Evolutionary Visual Exploration: Evaluation of an IEC Framework for Guided Visual Search

Nadia Boukhelifa, Anastasia Bezerianos, Waldo Cancino, Evelyne Lutton

Evolutionary Visual Exploration: Evaluation of an IEC Framework for Guided Visual Search

N. Boukhelifa nadia.boukhelifa@inria.fr
INRIA, Saclay, France

A. Bezerianos anastasia.bezerianos@lri.fr
Univ Paris-Sud & CNRS (LRI), INRIA, Saclay, France

W. Cancino wcancino@gmail.com
INRIA, Saclay, France

E. Lutton evelyne.lutton@grignon.inra.fr
INRA, Grigon, France

Abstract

We evaluate and analyse a framework for Evolutionary Visual Exploration (EVE) that guides users in exploring large search spaces. EVE uses an interactive evolutionary algorithm to steer the exploration of multidimensional datasets towards two-dimensional projections that are interesting to the analyst. Our method smoothly combines automatically calculated metrics and user input in order to propose pertinent views to the user. In this paper, we revisit this framework and a prototype application that was developed as a demonstrator, and summarise our previous study with domain experts and its main findings. We then report on results from a new user study with a clear predefined task, that examines how users leverage the system and how the system evolves to match their needs. While previously we showed that using EVE, domain experts were able to formulate interesting hypothesis and reach new insights when exploring freely, our new findings indicate that users, guided by the interactive evolutionary algorithm, are able to converge quickly to an interesting view of their data when a clear task is specified. We provide a detailed analysis of how users interact with an evolutionary algorithm and how the system responds to their exploration strategies and evaluation patterns. Our work aims at building a bridge between the domains of visual analytics and interactive evolution. The benefits are numerous, in particular for evaluating Interactive Evolutionary Computation (IEC) techniques based on user study methodologies.

Keywords

Interactive evolutionary algorithms, Interactive evolutionary computation, Genetic Programming, Data mining, Visual analytics, Information visualization.

1 Introduction

Information visualization transforms data into interactive visual representations to amplify human cognition (Card et al., 1999). It typically deals with abstract, nonspatial and high-dimensional data (Chen, 2005).[1] Visualizations can be described as *explanatory* when the aim is to communicate insight already known in the data, or it can be *exploratory* when the focus is on the dynamic discovery process of insights hidden in the data. This process is relatively unpredictable due to the lack of a-priori knowledge of what the user is looking for. The analyst's role in this case is to organise, test, develop

[1] Contrary to its sister domain of Scientific Visualization (SciVis) that traditionally deals with spatial data.

concepts, look for trends, and define hypothesis (Grinstein, 1996). When the search space is large, as it is often the case for multi-dimensional datasets, the task of exploring and finding interesting patterns in the data becomes tedious. Dimension reduction techniques reduce the search space, but often require to specify objective criteria for filtering views *prior* to user exploration. Other techniques (Brown et al., 2012b; Endert et al., 2011) steer the exploration to interesting areas of the search space based on information learnt during the exploration. This seems more adapted to the freeform nature of exploratory visualization, but in many cases, it requires an internal representation of the user or at the very least a mechanism for predicting what the user is interested in.

One way to infer information about users is to examine their interaction logs. Research in user modeling (Fischer, 2001) has shown that much information can be inferred about the user from their interactions, such as their exploration strategies, personality characteristics and cognitive traits (Brown et al., 2014). Research in the field of visual analytics, a research field concerned with building interactive visual interfaces to facilitate analytical reasoning (Thomas and Cook, 2005), showed that feeding knowledge about the user back into the visualization pipeline can have many advantages. For example, it can improve the overall layout of the visualization, reduce the complexity of a model, and help to steer the exploration of large search spaces towards more pertinent views of the data (Endert et al., 2012; Brown et al., 2012b; Boukhelifa et al., 2013).

In previous work, we introduced a framework for Evolutionary Visual Exploration (EVE)[2] that combines visual analytics with stochastic optimisation to aid the exploration of multidimensional datasets. Starting from a set of data dimensions, an Interactive Evolutionary Algorithm (IEA) progressively evolves non-trivial viewpoints in the form of linear and non-linear dimension combinations. These views are built using the classical evolutionary loop of selection of parent dimensions, and the recombination and mutation of these dimensions. The crossover and mutation operators support the exploratory process by introducing users to new views that may help them discover new relationships or interesting structures in their data. The criteria for evolving new dimensions is not known a-priori and is partially specified by the user via an interactive interface. Our method leverages automatic tools to detect interesting visual features and human interpretation to derive meaning, validate the findings and guide the exploration without having to grasp advanced statistical concepts.

The contributions of this article are three fold: (i) a summary of related work in the topic of guided visual exploration; (ii) results from a new user study that examines in detail how users leverage the system and how the system evolves to match their needs when a clear task is specified; and (iii) a detailed methodology for evaluating an EVE system that takes into account both the effectiveness of the underlying algorithm and user behaviour with regards to how they explored the search space and how they evaluated views. This methodology can be applied to other Interactive Evolutionary Computation (IEC) systems.

2 Related Work

Seo and Shneiderman (2005) discuss two different approaches for exploring multidimensional datasets; axis-parallel and non-axis parallel projections, and the tradeoff between the simplicity of the former and the power of the latter. Axis-parallel projection methods use existing dimensions as axes of the projection plane, and thus produce

[2]EVE video and a prototype demo are available at http://www.aviz.fr/EVE

familiar and comprehensible projections. Non-axis parallel projection methods use linear or non-linear combination of two or more dimensions for the axis of the projection plane, and have the advantage of a larger projection space which can potentially reveal structures not visible in the axis-parallel projection space. It is, however, harder to search for useful projections in such an extended space. Furthermore, combined dimensions are not always easy to interpret. In our paper, we use both axis-parallel projections to explore the original dimensions space, and non-axis parallel projection for a more extended search. We hypothesise that this may be useful to users who are, for instance, familiar with PCA (Principle Component Analysis) type of analysis. Throughout the paper we refer to non-axis parallel projections as "combined dimensions".

Related work is organised in three sections: (1) a brief overview of quality metrics used to describe specific properties of data projections, including metrics we use in this work as part of the automatic evaluation of scatterplots; (2) a general introduction to IEC and the role of visualization in this context; and (3) a brief review of work on guided visual exploration from different perspectives (interaction, optimisation and dimensionality reduction).

2.1 Quality Metrics

Faced with the overwhelming possibilities of exploration paths in multidimensional visualization, researchers in the field designed quality metrics that *automatically* evaluate the various projections of the data, in the hope of focusing user search on the most promising views. In a comprehensive survey, Bertini et al. (2011) used the data flow model to classify quality metrics into three types: metrics that draw information from the data space (i.e. data dimensions and values)[3], from the image space (i.e. the views and rendered images presented to the user) or from both. We add to this list, metrics that operate at the user level taking into account both user task and perception.

Amongst metrics calculated at the data space are clustering and outliers. The rank-by-feature framework (Seo and Shneiderman, 2005), for instance, visualises an optimal set of features according to a user selected quality metric such as correlation or uniformity. They use axis-parallel projections to produce 1D or 2D views and colour brightness to denote ranking scores. Amongst image based metrics are scagnostics (Wilkinson and Wills, 2008) which describe measures of interest for pairs of dimensions based on their geometrical appearance on a scatterplot. A mixed metrics approach consists of combining information from the data and image space at the same time. Peng et al. (2004), for example, combine data features such as correlation information, with view features such as axes adjacency, to measure clutter as a result of reordering visualization axes (Bertini et al., 2011).

Amongst perception-based metrics, Albuquerque et al. (2011) attempted to find a quality measure for scatterplots where the goodness value assigned to the visualization is based on human observations from paired comparison studies in which users were asked to analyse clusters and separate labeled classes. Our approach differs in that the "goodness" of a view is defined both by an automated set of measures and user evaluation. In the next section we describe the nine scagnostics measures designed by Wilkinson and Wills (2008) which we use as one of our quality metrics. We use scagnostics to quantify the amount of pattern that exists in a scatterplot. This is the aspect of our system that lends itself to the image-centric quality metrics approach. Additionally, we use a data metric corresponding to the complexity of a proposed dimension and a user metric relating to their satisfaction with a view as will be discussed in section 3.2.

[3]Features drawn from the data such as clusters, outliers and correlations can also be seen in the view.

Scagnostics[4] are based on geometric graphs which are calculated from areas, perimeters and lengths of these graphs. They include nine measures to characterise scatterplots (Fig. 1) and are useful for quickly discovering regularities and anomalies in scatterplot matrices. The underlying algorithm detects different types of point distributions including multivariate normal, log normal, multinomial, sparse, dense, convex and clusters. It does so by binning, detecting outliers, and computing measures based on the following three statistical properties: *shape* for convex, skinny and stringy distributions; *trend* for monotonic distributions; and *density* for skewed, clumpy, outlying, sparse and striated. These measures have proven statistical properties and are computable for moderately large datasets (Wilkinson and Wills, 2008)[5].

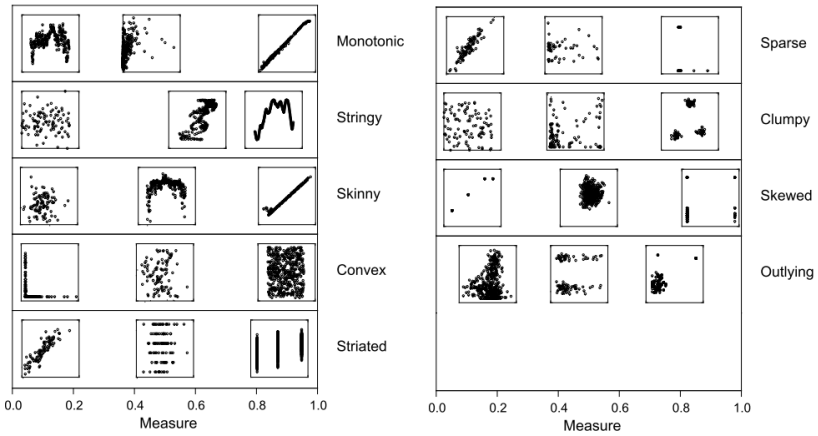

Figure 1: Nine scagnostics measures from (Wilkinson and Wills, 2008).

2.2 IEC and Visualization

Interactive Evolutionary Computation (IEC) corresponds to evolutionary computational models where humans, via suitable user interfaces, play an active role, implicitly or explicitly, in evaluating the outputs evolved by the evolutionary computation. Applications of IEC are varied ranging from art to science (Lutton, 2006; Fukumoto et al., 2010; Takagi, 1998a). IEC lends itself very well to art applications such as for melody or graphic art generation where creativity is essential, due to the subjective nature of the fitness evaluation function. For scientific and engineering applications, IEC is interesting when the exact form of a more generalised fitness function is not known or is difficult to compute, e.g. for producing a visual pattern that would interest a particular user. Here, the human visual system, together with the emotional and psychological responses of the user, can outperform a pattern detection or learning algorithm.

Visualization has been used in IEC both as representation and exploration tools to help users better evaluate the output of interactive evolutionary algorithms (Hayashida and Takagi, 2000; Llorà et al., 2006). The relationship between the visual part and the algorithmic component of the IEC can be characterised using the three-level integration framework of Turkay et al. (2014). In the first level, visualization is mainly a presentation tool (e.g. statistical analysis software such as *R*); the second level refers to semi-

[4]Available as a free downloadable package in R from `http://www.rforge.net/scagnostics/`

[5]Dang and Wilkinson (2014) evaluated the feasability of handling a huge collection of scatterplots in a scagnostics-based system using datasets having up to 3k dimensions and found this to be a bottleneck. We tested scagnostics for datasets up to 12 dimensions (Boukhelifa et al., 2013) and found that we were more limited by the size of the display to accommodate new scatterplots than the algorithms we used.

interactive methods (Johansson and Johansson, 2009; Perer and Shneiderman, 2009) where user interactions are typically limited to parameter tuning or altering the "data domain", for instance via data filtering and aggregation; and the third level refers to tight integration where the coupling is achieved in a seamless and flexible way (Nam and Mueller, 2013; Ingram et al., 2010).

