Machine Learning Journal manuscript No.
(will be inserted by the editor)

Information Retrieval Approach to Meta-visualization

Jaakko Peltonen · Ziyuan Lin

the date of receipt and acceptance should be inserted later

Abstract Visualization is crucial in the first steps of data analysis. In visual data exploration with scatter plots, no single plot is sufficient to analyze complicated high-dimensional data sets. Given numerous visualizations created with different features or methods, meta-visualization is needed to analyze the visualizations together. We solve *how to arrange numerous visualizations onto a meta-visualization display*, so that their similarities and differences can be analyzed. Visualization has recently been formalized as an information retrieval task; we extend this approach, and formalize meta-visualization as an information retrieval task whose performance can be rigorously quantified and optimized. We introduce a machine learning approach to optimize the meta-visualization, based on an information retrieval perspective: two visualizations are similar if the analyst would retrieve similar neighborhoods between data samples from either visualization. Based on the approach, we introduce a nonlinear embedding method for meta-visualization: it optimizes locations of visualizations on a display, so that visualizations giving similar information about data are close to each other. In experiments we show such meta-visualization outperforms alternatives, and yields insight into data in several case studies.

Keywords meta-visualization · neighbor embedding · nonlinear dimensionality reduction

Jaakko Peltonen
Helsinki Institute for Information Technology HIIT
Department of Information and Computer Science
Aalto University and School of Information Sciences
University of Tampere
P.O. Box 15400, FI-00076 Aalto, Finland
E-mail: jaakko.peltonen@aalto.fi

Ziyuan Lin
Helsinki Institute for Information Technology HIIT
Department of Information and Computer Science
Aalto University
P.O. Box 15400, FI-00076 Aalto, Finland
E-mail: ziyuan.lin@aalto.fi

1 Introduction

Visualization is crucial especially in the first stages of data analysis when strong hypotheses or models are not yet available for the data. We consider exploration of high-dimensional data by scatter plots. A scatter plot can show 2–3 original data features, or a mapping created by dimensionality reduction; visualization by low-dimensional scatter plots has been a traditional application of nonlinear dimensionality reduction (NLDR) methods (Roweis and Saul 2000; Belkin and Niyogi 2002; Weinberger and Saul 2006; Zhang and Zha 2004; Yan et al 2007; Zhang et al 2009; Guan et al 2011; Zhou et al 2011; see van der Maaten et al 2009 for a recent review). It is easy to see that a single low-dimensional scatter plot *cannot represent all properties of a high-dimensional data set*; even NLDR methods cannot preserve all essential data properties when the output is lower-dimensional than the effective data dimensionality (see Venna et al 2010). No single scatter plot is then enough to comprehensively explore the data; instead, *multiple visualizations* must be created and studied.

For high-dimensional data there are numerous possible ways to create visualizations. At simplest, traditional two-dimensional scatter plots could be created where each scatter plot would show two of the original features; with D features there are $(D^2 - D)/2$ such traditional scatter plots. Linear dimensionality reduction methods and NLDR methods can each yield infinitely many scatter plots by emphasizing different features in the similarity metric and by different hyperparameter values. Each plot reveals different data properties. The remaining problem is that it is hard and time-consuming to get an overview of a data set from a large *unorganized* set of scatter plots; to aid analysis, the multiple plots must be related to one another. Analyzing and displaying the similarities and relationships between visualizations can be called *meta-visualization*.

In this paper we introduce a machine learning approach for meta-visualization: we solve *how to arrange numerous scatter plots of a data set onto a display*, to show their relationships. Such a meta-visualization can reveal which plots have redundant information, and which different aspects of the data are shown in a set of plots. Our solution principle is that *visualizations showing similar information about the data should be close-by on the display*. Our approach yields a well-defined task for meta-visualization whose success can be quantitatively measured and optimized.

NLDR for visualization has recently been formalized as an information retrieval task (Venna et al 2010); the formalization has yielded an information retrieval perspective to existing NLDR methods and new well-performing methods (Venna et al 2010; Peltonen and Kaski 2011; Yang et al 2013). Our work in this paper extends this information retrieval perspective and formalizes meta-visualization of several scatter plots as an information retrieval task.

Given several scatter plot visualizations of a data set, the first step in our approach is to evaluate similarity or distance between them. We introduce an *information retrieval approach* to evaluate the similarity: two scatter plots are similar if they reveal similar neighborhoods between data samples. The similarity is quantified as an information retrieval cost of retrieving neighbors seen in one plot from the other plot. High similarity often indicates the same structure of data is visible in both plots. Given the similarities, the plots must be mapped onto the meta-visualization display. This is an NLDR task where each complex object is an individual visualization. We introduce *an NLDR approach for meta-*

visualization: locations of plots on the meta-visualization display are optimized for an information retrieval task, so that close-by plots show similar data relationships, under a non-overlappingness constraint. In experiments our approach yields informative meta-visualizations for analyzing data through different feature sets, NLDR with different hyperparameters, and numerous NLDR methods.

Meta-visualization lessens the workload of the analyst: rather than having to analyze and each plot separately in an unordered set of plots, from a well-organized meta-visualization the analyst can see which plots provide similar information, since plots physically close-by on the meta-visualization show similar data relationships, whereas plots physically far-away (such as separated clusters of plots) show different aspects of the data. The arrangement of plots thus reveals the different aspects of data as groups of plots. The analyst can then make insights about the shown similarities and differences: for example, two plots might show similar information because they are based on separate but redundant feature sets. We demonstrate this in a bioinformatics study, where a set of tissue samples are plotted based on different biological pathways in each plot. Some pathways turn out to have a similar ability to discriminate diseases in the tissue samples, that is, they yield similar plots where samples of some diseases are separated from the rest. Then the plots along the different pathways become grouped by the ability of the pathways to discriminate the different diseases in the tissue samples. A more detailed discussion is provided in Section 3.2.

To summarize, we contribute, based on an information retrieval approach, 1) an NLDR formalization of the meta-visualization task; 2) a data-driven divergence measure between scatter plots; 3) an NLDR method arranging plots on a meta-visualization display, optimized for retrieval of related plots.

This paper extends our conference paper (Peltonen and Lin 2013) by introducing two comparison approaches for meta-visualization, an empirical comparison showing how our full information retrieval approach is needed for best results, an experiment showing the benefit of emphasizing non-overlappingness for readable meta-visualization, as well as extended discussion of case studies and methodological details.

We start the paper with a review of related work in Section 2. In Section 3 we then introduce our approach: we first present the information retrieval principle for comparing plots as well as the resulting computational measure in Section 3.1, and the information retrieval principle for laying out a meta-visualization as well as the resulting meta-visualization NLDR method in Section 3.2. In Section 3.3 we discuss potential alternative approaches that we will compare our approach to in experiments. In Section 4 we perform a series of experiments including two quantitative comparisons to alternative approaches (Sections 4.1 and 4.2) and an illustration of the influence of a readability parameter (Section 4.3), as well as three case studies using meta-visualization to study hyperparameter influence on a prominent NLDR method (Section 4.4), differences among NLDR methods (Section 4.5), and exploration of a gene expression experiment collection along different gene pathways (Section 4.6). Lastly, we conclude with discussion in Section 5.

2 Background

Historically, the term "meta-visualization" has been used with several meanings; in most cases it has denoted working with several visualizations, such as manual and interactive design of coordinated multiple views with a visualization system (Robinson and Weaver 2006; Weaver 2006). The term has also been used for visualization of an algorithm workflow using plots at different levels of abstraction (Sikachev et al 2011); we do not focus on such work. We use "meta-visualization" to denote works that relate several visualizations, potentially without user's direct intervention: our usage corresponds to that of Bertini et al (2011) who described meta-visualization as "a visualization of visualizations", and more specifically as "a visualization layout strategy that organizes single visualizations into an organized form"; our proposed method is such a strategy for organizing visualizations. We concentrate on meta-visualization of scatter plots; parallel coordinate plots and recent visualizations (Wickham and Hofmann 2011) are alternatives.

The need to organize visualizations has been noted (Bertini et al 2011); common organizations are simple lists or matrices. In a *scatter plot matrix*, an element (i, j) is a plot of the ith feature vs. the jth feature; related methods include HyperSlice (see Wong and Bergeron 1997). Figure 4 (right) shows an example of a scatter plot matrix. Traditional scatter plot matrices have the limitation that the organization of plots depends only on feature indices and not on content of the plots; additionally, the scatter plot matrix cannot be easily constructed for a more general set of plots that do not arise from combinations of two feature indices.

Some methods find orderings of visualizations (Peng et al 2004). The Grand Tour (Asimov 1985) animates overviews of data projections. Rankings are used to find the most "interesting" visualizations, see Tatu et al (2009). Some NLDR methods (Cook et al 2007) arrange data onto several displays, but do not solve how to relate numerous displays.

Interactive systems like DEVise (Weaver 2006) show multiple visualizations and let users lay them out. *Overview+detail* techniques show data subsets next to an overall view in (see Cockburn et al 2009). Methods with linked views (Kehrer and Hauser 2013) highlight items in several views. Claessen and van Wijk (2011) integrate scatter plots, parallel coordinate plots, and histograms in regular arrangements. Viau and McGuffin (2012) connect multivariate charts by curves showing relations between feature tuples.

We point out that machine learning methods have been proposed to learn from multiple views of a data set, for example by canonical correlation analysis (CCA) to discover correlated linear components in the views or by other multi-view learning methods (see Xu et al 2013 for a recent survey on multi-view learning). Such methods are complementary to ours but have a different goal in that they typically aim to extract a small set of new low-dimensional components such as CCA components describing related characteristics among the original high-dimensional views, rather than analyzing the original set of views. In contrast, we aim at visual analysis of the original set of views which are already low-dimensional plots each, and we will allow analysis of similarities among plots by arranging them onto a meta-visualization.

Most works above relate a small number of visualizations. Given numerous plots, *arranging them onto the meta-visualization* becomes crucial; we solve this task. One can then e.g. add parallel coordinate plots connecting axes of nearby plots

or axes interactively chosen by the analyst; the above works thus complement our method.

Tatu et al (2012) arranged plots of subspaces by applying multidimensional scaling to Tanimoto similarities, which evaluate dimension overlap between subspaces. Such arrangements are not based on the data, only on annotation of subspace parameters. Such layouts cannot be computed when plots arise from more complicated NLDR. Tatu et al. also used a similarity based on the percentage of agreement within k-NN lists, but not for laying out plots, only for grouping them. Unlike Tatu et al (2012), the approach we propose creates data-driven layouts of plots. Binary neighborhoods such as k-NN lists (where each point is or is not a neighbor) only change if a point enters or leaves the neighborhood, that is, if the set of neighbors changes; thus such binary neighborhoods do not reflect more nuanced changes, such as changes in the order of the neighbors within the neighborhood (which neighbor is nearest to the central point), changes in distances of neighbors from the central point, or changes in the order or distances of the non-neighbors outside the neighborhood. Our approach is based on probabilistic neighborhoods where the continuous-valued probabilities of neighbors can take into account such nuances.

For the task of constructing a single scatter plot visualization, a common approach is to apply a NLDR method to reduce data to a two-dimensional representation and plot the result. Numerous NLDR methods have been proposed. Many NLDR methods are designed for *manifold learning*, that is, the methods aim to find an underlying lower-dimensional manifold of the data embedded in the high-dimensional space and then unfold the manifold. Many successful manifold learning methods exist including Isomap Tenenbaum et al (2000), Locally Linear Embedding (LLE; Roweis and Saul 2000), Laplacian Eigenmap (LE; Belkin and Niyogi 2002), Maximum Variance Unfolding (MVU; Weinberger and Saul 2006) and several others. Several recent NLDR approaches have been based on the concept of *neighbor embedding* , including Stochastic Neighbor Embedding (SNE; Hinton and Roweis 2003), t-distributed SNE (t-SNE; van der Maaten and Hinton 2008) and others. See, for example Venna and Kaski (2007), van der Maaten et al (2009), and Wismüller et al (2010) for extensive reviews and comparisons of nonlinear dimensionality reduction approaches. Some dimensionality reduction methods aim to find a sparse linear mapping, in order to make computation of low-dimensional representations efficient and easier to interpret, and to potentially reduce overfitting in further predictive tasks; for example a manifold elastic net (Zhou et al 2011) can be used for this purpose. Some recent works have aimed to unify dimensionality reduction algorithms, for example several spectral analysis based dimensionality reduction methods have been unified in a patch alignment based framework Zhang et al (2009); Guan et al (2011).

Several manifold learning approaches have had difficulties in low-dimensional information visualization (Venna and Kaski 2007), as they have been designed to find and unfold a manifold but not to compress the data below the intrinsic dimensionality of the manifold. In a low-dimensional visualization, all original data properties cannot be represented perfectly on the output display, and being able to define and quantify the goodness of the representation is crucial. Venna et al (2010) proposed a recent well-performing NLDR approach for visualization, where visualization by scatter plots is formalized as an information retrieval task: original neighbors of data points are retrieved from the display, and the visualization

is optimized to minimize retrieval errors, which can be quantified by information retrieval measures precision and recall. The approach has yielded state of the art performance in visualization (Venna et al 2010). The approach was proposed only for creating a single plot of data; analyzing several plots was not considered. In this paper, we take an information retrieval perspective to formalize the meta-visualization task of organizing a set of several scatter plots. Our formalization also involves information retrieval concepts such as retrieval errors and the precision and recall goodness measures, but unlike Venna et al (2010) we bring the information retrieval perspective and concepts to solve the needs of the new meta-visualization setting, in particular for quantifying differences between plots and for quantifying the goodness of a meta-visualization display containing an arrangement of several plots.

Note that NLDR is often applied in other data transformation tasks than visualization. Using the lower-dimensional NLDR output data can reduce computational complexity of further processing and reduce memory and disk space needed for data storage. The lower-dimensional representation may also be beneficial in predictive tasks; for example, Chang et al (2004) used LLE as part of an image super-resolution task, Patwari and III (2004) used several manifold learning algorithms including Isomap, LLE, and Hessian LLE (HLLE; Donoho and Grimes 2003) for sensor localization in wireless sensor networks, Nguyen and Worring (2008) integrated SNE into the visualization stage of a content-based image retrieval (CBIR) engine, and van der Maaten (2009) proposed a fine-tuning method based on t-SNE for a stack Restricted Boltzmann Machine. In this paper we focus on the task of information visualization, in particular on meta-visualization.

The method we propose in this paper is the first neighbor embedding method organizing plots onto a meta-visualization.

3 The Method: Information Retrieval Approach to Meta-Visualization

We optimize meta-visualizations for analysts studying data through neighborhood relationships. From each scatter plot, the analyst visually retrieves neighborhood relationships of samples. Given many plots the analyst retrieves which plots show similar neighborhoods as a plot she is interested in, vs. which ones show different information.

Let $\{\mathbf{x}_i\}_{i=1}^N$ be a set of input data samples. Let there be M different low-dimensional scatter plots of the data set; in the mth plot the samples have positions $\{\mathbf{y}_{m,i}\}_{i=1}^N$ on the plot. The different plots might arise from different features or similarity metrics for the data, different NLDR methods, or different parameters within an NLDR method. Since a low-dimensional plot cannot represent all features of the high-dimensional data, each plot will show different data aspects; in particular, each plot will show different neighborhood relationships between data. In the mth plot, let each data point i have a probabilistic *output neighborhood*, defined as a distribution $q_m^i = \{q_m(j|i)\}$ over the possible neighbors $j \neq i$, where $q_m(j|i)$ is the probability that an analyst starting from point i on the display would retrieve point j as an interesting neighbor for further study.

The output neighborhood. The $q_m(j|i)$ should depend on positions of data on the mth plot, so that samples j close to i are more likely to be retrieved as

neighbors. We set

$$q_m(j|i) = \exp(-\|\mathbf{y}_{m,i} - \mathbf{y}_{m,j}\|^2/\sigma_{m,i}^2) \cdot \left(\sum_{k \neq i} \exp(-\|\mathbf{y}_{m,i} - \mathbf{y}_{m,k}\|^2/\sigma_{m,i}^2) \right)^{-1} \quad (1)$$

where $\sigma_{m,i}^2$ controls how quickly $q_m(j|i)$ falls off with distance. If more accurate user models are available, e.g. estimated from eye tracking, they can be plugged in place of Eq. (1). We set $\sigma_{m,i}$ to half of the maximum pairwise distance between points in m. Alternatively $\sigma_{m,i}$ can be set to a value according to the "perplexity" of point i in visualization m (Hinton and Roweis 2003; Venna et al 2010). But the first simple choice already worked well in experiments.

3.1 Information Retrieval View of Comparing Neighborhoods between Plots

In visual information retrieval an analyst looking at a scatter plot retrieves neighbors for each data point. When several plots are available for the data, the analyst can *compare the neighborhoods* between plots. If two plots show similar neighborhoods, findings from them support each other; if they show different neighborhoods, they reveal different data aspects.

