Computers & Graphics 60 (2016) 107–118

Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

Special Section on SIBGRAPI 2016

iStar (i*): An interactive star coordinates approach for high-dimensional data exploration

Germain Garcia Zanabria [a], Luis Gustavo Nonato [b], Erick Gomez-Nieto [b,c,*]

[a] Universidad Católica San Pablo, Arequipa, Peru
[b] ICMC, University of Sao Paulo, Brazil
[c] IBM Research, Brazil

ARTICLE INFO

Article history:
Received 1 March 2016
Received in revised form
8 August 2016
Accepted 16 August 2016
Available online 13 September 2016

Keywords:
Visualization
Multidimensional Data
Star Coordinates

ABSTRACT

Star Coordinates is an important visualization method able to reveal patterns and groups from multi-dimensional data while still showing the impact of data attributes in the formation of such patterns and groups. Despite its usefulness, Star Coordinates bears limitations that impair its use in several scenarios. For instance, when the number of data dimensions is high, the resulting visualization becomes cluttered, hampering the joint analysis of attribute importance and group/pattern formation. In this paper, we propose a novel method that renders Star Coordinates a feasible alternative to analyze high-dimensional data. The proposed method relies on a clustering mechanism to group attributes in order to mitigate visual clutter. Clustering can be performed automatically as well as interactively, allowing the analysis of how particular groups of attributes impact on the radial layout, thus assisting users in the understanding of data. The effectiveness of our approach is shown through a set of experiments and case studies, which attest its usefulness in practical applications.

1. Introduction

Developing mechanisms able to reveal patterns, trends, and groups from high-dimensional data is one of the main goals of visual analytics. Understanding the impact of data attributes in pattern and group formation is another important task in this context. Among the multitude of techniques devoted to visualize and understand high-dimensional data [1], radial visualization is noteworthy, as this class of techniques is able to uncover patterns and groups while simultaneously depicting how data attributes influence pattern and group formation [2].

A particularly important instance of radial visualization is the so-called Star Coordinates (SC) [3,4], which builds layouts through dimensionality reduction. More specifically, SC layouts consist of circularly arranged vectors v_i with a common origin, each vector corresponding to a data attribute. Data instances are mapped to the layout as a linear combination of the vectors v_i. In order to improve user experience, SC methods usually enable interactive resources that allow users to modify the length and angle of the vectors v_i so as to find configurations where patterns and groups are more clearly pronounced [5,6].

Despite their usefulness for multivariate data visualization, SC methods bear limitations that impair their use in relevant scenarios. For instance, when the number of data dimensions/attributes is large, the resulting visualization becomes cluttered, hampering readability and the joint analysis of attribute importance and group/pattern formation. Surprisingly, few has been done to mitigate such deficiency of Star Coordinates methods.

In this paper, we propose iStar (interactive Star Coordinates), a Star Coordinates based visualization technique able to handle data with a large number of attributes. iStar relies on attribute clustering, which can be performed automatically as well as interactively through user intervention. Moreover, iStar enables visualization resources to assist users in the analysis of clustered dimensions (attribute axes) and their impact on the resulting layout. Non-clustered attribute axes can also be automatically and interactively arranged in the layout to better explore patterns and groups. The combination of automatic and interactive resources make our approach an useful alternative for data exploration, as most patterns are liable to be revealed.

In summary, the main contributions of this work are:

- iStar, a method that combines attributes clustering and visualization resources to enable the analysis of high-dimensional data via Star Coordinates;
- the combination of automatic and interactive resources to assist users during data exploration via Star Coordinates. The proposed

* Correspondence to: Rua Tutóia, 1157 - Vila Mariana, São Paulo, SP, Brazil.
E-mail addresses: germain.garcia@ucsp.edu.pe (G. Garcia Zanabria), gnonato@icmc.usp.br (L.G. Nonato), erimgn@br.ibm.com (E. Gomez-Nieto).

http://dx.doi.org/10.1016/j.cag.2016.08.007
0097-8493/

methodology reduces visual clutter while preserving gist information in the layout. Clustered attributes can be further explored through interactive tools, making Star Coordinates a feasible alternative for high-dimensional visual data analysis.

2. Related work

The literature about radial visualization is extensive and a comprehensive survey is out of the scope of this paper. More detailed discussions can be found in specialized surveys [7,2].

In this section, we focus on the two radial visualization methods that are more closely related to our approach, namely, Radviz [8] and Star Coordinates [3,4]. RadViz and Star Coordinates are indeed well known methods widely employed in the context of information visualization [9]. Radviz and its variants [10–12] rely on mass-spring force paradigm to map multidimensional data onto a two-dimensional visual space. Each attribute variable (coordinate) is associated to a dimensional anchor placed over a circle in the visual space. Data instances are attached to the each dimensional anchor through a spring whose strength is proportional to the value of the instance attribute that corresponds to the anchor. The equilibrium state of the spring system provides the position of the instances in the visualization layout.

Star Coordinates associates each attribute (dimension) to a vector (or axis) in the visual space, which are arranged circularly around a common center. The position of each data instance is given by a linear combination of the vectors representing attributes. The length and orientation of the attribute axes in the visual space directly impacts in the final layout, thus fostering the development of different strategies to optimally arrange those axes (see [13,14,5,6,15]). The effectiveness of each strategy depends on the underlying application, as for example data classification [16,17].

Radviz and Star Coordinates have interesting properties such as esthetic appeal, compact layout, and easy interpretability, which justify their popularity as visualization metaphors. However, they suffer from issues as to the dimensionality of the underlying data, that is, the resulting layouts become cluttered when the number of dimensions is large. For instance, consider the SC example illustrated in Fig. 1. In Fig. 1(a), which involves a small number of dimensions, the layout resulting from SC is clear and readable. When the number of dimensions increases, the layout becomes more cluttered (Fig. 1(b)), turning out unreadable when a few hundred dimensions have to be handled, as depicted in Fig. 1(c).

The method presented in this paper, iStar, combines attribute clustering and interactive mechanisms to make Star Coordinates a feasible visual analytics alternative for exploring data with a large number of dimensions. Moreover, iStar seeks a balance between automated and interactive mechanisms, being able to perform tasks such as attribute clustering and axes arrangement in an optimal manner, enabling interactive resources to tune SC layouts according to users expertise. To the best of our knowledge, this is the first time attribute clustering is employed to improve the scalability of SC methods. Most of the ideas incorporated into Star Coordinates can also be adapted to RadViz, although we keep our discussion focused only on Star Coordinates.

3. The iStar method

iStar comprises three main modules: *linear mapping*, *clustering*, and *reordering* of attribute axes, as illustrated in Fig. 2. Those modules can operate in an automated manner, but iStar also enables a set of interactive resources to support users during the information discovery process.

Moreover, interactive tools are also combined with visual widgets to improve users experience. In particular, we propose, *Node Explorer* and *Node Preview* (described in the following) as visualization widgets that can be triggered interactively to assist users in the inspection of clustered attributes. *Quality visualizer* is another widget that allows users to evaluate the quality of layout arrangements, helping them in the search for ideal configurations.

3.1. Linear mapping

Star Coordinates maps data instances to the visual space through linear combination of attribute axes. In mathematical terms, the position of each data instance P_i is given by:

$$\vec{P}_i = p_{i1}\vec{v}_1 + p_{i2}\vec{v}_2 + \ldots + p_{in}\vec{v}_n = \sum_{j=1}^{n} p_{ij}\vec{v}_j \quad (1)$$

where n is the data dimension, $\vec{v}_j$ is the j-th attribute axis, and p_{ij} is a scalar defining the contribution of the j-th axis to the position of P_i. Orientation and scale of each attribute axis impacts the position of P_i, as illustrated in Fig. 3. In other words, each instance P_i scales attribute axes independently. In practice, the scaling factor of $\vec{v}_j$ depends on the value of the j-th attribute in P_i.

