Adaptable Radial Axes Plots
for Improved Multivariate Data Visualization

M. Rubio-Sánchez[1], A. Sanchez[1,2] and D. J. Lehmann[1,3]

[1]Universidad Rey Juan Carlos, Madrid, Spain
[2]Center for Computational Simulation, Madrid, Spain
[3]University of Magdeburg, Magdeburg, Germany

Abstract

Radial axes plots are multivariate visualization techniques that extend scatterplots in order to represent high-dimensional data as points on an observable display. Well-known methods include star coordinates or principal component biplots, which represent data attributes as vectors that define axes, and produce linear dimensionality reduction mappings. In this paper we propose a hybrid approach that bridges the gap between star coordinates and principal component biplots, which we denominate "adaptable radial axes plots". It is based on solving convex optimization problems where users can: (a) update the axis vectors interactively, as in star coordinates, while producing mappings that enable to estimate attribute values optimally through labeled axes, similarly to principal component biplots; (b) use different norms in order to explore additional nonlinear mappings of the data; and (c) include weights and constraints in the optimization problems for sorting the data along one axis. The result is a flexible technique that complements, extends, and enhances current radial methods for data analysis.

Categories and Subject Descriptors (according to ACM CCS):   [Human-centered computing]: Visualization—Visualization techniques [Probability and statistics]: Statistical paradigms—Statistical graphics
[Human-centered computing]: Visualization—Visualization theory, concepts and paradigms [Probability and statistics]: Statistical paradigms—Exploratory data analysis

1. Introduction

Multivariate visualization is an active research field whose goal consists of representing information related to multidimensional data samples, attributes, and the relationships between them, as faithfully as possible on low-dimensional observable displays. In this paper we focus on radial axes plots (see [DLR09, DBB10, LT16]), which are visualization techniques that extend scatterplots in order to consider as many variables as desired. Specifically, they show attribute information through vectors that represent radial axes, and high-dimensional data samples as points, on a two or three-dimensional graphic with one coordinate system.

Among these methods, star coordinates (SC) [Kan00, Kan01] and principal component biplots (PCB) [Gab71, GH95, Gre10, GGLlR11] stand out as one of the most prominent radial axes methods for exploratory data analysis. Firstly, SC is highly interactive since it allows users to select any configuration of axes, and therefore to generate plots associated with any linear mapping, when performing exploratory analysis tasks like searching for cluster structure, outliers, or data with desired characteristics [LKZ*15, RSRDS16]. However, one of its major drawbacks is the difficulty to estimate (i.e., recover) original attribute values visually [DLR09, RSS14]. On the other end of the spectrum, PCB [Gab71]

consist of specific scaled versions of the principal component analysis (PCA) plot. In particular, they can be understood as the product of solving a mathematical optimization problem whose goal consists of finding a set of axis vectors and embedded points that will allow to estimate attribute values optimally, by projecting points orthogonally onto adequately calibrated (i.e., labeled) axes. Moreover, depending on scaling factors, PCB can also optimize properties such as distance preservation between samples, or correlation approximation between variables. Nevertheless, in contrast with SC, they build static visualizations where users can not update the axis vectors interactively.

The main contribution of this paper is a flexible projection technique based on a set of radial axes that can be understood as a hybrid approach between SC and PCB. The projections are defined through convex norm approximation problems that are extensions to the PCB optimization problem. In particular, the approach also minimizes the errors users would make when approximating high-dimensional attribute values by projecting mapped points orthogonally onto the axes. However, it incorporates elements and variants that not only offer users a richer set of projections to explore, but it also facilitates several data analysis tasks. In particular, the proposed framework considers: (a) the possibility of modifying the

axis vectors freely and interactively, which bridges the gap between SC and PCB; (b) using alternative norms in order to project the data nonlinearly and explore it from additional perspectives; and (c) including weights and constraints in the optimization problem for sorting the plotted points according to one attribute. The result is a flexible and powerful technique that complements, extends, or enhances current radial methods for data analysis.

The rest of the paper is organized as follows. Section 2 describes the most relevant radial axes methods, introducing the used mathematical notation. In Sec. 3 we describe our proposed approach and its relationships with other radial methods. Finally, Sec. 4 presents the main conclusions and a discussion.

2. Related work and notation

This section recapitulates several radial axes methods that map data samples from a high n-dimensional data space onto a lower m-dimensional observable display, with $n \geq 3 \geq m$ (in this paper we will focus on visualizations on a plane, i.e., $m = 2$). The methods therefore represent each data sample $\mathbf{x} \in \mathbb{R}^n$ by an embedded point $\mathbf{p} \in \mathbb{R}^m$ on the visualizations. The mappings rely on a set of n m-dimensional axis vectors $\mathbf{v}_i$, for $i = 1, \ldots, n$, with a common origin point, where $\mathbf{v}_i$ is associated with the i-th data variable. Throughout the paper $\mathbf{V}$ will denote an $n \times m$ matrix whose rows contain the axis vectors $\mathbf{v}_i$. In addition, given a data set of cardinality N, $\mathbf{X}$ will represent the $N \times n$ data matrix whose rows contain the data samples, while $\mathbf{P}$ will be an $N \times m$ matrix whose rows consist of the data samples' low-dimensional representations $\mathbf{p}_i$.

2.1. Principal component biplots

Principal component biplots are fixed static visualizations that show optimal configurations of axis vectors ($\mathbf{V}$) and embedded points ($\mathbf{P}$) for numerical data sets (other types of biplots can use categorical data [GH95, Gre10, GGLlR11]). They provide the most accurate estimates of attribute values across an entire data set (in a least-squares sense), and can also optimize distance preservation between samples, or correlation approximation between variables. Formally, they can be understood as the result of solving the following optimization problem:

$$\underset{\mathbf{P} \in \mathbb{R}^{N \times m}, \mathbf{V} \in \mathbb{R}^{n \times m}}{\text{minimize}} \quad \|\mathbf{P}\mathbf{V}^\mathsf{T} - \mathbf{X}\|_\mathrm{F}^2, \tag{1}$$

where the subscript F denotes the Frobenius norm. It is important to note that in this problem both axis vectors and embedded points are variables. The optimal solution follows:

$$\mathbf{P}\mathbf{V}^\mathsf{T} = \hat{\mathbf{X}}, \tag{2}$$

where $\hat{\mathbf{X}}$ is the optimal rank m approximation of the data matrix $\mathbf{X}$, according to the (squared) Frobenius norm, which can be characterized by the compact singular value decomposition of $\mathbf{X}$ [EY36]. In particular, let $\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{Z}^\mathsf{T}$ represent the singular value decomposition of $\mathbf{X}$, where the diagonal elements (singular values) in $\mathbf{D}$ appear in descending order. In that case $\hat{\mathbf{X}}$ can be expressed as follows:

$$\hat{\mathbf{X}} = \mathbf{U}_m \mathbf{D}_m \mathbf{Z}_m^\mathsf{T}, \tag{}$$

where the $N \times m$ matrix $\mathbf{U}_m$ corresponds to the first m columns of $\mathbf{U}$, the $m \times m$ diagonal matrix $\mathbf{D}_m$ is the first $m \times m$ diagonal block of $\mathbf{D}$ (i.e., it contains only the m largest singular values of $\mathbf{X}$), and the $n \times m$ matrix $\mathbf{Z}_m$ consists of the first m columns of $\mathbf{Z}$.