Suppose the analyst studied plot m, and now studies plot m'. As the plots have different data arrangements, when the analyst tries to retrieve the neighborhoods visible in m from m', *two kinds of differences* arise. For each query point i, some points j that used to be neighbors of i in plot m (having high probability $q_m(j|i)$) no longer look like neighbors in plot m' (low $q_{m'}(j|i)$); they are *missed* when neighbors are retrieved from m'. Conversely, some points j that were not neighbors of i in plot m (low $q_m(j|i)$) look like neighbors in plot m' (high $q_{m'}(j|i)$); they are *novel neighbors* when neighbors are retrieved from m'. Figure 1 illustrates the setup. The concept is symmetric: if plot m' misses a neighbor that was visible in plot m, equivalently m yields the neighbor as a novel neighbor compared to m'.

Cost of differences. In information retrieval literature, if an analyst is trying to retrieve a set of items (here the set of neighbors previously seen in plot m) and instead retrieves another set of items (here the set of neighbors seen in plot m'), the differences between the sets are called "retrieval errors". Since we formulate the comparison of plots as information retrieval, we will temporarily use the term "retrieval errors" to denote the differences between plots, but we stress that in our setting the "errors" are actually natural differences between plots of data arising for example from different feature sets used to create the plots, and the analyst will ultimately want to analyze such differences using a well-organized meta-visualization.

If the analyst has found interesting relationships from plot m but fails to find them in m', each difference (each missed neighbor or novel neighbor) can have a cost to the analyst; for example such cost could arise in terms of time and attention spent on locating the corresponding neighbors in both plots. The difference measure between plots arising from the information retrieval task is the *total cost of information retrieval errors* when retrieving the neighbor relationships in m from m'. The total cost can be shown to be a sum of Kullback-Leibler divergences D_{KL}

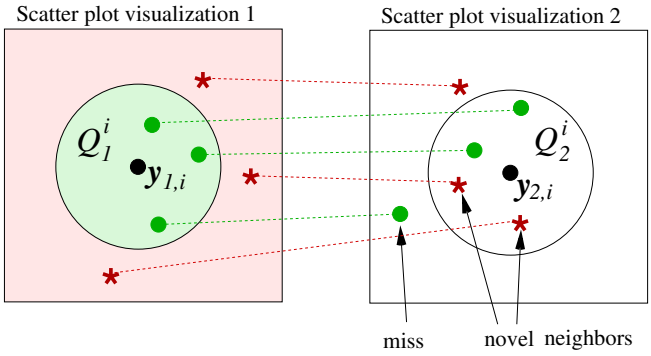

Fig. 1: Differences between plots in visual information retrieval. We consider a query point i, and try to retrieve the neighbors seen in one scatter plot (left) from a second plot (right). Q_1^i denotes points with high neighborhood probability $q_1(j|i)$ in the first plot, Q_2^i denotes points with high $q_2(j|i)$ in the second plot. *Missed neighbors* have high $q_1(j|i)$ but low $q_2(j|i)$; an analyst looking at the second plot would miss them. *Novel neighbors* have low $q_1(j|i)$ but high $q_2(j|i)$; they were not apparent in the first plot. Note that this figure is similar to Fig. 1 of Venna et al (2010) as both figures represent retrieval situations, but their setting is different. The figure above illustrates a meta-visualization setting where the left-hand side and right-hand side are two low-dimensional scatter plots of the same data set; the figure represents differences that arise when the analyst compares neighborhood relationships between the plots. In contrast, the figure of Venna et al (2010) represents retrieval of neighborhoods from a single plot compared to high-dimensional ground truth neighborhoods of data.

between neighborhood distributions. [1] In detail, if q_m^i and $q_{m'}^i$ are "nearly discrete" so $q_m(j|i)$ is uniformly high for a small number of neighbors j and very small for other points, and similarly for m', then $D_{KL}(q_m^i, q_{m'}^i) \approx Const \cdot (N_{m,m'}^{MISS,i}/r_m^i)$ where r_m^i is the total number of neighbors of i in m and $N_{m,m'}^{MISS,i}$ is the number of those neighbors missed when retrieving the neighbors from visualization m'. We thus use D_{KL} to measure the cost of misses around query point i between plots m and m'. The total amount of misses between two plots is

$$D_{m,m'} = \sum_i D_{KL}(q_m^i, q_{m'}^i) = \sum_{i,j \neq i} q_m(j|i) \log \frac{q_m(j|i)}{q_{m'}(j|i)} \,. \tag{2}$$

Similarly, it can be shown[2] the total cost of novel neighbors for each query point i is equivalent to $D_{KL}(q_{m'}^i, q_m^i)$, we could use $\sum_i D_{KL}(q_{m'}^i, q_m^i)$ to measure the cost of novel neighbors between m and m'. However, the only difference between this and Eq. (2) is that roles of m and m' have been swapped, thus the cost of novel neighbors comparing m' to m is the same as the cost of misses comparing m to

[1] As in an earlier paper Venna et al (2010) but in a meta-visualization retrieval setting. Although the steps are similar, Venna et al (2010) is about traditional NLDR and not applicable in meta-visualization.

[2] Again similarly to Venna et al (2010), but in our meta-visualization setting.

m'. Costs of novel neighbors are thus are already included in the $M \times M$ matrix of pairwise miss costs between plots.

Discussion of the divergence measure. Eq. (2) measures how the different plots contribute differences in an information retrieval task of the analyst, that is, how different the neighborhoods retrieved from each plot are. This has useful properties: 1) The measure is data-driven and applies between any scatter plots of the data set, whether they arose from pairs of data features or from NLDR. Moreover, Eq. (2) only needs the plots, the original data $\{\mathbf{x}_i\}$ are not needed. 2) It can be seen from Eq. (1) that neighborhood probabilities are invariant to translation, rotation, and mirroring of plots, thus also Eq. (2) is invariant to them. 3) The measure considers all local information, not only a global shape of data; this is important especially when individual samples are meaningful to the analyst. In Section 4.5 we see cases where the overall shape of plots can be deceptively similar but neighborhoods are very different, our measure and meta-visualization reveals this.

3.2 Mapping the Visualizations onto the Meta-Visualization

Given M plots of a data set, we use Eq. (2) between each pair of plots m and m', to compute a matrix of divergences $D_{m,m'}$. The matrix could be used to order plots: at simplest, pick a plot m of interest then place other plots m' on a line in order of the $D_{m,m'}$; such ordering is based on one row of the matrix. We go further and create meta-visualizations based on the whole matrix. The matrix encodes desired properties of a meta-visualization: plots with small divergence are similar and should be close-by, and plots with large divergence large should be far-off. It remains to lay out the plots onto the meta-visualization based on the divergences; we introduce a meta-visualization NLDR method for this task.

Information retrieval approach for meta-visualization. Given a scatter plot of interest, the analyst may wish to find other plots for inspection containing similar neighborhoods. On a meta-visualization such plots should be nearby, so the analyst does not have to scan the entire meta-visualization to find similar plots. We formalize this as an *information retrieval task on the meta-visualization*; we then and optimize the ability of the meta-visualization to serve the information retrieval. The divergence in Eq. (2) measures how similar information two plots give to the analyst; we use it to define a *true neighborhood* for each plot m. The true neighborhood is defined as a neighborhood distribution $u_m = \{u(m'|m)\}$, which tells the probability that after the analyst has inspected plot m, the neighboring plot m' would be chosen for inspection next:

$$u(m'|m) = \exp(-D_{m,m'}/2\sigma_m^2) \cdot \left(\sum_{\tilde{m} \neq m} \exp(-D_{m,\tilde{m}}/2\sigma_m^2) \right)^{-1} \tag{3}$$

where σ_m^2 controls the falloff rate of the probability and is set as in (Hinton and Roweis 2003; Venna et al 2010) to have a desired effective amount of neighbors k around each plot, where k is a rough upper bound for the number of relevant neighbors set by the user.[3] We next define *neighborhoods on the meta-visualization*

[3] In detail, the entropy of the neighborhood distribution u_m around plot m is smallest when σ_m approaches zero and hence only the nearest other plot has high neighborhood probability;

display, based on the on-screen locations of plots. Let each plot m have a location $\mathbf{z}_m$ on the meta-visualization display, e.g. as a small "mini-plot" drawn inside the meta-visualization. We define physical neighborhood distributions $v_m = \{v(m'|m)\}$ for plots by their locations on the meta-visualization:

$$v(m'|m) = \exp(-||\mathbf{z}_m - \mathbf{z}_{m'}||^2/2\sigma_m^2) \cdot \Big(\sum_{\tilde{m} \neq m} \exp(-||\mathbf{z}_m - \mathbf{z}_{\tilde{m}}||^2/2\sigma_m^2) \Big)^{-1} \quad (4)$$

where $||\mathbf{z}_m - \mathbf{z}_{m'}||$ is the Euclidean distance between the plot locations. The physical neighborhood probabilities $v(m'|m)$ represent which nearby plot m' the analyst is likely to look at next after looking at plot m on the meta-visualization, based on physical locations of the plots. In other words, the physical neighborhood probabilities represent which other plots m' the analyst *retrieves* from the meta-visualization as neighbors of plot m based on their physical locations: the retrieval is done stochastically, so that the analyst retrieves for each plot m a neighboring plot m', and the closer the plot m' is to the central plot m, the higher the probability $v(m'|m)$ is that plot m' will be retrieved as a neighbor. The $u_m = \{u(m'|m)\}$ and $v_m = \{v(m'|m)\}$ are neighborhoods between entire plots in a meta-visualization, instead of neighborhoods of data within one plot like Eq. (1); we call u_m and v_m *meta-level neighborhoods*.

Information retrieval cost in retrieval of plots from the meta-visualization. Suppose the analyst studied plot m and wants to retrieve similar plots from the meta-visualization. If plots are not well arranged on the meta-visualization, retrieval may yield two kinds of errors: *missed neighbor plots* (which could also be called *false negative plots*) and *false neighbor plots* (which could also be called *false positive plots*). The difference between these two kinds of errors is that missed neighbor plots (false negative plots) are plots that are similar to plot m according to the comparison measure of Section 3.1 but are not physically close-by to m on the meta-visualization display, whereas false neighbor plots (false positive plots) are plots that are close-by to m on the meta-visualization display but are not similar to m according to the comparison measure. The setup is illustrated in Figure 2; the setup is similar to Figure 1, but instead of comparing data points retrieved from two plots, we retrieve entire plots from the meta-visualization and compare them to true neighborhoods of plots.

Suppose each missed plot (false negative plot) or false neighbor plot (false positive plot) has a cost to the analyst; a good meta-visualization should minimize the *total meta-visualization information retrieval cost*: the smaller the cost, the less errors there are, and the better the meta-visualization shows the relationships between plots. In Appendix A we show that the total cost of errors can be represented using the information retrieval measures precision and recall, and further show that in the case of probabilistic neighborhoods the cost can be generalized as a sum of two types of Kullback-Leibler divergences:

$$E(\{\mathbf{z}_m\}) = \lambda \sum_m D_{KL}(u_m, v_m) + (1-\lambda) \sum_m D_{KL}(v_m, u_m) \; . \quad (5)$$

correspondingly the entropy is largest when σ_m approaches infinity and hence the neighborhood probability is spread uniformly over all other plots m'. The entropy of u_m can thus be controlled simply by increasing σ_m from a near-zero initial value until the entropy is at the desired level. If the neighborhood probability would be uniformly spread over k plots, the entropy of the distribution would be $\log k$. Therefore, to reach an effective number of neighbors k around each visualization m, we adjust σ_m until the entropy of u_m is $\log k$.

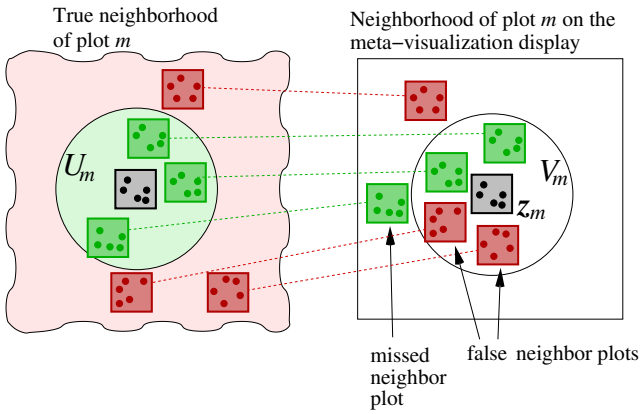

Fig. 2: Errors in visual information retrieval from a meta-visualization, when neighbor plots of a query plot m are retrieved from the meta-visualization display. In this illustration the scatter plots are shown as colored squares where each plot shows a slightly different arrangement of data (shown as dots). The **true neighborhood** of the query plot is represented on the left; the query plot (gray square) is at the center and its true neighbor plots (green squares) are the other plots m' that contain similar information about data and thus achieve high neighborhood probability $u(m'|m)$. Here U_m denotes the plots having high true neighborhood probability. The other plots that have low neighborhood probability $u(m'|m)$ are shown as red squares. The true neighborhoods of the plots arise from a comparison measure between plots as discussed in Section 3.1; to represent the high-dimensional nature of these neighborhoods as well as the fact that they are the "ground truth" neighborhoods, the borders of this subfigure are shown as wavy lines. The **meta-visualization display** is represented on the right, where $\mathbf{z}_m$ is the location of the query plot on the display; the plots close to m on the display achieve high neighborhood probability $v(m'|m)$, and V_m denotes the plots having high neighborhood probability on the display. The neighborhoods of the plots are physical neighborhoods that arise from the physical locations of plots on the meta-visualization display; to represent the low-dimensional physical nature of these neighborhoods, the borders of this subfigure are shown as straight lines. *Missed neighbor plots* (false negative plots) have high $u(m'|m)$ but low $v(m'|m)$; an analyst looking at the meta-visualization display would miss them. *False neighbor plots* (false positive plots) have low $u(m'|m)$ but high $v(m'|m)$; they could be falsely picked as neighbors of m based on the meta-visualization display. The performance of meta-visualization can be quantified as the total cost of both kinds of errors.

Here $D_{KL}(u_m, v_m)$ is a generalization of the total cost of missed neighbor plots from plot m (false negative plots; that is, plots that are similar to m according to the comparison measure of Section 3.1 but are physically far-off on the meta-visualization). Similarly, $D_{KL}(v_m, u_m)$ is a generalization of the total cost of false neighbor plots retrieved for plot m (false positive plots; that is, plots that are not similar to m but are physically close-by). Note that the cost is a function of the locations $\mathbf{z}_m$ of plots m on the meta-visualization since the distributions v_m are

defined based on the $\mathbf{z}_m$ according to Eq. (4). To optimize the meta-visualization, one can minimize the cost in Eq. (5) with respect to the locations $\mathbf{z}_m$ of the plots.

Trade-off between costs of misses and false neighbors. In Eq. (5) the sum $\sum_m D_{KL}(u_m, v_m)$ is the total cost of missed neighbor plots (false negative plots) from all plots. Optimizing the meta-visualization to minimize this sum term would try to keep the neighbors of each plot physically close to the plot, to minimize the cost of misses and thus to maximize the recall of retrieving the similar plots from the meta-visualization. Similarly, $D_{KL}(v_m, u_m)$ is the total cost of false neighbor plots retrieved for plot m (false positive plots; that is, plots that are dissimilar but are physically close-by), and the sum $\sum_m D_{KL}(v_m, u_m)$ is the total cost of false neighbor plots (false positive plots) for all plots. Optimizing the meta-visualization to minimize this sum term would try to keep non-neighbors of each plot physically away from the plot, to minimize the cost of false neighbors and thus to maximize the recall of retrieving the similar plots from the meta-visualization. There is then a trade-off between minimizing the cost of misses versus the cost of false neighbors; the optimal meta-visualization for minimizing misses versus false neighbors can differ as illustrated in Figure 3. In Eq. (5) λ controls the trade-off between costs of missed plots (false negative plots) and false neighbor plots (false positive plots) desired by the analyst: all λ give good visualization, large λ avoids misses and small λ avoids false neighbor plots, we use $\lambda = 0.5$ to emphasize both kinds of errors equally.

Since high-dimensional neighborhood relationships between plots typically cannot be perfectly preserved on a two-dimensional meta-visualization display, formulating the objective function in terms of minimizing the total cost of errors provides a rigorous quantitative objective for meta-visualization.