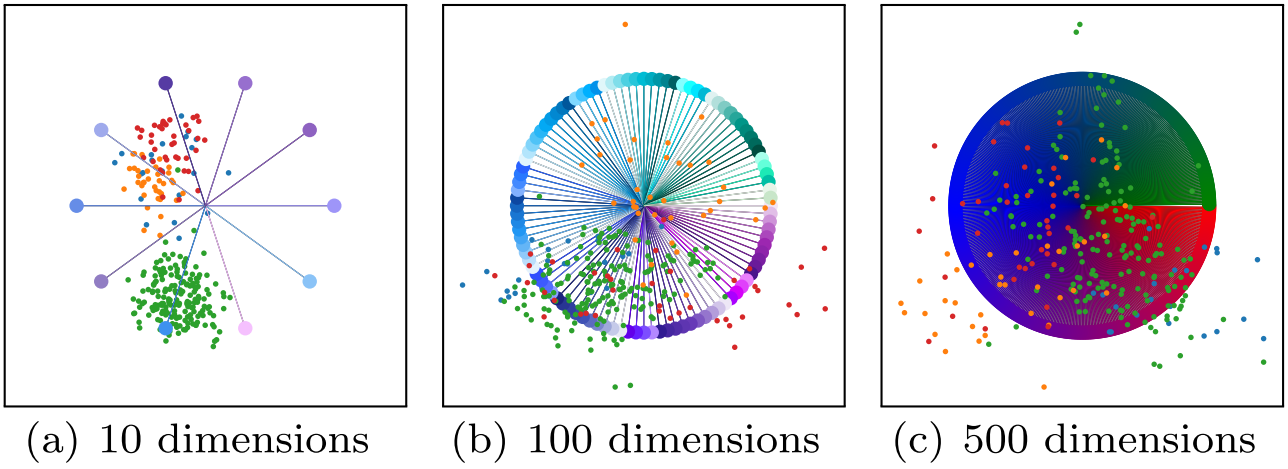


Fig. 1. Layout generated by Star Coordinates using a different number of dimensions. (a) 10 dimensions (b) 100 dimensions (c) 500 dimensions.

3.2. Attribute clustering

Several methods can be used to measure the similarity between attributes. We have implemented three different methods: Variance, Principal Component Analysis (PCA) [18], and Centroids.

Denoting the j-th attribute in P_i as p_{ij}, the variance (σ_j^2) is given by:

$$\sigma_j^2 = \frac{\sum_{i=1}^{m} (p_{ij} - \mu_j)^2}{m} \tag{2}$$

where m is the number of instances and μ_j is the average of the j-th attribute. The attributes j and k are considered similar if $|\sigma_j^2 - \sigma_k^2|$ is closer to zero. Given the variances, k-Means [19] can then be used to group similar attributes.

To measure the similarity between attributes using PCA, we consider each attribute as a point in a m-dimensional space (m is the number of data instances) and we use PCA to map those points onto the two-dimensional space generated by the first two principal components. Attributes mapped close to each other are considered similar. The groups of similar attributes are obtained by clustering points that have been projected onto the two-dimensional PCA space using k-Means algorithm.

The centroid mechanism relies on class information to identify similar attributes. Therefore, this method can only be employed when instances are provided with class information. The centroid $\overline{p}_{C_i}$ for a class C_i is given by:

$$\overline{p}_{C_i} = \frac{1}{N_{C_i}} \sum_{p \in C_i} p \tag{3}$$

where N_{C_i} is the number of instances in the class C_i. Considering each centroid as a representative instance of each class, we build a matrix M with centroids $\overline{p}_{C_i}$ ($\forall C_i$) as column vectors. Then, k-Means is applied to the rows of M to group similar centroids attributes.

Each group of attributes computed by one of the methods described above gives rise to an attribute axis in the iStar layout. The length of a grouped axis is set to one. The contribution of each grouped axis to the position of an instance P_i is given by averaging the values p_{ji} for all attribute j in the corresponding group.

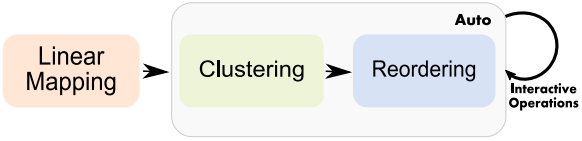


Fig. 2. The i* pipeline comprises 3 main steps: (i) A linear mapping of data, (ii) Attribute Clustering and (iii) Axes Reordering.

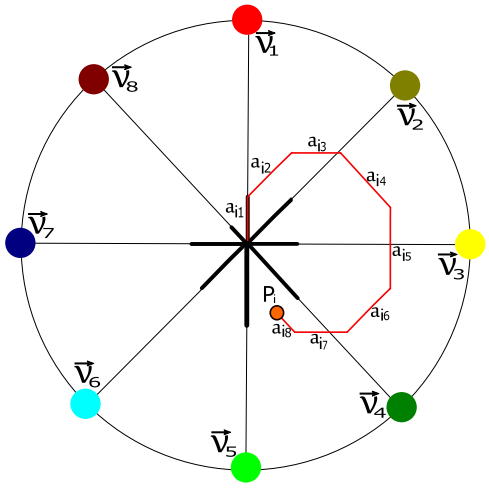


Fig. 3. Star coordinates representation of a data instance P_i.

3.3. Reordering

Arranging attribute axes properly is of paramount importance to uncover patterns in SC layouts and the order of the axes plays an important role in this context. Our prototype system provides two automated mechanisms to arrange attribute axes, one based on a combinatorial optimization, as proposed in [20], and another based on a brute force scheme.

Fig. 4 illustrates how the combinatorial optimization scheme operates to reorder attribute axes. Initially, we build the $k \times k$ dissimilarity matrix M, where k is the number of attribute axes (some axes can correspond to clusters of attributes), as follows:

$$M_{ij} = \frac{1}{m} \sum_{s=1}^{m} \left| \frac{p_{si} - \min_i}{\max_i - \min_i} - \frac{p_{sj} - \min_j}{\max_j - \min_j} \right| \tag{4}$$

where m is the number of instances, p_{si} (p_{sj}) is the i-th (j-th) attribute of an instance P_s, and min_i (min_j) and max_i (max_j) are the minimum and maximum values of the i-th (j-th) attribute, respectively. Notice that $M_{ij} \in [0, 1]$ and the closer to zero the more similar the i-th and j-th attributes are. Any alternative similarity measure based on correlation might be employed to fill the elements in matrix M such as Pearson correlation or Kendall's τ (see [21] for a comprehensive review focus on parallel coordinates).

The dissimilarity matrix is then represented as a complete graph where each node corresponds to an attribute axis. Edge weights are given by the entries M_{ij} (Fig. 4 middle). As proposed in

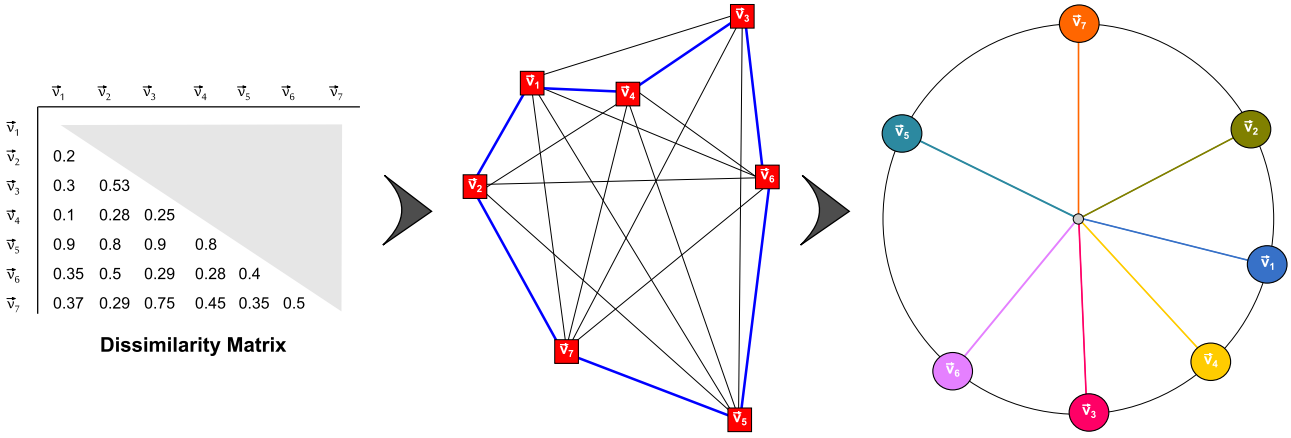


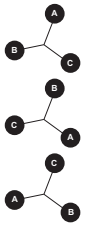
Fig. 4. Illustration of our combinatorial optimization scheme for attributes reordering.

[20], we use a Genetic Algorithm [22] to find an optimal close path connecting all nodes. The attribute axes are arranged in the same order as given by the optimal path.

The procedure above provides the order in which axes should be placed, but not the angle between axes. We adopted a simple scheme to set the angle (α_{ij}) between axes $\vec{v}_i$ and $\vec{v}_j$. Let W be the sum of the weights of the optimal path found by the reordering procedure (the sum of the blue edges weights in Fig. 4). Then, the angle is given by:

$$\alpha_{ij} = \frac{2\pi M_{ij}}{W} \tag{5}$$

The brute force scheme distributes the attributes axes uniformly in the unity circle and then "swap" their positions to find the best possible configuration, as illustrated in the inline figure on the right. Layout evaluation is performed based on layout quality metrics. As detailed in Section 4.1, we use *topology preservation* and *Dunn index* as quality metrics.



Both methods are illustrated in Fig. 5 using a dataset of 500 instances clustered in 5 groups. One can see that the method based on combinatorial optimization drastically changes the initial configuration while the brute force scheme only switches some axes.