Since PCB require decomposing $\hat{\mathbf{X}}$ into the product of two matrices (see (2)), it is factorized as follows:

$$\hat{\mathbf{X}} = (k\mathbf{U}_m\mathbf{D}_m^{1-c})\left(\frac{1}{k}\mathbf{D}_m^c\mathbf{Z}_m^\mathsf{T}\right) = \mathbf{P}\mathbf{V}^\mathsf{T}, \tag{3}$$

for some suitable (scaling) constants k and c. Although users could select these values interactively, certain pairs are especially relevant since they optimize properties of the visualizations. For instance, when the decomposition consists of:

$$\mathbf{P} = \mathbf{U}_m\mathbf{D}_m, \quad \text{and} \quad \mathbf{V}^\mathsf{T} = \mathbf{Z}_m^\mathsf{T},$$

the obtained plot corresponds to the orthogonal projection of the data onto the subspace spanned by the m first eigenvectors of the data's covariance matrix (i.e., the orthogonal directions of maximum variance in the data). In other words, the result is the PCA plot. Another interesting PCB assigns:

$$\mathbf{P} = \sqrt{N}\mathbf{U}_m, \quad \text{and} \quad \mathbf{V}^\mathsf{T} = (1/\sqrt{N})\mathbf{D}_m\mathbf{Z}_m^\mathsf{T},$$

which not only allows to estimate original distances between data samples through distances between embedded points, but the dot products between the resulting axis vectors lead to optimal approximations of the covariances, variances, and correlation coefficients between the data variables (see [Gab71]).

One of the most interesting properties of PCB is the possibility to estimate original data attributes by projecting embedded points orthogonally onto adequately "calibrated" (i.e., labeled) axes, as illustrated in Fig. 1(a) with a subset of the Breakfast Cereal data set used in [YMSJ05]. Calibration consists of placing tick marks, and optionally numerical labels, along the axes. When the data is standardized these are usually located at integer positions, indicating standard deviation units. In the figure we have used standardized data, but have labeled the axes with original (non standardized) values.

In PCB the estimates are the rows of $\hat{\mathbf{X}}$. Therefore, due to (2), they correspond to the dot products between the embedded points in $\mathbf{P}$ and the axis vectors in $\mathbf{V}$. In particular, the vector of estimated attributes of a data sample $\mathbf{x}$ is defined through:

$$\hat{\mathbf{x}} = \mathbf{V}\mathbf{p}, \tag{4}$$

where $\hat{x}_i = \mathbf{v}_i^\mathsf{T}\mathbf{p}$. This implies that the distance between tick marks separating consecutive integers on the i-th axis must be $1/\|\mathbf{v}_i\|$.

In addition, it can be shown that the most accurate estimates can be obtained when the mean of the data is represented at the origin point of the axis vectors. In this regard, centering the data allows to represent the 0 value for every attribute at the origin point of the axis vectors, which facilitates labeling the axes. Moreover, standardizing the data allows to interpret distances between consecutive tick marks as standard deviation units.

In PCB users do not interact in order to select a set of desired axes. In other words, $\mathbf{V}$ can not be any arbitrary matrix. Thus, interactive applications include controls for data analysis and graphical
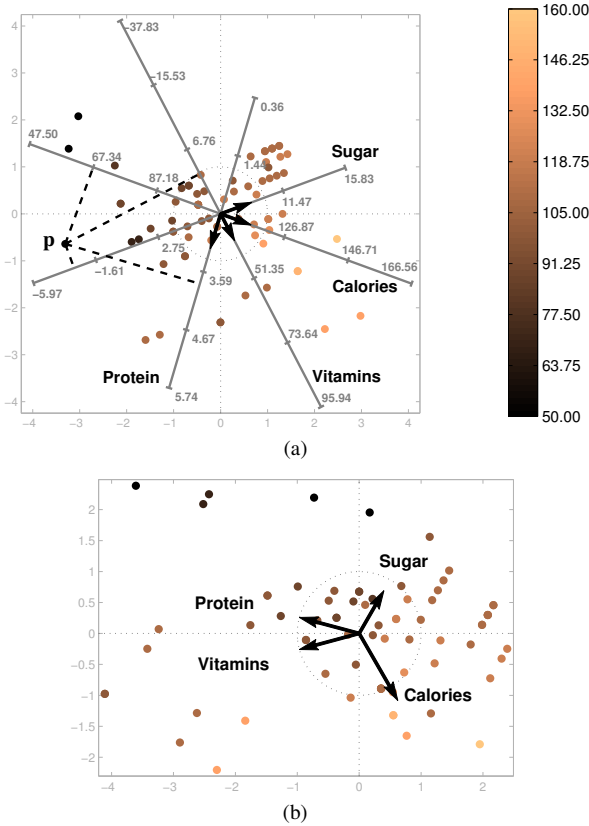
(a)

(b)

Figure 1: *Examples of radial axes plots. A principal component biplot of the Breakfast Cereal data set is shown in (a), where the specific configuration of the axis vectors leads to a PCA plot. The dots symbolize individual cereals, while the vectors and axes represent data variables. Users can find optimal approximations (estimates) of data attributes by projecting embedded points onto the labeled axes. In the example **p** represents the cereal "All-Bran with Extra Fiber", whose attribute values are **x** = (0, 50, 4, 25), for sugar, calories, protein and vitamin content. The corresponding estimates obtained through orthogonal projections are $\hat{\mathbf{x}}$ = (-3.13, 67.32, 3.81, 15.21). The color bar represents caloric content. In the star coordinates plot in (b) users can manually select specific sets of axis vectors. In this case, healthier cereals are located towards the left.*

option manipulation [Hof04, Udi05, FGL13], but not for updating the orientation or lengths of the axis vectors.

Finally, given a valid choice for **V** (in accordance with (3)), PCB map high-dimensional data samples (**x**) onto their low-dimensional representations (**p**) according to the following linear equation:

$$\mathbf{p} = \mathbf{V}^{\dagger}\mathbf{x}, \tag{5}$$

where † denotes the Moore-Penrose pseudoinverse. In practice **V** is usually full rank, in which case:

$$\mathbf{p} = (\mathbf{V}^{\mathsf{T}}\mathbf{V})^{-1}\mathbf{V}^{\mathsf{T}}\mathbf{x}. \tag{6}$$

2.2. Star coordinates

The simplest radial method is star coordinates (SC) [Kan00, Kan01, RSRDS16]. The low-dimensional representation $\mathbf{p} \in \mathbb{R}^m$ of a data sample $\mathbf{x} \in \mathbb{R}^n$ is simply a linear combination of the vectors $\mathbf{v}_i$, where the linear coefficients correspond to the attribute values of **x**. Formally:

$$\mathbf{p} = x_1\mathbf{v}_1 + x_2\mathbf{v}_2 + \cdots + x_n\mathbf{v}_n = \mathbf{V}^{\mathsf{T}}\mathbf{x},$$

which can be written in matrix form as $\mathbf{P} = \mathbf{X}\mathbf{V}$.

In SC the interpretation of the vectors is straightforward: the orientation determines the direction in which a variable increases, and the length specifies the amount of contribution of a particular variable in the resulting visualization, given that all variables have a similar scaling. Traditionally, the data has been often normalized so that its range becomes $[0, 1]$ for every variable. However, centering the data allows to estimate original data values more accurately [RSS14].

Lastly, SC is an interactive method that allows users to specify any layout of the axis vectors in order to visualize the data from different perspectives. Thus, it is a general linear method that can produce any linear mapping from the data space onto the observable display. Figure 1(b) shows an example where the configuration of axis vectors allows to characterize healthy vs. unhealthy breakfast cereals.

2.3. Orthographic star coordinates and axis calibration

Orthographic star coordinates (OSC) [LT13] is a variant of SC that consists of constraining the columns of **V** so that they form an orthonormal set of vectors. This provides more faithful representations of the data since it avoids introducing distortions, and enhances preserving relative distances between samples. In addition, it helps reducing attribute estimation errors when approximating values through projections onto calibrated axes [RSS14]. A drawback of this approach concerns the possibility of choosing specific layouts for the axes. For instance, when given some initial matrix of axis vectors **V** defined by the user, the new orthogonal matrix $\mathbf{V}_{\perp}$ will contain different axis vectors that may not reflect the users' initial intentions. The transformed matrix $\mathbf{V}_{\perp}$ can be obtained, for example, by simply applying the well-known Gram-Schmidt orthogonalization procedure, and will share the same range as that of **V**. Finally, the OSC mapping is defined through:

$$\mathbf{p} = \mathbf{V}_{\perp}^{\mathsf{T}}\mathbf{x}, \tag{7}$$

where $\mathbf{V}_{\perp}^{\mathsf{T}}\mathbf{V}_{\perp} = \mathbf{I}$ is the identity matrix.

