Visual Analytics for Big Data – Examination Questions

This course is evaluated by means of a multiple-choice exam. The 10 questions below cover the material mentioned during the course. The study material is given by the slides discussed during the lectures.

The questions are grouped with respect to the theoretical visualization aspects they address. Each question has only **one** correct answer, given the context of the question. To find it, study the formulation of the question carefully. For each question, mark the right answer, by filling in the rectangle next to that answer.

During the exam, students are allowed to browse the course textbook "Data Visualization – Principles and Practice" (1st or 2nd edition). Any other written or electronic material besides this book is not allowed.

A. Aims and scope of visualization

1. Within data visualization, we distinguish several subfields, such as scientific visualization (SciVis) and information visualization (InfoVis). Concerning InfoVis, which is the correct characterization?

- **a.** □ InfoVis focuses on the creation of visualizations of multidimensional data.
- **b.** □ InfoVis focuses on the creation of interactive visualizations for data exploration.
- **c.** □ InfoVis focuses on the creation of visualizations for non-spatial abstract data.
- **d.** □ InfoVis aims at visualizing datasets which are typically larger than SciVis ones ('big data').

The correct answer here is (c). (a) is not correct since InfoVis also treats non-multidimensional data, e.g. trees or timelines. (b) is not correct since interactive visualizations are also used in exploration of spatial continuous data, such as treated by SciVis. (d) is not correct since both SciVis and InfoVis can examine small or big data equally.

2. Resampling is a critical problem for InfoVis applications, as these may involve attributes which are not of the quantitative or integral types. Consider the case where we need to aggregate N values v1….vN of a categorical attribute, which is encoded (stored) as an integer value. The result of the aggregation should be a categorical value in the same domain as the values v1…vN. Which is the best strategy to use to compute the value v of the aggregated attribute?

- **a.** □ Use averaging, i.e. define v as the average of the values v1…vN.
- **b.** □ Define v to the median of the set of values v1…vN.
- **c.** □ Set v to the histogram of the values v1…vN.
- **d.** □ Set v to the categorical value whose frequency, in the above histogram, is largest, if this frequency is above a given threshold. If the frequency is below that threshold, set v to some default 'unknown' value.

The correct answer here is (d). The answers (a) and (b) are wrong since we cannot do math on categorical values to compute averages or medians. The answer (c) is also not correct since the output of that operation is a histogram, and the question required the output to be a single value. The answer (d) delivers a single value in the same categorical domain as the inputs, or the reserved value 'unknown' in case we do not have enough observations with the same categorical value to be able to say confidently that that value is representative for the input.

B. Visual encoding

3. Colors are typically represented, in visualization, in the RGB and HSV systems. The two systems are dual, in the sense that a RGB color can be represented in the HSV space and conversely. Given this, which of the following statements is true?

a. □ The hue of black (in the HSV system) is undefined.

b. □ The mapping between RGB and HSV can be described by functions that are linear in the R, G, and B coordinates.

c. □ Not all RGB colors have an HSV equivalent.

d. □ Not all HSV colors have an RGB equivalent.

The correct answer here is (a). Indeed, given the definition of hue and/or the formulas used for converting RGB to HSV values, all colors having zero luminance (V=0) are rendered as black. Hence, hue is undefined for zero luminance. The answer (b) is not correct, the mapping of HSV to RGB and/or back is not linear, as it includes a mapping between polar and Cartesian coordinates. The answers (c) and (d) are also not correct – for any RGB color, we can compute a HSV color and conversely (this is e.g. precisely what tools do which convert colors input by users in HSV to the RGB needed by graphics libraries).

4. Different colormaps exist for different attribute types, such as categorical, ordinal, integral, and quantitative. Which of the following statements is true?

a. □ Grayscale and heat colormaps are good options for all attribute types.

b. □ No categorical colormap can use a single hue design.

c. □ Some quantitative color maps do not make sense for integral data.

d. □ No ordinal colormap can use a multiple hue design.

The correct answer here is (b). (a) is not correct, since it would imply that a grayscale colormap is good for categorical data; this is not true, since grayscale colormaps imply ordered data and categorical data is typically unordered. (b) is correct: If we have a single-hue colormap, then colors can vary only in saturation and/or brightness. Such a colormap would then contain an implicit ordering of the colors, e.g. from dark to bright, and we just concluded that categorical data is typically not ordered. (c) is not correct: integral is just a 'subtype' of quantitative (continuous) data; the differences of the two data types are only in terms of resolution and interpolation ability. None of these aspects influences the kind of colormap we can use for these data types. (d) is also not correct; divergent colormaps actually do use a two-hue or three-hue design in their construction, and they are used for ordinal data.

