see fourth constraint in (2)).

We now suggest the following coordinate descent algorithm for the minimization of the objective (3) (the method is summarized in Algorithm 2):

1. Fill in missing values using median/mean (or any other) imputation
2. Repeat until convergence:
   (a) Minimize the objective (3) w.r.t. A (compute the columns of the matrix A)
   (b) Minimize the objective (3) w.r.t. M (compute the missing values entries in matrix M)
3. Return M [2]

As we will show shortly, step (a) in the iterative part of the proposed algorithm reduces to calculating the linear regression for each feature on the basis of all other features, essentially the same as the first step in the IRMI algorithm (Templ et al., 2011) Algorithm 1. Step (b) can be solved either as a system of linear equations or in itself as an iterative procedure, by gradient descent on (3) w.r.t M using (5).

First, we show that step (a) reduces to linear regression. Taking the derivatives of (4) w.r.t the non-diagonal elements of column i of A we have:

$$\frac{\partial L}{\partial \beta_i} = 2(X+M)^T_{-i}[(X+M)_{-i}\beta_i - (X+M)_i]$$

Setting the partial derivatives to zero gives:

$$(X+M)^T_{-i}[(X+M)_{-i}\beta_i - (X+M)_i]=0$$
$$\Rightarrow \beta_i = ((X+M)^T_{-i}(X+M)_{-i})^{-1}(X+M)^T_{-i}(X+M)_i$$

which is exactly the linear regression coefficients for the $i-th$ feature from all other (imputed) features, as claimed.

Next, we obtain the derivatives of the objective function w.r.t M:

$$\nabla_M = \frac{\partial L}{\partial M} = 2[(X+M)A - (X+M)](A-I)^T \quad (5)$$

leading to the following gradient descent algorithm for step (b): step (b), Repeat until convergence:

(i) $M := M - \alpha\nabla_M L(A,M)$

(ii) $\forall_{i,j}: \ M_{i,j} = M_{i,j}m_{i,j}$

where α is a predefined step size and the gradient is given by (5). Step (ii) makes sure that only missing

[2] Alternatively, in order to stay close in spirit to the linear IRMI method, we may prefer to use $(X+M)A$ as the imputed data.

values are assigned imputation values[3].

Our proposed algorithm uses a gradient descent procedure for the minimization of the objective (3) w.r.t M. Alternatively, one could use a closed form solution by directly setting the partial derivative to zero. More specifically, let

$$\frac{\partial L}{\partial M}=0 \tag{6}$$

Substituting (5) into (6), we get

$$M(A-I)(A-I)^T = -X(A-I)(A-I)^T$$

which we rewrite as:

$$MP = Q \tag{7}$$

with the appropriate matrices P,Q. Now, since only elements of M corresponding to missing values of X are optimization variables, only these elements must be set to zero in the derivative (6), and hence only these elements must obey the equality (7). Thus, we have:

$$(MP)_{i,j} = Q_{i,j} \;\; \forall i,j|m_{i,j} = 1$$

which is a system of $\sum\limits_{i,j} m_{i,j}$ linear equations in $\sum\limits_{i,j} m_{i,j}$ variables.

2.4 Discussion

In order to better understand the difference between the IRMI and OLI methods, we rewrite the IRMI iterative method (Templ et al., 2011) using the same notation as used for our method. We start by defining an error matrix:

$$E = (X+M)A-(X+M)$$

E is the error matrix of the linear regression models on the basis of the imputed data. Unlike our method, however, IRMI considers the error only in the non-missing values of the data, leading to the following objective function:

$$L(M,A) = \sum_{i,j|m_{i,j}=0} E^2_{i,j}$$

In order to minimize this loss function, at each step the IRMI method (Algorithm 1) optimizes over a single column of A (which in effect reduces to fitting a single linear regression model), and then assigns as the missing values in the corresponding column of M the values predicted for it by the regression model. While this heuristic for choosing M is quite effective, it is **not** a gradient descent step and it therefore leads to a process with unknown convergence properties. The main motivation for proposing our method was to fix this shortcoming within the same general framework and propose a method that is similar in spirit, with a convergence guarantee.

