








Supervised Human-Guided Data Exploration

Emilia Oikarinen¹, Kai Puolamäki¹, Samaneh Khoshrou²,
and Mykola Pechenizkiy²

¹ Department of Computer Science, University of Helsinki, Helsinki, Finland
{emilia.oikarinen,kai.puolamaki}@helsinki.fi

² Department of Computer Science, Eindhoven University of Technology,
Eindhoven, The Netherlands
{s.khoshrou,m.pechenizkiy}@tue.nl

Abstract. An exploratory data analysis system should be aware of what a user already knows and what the user wants to know of the data. Otherwise it is impossible to provide the user with truly informative and useful views of the data. In our recently introduced framework for human-guided data exploration (Puolamäki et al. [20]), both the user’s knowledge and objectives are modelled as distributions over data, parametrised by tile constraints. This makes it possible to show the users the most informative views given their current knowledge and objectives. Often the data, however, comes with a class label and the user is interested only of the features informative related to the class. In non-interactive settings there exist dimensionality reduction methods, such as supervised PCA (Barshan et al. [1]), to make such visualisations, but no such method takes the user’s knowledge or objectives into account. Here, we formulate an information criterion for *supervised human-guided data exploration* to find the most informative views about the class structure of the data by taking both the user’s current knowledge and objectives into account. We study experimentally the scalability of our method for interactive use, and stability with respect to the size of the class of interest. We show that our method gives understandable and useful results when analysing real-world datasets, and a comparison to SPCA demonstrates the effect of the user’s background knowledge. The implementation will be released as an open source software library.

1 Introduction and Related Work

Exploratory data analysis (EDA) is a long studied topic [24]. More often than not, the data is so high-dimensional that it is not possible for a user to view it at once. This problem can be solved, e.g., by various *dimensionality reduction* (DR) methods that attempt to embed the data in a lower-dimensional manifold so that a chosen metrics is preserved as accurately as possible [15]. The main drawback in almost all DR methods is that the criteria by which dimensionality is reduced are often fixed, or at least it is not clear *how to take into account what the*

user already knows and what are the objectives of the user when computing the embedding; see [23] for a survey of recent work on interactive DR. EDA systems also incorporate *visual* and *interactive* components, and visual interactive EDA has applications in different contexts, e.g., in item-set mining and subgroup discovery [3, 8, 16], information retrieval [22], and network analysis [4].

One approach to incorporate the user’s knowledge to EDA is to model this as a distribution over datasets—*background distribution*—and then show the user an embedding that gives the user as much information as possible that the user did not already know. One of the original works in modelling the background distribution using randomisation was [11], and in [6] maximum entropy distributions were used. In both of these works the users can encode their knowledge as constraints. Later, these ideas have been realised as parts of working EDA systems with DR methods able to show the user what the user does not already know and able to absorb the relations the user has learned from the data, see, e.g., [5, 12, 13, 18, 19, 21, 25]. The drawback in all of these works is, however, that the *EDA process is unguided*: the user is shown something she or he does not know and what is therefore by definition always a surprise. Recently, we solved this problem in [20] by allowing the user to formulate also her or his *objectives* in terms of the relations of attributes the user is interested in. This allows the user to *guide the exploration to patterns of interest*.

Often, however, the user is not interested in all possible features of the data, but only in features that are informative, e.g., of a given class label. *Supervised DR* methods try to find an embedding that shows only the features of the data that are informative in such cases. Typical examples of supervised DR, such as Fisher’s discriminant analysis [9], metric learning [26], sufficient dimensionality reduction [10], and supervised PCA [1] are however all based on a fixed embedding criteria. User interaction in guiding data exploration has been considered in the context of database management systems, e.g., in [7], where the user tells the system which samples are relevant and which are not, allowing the system to incrementally lead the user to explore towards interesting data areas. However, to the best of our knowledge there are no earlier approaches that take into account *both the human’s subjective background knowledge and allow for supervised dimensionality reduction*.

Contributions. The objective of this work is to propose a method of supervised DR for interactive EDA systems that take both the user’s background knowledge and the user’s objectives into account. Our contributions are as follows: (i) An information criterion for *supervised human-guided data exploration*, where we can find the most informative views about the class structure of the data. (ii) An experimental study of scalability for interactive use, and stability with respect to the size of the class of interest. (iii) A demonstration showing that our method gives understandable and useful results when analysing real-world datasets.

Organisation. We provide a recap of the necessary concepts of the human-guided data exploration framework proposed in [20] in Sect. 2. In Sect. 3 we extend and modify the framework from [20] into a supervised setting. In Sect. 4 we evaluate

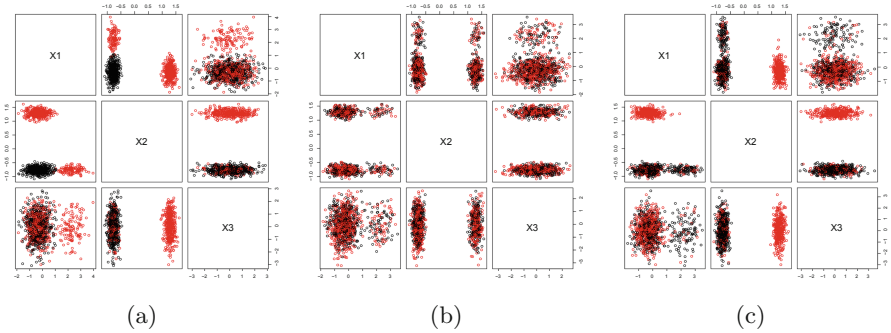


Fig. 1. Samples drawn from the distribution of datasets where each attribute has the same marginal distribution as the toy data Z_{toy} , see Examples 1 and 2 for details. Class attribute Y_{toy} shown with colour: class ‘1’ in red and class ‘-1’ in black. A sample of 1000 data points plotted for illustration. Here Z_{toy} is permuted using (a) a vector of identity permutations (i.e., the plot shows Z_{toy}) (b) a vector of random permutations (i.e., the plot shows an unconstrained permutation of Z_{toy}) (c) a vector of permutations allowed by tile t from Example 2. (Color figure online)

the scalability of our method for interactive use using crafted datasets. We also provide real-life data use cases demonstrating the utility of our method. We present our conclusions and directions for further work in Sect. 5.

