Interactive High-Dimensional Data Analysis Using The "Three Experts"

Georg Albrecht, Alex Pang; University of California at Santa Cruz; Santa Cruz, California

Abstract

With the increasing availability of data from various domains such as health care, finance, social networks, etc. there is a need to provide analytic tools that are more accessible to lay people. In this paper, we present a software tool which can be used to aid inexperienced users in understanding high dimensional data. To facilitate the understanding of such data, we place special emphasis on how the data is presented, using the "Three Experts". The Three Experts display shows the results of three different dimension reduction techniques, similar in notion to seeking several expert opinions on a particular topic. This will help the user to discern between pertinent structures in the data, and those resulting from distortions inherent in dimension reduction. The second emphasis is on providing the ability for users to insert and manipulate new points within the data, as well as observe high dimensional trajectories to convey positional displacement of points. Observing these changes will enable the user to develop an actionable intuition for the data in question. These methods could be used in any field where high dimensional numeric data needs to be analyzed, with potential benefits for both novice and expert users.

Introduction

The continued proliferation of complex data sets across various domains has created a demand for both automatic and human-in-the-loop methods that can be used to extract actionable information. As the amount of accessible data continues to increase, so will the demand for new and improved methods with which to process this data. For most high-dimensional data sets, domain experts can make use of various software tools which enable the observation, inspection, and analysis of such data. Visual data mining tools with interactive presentation and query capabilities allow domain experts to quickly examine complex data by interacting with multivariate visual displays. In addition, researchers are realizing that visual feedback has a role to play in the data mining process, as well as the analysis of the results. The ability to create a good mental model of how high dimensional data is structured is essential if end users expect to develop a sound understanding of the data. Unfortunately, many of these software tools are intended for use by professionals who are likely already familiar with high dimensional data. However, there are many situations in which users, who are otherwise unfamiliar with high dimensional data, could benefit from exposure to, and exploration of, such data.

In this paper we present a software tool designed to help make high dimensional data understandable to users who are inexperienced or unfamiliar with such information. This is done by providing a simplified visual analytics platform to enable exploration of, and interaction with, a particular data set in a controlled fashion. There are two novel aspects to this software: the "Three Experts" display and the ability to insert and manipulate new data in a directly observable way.

The Three Experts display provides the user with three views of the data. These views show the differing results of three different dimension reduction techniques. Because a user may not be familiar with high dimensional data, much less dimension reduction techniques, these views are meant to act as "expert's interpretation" of the data. Most feature extraction based dimension reduction techniques work by emphasizing some measurable relationship between points to transform the data into a form which can be represented using fewer attributes (dimensions). However, because of this transformation the resulting data may exaggerate particular features, while understating others. So while the overall structure of the data will remain intact, there will likely be some features which are unique to the particular transformation used. By comparing the Three Expert views, a user can quickly become cognizant of which features (such as groupings, spacings, or outliers) are consistent among the views, and which are not. While a user inexperienced with high dimensional data would be more concerned with consistent overall structure, an knowledgeable user will be able to identify smaller features indicative of data worth further investigation.

The second important aspect of the software is the exploratory features made available to the user. These exploratory features allow the user to gain an intuition for the data by directly observing the effects of their interactions through the dimension reduction techniques. To this end, the user is able to create a new data point, based on attribute values specified by the user. Once created, the user can observe the new point's position within the data set. In addition, the user can view the movement of this unique point resulting from manipulation of the point's attribute values. Finally, the user is also able to create trajectories which help to show the changes necessary to re-position a point.

The main motivation for this research is in the field of health care. Such a system could enable doctors and patients to determine potential courses of prevention or treatment of a condition, based on the specific attributes of the patient. Moreover it would allow a patient to view their position, as a new point based on their unique attributes, relative to others with a similar condition, spanning a range from afflicted to healthy. The user can then manipulate attribute values, which represent aspects of their health and lifestyle, and see how these changes affect the position of their data point relative to the original data set. Furthermore, by using the trajectory feature, the patient would be able to observe the combination of changes necessary for them to reach a targeted desirable state of health.

The remainder of the paper is organized as follows: Prior work done in several related areas is discussed in section 2. Sec-

tion 3 discusses the main interface design considerations and provides an overview of the software. Section 4 offers a short case study by demonstrating potential usage with a well known data set. Section 5 addresses some shortcomings as well as areas for improvement, and section 6 provides concluding remarks.

Previous Works

Humans are not directly capable of visualizing information or structured objects in more than three dimensions without some form of abstraction or manipulation. In an effort to overcome this hindrance much research has been done to make higher dimensional data visually perceivable and understandable. The use of glyphs [1] can be used to provide a notion of higher dimensions, but are imprecise and only serve to convey a few additional dimensions. Techniques such as parallel coordinates [12] and star coordinates [13] work reasonably well but can suffer from clutter and compaction when a larger number of dimensions is being used. Scatterplot matrices [3] are another viable alternative, but they too can become unwieldy with a high number of dimensions, especially to a novice user.

When dealing with high dimensional data (as opposed to structural objects or shapes), the high number of attributes can make working with, or visualizing, the data an unwieldy and resource intensive task. It should be noted that the term "attribute" is representative of a corresponding dimension within the data. To transform the data into a more amicable form, some form of dimension reduction is often used. Sometimes this involves feature selection, where selected features are kept while others are excluded, creating a subset of the most indicative attributes. Another form is called feature extraction. As the number of dimensions increases, so does the likelihood for measurable relationships between the various attributes.

Many such dimension reduction methods exist [2] [8]. Each method is designed to take into account some particular measurable aspect of the data in question, and are often better suited for particular kinds of data. Our current software implementation makes use of Principal Component Analysis [6], Independent Component Analysis [4], and Multi-Dimensional Scaling [14], which are discussed further below. However, regardless of the method used, some aspects of the data will be emphasized, while other will inherently be lost. The meaning of the emphasized and lost information depends on the particular method used. With our software we hope to provide a straight forward method to highlight these discrepancies by providing a visually qualitative comparison.

There are many software packages available which facilitate visual data analysis for high dimensional data. Examples of these include Tableau [18], Microsoft Business Analytics [16], ggobi [9], and XmdvTool [5]. Research in how to effectively visualize and interact with high dimensional data is still ongoing. For example, "ClusterSculptor" by Nam et.al. [17], which allows for interactive tuning of clustering parameters, principal components, and other aspects of the data. Another method, called Visual Hierarchical Dimension Reduction (VHDR) [20][19], works by grouping similar dimensions to create a hierarchy. Lower dimensional spaces can then be produced based on the clusters derived from the hierarchy. Throughout the process the user is able to modify most steps of the process if desired. However, these systems do not allow for the insertion and interaction with user specific data points, nor do they provide an interactive way to show the changes necessary to displace various points within the data.