Repulsion to avoid overlap of plots on the meta-visualization display. Minimizing the cost Eq. (5) makes the meta-visualization *informative* in the sense that physically neighboring plots yield similar neighborhood information of data samples. However, the meta-visualization must also be *readable* by the analyst. We address one simple aspect of readability: if plots are placed physically too close-by they will overlap, making it hard to see the data in individual plots. To preserve readability of the meta-visualization, we add a repulsion term to the cost, which gives an additional cost for any pair of plots closer on the meta-visualization than a desired distance threshold. Optimization then tends to keep plots further apart than this threshold, and plots do not overlap when drawn with a size smaller than the threshold. Minimizing the final cost then optimizes *information retrieval performance of the meta-visualization, under a readability constraint of non-overlappingness*; to optimize the meta-visualization we minimize the final cost with respect to the locations $\mathbf{z}_m$ of plots m on the meta-visualization. The final cost is

$$E(\{\mathbf{z}_m\}) = \lambda \sum_m D_{KL}(u_m, v_m) + (1-\lambda) \sum_m D_{KL}(v_m, u_m) + \mu \sum_{m \neq m'} g(\mathbf{z}_m, \mathbf{z}_{m'}) \quad (6)$$

where the last sum term is the repulsion term, μ controls importance of repulsion, and g is a simple shrinkage Gaussian function: $g(\mathbf{z}_m, \mathbf{z}_{m'}) = \frac{\exp(-\|\mathbf{z}_m - \mathbf{z}_{m'}\|^2/\sigma_r^2) - t}{1-t}$ if $\|\mathbf{z}_m - \mathbf{z}_{m'}\|^2 < T$ and zero otherwise. Here $t = 0.95$ and $\sigma_r^2 = -T/\log(t)$ where T is the desired threshold; the value of t is chosen empirically since it performs well; each repulsion term yields zero cost if plots are further apart than T and cost one if plots fully overlap. The threshold T is set by the analyst according to how

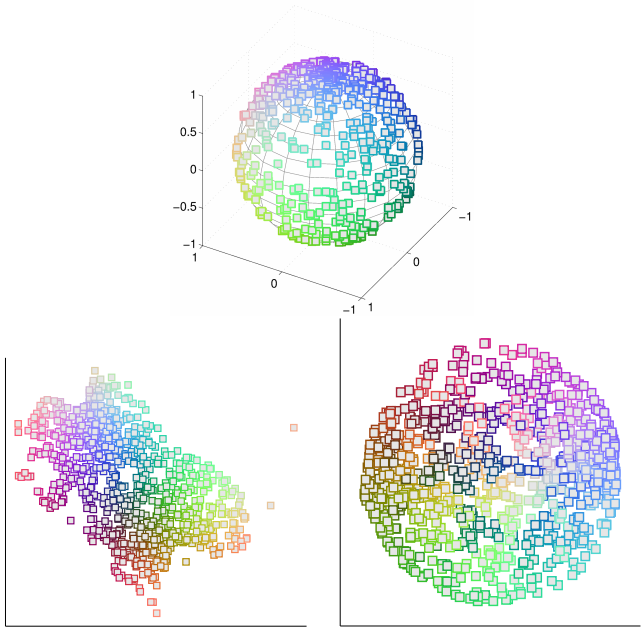

Fig. 3: Illustration of minimizing missed neighbor plots (false negative plots) versus false neighbor plots (false positive plots) in meta-visualization. Suppose we have a set of scatter plots of a data set, such that for this data the true neighborhoods of plots have a simple form: they can be represented by drawing the plots onto the surface of a three-dimensional sphere. **Top:** The original neighborhoods of the plots. Here plots are represented by colored squares whose colors correspond to their locations on the three-dimensional sphere. Since the neighborhoods of the plots form a three-dimensional sphere, they cannot be perfectly represented on a two-dimensional meta-visualization display, but the display can be optimized to minimize the resulting errors. **Bottom left:** A two-dimensional meta-visualization optimized to minimize false neighbor plots (false positive plots) on the display. The optimized meta-visualization 'cuts open' the sphere, so that for each plot all physically nearby plots on the display are true neighbors, but some original neighbors are missed across the cuts. **Bottom right:** A two-dimensional meta-visualization optimized to minimize missed plots (false negative plots). The resulting meta-visualization 'flattens' the sphere so that there are no cuts and all true neighbors remain physically close-by, but some false neighbor plots are introduced from the opposite side of the sphere. Both kinds of meta-visualizations are useful for different needs of the analyst, depending on whether avoiding missed neighbor plots (false negative plots) or false neighbor plots (false positive plots) is more crucial. In general, meta-visualization can be optimized for any trade-off between minimizing missed neighbor plots and false neighbor plots; we focus on both kinds of errors equally. (Note that to concentrate on illustrating the trade-off, repulsion between plots is not used in this illustration.)

large plots are needed on the display. We use simple data-driven choices: after an initial optimization we set T to an average (squared) distance to nearest plots, and μ to make the repulsion term have the same overall weight (times a constant) as the information retrieval terms. To help find good local minima, we increase μ iteratively during optimization from zero to the final value.

As shown in Appendix A, minimizing Eq. (5) corresponds to minimizing the total cost of information retrieval errors; therefore, minimizing Eq. (6) corresponds to minimizing the total cost of information retrieval errors plus a penalty term (repulsion term) for overlapping visualizations. We will demonstrate the effect of the repulsion term in Section 4.3.

Optimization of the meta-visualization. The cost Eq. (6) is our final measure of meta-visualization quality, in terms of performance in the information retrieval task and readability; the smaller the cost, the better the meta-visualization is. To optimize the meta-visualization we directly minimize the cost with respect to the coordinates $\mathbf{z}_m$. Note that the final cost in Eq. (6) is a continuous function of the plot locations $\mathbf{z}_m$, since the neighborhood distributions v_m on the meta-visualization are smooth functions of the $\mathbf{z}_m$ as defined in Eq. (4), and the repulsion term in Eq. (6) is also a continuous function defined based on the $\mathbf{z}_m$. To optimize the meta-visualization, we minimize Eq. (6) with respect to all the $\mathbf{z}_m$ by conjugate gradient descent. The optimization yields a meta-visualization optimized for information retrieval: physical neighborhoods of plots on the meta-visualization are optimized under the readability constraint for minimal retrieval errors compared to true neighborhoods of the plots, which in turn are defined based on neighborhoods of data in the plots. Thus the *entire process of meta-visualization*, from comparing the individual plots to placing them on the meta-visualization, *is based on an information retrieval formulation.*

Theoretical connections. Preservation of neighborhood information has been used as a cost function for NLDR of data points onto a single scatter plot by neighbor embedding (NE; see, e.g., Hinton and Roweis (2003); Venna et al (2010)). Such NE methods are unsuitable for meta-visualization as they do not trivially have available a measure to compare visualizations; moreover, they are designed to embed simple data points as dots onto a scatter plot and do not consider overlap of larger objects. Our comparison measure $D_{m,m'}$ is similar to a stochastic neighbor embedding (SNE) cost function (Hinton and Roweis 2003), but SNE and other NE methods only used such costs to compare a visualization to a high-dimensional ground truth, whereas we have turned it into a pairwise difference measure where no single visualization is a "ground truth". Our approach takes advantage of theory, bounds and optimization tools inherited from NE, but brings it into the domain of meta-visualization, with three novelties: 1) the meta-visualization setting, 2) an information retrieval based distance measure between visualizations, and 3) an NLDR method that optimizes both information retrieval performance and readability of the meta-visualization.

A precursor of readability was used in a limited setting by Vesanto (1999) to arrange component planes of a Self-Organizing Map, by a glyph placement method where overlapping component planes were moved to next-best-matching units. This could be seen as a precursor of our cost which preserves readability (non-overlappingness) as part of optimization. Glyph positioning approaches are not typical in meta-visualization of two-dimensional scatter plots. The method of

Vesanto (1999) uses global correlation of one-dimensional component planes and does not apply to two-dimensional plots.

Using and interpreting the meta-visualization. Plots physically close-by on the meta-visualization (for example, a tight cluster of plots) have similar data neighborhoods. Plots physically far away from each other (for example, separated clusters of plots) show different neighborhood information about the data, i.e., different aspects of the data. The arrangement of plots reveals the different aspects of data as groups of plots, and relationships between data aspects by closeness of groups and by plots in-between groups.

Meta-visualization lessens the workload of the analyst compared to analyzing an unordered set of plots: instead of analyzing each plot separately, the analyst can see which plots provide similar information, and can notice different aspects of the data shown by the plots. Insights about shown similarities and differences can be made: for example, two plots might show similar information because they are based on separate but redundant feature sets. Section 4 shows benefits of meta-visualization in different analysis scenarios. As an example, one of the scenarios is a bioinformatics case study where the data points are gene expression experiments of different healthy-vs-disease comparisons, several scatter plot visualizations are created by plotting the data along different subsets of active gene pathways, and meta-visualization is used to study the plots. The arrangement of plots on the meta-visualization then reveals how the ability to discriminate the different diseases varies between the plots: plots that are close-by on the meta-visualization have a similar ability to discriminate the diseases. Several groups of plots with similar discriminative ability are found, and the biological properties of the active pathways in each group can then be analyzed.

Computational aspects. Our meta-visualization arranges multiple scatter plots, which can be created in parallel; the computational complexity of creating each plot is determined by the complexity chosen method. If the plots are simple plots of pairs of original data features, the time needed to create each plot is simply linear with respect to the number of data points. If the plots are created by more advanced data-driven mappings, the complexity may depend both on the data and the original dimensionality; we describe selected examples. If a plot is created by a Principal Component Analysis (PCA) projection, computing the projection with a standard eigenvector decomposition approach takes $O(ND^2 + D^3)$ time for N data samples with original dimensionality D; some more efficient approaches have been proposed, see for example Sharma and Paliwal (2007). Many nonlinear dimensionality reduction (NLDR) approaches work based on the distance matrix without requiring knowledge of the original feature values: for example Sammon's Mapping (Sammon; Sammon 1969), Stochastic Neighbor Embedding (SNE; Hinton and Roweis 2003), t-distributed SNE (t-SNE; van der Maaten and Hinton 2008), and Neighbor Retrieval Visualizer (NeRV; Venna et al 2010) are all based on a matrix of Euclidean distances between data points; computing the matrix takes $O(N^2D)$ time, and the remaining iterative computation of the methods takes $O(N^2)$ time per iteration and is independent of the original dimensionality D. For some NLDR methods faster variants have been created; for example, Maximum Variance Unfolding (MVU; Weinberger and Saul 2006) involves semidefinite programming and a faster variant called Landmark MVU (LMVU; Weinberger et al 2005) has been created to improve scaling to larger data sets. For neighbor embedding approaches, a fast computation approach was recently proposed (Yang et al

2013), based on approximating distances to far-off points by distances to means of clusters in a quad-tree, yielding with $O(N \log(N))$ complexity; see Peltonen and Georgatzis (2012); Vladymyrov and Carreira-Perpinan (2014) for related speedup approaches.

Optimizing the meta-visualization first computes pairwise distances between plots in $O(N^2M^2)$ time for N data samples and M plots. The iterative NLDR optimization of the meta-visualization has $O(M^2)$ complexity per iteration. To avoid local minima, the method can be run in parallel from several initializations, taking the result with the smallest cost. In most cases the method yielded good results from a single random initialization. The fast approximate computation approach that was proposed for neighbor embedding by (Yang et al 2013) can also be used in meta-visualization, but we did not implement such approximations as the method was fast enough without approximation.

3.3 Alternative Approaches

As seen in Section 2 the related work has either not considered the task of how to automatically relate and arrange numerous plots, or has done so on an annotation-driven basis only rather than a data-driven basis, and our approach is the first neighbor embedding method organizing plots onto a meta-visualization. Out of the earlier methods we will provide a quantitative comparison to the most well-known one, the scatter plot matrix, in Section 4.1.

In this section we introduce two new alternative approaches that represent alternative ways how data-driven meta-visualization could be attempted without following our information retrieval principle. We will compare to these methods in Section 4.2, to demonstrate the benefit of the rigorous information retrieval approach.

An alternative approach needs to carry out the same two subtasks as our approach, distance measurement between plots and subsequent arrangement. We consider two alternatives for the first subtask.

As the first alternative, one can compare several plots of a data set by comparing the actual data coordinates shown in the plots directly. Let $\{\mathbf{y}_{m,i}\}_{i=1}^{N}$ and $\{\mathbf{y}_{m',i}\}_{i=1}^{N}$ denote locations of N data points in visualization m and m' respectively. We can define a distance measure between plots as

$$D_{m,m'}^{\text{naive}} = \sum_{i=1}^{N} \|\mathbf{y}_{m,i} - \mathbf{y}_{m',i}\| \,. \tag{7}$$

This metric is simple to compute and is data-driven; however, it is easy to show that such a metric is not invariant under transformations such as translation and rotation. While simple invariance could be added by measuring the minimal distance under affine transformations, the metric would remain non-invariant to local transformations and other nonlinear transformations. Nevertheless, the metric a useful first baseline for a data-driven comparison.

As a second alternative, we consider comparing the apparent shapes of data seen in different visualizations. To characterize data features like shapes that an analyst may be interested in, one can build a representation of the data shape in

a plot by concatenating different orders of moments of the data point coordinates as a feature vector for the plot. For plot m, let

$$f_{m,\alpha,\beta} = \sum_{i=1}^{N} \left[y_{m,i}^{(1)}\right]^{\alpha} \left[y_{m,i}^{(2)}\right]^{\beta} \tag{8}$$

where $\mathbf{y}_{m,i} = (y_{m,i}^{(1)}, y_{m,i}^{(2)})^{\mathrm{T}}$. We can then define the feature vector for plot m and the corresponding distance measure between plots m and m' as

$$f_m = (f_{m,1,0}, f_{m,0,1}, f_{m,2,0}, f_{m,1,1}, f_{m,0,2}, f_{m,3,0}, f_{m,2,1}, f_{m,1,2}, f_{m,0,3})^{\mathrm{T}} \tag{9}$$

$$D_{m,m'}^{\text{moment}} = \|f_m - f_{m'}\| , \tag{10}$$

that is, we will use a vector of moments computed up to the third order as the features of a plot, and norm of the difference between the moment vectors as the distance between plots. Note that moments up to any higher order can naturally be considered in the same way.

For both of these two alternatives, given the pairwise distances computed between visualizations, rather than using our information retrieval based layout an alternative simple approach would be to feed the distances into an off-the-shelf dimensionality reduction algorithm; here we consider giving the distances as input to one of the most well-known NLDR methods, Metric Multidimensional Scaling (MDS; see Borg and Groenen 2005). We will compare these two proposed alternative methods with our meta-visualization approach in Section 4.2.

Note that to our knowledge, no previous approaches to arrange plots onto a meta-visualization in a data-driven way exist; the closest method we are aware of is the method of Tatu et al (2012) which also used MDS as proposed above, but applied to Tanimoto similarities which were only based on an annotation of subspace parameters and not on the data. Thus the two alternative approaches proposed in this section already represent new approaches in that they are data-driven. In principle, other NLDR methods could be used in place of MDS; the choice of MDS is here reasonable as our proposed alternative methods can then be interpreted as data-driven variants to Tatu et al (2012).

4 Experiments

We demonstrate the meta-visualization in case studies. We use a benchmark S-curve data set, Olivetti faces data (400 face images of 40 persons, 64×64 pixels each) from `http://www.cs.nyu.edu/~roweis/data.html`, Face Pose data (images of 15 persons from 63 angles) from Gourier et al (2004), and a collection of gene expression experiments.

4.1 Meta-visualization of Feature Pairs, versus a Scatter Plot Matrix

We first show the ability of the meta-visualization to reveal to the analyst which plots are similar. At the same time we perform a simple quantitative comparison to the most well-known traditional meta-visualization method, the scatter plot matrix.

Consider analyzing a multivariate data set based on plots of each feature pair, where scatter plot matrix is a popular tool for such a task. Suppose some pairs actually provide the same information as other pairs; then this should be revealed to the analyst. Relationships between different feature pairs can be hard to see from a simple scatter plot matrix, but a well-optimized meta-visualization can reveal them.

We create a data set where each individual feature is unique, but some feature pairs contain the same neighborhood information as other pairs; we create a scatter plot of each feature pair, and show meta-visualization arranges the known-to-be similar pairs close-by.

In detail, we take a 5-dimensional face image data (a subset of 405 images from the Face pose data, each image rescaled to 16×16 pixels and projected to the 5 largest PCA components of the data set). We then add 20 new features: the original data has 10 feature pairs, and from every such pair $[x,\ y]$ we add two new features $[\cos(\pi/4)x - \sin(\pi/4)y,\ \sin(\pi/4)x + \cos(\pi/4)y]$ as a 45-degree rotation of the original features. The resulting 25-dimensional data contains $25 \cdot 24/2 = 300$ feature pairs to be visualized. Each of the 10 pairs of original features contains the same information as its rotated version, but noticing the 10 pairs and their matching other pairs without meta-visualization would be arduous.

Figure 4 (left) shows the meta-visualization. It reveals an interesting grouping of feature pairs, with several major groups which are further split into subgroups; such structure will be analyzed in later experiments, here we concentrate on analyzing the known ground-truth pairings of plots. Visually, the meta-visualization is very readable: as desired, optimizing the readability cost (repulsion) has kept plots at a distance so that they do not overlap. Note that in an interactive system the meta-visualization can be combined with focus+context techniques such as further enlargement of selected plots.

The 10 matching plot pairs we are interested in are shown with colored borders (same color for both plots in each pair). The meta-visualization placed the plots of the matching pairs close to one another as desired, which is intuitive as they contain the same information.

We compare the result to the widely used scatter plot matrix. Figure 4 (right) shows the same plots in a 25×25 scatter plot matrix. We colored the 10 original feature pairs and their 10 rotated versions with corresponding background colors. Unlike our meta-visualization, the 10 matching pairs of plots are now essentially in arbitrary positions which depend on the order of feature indices. It would be difficult to notice correspondence between a pair and its match from the scatter plot matrix; in contrast our meta-visualization finds the correspondence and shows it by plot locations on the meta-visualization.

