Journal of Classification (2020) 37:42–57
https://doi.org/10.1007/s00357-019-9304-3

Pointed Subspace Approach to Incomplete Data

Lukasz Struski[1] · Marek Śmieja[1] · Jacek Tabor[1]

Published online: 28 March 2019

Abstract
Incomplete data are often represented as vectors with filled missing attributes joined with flag vectors indicating missing components. In this paper, we generalize this approach and represent incomplete data as pointed affine subspaces. This allows to perform various affine transformations of data, such as whitening or dimensionality reduction. Moreover, this representation preserves the information, which coordinates were missing. To use our representation in practical classification tasks, we embed such generalized missing data into a vector space and define the scalar product of embedding space. Our representation is easy to implement, and can be used together with typical kernel methods. Performed experiments show that the application of SVM classifier on the proposed subspace approach obtains highly accurate results.

Keywords Incomplete data · SVM · Linear transformations · Imputation · Missing values

1 Introduction

Incomplete data analysis is an important part of data engineering and machine learning, since it appears in many practical problems. In medical diagnosis, a doctor may be unable to complete the patient examination due to the deterioration of health status or lack of patient's compliance (Burke et al. 1997); in object detection, the system has to recognize the shape from low resolution or corrupted images (Berg et al. 2005); in chemistry, the complete analysis of compounds requires high financial costs (Stahura and Bajorath 2004). In consequence, the understanding and the appropriate representation of such data is of great practical importance.

A missing data is typically viewed as a pair (x, J_x), where $x \in \mathbb{R}^N$ is a vector with missing components $J_x \subset \{1, \ldots, N\}$. In the most straightforward approach, one can fill missing attributes with some statistic, e.g., mean, taken from existing data. Although such a strategy can be partially justified when the features are missing at random, we lose the

✉ Marek Śmieja
marek.smieja@ii.uj.edu.pl

[1] Faculty of Mathematics and Computer Science, Jagiellonian University, Lojasiewicza 6, 30-348 Kraków, Poland

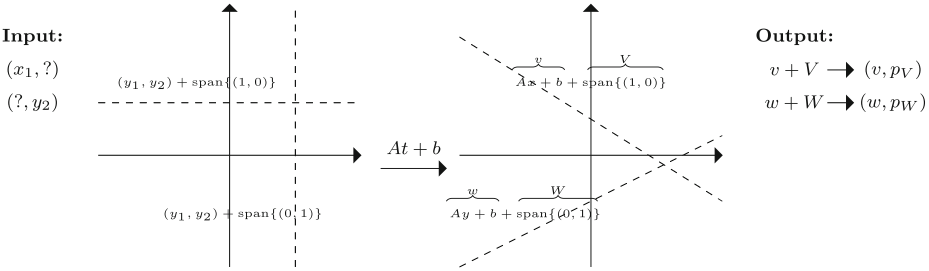

Input:
$(x_1, ?)$
$(?, y_2)$
$(y_1, y_2) + \mathrm{span}\{(1,0)\}$
$(y_1, y_2) + \mathrm{span}\{(0,1)\}$
$At + b$
$\underbrace{Ax + b}_{v} + \underbrace{\mathrm{span}\{(1,0)\}}_{V}$
$\underbrace{Ay + b}_{w} + \underbrace{\mathrm{span}\{(0,1)\}}_{W}$
Output:
$v + V \longrightarrow (v, p_V)$
$w + W \longrightarrow (w, p_W)$

Fig. 1 Representation of incomplete data as pointed subspaces, their affine transformation, and final embedding as projections onto subspaces

knowledge about unknown attributes[1]. To preserve this information, we usually add a flag indicating which components were missing. More precisely, we supply x with a binary vector $\mathbb{1}_{J_x}$, in which 1 denotes absent feature while 0 means the present one (Fig. 1).

Summarizing, we perform the embedding $(x, J_x) \to (x, \mathbb{1}_{J_x})$ of missing points into a vector space of extended complete data. This allows us to apply typical classification tools, like SVM, with the scalar product defined by the following:

$$\langle (x, J_x), (y, K_y) \rangle = \langle x, y \rangle + \langle \mathbb{1}_{J_x}, \mathbb{1}_{K_y} \rangle. \tag{1}$$

In practical classification problems, we usually perform various affine transformations of data, as whitening or dimensionality reduction, before training a classifier. Moreover, we may know that the data satisfy some affine constraint. It is nontrivial how to modify the flag vectors so as to keep the correspondence with such affine transformations. Thus, our main problem behind the paper can be stated as follows:

> *How to transform the flag vectors indicating the missing components if we perform the linear (or affine) mapping of data?*

In this contribution, we show that the answer can be given by viewing the incomplete data as pointed affine subspaces, i.e., the subspace with a distinguished point called basepoint. We first observe that a pair (x, J_x) can be formally associated with a pointed affine subspace of $\mathbb{R}^N$:

$$x + \mathrm{span}(e_j)_{j \in J_x},$$

where $(e_j)_{j=1}^N$ denotes the canonical base of $\mathbb{R}^N$ and x is a selected basepoint. In other words, this is the set of all points which coincide with the representative x on the coordinates different from J_x. In consequence, by a *generalized missing data point* in $\mathbb{R}^N$, we understand a pointed affine subspace $S_x = x + V$ of $\mathbb{R}^N$, where $x \in \mathbb{R}^N$ is a basepoint and $V = S_x - x$ is a linear subspace.

Such a definition allows us to efficiently extend linear and affine operations from the standard points to missing ones, by taking the image of the subspace and the point. For example, a linear mapping $F : w \to Aw + b$, can be extended to the case of pointed subspace $x + V$ by the following:

$$F(x + V) = F(x) + AV.$$

[1] In the medical data, typically some component is missing if the state of the patient is so bad, that a given numerical procedure cannot be performed. Consequently, the knowledge that given component is missing could say a lot about the state of the patient.