The tight coupling of the visual and algorithmic parts of an IEC is difficult to achieve despite efforts to design good user interfaces, as human interaction with these systems usually raises several issues, mainly linked to the "user bottleneck" problem (Poli and Cagnoni, 1997), human fatigue and slowness. Various solutions have been considered (Poli and Cagnoni, 1997; Takagi, 1998b; Banzhaf, 1997) such as reducing the population size (micro-EAs), constraining the search space to focus on a-priori "interesting" areas, and deploying approximated user models (also called *surrogate functions*) to filter obvious bad solutions (Lutton et al., 2005) and only present to the user the most interesting individuals of the population. This model can be learned from past interactions (Lutton et al., 2005). Among the various strategies to address this issue, we choose to use a small population size of suggested dimensions, and to deploy an approximated user model (i.e. the *surrogate function*) based on a series of geometric measures modeled by the scagnostics distributions (Wilkinson and Wills, 2008).

2.3 Guided Visual Search

Chen and Hagen (2010) argue that interaction alone (e.g. zooming and detail on demand) cannot address the challenges posed by complex visualizations, such as the management of large search spaces. Equally, computational tools cannot address these challenges alone as problem solving remains a human activity. There is a growing body of work in the visualization domain and related disciplines to try and address these issues, combining knowledge from the visualization process itself (e.g. user preferences and chosen visualization parameters) with computational analysis. Typically, information, other than the data being visualised, is fed to the visualization pipeline. This information can be topological, statistical, geometrical, semantic or other forms of data captured from interactive and learning algorithms (Chen and Hagen, 2010).

The idea of taking user interactions into account and learning from them is not new, although getting rich information that benefits the user in an exploratory context is challenging. Endert et al. (2011) talk about *semantic interaction*, where the aim is to get meaning from interactions and use this meaning to close the visual analytics sense-making loop. Their Visual to Parametric Interaction technique (V2PI) describes a framework where the sense making visualization pipeline becomes bi-directional and users are embedded in the pipeline: users learn from visualizations and the visualizations adjust to expert judgement (Leman et al., 2013).

Work that focuses more on the computational part includes parameter space exploration and optimisation, and is directly related to ours. Matkovic et al. (2008, 2011), for instance, tried to interactively find an optimal combination of input parameters for a complex diesel engine injection system using visual analysis techniques. In visual parameter space analysis, a typical research goal is the optimisation of the output of model parameters, by identifying reasonable input parameter settings, while keeping the human in the loop (Sedlmair et al., 2014).

Very recently, Behrisch et al. (2014) proposed a feedback-driven framework for user exploration of large multidimensional data, which combines both automatically calculated metrics and user feedback. Their notion of "pertinence" is defined by visual relevance and is learnt interactively using machine learning techniques (naive Bayes

classifier). Their approach is more focused on the smooth integration of computational and interactive analysis. Similarly, Stolper et al. (2014) propose a progressive framework where analytical algorithms are continuously adapted to produce partial results to support what they call "progressive insight".

Work on interactive dimensionality reduction relates to ours. For example, Johansson and Johansson (2009) and Fernstad et al. (2013) combine both user-defined and automatically calculated metrics (namely for correlation, outliers and clusters) in order to filter data dimensions. However, our approach is more interactive in that the user does not have to explicitly specify the weights of the quality metrics themselves, but these are learnt from and during the exploration. In this sense, exploration of data and the underlying model are not separate steps in our case. In the same spirit, Brown et al. (2012a) implemented the "dis-function" tool where multidimensional data is projected onto a 2D scatterplot using Multi-Dimensional Scaling (MDS), and the user can then move incorrectly positioned points to other similar points, in order to reflect his or her understanding of the data. A feature weight optimization is then applied to calculate a new distance function that is used to reproject the data (by applying PCA to a pairwise distance matrix). These three examples are similar to our work, in that they try to overcome the usability issue mentioned by Turkay et al. (2014) and present in most of the intelligent visual analysis tools, where significant statistical literacy and understanding of the underlying computational methods are required.

When it comes to combining IEC with visual exploration, Mouradian et al. (2012) use a genetic algorithm to create projections of multidimensional data. However, unlike our work, their method focuses on generating linear projections only, and tries to preserve, as much as possible, a predefined data quality metric. More similar to our method, Malinchik and Bonabeau (2004) use an IEC to perform exploratory data analysis, combining computational search with human evaluation. They show how IEC can be used to evolve two-dimensional linear projections that bring insight about the data. Similar to "dis-fuction" (Brown et al., 2012a), they used the IEC to evolve a distance function in attribute space in order to produce the most compelling or interesting clusters to the viewer using a parametric clustering algorithm. However, they also focused only on linear combined-dimension projections, and did not conduct a formal user study to evaluate their work.

3 Evolutionary Visual Exploration

Our proposed framework (Ticona et al., 2013; Boukhelifa et al., 2013) combines visual analytics with stochastic optimisation, to aid the exploration of multidimensional datasets characterised by a large number of possible views or projections. Starting from dimensions whose values are automatically calculated by a PCA, an interactive evolutionary algorithm progressively builds (or evolves) non-trivial viewpoints in the form of linear and non-linear dimension combinations, to help users discover new interesting views and relationships in their data. The criteria for evolving new dimensions is not known a-priori and is partially specified by the user via an interactive interface. Pertinence of views is modeled using a fitness function that plays the role of a predictor: (i) users select views with meaningful or interesting visual patterns and provide a satisfaction score; (ii) the system calibrates the fitness function –optimised by the evolutionary algorithm– to incorporate user's input, and then calculates new views.

In order to validate our method, we embedded a genetic engine into an existing scatterplot matrix visualization system (Elmqvist et al., 2008) that manages the various projections of the data. The prototype system interface is described next in section 3.1,

along with the genetic engine and search space as implemented in *EvoGraphDice* (section 3.2). We end this section with a discussion on issues related to diversity management (section 3.3).

3.1 EvoGraphDice

EvoGraphDice uses GraphDice (Elmqvist et al., 2008; Bezerianos et al., 2010) to manage the various projections of the data. Views are organised in a scatterplot matrix (SPLOM) of 2D projections (Fig. 2a). Users can select a view from the SPLOM (highlighted cells have a green background), which is then displayed in the main plot view (Fig. 2b). They can also perform brushing and linking using a lasso tool to select data points, such that the selection and highlighting of data points in one view is reflected in all other views. *EvoGraphDice* displays the dimensions proposed by the IEA as additional rows (and columns) in the SPLOM, ranked by their fitness evaluation value such that the higher the system evaluation, the higher the y-position of the proposed dimensions in the matrix. The system initially displays dimensions returned by a PCA, after which the user can evolve new dimensions by pressing the "evolve" button (Fig. 2d).

The proposed views are displayed in yellow background to differentiate them from other cells where the intensity of the colour is proportional to an a-priori assessment of user interest. Thus, the darker the colour the more pertinent the view. The system provides an initial score (1 to 5) for each new view, but the user can adapt this score using the slider in Fig. 2d. User-evaluated cells are flagged using a small black square to distinguish them from system-evaluated cells. *EvoGraphDice* can be initialised at any time using the "restart" button which resets parameters of the IEA. Users can save views (Fig. 2f) and bring them back into the SPLOM if they have been replaced during the exploration. The current population is also displayed as a table (Fig. 2h), where each row corresponds to a combined dimension described by a mathematical expression and various components of the fitness function such as the scagnostics measures. The user can limit the dimension search space using the widget in Fig. 2i, which results in a system reset similar to precessing the "restart" button. They can also edit an individual using the "dimension editor" in Fig. 2j.

3.2 Search Space and Genetic Engine

EvoGraphDice relies on a genetic programming engine to evolve a population of combined dimensions. We describe below the main components of this genetic engine, which has rather classical features. What is original – and complex – is the user interaction, which is indirect: The objects the genetic engine evolves are not directly evaluated by the user, but through the views they generate in conjunction with the other dimensions. Strictly speaking the algorithm evolves 2D projections in an implicit manner. This has an impact on the learning algorithm used for building the surrogate function[6].

Search Space: The space searched by the genetic engine is the set of all dimensions that can be built by combining the initial dimensions x_i with operators and constants, encoded as trees according to the classical Genetic Programming (GP) framework (Koza, 1992). These combinations can be complex mathematical expressions containing quadratic, exponential or logarithmic terms (the evolved expressions can be any combination using $+$, $-$, $*$, $/$, $(.)^{(.)}$, exp, log operators, real constants and initial dimensions).

[6] In the current version of the tool, we consider views with only one dimension being evolved, but in the future we can easily extend the fitness function to evolve both dimensions of a view. In that case we can talk about "co-evolution".

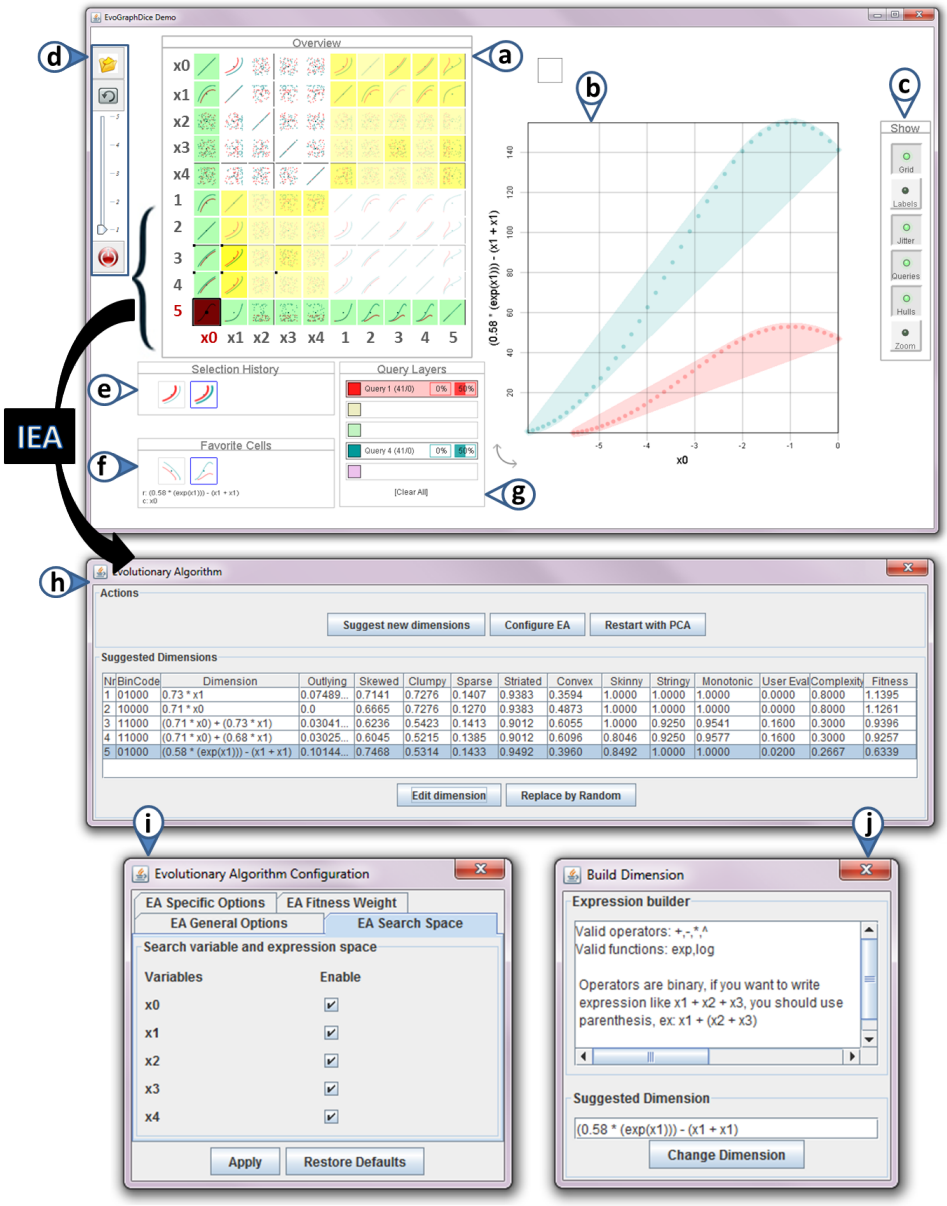

Figure 2: *EvoGraphDice* prototype showing an exploration session of a synthetic dataset. New extensions to the GraphDice system are indicated by filled label arrows. Widgets: (a) an overview scatterplot matrix showing the original dataset of five dimensions (x0..x4) and the new dimensions (1..5) as suggested by the evolutionary algorithm; (b) the main plot view; (c) a tool bar for the main plot view; (d) a tool bar with (top to bottom)"favorite" toggle button, "evolve" button, a slider to evaluate cells and a restart PCA button; (e) the selection history tool; (f) the favorite cells window; (g) the selection query window; (h) The IEA's main control window; (i) a window to limit the search space and set EA parameters; and (j) a dimension editor.