Another advantage of the proposed formulation is the ability to easily extend it to any regularized linear regression. This can be done by re-writing the itemized form of the objective (4) as follows:

$$L(A,M) = \sum_i [||(X+M)_{-i}\beta_i - (X+M)_i||^2_F + \Omega(\beta_i)]$$

where $\Omega(\beta_i)$ is the regularization term.

Now, assuming that the resulting regression problem can be solved (that is, minimizing each of the summands in the new objective with a constant M), and since step (b) of our method remains exactly the same (the derivative w.r.t M does not change as the extra term does not depend on M), we can use the same method to solve this problem as well.

Another possible extension is to use kernelized linear regression. This may be useful in cases when the dependencies between the features are not linear. Here too we can use the same type of method of optimization, but we defer to future research working out the details of the derivative w.r.t M, which will obviously not remain the same.

The method of initialization is another issue deserving further investigation. Since our procedure converges to a local minimum of the objective, it may be advantageous to start the procedure from several random initial points, and choose the best result. However, since the direct target (missing values) are obviously unknown, we would need an alternative measure of the "goodness" of a result. Since the missing values are assumed to be missing at random, it would make sense to use the distance between the distributions of known and imputed values (per feature) as a measure of appropriateness of an imputation.

3 EXPERIMENTS

In order to evaluate our method, we compared its performance to other imputation methods using various types of data. We used complete datasets (real or synthetic), and randomly eliminated entries in order to simulate the missing data case. To evaluate the success of each imputation method, we used the mean

[3] Note that this is not a projection step. Recall that the optimization problem is only over elements M_{ij} where x_{ij} is a missing value, encoded by $m_{ij} = 1$. The element-wise multiplication of M by m guarantees that all other elements of M are assigned 0. Effectively, the gradient descent procedure does not treat them as independent variables, as required.

square error (MSE) of the imputed values as a measure of error. MSE is computed as the mean square distance between stored values (the correct values for the simulated missing values) and the imputed ones.

In Section 3.1 we repeat the experimental evaluation from (Templ et al., 2011) using synthetic data, in order to compare the results of our method to the results of IRMI. In Section 3.2 we compare our method to 3 other methods - IRMI, MI and MICE - using standard benchmark datasets from the UCI repository (Lichman, 2013) . In Section 3.3 we augment the comparisons with an addition new reallife dataset of storks migration data.

For some real datasets in the experiments described below we report that the IRMI method did not converge (and therefore did not return any result). This decision was reached when the MSE of the IRMI method rose at least 6 orders of magnitude throughout the allocated 50 iterations, or (when tested with unlimited iterations) when it rose above the maximum valid number in the system of approximately $1e+308$.

3.1 Synthetic Data

The following simulation studies follow (Templ et al., 2011) and compare OLI to IRMI. All simulations are repeated 20 times with $10,000$ samples. 5% of all values across records are selected at random and marked as missing. Values are stored for comparison with imputed values. Simulation data is multivariate normal with mean of 1 in all dimensions. Unless stated otherwise, the covariance matrix has 1 in its diagonal entries and 0.7 in the off-diagonal entries.

The aim of the first experiment is to test the relationship between the actual values imputed by the IRMI and OLI methods. The simulation is based on multivariate normal data with 5 dimensions. Results show that the values imputed by the two methods are highly correlated (Fig. 1a). Furthermore, the signed error ($original - imputed$) is also highly correlated (Fig. 1b). Together, these findings point to the similarity in the results these two methods produce.