We measure the performance difference between our method and the scatter plot matrix quantitatively by a retrieval measure, recall of matching pairs, by evaluating the 8-neighborhoods of the 10 feature pairs: on the meta-visualization, each of the plots of the 10 feature pairs has its matching rotated version as one of the 5 nearest neighboring plots, whereas in the scatter plot matrix, none of the 10 plots of feature pairs has the matching pair in the 8 nearest neighbors on the matrix. Thus the meta-visualization is more faithful to the data than the scatter plot matrix is.

The standard way to create a scatter plot matrix is to simply order the rows and columns according to the feature indices, and the scatter plot matrix we evaluated

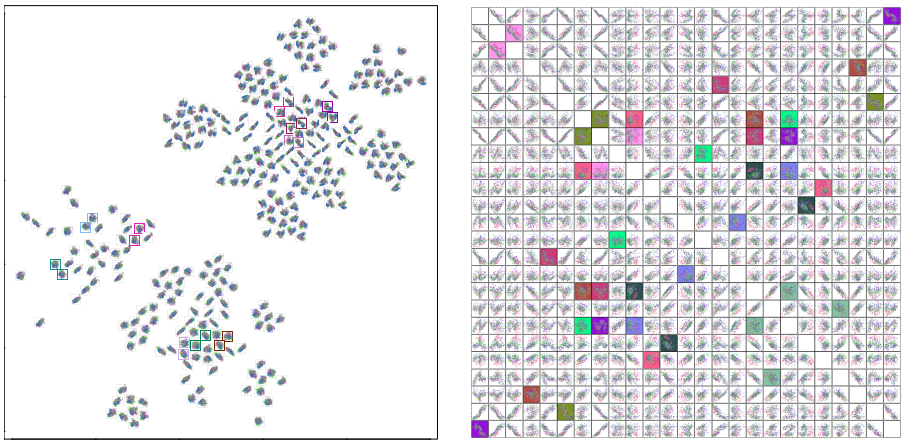

Fig. 4: **Left:** Meta-visualization of face pose image data. Each of the 300 mini-plots shows an individual feature pair. 10 plots m have a matching other plot m' where both plots show the exact same information up to rotation. For each of the 10 matches the meta-visualization placed the matching plots (colored mini-plot borders; corresponding colors are matches) close to each other. In each mini-plot, faces are shown as dots colored by person identity. **Right:** The same set of plots as a traditional scatter plot matrix. (Each plot in row i, column j also has a trivial match in the transposed cell, row j, column i.) The nontrivial matching plots are shown with background in the same color; it would be very difficult to notice the non-trivial matches from the scatter plot matrix. (A higher-resolution image of the matrix is available in supplementary material at `http://metavis.github.io/acml13`.)

above was based on this standard ordering. It turns out that even a more advanced data-driven ordering would not help the performance of the scatter plot matrix: in Appendix B we propose a reordering method where feature indices of a scatter plot matrix are reordered to keep the highly correlated features in close-by rows and columns of the scatter plot matrix. We show that even with such an advanced reordering, the scatter plot matrix nevertheless cannot keep the matching pairs of scatter plots nearby, thus providing even stronger evidence for the benefit of our meta-visualization approach.

The meta-visualization can also be used in cases where plots do not originate from feature pairs and thus an ordered scatter plot matrix cannot be trivially constructed; Section 4.5 shows meta-visualizations for such cases.

4.2 Comparison of Our Meta-visualization Approach to Alternative Approaches

The scatter plot matrix, which we compared our method to in the previous section, does not consider data-driven relationships between plots, but simply enumerates plots for each feature pair and arranges them into a grid. In this section, to further demonstrate the advantage of our data-driven information retrieval based meta-visualization approach, we quantitatively compare it with two alternative

data-driven methods introduced in Section 3.3 which represent alternative ways of computing similarities between plots and then arranging the plots according to similarities.

We compare the methods on a series of meta-visualization scenarios, where several plots of a data set are available, the plots differ in nonlinear ways created by local transformations, and a ground truth is available to evaluate which plots should be placed nearby in a meta-visualization.

We create 10 data sets, each of which consists of a mixture of several Gaussian clusters, where the data points arise from several ground truth classes which will be used as held-out information for performance evaluation. For each data set, several plots will be where the clusters are in different positions; the crucial difference between plots is then *which of the ground truth classes overlap in each plot*. We arrange the plots so that in each plot, some of the classes overlap each other so that they appear as a single Gaussian cluster, whereas other classes are shown as separated Gaussian clusters. Two plots where the same classes overlap are essentially similar in terms of the ground truth information, and should be shown nearby in a meta-visualization. We will show that our meta-visualization approach is capable of capturing the similarities between the plots so that the resulting display corresponds well to the underlying ground truth similarities. We also compare our results with the alternative approaches presented in section 3.3.

In detail, to create the plots corresponding to a data set, we generate 500 data points divided equally into 5 classes. We first create a 2D arrangement for the data points within each class, denoted as $\mathbf{y}_{1,1}, \cdots, \mathbf{y}_{1,100}, \cdots, \mathbf{y}_{5,100} \in \mathbb{R}^2$, where $\mathbf{y}_{i,j} \sim \mathcal{N}(0, I_2)$, and $i = 1, \cdots, 5$ is the index of the class and $j = 1, \cdots, 100$ is the index of the data point within the class. We then create 20 plots with different configurations of the classes, to be arranged by meta-visualization: for each plot k, we choose $2 \leq n_k \leq 5$ randomly as the number of distinct Gaussian clusters visible in the plot, and locations for the clusters centroids from the standard 2-dimensional uniform distribution; we then normalize the locations of the centroids so that the bounding boxes of the centroids are squares of the same size across different data sets. We then assign each class of data points to one of the clusters, by moving the origin of data points $\mathbf{y}_{i,j}$ from that class to the centroid of the cluster that the class was assigned to (see figure 5). Under this setting, the number of clusters is smaller than or equal to the number of classes, therefore some classes would probably be assigned to the same cluster and overlap on the plot as we hope.

Given each of the 10 data sets, we compute a meta-visualization with our approach and with the two alternative approaches from Section 3.3. We first show an example result and then perform the full quantitative comparison.

Example meta-visualizations for one of the data sets. Figure 6 shows the result for one of the 10 data sets. The three mini-plots shown with red frames are examples of visualizations where the same ground-truth classes overlap in each visualization, thus these three plots are examples of plots that should be kept nearby in a good meta-visualization. We can clearly see the highlighted mini-plots are clearly closer in the meta-visualization produced by our approach (top sub-figure in Figure 6) than in the meta-visualizations produced by the two alternative methods which are based on MDS layout of distances computed from data coordinate comparisons (bottom left sub-figure in Figure 6) and from comparison of data shape by moments of the distribution (bottom right sub-figure in Figure 6) sub-figure in Figure 6. While our meta-visualization approach successfully arranged the

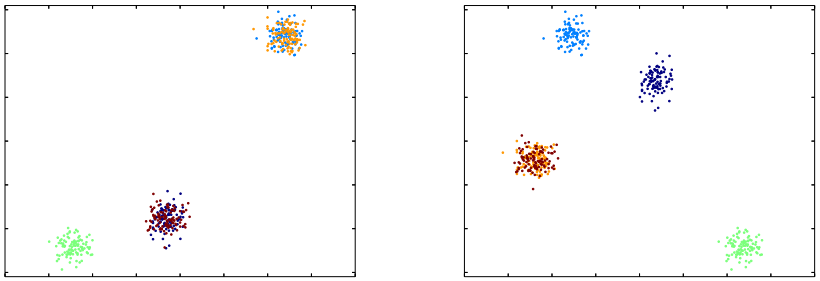

Fig. 5: Two examples of the scatter plots to be arranged in our quantitative comparison between meta-visualization methods. Two scatter plot visualizations of the same data set are shown, with different class overlap visible in both plots. In the left-hand plot the dark-red and dark-blue classes overlap at bottom center, and the orange and light-blue classes overlap at top right, whereas in the right-hand plot the orange and dark-red classes overlap at center left and other classes do not overlap.

highlighted similar plots, in the alternative methods the highlighted plots are not only located apart, but other non-similar plots have also been placed in-between them, potentially misleading the analyst. We next concretize the advantage of our approach by a quantitative evaluation of the comparison experiment over all 10 data sets. We perform the quantitative evaluation using two different performance measures.

Quantitative comparison of meta-visualization performance, part 1: comparison of information retrieval performance. The aim of our proposed meta-visualization approach is to make the physical neighborhoods (visual distance based neighborhoods) of the different plots on the meta-visualization consistent with the content-based neighborhoods (data-driven neighborhoods based on the content of the plots) in the sense of good information retrieval performance. We now measure the information retrieval performance for all the compared methods.

We use the standard *mean precision–mean recall* curve from the information retrieval field to quantitatively evaluate performance of the methods: the mean precision–mean recall curve plots the mean value of precision and recall (mean over queries) as the size of the retrieved set is varied.

In detail, for each plot m, $m = 1, \ldots, M$, let $U_m(r)$ denote the neighborhood of m containing the r plots m' (other than m itself) that are most similar to m according to the measure $D_{m,m'}$ defined in Eq. (2). Similarly, for each plot m, let $V_m(k)$ denote the physical neighborhood of m containing the k plots m' (other than m itself) that are closest to m according to the on-screen coordinates $\mathbf{z}_m$ of all the plots. The precision and recall for plot m are defined as

$$\text{precision}(m; r, k) = \frac{N_{TP,m}}{k} \tag{11}$$

$$\text{recall}(m; r, k) = \frac{N_{TP,m}}{r} \tag{12}$$

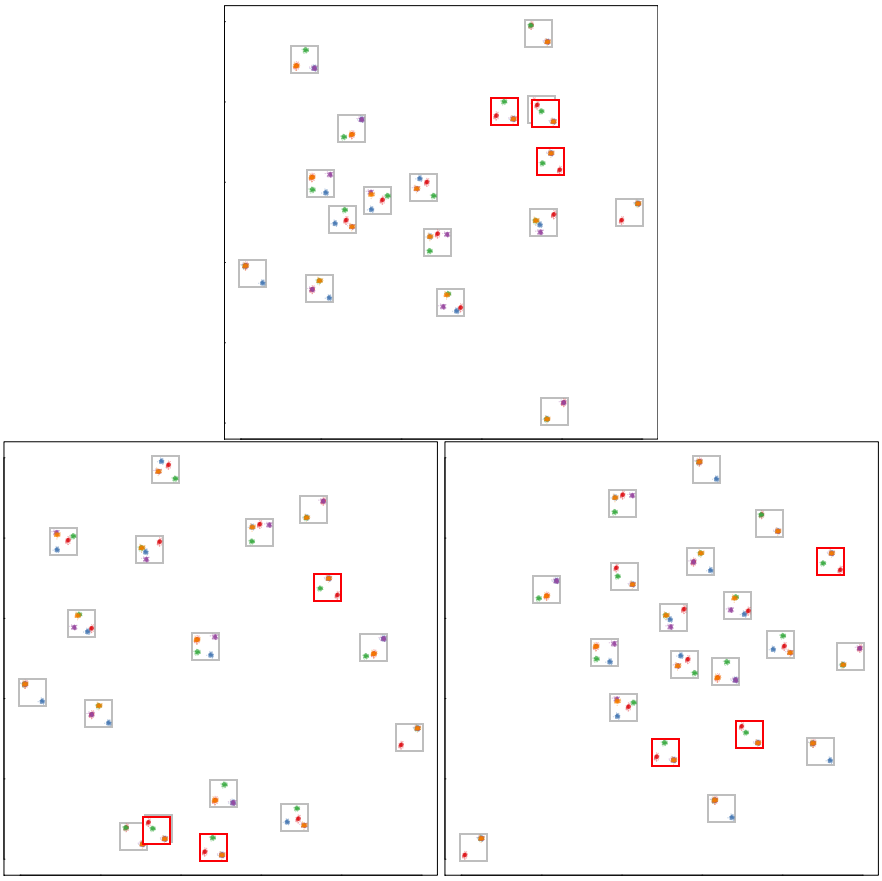

Fig. 6: Comparison of our meta-visualization approach to two MDS-based alternatives. The data comes from underlying classes shown by colors in each mini-plot; class labels are used as a held-out ground truth to compare meta-visualizations. In the ground truth, two scatter plots are similar if the same classes overlap in both plots. Three example plots where the same classes overlap are highlighted with a red border: in a good meta-visualization they should be nearby. **Top**: The result from our meta-visualization approach corresponds well to the ground truth: nearby plots show similar overlap of classes and far-off plots show different overlap, as desired. The three highlighted similar plots are nearby as desired. **Bottom left**: The result from MDS with locations of each data point in a plot used as features of the plot. The method overemphasizes simple changes of individual data point locations between plots, and the resulting display does not correspond well to the ground truth; for example the three highlighted plots are not nearby. **Bottom right**: The result from MDS with moment features (moments of the data distribution on each plot) for plots. The method overemphasizes shape of the distribution instead of noticing local similarities between plots. The result does not correspond well to the ground truth; for example the three highlighted plots are not nearby.

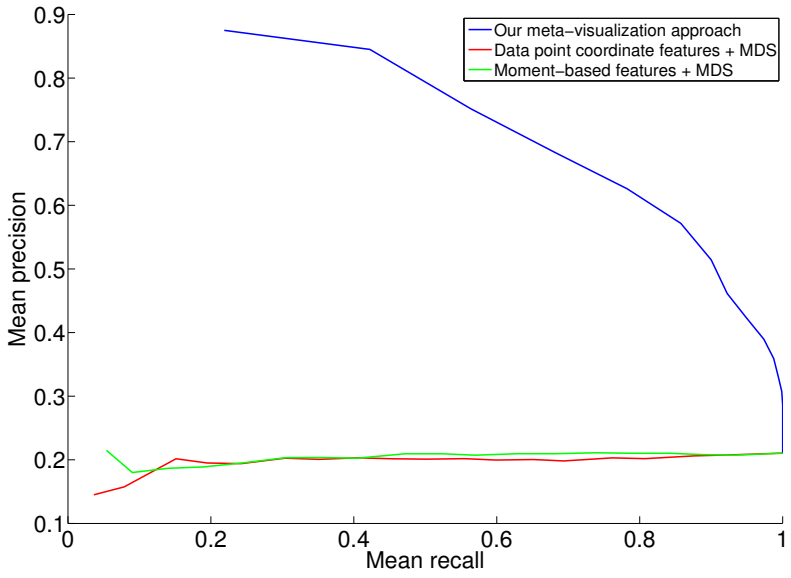

Fig. 7: Information retrieval performance of our proposed meta-visualization method and the two comparison methods (here denoted as "Data point coordinate features + MDS" and "Moment features + MDS") over the 10 Gaussian-cluster data sets, shown by the average (over the data sets) mean precision–mean recall curve with $r = 4$. High values of precision and recall mean that the visualizations that provide similar information to the analyst are placed nearby on the meta-visualization: high precision means few false neighbor plots (few false positive plots) and high recall means few missed neighbor plots (few false negative plots).Our meta-visualization approach clearly outperforms the alternative methods, achieving better precision and recall.

where $N_{TP,m} = |V_m \cap U_m|$ is the number of true positive neighbors, that is, the number of plots that are neighbors both according to V_m (that is, according to the physical distances on the meta-visualization display) and according to U_m (that is, according to the ground-truth similarity of the plots). The mean precision is the average of precision$(m; r, k)$ over m and mean recall is the average of recall$(m; r, k)$ over m. The mean precision–mean recall curve is plotted by fixing r (here we set $r = 4$) and varying the number of retrieved neighbor plots k between 1 and the maximum $M - 1$.

We plot the average of the mean precision–mean recall curve over the 10 artificial data sets for the three methods. The result is shown in Fig. 7. As shown in the figure, the mean precision–mean recall curve shows clearly better performance for our approach than for the two alternative approaches: we achieve clearly better mean precision at each value of mean recall. Our proposed meta-visualization approach has successfully arranged the plots according to their neighborhood relationships and thus has achieved good information retrieval performance.

Quantitative comparison of meta-visualization performance, part 2: comparison based on ground truth class labels. Since a ground truth classification is available for each of the 10 data sets, we can additionally measure the performance of the three methods by the *average class overlap mismatch*. The essential underlying difference between plots of that data set is which classes overlap in each plot, for each data set we evaluate the performance of meta-visualization approaches in arranging the plots by *average mismatch of class overlaps between a plot and its neighbor plots*. We define the performance measure as follows.

For classes C_i and C_j in the same scatter plot visualization V, let

$$D_V(C_i, C_j) = \begin{cases} 1 & \text{if } C_i \text{ and } C_j \text{ overlap} \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

so that the class overlaps present in visualization V can be described by the set of values $D_V(C_i, C_j)$ for all i and j. Using this definition, we can define a ground truth metric between two scatter plot visualizations, say, V_m and $V_{m'}$, as

$$D(V_m, V_{m'}) = \sum_{C_i, C_j} \left| D_{V_m}(C_i, C_j) - D_{V_{m'}}(C_i, C_j) \right| \tag{14}$$

which simply evaluates the *class overlap mismatch* by counting how many class pairs overlap differently between the two scatter plots (that is, the number of class pairs where the pair overlaps in one plot but not in the other).