3. Adaptable radial axes plots

The following sections describe our proposed extensions to PCB that stem from modifying their optimization problem. We denote these variants as "adaptable radial axes plots".

3.1. Arbitrary layouts of axis vectors

Principal component biplots provide a set of visualizations (linear mappings) that allow to optimally approximate attribute values, distances, or correlation coefficients. However, they do not allow users

to select arbitrary directions for the axis vectors associated with the variables. In contrast, SC allows full user interaction, where users can choose arbitrary layouts of axis vectors in order to produce any linear mapping. This permits searching for clusters, outliers, or data with certain features, but at the expense of losing accuracy when estimating attribute values, distances, or correlation coefficients (note that the mapping does not arise from any optimization procedure). In this section we explore a hybrid technique that allows users to choose arbitrary configurations of axis vectors, but maps high-dimensional samples as in PCB, which optimizes attribute estimation accuracy.

Principal component biplots find optimal sets of axis vectors $\mathbf{V}$ and embedded points $\mathbf{P}$ simultaneously by solving (1). Alternatively, we consider the possibility of selecting any arbitrary layout of axis vectors, where $\mathbf{V}$ is fixed and chosen by the user. The approach is able to build a linear mapping based on $\mathbf{V}$ that enables to estimate attribute values optimally through projections onto labeled axes (similarly to PCB). The idea is based on solving the following optimization problem:

$$\underset{\mathbf{P} \in \mathbb{R}^{N \times m}}{\text{minimize}} \quad \|\mathbf{P}\mathbf{V}^{\mathsf{T}} - \mathbf{X}\|_{\mathrm{F}}^2,$$

This problem (whose solution is $\mathbf{P} = \mathbf{X}(\mathbf{V}^\dagger)^{\mathsf{T}}$) can be decomposed into simpler ones, since the solutions to the rows of $\mathbf{P}$ are independent from each other. In particular, the mapping of a data sample $\mathbf{x}$ onto a point $\mathbf{p}$ on the observable display can be obtained by solving:

$$\underset{\mathbf{p} \in \mathbb{R}^{m}}{\text{minimize}} \quad \|\mathbf{V}\mathbf{p} - \mathbf{x}\|_{2}^2, \tag{8}$$

The optimal (closed-form) solution to this problem is:

$$\mathbf{p} = \mathbf{V}^\dagger \mathbf{x}, \tag{9}$$

which is identical to (5). Therefore, this variant also defines a linear mapping through $\mathbf{V}^\dagger$, but can be considered to be a more flexible generalization since it allows arbitrary layouts of axis vectors. Lastly, in comparison with SC, observe that the method is essentially replacing $\mathbf{V}$ by $\mathbf{V}^\dagger$.

Figure 2 shows the motivation for the optimization problem graphically. Given some high-dimensional data sample $\mathbf{x}$, the method finds an optimal point $\mathbf{p}$ on the observable display in order to minimize the differences ($|\hat{x}_i - x_i|$) between the attribute values of $\mathbf{x}$ and the estimates $\hat{\mathbf{x}}$ associated with orthogonal projections of $\mathbf{p}$ onto the axes. Principal component biplots minimize the sum of squared differences, but other options are also possible (see Sec. 3.2). Lastly, similarly to SC, the orientation of an axis vector in our approach determines the direction in which attribute values for the associated variable should increase.

Figure 3 illustrates the relationship between the previously mentioned radial axes methods regarding estimation accuracy, depicting adaptable radial axes plots as a hybrid approach between PCB and SC.

The following result shows that OSC and adaptable radial axes plots provide the same estimation accuracy.

Proposition 1 Given a full-rank matrix of axis vectors $\mathbf{V}$, the sum of squared estimation errors is identical for OSC and adaptable radial axes plots.
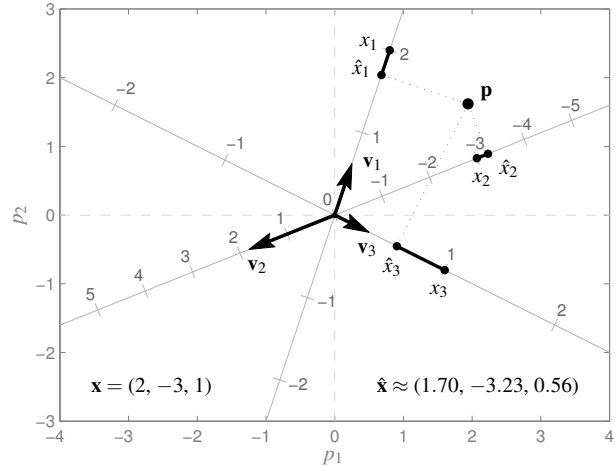

Figure 2: *Geometrical motivation for our proposed adaptable radial axes plots. Given an arbitrary set of axis vectors ($\mathbf{v}_i$), the method searches for an optimal low-dimensional representation $\mathbf{p}$ of a data sample $\mathbf{x}$ by minimizing a function of the lengths of the dark segments. These correspond to the differences between the data attributes x_i and their estimates $\hat{x}_i$ (according to the scaling of the i-th axis), which are associated with the orthogonal projections of $\mathbf{p}$ onto the axes. Lastly, note that the distance between tick marks on the i-th axis is $1/\|\mathbf{v}_i\|$.*
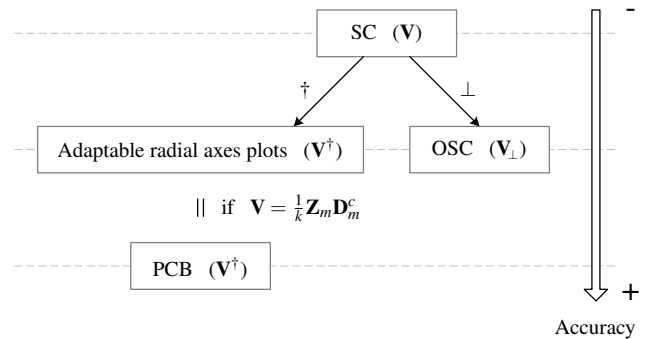

Figure 3: *Relationship between SC, OSC, adaptable radial axes plots, and PCB regarding estimation accuracy. With OSC and adaptable radial axes plots it is possible to achieve better estimates than with SC. PCB consider specific layouts of axis vectors that lead to the most accurate estimates for an entire data set. Finally, when $\mathbf{V} = 1/k\mathbf{Z}_m\mathbf{D}_m^c$ for any pair of valid values of k and c, an adaptable radial axes plot is a PCB.*

Proof In adaptable radial axes plots the estimate $\hat{\mathbf{x}}$ of some data sample $\mathbf{x}$ is:

$$\hat{\mathbf{x}} = \mathbf{V}\mathbf{p} = \mathbf{V}\mathbf{V}^\dagger\mathbf{x} = \mathbf{V}(\mathbf{V}^{\mathsf{T}}\mathbf{V})^{-1}\mathbf{V}^{\mathsf{T}}\mathbf{x},$$

due to (4), (6), and (9). Therefore, $\hat{\mathbf{x}}$ is the orthogonal projection of $\mathbf{x}$ on to the range of $\mathbf{V}$, which we denote as $\mathscr{R}(\mathbf{V})$.
In OSC the estimate $\hat{\mathbf{x}}$ can be expressed as (see (7)):

$$\hat{\mathbf{x}} = \mathbf{V}_\perp\mathbf{p} = \mathbf{V}_\perp\mathbf{V}_\perp^{\mathsf{T}}\mathbf{x} = \mathbf{V}_\perp(\mathbf{V}_\perp^{\mathsf{T}}\mathbf{V}_\perp)^{-1}\mathbf{V}_\perp^{\mathsf{T}}\mathbf{x},$$

since $\mathbf{V}_\perp^\mathsf{T}\mathbf{V}_\perp = \mathbf{I}$. Thus, it is the orthogonal projection of $\mathbf{x}$ onto $\mathscr{R}(\mathbf{V}_\perp)$.