In the next simulation we test the performance of the two methods as we vary the number of features. The simulation is based on multivariate normal data with $3-20$ dimensions. The results (Fig. 2b) show almost identical behavior of the IRMI and OLI algorithms, which also coincides with the results presented for IRMI in (Templ et al., 2011). Median imputation (MI) is also shown for comparison as baseline. Fig 3 shows a zoom into a small segment of figure 2.

As expected, imputing the median (which is also the mean) of each feature for all missing values results in an MSE equal to the standard deviation of the features (i.e., 1). While very close, the IRMI and the OLI methods do not return the exact same imputation values and errors, with an average absolute deviation of 0.053

Next we test the performance of the two methods as we vary the covariance between the features. The simulation is based on multivariate normal data with 5 dimensions. Non-diagonal elements of the covariance matrix are set to values in the range $0.1-0.9$. The results (Fig. 2a) show again almost identical behavior of the IRMI and OLI algorithms. As expected, when the dependency between the feature columns is increased, which is measure by the covariance between the columns (X-axis in Fig. 2a), the performance of the regression-based methods IRMI and OLI is monotonically improving, while the performance of the MI method remain unaltered.

3.2 UCI Datasets

The UCI machine learning repository (Lichman, 2013) contains several popular benchmark datasets, some of which have been previously used to compare methods of data imputation (Schmitt et al., 2015). In the current experiment we used the following datasets: *iris* (Fisher, 1936), *wine* (white) (Cortez et al., 2009), *Ecoli* (Horton and Nakai, 1996), *Boston housing* (Harrison and Rubinfeld, 1978), and *power* (Tüfekci, 2014). Each feature of each dataset was normalized to have mean 0 and standard deviation of 1, in order to make error values comparable between datasets. Categorical features were dropped. For each dataset, 5% of the values were chosen at random and replaced with a missing value indicator. The procedure was repeated 10 times. For these datasets we also consider the MICE method (Buuren and Groothuis-Oudshoorn, 2011) using the *winMice* (Jacobusse, 2005) software.

The results are quite good, demonstrating the superior ability of the linear methods to impute missing data in these datasets (Table 1, rows 1-5). In the Iris dataset our OLI method achieved an average error identical to IRMI, which successfully converged only 9 out of the 10 runs. Both outperformed the MI and MICE standard methods. In the Ecoli dataset both the IRMI and OLI methods performed worse than the alternative methods, with MICE achieving the lowest MSE. In the Wine dataset the IRMI failed to converge in all 10 repetitions, while the OLI method outperformed the MI and MICE methods. The IRMI method outperformed all other methods in the Housing dataset, but failed to converge 7 out of 10 times