Using the above definition we can measure the performance of a meta-visualization arrangement of plots by the average class overlap mismatch between plots and their neighbors, as

$$Cost = \frac{1}{k \cdot n_d} \sum_{V_m} \left[\sum_{V_{m'} \in N_k(V_m)} D(V_m, V_{m'}) \right] \tag{15}$$

where $n_d = 20$ is the number of data sets, and $N_k(V_m)$ denotes the plots in a k-nearest neighborhood of plot V_m on the meta-visualization display; we again use $k = 4$ for each experiment. The smaller the average mismatch is, the better the meta-visualization arrangement is.

We repeat the experiment for all 10 data sets. Table 1 shows the average and the standard deviation of the costs. We can see the performance of the coordinate features + MDS method is similar to the performance of the moment-based features + MDS method. And our meta-visualization approach not only achieves clearly lower average mismatch, but also has smaller standard deviation, which means our approach performs better and more stable.

Discussion. In this section we compared our meta-visualization approach to our suggested alternative methods. Note that we are not aware of other published data-driven methods to arrange plots on a meta-visualization display: the closest published method we are aware of is that of Tatu et al (2012) which is not data-driven. Therefore for the purposes of this comparison we used the novel alternative methods suggested in Section 3.3, which are the data-driven nearest equivalents to the method of Tatu et al (2012).

Our meta-visualization approach yielded clearly better performance than the comparison methods, both in terms of information retrieval performance (precision-recall curve) and in terms of a performance measure based on the ground truth

	Our meta-visualization approach	Data point coordinate features + MDS	Moment-based features + MDS
Cost	1.861 ± 0.307	3.128 ± 0.433	3.165 ± 0.541

Table 1: The average mismatch of class overlaps between a plot and its neighbors as measured by Eq. (15) across 10 data sets, for meta-visualizations created by our approach and by two alternative approaches here denoted as "Data point coordinate features + MDS" and "Moment-based features + MDS". Our meta-visualization approach achieves clearly better arrangements (smaller mismatch) than the other approaches.

class labels of data. The main reason for the better performance of our approach is likely that in the alternative methods, the similarity measure between the plots (based on comparison of data point coordinates between plots, or comparison of moment features between plots) is not able to capture the neighborhood relationship content in the plots as well as our proposed method where similarity of plots is measured based on an information retrieval approach. For example our information retrieval approach can notice similarity between plots that show the same clusters and the same neighborhood relationships, even if the locations of cluster centroids differ somewhat between the plots, and hence our meta-visualization arranges such plots close-by; in contrast, the alternative methods based on data coordinates or moments might be more strongly affected by the changes in the cluster centroid locations.

The good experimental performance suggests that our information retrieval approach is a promising approach for meta-visualization.

4.3 Effect of the Repulsion Term in Meta-visualization

To improve readability, our meta-visualization approach keeps the mini-plots non-overlapping on the display by including a Gaussian repulsion term in the objective function, Eq. (6). We here briefly demonstrate how different repulsion magnitudes will affect the meta-visualization.

We create a setting where we have a ground-truth clustering of the available plots for a data set. We create several plots of the Olivetti face image data set to be arranged by meta-visualization. In each plot, the Olivetti faces are arranged by NLDR based on similarity of a subpart of the image, and each plot uses a different subpart to arrange the faces. Thus each plot represents the identifying information among faces visible in a different subpart.

In detail, the Olivetti face images are each 64×64 pixels. To create a two-dimensional plot of the image set, we take a 32×32 sub-window from the same location in all images, compute distances between images as average mean-squared distance of the pixel values in the window, and give the resulting distance matrix to MDS which then embeds the face images onto a two-dimensional plot. To create several plots of the face image data set, we take the sub-windows from different locations each time: we take 9 sub-windows near the top-left corner, 9 near the top-right corner, 9 near the bottom-left corner, and 9 near the bottom-right corner, yielding 36 plots in total for the meta-visualization. The plots corresponding to

sub-windows near the same corner will naturally be similar, thus each of the four corners will yield a cluster of 9 plots in the meta-visualization.

Figure 8 shows the meta-visualizations by our method created with different repulsion coefficients μ; the figure shows the results with coefficient values 0, 5, 20, and 30; behavior with intermediate values is similar. The method behaves in a reasonable and expected manner with respect to the repulsion; from top-left to bottom-right, we can see the meta-visualizations show the ground-truth cluster structure in the plots correctly, and the mini-plots arrangement changes smoothly from an strongly clustered layout where repulsion is not used, to a more evenly spread layout where mini-plots are mostly non-overlapping but still preserve the cluster structure. The analyst can thus choose the amount of repulsion according to how important the non-overlappingness is for the particular data, and the resulting meta-visualizations can of course be combined with standard focus+context techniques for further investigation. In conclusion, our repulsion coefficient works in a reliable way to improve the readability of meta-visualization.

4.4 Case Study: Meta-visualization of Hyperparameter Influence on NLDR

In this subsection and the following two subsections we provide case studies of using our meta-visualization approach for data analysis in three different scenarios. We first analyze *hyperparameter influence on a prominent NLDR method.*

Besides analyzing data by feature pairs or simple projections, NLDR is often used to map high-dimensional data onto a two-dimensional plot, hoping to capture essential data structure. NLDR cannot preserve all properties of high-dimensional data in one low-dimensional plot (Venna and Kaski 2007; Venna et al 2010); an NLDR method implicitly chooses some aspect of the data to show, with trade-offs such as global vs. local preservation, trustworthiness vs. continuity, and others. A single NLDR result is thus insufficient to analyze a data set and multiple NLDR results should be created. To create multiple NLDR results one can (1) run multiple NLDR methods, or (2) run variants of an NLDR method by e.g. adjusting parameters to emphasize different data aspects. We treat the first case in Section 4.5, in this section we treat the second case. We create multiple plots with one NLDR method, and use meta-visualization to study the results. Besides the different views of data given by the NLDR method, meta-visualization can give insight into behavior of the NLDR method.

As a case study we create a meta-visualization of Olivetti faces data, where 20 different plots are created by the NLDR method Neighbor Retrieval Visualizer (NeRV; Venna et al (2010)), which has performed well in recent comparisons of NLDR methods. NeRV has a precision-recall trade-off hyperparameter λ between 0 and 1; we vary it with values in $[0, 0.04, \ldots, 0.96]$. With λ near 0 NeRV emphasizes precision and avoids false neighbors; with λ near 1 NeRV emphasizes recall and avoids misses. It has been shown Venna et al (2010) that emphasizing precision or recall yields different plots; we use our method to meta-visualize the trade-off. Figure 9 shows the result. The hyperparameter values yield a smooth continuum of plots; as an interesting discovery, the difference in results between close-by λ values is small at the recall-emphasizing end (λ near 1; green plot border) but at the precision-emphasizing end (λ near 0; dark plot border) differences are larger, indicating that the trade-off parameter λ is not linear w.r.t. the actual trade-off

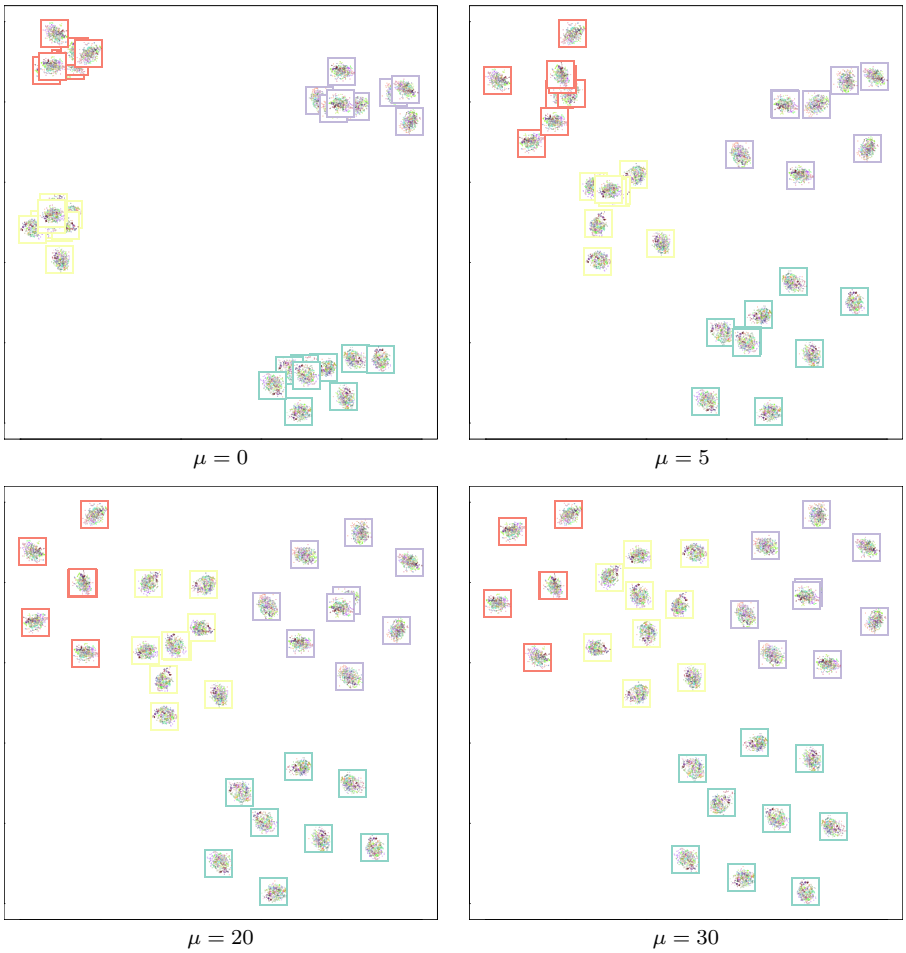

Fig. 8: Results with different repulsion coefficients μ. Small values of the coefficient yield visualizations highly informative of the similarities of the plots, at the expense of possible overlappingness of the plots. Higher values of the coefficient spread out the natural clusters of plots to improve readability (non-overlappingness). All values of the repulsion coefficient yield good meta-visualizations; the coefficient allows tuning of the importance of non-overlappingness in meta-visualization according to the preference of the analyst.

between precision and recall, thus care must be taken to set the λ when the analyst wants a trade-off mostly emphasizing precision. Thus our meta-visualization revealed insights into roles of the hyperparameters that would have been hard to find in a non-data-driven way, and would have been hard to see from one plot or an unorganized set of plots.

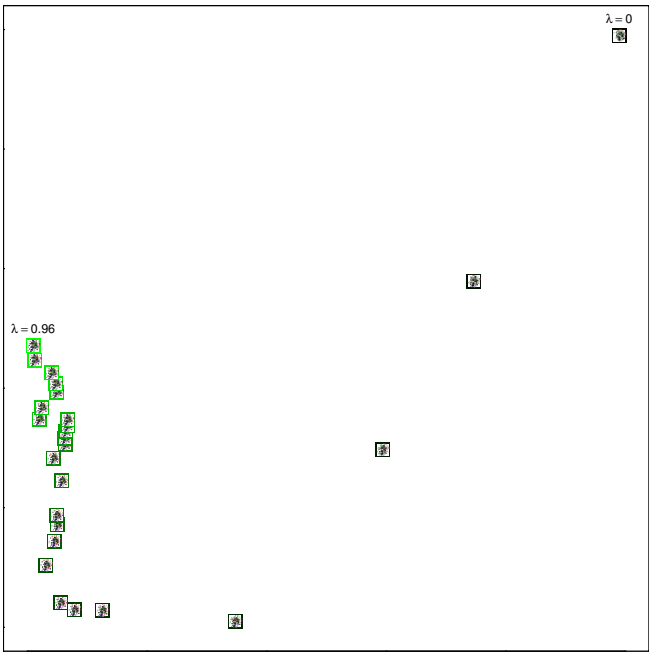

Fig. 9: Meta-visualization of the influence of the precision-recall trade-off hyperparameter λ on the NeRV method. 20 visualizations are shown for the Olivetti faces data, created by NeRV with different values $\lambda \in [0, 0.04, \ldots, 0.96]$. Intensity of green color = value of λ. The meta-visualization arranges the plots as a continuum where changes between successive λ values are larger at the precision end. Mini-plots show the face visualizations; for simplicity faces are shown as dots colored by identity of the person.

4.5 Case Study: Differences between Nonlinear Embedding Methods

We apply our meta-visualization method to visualize similarities between results of several state of the art linear and nonlinear dimensionality reduction methods on two data sets. Results of numerous NLDR methods, arranged by a meta-visualization, allow a more comprehensive understanding of a data set than the result of one NLDR method; such results can also yield insights into relationships of the NLDR methods themselves. An NLDR method implicitly chooses what aspect of data to show, based on their cost function or algorithm; what aspect each NLDR method will show can be hard to see from the mathematical formulation of the method; moreover, relationships between NLDR methods can be hard to analyze in a non-data-driven manner as the mathematical approaches vary greatly from generative models to spectral approaches to distance preservation criteria and others. For example, a developer of a new NLDR method might be interested to use meta-visualization to analyze how similar results of the new method are to results of established methods.

We use two data sets: a simple three-dimensional benchmark data set "S-curve" (points distributed along an S-shaped sheet) and the real-world Olivetti

face data set. We create plots of the data sets with 19 methods: Principal Component Analysis (PCA; Hotelling 1933), Kernel PCA (Schölkopf et al 1999), Probabilistic PCA (ProbPCA; Tipping and Bishop 1999), Factor Analysis (see Child 2006), Gaussian Process Latent Variable Model (GPLVM; Lawrence 2004), Metric Multidimensional Scaling (MDS; see Borg and Groenen 2005), Sammon's Mapping (Sammon; Sammon 1969), Curvilinear Distance Analysis (CDA; Lee et al 2004), Stochastic Proximity Embedding (SPE; Agrafiotis 2003), Locally Linear Embedding (LLE; Roweis and Saul 2000), Hessian LLE (HLLE; Donoho and Grimes 2003), Laplacian Eigenmap (LE; Belkin and Niyogi 2002), Diffusion Maps (Lafon and Lee 2006), Maximum Variance Unfolding (MVU; Weinberger and Saul 2006), Landmark MVU (LMVU; Weinberger et al 2005), Stochastic Neighbor Embedding (SNE; Hinton and Roweis 2003), Symmetric SNE (s-SNE; van der Maaten and Hinton 2008), t-distributed SNE (t-SNE; van der Maaten and Hinton 2008), Neighbor Retrieval Visualizer (NeRV; Venna et al 2010). We briefly discuss the methods below.

Principal Component Analysis finds a linear projection where the "variance", or the sum of squared distances of the projected data points from their mean, is maximized. **Kernel PCA** is a kernelized extension of PCA. **Probabilistic PCA** builds a Gaussian noise model for the latent projection, and solves it via maximum likelihood. **Factor Analysis** is similar to Probabilistic PCA but does not estimate the level of the isotropic Gaussian noise from the likelihood. Instead, it estimates the noise level for each component directly from the data. The **Gaussian Process Latent Variable Model** is a non-linear extension for Probabilistic PCA via Gaussian processes. **Metric Multidimensional Scaling** tries to preserve the high-dimensional pairwise distances as much as possible in the low-dimensional space. **Sammon's Mapping** can be seen as an variant of MDS, which gives more importance to preserving the smaller distances. **Curvilinear Distance Analysis** improves Sammon's Mapping with a more sophisticated weighting for small distances. It also substitutes *geodesic distances* for Euclidean distances. **Stochastic Proximity Embedding** has a similar goal as MDS, but does the task in a different iterative way. **Locally Linear Embedding** finds a local linear representation for each data point based on its neighbors. **Laplacian Eigenmap** constructs a *neighborhood graph* for the data where each data point is a vertex. An edge between a point pair is formed if and only if one point is within the k-nearest neighborhood of another. The lower dimensional representation can be obtained by the first non-trivial eigenvectors of the *Laplacian* of the graph. **Hessian LLE** is similar to Laplacian Eigenmap, where the Laplacian is replaced with the *Hessian*, which captures "curviness" characteristics of the data. **Diffusion Maps**, on the other hand, defines *diffusion distances* for the point pairs of the data set, and then similarly gives lower dimensional embedding by eigen-analysis. **Maximum Variance Unfolding** "unfolds" the manifold by finding a Gram matrix which maximizes the distances between points that are not connected in the neighborhood graph by semidefinite programming. **Landmark MVU** is a variant of MVU which increases speed by using representative landmark data points, at the cost of accuracy. The **Neighbor Embedding** family, including **Stochastic Neighbor Embedding**, **Symmetric SNE**, and **t-distributed SNE**, first defines neighborhood distributions for both input space and output space, and then minimizes some metric, e.g., Kullback-Leibler divergence, between the two distributions. **Neighbor Retrieval Visualizer** is a recent dimensionality reduction approach based on information

retrieval. It formalizes visualization as minimization of two kinds of errors – false neighbors and misses during retrieval of data points.