Finally, since $\mathscr{R}(\mathbf{V}) = \mathscr{R}(\mathbf{V}_\perp)$, the estimates, and the corresponding sums of squared estimation errors, are identical for both methods. $\square$

Figure 4 illustrates the differences between SC, the proposed variant, and OSC. The radial axes plots show projections of the same standardized data used in Fig. 1, where distances between tick marks on the axes indicate standard deviation units. Firstly, recall that the configuration of axis vectors in the PCB in Fig. 1 is constrained in order to maximize attribute estimation accuracy (in that PCB the sum of squared estimation errors over every sample is 85.97). Instead, SC and adaptable radial axes plots allow users to choose alternative layouts in order to search for data with certain desired features. Fig. 4(a) shows a SC plot in which the configuration of axis vectors characterizes healthy vs. unhealthy cereals, where the healthier ones are located towards the left. Nevertheless, estimates of attribute values through projections onto labeled axes are inaccurate for SC. In particular, the sum of squared estimation errors is 358.41. Alternatively, in (b) our approach applies the mapping in (9), which provides a different set of embedded points that reduces the sum of squared estimation errors down to 171.18, without altering the initial configuration of axis vectors. Lastly, in (c) the visualization corresponds to the associated OSC plot, which achieves the same estimation accuracy. Nevertheless, OSC uses a different set of axis vectors, provided by the new orthonormalized matrix $\mathbf{V}_\perp$. Note that the lengths and orientations of the axis vectors have been modified with respect to the initial configuration specified by the user.

Lastly, an important property of adaptable radial axes plots when using the ℓ_2^2 norm (i.e., the squared Euclidean norm) is that it allows to choose layouts of axis vectors in order to generate any linear mapping. Consider some linear transformation from $\mathbb{R}^n$ to $\mathbb{R}^m$ defined by some known matrix $\mathbf{A}$, i.e., $\mathbf{p} = \mathbf{A}\mathbf{x}$. Since the linear mapping of adaptable radial axes plots is defined through (9), the axis vectors can be recovered through $\mathbf{V} = \mathbf{A}^\dagger$. Thus, by letting the axis vectors be the columns of $\mathbf{A}^\dagger$, the adaptable radial axes plot will generate the linear mapping defined by $\mathbf{A}$.

3.2. Alternative norms

The optimization problem in (8) uses the ℓ_2^2 norm, and is therefore a least-squares problem that can be solved very efficiently through the closed form solution in (9). Geometrically, the goal consists of minimizing the sum of squared lengths of the segments between the attributes x_i and their estimates $\hat{x}_i$ in Fig. 2. In this section we study the possibility of using other types of norms, which minimize the lengths of the segments in alternative ways. Formally, the more general optimization problem is:

$$\begin{array}{ll} \text{minimize} & \|\mathbf{V}\mathbf{p} - \mathbf{x}\|, \\ \mathbf{p} \in \mathbb{R}^m & \end{array} \tag{10}$$

where $\|\cdot\|$ can be any norm on $\mathbb{R}^n$. The problem is convex (see [BV04]), which allows finding global solutions reliably. In this paper we consider the ℓ_1 and ℓ_∞ norms, which lead to problems that can be cast as linear programs (LP).

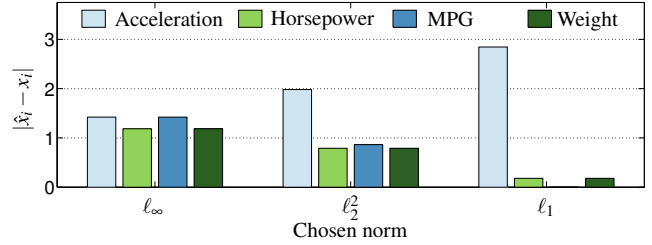The ℓ_1 norm considers minimizing the sum of the absolute values

Figure 6: *Estimation errors for sample 13 (Chevrolet Monte Carlo) of the Auto MPG data set according to the three plots in Fig. 5.*

of the differences between the attributes x_i and their estimates $\hat{x}_i$. Formally, the objective function is:

$$\|\mathbf{V}\mathbf{p} - \mathbf{x}\|_1 \;\;=\;\; \|\hat{\mathbf{x}} - \mathbf{x}\|_1 \;\;=\;\; \sum_{i=1}^{n} |\hat{x}_i - x_i|,$$

and the problem can be rewritten as the following LP:

$$\begin{array}{ll} \underset{\mathbf{t} \in \mathbb{R}^n, \, \mathbf{p} \in \mathbb{R}^m}{\text{minimize}} & \mathbf{1}^\mathsf{T}\mathbf{t} \\ \text{subject to} & -\mathbf{t} \preceq \mathbf{V}\mathbf{p} - \mathbf{x} \preceq \mathbf{t}, \end{array} \tag{11}$$

where $\mathbf{1}$ is the n-dimensional vector of all ones, and $\preceq$ denotes vector componentwise inequality.

Alternatively, the ℓ_∞ norm minimizes the maximum absolute difference between x_i and $\hat{x}_i$. The objective function is therefore:

$$\|\mathbf{V}\mathbf{p} - \mathbf{x}\|_\infty \;\;=\;\; \|\hat{\mathbf{x}} - \mathbf{x}\|_\infty \;\;=\;\; \max_{i=1..n} |\hat{x}_i - x_i|,$$

and this problem can also be expressed through a LP:

$$\begin{array}{ll} \underset{t \in \mathbb{R}, \, \mathbf{p} \in \mathbb{R}^m}{\text{minimize}} & t \\ \text{subject to} & -t\mathbf{1} \preceq \mathbf{V}\mathbf{p} - \mathbf{x} \preceq t\mathbf{1}. \end{array} \tag{12}$$