Given an affine constraint W, we restrict[2] $x + V$ by the formula $(x + V) \cap W = x + (V \cap (W - x))$.

There appears another question: how to work with such data and in particular how to embed the generalized missing data into a vector space in such a way to respect the scalar product (1) given by the flag embedding. Our main observation shows that this can be achieved by identifying a linear subspace V with an orthogonal projection $p_V : \mathbb{R}^N \to V$ by considering the embedding $(x, V) \to (x, p_V) \in \mathbb{R}^N \times \mathbb{R}^{N \times N}$. We show that the scalar product of embeddings coincides with (1), i.e.,

$$\langle (x, \mathbb{1}_{J_x}), (y, \mathbb{1}_{K_y}) \rangle = \langle (x, p_{\mathrm{span}(e_J : j \in J_x)}), (y, p_{\mathrm{span}(e_k : k \in K_y)}) \rangle.$$

The above scalar product strictly depends on the selection of basepoint. Although a basepoint can be chosen with use of any imputation technique, most of them are very simple and do not take into account intrinsic characteristic of data. In this contribution, we propose a new imputation method, which uses the most likely values to fill absent attributes. Given a mean m and a covariance Σ estimated from incomplete data, our imputation for $x + V$ is defined as a projection p_V^Σ of m onto the subspace $x + V$ with respect to the Mahalanobis scalar product given by Σ:

$$x - p_V^\Sigma (x - m).$$

The above formula uses data distribution to find the most likely values for missing features.

The paper is organized as follows. The next section covers related approaches to incomplete data analysis. In Section 3, we define a generalized missing data point and present how to embed such data into a vector space. Next, we propose a new imputation strategy and define a scalar product for generalized data points. The connections between our approach and existing flag representation is described at the end of Section 3. In Section 4, we examine the performance of our method in SVM classification experiments.

2 Related Works

Imputation is one of the most popular techniques for dealing with missing data (McKnight et al. 2007). Once missing attributes are filled, classical machine learning methods can be applied on complete data (Little and Rubin 2014). The simplest approaches relies on replacing missing features by a mean values, zeros, or by taking the average value from neighbors of a given point. To account the influence of missing attributes on classification accuracy (Liu et al. 2016) proposed an adaptive imputation strategy. In Conversano and Siciliano (2009), a nonparametric approach to missing data imputation was introduced, which iteratively fills missing values with use of tree-based method. D'Ambrosio et al. (2012) added some methodological improvements to this techniques in terms of accuracy and computational efficiency, and extended the approach used for missing data imputation also to data fusion. Although, the imputation-based techniques are easy to use for practitioners, they lead to the loss of information which features were missing and do not take into account the reasons of missingness. To preserve the information of missing attributes, one can use an additional vector of binary flags, which was discussed in the introduction (Chechik et al. 2007).

If absent attributes are missing at random (MAR), then one can estimate a distribution on incomplete data space with use of EM algorithm (Ghahramani and Jordan 1994; Schafer

[2] Observe that if such a constraint W is given the augmentation of the missing components must be performed in such a way as to choose the representation in W, and consequently we may assume that $x \in W$.

1997). Then, it is possible to generate the most probable values from obtained probability distribution for missing attributes (McKnight et al. 2007). Multiple imputation allows to draw several candidates for filling missing attributes without estimating density on the whole space (Azur et al. 2011; Sulis and Porcu 2017). Density model can also be used for a direct tuning of classifiers. This option was already investigated in the case of linear regression (Williams et al. 2005a), kernel methods (Smola et al. 2005; Williams and Carin 2005b), or by using second-order cone programming (Shivaswamy et al. 2006). Generalized RBF was constructed on incomplete data set using the information of data distribution (Smieja et al. 2017). One can also estimate the parameters of the probability model and the classifier jointly, which was considered in Dick et al. (2008) and Liao et al. (2007).

In Chechik et al. (2008), a modified SVM classifier is trained by individual scaling the margin according to observed features. The alternative approaches to learning a linear classifier, which avoid features deletion or imputation, are presented in Dekel et al. (2010) and Globerson and Roweis (2006). In Grangier and Melvin (2010), the embedding mapping of feature-value pairs is constructed together with a classification objective function. Pelckmans et al. (2005) modeled the expected risk of prediction using incomplete data. Hazan et al. (2015) designed a kernel method, which performs comparably to the classifier which have an access to complete data, under low-rank assumption. Goldberg et al. (2010) considered a transductive classification problem and used matrix completion strategy to fill both missing attributes and absent class labels.

In our contribution, we generalize the imputation-based techniques in such a way to preserve the information of missing features. To select a basepoint, we propose to choose the most likely point form a subspace identifying a missing data point; however, other imputation methods can be used as well. Constructed representation allows to apply various affine data transformations preserving classical scalar product before applying typical classification methods.

3 Generalized Incomplete Data

In this section, we introduce the subspace approach to incomplete data. First, we define a generalized missing data point, which allows to perform affine transformation of incomplete data. Then, we show how to embed generalized missing data into a vector space and select a basepoint. Finally, we define a scalar product on the embedding space.

3.1 Incomplete Data as Pointed Affine Subspaces

Incomplete data X can be understood as a sequence of pairs (x_i, J_i), where $x_i \in \mathbb{R}^N$ and $J_i \subset \{1, \ldots, N\}$ indicates missing coordinates of x_i. Therefore, we can associate a missing data point (x, J) with an affine subspace $x + \text{span}(e_j)_{j \in J}$, where $(e_j)_j$ is the canonical base of $\mathbb{R}^N$. Let us observe that $x + \text{span}(e_j)_{j \in J}$ is a set of all N-dimensional vectors which coincide with x on the coordinates different from J.