To simulate a realistic situation where the analyst does not spend equal amounts of time optimizing every visualization, we optimized parameters of CDA, Laplacian Eigenmap, LLE, HLLE, MVU, LMVU, and NeRV to maximize a F-measure of smoothed rank-based precision and recall within each visualization as described in Venna et al (2010) – we maximize $F = 2(P \cdot R)/(P + R)$ where P and R are the two Kullback-Leibler divergences as in Eq. (5), but only the $u(m'|m)$ and $v(m'|m)$ are replaced by the ranks of the nearest neighbors. For the other methods we used implementations in a recent software package[4] with default parameters. To avoid sensitivity to initialization, each method is performed several times.

S-curve benchmark data set. Figure 10 (top) shows the result of meta-visualization of the S-curve benchmark data. Notably, among the 19 methods there seem to be several alternative ways to arrange the data: PCA, GPLVM, MDS, and Diffusion Maps have each found an essentially linear projection of the S-curve along its major two directions, and are arranged close together. ProbPCA is similar but has rotated the data. LLE and HLLE are related methods and are shown close-by; they have unfolded the S-curve in a slightly more nonlinear fashion. Sammon's mapping, SPE and CDA are shown close-by, they have unfolded the data non-linearly except for some remaining curled parts near the ends of the S. NeRV and MVU, shown near to each other, have both found a clean-looking unfolding of the S-curve manifold. SNE and t-SNE are two methods from the same family and are shown close-by; they have unfolded the manifold at the expense of some twisting and tearing. Kernel PCA, LMVU and Laplacian Eigenmap have all found a U-shaped curve based visualization. An outlier is s-SNE which has yielded a curious ball shaped arrangement. The meta-visualization arrangement has thus revealed prominent groups of typical NLDR results, which are related to underlying theoretical similarities of the methods.

Olivetti faces data set. Figure 10 (bottom) shows the result of meta-visualization of the Olivetti faces data. Among the 19 methods there are again several alternative ways to arrange the data, but whereas on the S-curve several methods found essentially the same embedding, on this more complicated data there are more differences visible between methods. ProbPCA, Factor Analysis, and GPLVM have again found a similar embedding, and NeRV is also similar to them, but MDS now differs from them with slightly less outliers and is instead close to Sammon's mapping. On this more difficult high-dimensional face data data t-SNE finds a clearly different embedding than normal SNE, which is intuitive since the use of the t-distribution in t-SNE was specifically designed to help with embedding of higher-dimensional data sets; t-SNE is here close to CDA, and SPE is an intermediate method between the CDA/t-SNE type result, the Sammon's mapping type result, and the essentially linear result seen e.g. in PCA. MVU and LLE have found embeddings with prominent outlier clusters, and Laplacian Eigenmap again finds a somewhat U-shaped arrangement. Here Diffusion Maps, Kernel PCA, and HLLE all yield very scattered embeddings with strong outliers. SNE and s-SNE both yield spherical arrangements but closer inspection reveals that the arrangements are dissimilar, in particular s-SNE has a more regular arrangement of the points.

[4] MATLAB toolbox for dimensionality reduction 0.8.1b, Laurens van der Maaten 2013

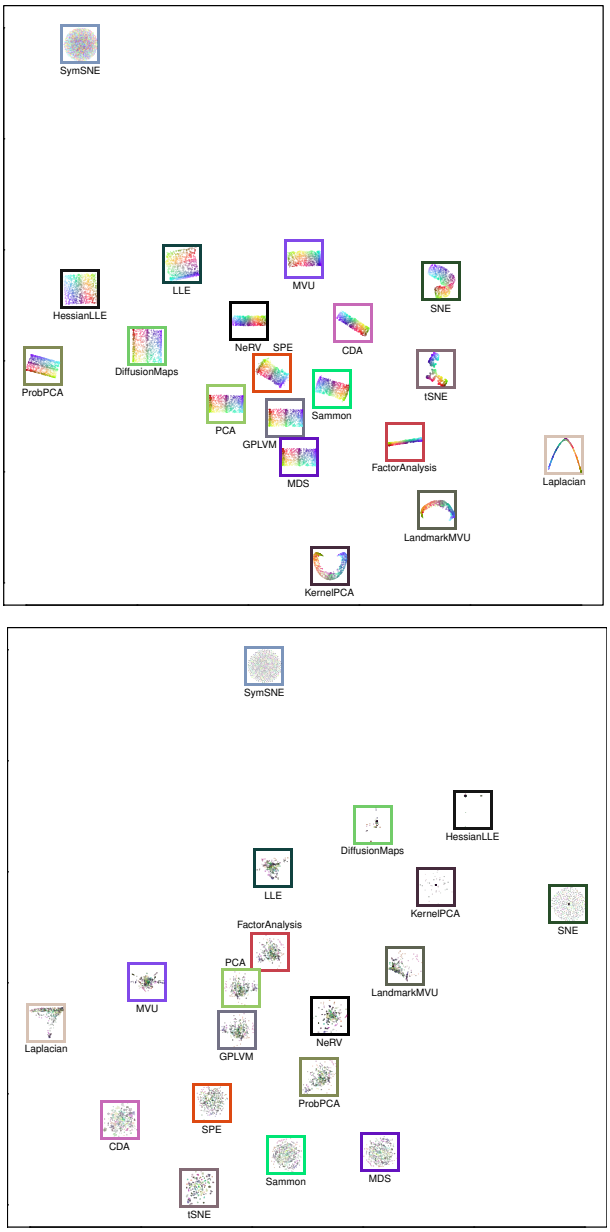

Fig. 10: **Top:** Meta-visualization of linear and nonlinear dimensionality reduction algorithms operating on the s-curve data set. The red-green-blue color components of each data point shows the original three-dimensional coordinates of the point. Border colors of the plots simply indicate the different NLDR methods. **Bottom:** Meta-visualization of the dimensionality reduction algorithms operating on the Olivetti face data set. Data points are colored according to the identity of the person. Border colors of plots again indicate the different NLDR methods.

Overall, the meta-visualization again yielded a helpful arrangement of plots, which revealed interesting behavior of the NLDR methods.

4.6 Case Study: Meta-Visualization of a Gene Expression Experiment Collection

We use meta-visualization to analyze a collection of human gene expression experiments from the ArrayExpress database Parkinson et al (2009), containing $d = 105$ "healthy-vs-disease" comparison experiments. Labels "*cancer*", "*cancer-related*", "*malaria*", "*HIV*", "*cardiomyopathy*", or "*other*" are available for the experiments. Our interest is how differences between experiments (diseases) are visible in activity of different sets of gene pathways.

As preprocessing we build on the work of Caldas et al (2009), who used gene set enrichment analysis (GSEA) to measure, for each experiment, activities of $w = 385$ known gene pathways, from the manually compiled C2-CP collection in the Molecular Signatures Database. They then trained a data-driven topic model on pathway activities; the topics are activity profiles of simultaneously active pathways across the experiments (Figure 11 shows an diagram telling for 13 selected topics which experiments the topics were active in and which pathways are active in each topic). We take the $t = 50$ topics modeled by Caldas et al (2009), and consider for each topic the subset of most active pathways as a feature set for the experiment collection. These $t = 50$ pathway subsets represent different aspects of biological activity across the experiments; we use each pathway subset to plot the experiment collection, and use meta-visualization to analyze how differences between diseases are visible in different pathway subsets. Caldas et al (2009) had visualized experiments only as a single plot of overall topic activities, not by detailed activities within pathway subsets; our meta-visualization complements their work.

In detail, let $\mathbf{Y}$ be the $d \times w$ matrix of pathway activities (for d experiments and w pathways), where each element y_{ij} is the activity (size of the leading edge gene subset) of pathway j in experiment i. Let $\mathbf{Z}$ be a $t \times w$ matrix inferred from $\mathbf{Y}$ by a topic model, representing t topics active across the experiments (when topic models are applied in text data $\mathbf{Z}$ is the "topic-to-word matrix"): here each element z_{mj} is the inferred activity of pathway j in topic m, and $\mathbf{z}^m$ is the vector of activities of all pathways in topic m.

From each topic m we create a feature set for the experiment collection, representing the pathways active in the topic.

To do so, for each topic we take the most active pathways, by taking the features in $\mathbf{Y}$ corresponding to s_m largest elements of $\mathbf{z}^m$. Denote the feature matrix consisting of the chosen features as $\mathbf{Y}_m$. For each topic the number of features s_m is chosen by power to discriminate diseases; the highest leave-one-out accuracy of k-nearest neighbor classification was first determined over k and s_m, and the minimal s_m reaching that accuracy was chosen.

For each topic m we plot the experiments as a linear discriminant analysis projection $\mathbf{V}_m = \mathbf{W}_m\mathbf{Y}_m$ where $\mathbf{W}_m = (w_{ij;m})_{2\times s_m}$ is the matrix of the linear discriminant weights. Each plot shows how much the pathways in the topic can discriminate the diseases in the collection. We then use meta-visualization to study how discriminative power varies across pathway subsets. Figure 12 shows the

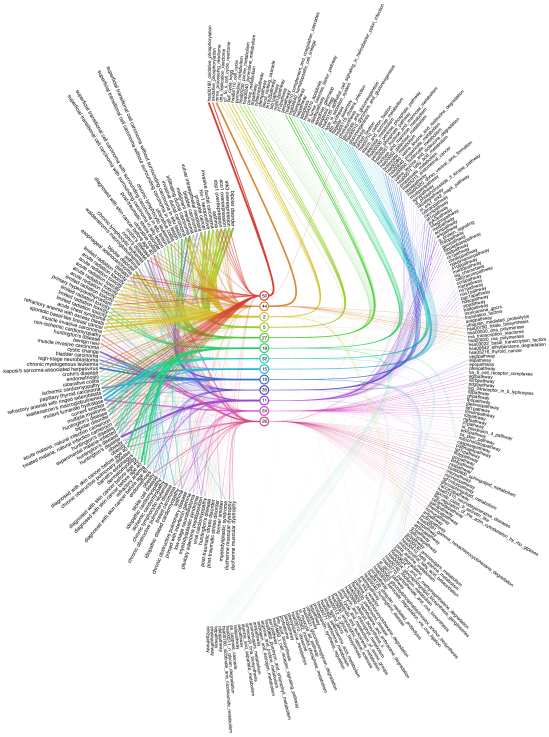

Fig. 11: An diagram of weights in a data-driven topic model on pathway activities, from Caldas et al (2009) with permission. The diagram shows activity of 13 of the 50 topics (small circles in the center). Lines connecting the topics to the left part of the diagram show which comparison experiments each topic was active in; the width of the line corresponds to activity of the topic. Lines connecting the topics to the right part of the diagram show which gene pathways are active in each topic; the width of the line denotes activity of the pathway in the topic. As many of the pathways are active in a particular topic, the topics provide interesting different representations of the gene activity in the experiments, which are well suited to create individual plots to be analyzed with meta-visualization.

result. Within each mini-plot, experiments are shown as dots colored by the disease annotation: *cancer* (cyan), *cancer-related* (blue), *malaria* (green), *HIV* (black), *cardiomyopathy* (red), and *other* (gray).

The meta-visualization finds groups of topics (pathway subsets) with similar discriminative power, which show different biological aspects of the experiment collection. We point out main groups. In group **A**, cancer-related, cancer, and malaria are discriminated. Cardiomyopathy is partly mixed with cancer and others. In group **B**, malaria is discriminated. Cancer-related and cancer have little overlap. Cardiomyopathy is mixed with cancer. Four plots below the group are similar to the group but also discriminate cardiomyopathy. In group **C**, most classes are heavily mixed, but cancer and cardiomyopathy have trails that spread out from the central mix. Group **D** is similar to group **C**, but with less overlap between cancer-

related and cancer. In group **E**, cardiomyopathy and cancer-related are mostly separated, and cancer-related is mixed with cancer. Malaria is not discriminated well in most visualizations of the group. Cancer is heavily mixed with others. In group **F**, cardiomyopathy and cancer are well separated; cancer-related and cancer are somewhat separated but cancer has heavy overlap with other. The differences of discriminative ability shown in the meta-visualization can be analyzed together with what pathways are active in each group of plots; see Caldas et al (2009) for annotations of pathways used in the topics. Table 2 lists for each cluster the top pathways having high activity within the cluster and being in the discriminative sets of at least two plots in the cluster. As an example, in group **A**, some of the most active pathways are related to apoptosis and to tumor necrosis; it is well known that apoptosis has a crucial role in cancer development (Lowe and Lin 2000), and tumor necrosis factor also has many functions in cancer biology (Waters et al 2013), thus the active pathways may explain why cancer and cancer-related diseases are well discriminated within the group. In group **B**, the TCR pathway and BCR Signaling pathway correspond to T-cell receptor and B-cell receptor respectively, and the FCER1 pathway is for the high-affinity IgE receptor, where IgE denotes Immunoglobulin E, an antibody involved in immunity against parasites including malaria parasites (Porcherie et al 2011); these immune system-related pathways may account for the discrimination of the malaria experiments in the group. In group **E** the active Pitx2 pathway is responsible for some heart diseases (Franco and Campione 2003), whereas inhibiting the 4-1BB pathway (Cheung et al 2007) or intravenous galactose (Frustaci et al 2001) can help with the treatment of heart diseases; the roles of these active pathways may then explain why cardiomyopathy is well separated in the group. In group **F**, the P38 MAPK pathway is a regulator of cancer progression (Bradham and McClay 2006), deregulation of elements of the mTOR pathway have been reported in many types of cancers (Pópulo et al 2012) and the ERK5 pathway has been suggested to be biologically important in prostate cancer (Ramsay 2010); these active pathways may then explain why cancer is well separated in the group.

Some biologically related topics had different abilities to discriminate diseases, potentially indicating their discriminative power comes from effects not shared among the topics, which can be analyzed in follow-up studies.

In summary, meta-visualization yielded insight into how differences between diseases in the collection are visible across subsets of gene expression pathways.

5 Conclusions and Discussion

We introduced a machine learning approach to meta-visualization; we arrange scatter plots onto a meta-visualization display so that similar plots are close-by. We contributed (1) an information retrieval based nonlinear dimensionality reduction (NLDR) formalization of the meta-visualization task; (2) a data-driven divergence measure between plots; (3) an information retrieval based NLDR method that arranges plots onto a meta-visualization.

Our distance measure and NLDR method were both derived from an information retrieval task. The similarity of visualizations (scatter plots) was defined by information retrieval costs in an information retrieval task of the analyst, retrieval of neighbor points from the plots. Plots are similar if, for each query point, they

Cluster	Top pathways
A	APOPTOSIS APOPTOSIS_KEGG APOPTOSIS_GENMAPP ST_TUMOR_NECROSIS_FACTOR_PATHWAY ANDROGEN_AND_ESTROGEN_METABOLISM
B	AMINOACYL_TRNA_BIOSYNTHESIS TCRPATHWAY SIG_BCR_SIGNALING_PATHWAY FCER1PATHWAY HSA04330_NOTCH_SIGNALING_PATHWAY
C	HSA04742_TASTE_TRANSDUCTION ALANINE_AND_ASPARTATE_METABOLISM STRIATED_MUSCLE_CONTRACTION HSA04950_MATURITY_ONSET_DIABETES_OF_THE_YOUNG TYROSINE_METABOLISM
D	HDACPATHWAY BADPATHWAY CHREBPPATHWAY INOSITOL_PHOSPHATE_METABOLISM CALCINEURINPATHWAY
E	HSA00052_GALACTOSE_METABOLISM GALACTOSE_METABOLISM PITX2PATHWAY 41BBPATHWAY
F	ST_P38_MAPK_PATHWAY ERK5PATHWAY ST_INTERLEUKIN_4_PATHWAY MTORPATHWAY

Table 2: Pathways having the highest activities within each cluster in the meta-visualization of Figure 12. Each plot in the meta-visualization was created based on a topic (a probability distribution over pathway activities) in a topic model of the experiment collection, using subset of pathways selected for their power to discriminate diseases. For each cluster we average the pathway activity probabilities over all topics corresponding to the plots in the cluster, leave out pathways that are discriminative only in one plot, sort the set of activity probabilities, and list the pathways having the highest probabilities. Some of the active pathways may explain disease discrimination capabilities within clusters, see the main text for discussion.

yield similar retrieved neighbors around the point. The dissimilarity between each pair of plots is quantified as the total cost of missing neighbors of one plot when retrieving them from the other plot, which was generalized to a rigorous divergence measure for probabilistic neighborhoods.

The meta-visualization is then optimized to arrange similar plots close-by, by minimizing a divergence between meta-level neighborhoods of the plots and corresponding neighborhoods of their locations on the meta-visualization, with additional costs measuring overlap of plots. This optimization has a rigorous interpretation as *optimization of a meta-visualization information retrieval task*, where the analyst retrieves similar plots from the meta-visualization.