The use of different norms provides different views of the data that can be useful in exploratory tasks (e.g., when searching for data with specific characteristics). Figure 5 shows three plots of samples from the Auto MPG data set [Lic13], for a fixed configuration of axis vectors that represent four data variables. The visualizations in (a), (b), and (c), use the ℓ_2^2, ℓ_1, and ℓ_∞ norms, respectively. In this example, users interested in finding and selecting samples with large values for Acceleration, Horsepower, and Weight, but low values of MPG, would have to examine and compare several candidates located in the second (top-left) quadrant, and ideally as far as possible from the origin. However, the choice of norm can affect analysts' decisions considerably for certain samples. In this example, the car "Chevrolet Monte Carlo", marked with a dark red $\times$ symbol, is represented at different locations depending on the norm used. Observe that the sample seems like an adequate candidate when searching for data with the mentioned characteristics in (b), which applies the ℓ_1 norm. However, the sample does not stand out in (a) when using the ℓ_2^2 norm, and would clearly be discarded in (c) when employing ℓ_∞. The example therefore illustrates how alternative norms provide different views of the data that can reveal interesting samples or patterns, offering analysts a wider range of possibilities in exploration and decision support tasks.
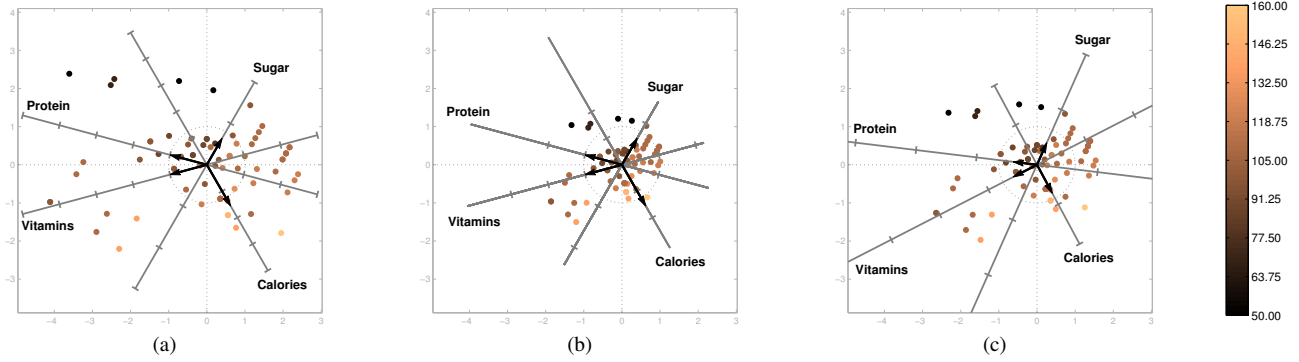
Figure 4: *Linear radial axes plots involving the (standardized) data set used in Fig. 1. In (a) the visualization corresponds to a SC plot where the axis vectors have been chosen in order to depict healthy cereals towards the left. However, estimates of attribute values through projections onto labeled axes (in this case tick marks represent standard deviation units) are highly inaccurate in SC. In particular, the sum of squared estimation errors in this plot is 358.41. Our approach applies a different mapping, through the pseudoinverse matrix $\mathbf{V}^{\dagger}$, which reduces the sum of squared estimation errors down to 171.18, maintaining the initial configuration of axis vectors, as shown in (b). Finally, (c) shows the related OSC plot, which also achieves the same enhanced estimation accuracy (171.18). However, the method needs to use a different set of axis vectors defined through a new orthonormalized matrix $\mathbf{V}_{\perp}$ (obtained through the Gram-Schmidt procedure).*
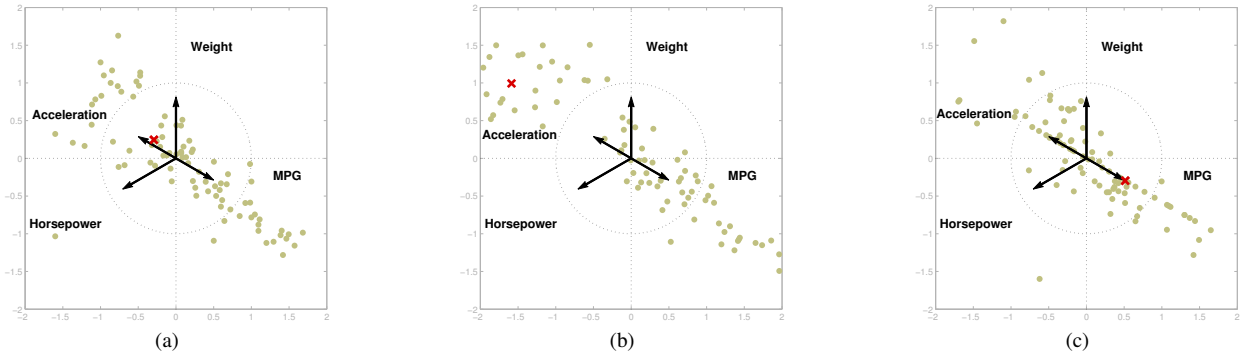

Figure 5: *Different views of a subset of the Auto MPG data set through the: (a) ℓ_2^2, (b) ℓ_1, and (c) ℓ_∞ norms. In this example, users interested in searching for samples with large values for Acceleration, Horsepower, and Weight, but low values of MPG, would focus on points in the top-left corner of the plots, but may make different decisions depending on the visualization. The location of the marked dark red × symbol (Chevrolet Monte Carlo) clearly differs in the three plots, and only seems interesting when using the ℓ_1 norm.*

Analysts can choose the type of norm depending on how they regard the severity of large estimation errors. With the ℓ_∞ norm samples are mapped in order to decrease as much as possible the maximum absolute difference between the attribute values and their estimates. The ℓ_2^2 norm also penalizes large estimation errors, but to a lesser extent than ℓ_∞. Finally, with the ℓ_1 norm some estimation errors may be large, if it helps others to be very small. Figure 6 shows the estimation errors associated with the marked sample (Chevrolet Monte Carlo) in the three plots in Fig. 5. We have chosen that sample for illustrative purposes since it can not be represented well for the particular layout of axis vectors (i.e., the estimation errors are large for the sample). For the ℓ_∞ norm the maximum (absolute) estimation error (1.42) is the lowest, but the estimates are inaccurate for every variable. The ℓ_2^2 norm hampers the estimate for Acceleration, but is able to reduce the estimation errors on the remaining variables. The ℓ_1 norm is able to achieve accurate estimates on the variables, but at the expense of an even larger estimation error on

Acceleration. Finally, note that these estimation errors could be visualized (e.g., through the size of the plotted points) in order to indicate which samples are represented well according to a particular layout of axis vectors (see [RSS14]).

Regarding efficiency, we carried out experiments in order to evaluate the average running time needed to map 1000 data samples onto a plane ($m = 2$) when using the three norms, and for $n \in [3, 21]$. The data components and coordinates of the axis vectors were drawn from a standard normal distribution. In particular, we measured the time needed to solve (9) for the ℓ_2^2 norm, and the linear programs associated with (10) for the ℓ_1 and ℓ_∞ norms. Figure 7 shows the median of the running times averaged over 20 trials, and measured on a personal computer with a fourth generation Intel® Core™ i7-4712HQ 3.3 GHz processor and 16 GB of RAM. We wrote the code in MATLAB®, using the `pinv` command to calculate pseudoinverse matrices, and the `linprog` script to solve linear programs. In particular, we chose an interior-point algorithm

when using the ℓ_1 norm, while we selected an active-set method when using the ℓ_∞ norm. The optimization problem for the ℓ_2^2 norm can be solved in the order of microseconds, due to its closed form solution. However, both LP require in the order of seconds. Thus, while the mapping can be computed in real time when using the ℓ_2^2 norm, the ℓ_1 and ℓ_∞ norms may not be suitable in interactive settings when working with large data sets.

3.3. Strategies for ordering data

Finding extrema, sorting, or determining ranges are fundamental analytic tasks [AES05]. However, to the best of our knowledge, current radial axes methods do not support them. In this section we consider several alternative convex optimization problems for ordering the embedded points according to one variable, which immediately provides information about the distribution of the data for such attribute, including extrema, range, shape, or outliers.

3.3.1. Constraints for exact estimates on one axis

In methods that depict data samples as points the values for a particular attribute can be encoded, for instance, through the size or color of the dots. Alternatively, the orthogonal projections of the points onto the axes allow to use their position, which is perceptually more accurate [Ber83, CM84, Mac86].

Firstly, consider the problem of plotting the data so that the projections of the embedded points onto one of the axes, say the *i*-th one, are exact (note that forcing the projections to also be exact for a second variable would fix the location of the points, where the resulting visualization would be equivalent to a skewed scatterplot between the two chosen variables). In that case, $\hat{x}_i = x_i$, and the plotted points would be ordered correctly along the *i*-th axis. This can be achieved by solving the following constrained convex optimization problem:

$$
\begin{aligned}
\underset{\mathbf{p} \in \mathbb{R}^m}{\text{minimize}} \quad & \|\mathbf{V}\mathbf{p} - \mathbf{x}\|, \\
\text{subject to} \quad & \mathbf{v}_i^\mathsf{T}\mathbf{p} = x_i,
\end{aligned}
\tag{13}
$$

Figure 8 illustrates the approach through an example that uses the Auto MPG data, and the same variables as in Fig. 6. Firstly, in (a) we have included an adaptable radial axes plot where the color coding corresponds to the variable MPG. Additionally, we
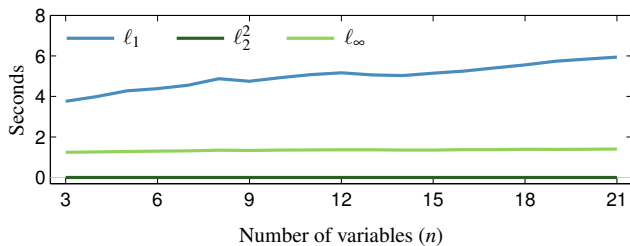
have marked the samples that contain the lowest and largest values for MPG (i.e., the extrema) with a red $+$ and $\times$ symbol, respectively. It is apparent that the data is not well ordered along the MPG axis. This can be seen by comparing the colors of neighboring points, and also since the point that exhibits the largest estimate is not an extremum. In (b) we have applied the optimization problem in (13) in order to represent the data perfectly, and therefore correctly ordered, along the MPG axis.