In this paper, we focus on transforming incomplete data by affine mappings. For this purpose, we generalize the above representation to arbitrary affine subspaces, or more precisely pointed affine subspaces, which do not have to be generated by canonical bases.

Definition 1 A generalized missing data point is defined as a pointed affine subspace $S_x = x + V$, where $x \in \mathbb{R}^N$ is a basepoint and $V = S_x - x$ is a linear subspace of $\mathbb{R}^N$.

A basepoint can be selected by filling missing attributes with a use of any imputation method, which will be discussed in the next subsection.

Remark 1 Observe that the notion of pointed affine subspace differs from classical affine subspace. In particular, pointed subspace depends on the selection of basepoint. In consequence, we can create two different generalized missing data points S_y, S_z from the same missing data point (x, J) by using different imputation methods.

First, we show that the above definition is useful for defining linear mappings on incomplete data. Let $S_x = x + V$ be a generalized missing data point and let $f : \mathbb{R}^N \ni w \to Aw + b$ be an affine map. We can transform a generalized missing data point $x + V$ into another missing data point by the formula:

$$f(x + V) = \{Aw + b : w \in x + V\}.$$

The basepoint x is mapped into $Ax + b$, while the linear part of $f(x + V)$ is given by the following:

$$f(x + V) - f(x) = AV.$$

Consequently, we arrive at the definition:

Definition 2 For a generalized missing data point $S_x = x + V$ and an affine mapping $f : w \to Aw + b$ we put:

$$f(x + V) = (Ax + b) + AV,$$

where $Ax + b$ is a basepoint and AV is a linear subspace.

One can easily compute and represent AV, if the orthonormal base $v_1, \dots, v_n$ of V is given, namely we simply orthonormalize the sequence $Av_1, \dots, Av_n$.

3.2 Embedding of Generalized Missing Data and Basepoint Selection

The above representation is useful for understanding and performing affine transformations of incomplete data, such as whitening, dimensionality reduction or incorporating affine constraints to data. Nevertheless, typical machine learning methods require vectors or a kind of kernel (or similarity) matrix as the input. We show how to embed generalized missing data into a vector space.

A generalized missing data point $S_x = x + V$ consists of a basepoint $x \in \mathbb{R}^N$ which is an element of vector space and a linear subspace V. To represent a subspace V, we propose to use a matrix of orthogonal projection p_V onto V. To get an exact form of p_V, let us assume that $(v_j)_{j \in J}$ is an orthonormal base of V. Then, the projection of $y \in \mathbb{R}^N$ can be calculated by the following:

$$p_V(y) = \sum_{j \in J} \langle y, v_j \rangle v_j = \sum_{j \in J} v_j v_j^T y = (\sum_{j \in J} v_j v_j^T) y,$$

which implies as follows;

$$p_V = \sum_{j \in J} v_j v_j^T.$$

Summarizing, our embedding is defined as follows:

Definition 3 A generalized missing data point is embedded in a vector space by the following:

$$S_x \to (x, p_V) \in \mathbb{R}^N \times \mathbb{R}^{N \times N},$$

where $S_x = x + V$ and x is a basepoint.

The selection of basepoint relies on filling missing attributes with some concrete values, which is commonly known as imputation. In our setting, by the imputation we denote a function $\Phi : X \to \mathbb{R}^N$ such that

$$\Phi(S_x) \in S_x,$$

for a generalized missing data S_x.

In the case of classical incomplete data, missing attributes are often filled with a mean or a median calculated from existing values for a given attribute. However, these imputations cannot be easily defined in a general case, because the linear part of generalized missing data point might be an arbitrary linear subspace (not necessarily a subspace generated by a subset of canonical base). Let us observe that another popular imputation method which fills the missing coordinates with zeros can be defined for generalized incomplete data. This is performed by selecting a basepoint of an incomplete data point $S_x = x + V$ as the orthogonal projection of missing data x onto the subspace orthogonal to V, i.e.,

$$x_{V^\perp} = x - p_V(x) = x - \sum_{j \in J} \langle x, v_j \rangle v_j, \tag{2}$$

where $(v_j)_{j \in J}$ is an arbitrary orthonormal base of V. If V is represented by canonical base, then this is equivalent to filling missing attributes with zeros.

The above imputations do not take into account intrinsic characteristic of data such its distribution. Therefore, it is highly probable that filled values are far from the correct ones. To overcome this problem, we propose another technique for setting missing values, which extends zero imputation method. Let us assume that (m, Σ) are the mean and the covariance matrix estimated for incomplete dataset X. To find the most probable values for missing data points $x + V$, we calculate the orthogonal projection of m onto $x + V$ with respect to the Mahalanobis scalar product parametrized by Σ, which we call *the most likely point imputation*:

Definition 4 Let m and Σ be a mean and covariance matrix estimated from the incomplete data set X. By the most likely point imputation for incomplete data points $x + V$, where $(v_j)_{j \in J}$ defines orthogonal base of V, we understand a projection:

$$x_V^{(m,\Sigma)} = x - p_V^{m,\Sigma}(x), \tag{3}$$

where $p_V^{(m,\Sigma)}(x) = \sum_{j \in J} \langle x - m, v_j \rangle_\Sigma \cdot v_j$ is the projection matrix onto V with respect to the Mahalanobis scalar product $\langle \cdot, \cdot \rangle_\Sigma$.

To obtain the values for m and Σ in practice, one can use a sample mean and a covariance matrix calculated using existing attributes of incomplete data. Alternatively, if data satisfy missing at random assumption, then the EM algorithm can be applied to estimate the probability model describing data (Schafer 1997).