In experiments the method was shown to have better performance than alternative approaches in quantitative comparisons, and it yielded promising results

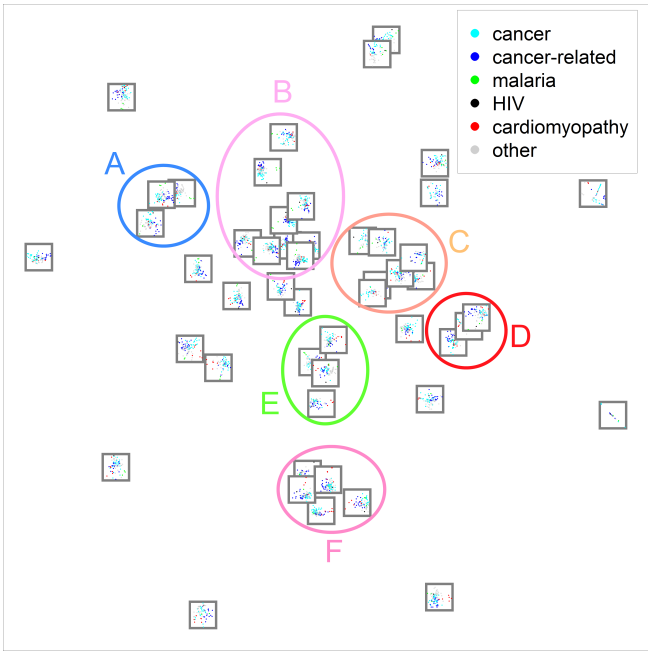

Fig. 12: Meta-visualization of a gene expression experiment collection from ArrayExpress; each mini-plot is a discriminative plot where disease experiments are separated based on activity in a subset of gene pathways (different pathway subset in each plot). Points within a plot are experiments, colored according to disease annotations. Ellipses and capital letters indicate groups discussed in Section 4.6. The meta-visualization varies smoothly with respect to hyperparameters, results at `http://metavis.github.io/acml13`.

in many tasks: finding visualizations that are equivalent despite using separate features; analyzing behavior of an NLDR method with respect to its hyperparameters; analyzing relationships of a large number of state of the art NLDR methods; and analyzing relationships of gene pathway subsets in a collection of gene expression studies over several disease types. Overall the meta-visualization method is a promising new approach for analysis of multiple plots of data sets.

Acknowledgements The work was supported by Academy of Finland, decisions 251170 (Finnish CoE in Computational Inference Research COIN), 252845 and 256233. Authors belong to COIN. We also acknowledge the computational resources provided by Aalto Science-IT project.

A Proof of the Connection between the Meta-Visualization Cost Function and Precision and Recall

In Section 3.2 we introduced a sum of two types of Kullback-Leibler divergences as a cost function for meta-visualization: the divergences compare neighborhoods u_m of plots according

to their content to neighborhoods v_m based on physical locations of the plots on the meta-visualization.

Here we provide the proof of the information retrieval connection: we show that the sum of divergences is a generalization of the total costs of two types of information retrieval errors, missed neighbor plots (false negative plots) and false neighbor plots (false positive plots).

In Section A.1 we first show that in a simple case of "binary neighborhoods" between plots the total cost of errors can be written in terms of precision and recall; in Section A.2 we then show that the Kullback-Leibler divergences reduce to precision and recall. Here "binary neighborhoods" means that, both according to the original information retrieval distances between plots and according to physical distances on the meta-visualization display, (i) the plot being considered has one or more relevant neighbor plots, and all other plots are irrelevant, and that (ii) the relevant plots are all equally relevant.

A.1 Connection between the Total Cost of Errors and Precision and Recall

Let m be the plot of interest, and let U_m be the set of relevant neighbor plots for plot m according to the information retrieval based comparison measure discussed in Section 3.1. U_m can be the set of all plots other than m whose distance from m according to the comparison measure is smaller than some fixed threshold, or it can be the set having a fixed number of points nearest to m according to the comparison measure. In either case, let r_m be the size of U_m.

Similarly, let V_m be the set of neighbor plots for plot m based on their physical locations on the meta-visualization display. Again, V_m can be the set of all plots (other than m itself) whose physical distance to m on the meta-visualization is smaller than some fixed radius, or it can be the set containing some fixed number of plots nearest to m on the meta-visualization. In either case, let k_m be the size of V_m. Note that the sizes of V_m and Q_m can be different, that is, k_m can be different from r_m.

Denote the number of samples that are in both U_m and V_m by $N_{\mathrm{TP},m}$ (true positives), samples that are in V_i but not in U_m by $N_{\mathrm{FP},m}$ (false positives), and samples that are in U_m but not V_m by $N_{\mathrm{MISS},m}$ (misses). Assume the user has assigned a cost C_{FP} for each false positive and C_{MISS} for each miss. The total cost E_m for query m, summed over all plots, then is

$$E_m = N_{\mathrm{FP},m} C_{\mathrm{FP}} + N_{\mathrm{MISS},m} C_{\mathrm{MISS}} \ . \tag{16}$$

This cost bears a close relationship to the traditional measures of information retrieval, precision and recall. If we let C_{MISS} be a function of the total number of relevant points r_m, specifically $C_{\mathrm{MISS}}(r_m) = C'_{\mathrm{MISS}}/r_m$, and take the cost per retrieved point by dividing by k_m, the total cost becomes

$$\begin{aligned} E(k_m, r_m) &= \frac{1}{k_m} E(r_m) = \frac{1}{k_m} \left(N_{\mathrm{FP},m} C_{\mathrm{FP}} + N_{\mathrm{MISS},m} C_{\mathrm{MISS}}(r_m) \right) \\ &= C_{\mathrm{FP}} \frac{N_{\mathrm{FP},m}}{k_m} + \frac{C'_{MISS}}{k_m} \frac{N_{\mathrm{MISS},m}}{r_m} \\ &= C_{\mathrm{FP}} (1 - \mathrm{precision}(m)) + \frac{C'_{MISS}}{k_m} (1 - \mathrm{recall}(m)) \ . \end{aligned}$$

The traditional definition of precision for a single query is

$$\mathrm{precision}(m) = \frac{N_{\mathrm{TP},m}}{k_m} = 1 - \frac{N_{\mathrm{FP},m}}{k_m} \ ,$$

and recall is

$$\mathrm{recall}(m) = \frac{N_{\mathrm{TP},m}}{r_m} = 1 - \frac{N_{\mathrm{MISS},m}}{r_m} \ .$$

Fixing the costs C_{FP} and C_{MISS} and minimizing Eq. (16) therefore corresponds to maximizing a specific weighted combination of precision and recall.

Lastly, to assess performance of the full meta-visualization the cost needs to be averaged over all plots (queries) m, which yields the mean precision and recall of the meta-visualization.

A.2 Connection between Precision and Recall and Kullback-Leibler Divergences

We now show that the divergences used in our meta-visualization cost function (Eq. 5) are generalizations of precision and recall. In detail, we show that in the above-discussed simple case of "binary neighborhoods" between plots, the divergence $D_{KL}(u_m, v_m)$ reduces to recall; the proof that $D_{KL}(v_m, u_m)$ reduces to precision is similar. Let m again be the plot of interest, let U_m be the set of relevant neighbor plots m according to the information retrieval based comparison measure discussed in Section 3.1, let r_m be the size of U_m, let V_m be the set of neighbor plots based on their physical locations on the meta-visualization display, and let k_m be the size of V_m.

In the probabilistic model of neighborhoods between plots, the binary neighborhoods can be interpreted as follows. We define that the relevant neighbor plots of the plot of interest m have an equal non-zero probability of being chosen, and all other plots have a near-zero probability of being chosen. That is, we define

$$u^*_{m'|m} = \begin{cases} a_m \equiv \frac{1-\delta}{r_m} , & \text{if plot } m' \text{ is in } U_m \\ b_m \equiv \frac{\delta}{M-r_m-1} , & \text{otherwise.} \end{cases}$$

Here M is the total number of plots, and $0 < \delta \ll 0.5$ gives the irrelevant plots a very small probability. Similarly, we define the probability of choosing a neighbor from the visualization as

$$v^*_{m'|m} = \begin{cases} c_m \equiv \frac{1-\delta}{k_m} , & \text{if plot } m' \text{ is in } V_m \\ d_m \equiv \frac{\delta}{M-k_m-1} , & \text{otherwise.} \end{cases}$$

Consider the Kullback-Leibler divergence $D_{KL}(u^*_m, v^*_m)$ for any fixed m. We now show that minimizing this divergence is equivalent to maximizing recall where point m is the query. The divergence is a sum over elements $u^*_{m'|m} \log \frac{u^*_{m'|m}}{v^*_{m'|m}}$, thus the sum can be divided into four parts depending on which value $u^*_{m'|m}$ takes (two possibilities) and which value $v^*_{m'|m}$ takes (two possibilities). We get

$$\begin{aligned} & D_{KL}(u^*_m, v^*_m) \\ = & \sum_{m' \neq m, u^*_{m'|m}=a_m, v^*_{m'|m}=c_m} \left(a_m \log \frac{a_m}{c_m} \right) + \sum_{m' \neq m, u^*_{m'|m}=a_m, v^*_{m'|m}=d_m} \left(a_m \log \frac{a_m}{d_m} \right) \\ + & \sum_{m' \neq m, u^*_{m'|m}=b_m, v^*_{m'|m}=c_m} \left(b_m \log \frac{b_m}{c_m} \right) + \sum_{m' \neq m, p^*_{m'|m}=b_m, q^*_{m'|m}=d_m} \left(b_m \log \frac{b_m}{d_m} \right) \\ & = \left(a_m \log \frac{a_m}{c_m} \right) N_{\mathrm{TP},m} + \left(a_m \log \frac{a_m}{d_m} \right) N_{\mathrm{MISS},m} \\ & \quad + \left(b_m \log \frac{b_m}{c_m} \right) N_{\mathrm{FP},m} + \left(b_m \log \frac{b_m}{d_m} \right) N_{\mathrm{TN},m} \end{aligned}$$

where after the last equality sign the terms inside parentheses are simply constant coefficients. Here $N_{\mathrm{TP},m}$ is the number of true positives for this query, that is, the number of plots for which the neighborhood probability is high both according to the comparison measure and according to physical locations on the meta-visualization. The number of misses, that is, the number of plots that have a low neighborhood probability according to physical locations on the meta-visualization even though their neighborhood probability according to the comparison measure is high, is $N_{\mathrm{MISS},m}$. The number of false positives (which have high neighborhood probability according to physical locations but low probability according to the comparison measure) is $N_{\mathrm{FP},m}$. Lastly the number of true negatives (which have low neighborhood probability both according to the physical locations and according to the comparison measure) is $N_{\mathrm{TN},m}$.

It is straightforward to check that if δ is very small, then the coefficients for the misses and false positives dominate the divergence. This yields

$$\begin{aligned} D_{KL}(u_m^*, v_m^*) &\approx N_{\mathrm{MISS},m}\left(a_m \log \frac{a_m}{d_m}\right) + N_{\mathrm{FP},m}\left(b_m \log \frac{b_m}{c_m}\right) \\ &= N_{\mathrm{MISS},m}\frac{1-\delta}{r_m}\left(\log \frac{(M-k_m-1)}{\delta} + \log \frac{(1-\delta)}{r_m}\right) \\ &+ N_{\mathrm{FP},m}\frac{\delta}{M-r_m-1}\left(\log \frac{\delta}{M-r_m-1} - \log \frac{(1-\delta)}{k_m}\right) \\ &= N_{\mathrm{MISS},m}\frac{1-\delta}{r_m}\left(\log \frac{(M-k_m-1)}{r_m} + \log \frac{(1-\delta)}{\delta}\right) \\ &+ N_{\mathrm{FP},m}\frac{\delta}{M-r_m-1}\left(\log \frac{k_m}{M-r_m-1} - \log \frac{(1-\delta)}{\delta}\right) . \end{aligned} \quad (17)$$

Because the terms $\log[(1-\delta)/\delta]$ dominate the other logarithmic terms, Eq. (17) further simplifies to

$$\begin{aligned} D_{KL}(u_m^*, v_m^*) &\approx \left(N_{\mathrm{MISS},m}\frac{1-\delta}{r_m} - N_{\mathrm{FP},m}\frac{\delta}{M-r_m-1}\right)\log \frac{(1-\delta)}{\delta} \\ &\approx N_{\mathrm{MISS},m}\frac{1-\delta}{r_m}\log \frac{(1-\delta)}{\delta} = \frac{N_{\mathrm{MISS},m}}{r_m}C \end{aligned}$$

where C is a constant that only depends on δ and not on m. Hence if we minimized this cost function, we would be maximizing the recall of the query, since the definition of recall is

$$\mathrm{recall}(m) = \frac{N_{\mathrm{TP},m}}{r_m} = 1 - \frac{N_{\mathrm{MISS},m}}{r_m} .$$

We can analogously show that for any fixed i, minimizing $D_{KL}(v_m^*, u_m^*)$ is equivalent to maximizing precision of the corresponding query.

Because minimizing $D_{KL}(v_m^*, u_m^*)$ and $D_{KL}(u_m^*, v_m^*)$ are equivalent to maximizing precision and recall respectively, and u_m and v_m can be seen as continuous-valued stochastic generalizations of u_m^* and v_m^*, we interpret $D_{KL}(v_m, u_m)$ and $D_{KL}(u_m, v_m)$ as generalizations of precision and recall. In the meta-visualization cost function (Eq. 5), the sum $\sum_m D_{KL}(v_m, u_m)$ over the query plots m then generalizes average precision and the sum $\sum_m D_{KL}(u_m, v_m)$ generalizes average recall. Therefore the total cost of information retrieval errors, which was shown in A.1 to be equivalent to a weighted sum of precision and recall, is generalized as the weighted sum of divergences in Eq. (5).

B Comparison between Our Meta-visualization Approach and an Ordered Scatter Plot Matrix

In Section 4.1 we compared our meta-visualization approach with a standard scatter plot matrix, in terms of their ability to keep known matching pairs of scatter plots (plots showing the same neighborhood relationships) close-by. In Section 4.1, our meta-visualization approach clearly outperformed the scatter plot matrix.

The most common way to create scatter plot matrices is to order the rows and columns of the matrix according to the order of feature indices in the original data. For this reason the scatter plot matrix in Section 4.1 uses this common way to order the rows and columns of the scatter plot matrix.

The locations of plots on the scatter plot matrix depend on the ordering of the rows and columns of the matrix; in principle, the performance of the scatter plot matrix could suffer from a poor original ordering of the feature indices as it would yield a poor ordering of the rows and columns. A *data-driven ordering* of the rows and columns could in principle yield better results. To our knowledge, no advanced data-driven ordering methods have been suggested so far for arranging plots within a scatter plot matrix. However, it could possible to arrange the

rows (and correspondingly the columns) so that the most related dimensions would be next to each other. However, this is already a novel method which has not been published to our knowledge.

In this appendix we now propose a new data-driven method to order the rows and columns of a scatter plot matrix. We then compare our meta-visualization approach against this new data-driven ordered scatter plot matrix, using the same data and setting as in Section 4.1. We show that even when such advanced ordering of the rows and columns is used, our meta-visualization approach still outperforms the scatter plot matrix.

We aim to order the features so that two highly correlated features, that is, two features whose values are highly positively correlated across the samples in the data set, should be given close-by feature indices. It turns out that creating such an ordering can be represented as a dimensionality reduction task from the pairwise matrix of feature-to-feature correlations to one-dimensional indices of the features.

Given a set of input data samples $\{\mathbf{x}_i\}_{i=1}^N$ we compute the sample Pearson correlation coefficient between features k and l in the standard fashion as

$$r_{kl} = \frac{\sum_{i=1}^N (x_{ik} - \mu_k)(x_{il} - \mu_l)}{\sqrt{\sum_{i=1}^N (x_{ik} - \mu_k)^2}\sqrt{\sum_{i=1}^N (x_{il} - \mu_l)^2}} \tag{18}$$

where x_{ik} is the kth feature value of sample $\mathbf{x}_i$ and $\mu_k = \frac{1}{N}\sum_{i=1}^N x_{ik}$. We then convert the pairwise feature correlations to pairwise distances between features as

$$d_{kl} = \sqrt{2(1 - r_{kl})} \tag{19}$$

where the resulting distances are in the range $[0, 2]$ with distance zero indicating the two features are strongly correlated and distance 2 indicating the features are strongly negatively correlated.[5]

The resulting matrix of the pairwise distances d_{kl} between all features can be used for dimensionality reduction to create an ordering of the features. We apply multidimensional scaling (MDS) on the pairwise distance matrix to obtain a one-dimensional embedding where each feature k has some one-dimensional embedding coordinate s_k. We then order the rows and columns of the scatter plot matrix in ascending order of s_k. Since two highly correlated features k and l have a small distance d_{kl}, the MDS embedding aims to preserve these distances and gives the features similar one-dimensional embedding coordinates s_k and s_l; thus the two features are plotted in nearby rows (and columns) in the ordered scatter plot matrix. The resulting scatter plot matrix has a data driven order where correlated features are shown near each other.

We create this ordered scatter plot matrix for the face pose image data set of Section 4.1. The result is shown in Fig. 13. As described in Section 4.1, in this data set there are 10 plots (feature pairs) that have a matching other plot (feature pair), such that both plots show the exact same information up to rotation. Although there is some structure visible in the new scatter plot matrix between the features, the known matching plots are still not shown nearby on the matrix: none of the matching pairs are in the 8-neighborhood of each other. Thus, our meta-visualization approach whose result is shown in Fig. 4 (left) achieves better results compared to the scatter plot matrix, regardless of whether the original order of feature indices is used or whether the advanced data-driven feature ordering is used.