2 Background

We start by introducing our notation and providing a brief recap to *human-guided data exploration* (HGDE) framework proposed in [20]. For now, we assume that X is a real-valued $n \times m$ data matrix (dataset) and $Y \in L^n$ a vector of class labels in L . Here $X(i, j)$ (resp. $Y(i)$) denotes the i th element (in column j). Each column $X(\cdot, j)$, $j \in [m]$, is an *attribute* in the dataset, where we used the shorthand $[m] = \{1, \dots, m\}$. Let $Z = (X|Y)$ denote the $n \times m'$ where $m' = m + 1$ data matrix obtained by augmenting X with Y .

A *permutation* of matrix Z is defined as follows.

Definition 1 (Permutation). Let \mathcal{P} denote the set of permutation functions of length n such that $\pi : [n] \mapsto [n]$ is a bijection for all $\pi \in \mathcal{P}$, and denote by $(\pi_1, \dots, \pi_{m'}) \in \mathcal{P}^{m'}$ the vector of column-specific permutations. A permutation \widehat{Z} of the data matrix Z is then given as $\widehat{Z}(i, j) = Z(\pi_j(i), j)$.

When permutation functions are sampled uniformly at random, we obtain a uniform sample from the distribution of datasets where each of the attributes has the same marginal distribution as the original data.

Example 1. We will use a running example throughout the paper to illustrate the main concepts. Our artificial toy data Z_{toy} consists of a three dimensional matrix $X_{toy} \in \mathbb{R}^{n \times 3}$ and a binary class attribute $Y_{toy} \in \{-1, 1\}^n$, where $n = 4000$,

shown in Fig. 1a. The matrix X_{toy} is centred and scaled to unit variance. There are 2000 data points in class ‘-1’ of Y_{toy} (coloured black in Fig. 1a) and they are clustered in the first two dimensions of X_{toy} . There are also 2000 data points in class ‘1’ of Y_{toy} (coloured red in Fig. 1a), but the points separate into two clusters (consisting of 500 points and 1500 points) in the first two dimensions of X_{toy} . The third dimension of X_{toy} is random noise for both classes.

We can produce a uniform sample from the distribution of datasets where each of the attributes has the same marginal distribution as our toy data, by sampling a vector of permutations (π_1, \dots, π_4) and permuting the toy data, see Fig. 1b for an example of such a sample. This sample represents user’s knowledge of the data if the user knows only the marginal distributions of the data but is unaware of any relations between the class and the attributes.

We will next parametrise this distribution with *tiles* preserving the relations¹ in the data matrix Z for a subset of rows and columns: a tile is a tuple $t = (R, C)$, where $R \subseteq [n]$ and $C \subseteq [m']$. In an unconstrained case, there are $(n!)^{m'}$ allowed vectors of permutations. The tiles constrain the allowed permutations as follows.

Definition 2 (Tile constraint). *Given a tile $t = (R, C)$, the vector of permutations $(\pi_1, \dots, \pi_{m'}) \in \mathcal{P}^{m'}$ is allowed by t iff the following condition is true for all $i \in [n]$, $j \in [m']$, and $j' \in [m']$:*

$$i \in R \wedge \{j, j'\} \subseteq C \implies \pi_j(i) \in R \wedge \pi_j(i) = \pi_{j'}(i).$$

Given a set of tiles T , $(\pi_1, \dots, \pi_{m'})$ is allowed iff it is allowed by all $t \in T$.

A tile defines a subset of rows and columns, and the rows in this subset are permuted by the same permutation function in each column in the tile. In other words, the *relations between the attributes inside the tile are preserved* (such as correlations etc.). Notice that the identity permutation is always an allowed permutation. Now, the sampling problem can be formulated as follows.

Problem 1 (Sampling problem). Given a set of tiles T , draw samples uniformly at random from vectors of permutations in $\mathcal{P}^{m'}$ allowed by T .

The sampling problem is trivial when the tiles are non-overlapping. In the case of overlapping tiles, one can always merge tiles to obtain an equivalent set of non-overlapping tiles (i.e., a *tiling*) as shown in [20].

Example 2. Let us consider again the toy data Z_{toy} and define a tile constraint $t = (R, C)$ as follows. Let R be the set of points from class ‘1’ that are separated from the points in class ‘-1’ along the second attribute in X_{toy} , i.e., the larger of the two red clusters, and let $C = \{1, 2, 4\}$, i.e., the first two attributes of X_{toy} and the class attribute Y_{toy} . Now, if we permute Z_{toy} using a vector of permutations allowed by t , we obtain a sample data in which the relations inside the tile are preserved. An example of such a data sample is shown in Fig. 1c. This distributions models the case where the user is aware that the points in the tile are in class ‘1’ and that they form a cluster in attributes $X1$ vs. $X2$.

¹ We use the general term *relation* for any structure in data that can be controlled using the constrained permutation scheme, e.g., correlation or cluster structure.