The above most likely point imputation is the imputation based on two first moments—mean and covariance. Since it takes into account the correlation between components, it is able to predict more reliable estimates than the mean imputation, which is based only on the first moment. Regression imputation is an example of imputation, which also uses two first moments (Aste et al. 2015). It is commonly used in multiple imputation procedure, where absent values are iteratively filled (Azur et al. 2011; Buuren and Groothuis-Oudshoorn 2011). In order to fill missing attributes, a regression imputation solves a mean square error problem for every coordinate. In contrast, our most likely point imputation finds the closest point to the mean of data set with respect to the Mahalanobis norm induced by a data set covariance. Two first moments are also used in other estimators derived from maximum likelihood approach using, e.g., EM algorithm (Aste et al. 2015). Clearly, there are also more flexible imputation techniques, which use various machine learning methods (García-Laencina et al. 2010).

Example 1 To illustrate the effect of missing data imputation and transformation, let us consider the whitening operation:

$$\text{Whitening}(x) = \Sigma^{-1/2}(x - m),$$

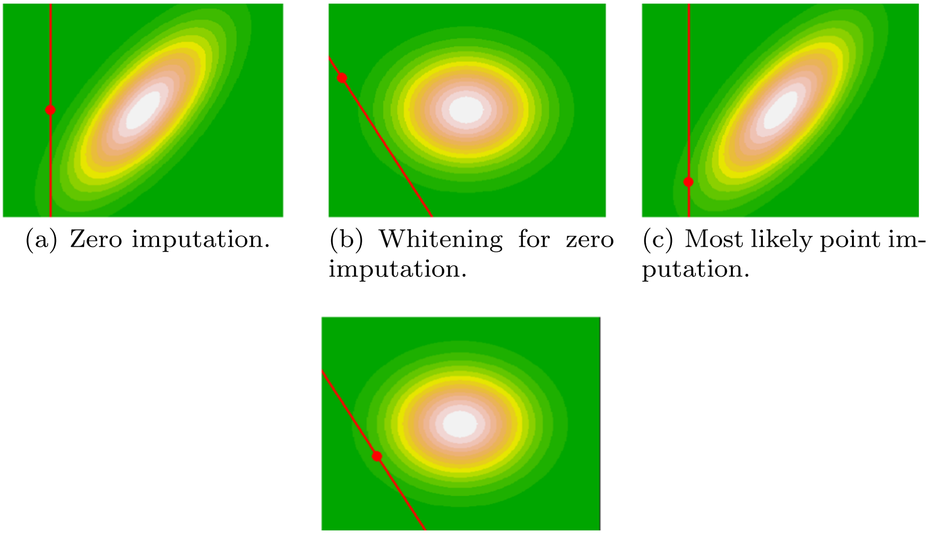
where Σ is the covariance, and m the mean of X. For a generalized missing data, the above operation is defined by the following:

$$\text{Whitening}(x + V) = \Sigma^{-1/2}(x - m) + \Sigma^{-1/2}V.$$

In other words, we map a basepoint in a classical way and transform a subspace V into a linear subspace $\Sigma^{-1/2}V$. The illustration is given in Fig. 2.

Example 2 In the case of high dimensional data, we sometimes reduce a dimension of input data space by applying the principle component analysis, which is defined by the following:

$$\text{PCA}(x) = W^T(x - m),$$

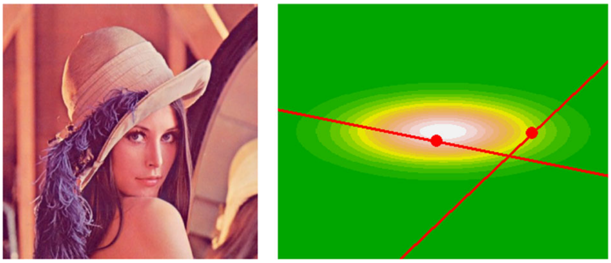

(a) Zero imputation.

(b) Whitening for zero imputation.

(c) Most likely point imputation.

(d) Whitening for most likely point.

Fig. 2 Whitening of data with a single element containing one missing attribute. Missing feature was filled with zero (**a**), (**b**) or most likely point imputation (**c**), (**d**)



(a) Image. (b) 2D projection.

Fig. 3 The image (**a**) with two missing pixels and its projection onto two principal components (**b**). Image was represented by the feature vectors consisting of 8 × 8 blocks. Missing pixels are identified by the pointed subspaces with basepoints chosen by zero imputation strategy

where m is a mean of a dataset and k columns of W are the leading eigenvectors of covariance matrix Σ. This operation can be extended to the case of generalized missing data by the following:

$$\mathrm{PCA}(x + V) = W^T(x - m) + W^T V.$$

An example of the above operation is illustrated in the Fig. 3.

3.3 Scalar Product for SVM Kernel

To apply most of classification methods, it is necessary to define a scalar product (kernel matrix) on a data space. As a natural choice, one could sum the scalar products between basepoints and embedding matrices, i.e.,

$$\langle x + V, y + W\rangle = \langle x, y\rangle + \langle p_V, p_W\rangle. \qquad (4)$$

However, for a data space of dimension N, we have $\|p_V\|^2 = N$, which implies that the weight of projection can dominate the first part of (4) concerning basepoints. Consequently, we decided to introduce an additional parameter to allow reducing the importance of projection part:

Definition 5 Let $D \in [0, 1]$ be fixed. As a scalar product between two generalized missing data points, we put:

$$\langle x + V, y + W\rangle_D = \langle x, y\rangle + D\langle p_V, p_W\rangle. \qquad (5)$$

Let us observe that the above parametric scalar product can be implemented by taking the embedding $x + V \to (x, \sqrt{D} p_V)$ and then using formula (4) for a scalar product.