Lastly, we have colored the points according to the values of MPG in order to show that the method indeed orders the data correctly. However, since the estimates for MPG along its axis are exact, color can be used to encode other information, such as the quality of the estimates (see [RSS14]). Thus, the approach provides an extra degree of freedom in the visualization that allows to include additional information.

3.3.2. Constraints for correct orderings on one axis

The strict constraint that forces exact estimates for an attribute can be used to order the data correctly along the corresponding axis. However, it can increase the estimation errors on the other variables considerably. In order to alleviate this issue, in this section we consider a milder constraint that only requires the points to be ordered correctly along the axis. The new convex optimization problem for ordering the points correctly along the *k*-th axis is:

$$
\begin{aligned}
\underset{\mathbf{P} \in \mathbb{R}^{N \times m}}{\text{minimize}} \quad & \|\mathbf{P}\mathbf{V}^\mathsf{T} - \mathbf{X}\|, \\
\text{subject to} \quad & \mathbf{v}_k^\mathsf{T}\mathbf{p}^{\pi(j)} \leq \mathbf{v}_k^\mathsf{T}\mathbf{p}^{\pi(j+1)}, \ j = 1, \ldots, N-1,
\end{aligned}
\tag{14}
$$

where $\mathbf{p}^{(j)}$ denotes the *j*-th row of $\mathbf{P}$, while π represents a permutation of the first N positive integers that sorts the samples according to the values of the *k*-th attribute. In particular, $x_k^{(i)} < x_k^{(j)} \Rightarrow \pi(i) < \pi(j)$. In addition, $\|\cdot\|$ denotes entrywise matrix norms. In particular, $\|\cdot\|_1$, $\|\cdot\|_F^2$, and $\|\cdot\|_\infty$ would minimize the sum of absolute estimation errors ($\sum_{i,j} |\mathbf{v}_i^\mathsf{T}\mathbf{p}^{(j)} - x_i^{(j)}|$), the sum of squared estimation errors ($\sum_{i,j} (\mathbf{v}_i^\mathsf{T}\mathbf{p}^{(j)} - x_i^{(j)})^2$), and the maximum absolute estimation error ($\max_{i,j} |\mathbf{v}_i^\mathsf{T}\mathbf{p}^{(j)} - x_i^{(j)}|$), respectively. In this case, the solutions for the plotted points depend on each other. This implies that the problem can not be broken up into N individual subproblems. Thus, the problem's variable is the entire matrix $\mathbf{P}$.

Figure 8(c) shows an example of the approach using $\|\cdot\|_F^2$ for the Auto MPG data set and the same configuration of axis vectors as in (a) and (b). Observe that the points are ordered along the MPG axis. Although the estimates for MPG are not exact, the overall sum of squared estimation errors decreases with respect to the plot in (b). In particular, the average squared estimation error (over the 392 samples and four attributes) is 0.62, 0.67, and 0.65, for the plots in (a), (b), and (c), respectively.

3.3.3. Enlarging an axis vector

A second idea that can be used to (approximately) sort the data along one axis consists of increasing the length of the axis vector of interest. Observe that enlarging an axis vector enhances the accuracy of the estimates for the associated attribute. In particular, since the distance between consecutive integers on the axis decreases, the
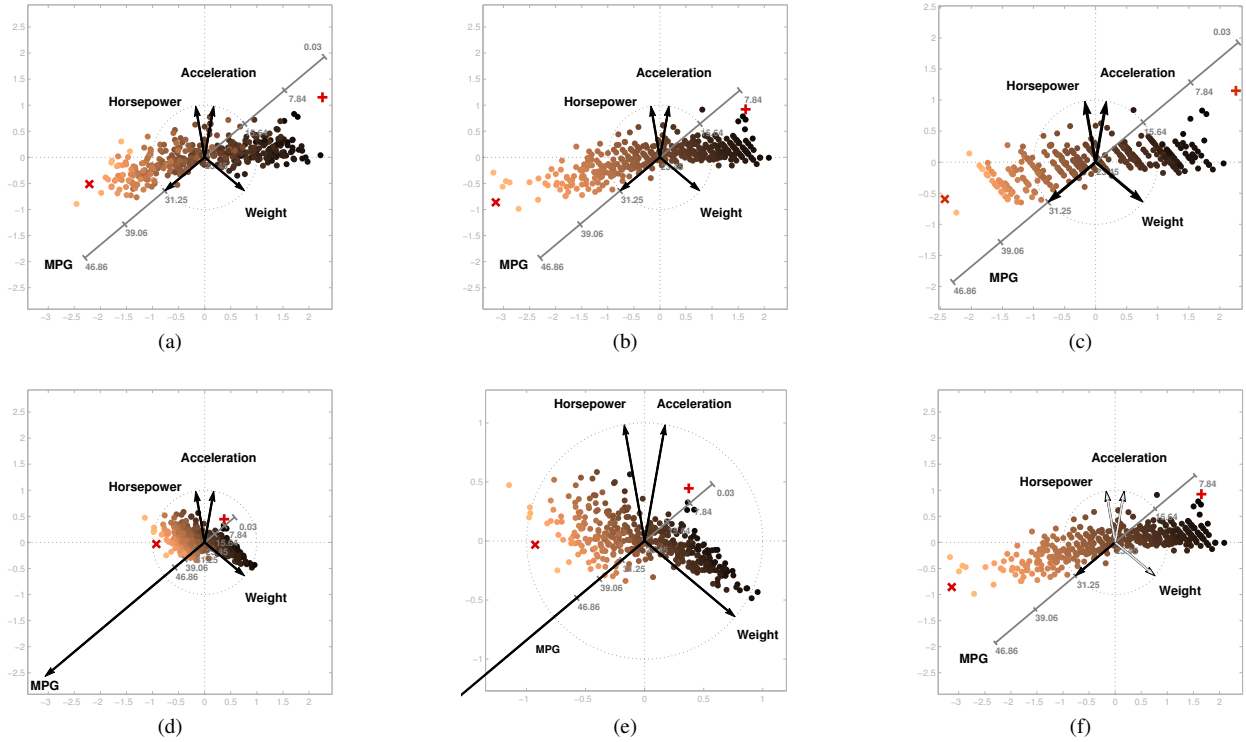

Figure 7: *Average runtime needed to solve* (10) *1000 times when using the* ℓ_1, ℓ_2^2, *and* ℓ_∞ *norms. The running times for the* ℓ_2 *norm range from 100 to 200 microseconds.*

Figure 8: *Strategies for ordering data according to one variable (axis). The example uses the four variables employed in Fig. 6. The color coding indicates values of MPG, where the samples represented by the red + and × symbols contain the lowest and largest values for MPG, respectively. In (a) the values for MPG are not ordered correctly in the adaptable radial axes plot (note that the orthogonal projection of the × symbol onto the MPG axis does not provide the largest estimate). In (b) the values of estimates for MPG are exact by applying (13). Another option consists of using (14), which orders the data correctly along one axis, as shown in (c) for $\|\cdot\|_F^2$. Although the estimates for the variable MPG are not exact, the overall accuracy considering all of the attributes is increased with respect to the plot in (b). It is also possible to sort the values by increasing the length of the corresponding axis vectors. In (d) the values for MPG can be estimated well by increasing the length (to 4) of the corresponding axis vector. However, since this strategy compresses the points in the direction of the axis, users would require zooming-in in order to visualize the plot correctly, as shown in (e). An alternative approach, which is more efficient than solving (13), consists of increasing the weight related to the variable and solving (15), as illustrated in (f). In this example, the weight for MPG is 10 times larger than the rest of the weights. Note the similarity between the plots in (b) and (f).*

same absolute distance on the plot will yield a larger estimation error. Thus, the optimization problem focuses on enhancing the accuracy of the estimates for the attribute.

The result can be seen in Figure 8(d), where increasing the length of the axis vector (in this case from 1 to 4) for MPG produces a visualization where the points are ordered better along the axis. However, this approach compresses the points in the direction of the axis, where the tick marks appear much closer to each other. Thus, this forces users to zoom-in on the plot, where the length of the MPG axis vector would no longer be visible, as shown in (e).