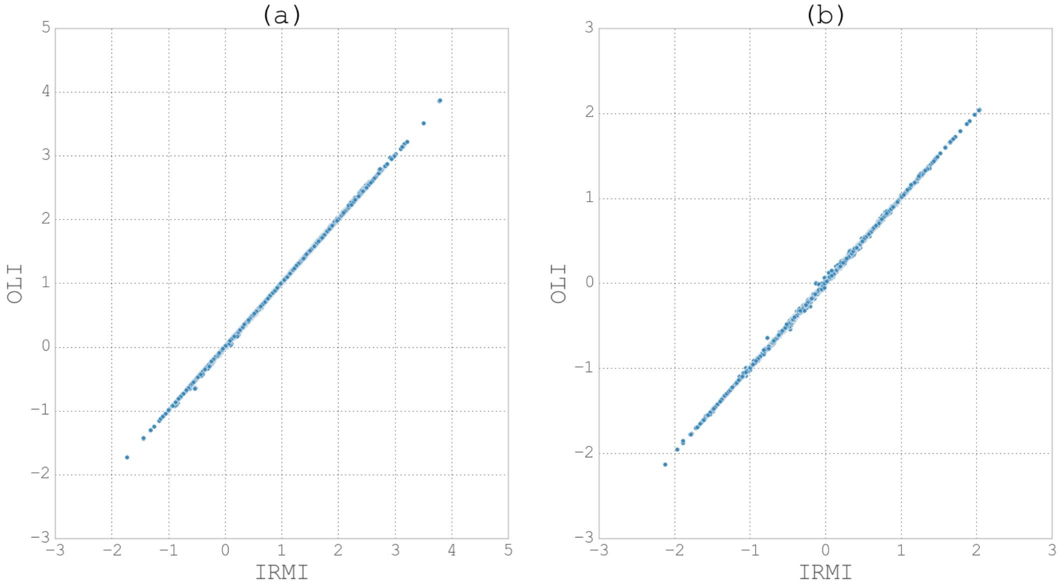

Figure 1: (a) Correlation between predicted values for missing data using the IRMI and OLI methods. (b) Correlation between the signed error of the prediction for the two methods.

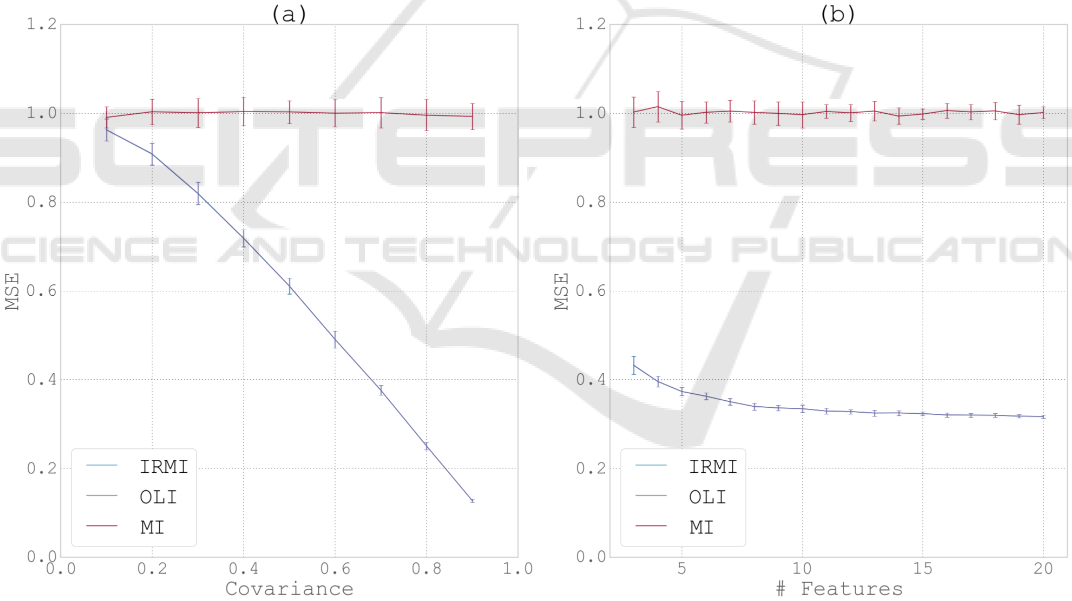

Figure 2: (a) MSE of the IRMI, OLI and MI methods as a function of the covariance. Data is 5 dimensional multivariate normal. (b) MSE of the IRMI, OLI and MI methods as a function of the dimensionality, with a constant covariance of 0.7 between pairs of features. In both cases error bars represent standard deviation over 20 repetitions.

for the Power dataset.

In summary, in cases where the linear methods were appropriate, with sufficient correlation between the different features (shown in the second column of Table 1), the proposed OLI method was comparable to the IRMI method with regard to mean square error of the imputed values when the latter converged, and superior in that it always converges and therefore always returns a result. While the IRMI method achieved slightly better results than OLI in some cases, its failure to converge in others gives the OLI method the edge. Overall, better results were achieved for datasets with high mean correlation between features, as expected when using methods utilizing the linear relationships between features.

Table 1: Comparison of the imputation results of the IRMI, OLI, MICE and MI methods with 5% missing data. The *converged* column indicates the number of runs in which the IRMI method converged during testing; the MSE of IRMI was calculated for converged repetitions only.