$V = (v_1, \cdots, v_N)$. Since the ordering of V reflects the correlations between the features to some extent,

References

Agrafiotis DK (2003) Stochastic proximity embedding. Journal of Computational Chemistry 24(10):1215–1221

[5] We also created a version of the distance matrix where we use absolute values $|r_{kl}|$ of correlations in Eq. (19) so that features are considered similar if they are either positively or negatively correlated. For the face pose image data that we use in this experiment, the resulting ordered scatter plot matrix has very similar performance regardless of whether we use correlations r_{kl} or their absolute values to create the distances; for brevity we only show the results based on the plain correlations r_{kl} without absolute values.

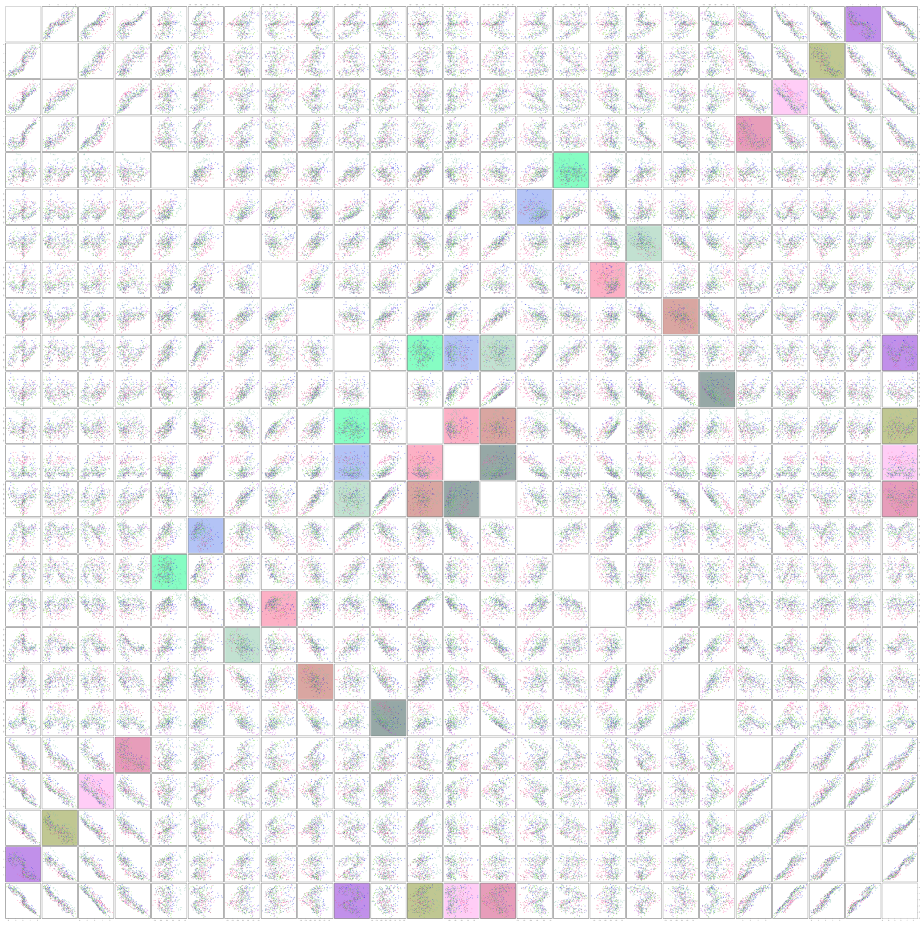

Fig. 13: The scatter plot matrix for face pose image data, based on a data-driven feature reordering (reordering of the rows and columns) to keep highly correlated features in nearby rows (and correspondingly in nearby columns). Each of the mini-plots is a scatter plot showing an individual feature pair; in each mini-plot, faces are shown as dots colored by person identity. For this data 10 plots m have a matching other plot m' where both plots show the exact same information up to rotation. (Each plot in row i, column j also has a trivial match in the transposed cell, row j, column i.) The nontrivial matching plots are shown with background in the same color; it would be very difficult to notice the non-trivial matches from the scatter plot matrix, even though a data-driven reordering of rows and columns has been used to create the scatter plot matrix. Compare to Fig. 4 where our meta-visualization approach successfully places the matching plots near each other.

Asimov D (1985) The grand tour: A tool for viewing multidimensional data. SIAM Journal on Scientific and Statistical Computing 6(1):128–143

Belkin M, Niyogi P (2002) Laplacian eigenmaps and spectral techniques for embedding and clustering. In: Advances in Neural Information Processing Systems 14, MIT Press, pp 585–591

Bertini E, Tatu A, Keim D (2011) Quality metrics in high-dimensional data visualization: An overview and systematization. Visualization and Computer Graphics, IEEE Transactions on 17(12):2203–2212

Borg I, Groenen P (2005) Modern Multidimensional Scaling: Theory and Applications. Springer

Bradham C, McClay DR (2006) p38 MAPK in development and cancer. Cell Cycle 5(8):824–828

Caldas J, Gehlenborg N, Faisal A, Brazma A, Kaski S (2009) Probabilistic retrieval and visualization of biologically relevant microarray experiments. Bioinformatics 25(12):i145–i153

Chang H, Yeung DY, Xiong Y (2004) Super-resolution through neighbor embedding. In: Proceedings of CVPR 2004, the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol 1, pp I–I

Cheung CT, Deisher TA, Luo H, Yanagawa B, Bonigut S, Samra A, Zhao H, Walker EK, McManus BM (2007) Neutralizing anti-4-1BBL treatment improves cardiac function in viral myocarditis. Laboratory Investigation 87(7):651–661

Child D (2006) The Essentials of Factor Analysis. Continuum International

Claessen J, van Wijk J (2011) Flexible linked axes for multivariate data visualization. Visualization and Computer Graphics, IEEE Transactions on 17(12):2310–2316

Cockburn A, Karlson A, Bederson BB (2009) A review of overview+detail, zooming, and focus+context interfaces. ACM Computing Surveys 41(1):2:1–2:31

Cook J, Sutskever I, Mnih A, Hinton G (2007) Visualizing similarity data with a mixture of maps. In: Proceedings of AISTATS 2007, International Conference on Artificial Intelligence and Statistics, JMLR W&CP 2, vol 2, JMLR, pp 67–74

Donoho DL, Grimes C (2003) Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data. Proceedings of the National Academy of Sciences 100(10):5591–5596

Franco D, Campione M (2003) The role of Pitx2 during cardiac development. Linking left-right signaling and congenital heart diseases. Trends in Cardiovascular Medicine 13(4):157–163

Frustaci A, Chimenti C, Ricci R, Natale L, Russo MA, Pieroni M, Eng CM, Desnick RJ (2001) Improvement in cardiac function in the cardiac variant of Fabry's disease with galactose-infusion therapy. The New England Journal of Medicine 345(1):25–32

Gourier N, Hall D, Crowley JL (2004) Estimating Face Orientation from Robust Detection of Salient Facial Features. In: Proceedings of Pointing 2004, ICPR, International Workshop on Visual Observation of Deictic Gestures

Guan N, Tao D, Luo Z, Yuan B (2011) Non-negative patch alignment framework. IEEE Transactions on Neural Networks 22:1218–1230

Hinton G, Roweis S (2003) Stochastic neighbor embedding. In: Advances in Neural Information Processing Systems 15, MIT Press, pp 833–840

Hotelling H (1933) Analysis of a complex of statistical variables into principal components. Journal of Educational Psychology 24:417–41,498–520

Kehrer J, Hauser H (2013) Visualization and visual analysis of multifaceted scientific data: A survey. IEEE Transactions on Visualization and Computer Graphics 19(3):495–513

Lafon S, Lee A (2006) Diffusion maps and coarse-graining: a unified framework for dimensionality reduction, graph partitioning, and data set parameterization. IEEE Transactions on Pattern Analysis and Machine Intelligence 28(9):1393–1403

Lawrence ND (2004) Gaussian Process Latent Variable Models for Visualisation of High Dimensional Data. In: Advances in Neural Information Processing Systems 16, MIT Press, Cambridge, MA

Lee JA, Lendasse A, Verleysen M (2004) Nonlinear projection with curvilinear distances: Isomap versus curvilinear distance analysis. Neurocomputing 57(0):49 – 76

Lowe SW, Lin AW (2000) Apoptosis in cancer. Carcinogenesis 21(3):485–495

van der Maaten L (2009) Learning a parametric embedding by preserving local structure. In: Dyk DAV, Welling M (eds) Proceedings of AISTATS 2009, International Workshop on Artificial Intelligence and Statistics, JMLR W&CP 5, JMLR, pp 384–391

van der Maaten L, Hinton G (2008) Visualizing data using t-SNE. Journal of Machine Learning Research 9:2579–2605

van der Maaten L, Postma E, van der Herik J (2009) Dimensionality reduction: A comparative review. Tech. rep., Tilburg centre for Creative Computing, Tilburg University

Nguyen GP, Worring M (2008) Interactive access to large image collections using similarity-based visualization. Journal of Visual Languages and Computing 19(2):203–224

Parkinson HE, Kapushesky M, Kolesnikov N, Rustici G, Shojatalab M, Abeygunawardena N, Berube H, Dylag M, Emam I, Farne A, Holloway E, Lukk M, Malone J, Mani R, Pilicheva E, Rayner TF, Rezwan FI, Sharma A, Williams E, Bradley XZ, Adamusiak T, Brandizi M, Burdett T, Coulson R, Krestyaninova M, Kurnosov P, Maguire E, Neogi SG, Rocca-Serra P, Sansone SA, Sklyar N, Zhao M, Sarkans U, Brazma A (2009) Arrayexpress update - from an archive of functional genomics experiments to the atlas of gene expression. Nucleic Acids Research 37(Database-Issue):868–872

Patwari N, III AOH (2004) Manifold learning algorithms for localization in wireless sensor networks. In: Proceedings of ICASSP 2004, International Conference on Acoustics, Speech, and Signal Processing, IEEE, pp III–857–III–860

Peltonen J, Georgatzis K (2012) Efficient optimization for data visualization as an information retrieval task. In: Proceedings of MLSP 2012, the 2012 IEEE International Workshop on Machine Learning for Signal Processing, IEEE, electronic proceedings

Peltonen J, Kaski S (2011) Generative modeling for maximizing precision and recall in information visualization. In: Gordon G, Dunson D, Dudik M (eds) Proceedings of AISTATS 2011, the Fourteenth International Conference on Artificial Intelligence and Statistics, JMLR W&CP 15, JMLR, vol 15, pp 597–587

Peltonen J, Lin Z (2013) Information retrieval perspective to meta-visualization. In: Ong CS, Ho TB (eds) Proceedings of ACML 2013, Fifth Asian Conference on Machine Learning, JMLR W&CP 29, JMLR, pp 165–180

Peng W, Ward MO, Rundensteiner EA (2004) Clutter reduction in multi-dimensional data visualization using dimension reordering. In: Proceedings of INFOVIS '04, the IEEE Symposium on Information Visualization, IEEE Computer Society, pp 89–96

Pópulo H, Lopes JM, Soares P (2012) The mTOR signalling pathway in human cancer. International Journal of Molecular Sciences 13(2):1886–1918

Porcherie A, Mathieu C, Peronet R, Schneider E, Claver J, Commere PH, Kiefer-Biasizzo H, Karasuyama H, Milon G, Dy M, Kinet JP, Louis J, Blank U, Mecheri S (2011) Critical role of the neutrophil-associated high-affinity receptor for IgE in the pathogenesis of experimental cerebral malaria. The Journal of Experimental Medicine 208(11):2225–2236

Ramsay AK (2010) Validation of the MEK5 and ERK5 pathway as targets for therapy in prostate cancer and analysis of the erk5 signalling complex. Md thesis, University of Glasgow

Robinson A, Weaver C (2006) Re-visualization: Interactive visualization of the process of visual analysis. In: Proceedings of the GIScience Workshop on Visual Analytics & Spatial Decision Support 2006, electronic proceedings

Roweis S, Saul L (2000) Nonlinear dimensionality reduction by locally linear embedding. Science 290:2323–2326

Sammon JW (1969) A nonlinear mapping for data structure analysis. IEEE Transactions on Computers 18(5):401–409

Schölkopf B, Smola AJ, Müller KR (1999) Kernel principal component analysis. In: Advances in kernel methods: support vector learning, MIT Press, pp 327–352

Sharma A, Paliwal KK (2007) Fast principal component analysis using fixed-point algorithm. Pattern Recognition Letters 28:1151–1155

Sikachev P, Amirkhanov A, Laramee RS, Mistelbauer G (2011) Interactive algorithm exploration using meta visualization. Tech. rep., Institute of Computer Graphics and Algorithms, Vienna University of Technology, Favoritenstrasse 9-11/186, A-1040 Vienna, Austria

Tatu A, Albuquerque G, Eisemann M, Schneidewind J, Theisel H, Magnor MA, Keim DA (2009) Combining automated analysis and visualization techniques for effective exploration of high-dimensional data. In: Proceedings of IEEE VAST 2009, the IEEE Symposium on Visual Analytics Science and Technology, IEEE, pp 59–66

Tatu A, Maas F, Farber I, Bertini E, Schreck T, Seidl T, Keim D (2012) Subspace search and visualization to make sense of alternative clusterings in high-dimensional data. In: Proceedings of IEEE VAST 2012, the IEEE Conference on Visual Analytics Science and

Technology, IEEE, pp 63–72
Tenenbaum JB, de Silva V, Langford JC (2000) A Global Geometric Framework for Nonlinear Dimensionality Reduction. Science 290(5500):2319–2323
Tipping ME, Bishop CM (1999) Probabilistic principal component analysis. Journal of the Royal Statistical Society, Series B 61:611–622
Venna J, Kaski S (2007) Comparison of visualization methods for an atlas of gene expression data sets. Information Visualization 6(2):139–154
Venna J, Peltonen J, Nybo K, Aidos H, Kaski S (2010) Information retrieval perspective to nonlinear dimensionality reduction for data visualization. Journal of Machine Learning Research 11:451–490
Vesanto J (1999) SOM-based data visualization methods. Intelligent Data Analysis 3:111–126
Viau C, McGuffin MJ (2012) Connectedcharts: Explicit visualization of relationships between data graphics. Computer Graphics Forum 31(3pt4):1285–1294
Vladymyrov M, Carreira-Perpinan M (2014) Linear-time training of nonlinear low-dimensional embeddings. In: Proceedings of AISTATS 2014, International Conference on Artificial Intelligence and Statistics, JMLR W&CP 33, JMLR, pp 968–977
Waters JP, Pober JS, Bradley JR (2013) Tumour necrosis factor and cancer. The Journal of Pathology 230(3):241–248
Weaver C (2006) Metavisual exploration and analysis of DEVise coordination in Improvise. In: Proceedings of CMV '06, the Fourth International Conference on Coordinated & Multiple Views in Exploratory Visualization, IEEE Computer Society, pp 79–90
Weinberger K, Saul L (2006) Unsupervised learning of image manifolds by semidefinite programming. International Journal of Computer Vision 70(1):77–90
Weinberger KQ, Packer BD, Saul LK (2005) Nonlinear dimensionality reduction by semidefinite programming and kernel matrix factorization. In: Proceedings of AISTATS 2005, the 10th International Workshop on Artificial Intelligence and Statistics
Wickham H, Hofmann H (2011) Product plots. IEEE Transactions on Visualization and Computer Graphics 17(12):2223–2230
Wismüller A, Verleysen M, Aupetit M, Lee JA (2010) Recent advances in nonlinear dimensionality reduction, manifold and topological learning. In: Proceedings of ESANN 2010, European Symposium on Artificial Neural Networks - Computational Intelligence and Machine Learning, d-side, pp 71–80
Wong PC, Bergeron RD (1997) 30 years of multidimensional multivariate visualization. In: Scientific Visualization: Overviews, Methodologies & Techniques, IEEE Computer Society Press, pp 3–33
Xu C, Tao D, Xu C (2013) A survey on multi-view learning. CORR abs/13045634 Available at http://arxiv.org/abs/1304.5634
Yan S, Xu D, Zhang B, Zhang H, Yang Q, Lin S (2007) Graph embedding and extensions: A general framework for dimensionality reduction. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(1):40–51
Yang Z, Peltonen J, Kaski S (2013) Scalable optimization of neighbor embedding for visualization. In: Proceedings of ICML 2013, the 30th International Conference on Machine Learning, JMLR W&CP 28, JMLR
Zhang T, Tao D, Li X, Yang J (2009) Patch alignment for dimensionality reduction. IEEE Transactions on Knowledge and Data Engineering 21(9):1299–1313
Zhang Z, Zha H (2004) Principal manifolds and nonlinear dimensionality reduction via tangent space alignment. SIAM Journal on Scientific Computing 26(1):313–338
Zhou T, Tao D, Wu X (2011) Manifold elastic net: a unified framework for sparse dimension reduction. Data Mining and Knowledge Discovery 22(3):340–371