Moreover, the length of the axis vector may have to be extraordinarily large in order to sort the points correctly (i.e., $\hat{x}_i^{(j)} < \hat{x}_i^{(k)}$ if $x_i^{(j)} < x_i^{(k)}$ for the i-th attribute, where the superscripts indicate the j-th and k-th data sample). For example, the axis vector corresponding to MPG in the figure needs to be multiplied by 110 in order to obtain a correct ordering of the points according to such variable.

The resulting plot turns out to be useless, even if we zoom in, due to the strong compression effect. Finally, axis vectors can also be enlarged in SC in order to sort the data. However, they also need to be scaled by a large factor. For instance, when using the configuration of axis vectors in Figure 8(a), it is necessary to multiply the MPG vector times 145. In this case the plotted points are stretched along the axis, giving the impression that they lie (roughly) on a straight line.

3.3.4. Weights for controlling attribute estimation accuracy

We now introduce an additional element in the optimization problem that not only yields efficient solutions, but does not require modifying the axis vectors. The idea consists of introducing multiplicative weights in the problem's objective function as follows:

$$\underset{\mathbf{p} \in \mathbb{R}^m}{\text{minimize}} \quad \|\mathbf{W}(\mathbf{V}\mathbf{p} - \mathbf{x})\|, \tag{15}$$
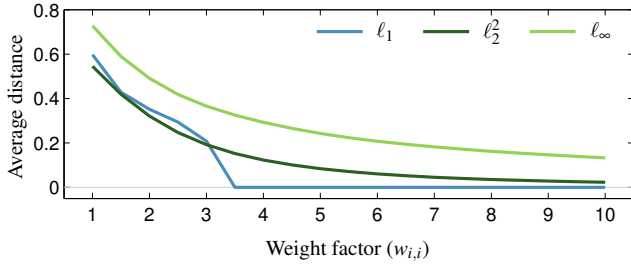
Figure 9: *Analogy between the problems in* (13) *and* (15). *The plot shows average distances between points plotted through* (13), *where the estimates are exact for the i-th variable, and the corresponding mapped points through* (15), *by scaling the weight* ($w_{i,i}$) *associated with the i-th variable, and where the rest of the weights are equal to 1. Using a larger weight for one variable is very similar to solving* (13), *especially when using* ℓ_1 *or* ℓ_2^2.

where $\mathbf{W}$ is a diagonal $n \times n$ matrix with nonnegative entries. Together with the length of an axis vector, the weight $w_{i,i}$ allows to control the relative importance of estimating attribute values correctly along the i-th axis. In particular, the algorithm that solves the optimization problem will focus on finding solutions that reduce estimation errors for variables related to larger weights. When using the ℓ_2^2 norm, (15) is a weighted least-squares problem whose solution is $\mathbf{p} = (\mathbf{WV})^\dagger \mathbf{Wx}$. Thus, it can be solved very efficiently in the order of microseconds, similarly to (9) (see Fig. 7).

Figure 8(f) shows an example where the weight for MPG is 1, but 0.1 for the remaining variables. Since the lengths of the axis vectors are all the same, the optimization problem provides a plot that represents the data very well for MPG, due to its larger weight. The corresponding plot is very similar to the one in (b), in which the estimates are exact. Figure 9 shows that by increasing the weight of a variable when using (15) it is possible to obtain a plot that will resemble the one generated by (13). The graphic shows the average distance between points plotted by both approaches, as a function of the weight for a variable, when the remaining ones are equal to 1. The results are averaged over 20 trials that use a random configuration of 6 axis vectors with components drawn from a standard normal distribution, and 100 data samples also drawn from a multivariate standard normal. It is apparent that the increasing the weight produces a plot that resembles more closely the one that provides exact estimates.

3.3.5. Efficiency comparison of the approaches

Figure 10 shows average (median) runtimes for solving (13) and (14). The experiments were run in MATLAB®. For solving the constrained least-squares problem related to (13) for the ℓ_2^2 norm we used the method `lsqlin`. For the ℓ_1 and ℓ_∞ norms we used the function `linprog`, since the problems can be cast as linear programs. Alternatively, for the problem in (14) we have used CVX, a package for specifying and solving convex programs [GB08, GB14]. The results are averaged over 20 trials where we solved both types of problems for 1000 data samples, and using the ℓ_1, ℓ_2^2, and ℓ_∞ norms. The components of the data and axis vectors were drawn from a standard normal distribution. The shorter run-

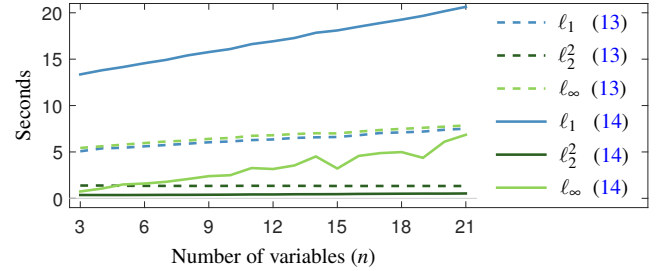**Figure 10:** *Average runtimes needed to solve* (13) *and* (14), *for* $N = 1000$ *data samples.*

times related to the use of weights in (15) are essentially identical to those shown in Fig. 7 for (10), since the cost of performing multiplications times matrix $\mathbf{W}$ is negligible. Thus, while the problems can be solved for the ℓ_2^2 norm very efficiently, using weights would be a preferable alternative when working with the ℓ_1 and ℓ_∞ norms and large data sets, since they can require running for several seconds.

4. Conclusions and discussion

This paper has introduced and analyzed in detail a multivariate visualization method based on a set of radial axes that can be viewed as a hybrid approach between PCB and SC. Adaptable radial axes plots expand principal component biplots' potential for data analysis by incorporating the possibility of selecting configurations of axis vectors freely in an interactive way, according to the analysts' needs. In addition, our approach mitigates one of the main disadvantages of SC, which is the difficulty to estimate (i.e., recover) data attributes accurately. Other important contributions include the use of alternative norms that project the data nonlinearly and allow users to explore it from additional perspectives, and new strategies based on solving convex optimization problems that can be used to sort data, find extremum values, or determine ranges of attributes.

Principal component biplots rely on Euclidean distances between data samples in order to construct the visualizations, while other types of biplots related to multidimensional scaling or correspondence analysis may use different metrics, such as the Mahalanobis or the chi-square distance [GH95, Gre10, GGLlR11]. In this regard, note that our approach does not consider distances between data samples. The norms proposed in this paper are used as a means to quantify estimation errors of individual samples in the low-dimensional plots, and therefore share no connection with metrics used in biplots. Similarly, in [GG16] the authors use a weighted Euclidean distance to construct "weighted Euclidean biplots", whose goal is to approximate distances or dissimilarities between data samples. Instead, the weights that we have introduced in Sec. 3.3.4 are used to control the accuracy of the estimates on each axis. Thus, they are used for a different purpose.

Recently, several dimensionality reduction methods have appeared in the literature where users can interactively modify visualizations by updating the coordinates of points, or the distances or neighboring relationships between them [PEP*11, MFNP13, MWT14, WTH15]. While these techniques are useful in order to

observe relationships between the elements of a data set, they do not show information related to its original attributes. In contrast, the covered radial methods not only represent data samples, but also depict data variables as customizable vectors/axes. Therefore, our approach allows users to observe, as accurately as possible, relationships between samples and data attributes. This allows performing tasks such as searching for data with specific characteristics (in directions suggested by the axis vectors), or visualizing common attribute values that characterize a cluster. In this regard, the essence and purpose of the plots related to radial axes methods are different from other dimensionality reduction techniques.

The usefulness of employing a particular norm depends on the information conveyed by the resulting visualization. Users should bear in mind that the proposed methods should be used primarily for exploratory purposes, in order to obtain an overview of the data, and possibly find unexpected patterns or other information. In this regard, using different norms provides analysts a greater array of visualizations that might reveal these interesting and unexpected structures or samples. Regarding efficiency, the method is interactive when employing the ℓ_2^2 norm (used in PCB), since the mappings can be obtained in real time. In contrast, the ℓ_1 and ℓ_∞ norms can generate interesting (nonlinear) mappings, but at the expense of requiring more computational time, which is in the order of seconds in our experiments.