Dataset	# Features	correlation	IRMI		OLI	MI	MICE
			converged	MSE			
Iris	4	0.59	9/10	**0.20**	**0.20**	1.00	0.33
Ecoli	7	0.18	9/10	8.26	5.75	1.72	**1.20**
Wine	11	0.18	**0/10**	-	**0.87**	1.05	1.10
Housing	11	0.45	10/10	**0.28**	0.30	1.14	0.56
Power	4	0.45	**3/10**	0.44	**0.47**	1.02	0.88
Storks	20	0.24	**0/10**	-	**0.31**	1.07	0.42

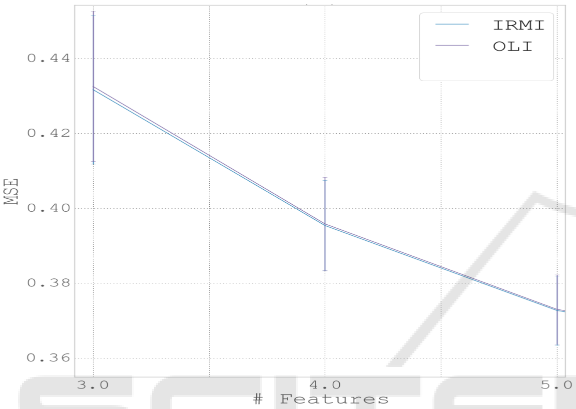

Figure 3: Zoom into a small part of figure 2.

3.3 Storks Behavioral Modes Dataset

In the field of Movement Ecology, readings from accelerometers placed on migrating birds are used for both supervised (Resheff et al., 2014) and unsupervised (Resheff et al., 2015)(Resheff et al., 2016) learning of behavioral modes. In the following experiment we used a dataset of features extracted from 3815 such measurements. As with the UCI datasets, 10 repetitions were performed, each with 5% of the values randomly selected and marked as missing. Results (Table 1, final row) of this experiment highlight the relative advantage of the OLI method. While the IRMI method failed to converge in all 10 repetitions, OLI achieved an average MSE considerably lower than the MI baseline, and also outperformed the MICE method.

4 CONCLUSION

Since the problem of missing values often haunts real-word datasets while most data analysis methods are not designed to deal with this problem, imputation is a necessary pre-processing step whenever discarding entire records is not a viable option. Here we proposed an optimization-based linear imputation method that augments the IRMI (Templ et al., 2011) method with the property of guaranteed convergence, while staying close in spirit to the original method. Since our method converges to a local optimum of a different objective function, the two methods should not be expected to converge to the same value exactly. However, simulation results show that the results of the proposed method are generally similar (nearly identical) to IRMI when the latter does indeed converge.

The contribution of our paper is twofold. First, we suggest an optimization problem based method for linear imputation and an algorithm that is guaranteed to converge. Second, we show how this method can be extended to use any number of methods of regularized linear regression. Unlike matrix completion methods (Wagner and Zuk, 2015), we do not have a low rank assumption. Thus, OLI should be preferred when data is expected to have some linear relationships between features and when IRMI fails to converge, or alternatively, when a guarantee of convergence is important (for instance in automated processes). We leave to future research the kernel extension of the OLI method.

REFERENCES

Buuren, S. and Groothuis-Oudshoorn, K. (2011). mice: Multivariate imputation by chained equations in r. *Journal of statistical software*, 45(3).

Cortez, P., Cerdeira, A., Almeida, F., Matos, T., and Reis, J. (2009). Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems*, 47(4):547–553.

Donders, A. R. T., van der Heijden, G. J., Stijnen, T., and Moons, K. G. (2006). Review: a gentle introduction to imputation of missing values. *Journal of clinical epidemiology*, 59(10):1087–1091.