Genetic Engine: We have chosen to evolve a small set of combined dimensions, y_i, in order to allow the user to examine all individuals of the population at a glance over the SPLOM: if n is the number of initial dimensions, a population of another n combined dimensions is evolved. At each iteration, that is each time the user clicks on the "evolve" button, a new generation is produced by applying the selection/crossover/mutation operators, which is then presented to the user whose judgment (evaluation) is explicitly collected via a slider.

Initialisation: A set of a-priori interesting dimensions has been chosen as a starting point. A PCA analysis (Smith, 2002) is performed on the original dataset and the corresponding n linear combinations form the initial population.

The fitness function: Evaluating the fitness of the suggested views requires taking into account user interactions and internal metrics. The user interaction criterion tries to adapt user preferences in the fitness function, while the internal metrics evaluate the relation between variables. The fitness function to be optimised by the genetic engine is a sum of three terms:

1. **A surrogate function** f_{sc}, that plays the role of a predictor, and helps the system to better adapt to user needs. It is based on scagnostics measurements computed for every cell of each combined dimension y_i (the x_j being the initial dimensions), and the corresponding fitness term is a linear combination of the highest values of the scagnostics ($SC_k(y_i, x_j)$) of each scatterplot cell (y_i, x_j) (these cells correspond to the yellow coloured cells of Fig. 2, that is the views of the combined dimensions y_i with respect to the initial dimension x_j):

$$f_{sc}(y_i) = \sum_{k=1..9} w_k(\max_j SC_k(y_i, x_j)) \tag{1}$$

   The weights w_k that govern the relative importance of each of the nine scagnostic measurements are initialised to a uniform weight ($1/9$). Then, as soon as n (the number of original dimensions) interactions are recorded, w_k are updated via a simple multilinear regression on the m past user interactions ($m \geq n$ corresponds to the length of the "memory" of the system).

2. **A Complexity term** that favours dimensions made of a small number of variables and simple mathematical expressions :

$$f_c(y_i) = \left(1 - \frac{nvars(y_i)}{n}\right) \times \frac{1}{depth(y_i)}, \tag{2}$$

   where $nvars(y_i)$ is the number of original variables involved in the mathematical expression of y_i, and $depth(y_i)$ is the depth of the GP tree representing y_i.

3. **A user evaluation term** $f_u(y_i)$, that is an average of the user evaluation for each cell corresponding to y_i (range of 1 to 5 from "bad" to "excellent").

3.3 Diversity management

The evolutionary mechanisms naturally tend to concentrate the population around good solutions. For small population sizes particularly, there is a risk of premature convergence if no diversity preservation is performed. We choose to use a very simple mechanism, similar to the crowding factor scheme (Mengshoel and Goldberg, 2008):

each time a new dimension y'_i is generated by the stochastic operators (mutation or crossover), its distance to each individual of the current population is computed. If y_i is too close to one of the individuals, it is replaced by a random individual. The distance is an euclidean distance on the scagnostics vectors, the ones precisely used for the computation of the surrogate function of equation 1. In other words, this is a phenotypic distance. The distance threshold that governs this mechanism can be tuned by the user (see next section). This allows a full range of exploration/exploitation compromises : if the distance threshold is large we get a quasi random search, while if it is 0, we get a genetic engine with no diversity management.

3.4 Parameters

EvoGraphDice allows users to fully configure the relevant EA parameters as showed in Fig. 2i. These parameters are: *crossover and mutation rates* to set the probabilities for the GP crossover and mutation operators (default 0.5 for both); *replacement rate* refers to the proportion of individuals to be replaced at each EA iteration (default 0.5); *minimal distance* corresponds to the crowding factor for diversity management (default 0.1)[7]; *fitness criteria weights* to tune the weights of each fitness component criterion (by default user evaluation has weight of 2 and complexity 1)[8]; *search variable and dimension space* for restricting the search to a subset of variables specified by the user (by default all dimensions are selected); *data subsets*, used to restrict the set of points of the scatterplot on which the fitness is calculated. The search can thus be performed only on a subset of the data corresponding to a selection query made by the user.

4 Case Studies with Expert Users

This section summarises a user study with domain experts that we conducted in previous work (Boukhelifa et al., 2013). We wanted to evaluate the usability and utility of EVE, by trying to answer these three questions: (i) is our tool understandable and can it be learnt; (ii) are experts able to confirm known insight in their data; and (iii) are experts able to evolve views that contain new insight or allow them to generate a new hypothesis. For this study, we did not analyse the behaviour of the IEA, nor compare user evaluation strategies since our study subjects worked on their own datasets, which were different in type, number of dimensions and research questions, making an in-between subject study comparison unfeasible. Instead, we chose a qualitative observational study methodology (Carpendale, 2008; Meyer et al., 2012; Sedlmair et al., 2012) that better suited our evaluation needs.

4.1 Participants, Tasks and Data

We evaluated our prototype with five domain expert users (2 female), ages $27 - 42$ (mean 34.2). Experts were academics and practitioners who had multidimensional datasets related to their domain of expertise (scientific simulation, medicine and geography) and were interested in further exploration. They consisted of one graduate student, three senior researchers and one medical surgeon. Participants had previously explored their datasets using graphical tools (e.g. Excel and JMP) or used statistical methods (PCA and regression analysis) but felt there was more to discover in their data than was identifiable by their current tools. Experience with advanced multidimensional visualization tools varied from none, to experts who already used GraphDice or

[7] The minimal distance is calculated in the scagnostics space, whose values range between 0 and 1. Thus, the default value of 0.1 corresponds to 10% of the maximum value of a scagnostics score.

[8] The scagnostics fitness component has a weight of 1 and is not currently tunable by the user.

other SPLOM-based tools (two experts). None of the participants had previously used dimension combination to analyse their data, but three performed PCA-type analysis. Each session lasted on average 2.5 hours.

Participants were asked to carry out two main tasks: (**T1**) show using the tool what they already know about their data, hypotheses and questions they wanted answered; and (**T2**) explore their data in light of these hypotheses and research questions. The main study ran in two parts; a training part similar to the game task described in section 5.1, then an open exploration part where participants loaded their own datasets and explored freely. At the end, participants filled in a short questionnaire rating aspects of the tool (on a 5-point Likert scale), such as the ease of performing the two main tasks, and open ended questions regarding their exploration strategy and helpful features of the tool. Log data of user interactions was gathered for further analysis (see table1).

Expert	T1	T2	Q	Data	Size	D	LimitSearch	Evolve	Eval	OVisits	NVisits	Insight
1	4	4	3	business	9x900	1:10	3	3	16	40	105	2(1)
2	4	4	3	timeseries	7x78	1:33	4	3	8	114	115	4(3)
3	5	5	3	geometrical	12x67	0:49	4	21	90	99	344	2(1)
4	5	4	3	statistical	10x200	2:23	7	13	83	110	309	6(1)
5	3	2	4	geospatial	11x653	1:27	5	5	20	64	229	-

Table 1: Log data showing: (*T1*&*T2*) experts scores for ease of completing tasks T1&T2 on a 5-point Likert scale where 5 signifies "very easy", (*Q*) score for user agreement with the system's cell evaluations on 5-point Likert scale where 5 indicates strong agreement, (*Data*&*Size*) type and size of dataset (dimensions x datapoints), (*D*) duration (hh:mm) of T2, (*LimitSearch*) breath of exploration indicated by the number of times the expert limited the search space, (*Evolve*) depth of exploration indicated by the maximum reached generation, (*Eval*) how many new cells were evaluated by the user, (*OVisits*&*NVisits*) number of times the expert visited the original cells and the new cells respectively, and (*Insight*) the number of times the expert limited the search space before finding the insight and the generation within that subspace (between parenthesis) where the insight was found.

4.2 Summary and Discussion of the Results

All participants in this study were able to easily confirm prior knowledge about their data except for one expert who found this task challenging because of the lack of data aggregation that her type of analysis requires. Overall, participants confirmed known correlations, clusters or outliers in their data. In the remainder of this section, we summarise our study findings highlighting new insight, successful tasks and exploration strategies. We end this section with a brief discussion on the limitations of our framework from a user evaluation point of view.

Insight Generation and Tasks. If we include hypothesis formation as part of insight generation, similar to Saraiya et al. (2005), *EvoGraphDice* helped our participants generate new insight in the form of distinct observations about the data (four experts), new hypothesis (one expert) and better formulation of research questions (four experts). Distinct observations found by the experts were either clustering, linear or non-linear relationships, and similarly to generated hypotheses, they always linked a dimension in the original dataset and a new proposed dimension. The subjective evaluation of ease of task T2 (table 1) shows most experts found it easy to reach new insight: 1 x 'very easy', 3 x 'easy' and 1 x 'not easy'. Unsurprisingly, those who reached a concrete new finding scored the tool highly in comparison to those who did not.

The found solutions were regarded by the experts as interesting because they had

one or more of the following properties: (a) a visual pattern such as those modeled by the scagnostics measures; (b) a simple formula involving few dimensions; (c) a selective choice of dimensions corresponding to an unformulated hypothesis or an inherent aspect of their data model; and (d) a domain value. Regarding the latter point, not all participants were able to state the immediate domain value, but in general our participants stated that *EvoGraphDice* helped them: (i) interact visually with their data (expert 3), (ii) try out alternative scenarios by editing dimensions (experts 1,2), (iii) think laterally (expert 2), (iv) quantify a qualitative hypothesis (expert 1), and (v) formulate a new hypothesis or refine an existing one (experts 1-4).

Exploration Strategies. Overall, participants followed the same exploration pattern consisting of first examining the original dimensions, then inspecting and evaluating the first generation of the proposed dimensions (returned by the PCA), followed by one or more iterations of the following steps: (i) limit the search space; (ii) select and rank cells; (iii) evolve; and (iv) interpret and verify. However, the frequency of using some tools (e.g. "evolve" vs. "limit the search space") varied depending on whether the expert had an a-priori focused hypothesis (i.e. a research question involving typically $3-4$ dimensions). We observed that the looser the initial hypothesis, the more often they tried to change the search space; and the more focused the hypothesis the more generations they inspected. Indeed, these two strategies of *exploration* and *exploitation* are supported by EAs (Banzhaf, 1997), where on the one hand the user wants to visit new regions of the search space, and on the other hand they want to explore solutions (combined dimensions) close to one region of the search space. However, our study also highlighted limitations to our approach, as discussed next.

Issues and Limitations. There are issues to using our framework, mainly related to the types of datasets we are able to visualise, the understandability of generated combined dimensions, and issues related to algorithm convergence. Most relevant to this paper are the convergence issues (for a detailed discussion on the first two issues, refer to Boukhelifa et al. (2013)). As we are dealing with a small population of dimensions evolved during only a few generations, the algorithm cannot be considered as having converged in the classical sense. Theoretical analysis considers two main mechanisms that govern the behaviour of EAs: focus (convergence or exploitation) and diversity (random search or exploration). In their most classical uses, i.e. computationally expensive optimisation, the exploitation mechanism is privileged and the exploration component is used only to ensure the robustness of the results. In the interactive framework where creativity or new feature discovery are sought, the same mechanisms operate but with a different balance: exploration capability seems to have a bigger impact. Additionally, talking about convergence for EVE systems is even more difficult as usually the users themselves do not clearly know what they are searching for. We have noticed from this study that the "guided" exploration ability of *EvoGraphDice* is exploited in different ways by experts: some explored, thus it can be said they focused on the random search ability of the algorithm, while others exploited with longer runs (> 10 generations) thus focused on guided search (convergence). In both cases, the IEA provides a unified framework for users that sometimes are interested in focussed search, e.g. if they know what they want, or in explorative suggestions if they do not have a-priori precise hypothesis. The question of convergence and issues related to the search and exploitation mechanisms of *EvoGraphDice* are further analysed in a more controlled study where users had a precise and well defined task to perform, as described in the next section.