Finding the optimal spacing between attribute axes is an important feature that our method also addresses. Fig. 6 illustrates three different approaches for accomplish this task, namely Radviz presents axes uniformly distributed resulting in a very dense mapping of data, DGBC is an intermediate step of DIFGBC [13] method and iStar based on variance which make use of similarity between attributes to set up an optimal spacing for their associated axes. As a result of the last two, clusters were better defined turning them easy to recognize by simple inspection, being that iStar managed to separate the red and orange clusters even more efficiently. The main advantage provided by iStar will be noticed according as the number of attributes increases, as evidenced in the following sections.

3.4. Interactive tools and visual resources

Our prototype provides a set of interactive tools and visual resources to assist users in the visual data exploration process.

The main interactive functionalities are:

- **Scaling** allows users to change the length of attribute axes, thus increasing or decreasing their contribution in the positioning of instances.
- **Rotation** operation modifies the direction of attribute axes in order to make particular axis more (or less) correlated with others.
- **Union** allows users to group attribute axes, combining their contribution during instance placement.
- **Separation** splits clusters of attributes axes so as to incorporate their contribution to the layout.
- **Removal** allows users to remove axes from the layout, disregarding their influence in the layout.
- **Re-insertion** allows users to re-insert removed dimensions.
- **Position Tuning** triggers the automated mechanism to arrange attribute axes in the visual space. This functionality takes into account only active axes or clusters of axes.

Cluster and union operations group a set of attribute axes into a single axis. Once a set of axes has been clustered into a node, the size of the corresponding node is increased in order to visually convey the grouping information. Analogously, after the separation, the node size is decreased. Understanding the influence and impact of grouped attributes in the quality of the layout is important to assist users during their exploration. Therefore, we provide the following visual widgets to support the analysis of clustered attribute axes:

- **Node Preview** magnifies a clustered axis as a "local" Star Coordinates visualization involving only the clustered attributes (Fig. 7(b)).
- **Node Explorer** is triggered together with Node Preview as a new panel where clusterized axes can be handled independently (Fig. 7(c)). User interactions performed in the Node Explorer panel such as scale and rotation, are reflected in the main visualization.
- **Quality Visualizer** panel provides information about the quality of the layout during user interaction (Fig. 7(d)). The layout quality evolution over time is depicted using a stacked graph metaphor [23]. The quality metrics are described in the next section.

4. Results, comparisons and evaluation

The performance of iStar (in its three variations) is assessed through a set of comparisons against known radial visualizations.

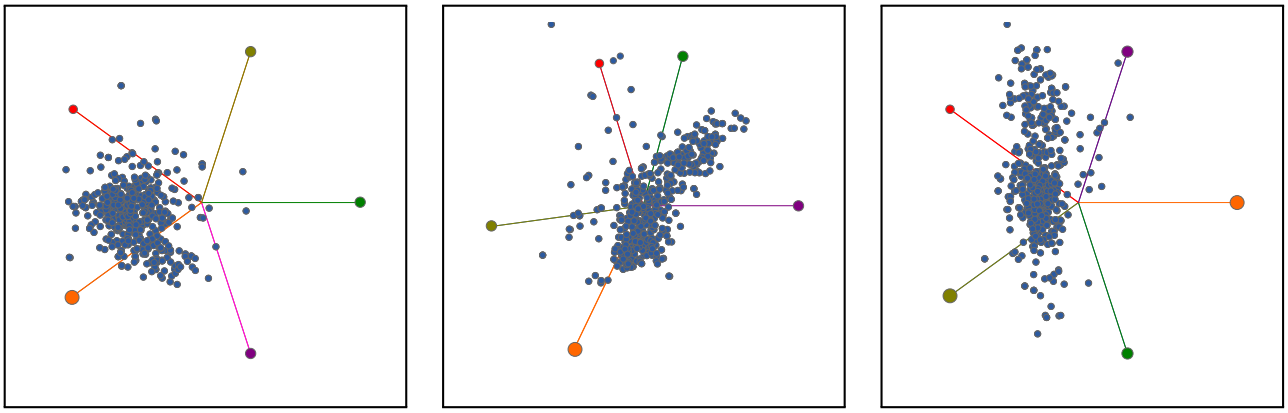


Fig. 5. Reordering a dataset of 500 instances and 11 attributes using the two proposed methods. (a) initial arrangement clustered in 5 groups (axes), reordering by (b) optimization and (c) brute force mechanisms.

More precisely, we employ two distinct metrics to quantitatively measure the quality of the produced layout. In the following, we describe the metrics used in our comparisons as well as the datasets used in our experiments.

4.1. Metrics and datasets

We rely on two different metrics to assess the quality of iStar layouts when compared against conventional RadViz, DIFGBC, and Star Coordinates methods.

Topology Preservation [24] compares rank order of neighbors in the original and visual space. Denoting by $NN_{ji}(i \in [1,k], j \in [1,m])$ and $nn_{ji}(i \in [1,k], j \in [1,m])$ the k nearest neighbors of instance j in the original and visual space respectively, the rank order of each instance j is assessed by the credit assignment:

$$r_j = \begin{cases} 3, & if\ NN_{ji} = nn_{ji} \\ 2, & if\ NN_{ji} = nn_{jl},\ l \in [1,k],\ i \neq l \\ 1, & if\ NN_{ji} = nn_{jt},\ t \in [k,s],\ k < s \\ 0, & otherwise \end{cases} \tag{6}$$

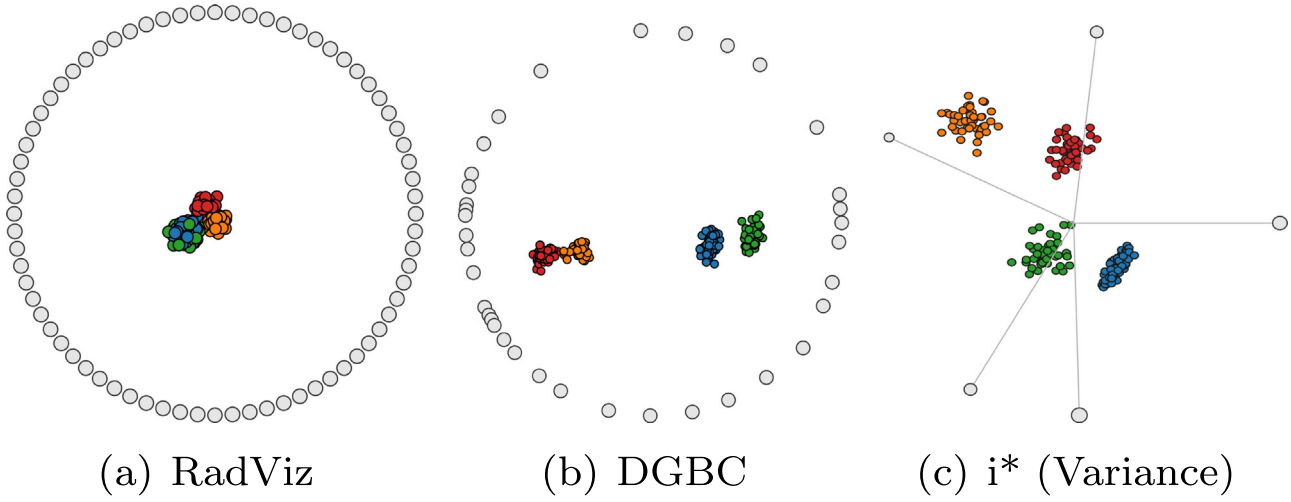


Fig. 6. Three different approaches for attribute axes spacing on Mammals dataset (DT1 in Table 1): (a) Radviz shows a uniform spacing between axes, (b) DGBC improves this distribution based on similarity between attribute axes (c) iStar based on variance without user interaction uses a similar axes spacing function, however, the attribute clustering step was previously performed. It can be noted that iStar shows a better data mapping in terms of cluster separation since it makes use of a lower and more representative number of attribute axes.