User Interface Design

The main goal of this project was to create an interface which will enable an otherwise inexperienced user to gain an intuitive understanding of high dimensional data. To achieve this requires the proper balance of exposing the user to the complexity inherent within the data, and taking care not to overwhelm them with intricate details. The user will likely not benefit from exposure to the procedures used to process and present the data. However, if the user is to gain an actionable intuition for the data, controlled exposure to the defining aspects of this complexity is imperative. Finding the appropriate balance between shielding and exposure is what makes this task challenging.

With this in mind, we formulated the following functional requirements as necessary to properly facilitate the exploration and interaction with of high dimensional data. These requirements are *a)* provide three visual interpretations of the data, each using different dimension reduction techniques; *b)* show the relative distributions of the data on a per attribute basis; *c)* allow the creation and modification of new user specified data points; and *d)* allow the creation of trajectories to demonstrate the changes necessary to displace a point from its current position, to a new user specified target position. These four functional requirements work together to enable controlled exploration of, and interaction with, the data. Each helps provided information about the salient features of both the individual attributes, and the data set as a whole.

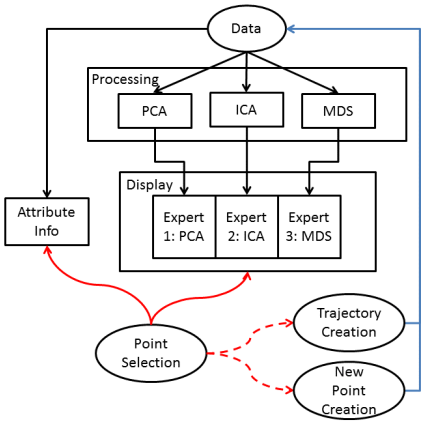


Figure 1: The basic software architecture. Black lines indicate the data flow within the program, red lines indicate user selections (dashed lines are optional), and blue lines indicate modifications to the data set. The data is processed using three dimension reduction techniques, which are displayed as three "Expert Interpretations". The user has three forms of interaction: Point selection, which is linked to the Attribute Info histograms; Point Creation, which inserts a new point into the data set; and Trajectory creation, which shows the changes necessary to displace a point to a desired target position.

In general, the interface is broken into four distinct groupings of components. These groupings are consistent with the four functional requirements stated above. An example of the default interface can be seen in Fig. 2. In this figure, the three data dis-

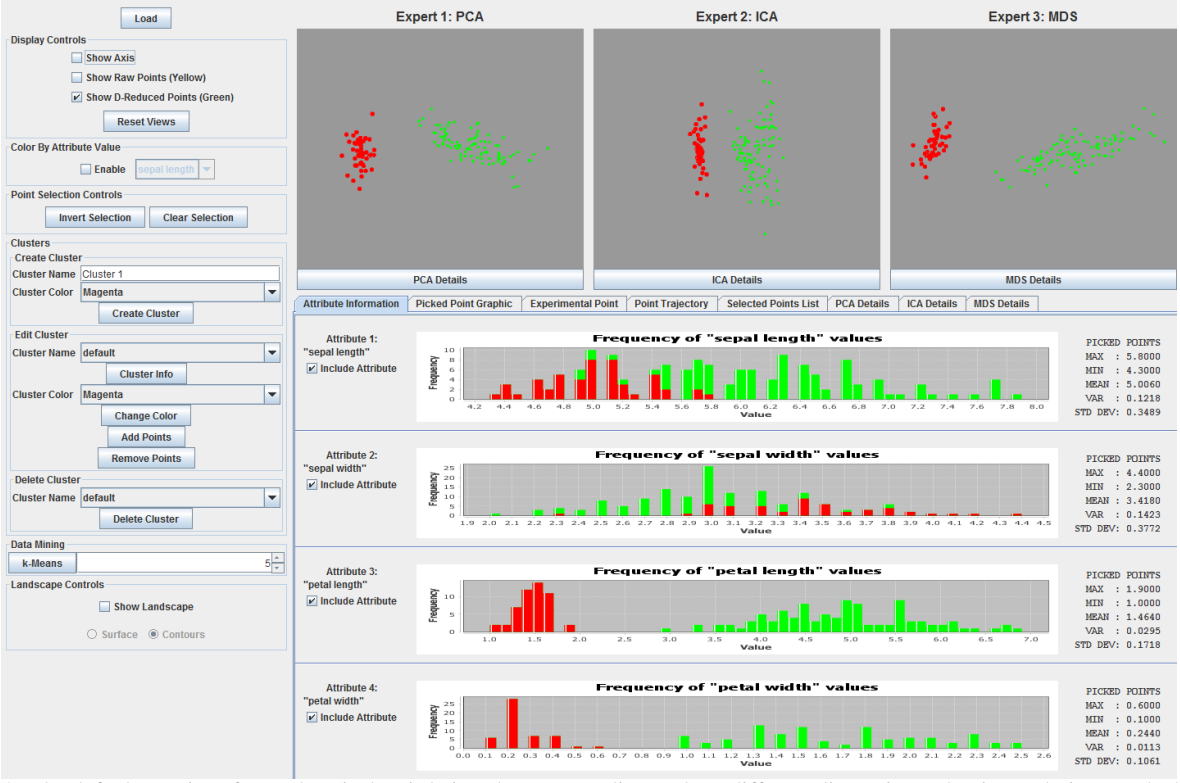


Figure 2: The default user interface. The Iris data is being shown according to three different dimension reduction techniques. The bottom tab shows a histogram indicating the distribution of attribute values. The "include attribute" check boxes allow the user to dynamically toggle attributes being considered. A selection of points in the data set is shown in red. The histograms below are linked to the selection and show the distribution of the attribute values for the selection.

plays showing the expert interpretations are seen in the top right. These displays are always visible to the user, regardless of the actions being taken. The bottom consists of a tabbed pane with each tab labeled according to a particular task or form of information it shows. The default visible panel shows the basic information for each attribute present in the data. The interface itself has been constructed with components which should be familiar to anyone who has used a computer. There are no custom interface components, thus helping to promote a baseline affordance for new users, who will not need to spend extra time learning new interface components. The following sections will provide more discussion of the various functional requirements, as well as their implementation details.