5 Experimental Analysis of User and Algorithm Behaviour

We conducted a study in order to collect data about user interactions and the fitness function. In particular, we wanted to understand user strategies in solving an exploratory task, and the IEA's convergence focusing on the learning behaviour of the algorithm across generations and its ability to adapt to user focus.

5.1 Participants and Tasks

We run our study with 12 participants (5 female), ages $23 - 43$ (mean 28.5). Participants were researchers from two different institutions who had limited experience with SPLOM-based visualizations (only three had previous experience with SPLOMs).

The task was designed as a game. A 5D dataset was synthesised with two enclosed curvilinear dependencies between two variables (x_0 and x_1) and random data for the rest of the dimensions. Participants were asked to evolve a scatterplot where it is possible to separate the two curves in Fig.3 (left) with a straight line (equivalent to separating the two corresponding convex hulls). They were given around 20 minutes to complete the task. Ten participants successfully separated the two curves, while the remaining participants evolved views very close to a correct solution within the allocated time.

We generated two levels of difficulty for the game, where difficulty relates to the amount of enclosure between the curves – the bigger the overlap area between the curves the more difficult it is to find a solution. Participants started with level-0 and depending on their performance, they also did a level-1 session. When participants could not reach a solution in 20 minutes, we prompted them to restrict the search space to dimensions x_0 and x_1 to help them find a solution more quickly. Participants stopped the game when they found a solution, or when they felt the tool is no longer proposing interesting views (for struggling participants, a minimum exploration time of 20 minutes was always respected). With the exception of two users, all participants successfully evolved a view separating the two convex hulls for level-0 (average time to find a solution was 17 minutes), but only 6 out of the 11 participants who tried level-1 managed to find a solution (average time 15 minutes) as this level proved to be too difficult.

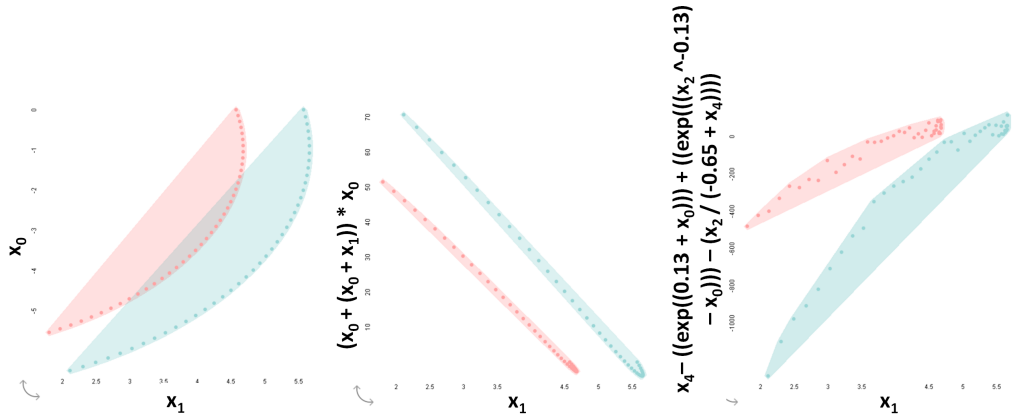

Figure 3: Two different solutions (screenshots of plots) for the game task (left) that involve a simple dimension combination (middle) and a complex formula (right).

For each generation	9 scagnostics weights, w_k.
For each individual	generation number, genome (math. formula), surrogate function term, f_{sc}, complexity term, f_c, average user evaluation, f_u, resulting fitness.
For each evaluated cell	generation number, 9 scagnostics, predicted evaluation (= $\sum w_k$ x Cell Scagnostics), user evaluation.

Table 2: Log data capturing information on the GP.

5.2 Data Collection

Log data gathered and analysed (Table 2) includes three types of information related to: (a) user interactions with the tool such as cell selections and evaluations via the slider; (b) genetic engine status at each generation such as details about the individuals in each generation, including their fitness components and scagnostics scores, and the cells these individuals participated in; and (c) the overall learned scagnostics weights. A total of 23 log files were collected and stored in a database. However, in the following section we only analyse data corresponding to level-0 as it was the session performed by all participants. In total, we analysed 12 game sessions (one per participant).

5.3 Data Analysis

We carried out four different types of analysis for this game scenario, two examining user behaviour and two the algorithm's. In particular, we wanted to study whether people are attracted to important data variables for their task or particular visual patterns in the views, and whether they are distracted by the interface or system suggestions (analyses 1 & 2). We also wanted to assess in detail the exploitation and exploration capabilities of the system by investigating the system's ability to take user evaluations into account and examining population diversity (analyses 3 & 4 below).

1. *user strategy analysis* to understand the different approaches users took to solve the task,

2. *user focus analysis* to highlight hot spots in the user interface and assess user evaluation strategies,

3. *convergence analysis* to assess the algorithm's ability to steer the exploration toward a focused area of the search space, and

4. *diversity analysis* to assess the richness and variability of solutions provided by the algorithm.

For comparison purposes, and since the number of generations per exploration session differs between users, we divided the generations into three bins, corresponding to the start, middle and end of the game session. We tried to get the same bin size for all groups when possible, and at the very least ensure that the start and end bins always have the exact same size when integer division by three was not possible.

5.3.1 Learning User Strategy Through Scagnostics

Our observational study (section 4) revealed that users followed different strategies to solve the game task. We were interested in characterising these strategies, and comparing successful and unsuccessful game attempts. To achieve this, we looked at the scagnostics weights distribution along generation bins for all users that explored at least three generations. Fig. 4 displays the types of scagnostics of various exploration sessions for level-0 of the game including sessions where the user restricted the search space (Fig. 4d,e,m,o). In such cases, and since the scagnostics weights are preserved from one subspace to another, we concatenate the generations of the consecutive search spaces. The success rate for this game level was 83% with an average exploration session of 19 generations and a standard deviation of 18 generations (for successful users). The high success rate implies that the algorithm is behaving correctly, i.e. overall, it is learning from user interactions and providing pertinent views to the user. However, the high variability in how quickly users found a solution may be influenced by the *type* of the searched visual pattern and the *stability* of the exploration strategy.

Type: We first looked at the three highest scagnostics weights for all sessions at any bin (Fig. 4). It appears that overall, *skinny*, *convex* and *sparse* distributions are the dominant patterns of exploration (Fig. 4a). For the task of separating two curved lines, these scagnostics distributions correspond to the following strategies: a) *skinny*: trying to straighten the curves (33% of games sessions); b) *convex*: preserving the original shapes but the curves are slightly separated laterally (25% of game sessions), and c) *sparse*: trying to introduce holes in the view (41% of game sessions). These patterns can also be observed for overall success games (Fig. 4b) and to a slightly lesser extend for unsuccessful sessions (Fig. 4c) where the dominant scagnostics on average are *skinny*, *sparse* and *outlying*. Since the types of dominant scagnostics are very similar regardless of outcome (although we acknowledge the lack of data for failed sessions for level-0), we hypothesise that the discriminating factor that determines convergence (and its speed) may be more related to the *stability* of the exploration strategy in relation to the start, middle and end bins discussed earlier. As we discuss in the next section, users having a more stable strategy early in the exploration session seem to converge more quickly.

Stability: We then looked at the stability of exploration strategies, which we define as the user's persistence in searching for the same visual pattern from one bin to the next. We identified four levels of persistence (from stability-0 to stability-3) where the level number refers to the average co-occurrence of top scagnostics types (i.e. the highest three) between consecutive bins. Thus stability-0 strategies have no co-occurrences of scagnostics types between bins #1 and #2 and between bins 2 and 3 (none for this study level); stability-1 strategies have at most one co-occurrence on average (session Fig. 4n) and so on for stability-2 (sessions Fig. 4d,e,h,k,o) and stability-3 (sessions Fig. 4f,g,i,j,l,m). When comparing successful and unsuccessful sessions it seems, indeed, that successful ones had more stable strategies (stability-2 or stability-3) whereas unsuccessful sessions had a more erratic behaviour (namely Fig. 4n). The session shown in Fig. 4o i s an exception in that although a solution to the game was not found, the exploration strategy was stable enough (stability-2). It may well be that the solution was just around the corner (after 23 generations the user gave up). Looking at only successful attempts, it appears that sessions having stability-2[9] converged to a solution after, on average, 24 generations, whereas, for stability-3[10], users converged after on

[9] Convergence generation per session having stability level-2: d(59), e(25), h(6) and k(9).

[10] Convergence generation per session having stability level-3: f(3), g(29), i(5), j(13), l(10) and m(40).

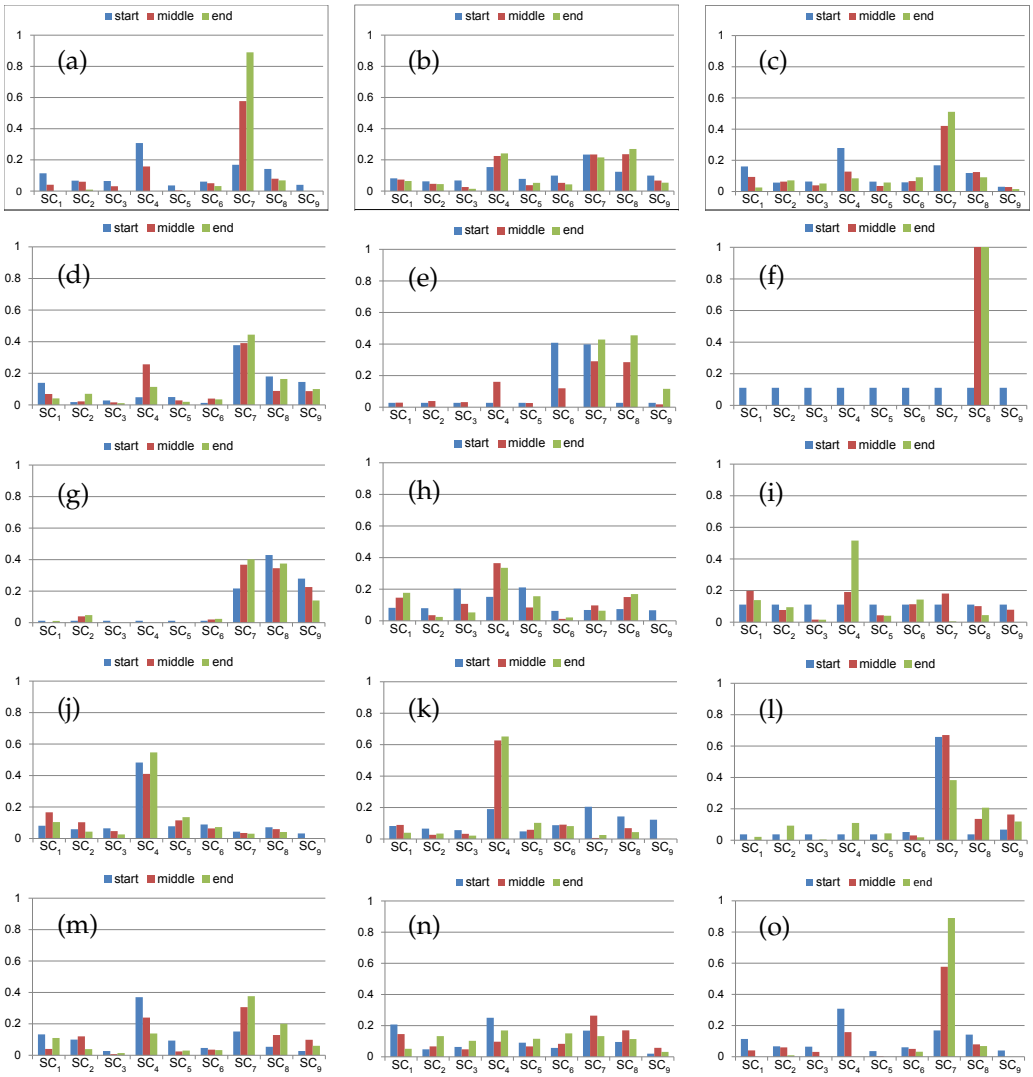

Figure 4: Scagnostics weights over time, where time is split into three generation bins for the start, middle and end of the session; scagnostics types on the x-axes: SC_1:monotnoic, SC_2: stringy, SC_3:skinny, SC_4:convex, SC_5:striated, SC_6:sparse, SC_7:clumpy, SC_8:skewed and SC_9:outlying. The plots are for (a) overall, (b) overall success, (c) overall failure, (d–m) successful sessions per participant and (n–o) unsuccessful sessions per participant, all for level-0 of the game.

average 16 generations. Although we do not have many examples of sessions with low stability levels, session (n) in Fig. 4 shows an unclear stragety having only stability-1, which might have led to non-convergence and thus to a failed game outcome[11]. Although variance across participants is large, it seems that on average the more stable the strategy the sooner the convergence.