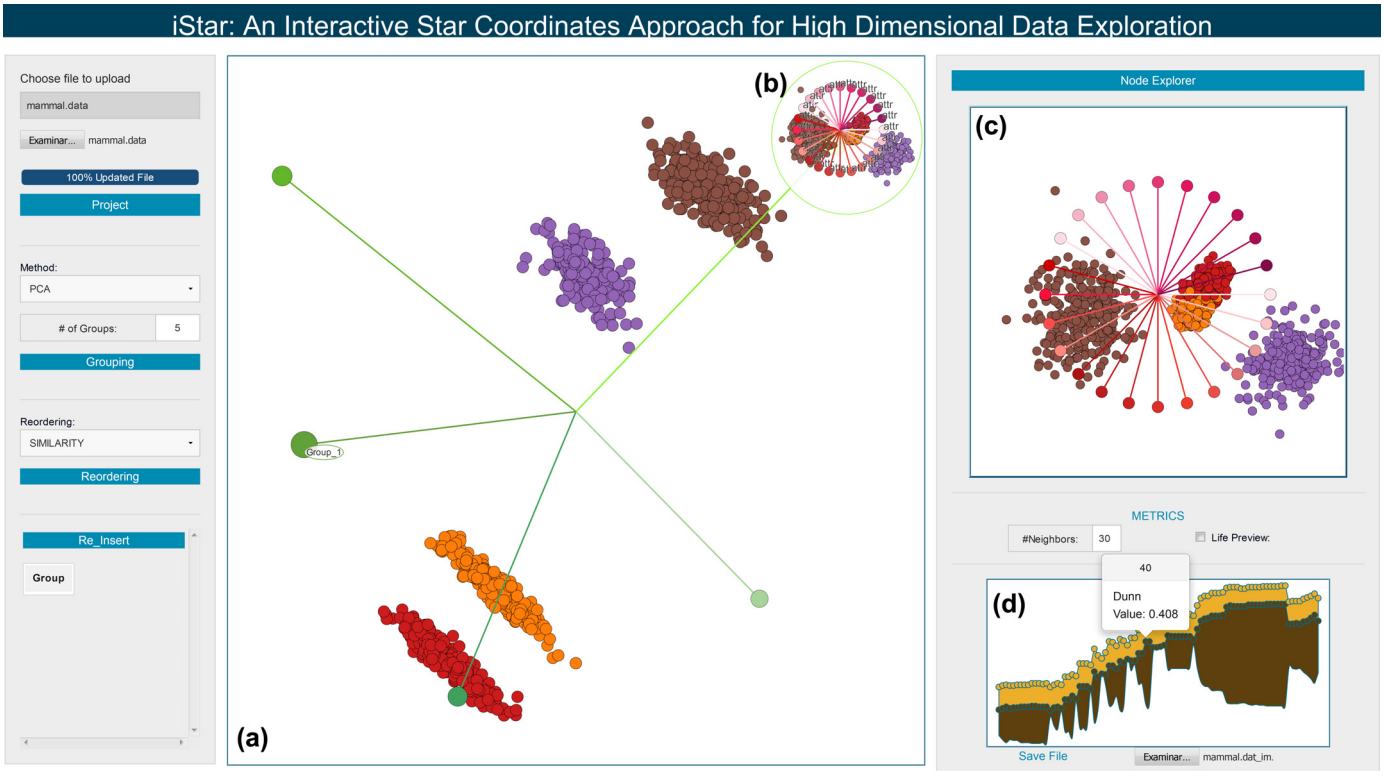


Fig. 7. An overview of iStar components: (a) Main visualization panel, (b) Node Preview magnifies axes grouped in a given node, (c) Node Explorer details clustered axes of a given node as an independent visualization and (d) Quality Visualizer shows step-by-step information about the layout quality during user interaction, each colored layer represents the evolution of a metric, i.e., topology preservation in yellow and Dunn index in brown.

Table 1
Datasets description and quality metrics results. Bold values indicate the best scores.

ID	Name	#Instances	#Attr	Class?	#Groups	Topology Preservation						Dunn Index					
						RadViz	Star Coordinates	DIFGBC	i* (PCA)	i* (Var)	i* (Ctr)	RadViz	Star Coordinates	DIFGBC	i* (PCA)	i* (Var)	i* (Ctr)
DT1	Mammals	200	72	✓	5	0.4078	0.4025	0.4812	0.4898	**0.4924**	0.4711	0.2599	0.2731	0.5343	0.7615	**0.9870**	0.3706
DT2	WDBCSTD	569	30	✓	5	0.0992	0.1570	0.1848	0.1917	0.2050	**0.2090**	0.2715	0.5132	0.2486	0.4888	**0.6226**	0.6209
DT3	SpamBase	1000	57	✓	8	0.1150	0.1368	0.1246	**0.1438**	0.1230	0.1371	0.1440	0.1358	0.1805	0.2178	0.1842	**0.2381**
DT4	texture	792	40	✓	6	0.2409	0.2424	0.2248	**0.3429**	0.3238	0.3302	0.0163	0.0282	0.0146	**0.1901**	0.1322	0.1026
DT5	Segmentation Normcols	2100	19	✓	8	0.2519	0.2292	0.2161	0.2892	**0.3060**	0.2827	0.0868	0.0584	0.0525	0.0866	0.0341	**0.0962**
DT6	ETHZ	917	3963	✓	8	0.1928	0.2051	0.1321	0.2425	0.1917	**0.3101**	0.0192	0.0284	**0.0346**	0.0298	0.0102	0.0321
DT7	Basketball	462	36	✗	5	–	–	–	–	–	–	–	–	–	–	–	–
DT8	Image database	420	384	✗	5	–	–	–	–	–	–	–	–	–	–	–	–
DT9	MICE	1080	80	✓	5	0.1156	0.1133	0.1403	0.2469	0.2766	**0.2839**	0.0118	0.0270	0.0367	0.0714	**0.1689**	0.0665
DT10	satimage	996	36	✓	10	0.0723	0.0666	0.1995	**0.3424**	0.3192	0.2890	0.0039	0.0247	0.0150	**0.1273**	0.1147	0.0950
DT11	All reduced	1000	63	✓	8	0.0503	0.0508	0.0543	0.0550	0.0526	**0.0595**	0.0174	0.0146	0.0170	0.0622	0.0240	**0.0721**
DT12	Optdigits	1000	64	✓	5	0.1736	**0.1748**	0.0761	0.1449	0.1062	0.1215	0.0493	**0.0525**	0.0150	0.0277	0.0517	0.0387
DT13	Movement Libras	360	90	✓	5	0.2370	0.2317	0.2842	**0.3008**	0.2832	0.2911	0.0212	0.0304	0.0194	0.0396	**0.0417**	0.0350
DT14	Sonar	208	60	✓	5	0.3316	0.3270	0.3379	**0.3624**	0.3267	0.3321	0.3103	0.2308	0.4310	**0.4355**	0.3199	0.3271
DT15	Spectf Heart	267	44	✓	8	0.1752	0.1676	0.2323	**0.2543**	0.2457	0.2512	0.0044	0.0093	0.0311	0.0414	**0.0924**	0.0554
DT16	Ionosphere	351	33	✓	10	0.2332	0.2253	0.2945	**0.3648**	0.3367	0.3251	0.4335	**0.4670**	0.1680	0.4326	0.4497	0.3080
DT17	FiberNotnorm	901	30	✓	10	0.3189	0.3333	0.2200	0.3300	0.2589	**0.3515**	0.0252	0.0177	0.0193	0.0572	0.0544	**0.0673**
DT18	FreeFoto	3462	128	✓	8	0.0474	0.0496	0.0940	**0.1108**	0.1009	0.1014	0.0300	0.0211	0.0143	0.0317	0.0158	**0.0371**
DT19	Elephant	1289	231	✓	6	0.0530	0.0530	0.0736	**0.0907**	0.0763	0.0880	0.2007	0.2136	0.0046	**0.3747**	0.1465	0.3084
DT20	Primarytumor	339	17	✓	6	0.1923	0.1898	0.1782	**0.2699**	0.2368	0.2043	0.0059	0.0049	0.0090	0.0343	0.0142	**0.0343**
DT21	Shapes	480	28	✓	10	0.3706	0.3576	0.3938	**0.4414**	0.4306	0.4265	0.0022	0.0079	0.0049	0.0184	**0.0185**	0.0155
DT22	Segment	2310	19	✓	5	0.2457	0.2254	0.2236	0.2743	0.2855	**0.3684**	0.0446	0.0444	0.0204	0.1072	0.1340	**0.8648**
DT23	Dermatology	358	34	✓	10	0.1872	0.2274	0.2880	0.3351	**0.3509**	0.3140	0.1218	0.0367	0.0649	0.1583	**0.2162**	0.1480
DT24	Physhing	1000	30	✓	5	0.0876	0.0792	0.1384	**0.1864**	0.1413	0.1416	0.0544	0.0437	0.3917	**0.4443**	0.4392	0.4150
DT25	Qsar	1055	41	✓	5	0.1399	0.1439	0.1612	**0.2013**	0.1876	0.1884	0.1887	0.0930	0.3354	**0.5600**	0.5050	0.5073
DT26	VEHICLE	846	18	✓	5	0.1573	0.1460	0.2107	0.2706	**0.2832**	0.2644	0.0108	0.0344	0.0653	0.0935	**0.1232**	0.0686
DT27	Twonorm	1000	20	✓	5	0.0496	0.0477	**0.0800**	0.0732	0.0782	0.0580	0.0471	0.0396	0.1714	**0.2890**	0.2482	0.1916