The Three Experts

Once a data set of interest has been loaded, the user is shown three "expert interpretations" of the data. These interpretations correspond to three different dimension reduction methods: Principal Component Analysis (PCA), Independent Component Analysis (ICA), and Multi-Dimensional Scaling (MDS). Since the user will likely not be familiar with (or interested in) the methods used, the representations are instead referred to as "expert interpretations". This choice of nomenclature is used to suggest the notion of seeking a second or third opinion by consulting alternative experts in a particular field. The three specified methods were chosen because of their perceived popularity when low dimension (in our case 3D) visualization of higher dimensional data is required. In addition, each method uses a fundamentally different technique, discussed below, in order to achieve the desired reduction in dimensions. While these three methods are in the current implementation, other dimension reductions could be used as desired. Regardless of the methods used, any form of dimension reduction will have the implicit side effect of distorting of the data.

Therefore the purpose of presenting the results of three different dimension reduction methods is to help mitigate potential misconceptions about the data as a result of the distortion (e.g. bias of an expert). Observation of the similarities and differences between the three views will show the user how open to interpretation various structures are. Each form of dimension reduction works to represent the data in a 2D space while retaining its important characteristics, but each uses an alternate approach by considering different aspects of the data. For example, PCA seeks to preserve variability, ICA attempts to uncover maximally independent sub-components inherent in the data, while MDS works to maintain the distances between each point. More details about these methods are provided below.

With Principal Component Analysis (PCA) [6] one is able to combine potentially correlated attributes into fewer linearly uncorrelated attributes, called principal components. These principal components serve as the new orthogonal axis of the lower dimensional representation, and can be ranked according to the variability present along each newly defined axis. In addition, the resulting principal components are represented by the weighted

contributions from the original attributes. While information loss is inherent, because the principal components are rankable, the trade off between this loss and the reduction in dimensions can be optimized.

Independent Component Analysis (ICA) [4] [10] attempts to determine the latent independent factors which together comprise the data being analyzed. ICA is often used to separate linearly mixed sources (sometime called blind source separation). The technique attempts to estimate the original signals when the given input is a result of mixing multiple independent data sources. While PCA is constrained by the need for the components to be orthogonal, ICA's constraint focuses on the independence of the underlying structures. This results in components which maximize their statistical independence and non-Gaussianity from one another. Like PCA, the resulting independent components are also formed by weighted contributions from the original attributes. The particular implementation used was the FastICA algorithm [11].

If one is primarily concerned with visualizing the data, and less concerned with the formation of an underlying representative model, then Multi-Dimensional Scaling (MDS) [14] is a possible solution. The goal of MDS is to provide a low dimensional visual representation of data which maintains the distances observed among the points in higher dimensions. To achieve this, MDS uses a function minimization algorithm which evaluates different configurations of points in an attempt to minimize the disparity in distance between the original and lower dimensional representations.

The third method, Multi-Dimensional Scaling (MDS) [14], produces a representation of the data in a lower dimension that maintains the relative distances observed among the points in higher dimensions. That is to say, MDS aims to place each point in a lower dimensional space such that the relative distances between each point is preserved as much as possible. To achieve this, MDS uses a function minimization algorithm which evaluates different configurations of points in an attempt to minimize the disparity in distance between the original and lower dimensional representations. Because the algorithm only considers the distance between points, there are several important factors to be aware of. As the axes of the resulting lower dimensional representation are not scaled, they become arbitrary, and may change as the data changes. Because of this, the orientation of the resulting display is arbitrary, and may also change as a result of changes in the data set. In addition, while the formation and spacing of larger distances between clusters will be well represented, the tighter spacing of points within clusters will be less accurately represented.

The resulting Expert Interpretations should be able to capture the most salient structures, but will likely differ in the representation of smaller features. This allows the user to visually filter the structures within the data. If similar clusters, shapes, or spacing, are present in the majority of views, then it serves as a good indication these structures are inherent in the data and may be worth investigating further. However, if there is a seemingly indicative cluster, spacing, or outlier present in only one of the views, then it is more likely an artifact of a particular dimension reduction technique, and not a feature present within the data. To a more experienced user this may hold particular value. Such a deviation may be indicative of a unique circumstance which will have a different meaning depending on the dimension reduction technique used for that particular Expert Opinion.

Data Inspection

Two of the key functional requirements for this user interface were the ability to view the distribution of the data, as well as user specified subsets, and the ability to create and insert a modifiable point into the data set.

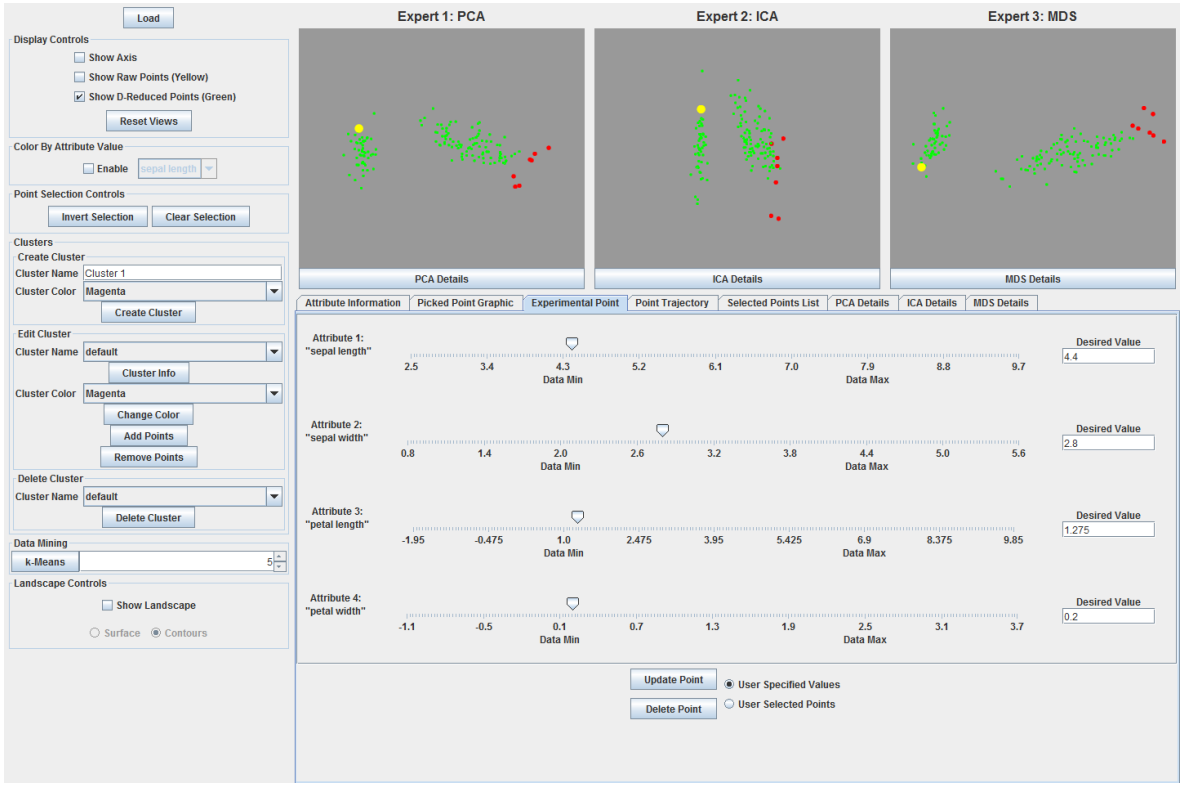
The inspection ability is provided by the default Attribute Information tab, shown in Fig. 2. This shows the user the range over which the attributes values span, as well as the distribution of values across this range in the form of a histogram. The user also has the ability to select a subset of points within the expert displays for closer inspection. Points can be selected individually, or by dragging a selection box over points of interest. The selected points are then highlighted across all three expert displays, and are linked to the attribute histograms. Thus, the user can quickly see the variation and distribution of the subset of interest along each attribute, both within the selection, and relative to the entire data set. In addition, the user is allowed to disable individual attributes as they see fit (though at least two must be enabled for display purposes). The user can then perform all the same tasks as if the data only consisted of the selected attributes.