Finally, we investigated whether users who limited the search space (sessions

[11]Note that no session for this game level had stability-0.

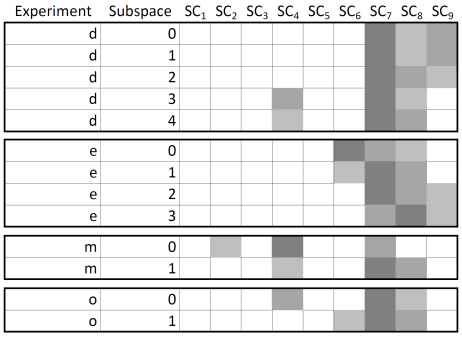

Figure 5: Stability of exploration strategies based on subspace change. Sessions with more than one subspace are [d,e,m,o] where o is the only unsuccessful session. Coloured cells refer to the highest three scagnostics in each subspace; the darker the colour the higher the scagnostic weight (SC_1:monotnoic, SC_2: stringy, SC_3:skinny, SC_4:convex, SC_5:striated, SC_6:sparse, SC_7:clumpy, SC_8:skewed, SC_9:outlying).

d,e,m,o) tended to have a less stable strategy than those who did not. It appears that, on average, sessions where the user did not limit the search space had a more consistent strategy with stability-3 observed in five sessions, versus only one such session in the limited search space condition. However, looking at the dominant scagnostics weights for each subspace, it appears that, on average, users who limited the search space are more likely to keep the same method of exploration between subspaces but not by much (five out of the nine subspace changes did not involve a change of target visual pattern described by the top three scagnostics types) as shown in Fig. 5.

Summary: *This analysis showed that EvoGraphDice allows for different types of exploration strategies centred around three dominant scagnostics (skinny, convex and sparse) that appear to be relevant for the game task. We also found that the stability of the exploration strategy may be an important factor for determining the outcome of the exploration task and the speed of convergence, since successful game sessions had a more consistent strategy when compared to the unsuccessful ones, and they converged more quickly on average. Moreover, users who limit the search space are likely to keep their exploration strategy, implying that perhaps the most important reason for changing a search space is to focus on a specific set of data dimensions.*

5.3.2 User Visitation and Evaluation

We analysed user focus of attention in terms of their cell *visitation* and *evaluation* patterns. The visitation patterns are examined so as to verify that participants focused more on cells with proposed dimensions that included the original target dimensions (x_0, x_1 located in the first two columns of the SPLOM, see Fig. 2). Similarly, the evaluation patterns were examined to verify that participants ranked cells with the desired variables higher. Patterns to the contrary would indicate that participants' attention was attracted elsewhere, which could indicate either that our system failed to provide interesting views, or maybe that there was an interface bias.

Visitation: We counted how many times each user visited a cell, by selecting it in the yellow matrix quarters of proposed dimensions[12]. We mapped this count to colour intensity, thus, the more visited the cell the more intense its background colour (Fig.

[12]Note that we only report on cell selections occurring at the bottom left quarter in Fig. 2 where 87% of cell visits took place. The remaining selections were located at the top right quarter of the matrix as it is not possible to select cells from the bottom right quarter of the SPLOM for this version of the *EvoGraphDice*.

6). In the same cell, we also draw a line graph to show the number of visits across participants per generation for all sessions (Fig. 6I), for success (Fig. 6II) and failure (Fig. 6III) where visit counts are normalised between the minimum and the maximum values per cell.

As expected, users had a greater tendency to visit the cells where the proposed dimension contained either of the original dimensions x_0 or x_1 (i.e. columns x_0 and x_1); 70% of all cell visits were in these two columns. We performed a statistical test to compare the percentage of visits on each column. As our data does not follow a normal distribution, we conducted a Kruskal-Wallis non-parametric test. The analysis revealed a significant effect of column on number of visits ($\chi^2(4) = 31.4, p < 0.001$). A post-hoc pair-wise comparison using a Mann-Whitney test, showed indeed significantly more visits in cells of column x_0 and x_1 (mean percentage of visits 36.5% and 33.5% respectively, i.e. more than half of all visits), compared to visits in all other columns (mean percentage of visits 12%, 9.4%, 8.6%). The fact that these two columns are placed in a prominent position in the matrix (i.e. the first columns on the left) may have encouraged participants to visit them. There was, however, no difference in visits between columns x_0 and x_1, or between the other columns (all $p < 0.05$).

As we do not have enough data for failure attempts for level-0, we refrained from statistically comparing evaluation patterns per outcome. Nevertheless, a visual comparison indicates that, overall, visitation patterns do not seem to differ for success or failure (focusing on average visitation per cell, i.e. colour saturation). However, we can see that for success there is a tendency to focus more on the first two columns (containing the original dimensions x_0 and x_1); whereas for failure, visitation patterns are more scattered and include column x_3 and sometimes x_4. As users do not seem to find interesting patterns in the first two columns, their search seems to extend to other dimensions, despite the fact that these will most likely contain noise.

In the first two columns, participants also visited more cells that are highly placed in the bottom left matrix quarter (normalised average visitation values for pertinent cells, from highly placed to bottom placed, equals to 0.9, 0.8, 0.7, 0.7 and 0.6). This last point is not a surprising finding, rather it is likely that cells having these dependent variables also show an interesting pattern, and are thus more likely to be visited. These cells are also highly ranked by the system as indicated by their position in the matrix (the higher the row in the proposed dimension the higher its fitness value as described in section 3.1). The declining slope of the graphs corresponds mainly to the different generation runs of game sessions; there are only a few sessions with more than 20 generations with most of the sessions having less than 10 generations. We have also observed that with time participants based many of their evaluations on quick visual inspections of the SPLOM cells rather than explicitly selecting cells and viewing their contents in the main scatterplot view (effectively making their cell visits count decrease). Indeed, as participants gained confidence in using the tool, they tried to optimise their interactions with the system, persumably to avoid user fatigue.

Evaluation: We also looked at cell evaluations with a similar visualization to the cell visitation plots, but here we map the average user satisfaction score to background colour saturation rather than the average cell visits. Thus, more highly scored cells are more intense. The scaterplot in Fig. 7I shows which user evaluation scores appeared over consecutive generations for each of these cells. These are distinct evaluation scores rather than averages.

We observed again that more cells are highly ranked in the first two columns than the rest (often ranked low). A Kruskal-Wallis nonparametric test revealed a signifi-

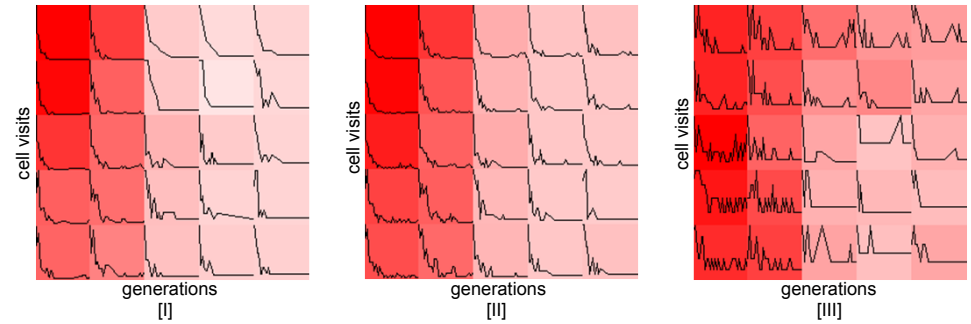

Figure 6: User visitation patterns for each cell located at the bottom left quarter of the SPLOM where proposed dimensions are plotted against the original dimensions in abscissa. The graphs represent the normalised per cell visits count over generations for [I] all sessions, [II] successful ones and [III] unsuccessful ones. Colour saturation corresponds to the overall number of visits across participants (normalised across cells). The first two columns are of proposed cells plotted against the original target dimensions x_0, x_1.

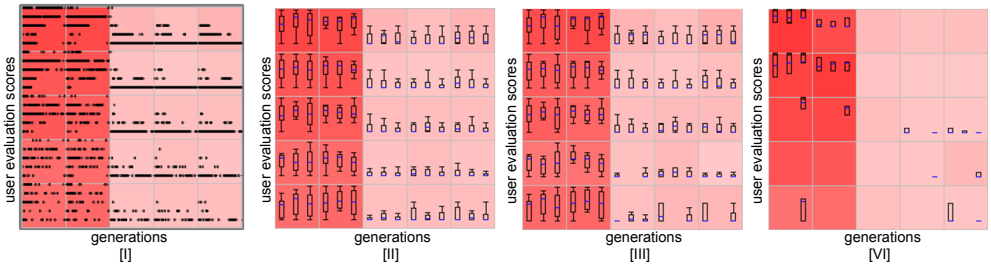

Figure 7: Plots of user evaluations for each cell located at the bottom left quarter of the SPLOM where proposed dimensions are plotted against the original dimensions in abscissa. Background colour saturation indicates mean user satisfaction score. Scatterplots in [I] show all user evaluation scores over generations. The emerging lines correspond to scores 1-5 (5 on top). Summary statistics for user evaluation scores for all three bins are shown for all sessions [II], successful [III] and unsuccessful sessions [VI].

cant effect of column on user evaluation score ($\chi^2(4) = 22.1$, $p < 0.001$). A post-hoc pair-wise comparison using a Mann-Whitney test, showed that participants assigned significantly higher evaluation scores in cells of column x_0 and x_1 (means 3.4 and 3.2 out of 5 respectively) compared to the values given to other columns (means 1.3, 1.4 and 1.5). There was no difference in evaluation values between column x_0 and x_1, nor between the other columns (all $p < 0.05$). As we do not have enough data for failure attempts for level-0 of the game, we refrained from statistically comparing evaluation patterns per outcome.

We then grouped user evaluations into three bins corresponding to the start, middle and end of the exploration session, and we plotted the box-plots for all participant scores across bins, for all sessions Fig. 7II, for success Fig. 7III, and failure Fig. 7VI. Our goal with this box-plot visualization over time (bins), was to examine if variability across evaluations tended to stabilise, indicating that the system consistently proposed suggestions that were preferred by users. Looking at the spread of box-plots, we can observe that overall, most user evaluations for non-pertinent columns (i.e. that do not include x_0 or x_1) are usually ranked consistently low regardless of outcome, irrespec-

tive of bin. Conversely, irrespective of bin, the range of evaluations is larger for the pertinent cells (i.e. cells with proposed dimensions that include x_0 or x_1 in the first two columns), perhaps due to the large diversity of proposed solutions, and consequently diversity of evaluation strategies adopted by our participants.

Summary: *This analysis shows that users are more likely to visit cells showing dimensions relevant to their task. Moreover, these cells are on average ranked highly by the user. Since for this game, the main dimensions relevant to the task appear on the top left side of the proposed cells, users intuitively started navigating that way. What we are seeing in the results is probably a mixture of task-relevance and intuitive-navigation, as the relevant original dimensions are placed in a prominent position in the matrix.*

5.3.3 Algorithm Convergence

A different type of analysis centers around the algorithm's convergence to a desirable solution or an interesting subspace. We examined the rate of concordance between user scores of evaluated cells, and their "predicted" values which are calculated from the current scagnostic weights learned at each generation and the scagnostics values of the corresponding cell (see equation 1): $f_{sc}(y_i) = \sum_{k=1..9} w_k(\max_j SC_k(y_i, x_j))$. Averages of actual user evaluations (1 to 5) throughout all the exploration stages (i.e. for bins 1, 2 and 3) have been plotted against the average system predictions in Fig. 8, for all sessions (I), for successful (II) and for failed game sessions (III).