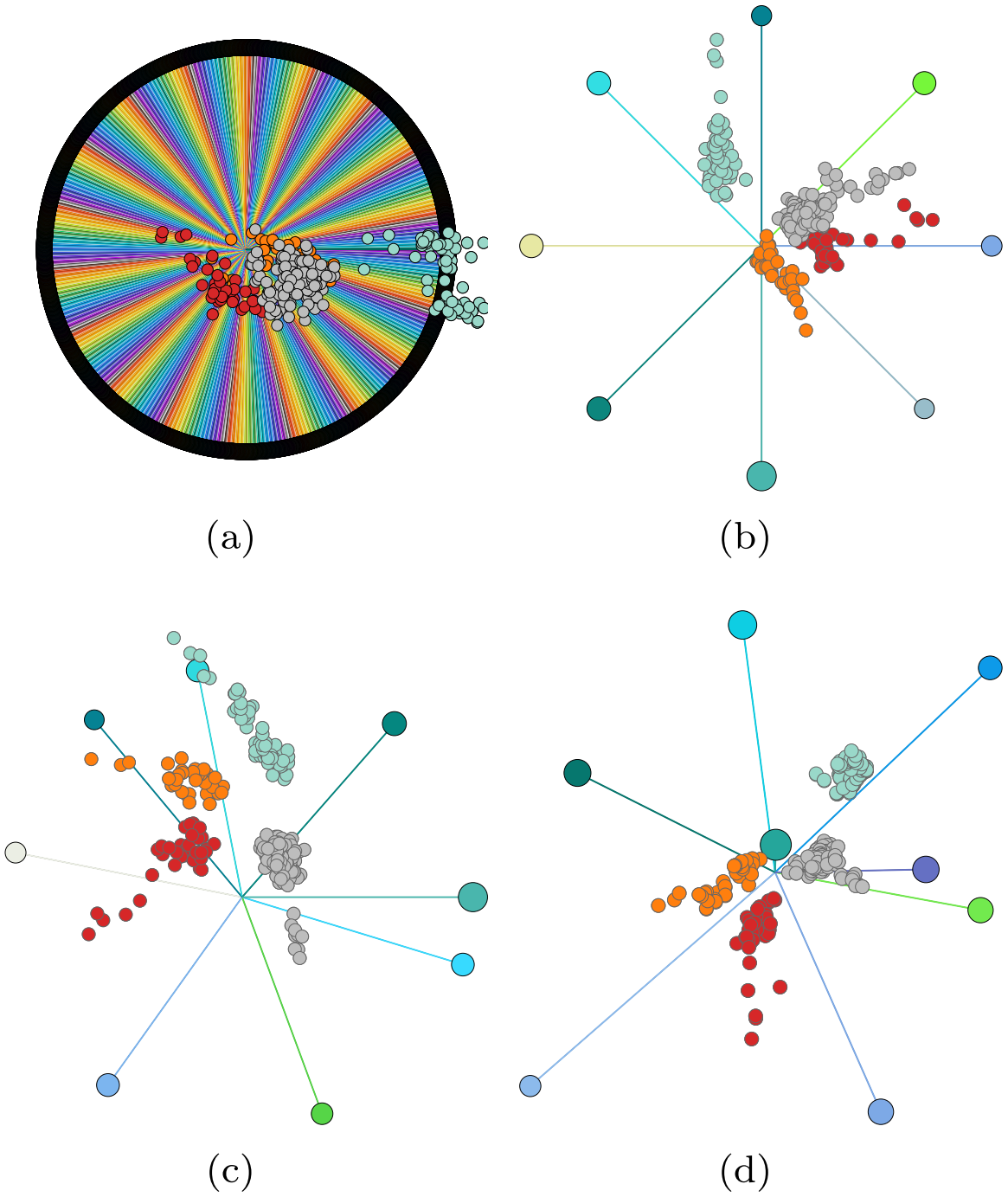


Fig. 8. Automatically setting an initial configuration for a 3963-dimensional dataset with 257 instances: (a) mapping data using all dimensions, (b) clustering by PCA, (c) reordering by optimization. In (d), a configuration slightly tuned using interactive operations is shown.

where s is a fixed number. Typically $k=4$ and $s=10$. The global topology preservation is:

$$r = \frac{1}{3mk}\sum_{j=1}^{m} r_j \tag{7}$$

The value $r \in [0,1]$ where $r=1$ is the perfect topology preservation.

Dunn Index (Du) [25] identifies clusters that are compact and well separated. Du is defined as follows:

$$Du(K) = \min_{i=1,\ldots,K}\left(\min_{j=i+1,\ldots,K}\left(\frac{D(C_i,C_j)}{\max_{l=1,\ldots,K} d(C_l)}\right)\right) \tag{8}$$

where $d(C_i) = \max_{x,y \in C_i}(Dist(x,y))$ is the diameter of a cluster, $D(C_i,C_j) = \min_{x \in C_i, y \in C_j}(Dist(x,y))$ is the distance between clusters and K is the number of clusters. Large values of $Du(K)$ suggest the presence of compact and well separated clusters.

Datasets and Test Settings.

We use 25 distinct datasets to assess the performance of iStar when compared against traditional RadViz [8], Star Coordinates [4] and the recent DIFGBC [13] method. Table 1 shows the datasets and their characteristics. The datasets DT1, DT2, DT6 were extracted from [26], DT3-DT5 from [27] and DT9-DT27 from [28].

Fig. 8 shows our proposed mechanism for data exploration, step-by-step, on a dataset containing 3963 attributes and 257 instances.

4.2. Comparison

The metrics previously describe are used to assess the effectiveness of our approach when visualizing high-dimensional data. The conventional implementations of Radviz, Star Coordinates, and DIFGBC are used for the sake of comparison.

Fig. 9 presents quantitative results from the Dunn Index and Topology Preservation metrics. Box plots gather metrics values from all datasets described in Table 1. When running iStar, attributes were clustered in a drastically lower number of groups as described in the sixth column of the Table 1. Note that all the three variations of iStar outperformed RadViz, SC and DIFGBC visualizations, showing that attribute clustering is a feasible alternative. In fact, Fig. 9 shows that neighborhood and layout structure are better represented when attributes are properly clustered, mainly when using PCA and Centroid schemes. It is worth noticing that the arrangements obtained by iStar have not been interactively changed, therefore the results could be improved, in particular cases, via expert user interaction.

Fig. 10 depicts qualitative results comparing iStar against RadViz, Star Coordinates and DIFGBC. Notice that the layouts automatically generated by iStar tend to be similar and, in some cases, better defined than the ones resulting from the other techniques, showing once again that the process of grouping attributes is a viable option. In fact, for datasets DT1, DT2, DT5, DT9 and DT10, iStar clearly performed better than RadViz, SC, and DIFGBC. For more intricate datasets such as DT4 and DT6 (higher dimensional datasets), Radviz and DIFGBC tend to concentrate instances close to the center of the layout, while SC provides more spread configurations. iStar results in intermediate configurations from which users can start their interactive exploratory analysis.

4.3. Evaluation

With the purpose of verifying the statistical significance of the variability results, we applied one-way ANOVA test with 5% of significance level and F-Critical (FC) value of 2.277, assuming the values obtained from the metrics are independent samples (columns in Table 1). Two hypotheses are handled, a null hypothesis which states that all population means are equal and the alternative hypothesis stating that at least one is different.

For the topological preservation metric, we obtained an F-calculated (F_c) value of 2.37 with a probability of occurrence (p) of 0.043. Notice that $p < 0.05$ (5%) and $F_c < FC$ so the alternative hypothesis is validated stating that at least one method has significant results. For the Dunn index metric, we obtained $F_c = 2.38$ and $p = 0.042$, validating the alternative hypothesis as well. These conclusions are supported by the boxplots in Fig. 9.

Now that the null hypothesis has been rejected, we want to know what are the method(s) with significant difference. For this task we have used Fisher Test which performs a comparison between the means for each pair of methods by using a threshold calculated with t-test. If the difference between two methods is above their threshold then they will belong to different groups. The methods that do not share a group are significantly different. For both of our metrics Fisher Test divided and sorted the methods into two groups A and B as shown in Tables 2 and 3.

For topological preservation metric (Table 2), group A contains iStar based on PCA and Centroids methods. Group B contains SC and Radviz methods. However, iStar based on Variance and DIFGBC methods are in both groups, which means that these methods do not have significantly different results from the others. For Dunn index metric (Table 3), group A contains iStar based on PCA, Variance, and Centroids methods. Group B contains SC and Radviz methods. Finally, only DIFGBC method is in both groups.

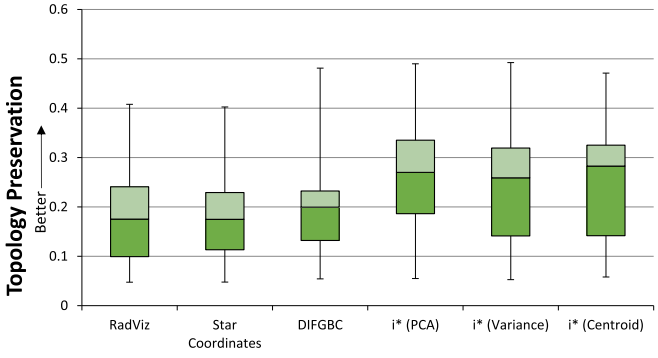
Note that iStar based on PCA obtained highest mean value for both metrics. Moreover, iStar based on PCA and Centroids were kept on the group A for both metrics, i.e., they showed significant difference in both cases.