Focusing Exploration Using Hypotheses. The tile constraints can also be used to specify the relations in which the user is interested [20]. The so-called *hypothesis tilings* define the items R and attributes C of interest, and the relations between the attributes that the user is interested in through a partition of C . To simplify the presentation here, we will make the assumption that the user is interested in *all* relations between all the attributes. This restricted setting reduces to *unguided data exploration*, where the user is interested in all unknown inter-attribute relations in the data. Notice that the HGDE framework allows the user to define more general hypotheses in a flexible way (see [20] for details) and our current approach is compatible with the more general hypothesis as well.

The intuition is that we model two distributions over data sets: (i) the one which models what the user can learn of the interesting relations in the data (formalised by HYPOTHESIS 1), and (ii) the other which models what the user already knows of the interesting relations in the data (formalised by HYPOTHESIS 2). The dimensionality reduction problem is then to find a direction $v \in \mathbb{R}^m$ in which the two distributions differ the most, using a suitable objective function. In [20], e.g., the objective in DR was essentially to find the direction maximising variance, which will by definition give a user a view (projection) that is the most informative. More formally, let us thus consider the following hypotheses:

- HYPOTHESIS 1: there are relations in data between all the attributes, and
- HYPOTHESIS 2: there are no relations in data between any of the attributes.

Now, a distribution p_1 conforming to HYPOTHESIS 1 can be characterised using the tile $t_1 = ([n], [m'])$, which restricts the set of allowed vectors of permutations so that every column (attribute) has to be permuted using the same permutation. On the other hand, a distribution p_2 conforming to HYPOTHESIS 2 can be characterised using the set of tiles $\{([n], \{j\}) \mid j \in [m']\}$, which places no restrictions on the set of allowed vectors of permutations, i.e., every column (attribute) is permuted independently.

The knowledge of the user concerning relations in the data is described by tiles defined by the user during exploration process (user tiles), which are merged into the both of the hypothesis tilings. The process is iterative in the sense that after the user adds more constraints, a new direction v is sought. While the permutation-based randomisation scheme is general to all data types, the projection pursuit in [20] is restricted to real-valued data, and reduces to *principal component analysis* (PCA) when the user has initially no background knowledge and the hypotheses cover all the data.

Example 3. In Fig. 2a the projection of the real-valued part X_{toy} to the first two principal components is shown, which corresponds to the most informative projection in the HGDE framework when the user has no background knowledge and the hypotheses cover all the data. While this projection provides the view to data maximising variance, it is not very useful in case if the user was interested in, e.g., the class ‘1’.

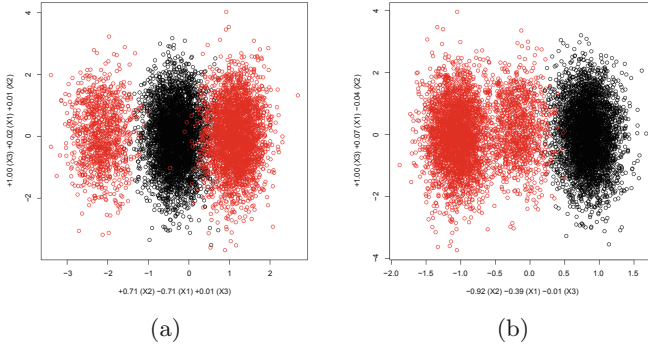


Fig. 2. Toy data Z_{toy} projected into first two principal components using PCA (a) and SPCA (b). Colors as in Fig. 1. (Color figure online)

3 Supervised Exploration

Example 3 shows that the most informative projection in the HGDE framework does not take into account the class information, which is by no means surprising, since only the real-valued part of the data was used. We now extend the HGDE framework to a supervised setting, i.e., instead of looking for directions in which the distributions corresponding to hypotheses differ the most in general, we are interested in finding directions which give most information about a class.

Example 4. Let us assume that a user is interested in class ‘1’ in our toy data Z_{toy} . One alternative could be to use supervised PCA (SPCA) [1]. In Fig. 2b we provide a projection obtained performing SPCA on X_{toy} with delta-kernel for Y_{toy} . Clearly, the x -axis separates the data with respect to Y_{toy} . However, if we assume that the user already has some background knowledge about the data, e.g., the user knows the relations formulated in terms of tile t from Example 2, this projection becomes less informative and there is no direct way to incorporate the user’s knowledge into SPCA.

As a further observation, we note that when there is only a single target attribute (as it is the case with our present work), the resulting optimisation problem in SPCA involves a rank-1 matrix, and thus only the first component contains meaningful information.

We formulate now our *main problem*, i.e., how to find the direction $v \in \mathbb{R}^m$ that is the most informative with respect to a particular class $c \in L$. We will use two hypotheses, HYPOTHESIS 1 and HYPOTHESIS 2, formulated as described in Sect. 2. Furthermore, we assume that the tile constraints used to represent the background knowledge of a user are merged into both hypotheses, and when we refer to HYPOTHESIS 1 and HYPOTHESIS 2, we always assume that the current user tiles are merged into both.

Problem 2 (Main problem). Given distribution p_1 conforming to HYPOTHESIS 1 and p_2 conforming to HYPOTHESIS 2 together with a class $c \in L$, find the direction $v \in \mathbb{R}^m$ providing the most information about the class c , i.e., the direction v in which p_1 and p_2 differ the most in terms of c .

Let $X_{Y=c}$ denote the restriction of the real-valued part X of $Z = (X|Y)$ to those rows i for which $Y(i) = c$. Our problem can then be formalised as finding a direction v in which $X_{Y=c}$ and $X'_{Y'=c}$ differ most by some suitable measure, where $Z = (X|Y)$ and $Z' = (X'|Y')$ have been sampled from p_1 and p_2 , respectively. Thus, to solve Problem 2, we need a function that measures how well the class c is separated in p_1 and p_2 in a direction v .