A potential artifact when forming lower dimensions is that points, which are spatially separated at higher dimensions, may project to the same location in lower dimensional space. To help the user recognize instances of high point density, we included the ability to display a density map, using either contours or a landscape, for each expert. The density map will convey to the user the obvious point concentrations. More importantly with inspection the user can identify areas that may appear to contain only a few points when in fact there are many, or areas that appear to contain many points, when in fact there are only very few. Upon selection of one of these seemingly single points, all the points sharing that location in the low dimension space will be distinguishable by their variations in higher dimensions, shown in the per attribute display.

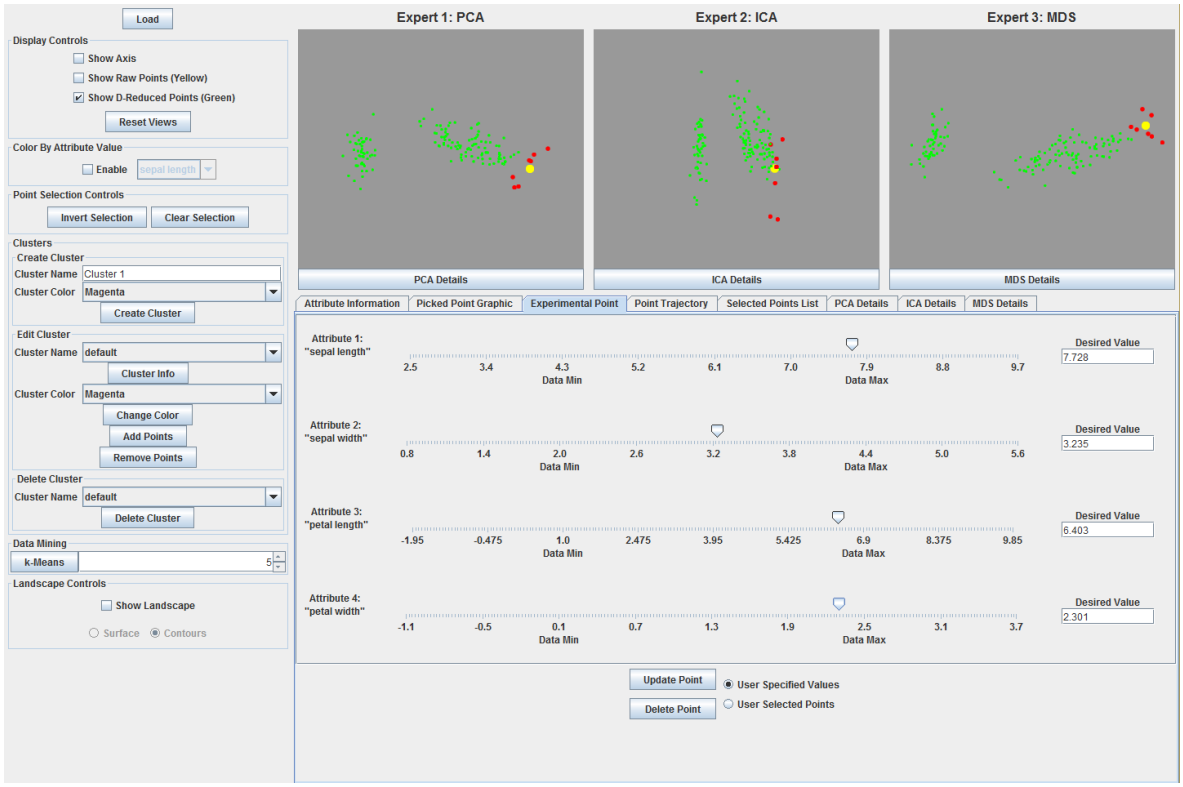
Additional useful features include the ability to color data according to attribute values, and to create custom clusters. Attribute coloring helps users to identify groupings within denser concentrations of points with similar values, as well as verifying distinct features. Custom cluster creation can be used to organize points under consideration across the multiple views, as well as also verifying possible features.