5. Case study

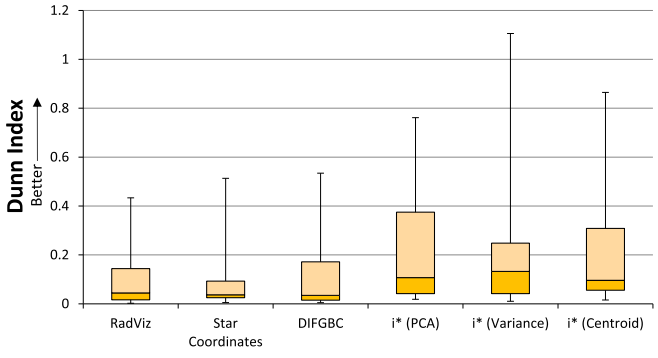
In order to evaluate iStar in a real application, we provide two case studies. The first case study handles a dataset containing the performance of 462 National Basketball Association (NBA) players during 2009-2010 season obtained from [29] (named as DT7 in Table 1). The second case study explores a set of images containing 420 instances obtained from the Caltech 101 [30] and CBIR [31] image databases (named as DT8 in Table 1).

On the first case study we handle data instances with 36 attributes which describe statistics about players, on/off court, and information related to teammates and opponents. Among some of the most relevant attributes are: points scored by the player or team, number of possessions (offensive/defensive), number of rebounds by the player and team (when player is off court), and number of minutes played by the player.

Fig. 11 depicts iStar when handling DT7 data. Fig. 11(a) shows the 36 attributes and the point cloud resulting from SC mapping. The initial 36 attribute axes visualization gives rise to a linearly spread layout that does not reveal any prominent cluster or pattern. In the initial configuration, length and position of the attribute axes are uniform. Fig. 11(b) shows the layout resulting from applying iStar's attribute clustering and reordering. Clustering has been computed with the PCA-based scheme (number of clusters equal to 5) and the attribute axes were arranged using the brute force mechanism. Notice that three clusters show up in the layout, the larger one



(a)



(b)

Fig. 9. Comparison against radial visualizations using twenty-five datasets endowed with class information (DT1-DT6, DT9-DT27) as detailed in Table 1.

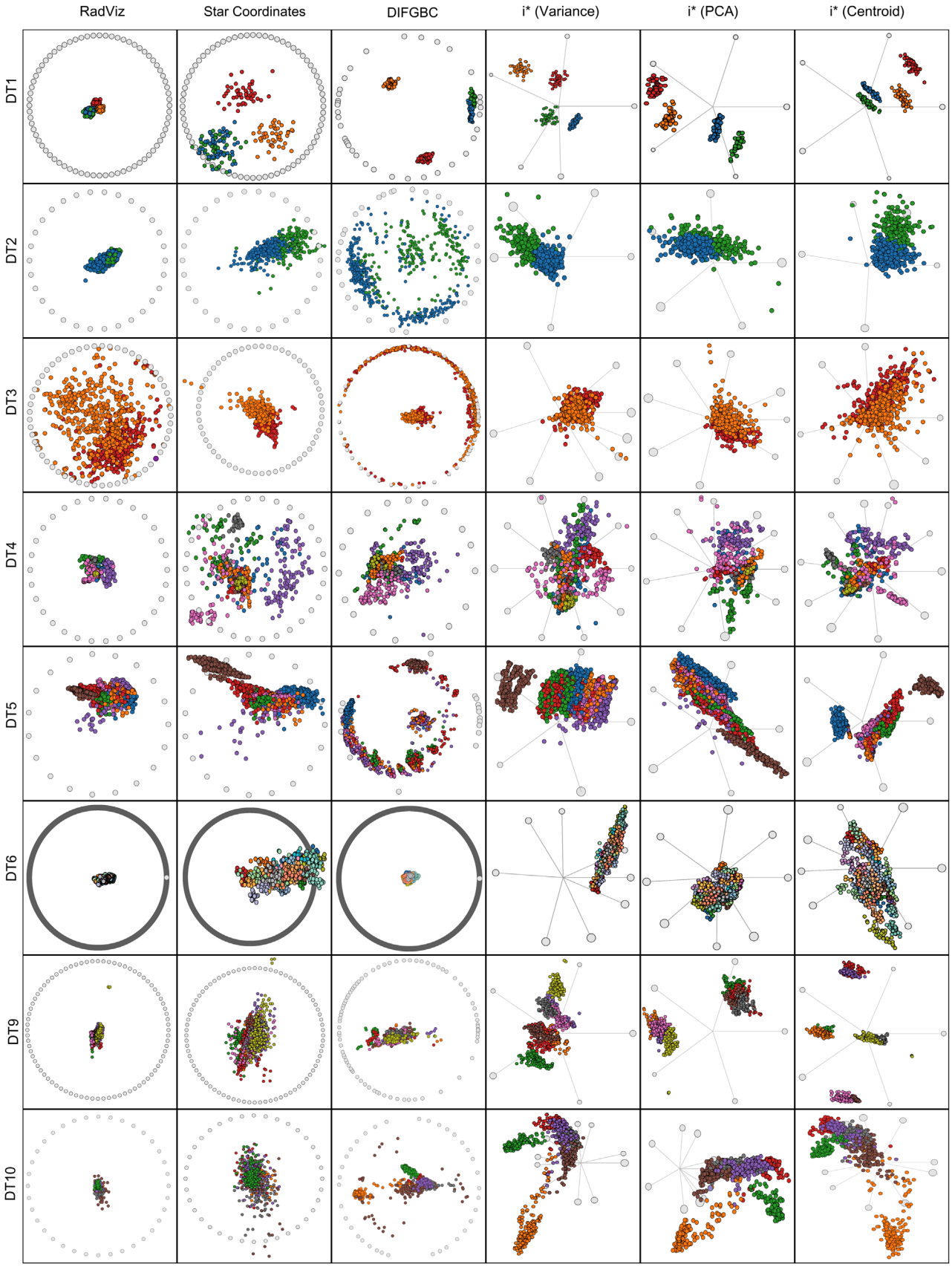


Fig. 10. Configurations produced by RadViz, Star Coordinates, DIFGBC and iStar (variance, PCA, centroid) for the first 8 classified datasets in Table 1. For iStar results, we use only the automatic axes grouping/reordering mechanism, i.e., no user interaction was performed. It is easy to notice that our method better discriminates the clusters when compared against other techniques, even without the use of interaction.

Table 2
Fisher Test for Topology Preservation metric results.

Topology Preservation		
Method	**Mean**	**Group**
i* (PCA)	0.2542	A
i* (Centroid)	0.2480	A
i* (Variance)	0.2408	A B
DIFGBC	0.2018	A B
Star Coordinates	0.1833	B
RadViz	0.1818	B

Table 3
Fisher Test for Dunn Index metric results.

Dunn Index		
Method	**Mean**	**Group**
i* (PCA)	0.2072	A
i* (Centroid)	0.2059	A
i* (Variance)	0.2046	A
DIFGBC	0.1160	A B
Star Coordinates	0.0980	B
RadViz	0.0952	B

placed in the center, and the other two on the bottom part of the layout. The chart next to each attribute axis shows the attributes clustered in that axis. Most attributes have been grouped into the dark blue axis, indicating they are correlated. Fig. 11(c) shows twenty different arrangements for the DT7 dataset. The top row shows results from PCA-based clustering, while the bottom row shows clusters generated by the variance method. Most results reveal the presence of two main groups, whereas only a few ones expose three clusters. In most cases, our reordering method by brute force was applied. Fig. 11(d) illustrates some interactive resources in action. For instance, some attributes were interactively moved from one cluster to another, as for example the attribute named *WeightedDRebRateOnCourtMinusOffCour*, which moved from the dark blue to the blue axis. Scaling and rotation operations were also interactively performed to better discriminate the three clusters. The zoomed disks show the Node Explorer widget depicting the content of clustered axes. We highlight two clusters with relevant information to be reported. In the blue disc, we see that the green cluster is isolated from the rest, revealing that the attribute *WeightedDRebRateOnCourtMinusOffCour* is very relevant for that cluster. More precisely, the attribute says that players in the green cluster performed a considerable number of rebounds for their team when those players were on court, but a lower number of