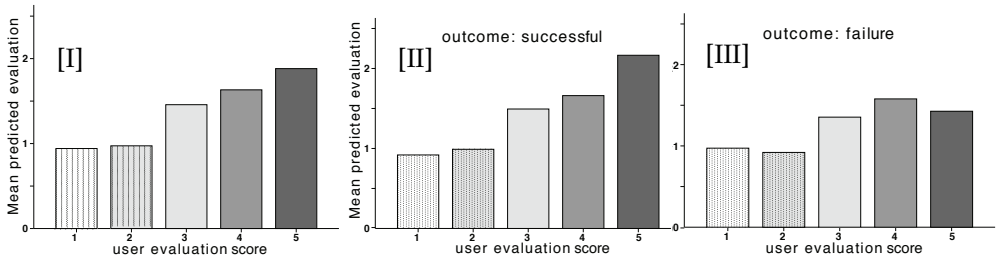

Figure 8: Average system prediction of plot evaluation, by the actual user evaluation score (1 to 5), overall across all game sessions [I], for successful sessions [II], and unsuccessful sessions [III].

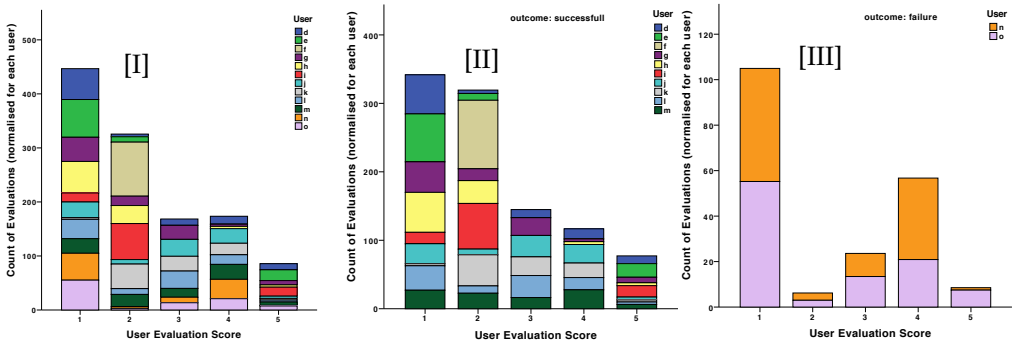

Figure 9: Number of evaluations conducted per user for each evaluation score (1 to 5) in bin 3. As each user provided a different number of evaluations, the count is normalised for each user as a percentage of all the evaluations of the specific user. Overall across all game sessions [I], for successful sessions [II], and unsuccessful ones [III].

Overall (Fig. 8I) system predictions tend to consistently under-evaluate, but in a way that follows the ordering of user scores. Thus an ascending pattern can be observed respecting the order of evaluation scores (so for example cells scored 5 by the user are also ranked the highest by the system on average and so on). Although for individual cells the difference between user and system evaluations may differ (whether they preferred the proposed cell or not) on average the system prediction order tends to follow that of the user. A Pearson correlation test revealed a weak positive relationship between user evaluation scores and predicted ones ($r = 0.3$, $p < .01$).

Visual inspection of values for success (Fig. 8II) and failure (Fig. 8III) seems to indicate a tendency for the system to recognise plots as mostly *good* or *bad* (notice the difference of scores below 2 and over 3), where bad plots are given a score of 1 or 2. For successful attempts (Fig. 8II), an ascending pattern is also clearly observed, but the system has fairly similar predictions for cells evaluated by users in the middle scores (3,4). This may be due to specific users being inconsistent in their search or ranking strategies in these middle-range scores. Looking at the failed attempts (Fig. 8III), an ascending pattern is present, although the system had more trouble with high scores (where the order is reversed between 4 and 5). This may be due to specific users being inconsistent in their scoring strategies, or lack of data points for this evaluation score. Nevertheless, even here the distinction between good and bad graphs is clear.

As mentioned, a large variety of behaviors can be observed regarding user scoring strategies (Fig. 9). Some participants have a coarse scoring strategy (e.g. Fig. 9II.e), tending to lump evaluation scores to fewer levels, others provide more fine-tuned ratings covering the five scoring levels (e.g. 9II.m), or a combination of the above at various stages of exploration. What can be observed is that, for the successful game outcome, more participants adopted the fine-tuned evaluation strategy than the coarse one: sessions (d,g,j,k,l,m) for fine-turned and (e,f,h,i) for coarse. Interestingly, those who adopted a fine-tuned evaluation strategy did not converge more quickly (in 26 generations on average versus only 9 for coarse).

We next looked at convergence at the generation level, with the predicted values averaged per generation to observe the progression of the algorithm predictions. We focused this exploration on bin 3, i.e. the last part of the exploration for all users. We chose to look at bin 3 only, because the number of generations across participants differs, and thus they each reach a consistent strategy at different generations. Focusing on bin 3 we can assume that users have a clear strategy at that stage and thus system predictions should clearly follow. This data is plotted as line graphs in Fig. 10 where the x-axis corresponds to the generation number (from first generation to the last generation in bin 3) and the y-axis refers to the average system prediction. We plot a system prediction line for each actual user evaluation score level (1 to 5). Note that successful game sessions reached 8 generations at most in the third bin, while unsuccessful ones reached 10.

The order of predicted levels is fairly consistent with that of the user evaluations for successful sessions (Fig. 10I), as in the ordering of the predicted levels is similar to the user evaluation (e.g. predicted values for cells evaluated by users as 3/5 are higher than the predicted values for cells evaluated by users as 2/5). For failed sessions (Fig. 10II), this pattern is noisy after a few generations for most evaluation levels. For successful user g (Fig. 10III), that uses all evaluation values across generations, we see clearly the system evaluating "good" and "bad" cells. Whereas a failed participant n (Fig. 10VI) has a coarse scoring strategy with missing evaluation values, and system predictions fluctuate as it tries to follow the user's changing strategy (see also Fig. 4n).

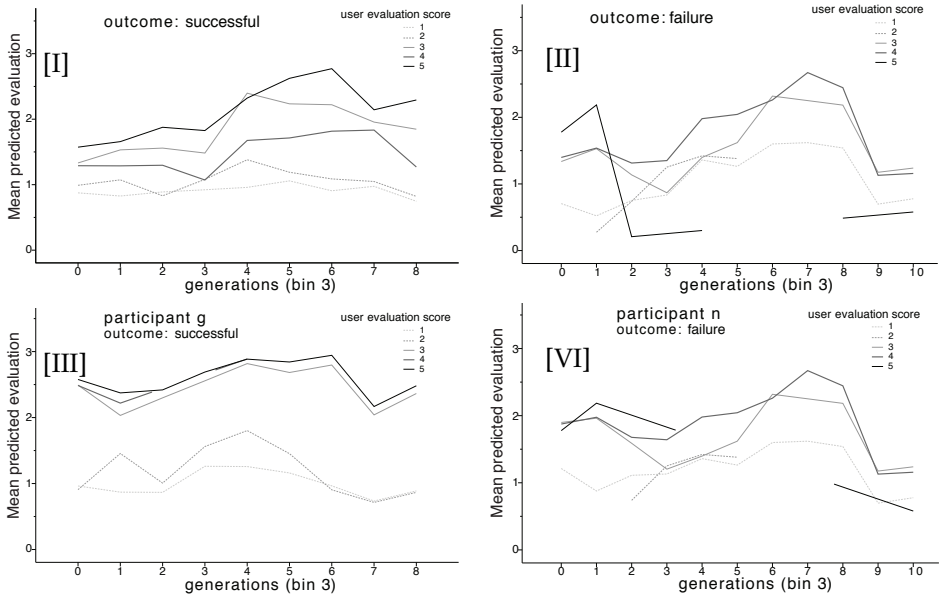

Figure 10: Average user evaluation vs. predicted evaluation over generations, for [I] successful sessions on average, [II] unsuccessful sessions. In [III] a typical successful participant session (Fig. 4g), and in [VI] a typical unsuccessful one (Fig. 4n). Line breaks in (III,IV) indicate lack of user evaluation of that value in these generations.

Summary: *Our analysis shows that on average the surrogate function follows the order of user ranking of scatterplots fairly consistently, even though users seem to take different search strategies (described earlier in section 5.3.1), as well as different evaluation strategies that are either coarse, tending to lump evaluation scores to fewer levels, more fine-tuned covering the five score levels or a combination of both at different stages of the exploration. Our results seem to suggest a link between user evaluation strategy, and outcome of exploration and speed of convergence, where users taking a more consistent approach (either fine-tuned or coarse) seem to converge more quickly.*

5.3.4 Diversity

A final type of analysis focused on the diversity mechanism of the EA which ensures that each suggested individual is different enough from others in the current population (see section 3.3). Since we particularly characterise scatterplots in terms of their visual appearances using the scagnostics distributions, we wanted to examine how diverse the proposed views were with regards to their dominant visual pattern and whether this difference can be observed more strongly at a particular stage of the exploration session (with regards to start, middle or end bin).

We consider 13 diversity factors, consisting of the 9 scagnostics distributions (we take values for each generation rather than scagnostics weights), in addition to four factors related to the fitness function evaluation: the overall fitness value, the user evaluation, the complexity evaluation and the scagnostics evaluation. We used two metrics to quantify the diversity with regards to each of these factors:

i) The mean difference *MD* which measures the statistical dispersion of views by calculating the average absolute difference between each two individuals in the current population; and

ii) the Shannon-Index *H* (Shannon, 1948), an indirect diversity measure which de-

scribes the average degree of uncertainty of predicting the species of an individual picked at random from the community, factoring in both the *abundance* and *evenness* of the species in that community. As with the *MD* measure, a high *H* value implies a more diverse population.

We found that the Shannon-Index *H* was not discriminative enough for our small population data (each generation having only five individuals), therefore, the diversity analysis described below is based solely on the mean difference *MD* metric.

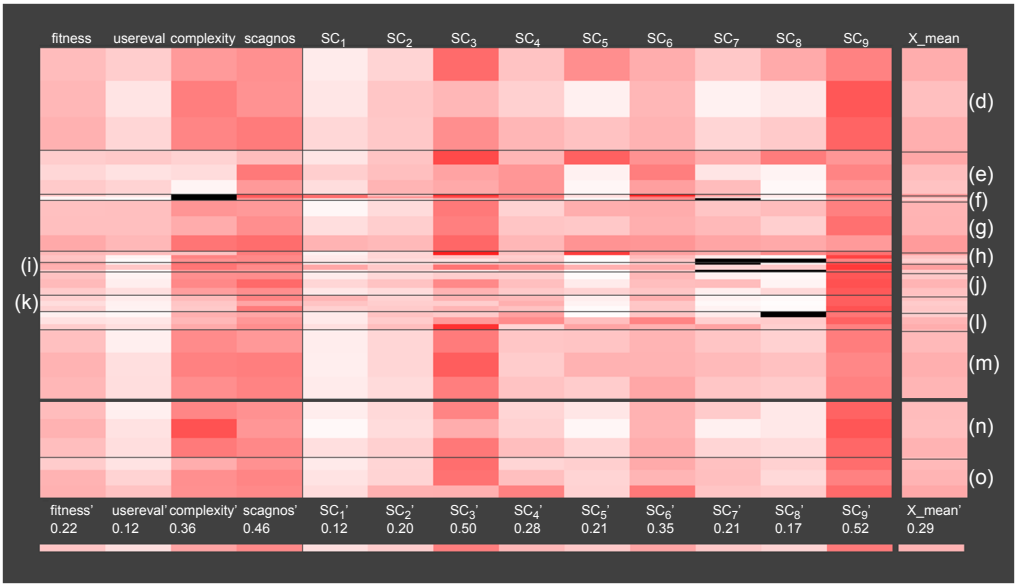

Figure 11: Diversity heat maps for level-0 of the game for all sessions. The first 13 columns describe the diversity factors, where scagnostics types are SC_1:monotnoic, SC_2:stringy, SC_3:skinny, SC_4:convex, SC_5:striated, SC_6:sparse, SC_7:clumpy, SC_8:skewed and SC_9:outlying. Each row in the map corresponds to a generation bin of one user session. Sessions are listed vertically (top to bottom) and are separated using a dark horizontal line. Sessions (d–m) are for successful game attempts and sessions (n) and (o) are for failed ones. Cells are coloured to reflect the bin's mean difference BMD value for that factor (the higher the BMD value the darker the colour). Mean values for each bin (rows) and factor (columns) are displayed vertically and horizontally. X_mean is the average of all diversity factors for each bin, and X_Mean' value is the average of X_mean values for all sessions.