Data Creation and Interaction

Allowing the user to create and interact with their own unique data point is paramount. This point creation and interaction is enabled by the Experimental Panel, shown in Fig. 3. Initial values for each attribute can be specified in one of three ways: The user can specify the desired numeric values themselves, create a copy of a preexisting point, or create a new point which is representative of the average value of a specified cluster. Numerically specified attribute values can be used to test specific predetermined values, such as an arbitrary what-if scenario or, as in the prior described medical application, as a representation of a patient. This method requires the user to enter the desired numeric value for each attribute, and then press the Create/Update button to create the new point. The other options, requiring the preselec-

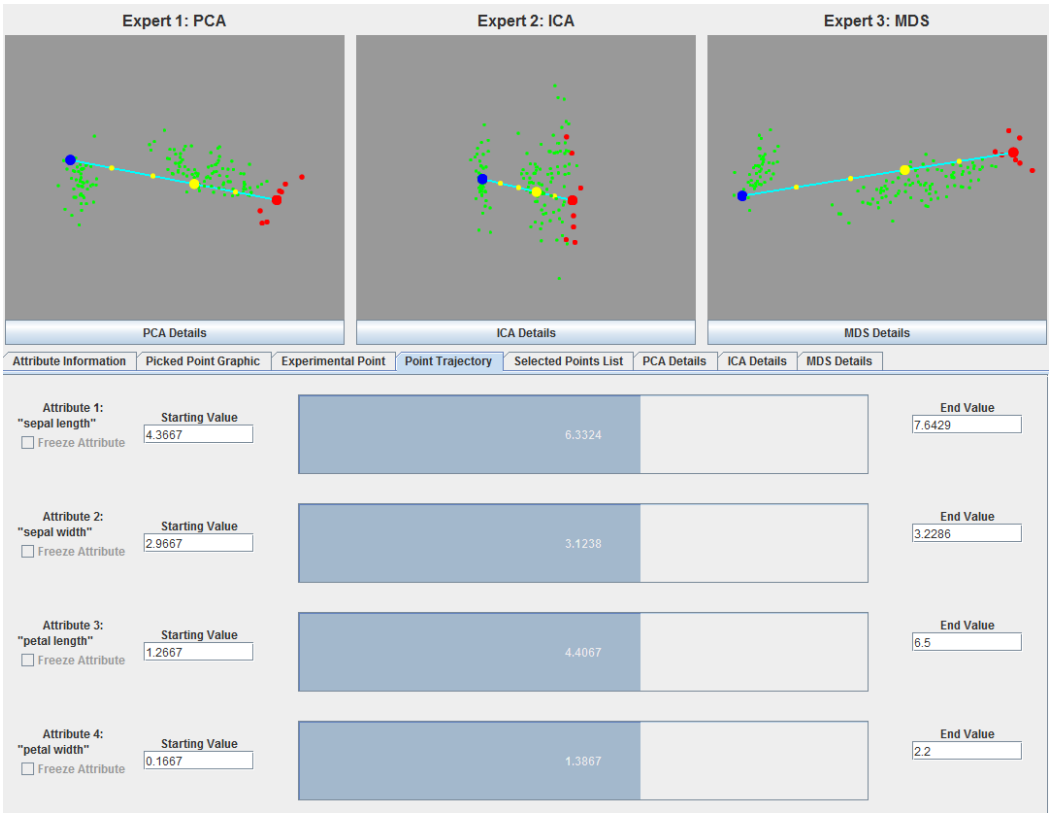


(a) Newly created experimental point shown in yellow with the target cluster in red.

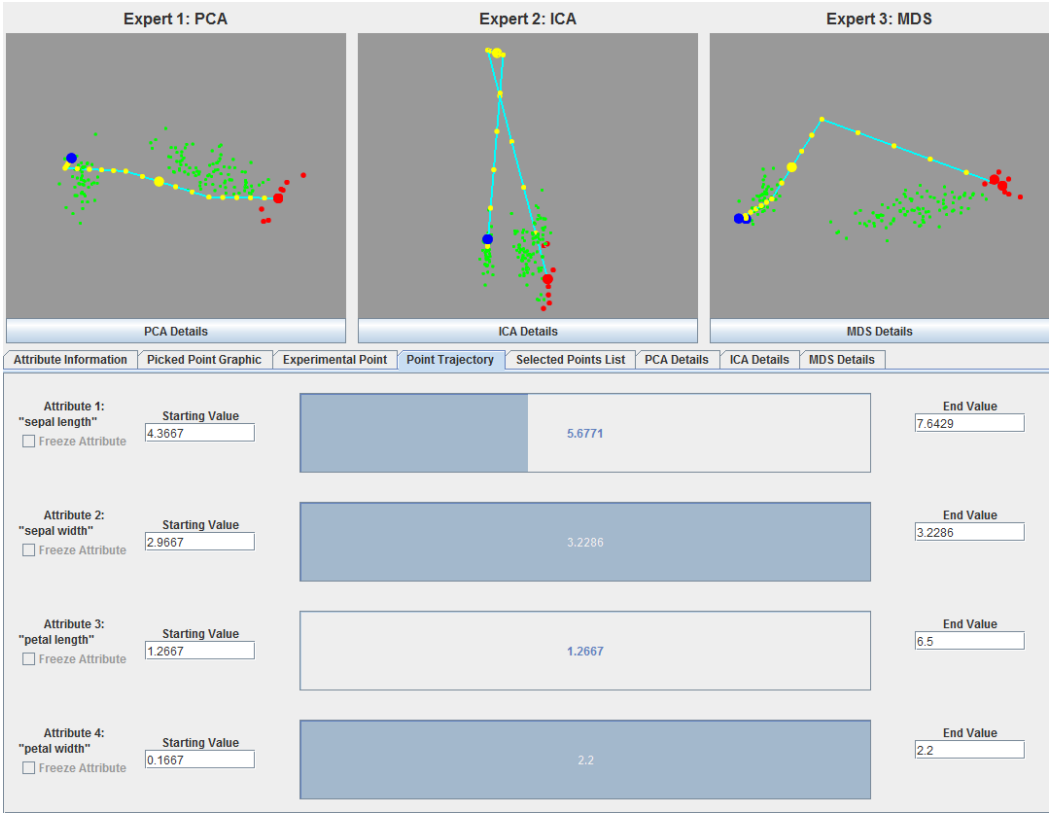


(b) After changing attribute values, the experimental point is now among target cluster.

Figure 3: After creating a unique point (shown in yellow) the user can manipulate the attribute values and observe the points movement within the high dimensional space in real time. 3a shows the unique point in yellow and the target points in red. After using the sliders to change the values of various attributes, 3b shows the unique point now among the target cluster.



(a) Linear trajectory.



(b) Shortest-step-first trajectory.