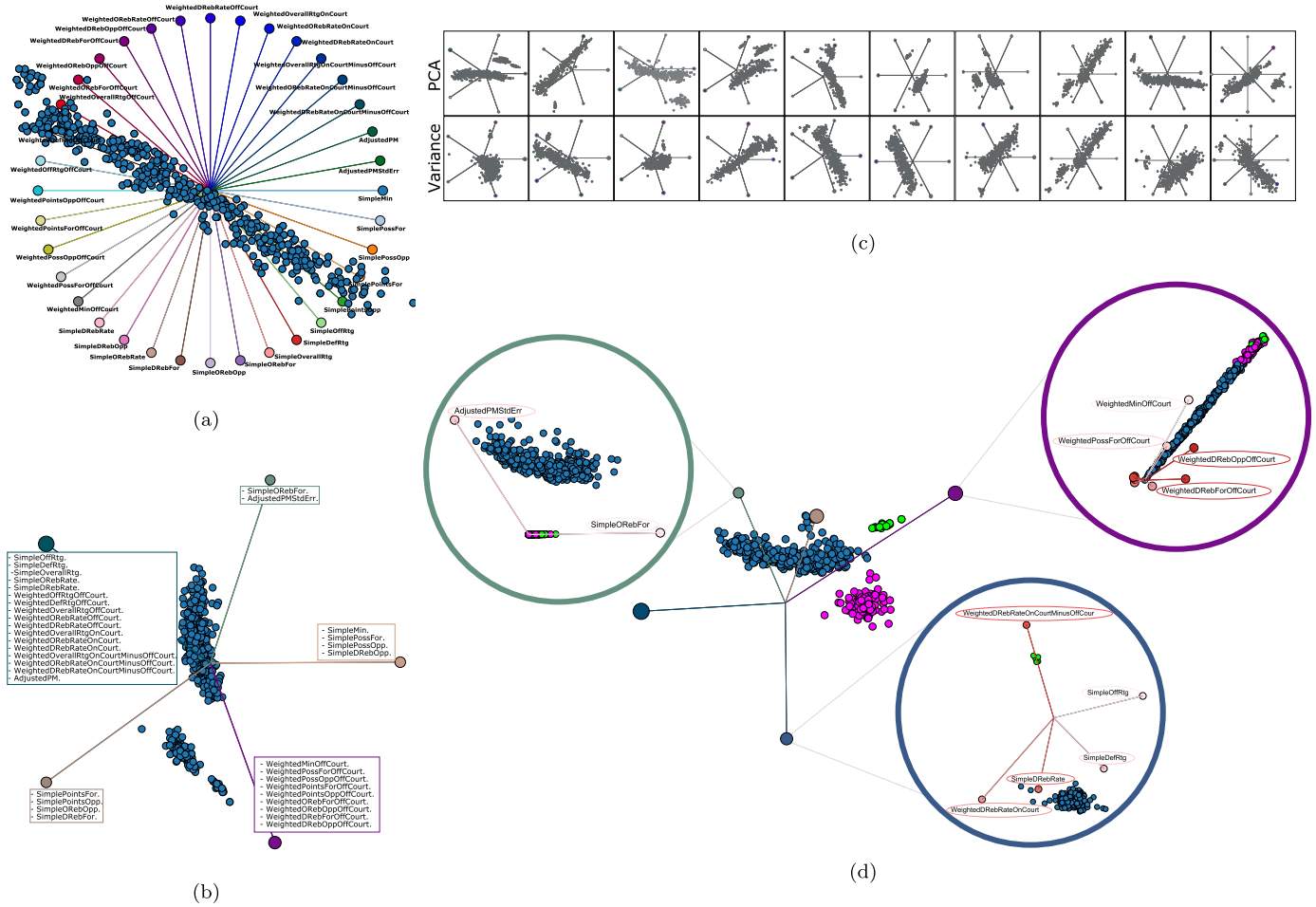


Fig. 11. Visualizing a dataset containing 462 National Basketball Association (NBA) players during 2009–2010 season. (a) projection of the entire set of attributes, (b) the resulting arrangement after applying both PCA-based clustering and reordering of attributes, (c) 20 other possible projections using PCA and Variance iStar variations without user interaction reveal 2 clusters in most cases, and (d) final projection has discovered a new well-defined cluster after perform some interactive operations, in addition, this configuration shows the Node Explorer releasing important characteristics of the data in each node.

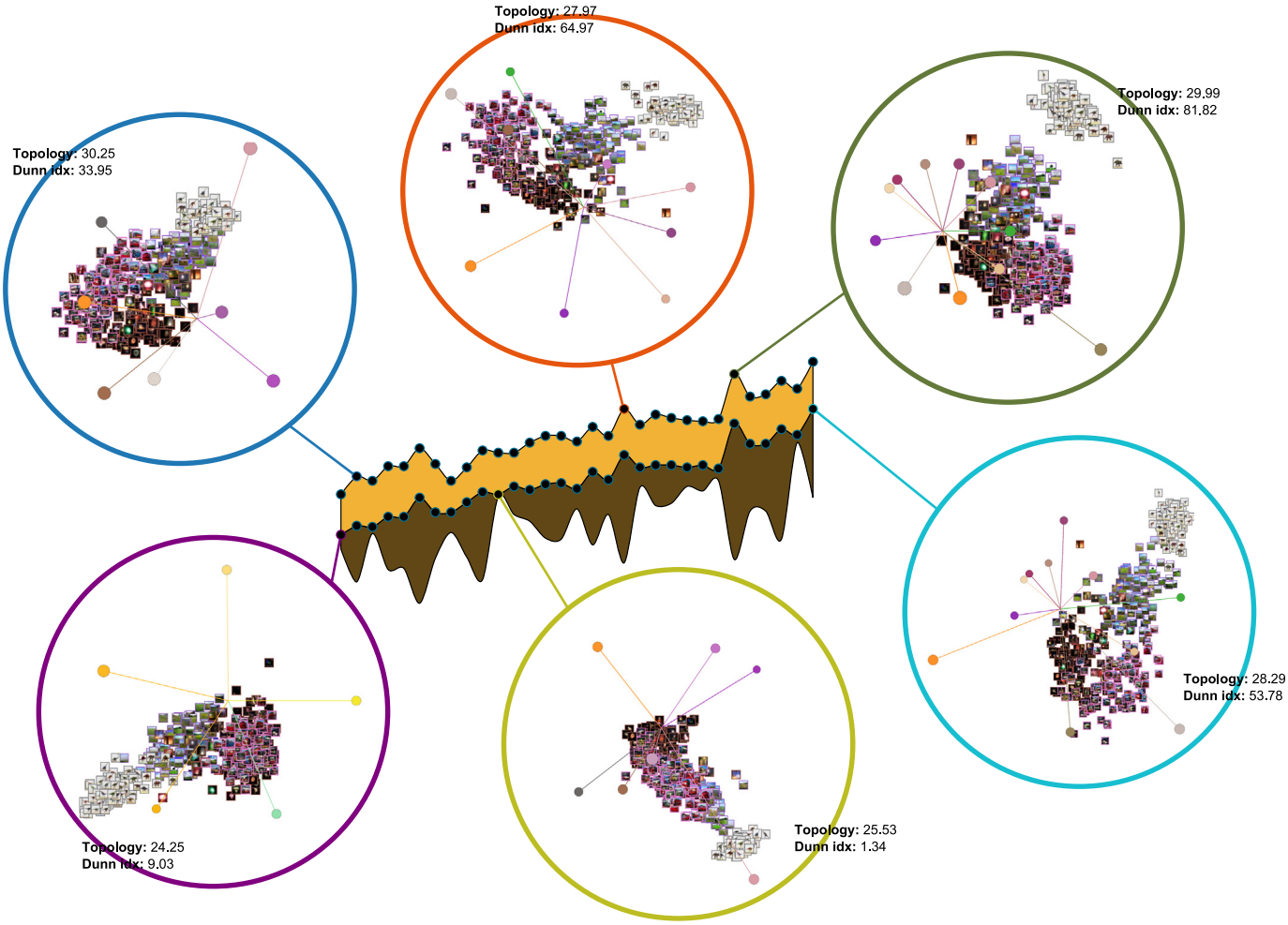


Fig. 12. Exploring 6 different states on the Quality Visualizer widget during user interaction for a color images collection (DT8 in Table 1). Each colored layer represents the evolution of a metric: topology preservation in yellow and Dunn index in brown.

accumulative rebounds for themselves since the *SimpleDRebRate* attribute (individual rate of defense rebounds) has not strongly influenced in their position in the layout. The purple disc magnifies attributes related to off court. The green cluster is now located on the top right part of the layout. It is easy to see that the green group of players have a large value of *WeightedMinOffCourt* and *WeightedPossForOffCourt*, meaning long periods and possessions off court. Finally, in the green disc, the green group of players is located near the center of the layout, meaning poor performance for *SimpleOREbFor* (accumulative number of offensive rebounds) and *AdjustedPMStdErr* (relevance of the player for his team on game). Therefore, those players presented a good performance with very little contribution for their teams during the season.