We want to choose a measure that will separate the distributions as much as possible *visually*. To illustrate what we mean by this, consider, e.g., a case where distributions p_i^v , $i \in \{1, 2\}$, are defined by a uniform distribution plus a narrow peak² at $x_i(v) \in [-1, 1]$ to direction v . We would want to find a measure that is largest when the distance between the peaks $|x_1(v) - x_2(v)|$ is maximised. From information-theoretic view an obvious alternative would be Kullback-Leibler divergence between distributions p_i^v , but, in fact, it is insensitive to the distance between peaks. Thus, we choose to use the numerically more stable *L1-norm between cumulative distributions*. For example, in the case of p_1^v and p_2^v this measure is maximised for v for which the distance between the peaks is the largest.

Definition 3. Given distributions p_1 and p_2 and a class of interest $c \in L$, the difference between p_1 and p_2 with respect to c in direction $v \in \mathbb{R}^m$ is computed using the L1-distance between the empirical cumulative distribution functions for the real-valued parts of samples $Z = (X|Y)$ and $Z' = (X'|Y')$ from p_1 and p_2 , respectively, restricted to c and projected to v :

$$f(Z, Z', c, v) = \|F(X_{Y=c}v) - F(X'_{Y'=c}v)\|_1, \quad (1)$$

where $F(x) : \mathbb{R}^n \mapsto [0, 1]$ is the empirical cumulative distribution function for the set of values in vector x .

Now, given a sample Z from the distribution p_1 conforming to HYPOTHESIS 1 and a sample Z' from the distribution p_2 conforming to HYPOTHESIS 2, we obtain the solution to Problem 2 by finding the direction v maximising $f(Z, Z', c, v)$:

$$v^* = \arg \max_{v \in \mathbb{R}^m} f(Z, Z', c, v). \quad (2)$$

In visualisations where we use two-dimensional scatterplots, we find the second dimension of the scatterplot by optimising the same objective while requiring the direction to be orthogonal to the first dimension. We will solve the optimisation problem above in practice using the standard quasi-Newton solver in R with random initialisation and default settings (i.e., the general-purpose `optim`

² More formally defined by $p_i^v(t) = U_{1+\sigma}(t)/2 + U_\sigma(t - x_i(v))/2$, where $U_a(t) = 1/(2a)$ if $-a \leq t \leq a$ and $U_a(t) = 0$ otherwise, at the limit of small σ or $\sigma \rightarrow 0^+$.

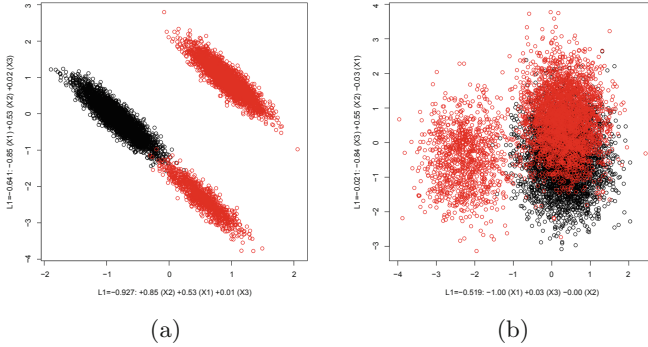


Fig. 3. The most informative projection about class ‘1’ for the toy data Z_{toy} without background knowledge (a) and using the tile t constraint from Example 2 as background knowledge (b). Colors as in Fig. 1. (Color figure online)

function in R with `method="BFGS"`). This approach proved to be sufficiently efficient for the data sizes typical for visual exploratory data analysis (in the order of thousands data points), as demonstrated in the experimental evaluation.

Example 5. We now apply Definition 3 to find the most informative view to the user with respect to class ‘1’. Assuming no initial background knowledge, the datasets shown in Fig. 1a, b are examples of data samples from the distributions p_1 and p_2 , respectively. By solving Eq. (2) we obtain the projection in Fig. 3a. The difference between the distributions is maximised along the x -axis, and we observe that the class ‘1’ consists of two group of points. We can now add this observation to the background knowledge³, e.g., by using the tile t from Example 2. Because the tile is added to *both* HYPOTHESIS 1 and HYPOTHESIS 2, the information we have learned is reflected in both distributions, and any samples conforming to the updated hypotheses will not differ in terms of the relations constrained by t . The most informative projection for Z_{toy} with the background knowledge (tile t) is shown in Fig. 3b. This projection is different to Fig. 3a, and we see that the most informative direction (x -axis) separates the data items in class ‘1’ for which we did not yet add background knowledge from the rest of the data.

4 Experimental Evaluation

In this section we first consider the scalability (in terms of the dimensions of the data) and stability (in case the class contains only a few samples) of the method presented in this paper. After this, we present use cases of exploration of relations in data relevant for a class. The experiments were performed with a

³ In an interactive setting, the selection of data items would be easy from the scatter-plot. For the selection of attributes, one can use, e.g., the method from [20, Sec. 2.4].

single-threaded R 3.5.0 implementation on a MacBook Pro laptop with a 3.1 GHz Intel Core i5 processor.⁴

Datasets. In the experiments, we utilise the following datasets. We scale the real-valued variables to zero mean and unit variance.

The GERMAN socio-economic dataset [3, 12] contains records from 412 administrative districts in Germany. Each district is represented by 46 attributes describing socio-economic and political aspects in addition to the *type* of the district (rural/urban), area name/code, state, *region* (East, West, North, South) and the geographic coordinates of each district centre. The socio-ecologic attributes include, e.g., population density, age and education structure, economic indicators, and the proportion of the workforce in different sectors. The political attributes include election results of the five major political parties (CDU/CSU, SPD, FDP, Green, and Left) in the German federal elections in 2005 and 2009, as well as the voter turnout. We exclude the election results from 2005, the area code and coordinates of the districts, and all non-numeric variables except those for *region* and *type*. This results in 32 real-valued attributes and two class variables (*region* and *type*) used in our experiments.