To have an overview of the diversity information that we calculated, we created a heat map visualization starting from a matrix where columns represent the diversity factors mentioned earlier and rows represent generation bins, see Fig. 11. We concatenate all game sessions vertically, first successful sessions from (d) to (m) then unsuccessful ones (n) and (o) for level-0 of the game. We had in total 248 generations and 1275 individuals. We use a dark horizontal line to denote the start of a new user game session and we add summary information (the mean) across factors both horizontally and vertically (we denote these by *factor'*). The X_mean column refers to the cross factors average values, and the corresponding X_mean' is the average of all the X_mean values.

Since we are interested in comparing the diversity of generations in relation to

the exploration stage, we report the average mean difference per generation bin *BMD*, where each game session is split into start, middle and end bins (Fig. 11). Therefore, each cell corresponds to a generation bin and is coloured to reflect the amount of diversity in that generation using a linear scale, such that the more saturated the colour the more diverse the population (i.e. the bigger the BMD). Black cells represent non-diverse generation bins where individuals have identical values for that diversity factor. This visualization is similar to the diversity map by Pham et al. (2010) where rows are attributes of a multidimensional dataset and columns are attribute value buckets.

Overall, it appears that the most diverse fitness factor is *scagnostics* and the least diverse is the *user evaluation* regardless of outcome. The former observation may be due to the highly variable nature of the scagnostics factor given that it comprises various components (the nine scagnostics measurements and weights that are being updated over time). Out of the nine scagnostics diversity factors, *clumpy* and *monotonic* show more variations than the other factors (see Fig. 11 bottom, primed factors). With regards to the low diversity in user evaluations, this might be attributed to the lumping effect reported in section 5.3.3 where some users tended to use a few ranking scores.

Looking at the individual factors and focusing at the scagnostics distributions, it appears that these factors are diverse at different times of the exploration. On average, most factors are diverse at the start of the exploration (five out of nine scagnostics: *clumpy* for sessions (d,e,f,n,h,i,j,k,l,m,o), *striated* for sessions (d,e,f,n,i,j,k,l,m,o), *stringy* for sessions (d,f,n,i,j,k,l,m), *skinny* for sessions (d,e,f,h,i,j,k,l,m) and *convex* for sessions (d,e,f,n,i,j,k,l,m)); one scagnostics type is more diverse in the middle (*monotonic* for sessions (d,e,g,n,i,l,m)); and only three scagnostics are more diverse at the end of the exploration (*outlying* for sessions (d,g,n,h,i,k,m,o), *skewed* for sessions (e,g,n,i,j,k,m,o) and *sparse* for sessions (d,e,g,h,l,m,o)).

These findings may be difficult to interpret at this level without examining the different exploration strategies. In principle, if the user adopts a particular strategy, say favoring sparse distributions, the system should converge to provide more of these solutions especially towards the end of exploration for successful sessions, and thus the BMD for this diversity factor should decrease. This can be indeed observed for six out of the ten successfull game sessions (see Fig. 4 and 11), where sessions (i,j,k) having *convex* as the highest scagnostics for bin 3, session (m) with *clumpy* as highest scagnostic for bin 3, and sessions (e,f) for *skewed*. These sessions all saw their BMD drop in bin 3 (at least from bin 2) for their respective dominant scagnostics (Fig. 11)

Summary: *The diversity analysis shows that, in terms of the visual pattern, the IEA provides more diverse solutions at the beginning of the exploration session (bin 1) before slowly converging to a more focused search space (in bin 2 or 3 for the success outcome) for most sessions. These effects correspond to the exploration component (random search) and the exploitation component (focus) of the genetic engine.*

6 Discussion and Future Directions

To fully evaluate an IEA system, we feel a collection of analysis methods is needed, both user-centered, observing the utility and effectiveness of the system for the end-user, and algorithm-centered, analysing the algorithmic behaviour of the system. To this end we previously conducted an observational study with experts analysing their own data using our system (Section 4), and a new controlled user study with synthetic data to analyse different aspects of the algorithmic behaviour and its use (Section 5).

In the observational study, our experts were able to verify known patterns in their data, and generate new insights using our tool. As discussed before, due to the SPLOM representation of *EvoGraphDice* the system can visually handle datasets with relatively few data dimensions, and cannot handle at all data types such as time series (an issue raised by one expert). Nevertheless, the algorithmic part of the system has no such limitations. It remains future work to develop visualizations that can express temporal combinations of dimensions proposed by an IEA. Another issue that was raised is the scalability of our matrix representations to a large number of dimensions. Aside from using known dimensionality reduction techniques (such as clustering), there is a need for further research on how to select appropriate visual representations of the original and proposed dimensions, and potentially how to adapt these views on the fly based on the underlying data. As such, we believe there still is a lot of potential in continuing the dialogue between the visualization and IEA communities.

In our new experimental analysis, we were able to compare the use of the system across different users in a more controlled setting and scenario. Our analysis of strategies (Section 5.3.1) shows that participants adopted different strategies, and in particular different exploration patterns for successful and unsuccessful sessions. The analysis of the content of the surrogate function, via the observation of the variation of the learned weights of the scagnostics measurements, highlights a difference in users' focus of attention (i.e. searched visual pattern). For successful game sessions, there are clearly two main strategies: one tending to "unfold" the curved shapes by favouring linear scagnostic measurements (e.g. middle solution in Fig. 3), the other trying to spread the figures laterally by favouring sparse or skinny scagnostics (e.g. right solution in Fig. 3). Our user strategy analysis also showed that stability may be an important factor for determining the outcome of the exploration task and the speed of convergence, since successful game sessions had a more stable strategy when compared to the unsuccessful ones.

The choice of visualization and the order of dimensions relevant to the task may have influenced the way users visited the new views or evaluated them. Our user focus analysis showed that users were more likely to visit cells that included dimensions relevant to the task (in our case, x_0 and x_1 columns) although not all users were conscious about this choice (from our observational study). It is likely that users found interesting patterns in these cells more than others, which also explains why these were ranked higher than the other cells. When these areas of the SPLOM did not show any interesting pattern, the search extended to other areas of the matrix. Nevertheless, there is always the possibility that the placement of proposed dimensions in the interface affects users' choices and attracts their attention. More work is needed to delineate the influence of interface design on evaluation strategies with regards to different visual search and analytics tasks.

The surrogate function that "approximates" user evaluations, is clearly not able to embed the explicit aim of the game (i.e. separating the convex hulls of two geometrical subsets), as it only performs calculations on the whole set of points of the scatterplot. However, our analysis suggests that the surrogate function is able to predict user's ranking order of scatterplots fairly consistently and is discriminative enough to allow various search strategies (e.g. favoring linear, convex or sparse distributions to solve a specific task), as well as different evaluation strategies that are either coarse or more fine-tuned. More extensive experimental analysis is needed in order to be able to characterise and generalise these strategies for tasks other than the game task studied here.

EvoGraphDice seems to exhibit a learning behaviour controlled by the diversity

component of the genetic engine which aims, on the one hand, to provide a diverse set of solutions and on the other hand to converge quickly to a more pertinent subspace. The diversity analysis in section 5.3.4, shows that on average the EA provides more diverse solutions at the beginning of the exploration session (bin 1) before slowly converging to a more focused search space (in bin 2 or 3 for successful exploration sessions). These effects correspond to the exploration component (random search) and the exploitation component (focus) of the genetic engine. These two mechanisms are transparent to the user. However, it would be interesting to provide the user with a meta-visualization of their exploration paths highlighting stages where they *explored* and others where they *exploited*, and to allow them to roll the system back to previous exploration stages. Such visualization tools may give the user a better understanding of their exploration behaviour, and may help them establish a more stable strategy.

Our general approach for steering visual exploration has the following characteristics: (i) **Intuitiveness:** a visual approach to interact with data requiring no prior statistical knowledge; (ii) **Interactivity:** rather than fitting the data to pre-defined shapes in a static manner, using an IEA the user can dynamically steer the exploration process towards a pattern of interest. These patterns can involve dimension concatenations that are not obvious at the outset of the exploration; (iii) and **Adaptability:** the system can adjust to user change of interest over time. However, there are also limitations to our approach related to the fitness function design (including the surrogate function), the IEA implementation and user-related issues, some of which we hope to address in future research informed by a deeper analysis of our collected log data.

Extensions to the fitness function: The main challenge in guided search is to determine what views of the data are interesting to the analyst. Currently, our fitness function has three components: the surrogate function, a complexity term and a user evaluation term. Other terms to help approximate user interest could be considered (in place or in addition) such as data related quality metrics (e.g. variance), or perception-based metrics (such as for correlation perception (Rensink and Baldridge, 2010) or similarity perception (Albuquerque et al., 2011)). Moreover, these different terms may have varying weights depending on the task at hand and user's domain knowledge of the data, emphasising either the automated components or the interactive term.

Robustness of the IEA: In general, we feel that the speed of convergence of the IEA depends on many factors including the size of the search space, the complexity of the sought pattern, the number of evaluated scatterplots and how often the user changed their focus and target search pattern. All these variables make it difficult to predict a convergence ratio or speed. As discussed in section 4.2, this is not easy to study as there is no unique solution to converge to, rather the optimisation is dynamically adapted to follow user interest over time. Visualizing past exploration paths, again, could help the user better understand their target pattern and how far or close they have been exploring in relation to it.

User-related issues: despite the complexity component of the fitness function that favours combined dimensions with fewer variables and simple formulae, our method can still yield complex dimensions that are difficult to interpret, something that was also observed in our study with experts (section 5). Another issue is related to user fatigue which is a well documented problem in interactive evolution (Poli and Cagnoni, 1997). Other methods to collect user feedback need to be investigated. Our controlled study (section 5) showed occasionally a chunking effect of evaluations to binary values ("good" or "bad") which does not seem to reduce convergence speed (number of generations evolved before a solution was found). Careful selection of a user evaluation

scale or method such as sketching (Shao et al., 2014) can help reduce user fatigue. There is indeed a trade-off between the accuracy of user evaluations and the cost related to user fatigue.

Guiding users in an exploratory visualization environment involves careful considerations of *what* views to propose, *when* to propose them and *how* to present them to the user. Thus far, we elaborated on a framework that combines automatic methods and user input in order to steer the user exploration (i.e. the *what* part), much work is still needed to establish *when* and *how* interesting views should be best presented to the user without interrupting or distracting them from their main exploration tasks.

7 Conclusion and Reflections

Besides the development of a complete EVE framework, and the experimental proof of its efficiency for various data exploration tasks, this paper proposes a full experimental analysis of an IEC system. Assessing convergence, versatility and user satisfaction is a very difficult task for interactive systems, that has been rarely addressed in a very systematic way in IEC. The visual analytics community has developed tools and methodologies for addressing such issues, which have been applied in our collective work on EVE, yielding important insights concerning the behaviour of a genetic programming system in an interactive and changing fitness landscape. It is now obvious, as we are dealing with dynamic landscapes, that pure mathematical convergence (as usually defined in EC) is less meaningful than adaptation behaviour for IEC systems. Interactive systems are often dealing with very small populations, for which premature convergence is a major risk: convergence may then be considered as a drawback. Maintaining an exploration and adaptation ability strongly depends on diversity management. For *EvoGraphDice* adaptation is also relying on a surrogate function, learned from past user interactions: the efficiency of this mechanism has been proved using a very simple scheme. More sophisticated surrogate functions may be interesting to explore and evaluate, in order to improve the versatility of the system. Another advantage may be to use the surrogate function as an underlying optimisation fitness, allowing the use of larger EA populations for which only the best individuals are presented to the user for evaluation.

There are many open questions for EVE systems. Regarding initialisation, for instance, we choose a PCA for providing a known initial environment for users that are not familiar with evolutionary approaches. This may not be the best choice for some datasets, in particular for non numerical ones, albeit the GP search space still remains convenient for providing combined dimensions. Another challenge that EVE tools are facing is the exploration of highly multidimensional datasets and the related big data issues, where all dimensions cannot be displayed in a single view. Evolutionary search may serve as a pre-selection tool to filter interesting dimensions for the user. The main difficulty is then to learn user preferences on data they cannot entirely view. Learning solutions proposed by systems such as VisAsist (Guettala et al., 2012) may be interesting, but rely on some a-priori and/or more sophisticated surrogate functions.