Duan, Y., Yisheng, L., Kang, W., and Zhao, Y. (2014). A deep learning based approach for traffic data imputation. In *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*, pages 912–917. IEEE.

Engels, J. M. and Diehr, P. (2003). Imputation of missing longitudinal data: a comparison of methods. *Journal of clinical epidemiology*, 56(10):968–976.

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7(2):179–188.

García-Laencina, P. J., Sancho-Gómez, J.-L., and Figueiras-Vidal, A. R. (2010). Pattern classification with missing data: a review. *Neural Computing and Applications*, 19(2):263–282.

Harrison, D. and Rubinfeld, D. L. (1978). Hedonic housing prices and the demand for clean air. *Journal of environmental economics and management*, 5(1):81–102.

Heitjan, D. F. and Basu, S. (1996). Distinguishing missing at random and missing completely at random. *The American Statistician*, 50(3):207–213.

Hope, T. and Shahaf, D. (2016). Ballpark learning: Estimating labels from rough group comparisons. *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 299–314.

Horton, N. J. and Kleinman, K. P. (2007). Much ado about nothing. *The American Statistician*, 61(1).

Horton, P. and Nakai, K. (1996). A probabilistic classification system for predicting the cellular localization sites of proteins. In *Ismb*, volume 4, pages 109–115.

Jacobusse, G. (2005). Winmice users manual. *TNO Quality of Life, Leiden. URL http://www. multiple-imputation. com*.

Lichman, M. (2013). UCI machine learning repository.

Little, R. J. (1988). A test of missing completely at random for multivariate data with missing values. *Journal of the American Statistical Association*, 83(404):1198–1202.

Little, R. J. and Rubin, D. B. (2014). *Statistical analysis with missing data*. John Wiley & Sons.

Pigott, T. D. (2001). A review of methods for missing data. *Educational research and evaluation*, 7(4):353–383.

Raghunathan, T. E., Lepkowski, J. M., Van Hoewyk, J., and Solenberger, P. (2001). A multivariate technique for multiply imputing missing values using a sequence of regression models. *Survey methodology*, 27(1):85–96.

Resheff, Y. S., Rotics, S., Harel, R., Spiegel, O., and Nathan, R. (2014). Accelerater: a web application for supervised learning of behavioral modes from acceleration measurements. *Movement ecology*, 2(1):25.

Resheff, Y. S., Rotics, S., Nathan, R., and Weinshall, D. (2015). Matrix factorization approach to behavioral mode analysis from acceleration data. In *Data Science and Advanced Analytics (DSAA), 2015. 36678 2015. IEEE International Conference on*, pages 1–6. IEEE.

Resheff, Y. S., Rotics, S., Nathan, R., and Weinshall, D. (2016). Topic modeling of behavioral modes using sensor data. *International Journal of Data Science and Analytics*, 1(1):51–60.

Rubin, D. B. (1996). Multiple imputation after 18+ years. *Journal of the American statistical Association*, 91(434):473–489.

Schmitt, P., Mandel, J., and Guedj, M. (2015). A comparison of six methods for missing data imputation. *Journal of Biometrics & Biostatistics*, 2015.

Templ, M., Kowarik, A., and Filzmoser, P. (2011). Iterative stepwise regression imputation using standard and robust methods. *Computational Statistics & Data Analysis*, 55(10):2793–2806.

Tüfekci, P. (2014). Prediction of full load electrical power output of a base load operated combined cycle power plant using machine learning methods. *International Journal of Electrical Power & Energy Systems*, 60:126–140.

Van Buuren, S. and Oudshoorn, K. (1999). Flexible multivariate imputation by mice. *Leiden, The Netherlands: TNO Prevention Center*.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., and Manzagol, P.-A. (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *The Journal of Machine Learning Research*, 11:3371–3408.

Wagner, A. and Zuk, O. (2015). Low-rank matrix recovery from row-and-column affine measurements. *arXiv preprint arXiv:1505.06292*.