Remark 2 Observe that the value of function (5) strictly depends on the selection of basepoints, which makes it a not well-defined scalar product in the space of classical affine subspaces. Indeed, $x + V$ defines the same affine subspace as $x + v + V$, where $v \in V$, but such shifts may lead to different values of the right hand side of (5). However, this is a well-defined scalar product in the case of pointed affine subspaces, because two different selections of basepoints give different pointed affine subspaces (see Remark 1). In consequence, it might be safely used in the case of generalized missing data points considered in this paper.

The following proposition shows how to calculate a scalar product between matrices defining two orthogonal projections onto linear subspaces.

Proposition 1 *Let us consider subspaces*

$$V = span(v_j : j \in J), W = span(w_j : j \in K).$$

where v_j and w_k are orthonormal sequences. If p_V, p_W denote orthogonal projections onto V, W, respectively, then

$$\langle p_V, p_W \rangle = \sum_{j \in J, k \in K} \langle v_j, w_k \rangle^2.$$

Proof By the definition of orthogonal projections and the scalar product between matrices, we have as follows:

$$\langle p_V, p_W \rangle = \sum_{j \in J, k \in K} \mathrm{tr}((v_j v_j^T)^T (w_k w_k^T)). \tag{6}$$

Making use of $\mathrm{tr}(AB) = \mathrm{tr}(BA)$, we get

$$\mathrm{tr}((v_j v_j^T)^T (w_k w_k^T)) = \mathrm{tr}(v_j v_j^T w_k w_k^T) = \mathrm{tr}(v_j^T w_k w_k^T v_j) = (v_j^T w_k) \cdot (w_k^T v_j) = \langle v_j, w_k \rangle^2.$$

Finally,

$$\langle p_V, p_W \rangle = \sum_{j \in J, k \in K} \langle v_j, w_k \rangle^2.$$

□

Concluding, the scalar product between embedding of two generalized missing data points given by Definition 5 can be calculated as follows:

$$\langle x + V, y + W \rangle_D = \langle x, y \rangle + D \sum_{i,j} (p_V)_{ij} (p_W)_{ij} = \langle x, y \rangle + D \sum_{j \in J, k \in K} \langle v_j, w_k \rangle^2,$$

where $(v_j)_{j \in J}$, $(w_k)_{k \in K}$ are orthonormal bases of V, W, respectively. The last expression can be more numerically efficient if the dimension of the subspaces (the number of missing attributes) is much smaller than the dimension of the whole space.

Remark 3 One of typical representations of missing data (x, J) relies on filling unknown attributes and supplying it with a binary flag vector $\mathbb{1}_J \in \mathbb{R}^N$, in which bit 1 denotes coordinate belonging to J. This leads to the embedding of the missing data into a vector space given by the following:

$$(x, J) \to (x, \mathbb{1}_J) \in \mathbb{R}^N \times \mathbb{R}^N.$$

Then, the scalar product of such embedding can be defined by the following:

$$\langle (x, \mathbb{1}_J,)(y, \mathbb{1}_K) \rangle = \langle x, y \rangle + \langle \mathbb{1}_J, \mathbb{1}_K \rangle = \langle x, y \rangle + \mathrm{card}(J \cap K). \tag{7}$$

It is worth noting that the formula (7) coincides with a scalar product defined for generalized missing data (4) (for $D = 1$). Indeed, if $V = \mathrm{span}(e_j : j \in J)$ and $W = \mathrm{span}(e_k : k \in K)$, for $J, K \subset \{1, \ldots, N\}$, then by Proposition 1, we have as follows:

$$\langle p_V, p_W \rangle = \sum_{j \in J, k \in K} \langle e_j, e_k \rangle^2 = \sum_{l \in J \cap K} \langle e_l, e_l \rangle^2 = \sum_{l \in J \cap K} 1 = \mathrm{card}(J \cap K),$$

which is exactly the RHS of (7).

Table 1 Summary of data sets used in the experiments

Data	# Instances	# Attributes
Australian	690	14
Bank	1372	4
Breast cancer	699	8
Crashes	540	18
Heart	270	13
Liver disorders	345	6
Pima	768	8
Phoneme	5404	5
Ring	7400	20
Sonar	208	60

Therefore, our approach generalizes and theoretically justifies the flag approach to missing data analysis. The importance of our construction lies in its generality, which in particular allows for performing typical affine transformations of data. In the case of flag representation, there is no obvious solution how to perform such mappings on flag vector.

4 Experiments

To verify our pointed subspace approach, we applied it in SVM classification experiments. We examined how a missing data mechanism influences its classification results. Moreover, we tested various imputation strategy for a basepoint selection. Since SVM classifiers works better on normalized data, we applied a whitening to scale the feature vectors to unit covariance (on a training set) as described in Definition 2.

4.1 Experimental Setting

If not stated otherwise, for a basepoint selection we used the most likely point imputation (3). This parametrization of our method will be denoted by *most likely*+. Its results were compared with classical imputation strategy, which also fills absent coordinates using most likely point imputation, referred to as *most likely*. We also used multiple imputation by chained equation (mice) (Buuren and Groothuis-Oudshoorn 2011) implemented in R package mice[3]. Additionally, we considered two specialized SVM classifiers, which do not use any direct imputation. The first one is a geometric max-margin classifier, *geom*, proposed in Chechik et al. (2008), which focuses on maximizing the margin for each example in its own relevant subspace determined by visible features. The second technique, referred to as *karma*, is an iterative SVM algorithm based on low-rank assumption (Hazan et al. 2015). Finally, we used an extension of classical RBF kernel to the case of incomplete data (Smieja et al. 2017), which is denoted by *genRBF*. Since most data sets are not linearly separable, every method was combined with RBF kernel.