Figure 4: The point trajectory interface. The Iris data is shown in green according to three different dimension reduction techniques. The top 4a shows linear interpolation trajectory, while the bottom 4b shown the shortest-step-first interpolation trajectory. The interface shows the start and end points of the trajectory, as well as the progression along each attribute as the user steps through the trajectory. In each view the large blue point is the starting point, the large yellow point is the current location, and the large red point represents the target location.

tion of a point or cluster, will more likely be used to further investigate some aspect of the data itself. For this investigation the user must first select a point or set of points within the data set, specify with the appropriate radio button that selected points are being used, and then click the Create/Update button. If the user selected a single point, then the values will be replicated, resulting in the experimental point being a copy of the selected point. If multiple points were selected, the attribute values for the new point will be derived from the average attribute values of the selected points. Once all necessary attribute values have been specified, the new point can be created and inserted. To insert a newly created point involves applying the three dimension reduction techniques to the newly-expanded data set and displaying the respective results.

Furthermore, once the user specified point has been created, the user is also able to manipulate the various attribute values as they see fit. Manipulating these values will allow the user to directly observe the degree to which changes along a particular attribute will affect the position of the experimental point within the original data set. This manipulation is done either by moving the value slider or by entering a new numeric value for the appropriate attribute. As changes are made, the position of the experimental point is updated in real time across all three expert displays. Though computationally expensive for some experts (MDS do not allow for dynamic addition of points and so must be recomputed), this real-time display enables direct feedback for the user.

This direct visual feedback will help the user to get a sense for the spatial relation between the attributes and the data set itself. After some experimentation, the user will be able to discern how the various attributes contribute to the position of the points in the data set. With time the user should be able to develop a base qualitative understanding of the high dimensional data. In the context of the previously discussed medical application, the user would enter their pertinent medical information to create a new point with attribute values based on their own unique health characteristics. Once displayed, the user would then see where their point lies in space relative to the data set of healthy and unhealthy people. From here, the user could modify various personal health related attributes and observe how these changes alter the position of their unique point. Hopefully, the user will see and better understand what alterations to their current state would be necessary to reach a position indicative of improved health.

Point Trajectory

However, the changes necessary for a user created point to reach a desired position may not be straightforward to the user. To help overcome this obstacle, the user can activate a form of guided interpolation along a trajectory. This guided interpolation will show the user how the values of each attribute will need to change in order to displace a point from its current position to a user specified target position. Both the start and end points can be specified in the same manner as the initial experimental point. The user can either specify them numerically, or as separately selected points or grouping of points. Some attributes, such as age or sex, may not be strictly modifiable and so should not be considered when creating a trajectory. The user may want to disable these attributes, or any others they deem modifiable, in the attribute information panel. This will exclude the disabled attributes from the interpolation trajectory.

The user has a choice of the type of interpolation used to create the trajectory: linear or shortest-step-first. As implemented, linear interpolation works by independently performing a discrete number of linear interpolation steps along each attribute. The resulting points along the trajectory represent a simultaneous progression across all attributes, as can be seen in Fig. 4a. This progression results in a straight line between the start and end points in high dimensional space. However it may be difficult for the user to keep track of the changes along each attribute simultaneously. To help overcome this the user may instead make use of shortest-step-first trajectory. Shortest-step-first interpolation performs a discrete number of interpolation steps along each attribute individually, as shown in Fig. 4b. The greedy nature of this form of interpolation is due to the algorithm always choosing to interpolate along the attribute with the smallest absolute change in value between its start and end points. Because the shortest-step-first interpolation focuses on progressing one attribute at a time, the final trajectory will be composed of as many line segments as there are attributes. The initial segments will be short, representing the smaller change in attribute value, while the later segments will be longer, representing the larger changes necessary.

After activating the trajectory of their choice, the user is able to step through the intermediate points along the trajectory. The current point is identifiable as the largest step marker along the displayed trajectory, and will change as the user progresses along it. As the user continues stepping along the trajectory, the progression of the currently changing attributes is shown in their respective progress bars. This provides the necessary feedback to allow the user to easily follow the changes along the various attributes as they occur. For a linear interpolated trajectory the progression will occur evenly across all attributes, whereas for shortest-step-first interpolation the progression will occur one attribute at a time.

Usage Examples

In the previous sections, we provided an overview of the software and an in-depth description of the important features. We will now describe a brief-walk through using two different data sets, and making use of different features provided by the software. The first will use the canonical Iris data set [7], and the second will use the most recent Fortune 500 list [15].

Iris

The Iris data set includes 150 samples, each containing four attributes. The attributes are measurements of the length and the width of the sepals and petals of three different Iris species. Fig. 2 shows the software with the Iris data set displayed. The histograms displayed under the attribute information tab are linked to the point selections made by the user. Once a point or group of points is selected, the distribution of the selected point(s) is shown on a per attribute basis. In the prior figure, the user has selected a cluster of points, which are shown in red in both the expert displays and the linked histograms.

For the sake of example, assume a user decides to insert a new experimental point, perhaps based on values from a newly-collected specimen. To create this point, the user selects the Experimental Point tab. Here the attribute values can be entered by typing the values directly, by adjusting the value sliders, or by using the mean attribute values of a user-selected point or points. Once the values for the respective attributes have been entered,

the new point is inserted into the original data set and the three dimension reduction methods are applied. An example of the newly created point can be seen in Fig. 3a as the larger yellow point amid the smaller green points within each expert view. From here the user can make changes to the various attribute values. Altering these values with the sliders will allow the user to observe the resulting displacement in real time, the results of which are shown in Fig. 3b. These results indicate that to reach the target grouping requires relatively large changes to attributes 1, 3, and 4 (sepal length, petal length and width), but a much smaller change to attribute 2 (sepal width). The real-time displacement serves to convey to the user how each attribute contributes to the position of the data point. The user can attempt to move the experimental point to a particular target area; however this may not lead to the intended result. Because there are many attributes which affect the position of a point, there may be many attribute value combinations that would result in visual proximity of the experimental point to the target area.

However, since visual proximity alone does not necessarily guarantee that all attribute values are similar to the surrounding data points, the user can make use of the trajectory generator. The trajectory generator will display a path between the specified start and end positions. The user is then able to advance incrementally along this trajectory until the desired target is reached. Both the attribute values for the current trajectory step and a progress bar display to the user the progression along the trajectory. Fig. 4a shows a partial advancement along a linearly interpolated trajectory. As previously mentioned, the linearly interpolated trajectory is calculated by dividing the distance between the start and end value of each included attribute from the original data space by an equal number of steps. This calculation leads to a straight line trajectory. While easy to visualize, this trajectory may be of limited value if the user finds it difficult to keep track of all the simultaneous changes along each attribute. Instead it may be beneficial for the user to utilize the shortest-step-first trajectory, shown in Fig. 4b. The shortest-step-first trajectory is created by interpolating along the attributes one at a time. The attributes are ordered according to the lowest absolute change necessary to reach the target value. In other words, the shortest-step-first interpolation algorithm selects the attributes of least resistance first, under the assumption that first attribute which undergoes modification may be considered the easiest change to make, and so possibly the most desirable. The results of these trajectories show that, in this case, the target position is very similar to the position reached using manual attribute manipulation above.