Finally, we highlight the possibility of collaborative EVE systems, and in particular crowd-sourcing ones. Crowd sourcing approaches are indeed very attractive to deal with very large and complex datasets like genomic databases. More generally, collaborative EVE systems may serve as a communication framework for multi- to many-user search, as a shared population allows to simultaneously maintain and compare various interesting solutions. In this context we may have to consider large populations, and a set of user-dependent surrogate functions to yield views adapted to each user.

The evaluation of an IEC system remains a difficult task, as these systems adapt to the user, but at the same time the user also adapts to the system. Getting a clear understanding of the subtle mechanisms of co-adaptation (Mackay, 2000) is challenging. The research domains of visualization and human-computer interaction not only provide tools to help understand complex datasets but they also have study methodologies that can help shed some light on how users interact with evolutionary systems. We expect the synergy between experts from the IEC and the visualization communities to bring forward advances in conducting optimisation with a human-centered approach.

References

Albuquerque, G., Eisemann, M., and Magnor, M. (2011). Perception-based visual quality measures. In *Proc. IEEE Symposium on Visual Analytics Science and Technology (VAST) 2011*, pages 13–20.

Banzhaf, W. (1997). *Handbook of Evolutionary Computation*, chapter Interactive Evolution. Oxford University Press.

Behrisch, M., Korkmaz, F., Shao, L., and Schreck, T. (2014). Feedback-Driven Interactive Exploration of Large Multidimensional Data Supported by Visual Classifier. *Proc. IEEE Symposium on Visual Analytics Science and Technology (VAST), to appear*.

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

Bezerianos, A., Chevalier, F., Dragicevic, P., Elmqvist, N., and Fekete, J.-D. (2010). GraphDice: A System for Exploring Multivariate Social Networks. *Computer Graphics Forum (Proc. EuroVis 2010)*, 29(3):863–872.

Boukhelifa, N., Cancino Ticona, W. G., Bezerianos, A., and Lutton, E. (2013). Evolutionary Visual Exploration: Evaluation With Expert Users. *Computer Graphics Forum (EuroVis 2013, June 17–21, 2013, Leipzig, Germany)*, 32(3):31–40.

Brown, E., Liu, J., Brodley, C., and Chang, R. (2012a). Dis-function: Learning distance functions interactively. In *Visual Analytics Science and Technology (VAST), 2012 IEEE Conference on*, pages 83–92.

Brown, E. T., Liu, J., Brodley, C. E., and Chang, R. (2012b). Dis-function: Learning distance functions interactively. In *IEEE VAST*, pages 83–92. IEEE Computer Society.

Brown, E. T., Ottley, A., Zhao, H., Lin, Q., Souvenir, R., Endert, A., and Chang, R. (2014). Finding waldo: Learning about users from their interactions. *IEEE Transactions on Visualization and Computer Graphics*, 99(PrePrints):1.

Card, S. K., Mackinlay, J. D., and Shneiderman, B., editors (1999). *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Carpendale, S. (2008). Evaluating information visualizations. In Kerren, A., Stasko, J. T., Fekete, J.-D., and North, C., editors, *Information Visualization*, pages 19–45. Springer-Verlag, Berlin, Heidelberg.

Chen, C. (2005). Top 10 unsolved information visualization problems. *IEEE Comput. Graph. Appl.*, 25(4):12–16.

Chen, M. and Hagen, H. (2010). Guest editors' introduction: Knowledge-assisted visualization. *Computer Graphics and Applications, IEEE*, 30(1):15–16.

Dang, T. N. and Wilkinson, L. (2014). Scagexplorer: Exploring scatterplots by their scagnostics. In *Proceedings of the 2014 IEEE Pacific Visualization Symposium*, PACIFICVIS '14, pages 73–80, Washington, DC, USA. IEEE Computer Society.

Elmqvist, N., Dragicevic, P., and Fekete, J.-D. (2008). Rolling the Dice: Multidimensional Visual Exploration using Scatterplot Matrix Navigation. *IEEE Transactions on Visualization and Computer Graphics (Proc. InfoVis 2008)*, 14(6):1141–1148.

Endert, A., Fiaux, P., and North, C. (2011). Unifying the sensemaking loop with semantic interaction. In *IEEE Workshop on Interactive Visual Text Analytics for Decision Making at VisWeek 2011*, Providence, RI.

Endert, A., Fiaux, P., and North, C. (2012). Semantic interaction for visual text analytics. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems*, CHI '12, page 473–482, New York, NY, USA. ACM, ACM.

Fernstad, S. J., Shaw, J., and Johansson, J. (2013). Quality-based guidance for exploratory dimensionality reduction. *Information Visualization*, 12(1):44–64.

Fischer, G. (2001). User modeling in human&computer interaction. *User Modeling and User-Adapted Interaction*, 11(1-2):65–86.

Fukumoto, M., Ogawa, S., Nakashima, S., and ichi Imai, J. (2010). Extended interactive evolutionary computation using heart rate variability as fitness value for composing music chord progression. In *NaBIC*, pages 407–412.

Grinstein, G. G. (1996). Harnessing the human in knowledge discovery. In Simoudis, E., Han, J., and Fayyad, U. M., editors, *KDD*, pages 384–385. AAAI Press.

Guettala, A., Bouali, F., Guinot, C., and Venturini, G. (2012). A user assistant for the selection and parameterization of the visualizations in visual data mining. In *Information Visualisation (IV), 2012 16th International Conference on*, pages 252–257.

Hayashida, N. and Takagi, H. (2000). Visualized IEC: interactive evolutionary computation with multidimensional data visualization. In *IECON 2000. 26th Annual Conference of the IEEE*, volume 4, pages 2738–2743.

Ingram, S., Munzner, T., Irvine, V., Tory, M., Bergner, S., and Moller, T. (2010). Dimstiller: Workflows for dimensional analysis and reduction. In *Visual Analytics Science and Technology (VAST), 2010 IEEE Symposium on*, pages 3–10.

Johansson, S. and Johansson, J. (2009). Interactive dimensionality reduction through user-defined combinations of quality metrics. *Visualization and Computer Graphics, IEEE Transactions on*, 15(6):993–1000.

Koza, J. R. (1992). *Genetic Programming* . MIT Press.

Leman, S., House, L., Maiti, D., Endert, A., and North, C. (2013). Visual to parametric interaction (v2pi). *PLoS ONE*, 8:e50474.

Llorà, X., Sastry, K., Alías, F., Goldberg, D. E., and Welge, M. (2006). Analyzing active interactive genetic algorithms using visual analytics. In *Proceedings of the 8th annual conference on Genetic and evolutionary computation*, GECCO '06, pages 1417–1418, New York, NY, USA. ACM.

Lutton, E. (2006). Evolution of fractal shapes for artists and designers. *IJAIT, International Journal of Artificial Intelligence Tools*, 15(4):651–672. Special Issue on AI in Music and Art.

Lutton, E., Pilz, M., and Lévy Véhel, J. (2005). The fitness map scheme. application to interactive multifractal image denoising. In *CEC2005*, Edinburgh, UK. IEEE Congress on Evolutionary Computation.

Mackay, W. E. (2000). Responding to cognitive overhead: co-adaptation between users and technology. *Intellectica*, 30(1):177–193.

Malinchik, S. and Bonabeau, E. (2004). Exploratory data analysis with interactive evolution. In Deb, K., editor, *Genetic and Evolutionary Computation GECCO 2004*, volume 3103 of *Lecture Notes in Computer Science*, pages 1151–1161. Springer Berlin Heidelberg.

Matkovic, K., Gracanin, D., Jelovic, M., and Hauser, H. (2008). Interactive visual steering rapid visual prototyping of a common rail injection system. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):

Matkovic, K., Gracanin, D., Jelovic, M., and Hauser, H. (2011). Interactive visual analysis supporting design, tuning, and optimization of diesel engine injection. *Proceedings of IEEE Visualization 2011 (Discovery Exhibition)*.

Mengshoel, O. J. and Goldberg, D. E. (2008). The crowding approach to niching in genetic algorithms. *Evolutionary Computation.*, 16(3):315–354.

Meyer, M., Sedlmair, M., and Munzner, T. (2012). The Four-Level Nested Model Revisited: Blocks and Guidelines. In *Proceedings of the VisWeek Workshop Beyond Time and Errors: Novel Evaluation Methods for Information Visualization (BELIV)*. ACM Press.

Mouradian, J. A., Hamann, B., and Rosenbaum, R. (2012). A general approach for similarity-based linear projections using a genetic algorithm. In *Proceedings of SPIE*, volume 8294, pages 82940L–82940L–12.

Nam, J. E. and Mueller, K. (2013). Tripadvisor^{N-D}: A tourism-inspired high-dimensional space exploration framework with overview and detail. *IEEE Transactions on Visualization and Computer Graphics*, 19(2):291–305.

Peng, W., Ward, M. O., and Rundensteiner, E. A. (2004). Clutter reduction in multi-dimensional data visualization using dimension reordering. In *Proceedings of the IEEE Symposium on Information Visualization*, INFOVIS '04, pages 89–96, Washington, DC, USA. IEEE Computer Society.

Perer, A. and Shneiderman, B. (2009). Integrating statistics and visualization for exploratory power: From long-term case studies to design guidelines. *Computer Graphics and Applications, IEEE*, 29(3):39–51.

Pham, T., Hess, R., Ju, C., Zhang, E., and Metoyer, R. A. (2010). Visualization of diversity in large multivariate data sets. *IEEE Trans. Vis. Comput. Graph.*, 16(6):1053–1062.

Poli, R. and Cagnoni, S. (1997). Genetic programming with user-driven selection: Experiments on the evolution of algorithms for image enhancement. In *Genetic Programming 1997: Proceedings of the Second Annual Conference*, pages 269–277. Morgan Kaufmann.

Rensink, R. A. and Baldridge, G. (2010). The perception of correlation in scatterplots. *Computer Graphics Forum*, 29(3):1203–1210.

Saraiya, P., North, C., and Duca, K. (2005). An insight-based methodology for evaluating bioinformatics visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 11(4):443–456.

Sedlmair, M., Heinzl, C., Bruckner, S., Piringer, H., and Moller, T. (2014). Visual parameter space analysis: A conceptual framework. *Visualization and Computer Graphics, IEEE Transactions on*, PP(99):1–1.

Sedlmair, M., Meyer, M., and Munzner, T. (2012). Design Study Methodology: Reflections from the Trenches and the Stacks. *IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis)*, 18(12):2431–2440.

Seo, J. and Shneiderman, B. (2005). Rank-by-feature framework for interactive exploration of multidimensional data. *Information Visualization*, 4(2):99–113.

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423, 623–656.

Shao, L., Behrisch, M., Schreck, T., von Landesberger, T., Scherer, M., Bremm, S., and Keim, D. A. (2014). Guided Sketching for Visual Search and Exploration in Large Scatter Plot Spaces. In *Proc. EuroVA International Workshop on Visual Analytics*.

Smith, I. (2002). A tutorial on principal component analysis.

Stolper, C. D., Perer, A., and Gotz, D. (2014). Progressive visual analytics: User-driven visual exploration of in-progress analytics. *IEEE Trans. Vis. Comput. Graph.*, 20(12):1653–1662.

Takagi, H. (1998a). Interactive evolutionary computation : System optimisation based on human subjective evaluation. In *IEEE Int. Conf. on Intelligent Engineering Systems (INES'98)*, Vienna, Austria.

Takagi, H. (1998b). Interactive Evolutionary Computation: System Optimization Based on Human Subjective Evaluation. *INES'98*.

Thomas, J. J. and Cook, K. A. (2005). *Illuminating the Path: The Research and Development Agenda for Visual Analytics*. National Visualization and Analytics Ctr.

Ticona, W. C., Boukhelifa, N., Bezerianos, A., and Lutton, E. (2013). Evolutionary visual exploration: experimental analysis of algorithm behaviour. In Blum, C. and Alba, E., editors, *GECCO (Companion)*, pages 1373–1380. ACM.

Turkay, C., Jeanquartier, F., Holzinger, A., and Hauser, H. (2014). On computationally-enhanced visual analysis of heterogeneous data and its application in biomedical informatics. In *Interactive Knowledge Discovery and Data Mining in Biomedical Informatics - State-of-the-Art and Future Challenges*, pages 117–140. Springer.

Wilkinson, L. and Wills, G. (2008). Scagnostics Distributions. *Journal of Computational and Graphical Statistics*, 17(2):473–491.