All experiments assumed double fivefold cross-validation. More precisely, for every division into train and test sets, the required hyperparameters were tuned using inner fivefold cross-validation applied on training set. The combination of parameters maximizing mean

[3] https://cran.r-project.org/web/packages/mice/index.html

Table 2 Classification accuracy of examined methods on UCI data sets, where missing entries satisfy MAR

Data set	Most likely	Most likely+	mice	geom	karma	genRBF
Australian	0.822	0.821	0.819	0.785	0.814	*0.850*
Breast cancer	0.955	0.953	0.952	0.936	0.944	*0.957*
Bank	0.900	*0.917*	0.885	0.769	0.907	0.909
Crashes	*0.923*	*0.923*	0.918	0.914	0.907	0.917
Heart	0.811	0.811	0.808	0.796	0.818	*0.819*
Liver disorders	*0.617*	0.607	0.610	0.573	0.587	0.598
Pima	0.741	0.743	0.740	0.705	0.743	*0.746*
Phoneme	0.751	*0.793*	0.731	0.718	0.778	*0.793*
Ring	0.950	*0.970*	0.956	0.936	0.964	0.957
Sonar	0.612	0.589	0.587	0.636	0.670	*0.676*

accuracy score (on validation set) was used to learn a final classifier on a entire training set, while the performance was evaluated on a test set (it was not used during training). The accuracy was averaged over all five trails. All data set was scaled using a whitening operator applied on train set. Every method tuned a standard margin parameter $C \in \{10^k : k = -1, 0, \ldots, 8\}$ and the width of RBF radius $\gamma \in \{10^k : k = -8, -7, \ldots, 1\}$. Additionally, our method selected a trade-off parameter D from the range $\{0.1, 0.2, \ldots, 1\}$, while *karma* was parametrized by the factor $\gamma_{\text{karma}} \in \{1, 2, 3\}$.

We considered examples from UCI repository summarized in Table 1, where selected attributes were removed based on two strategies. In the first one, attributes were removed at random, while in the second case missing features were determined by the class labels. Details of these strategies are described in the following subsections.

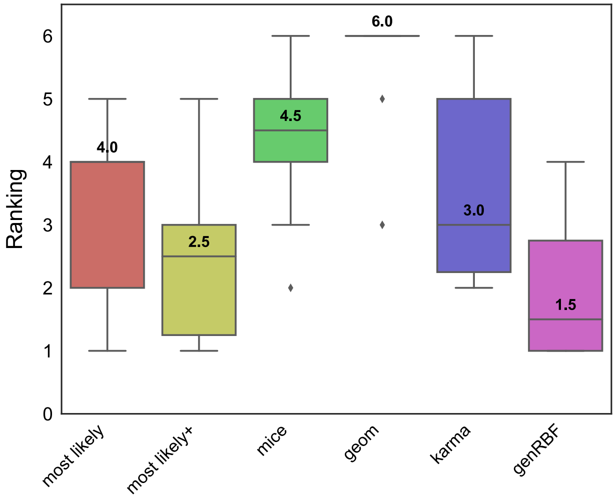


Fig. 4 Box plots of ranks for MAR removing strategy (the lower the better)

4.2 Missing at Random

In the first experiment, we considered a removing process following MAR assumption, where absence of attributes depends on visible features. For this purpose, we drawn N points $x_1, \ldots, x_N$ of a dataset $X \subset \mathbb{R}^N$. Then, for every $x \in X$, we removed its ith attribute with a probability $\exp(-t\|x - x_i\|_\Sigma))$, where $\|x\|_\Sigma$ denotes the Mahalanobis norm of x with respect to Σ. In other words, the removal of ith attribute was determined by a distance to ith selected point. Parameter $t > 0$ was chosen to remove approximately 30% of attributes.

Table 2 shows that our method usually provided one of the highest results. It significantly outperformed comparative methods on ring data set, while in other cases, except sonar data, had similar accuracy to the best techniques. However, in many cases, the differences between the results were slight. Since missing coordinates were removed at random, the use of additional information about absent features was not a significant benefit over other methods.

To summarize the results, we ranked examined methods over all data sets. The best performing technique on a given data set got rank 1, second best got rank 2, etc. Box plot of ranks presented in Fig. 4 shows that our approach was the second best performing method on average, just after *genRBF*. In particular, it was better than typical imputation method (Table 3).

4.3 Type of Imputation

Our approach can use various imputation strategies for basepoint selection. In this experiment, we examined the influence of the type of imputation on the classification results.

Our subspace representation was combined with three imputation strategies. The first one, *zero*+, fills missing attributes with zeros. The second, *mean*+, uses the mean value of a given attribute, while the last one is the proposed most likely imputation, *most likely*+. The experiment used the same data sets as in previous case.

The summary of the results, presented in Fig. 5, show that most likely imputation was the best option in most cases. It also confirms statistical differences between the results.

4.4 Structural Removal

The first experiment suggested that the advantage of using subspace approach in MAR case is slight. However, in real situation, the removal process may be much more complex. In

Table 3 Score accuracy for two strategy removing data: MAR (left side) and beta distribution with parameters $\alpha = 1, \beta = 1.5$ (right side)

Data set	Zero+	Mean+	Most likely+
Australian	0.808	0.820	*0.821*
Breast cancer	0.949	0.951	*0.953*
Bank	0.914	0.914	*0.917*
Crashes	0.911	0.922	*0.923*
Heart	0.780	0.805	*0.811*
Liver disorders	0.592	0.602	*0.607*
Pima	0.738	0.734	*0.743*
Phoneme	0.784	0.781	*0.793*
Ring	0.959	0.967	*0.970*
Sonar	0.589	*0.603*	0.589