Fortune 500

The Fortune 500 is a list of the top 500 U.S. corporations ranked according to their gross revenue. This list is updated annually by Fortune magazine. In its pure form, the data contains 22 attributes. However only 13 of these are beneficial for this demonstration. These included attributes are comprised of numeric values relating to the rank of the company on the list, as well as its revenue, profits, number of employees, or market value, and boolean values, such as whether the current CEO is the founder, foreign, or a woman.

The excluded attributes were either textual or redundant. In addition, entries with missing fields were excluded.

Fig. 5 shows that, unlike the Iris data, the three experts each

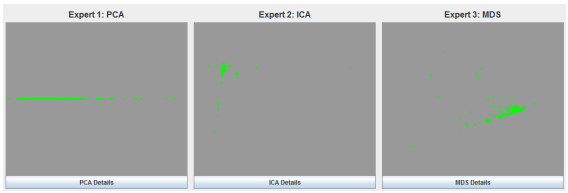


Figure 5: The Fortune 500 dataset with rankings included in the processed data.

produce noticeably different interpretations of the Fortune 500 data. Here, the first expert (PCA) shows an extremely linear structure to the data. By coloring the points according to their Fortune 500 ranking, it becomes clear that this ranking is a definitive attribute for this linear structure. Since the rank of the company is not strictly part of the data about each company, but a result of the data, we will exclude the rank attribute from the data. Instead we will color the points according to their ranking. The resulting plots are shown in Fig. 6, where blue corresponds to a higher position on the list, and red is a lower position (the higher the ranking the lower the rank number: rank 1, blue, is the top and rank 500, red, is the bottom). Now some similarities between the three expert views become visible. Each view has a single large concentration of points, with the surrounding area quickly becoming sparse. In addition we can see that the top companies (blue) are clearly removed from the others in all three expert views. Upon closer inspection, these companies can be split into two groupings, shown as the magenta and cyan clusters in Fig. 7. The formation of these two distinct clusters can be attributed to the amount of assets owned. The magenta cluster is comprised entirely of finance, investment, and insurance companies, whereas the cyan cluster is composed entirely of non-financial companies. This clustering can be attributed to the financial companies owning more interest bearing assets such as investments and loans, which other non-financial companies won't have.

Another grouping of interest, shared somewhat among the three views, is shown in blue in Fig. 8. All of the companies in this grouping were not profitable in 2014. This grouping is easily separable because there are several attributes which can correlate with profitability. However, there are two non-profitable companies which have a noticeable distance from the main grouping of non-profitable companies. Upon further inspection of the various attributes, we find that the outlying attribute is the assets owned by the company.

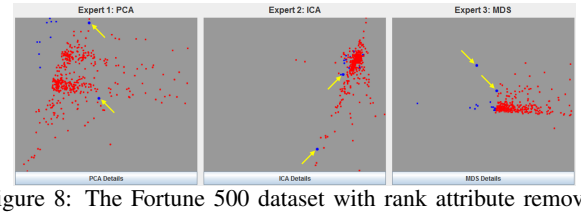


Figure 8: The Fortune 500 dataset with rank attribute removed and colored according to whether profitable (red) or not (blue). There are two blue outliers from the main cluster. Further inspection shows these two companies (Amazon.com and Target) have a significantly larger value of assets compared to the rest of the non-profitable companies.

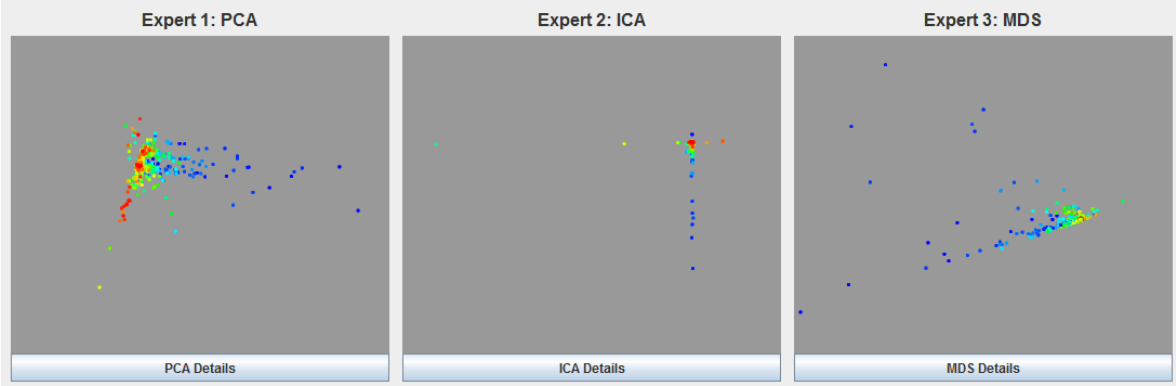


Figure 6: The Fortune 500 dataset with data points colored according to rank. Some similar groupings and separations are evident, especially for the highly ranked companies in blue.

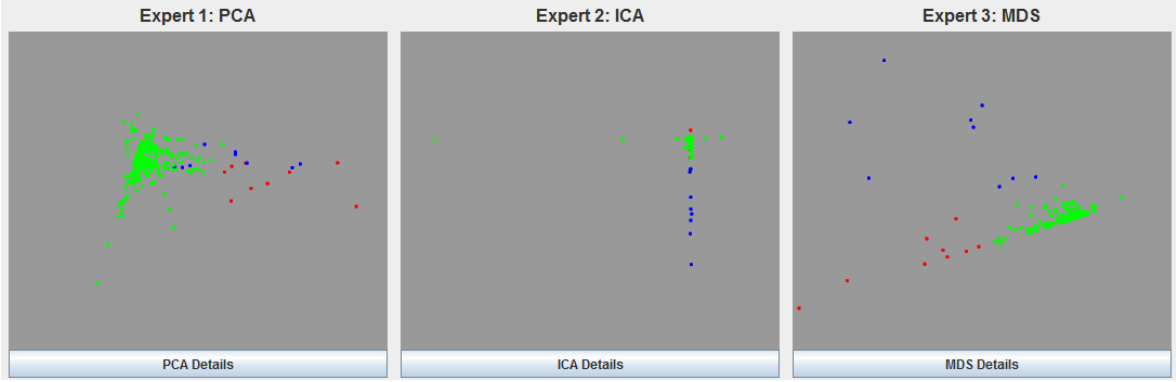


Figure 7: Fortune 500 dataset. Fig. 6 showed a separation of highly ranked companies. Further exploration shows a separation between these high ranking companies, particularly clean in the MDS view. The separation between financial companies (blue) and non-financial companies (red) can be attributed to the relatively high valuation of assets held by the financial companies.