The British National Corpus (BNC) [2] is one of the largest annotated text corpora freely available in full-text format. The texts are annotated with information such as author gender, age, and target audience, and all texts have been classified into *genres* [14]. We use a preprocessed data from [21] in which the vector-space model (word counts) is computed using the first 2000 words from each text belonging to one the four main *genres* in the corpus (‘prose fiction’, ‘transcribed conversations’, ‘broadsheet newspaper’, ‘academic prose’) as done in [17]. The BNC dataset has word counts for 1335 texts and the attributes are

Table 1. Median wall clock running time for the synthetic data with varying number of rows (n) and columns (m). We give the time to generate the hypothesis tilings, add three random tiles, and generate the data samples conforming to the hypotheses (t_{model}) and the time to find the most informative view (t_{view}), i.e., to solve Eq. (2).

n	m	t_{model} (s)	t_{view} (s)	n	m	t_{model} (s)	t_{view} (s)
500	16	0.01	0.97	2000	16	0.03	2.03
	32	0.01	2.26		32	0.05	7.57
	64	0.02	8.15		64	0.07	32.38
	128	0.03	66.15		128	0.12	114.76
1000	16	0.02	1.23	4000	16	0.09	4.54
	32	0.02	3.97		32	0.11	12.78
	64	0.04	18.91		64	0.16	45.05
	128	0.06	92.83		128	0.26	140.35

⁴ Code and data available at <https://github.com/edahelsinki/shgde>.

the 100 words with highest counts. The class attribute contains classification of each text into one of the 4 main *genres*.

The Kaggle Telco customer CHURN dataset⁵ contains information of 7043 customers with 21 attributes (18 categorical and 3 real-valued) including information about services of the customer, customer account, and demographic information. The task is to predict the value of binary class attribute ‘churn’ (whether the customer has left within the last month). We transform all the categorical attributes (except ‘churn’) using one-hot encoding, which creates a column for every label of every attribute and the presence (or absence) of a label is indicated by 1 (or 0). Note that variables with many labels are implicitly given more weight in the one-hot encoding. To overcome this effect, we scale the binary data in groups, that is, all columns that originate from the same attribute are scaled to have a total variance of 1. Finally, we drop 11 rows containing ‘NA’ for attribute ‘total charges’, and end up with 7032 rows and 46 columns.

4.1 Scalability

We started by evaluating the scalability of our method on synthetic data with $m \in \{16, 32, 64, 128\}$ dimensions and $n \in \{500, 1000, 2000, 5000\}$ data points. We generated the datasets similarly to [18]. The data points are scattered around 10 randomly drawn cluster centroids. We used the clusters to form a binary class attribute (by assigning the cluster centres closest to each other into same class). We added $k = 3$ random tiles as background knowledge: for each tile the rows were selected by taking the data points from one of the 10 clusters, and for the columns we randomly selected $[2..m]$ columns.

We report in Table 1 the median wall clock running times. We can observe that the time t_{model} to generate the hypothesis tilings, add three random tiles, and generate the data samples conforming to the hypotheses is negligible, i.e., we can update our hypotheses and obtain new samples very fast. The time t_{view} to find the most informative direction, i.e., to solve Eq. (2) scales roughly as $O(nm^{2..3})$. Even with our unoptimised R implementation the running times

Table 2. Stability experiment. In columns $\text{avg}(f)$, $\text{sd}(f)$, and $\text{sd}(f)/\text{avg}(f)$ we report the average of each of these over the six different classes used.

c_{\min}	k	$\text{avg}(f)$	$\text{sd}(f)$	$\text{sd}(f)/\text{avg}(f)$
100	0	2.03	0.070	0.042
	3	1.79	0.068	0.045
500	0	2.01	0.036	0.028
	3	1.80	0.034	0.028
1000	0	2.00	0.023	0.022
	3	1.78	0.026	0.023

⁵ Available at <https://www.kaggle.com/blastchar/telco-customer-churn>.

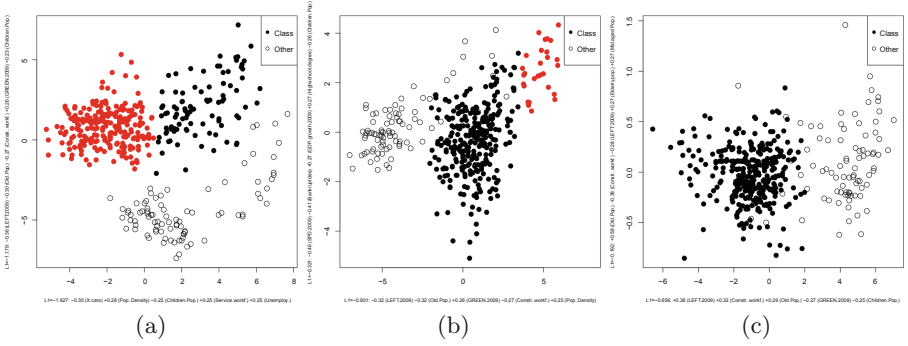


Fig. 4. Supervised exploration of the GERMAN data w.r.t. a class consisting of the districts in regions ‘West’, ‘South’, and ‘North’. (a) The most informative projection with no background knowledge. (b) The most informative projection with tile t_1^g as background knowledge. (c) The most informative projection with tiles t_1^g and t_2^g as background knowledge. See Sect. 4.3 for details of selections shown in red. (Color figure online)

are at the order of 10s for reasonably sized datasets. We note that for visual exploration the size of the data n should be reasonable and, it should be down-sampled as needed. Hence, the time complexity will be asymptotically constant with respect to n . The time complexity with respect the dimensionality m could be controlled by first reducing the dimensionality of the data, e.g., by PCA or by random projections, or by relaxing the convergence criteria of the numerical optimisation.