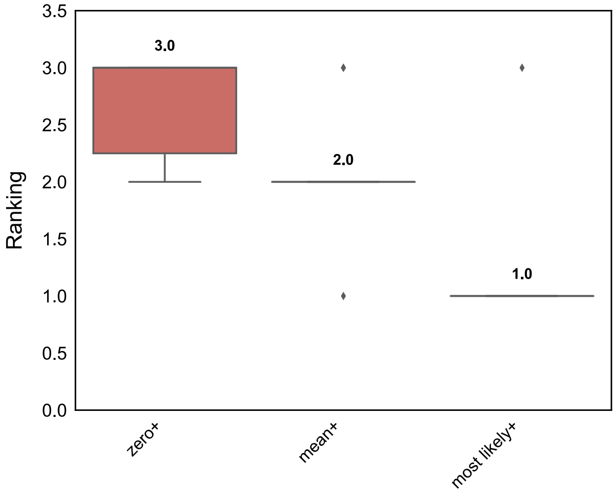


Fig. 5 Box plots of ranks for MAR strategy removing data

particular, the absence of some features may be determined by a corresponding class label. For instance, a doctor may be unable to complete patient examination due to the bad health status, which results in missing values. In consequence, the inability of measurements is influenced by a target value describing the illness.

Following the above motivation, we constructed a process for attribute removal, where the chance of attribute absence is induced by a class label. We considered two densities of beta distributions f_- and f_+ on [0, 1] with parameters:

- $f_- : \alpha = 1; \beta = 1.5$,
- $f_+ : \alpha = 1.5; \beta = 1$,

which are illustrated in Fig. 6. A density f_- is related with negative class, while positive class is connected with f_+. Given an example $x \in X$ taken from negative class, we

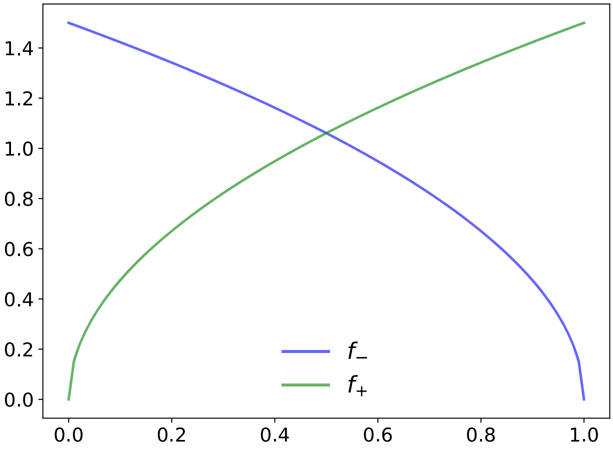


Fig. 6 Two densities of beta distributions f_- (with parameters: $\alpha = 1, \beta = 1.5$) and f_+ (with parameters $\alpha = 1.5; \beta = 1$) on [0, 1]

Table 4 Classification accuracy, when attributes removal was partially determined by a corresponding class labels

Data set	Most likely	Most likely+	mice	geom	karma	genRBF
Australian	0.836	*0.867*	0.836	0.800	0.845	0.843
Breast cancer	0.953	*0.955*	0.951	0.952	0.943	0.950
Bank	0.955	*0.966*	0.942	0.872	0.965	0.952
Crashes	0.919	*0.936*	0.916	0.913	0.929	0.924
Heart	0.815	*0.829*	0.802	0.808	0.826	0.827
Liver disorders	0.684	*0.687*	0.685	0.584	0.680	0.662
Pima	0.740	*0.791*	0.735	0.701	0.775	0.770
Phoneme	0.823	*0.851*	0.811	0.763	0.836	0.838
Ring	0.960	*0.973*	0.961	0.886	0.965	0.966
Sonar	0.772	*0.892*	0.620	0.594	0.614	0.812

remove its ith attribute with a probability $\int_{\frac{i-1}{N}}^{\frac{i}{N}} f_-(x)dx$. If x belongs to positive class its ith attribute is removed with probability $\int_{\frac{i-1}{N}}^{\frac{i}{N}} f_+(x)dx$. In other words, the values at first few coordinates are more likely to be removed for negative class, while these at last coordinates are usually absent for positive class. We would like to mention that the removal process is not deterministic and we selected α, β to allow high overlap between these two removal processes.

The classification results for such prepared data are presented in Table 4. It is clear that the subspace approach obtained the highest performance among all considered methods. Typical imputations strategies as *most likely* and *mice*, fill absent features and forget which one were missing, which leads to the loss of information. The ranking of methods presented in Fig. 7 confirms high performance of our approach. It shows that *karma* and *genRBF* were better than imputation strategies, but cannot performed as well as our method in this case.

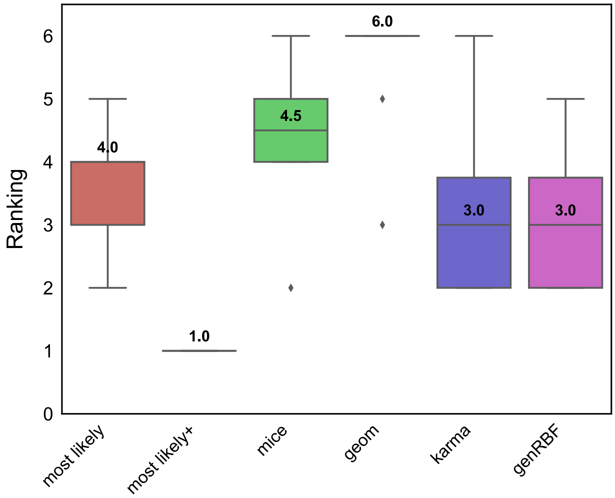


Fig. 7 Box plots, when attributes removal was partially determined by a corresponding class labels

4.5 Summary

In real-life problems, absence of attributes is usually determined by both random and deterministic processes. For example, some sensors may fail randomly, but also may stop registering a signal, when a device begins to deteriorate. Experimental results showed that our approach performs well in both cases; in random situations, it gives comparable results to other methods, but it can also successfully use the information about missing coordinates. These make our method a promising tool for analyzing incomplete data.