We have developed a data visualization prototype in MATLAB® in order to evaluate the method's potential for data analysis. Regarding the implementation, our approach is one of many methods that display data as points, simply requiring controls to manipulate the axis vectors, select a particular norm, fix the weights for the variables, specify how estimation errors are visualized, etc., and could be included in most data visualization software packages (see [HCL05, Udi05, STH08, FGL13]).

Finally, this paper has focused on the theoretical foundations of the approach. Therefore, a broad analysis of user interaction is beyond its scope, and is left as future work. Nevertheless, we carried out preliminary usability tests with the prototype tool, where users were able to use our method in order to carry out usual exploratory tasks related to radial methods such as SC (e.g., cluster analysis, outlier and trend detection, or searching for data with particular characteristics). We must point out that adaptable radial axes plots should be used mainly for exploratory purposes, and to obtain an overview of the data, since the estimates obtained through projections onto the axes are only approximations to the true attribute values. In this regard, note that PCB use specific layouts of axis vectors that allow to represent entire data sets as faithfully as possible (in a least square sense). Thus, the proposed technique should be used as an alternative in tasks where such optimal estimation accuracy is not required, and where user interaction is essential in order to explore the data.

Acknowledgments

References

[AES05] AMAR R., EAGAN J., STASKO J.: Low-level components of analytic activity in information visualization. In *Proceedings of the Proceedings of the 2005 IEEE Symposium on Information Visualization* (Washington, DC, USA, 2005), IEEE Computer Society, pp. 15–21. 7

[Ber83] BERTIN J.: *Semiology of Graphics*. University of Wisconsin Press, Madison, WI, 1983. 7

[BV04] BOYD S., VANDENBERGHE L.: *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004. 5

[CM84] CLEVELAND W. S., MCGILL R.: Graphical perception: Theory, experimentation, and application to the development of graphical methods. *Journal of the American Statistical Association 79*, 387 (1984), 531–554. 7

[DBB10] DIEHL S., BECK F., BURCH M.: Uncovering strengths and weaknesses of radial visualizations—an empirical approach. *IEEE Transactions on Visualization and Computer Graphics 16* (November 2010), 935–942. 1

[DLR09] DRAPER G. M., LIVNAT Y., RIESENFELD R. F.: A survey of radial methods for information visualization. *IEEE Transactions on Visualization and Computer Graphics 15* (September 2009), 759–776. 1

[EY36] ECKART C., YOUNG G.: The approximation of one matrix by another of lower rank. *Psychometrika 1*, 3 (1936), 211–218. 2

[FGL13] FRUTOS E., GALINDO M. P., , LEIVA V.: An interactive biplot implementation in r for modeling genotype-by-environment interaction. *Stochastic Environmental Research and Risk Assessment 28*, 7 (2013), 1629–1641. 3, 10

[Gab71] GABRIEL K. R.: The biplot graphic display of matrices with application to principal component analysis. *Biometrika 58*, 3 (Dec 1971), 453–467. 1, 2

[GB08] GRANT M. C., BOYD S.: Graph implementations for nonsmooth convex programs. In *Recent Advances in Learning and Control*, Blondel V., Boyd S., Kimura H., (Eds.), Lecture Notes in Control and Information Sciences. Springer-Verlag Limited, 2008, pp. 95–110. 9

[GB14] GRANT M., BOYD S.: CVX: Matlab software for disciplined convex programming, version 2.1. http://cvxr.com/cvx, Mar. 2014. 9

[GG16] GREENACRE M. J., GROENEN P. J.: Weighted euclidean biplots. *Journal of Classification 33*, 3 (2016), 442–459. 9

[GGLlR11] GOWER J., GARDNER-LUBBE S., LE ROUX N.: *Understanding Biplots*. John Wiley & Sons, 2011. 1, 2, 9

[GH95] GOWER J. C., HAND D. J.: *Biplots*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability. Taylor & Francis, 1995. 1, 2, 9

[Gre10] GREENACRE M.: *Biplots in Practice*. BBVA Foundation, 2010. 1, 2, 9

[HCL05] HEER J., CARD S. K., LANDAY J. A.: prefuse: a toolkit for interactive information visualization. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (New York, NY, USA, 2005), CHI'05, ACM, pp. 421–430. 10

[Hof04] HOFMANN H.: *COMPSTAT 2004 — Proceedings in Computational Statistics: 16th Symposium Held in Prague, Czech Republic, 2004*. Physica-Verlag HD, Heidelberg, 2004, ch. Interactive Biplots for Visual Modelling, pp. 223–234. 3

[Kan00] KANDOGAN E.: Star coordinates: A multi-dimensional visualization technique with uniform treatment of dimensions. In *In Proceedings of the IEEE Information Visualization Symposium, Late Breaking Hot Topics* (2000), pp. 9–12. 1, 3

[Kan01] 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* (New York, NY, USA, 2001), KDD'01, ACM, pp. 107–116. 1, 3

[Lic13] LICHMAN M.: UCI machine learning repository, 2013. URL: archive.ics.uci.edu/ml. 5

[LKZ*15] LEHMANN D. J., KEMMLER F., ZHYHALAVA T., KIRSCHKE M., THEISEL H.: Visualnostics: Visual guidance pictograms for analyzing projections of high-dimensional data. *Computer Graphics Forum (Proc. EuroVis) 34*, 3 (June 2015), 291Ű–300. 1

[LT13] LEHMANN D. J., THEISEL H.: Orthographic star coordinates. *IEEE Transactions on Visualization and Computer Graphics 19*, 12 (December 2013), 2615–2624. 3

[LT16] LEHMANN D. J., THEISEL H.: General projective maps for multidimensional data projection. *Computer Graphics Forum* (2016). 1

[Mac86] MACKINLAY J. D.: Automating the design of graphical presentations of relational information. *ACM Transactions on Graphics 5*, 2 (1986), 110–141. 7

[MFNP13] MAMANI G. M., FATORE F. M., NONATO L. G., PAULOVICH F. V.: User-driven feature space transformation. *Computer Graphics Forum 32*, 3 (2013), 291–299. 9

[MWT14] MIZUNO K., WU H. Y., TAKAHASHI S.: Manipulating bilevel feature space for category-aware image exploration. In *2014 IEEE Pacific Visualization Symposium* (March 2014), pp. 217–224. 9

[PEP*11] PAULOVICH F. V., ELER D. M., POCO J., BOTHA C. P., MINGHIM R., NONATO L. G.: Piecewise laplacian-based projection for interactive data exploration and organization. *Computer Graphics Forum 30*, 3 (June 2011), 1091–1100. 9

[RSRDS16] RUBIO-SÁNCHEZ M., RAYA L., DÍAZ F., SANCHEZ A.: A comparative study between radviz and star coordinates. *IEEE Transactions on Visualization and Computer Graphics 22*, 1 (January 2016), 619–628. 1, 3

[RSS14] RUBIO-SÁNCHEZ M., SANCHEZ A.: Axis calibration for improving data attribute estimation in star coordinates plots. *IEEE Transactions on Visualization and Computer Graphics 20*, 12 (December 2014), 2013–2022. 1, 3, 6, 7

[STH08] STOLTE C., TANG D., HANRAHAN P.: Polaris: A system for query, analysis, and visualization of multidimensional databases. *Commun. ACM 51*, 11 (Nov. 2008), 75–84. 10

[Udi05] UDINA F.: Interactive biplot construction. *Journal of Statistical Software 13*, 5 (January 2005), 1–16. 3, 10

[WTH15] WAKITA K., TAKAMI M., HOSOBE H.: Interactive high-dimensional visualization of social graphs. In *2015 IEEE Pacific Visualization Symposium (PacificVis)* (April 2015), pp. 303–310. 9

[YMSJ05] YI J. S., MELTON R., STASKO J., JACKO J. A.: Dust & magnet: multivariate information visualization using a magnet metaphor. *Information Visualization 4*, 4 (2005), 239–256. 2