4.2 Stability

When the class of interest has only a few items, the effect of a particular sample from the distribution conforming to HYPOTHESIS 2 to the direction that is optimal for Eq. (2) is potentially large. This potential instability caused by the sampling can be controlled by taking several samples from the distributions and concatenating them, thus making the sample used to solve Eq. (2) large enough. To study this effect, we used the GERMAN dataset, taking the districts from each *region* and of each *type* as classes (6 cases in total, the class sizes varying between 64 and 290) and added $k \in \{0, 3\}$ random clusters as the background knowledge. Then, we computed mean value and the standard deviation of Eq. (1) in the optimal direction for 10 samples for each $c_{\min} \in \{100, 500, 1000\}$. Here, the number of samples needed s was computed as $s = \lceil c_{\min} / |\{i \mid Y(i) = c\}| \rceil$. Looking at the ration of standard deviation and the mean in Table 2, we observe that setting $c_{\min} \geq 500$ suffices for practical purposes. For the remaining experiments we use this value.

4.3 Supervised Exploration of GERMAN Data

The separation in the socio-economic and political factors between districts in *region* ‘East’ and the districts in other regions is the most dominant factor in the GERMAN dataset, see e.g., [3, 12, 20]. We assume now that we are interested in exploring other factors in the data, in particular those representative for the non-Eastern regions. Thus, we choose a class consisting of districts in *regions* ‘West’, ‘South’, and ‘North’ for our first use case.

Figure 4a shows the most informative view with respect to our class (solid circles are used for districts in the class, circles without a fill are used for districts not in the class) without any background knowledge. The projection shown separates the districts in the class into two parts along x -axis. We define a tile t_1^g to add this observation into the background knowledge. We select the districts coloured red in Fig. 4a for the rows, and all attributes for the columns.⁶ Looking at the distribution of *region* (North = 46, South = 108, West = 78) and *type* (Urban = 7, Rural = 225) attributes for this selection we observe that we have defined a tile constraint for a set of mainly rural districts.

Figure 4b shows the most informative view the class given t_1^g as background knowledge. We obtain a different projection and observe the districts coloured red in Fig. 4b have higher values along x -axis than the rest of the districts. From the distribution of *region* (North = 4, South = 15, West = 11) and *type* (Urban = 25, Rural = 5) attributes for this selection we observe that these are mainly urban districts from the class. We add this observation into the background knowledge by defining a tile t_2^g . The rows in t_2^g are those coloured red in Fig. 4b, and for columns we include all attributes. Figure 4c then shows the most informative view with respect to the class given both t_1^g and t_2^g as background knowledge, demonstrating the division between the Eastern districts and the rest.

To understand the utility of the views shown, we compute values of the measure f in Eq. (1) using samples from the distributions conforming

Table 3. The GERMAN data use case. The value of f from Eq. (1) for different projection vectors v and cases of background knowledge.

GERMAN	No background	Tile t_1^g	Tiles t_1^g, t_2^g
v_0	1.627	0.148	0.073
v_1	1.079	0.901	0.641
v_2	1.115	0.880	0.656
v_{spca}	1.306	0.739	0.555
v_{pca}	1.336	0.417	0.322

⁶ For simplicity, we use the set of all attributes as the columns in the tiles in explorations of the GERMAN and BNC datasets. In [20, Sec. 2.4] we provide a principled way for selecting a subset of columns most relevant for a selection of rows, which could be used in a more subtle exploration.

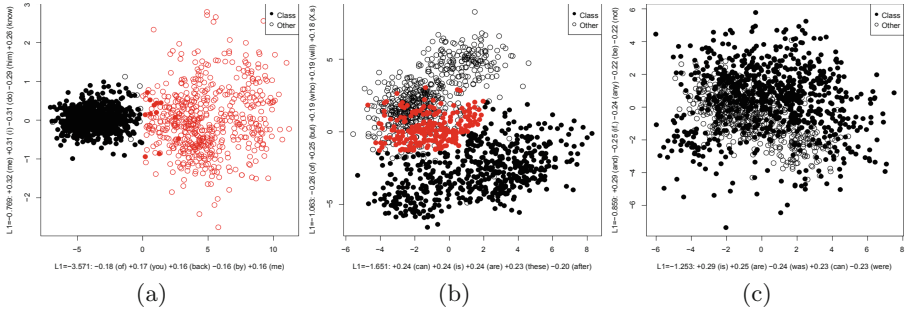


Fig. 5. Supervised exploration of the BNC data w.r.t. a class consisting of the texts from the *genres* ‘broadsheet newspaper’ and ‘academic prose’. (a) The most informative projection with no background knowledge. (b) The most informative projection when tile t_1^{bnc} is added to the background knowledge. (c) The most informative projection when tiles t_1^{bnc} and t_2^{bnc} are added to the background knowledge. See Sect. 4.4 for details of selections coloured red. (Color figure online)

to HYPOTHESIS 1 and HYPOTHESIS 2 given the background knowledge. We have three cases: no background knowledge (0 tiles), background knowledge represented using tile t_1^g (1 tile), and background knowledge represented using tiles t_1^g and t_2^g (2 tiles). For each case we compute the direction in optimising the measure f , i.e., a solution to Eq. (2), denoting these by v_i where i corresponds to the number of tiles in the background knowledge. For comparison, we also compute the first PCA and SPCA projection vectors, denoted by v_{pca} and v_{spca} , respectively. Then, we calculate the value for f in different cases. The results are presented in Table 3. We notice that the value of the measure f indeed is always the highest, when the projection vector matches the background knowledge (highlighted in the table), as expected. This shows that the views presented are indeed the most informative ones given the current background knowledge. We also notice that PCA and SPCA projection vectors are less informative in terms of the measure f .