5 Conclusion

The paper generalized the existing approach of identifying missing attributes with binary flags. To enable appropriate affine transformations of data, we represented incomplete data as pointed affine subspaces and embedded them into a vector space by linking a pointed subspace with a basepoint joined with a corresponding projection matrix. In the same spirit, we proposed to select a basepoint as the most probable point from a subspace, which extends the well-known zero imputation strategy. Such a combination provided satisfactory performance in conducted classification experiments.

Acknowledgments This work was partially supported by the National Science Centre (Poland) grant no. 2015/19/B/ST6/01819.

Funding Information This work was partially supported by the National Science Centre (Poland) grant no. 2015/19/B/ST6/01819 .

References

Aste, M., Boninsegna, M., Freno, A., Trentin, E. (2015). Techniques for dealing with incomplete data: a tutorial and survey. *Pattern Analysis and Applications*, *18*(1), 1–29.

Azur, M.J., Stuart, E.A., Frangakis, C., Leaf, P.J. (2011). Multiple imputation by chained equations: what is it and how does it work? *International Journal of Methods in Psychiatric Research*, *20*(1), 40–49.

Berg, A.C., Berg, T.L., Malik, J. (2005). Shape matching and object recognition using low distortion correspondences. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (pp. 26–33). IEEE.

Burke, L.E., Dunbar-Jacob, J.M., Hill, M.N. (1997). Compliance with cardiovascular disease prevention strategies: a review of the research. *Annals of Behavioral Medicine*, *19*(3), 239–263.

Buuren, S., & Groothuis-Oudshoorn, K. (2011). mice: multivariate imputation by chained equations in R. *Journal of statistical software*, *45*(3), 1–68.

Chechik, G., Heitz, G., Elidan, G., Abbeel, P., Koller, D. (2007). Max-margin classification of incomplete data. In *Advances in Neural Information Processing Systems* (pp. 233–240).

Chechik, G., Heitz, G., Elidan, G., Abbeel, P., Koller, D. (2008). Max-margin classification of data with absent features. *Journal of Machine Learning Research*, *9*, 1–21.

Conversano, C., & Siciliano, R. (2009). Incremental tree-based missing data imputation with lexicographic ordering. *Journal of Classification*, *26*(3), 361–379.

D'Ambrosio, A., Aria, M., Siciliano, R. (2012). Accurate tree-based missing data imputation and data fusion within the statistical learning paradigm. *Journal of classification*, *29*, 1–32.

Dekel, O., Shamir, O., Xiao, L. (2010). Learning to classify with missing and corrupted features. *Machine Learning*, *81*(2), 149–178.

Dick, U., Haider, P., Scheffer, T. (2008). Learning from incomplete data with infinite imputations. In: *Proceedings of the International Conference on Machine Learning* (pp. 232–239). ACM.

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

Ghahramani, Z., & Jordan, M.I. (1994). Supervised learning from incomplete data via an EM approach. In *Advances in Neural Information Processing Systems* (pp. 120–127). Citeseer.

Globerson, A., & Roweis, S. (2006). Nightmare at test time: robust learning by feature deletion. In *Proceedings of the International Conference on Machine Learning* (pp. 353–360). ACM.

Goldberg, A., Recht, B., Xu, J., Nowak, R., Zhu, X. (2010). Transduction with matrix completion: three birds with one stone. In *Advances in neural information processing systems* (pp. 757–765).

Grangier, D., & Melvin, I. (2010). Feature set embedding for incomplete data. In Advances in Neural Information Processing Systems (pp. 793–801).

Hazan, E., Livni, R., Mansour, Y. (2015). Classification with low rank and missing data. In *Proceedings of The 32nd International Conference on Machine Learning* (pp. 257–266).

Liao, X., Li, H., Carin, L. (2007). Quadratically gated mixture of experts for incomplete data classification. In *Proceedings of the International Conference on Machine Learning* (pp. 553–560) ACM.

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

Liu, Z.-G., Pan, Q., Dezert, J., Martin, A. (2016). Adaptive imputation of missing values for incomplete pattern classification. *Pattern Recognition*, *52*, 85–95.

McKnight, P.E., McKnight, K.M., Sidani, S., Figueredo, A.J. (2007). *Missing data: a gentle introduction*. Guilford Press.

Pelckmans, K., De Brabanter, J., Suykens, J.A.K., De Moor, B. (2005). Handling missing values in support vector machine classifiers. *Neural Networks*, *18*(5), 684–692.

Schafer, J.L. (1997). Analysis of incomplete multivariate data. CRC Press.

Shivaswamy, P.K., Bhattacharyya, C., Smola, A.J. (2006). Second order cone programming approaches for handling missing and uncertain data. *Journal of Machine Learning Research*, *7*, 1283–1314.

Smieja, M., Struski, L., Tabor, J. (2017). Generalized RBF kernel for incomplete data. arXiv:1612.01480.

Smola, A.J., Vishwanathan, S.V.N., Hofmann, T. (2005). Kernel methods for missing variables. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*. Citeseer.

Stahura, F.L., & Bajorath, J. (2004). Virtual screening methods that complement HTS. *Combinatorial Chemistry & High Throughput Screening*, *7*(4), 259–269.

Sulis, I., & Porcu, M. (2017). Handling missing data in item response theory. assessing the accuracy of a multiple imputation procedure based on latent class analysis. *Journal of Classification*, *2*, 1–33.

Williams, D., Liao, X., Ya, X., Carin, L. (2005a). Incomplete-data classification using logistic regression. In *Proceedings of the International Conference on Machine Learning* (pp. 972–979). ACM.

Williams, D., & Carin, L. (2005b). Analytical kernel matrix completion with incomplete multi-view data. In *Proceedings of the ICML Workshop on Learning With Multiple Views*.