Among the players in the green group we find famous players such as Blake Griffin, who had been selected from the draft 2009 but he missed the entire season due to a surgery, Ming Yao who did not play the entire season, Brandan Wright who underwent shoulder surgery and missed most the season, and Quincy Douby who was released by the Toronto Raptors but he did not play any match.

The second case study aims to evaluate iStar's effectiveness when visualizing classified color images. In this context, each data instance has 384 attributes describing color moments of images with dimensions 256x256. The images were split in 64 blocks, each block providing color means and standard deviation as attributes for each RGB channel. Contrary to the first case study, this dataset has class information, thus grouping instances as: Firewalls(100), Dinosaur(100), Grass(120) and Cars(100).

Fig. 12 shows the stacked graph resulting from 31 layout quality measures computed during the visual analytics process. The purple disc shows the layout in the very beginning of the exploratory process, which has been automatically generated via PCA attribute clustering and variance reordering. Notice that instances are divided in three groups: the bottom left containing images of Dinosaurs; the central group with images from Grass class and the right group containing a mixed of Cars and Firewalls images. The blue disc on the top left shows the layout in the first stage of the exploratory process, where instances tend to be divided into two groups: Dinosaurs and Firewalls. The disc on the bottom center shows one of the worst scenarios where instances are gathered on two groups: the Dinosaurs group (bottom right) and a large group (center) containing images from three different classes. The remaining disks on the top center and on the right show good quality layouts according to the quality metrics and clearly revealed by the stacked graph quality visualization widget. One can see that those layouts are compact while well separating the instances from the different classes.

The finds described above show the usefulness of iStar as a visual analytics tool. Attribute clustering, layout quality visualization and the interactive resources implemented in iStar turned out to be effective, making iStar an interesting alternative for Star Coordinates based data exploration.

6. Conclusion and future work

In this paper we presented iStar, a Star Coordinates based visualization technique that relies on attribute clustering, axes reordering, and interactive resources to handle data with a large number of attributes. Provided comparison shows that iStar outperforms conventional Star Coordinates and RadViz visualizations. Moreover, the provided case studies show the effectiveness of iStar as a visual analytics technique. The traits endowed in iStar render it an interesting alternative for visual analytics tasks through Star Coordinates plots.

As a future work we will focus on layout enrichment schemes as proposed in [32] and [33] in order to make radial visualization layouts more informative, conveying gist information to assist users in the visual analytics process.

Acknowledgment

We would like to thank the financial support from the National Council for Science, Technology and Technological Innovation - CONCYTEC, Peru (grant FONDECYT 011-2013 Master Program), the São Paulo Research Foundation - FAPESP (grants #2013/00191-0 and #2011/22749-8) and the National Counsel of Technological and Scientific Development - CNPq, Brazil (grant #302643/2013-3).

Appendix A. Supplementary data

Supplementary data associated with this article can be found in the online version at http://dx.doi.org/10.1016/j.cag.2016.08.007.

References

[1] Liu S, Maljovec D, Wang B, Bremer P-T, Pascucci V. Visualizing high-dimensional data: Advances in the past decade. In: Borgo R, Ganovelli F, Viola I. (Eds.), Proceedings of Eurographics Conference on Visualization (EuroVis) - STARs, The Eurographics Association, 2015, p. 127–47.
[2] Draper G, Livnat Y, Riesenfeld R. A survey of radial methods for information visualization. IEEE Trans Vis Comput Graph 2009;15(5):759–76.
[3] Kandogan E. Star coordinates: a multi-dimensional visualization technique with uniform treatment of dimensions. In: Proceedings of the IEEE information visualization symposium, late breaking hot topics; 2000, p. 9–12.
[4] Kandogan E. Visualizing multi-dimensional clusters, trends, and outliers using star coordinates. In: Proceedings of the seventh ACM SIGKDD international conference on knowledge discovery and data mining, ACM; 2001, p. 107–116.
[5] Rubio-Sanchez M, Sanchez A. Axis calibration for improving data attribute estimation in star coordinates plots. IEEE Trans Vis Comput Graph 2014;20(12):2013–22.
[6] Sun Y, Yuan J, Hu Y, Xiao W. An improved multivariate data visualization technique. In: Proceedings of international conference on information and automation, ICIA; 2008, p. 1525–30.
[7] Diehl S, Beck F, Burch M. Uncovering strengths and weaknesses of radial visualizations–an empirical approach. IEEE Trans Vis Comput Graph 2010;16(6):935–42.
[8] Hoffman P, Grinstein G, Marx K, Grosse I, Stanley E. Dna visual and analytic data mining. In: Visualization '97, Proceedings, 1997, p. 437–441.
[9] Rubio-Sanchez M, Raya L, Diaz F, Sanchez A. A comparative study between radviz and star coordinates. IEEE Trans Vis Comput Graph 2016;22(1):619–28.
[10] Sharko J, Grinstein G, Marx K. Vectorized radviz and its application to multiple cluster datasets. IEEE Trans Vis Comput Graph 2008;14(6):1427–44.
[11] Russell A, Daniels K, Grinstein G. Voronoi diagram based dimensional anchor assessment for radial visualizations. In: Proceedings of the 16th international conference on information visualisation (IV); 2012, p. 229–33.
[12] Ono JHP, Sikansi F, Correa DC, Paulovich FV, Paiva A, Nonato LG. Concentric radviz: visual exploration of multi-task classification. In: Proceedings of Sibgrapi – conference on graphics, patterns and images, 2015, p. 165–72.
[13] Cheng S, Mueller K. Improving the fidelity of contextual data layouts using a generalized barycentric coordinates framework. In: Proceedings of IEEE Pacific Visualization Symposium (PacificVis), 2015, p. 295–302.
[14] Lehmann DJ, Theisel H. Orthographic star coordinates. IEEE Trans Vis Comput Graph 2013;19(12):2615–24.
[15] Tsai C-Y, Chiu C-C. A clustering-oriented star coordinate translation method for reliable clustering parameterization. In: Proceedings of advances in knowledge discovery and data mining, Vol. 5012 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2008, p. 749–58.
[16] Machado Filho O, Evsukoff A, Ebecken NFF. Starcluster: a visualization, clustering and classification tool. In: Proceedings of fourth international conference on data mining, 2003, p. 205–14.
[17] Teoh ST, Ma K-L. Starclass: Interactive visual classification using star coordinates. In: Proceedings of the third SIAM international conference on data mining, 2003, p. 178–85.
[18] Pearson K. On lines and planes of closest fit to systems of points in space. Philos Mag 1901;2(6):559–72.
[19] Lloyd S. Least squares quantization in PCM. IEEE Trans Inf Theory 1982;28(2):129–37.
[20] Ankerst, M, Berchtold S, Keim D. Similarity clustering of dimensions for an enhanced visualization of multidimensional data. In: Proceedings of IEEE Symposium onInformation Visualization; 1998, 153, p. 52–60.
[21] Heinrich J, Weiskopf D. State of the art of parallel coordinates. In: Association E. (Ed.), STAR Proceedings of Eurographics, 2013, p. 95–116.
[22] Yang H-f, Wei Y. Improved genetic algorithm for TSP. J Chongqing Inst Technol 2007;5:022.
[23] Byron L, Wattenberg M. Stacked graphs - geometry amp; aesthetics. IEEE Trans Vis Comput Graph 2008;14(6):1245–52.
[24] König A. Interactive visualization and analysis of hierarchical neural projections for data mining. IEEE Trans Neural Netw 2000;11(3):615–24.
[25] Xu R, Wunsch D. Clustering (IEEE press series on computational intelligence); 2009.
[26] VICG, Multivariate Datasets, ⟨http://infoserver.lcad.icmc.usp.br/infovis2/DataSets⟩; 2014.
[27] VICG, PAn - Projection Analyzer, ⟨https://code.google.com/p/projection-analyzer/downloads/list⟩; 2011.
[28] Lichman M. UCI machine learning repository, ⟨http://archive.ics.uci.edu/ml⟩; 2013.
[29] CKAN, datahub, ⟨https://datahub.io/about⟩.
[30] Fei-Fei L, Fergus R, Perona P. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In: Proceedings of conference on computer vision and pattern recognition workshop, CVPRW '04, 2004, p. 178–178.
[31] Wang J, Li J. Simplicity semantics-sensitive integrated matching for picture libraries. IEEE Trans Pattern Anal Mach Intell 2001;23(9):947–63.
[32] Joia P, Petronetto F, Nonato L. Uncovering representative groups in multidimensional projections. Comput Graph Forum 2015;34(3):281–90.
[33] Gomez-Nieto E, San Roman F, Pagliosa P, Casaca W, Helou E, Oliveira MF, et al. Similarity preserving snippet-based visualization of web search results. IEEE Trans Vis Comput Graph 2014;20(3):457–70.