Table 4. The BNC data use case. The value of f from Eq. (1) for different projection vectors v and cases of background knowledge.

BNC	No background	tile t_1^{bnc}	tiles t_1^{bnc}, t_2^{bnc}
v_0	3.571	0.589	0.247
v_1	1.708	1.651	1.103
v_2	1.513	1.480	1.253
v_{spca}	3.561	0.572	0.241
v_{pca}	3.488	0.520	0.206

4.4 Supervised Exploration of BNC Data

As our second use case we consider the BNC dataset by exploring the high-level structure of the corpus. The exploration of the same data in [21] already reveals us that the *genres* ‘prose fiction’ and ‘transcribed conversations’ form rather clearly visible clusters in the PCA projection of the data, while the *genres* ‘broadsheet newspaper’ and ‘academic prose’ are not very distinct from each other. Thus, we focus our interest to a class containing texts from the *genres* ‘broadsheet newspaper’ and ‘academic prose’ to see whether our supervised method allows us to find projections which would provide us new information about these genres.

Figure 5a shows the most informative view with respect to the class (solid circles are used for texts belonging to the class, circles without a fill are used for the texts not in the class). The projection shown clearly separates the texts with respect to our class. We define a tile constraint t_1^{bnc} , by selecting the points with x -axis value greater than zero (coloured red in Fig. 5a) for the rows, and all attributes for the columns. The selection contains 144 texts from *genre* ‘transcribed conversations’, 413 from ‘prose fiction’, and 12 texts from *genre* ‘broadsheet newspaper’. Thus, we add a tile constraint covering mostly texts outside the class, making this way explicit to the system that we already know the main features of the texts not in our class. Figure 5b shows the most informative view after t_1^{bnc} has been added to the background knowledge. We observe that the texts in the class seem to separate in the direction along y -axis. By selecting the points with higher values in y -axis (coloured red in Fig. 5b) in our class, we observe that these are mainly texts from *genre* ‘broadsheet newspaper’ (211 texts), the remaining 10 texts are from *genre* ‘academic prose’. Thus, this view shows us how the two genres in our class are separated. If we now add a tile constraint t_2^{bnc} for this selection (taking again all attributes as the columns), we obtain the view shown in Fig. 5c, in which some outliers could be potentially studied further.

Similarly to the GERMAN data use case, we provide the value of the measure f for each projection vector in Table 4, and compare these to the first PCA and SPCA projection vectors. Here we observe, that both PCA and SPCA provide a direction with a very similar interestingness value to our method when there is no background knowledge. However, with background knowledge, the situation changes and our approach provides clearly more interesting views given the class.

4.5 Identification of Churners

Finally, we explore the CHURN data. The problem of identifying possible *churners*, i.e., customers likely to cancel a subscription to a service, has become a popular use case in business domain, because retaining one customer costs much less than gaining a new one. Churn prediction problem is typically addressed with off-the-shelf machine learning and statistical approaches which usually do not use any domain expert knowledge. In this example, our goal is to demonstrate how our method can help to put the domain-specific knowledge into better use.

We can use our framework to find the most informative direction with respect to the class containing customers who churn.

Now, let us assume that the domain experts have already identified from their previous experiences that ‘monthly charge’ and ‘total charges’ are the most salient features that cause customer to churn. We will use this background knowledge in the exploration, i.e., we add a tile t_{chu} covering attributes ‘monthly charge’, ‘total charges’, and ‘churn’ and all the rows in the data to the background distribution. The most informative direction in this case has the highest (absolute) weights for the attributes ‘tech support = no’, ‘online security = no’, and ‘internet service = fiber optic’.

We can compare this set of five features (i.e., ‘total charges’, ‘monthly charges’, ‘tech support’, and ‘online security’ and ‘internet service’) identified by the user to the whole set of features in the data, when classifying churners using the *non-preprocessed dataset*. Here we use fitted binary classification decision tree with 10-fold cross validation for the classification, and measure the performance with misclassification error (ME) and false positives (FP) rate. We observe, that using the user identified 5-feature set (ME = 0.263, FP = 0.127) the performance that is at least as good as using the full 20-feature set (ME = 0.264, 0.133), and even marginally better in terms of false positives rate. This demonstrates the potential human-guided exploration approach for a real-world dataset, in particular in a scenario in which a high false positive rate is a major concern.

5 Conclusions

In this paper we proposed a method for *supervised dimensionality reduction* for interactive EDA systems that take the user’s background knowledge and objectives into account. We defined an information criterion, which allows us to find the most informative views about the class structure of data by taking the user’s current knowledge and objectives into account. In the experimental evaluation we demonstrated that our method gives understandable and useful results when analysing real-world datasets. Taking the user’s background knowledge into account matters, as the use of the updating background knowledge allows an EDA system to show the user currently unknown and relevant projection to the data.

For potential future directions we note that our method could potentially be used for *human-guided classification* by using an updating class of interest, instead a fixed one. Initially, all items would belong to the class of interest, and the user is shown the most informative projection. The user could then identify set(s) of data items and classify them, and a new projection could be shown for an updated class of interest containing the data items unclassified so far. Moreover, the knowledge of the user of the found sets of data items could be added into the background knowledge. We also plan to implement our method in an interactive data analysis tool, and study how the optimisation problem in Eq. (2) can be solved more efficiently in practice. For a better interpretability of the views, we could consider, e.g., sparse projection vectors.