Limitations and Future Work

Though the current software implementation does provide the main tools we set out to implement, there are several limitations, as well as improvements which can be made. It should be noted that this software is only intended to be used with continuous numeric data. Nominal data is not supported as it does not lend itself to the dimension reduction or trajectory generation techniques used. Discrete stepwise or ranked data, though usable, will not work well with the current implementation, as the trajectory generation does not currently account for numeric attributes which are representative of rank-intrinsic or stepwise values.

Another issue stems from the nature of some of the dimension reduction techniques used. Both PCA and ICA produce an underlying representative model based on attribute weights, which can be dynamically applied to a new point. MDS does not produce such a model. Instead, MDS need to be rerun every time a point is added or modified. While this brute force method works well for smaller data sets, it can cause noticeable delays when larger data sets are used. However, the dimension reduction methods used can easily be exchanged for others as needed.

Conclusion

In this paper we have described the design of a software interface intended to provide unfamiliar users the ability to explore high dimensional data. Our intent is that using this interface will enable a novice to develop an actionable intuition for the data in question. The combination of the visualization methods and analysis tools discussed result in a software interface which enables interactive analysis and exploration of high dimensional data. By allowing the user to create and manipulate new data points we enable a novel form of data exploration. With this exploratory point creation, the user is able to directly observe how various attribute values affect the position of the new data point relative to the surrounding data set. In addition, by manipulating the attribute values of this exploratory point, the user can observe the resulting changes in the point's position in real time. These capabilities would be particularly useful when informing a patient of their current health status, based on their unique lifestyle choices, and how various alterations in habits would result in changes to their health. Furthermore, when coupled with the provided interpolation, the user will be shown various trajectories which present the changes necessary to re-position a specified point to a more desirable state. While the motivation behind this system was to aid medical prognostic and diagnostic applications, the ideas discussed are applicable to any high-dimensional data analysis task.

Acknowledgments

We are thankful for the support from NASA/UARC grant # 20150168.

References

[1] David F Andrews. Plots of high-dimensional data. *Biometrics*, pages 125–136, 1972.

[2] Miguel A Carreira-Perpinán. A review of dimension reduction techniques. *Department of Computer Science. University of Sheffield. Tech. Rep. CS-96-09*, 9:1–69, 1997.

[3] William C Cleveland and Marylyn E McGill. *Dynamic graphics for statistics*. CRC Press, Inc., 1988.

[4] Pierre Comon. Independent component analysis, a new concept? *Signal processing*, 36(3):287–314, 1994.

[5] UC Davis. Xmdvtool. http://davis.wpi.edu/xmdv. Acessed: 2014-7-1.

[6] Brian S Everitt and Graham Dunn. *Applied multivariate data analysis*, volume 2. Arnold London, 2001.

[7] Ronald A Fisher. The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7(2):179–188, 1936.

[8] Imola K Fodor. A survey of dimension reduction techniques, 2002.

[9] The GGobi Foundation. Ggobi. www.ggobi.org. Acessed: 2014-7-1.

[10] Aapo Hyvärinen, Juha Karhunen, and Erkki Oja. *Independent component analysis*, volume 46. John Wiley & Sons, 2004.

[11] Aapo Hyvärinen and Erkki Oja. A fast fixed-point algorithm for independent component analysis. *Neural computation*, 9(7):1483–1492, 1997.

[12] Alfred Inselberg and Bernard Dimsdale. Parallel coordinates. In *Human-Machine Interactive Systems*, pages 199–233. Springer, 1991.

[13] Eser Kandogan. Visualizing multi-dimensional clusters, trends, and outliers using star coordinates. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 107–116. ACM, 2001.

[14] Joseph B Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29(1):1–27, 1964.

[15] Fortune Magazine. Fortune 500, 2015.

[16] Microsoft. Microsoft business analytics. http://www.microsoft.com/en-us/server-cloud/audience/business-analytics.aspx. Acessed: 2014-7-1.

[17] Eun Ju Nam, Yiping Han, Klaus Mueller, Alla Zelenyuk, and Dan Imre. Clustersculptor: A visual analytics tool for high-dimensional data. In *Visual Analytics Science and Technology, 2007. VAST 2007. IEEE Symposium on*, pages 75–82. IEEE, 2007.

[18] Tableau Software. Tableau. www.tableausoftware.com. Acessed: 2014-7-1.

[19] Jing Yang, Wei Peng, Matthew O Ward, and Elke A Rundensteiner. Interactive hierarchical dimension ordering, spacing and filtering for exploration of high dimensional datasets. In *Information Visualization, 2003. INFOVIS 2003. IEEE Symposium on*, pages 105–112. IEEE, 2003.

[20] Jing Yang, Matthew O Ward, and Elke A Rundensteiner. Visual hierarchical dimension reduction for exploration of high dimensional datasets. 2002.

Author Biography

Alex Pang is a Professor of Computer Science at UC Santa Cruz. He received his PhD in Computer Science from UCLA in 1990, and his BS in Industrial Engineering from University of the Philippines with magna cum laude in 1981. His research interests are in comparative and uncertainty visualization, flow and tensor visualization, and collaborative visualization. His research has been supported by various funding agencies such as NSF, ONR, DARPA, DOE, LANL, and NASA, as well as industrial partners such as Sun and HP. He served as an associate editor of the IEEE Transactions on Visualization and Computer Graphics, papers co-chair for IEEE Visualization 2006 and 2007, and UCSC Chief Scientist for CITRIS from 2006-2007.

Georg Albrecht received his BS in Computer Science from UC Santa Cruz in 2010, after which he worked at NASA for two years. Following this, he returned to UC Santa Cruz and received his MSc in Computer Science in 2015. Since then, he has continued working at NASA.