Acknowledgements. We thank Buse Gül Atli for discussions and help. Supported by the Academy of Finland (decisions 326280 and 326339). This work is part of the research programme Commit2Data, specifically the RATE-Analytics project NWO628 003 001 (partly) financed by the Dutch Research Council.

References

1. Barshan, E., Ghodsi, A., Azimifar, Z., Jahromi, M.Z.: Supervised principal component analysis: visualization, classification and regression on subspaces and submanifolds. *Pattern Recogn.* **44**(7), 1357–1371 (2011)
2. The British National Corpus, v. 3 (BNC XML Edition). Distributed by Oxford University Computing Services on Behalf of the BNC Consortium (2007). <http://www.natcorp.ox.ac.uk/>
3. Boley, M., Mampaey, M., Kang, B., Tokmakov, P., Wrobel, S.: One click mining: interactive local pattern discovery through implicit preference and performance learning. In: *KDD-IDEA*, pp. 27–35 (2013)
4. Chau, D., Kittur, A., Hong, J., Faloutsos, C.: Apolo: making sense of large network data by combining rich user interaction and machine learning. In: *CHI*, pp. 167–176 (2011)
5. De Bie, T., Lijffijt, J., Santos-Rodríguez, R., Kang, B.: Informative data projections: a framework and two examples. In: *ESANN*, pp. 635–640 (2016)
6. De Bie, T.: Maximum entropy models and subjective interestingness: an application to tiles in binary databases. *Data Min. Knowl. Discov.* **23**(3), 407–446 (2011)
7. Dimitriadou, K., Papaemmanouil, O., Diao, Y.: AIDE: an active learning-based approach for interactive data exploration. *IEEE Trans. Knowl. Data Eng.* **28**(11), 2842–2856 (2016)
8. Dzyuba, V., van Leeuwen, M.: Interactive discovery of interesting subgroup sets. In: Tucker, A., Höppner, F., Siebes, A., Swift, S. (eds.) *IDA 2013*. LNCS, vol. 8207, pp. 150–161. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-41398-8_14
9. Fisher, R.A.: The use of multiple measurements in taxonomic problems. *Ann. Eugen.* **7**, 179–188 (1936)
10. Globerson, A., Tishby, N.: Sufficient dimensionality reduction. *J. Mach. Learn. Res.* **3**, 1307–1331 (2003)
11. Hanhijärvi, S., Ojala, M., Vuokko, N., Puolamäki, K., Tatti, N., Mannila, H.: Tell me something I don't know: randomization strategies for iterative data mining. In: *KDD*, pp. 379–388 (2009)
12. Kang, B., Lijffijt, J., Santos-Rodríguez, R., De Bie, T.: Subjectively interesting component analysis: data projections that contrast with prior expectations. In: *KDD*, pp. 1615–1624 (2016)
13. Kang, B., Lijffijt, J., Santos-Rodríguez, R., De Bie, T.: SICA: subjectively interesting component analysis. *Data Min. Knowl. Disc.* **32**(4), 949–987 (2018). <https://doi.org/10.1007/s10618-018-0558-x>
14. Lee, D.W.: Genres, registers, text types, domain, and styles: clarifying the concepts and navigating a path through the BNC jungle. *Lang. Learn. Technol.* **5**(3), 37–72 (2001)
15. Lee, J.A., Verleysen, M.: *Nonlinear Dimensionality Reduction*. Springer, New York (2007). <https://doi.org/10.1007/978-0-387-39351-3>

16. van Leeuwen, M., Cardinaels, L.: VIPER – visual pattern explorer. In: Bifet, A., et al. (eds.) ECML PKDD 2015. LNCS (LNAI), vol. 9286, pp. 333–336. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-23461-8_42
17. Lijffijt, J., Nevalainen, T.: A simple model for recognizing core genres in the BNC. In: Studies in Variation, Contacts and Change in English, vol. 19 (2017)
18. Puolamäki, K., Kang, B., Lijffijt, J., De Bie, T.: Interactive visual data exploration with subjective feedback. In: Frasconi, P., Landwehr, N., Manco, G., Vreeken, J. (eds.) ECML PKDD 2016. LNCS (LNAI), vol. 9852, pp. 214–229. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46227-1_14
19. Puolamäki, K., Papapetrou, P., Lijffijt, J.: Visually controllable data mining methods. In: ICDMW, pp. 409–417 (2010)
20. Puolamäki, K., Oikarinen, E., Henelius, A.: Guided visual exploration of relations in data sets. arXiv preprint [arXiv:1905.02515](https://arxiv.org/abs/1905.02515) (2019)
21. Puolamäki, K., Oikarinen, E., Kang, B., Lijffijt, J., Bie, T.D.: Interactive visual data exploration with subjective feedback: an information-theoretic approach. In: ICDE, pp. 1208–1211 (2018)
22. Ruotsalo, T., Jacucci, G., Myllymäki, P., Kaski, S.: Interactive intent modeling: information discovery beyond search. CACM **58**(1), 86–92 (2015)
23. Sacha, D., et al.: Visual interaction with dimensionality reduction: a structured literature analysis. IEEE Trans. Visual Comput. Graphics **23**(1), 241–250 (2017)
24. Tukey, J.W.: Exploratory Data Analysis. Addison-Wesley, Reading (1977)
25. Vartak, M., Rahman, S., Madden, S., Parameswaran, A., Polyzotis, N.: SeeDB: efficient data-driven visualization recommendations to support visual analytics. PVLDB **8**(3), 2182–2193 (2015)
26. Xing, E.P., Jordan, M.I., Russell, S.J., Ng, A.Y.: Distance metric learning with application to clustering with side-information. In: NIPS, pp. 521–528 (2003)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter’s Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter’s Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