5. Different visual variables can interfere with each other when used together in a visual encoding scenario. For instance, color and transparency interfere when both used for drawing a scatterplot. Consider such a scatterplot, where we have three quantitative attributes, which we would like to map to the size, color, and transparency of the scatterplot points. Which is the best scenario in terms of diminishing interference?

a. □ Map each attribute to size, transparency, and color respectively, but use a single hue colormap.

b. □ Map two attributes to size and color, using any desired colormap; refrain from mapping anything to transparency.

c. □ As (b), but draw the points from the largest first to the smallest last.

d. □ As (c), but use a single hue colormap.

This question is more complicated, since it asks for the "best scenario". As such, we proceed by eliminating answers which clearly lead to poor quality results, and ultimately remain with the best scenario answer. The goal here is to diminish (probably not fully eliminate) interference, i.e. to let the three quantitative attributes be readable independently on each other in the resulting scatterplot. Answer (a) is the worst, since transparency will interfere wit the perception of colors, even in the presence of a single hue colormap. Let s,t, and c be the three variables mapped to size, transparency, and color. Then in an area having samples with high values for s we will see many overlapping disks, and due to blending we will see there a color which suggests large values of c, even though c may actually be small. Similar problems occur even in areas with samples having low s values, as long as the spatial sample density is high. Moreover, large disks (s values) having low t (transparency) values will completely occlude samples behind them. Answer (b) is better, since it eliminates the interference between color and transparency. Answer (c) is better than (b), as large disks (s values) get a low chance to occlude smaller disks. Answer (d) is better than (c) since it removes the visual disorder created by many partially overlapping and multiple-hue disks. Hence, the answer is (d).

C. Visualization techniques

6. Pie charts are an useful tool for displaying a set of values which sum up to one (a so-called partition of unity). However, they have several challenges. Which of the following describes a relevant such challenge for pie charts?

a. □ A pie chart is visually less scalable than a bar chart; that is, given N data values, we can perceive less well these values if drawn via a pie chart than via a bar chart (both charts have the same screen-space dimension).

b. □ A pie chart is less effective than a bar chart for displaying a set of ordered data values.

c. □ Pie charts cannot represent categorical data values.

d. □ Comparing two (or more) pie charts to see trends in data values is harder than comparing two or more bar charts (for the same set of data values).

The correct answer here is (d). (a) is not correct – the visual scalability of both a bar chart and a pie chart are very much the same. In more detail, for both charts one can imagine cases where it is hard to perceive very small values. However, in the average case, both have roughly similar scalability. (b) is also not correct – we can sort both bars and pie sectors in both diagrams. (c) is strictly speaking not correct – we can encode categorical values in a pie chart by means of color, like for any other chart. I this sense, pie charts can show categorical data values as well as any other chart. (d) is indeed correct *and has to do with the fact that we cannot control the angular position of individual pie sectors in all the charts to compare, whereas the comparable factor for bar charts, i.e. the bar position, can be easily controlled.*

7. Table data and tree data can be thought as dual representations. That is, we can encode an attributed tree as a table, and given a table, we can construct a tree representing its data values. Which of the following is true regarding the conversion process of tables to trees?

a. □ A tree can be converted to a table automatically (that is, without any user decision). However, converting a table to a tree requires some user decisions during the conversion process.

b. □ The tree resulting from the conversion of a table depends on the order of the data columns in the table.

c. □ We can convert a table to a tree only if the table contains just categorical values.

d. □ The conversion of a tree to a table is unique. That is, given a tree, there is a single table (having the same order of columns and rows) generated by this process.

Recall the corresponding lecture slides. The correct answer here is (a). Indeed, we can convert a tree automatically to a table: For every level in the tree, we define an attribute (table column) having e.g. as many categorical values as nodes on that level in the tree. For every node in the tree, we create a table row containing all the above-defined attributes, whose values are given by the position of the node in the tree. A node only has table values defined for all levels above and equal to it in the tree (for all other attributes, it has an 'undefined' value). Alternatively, we can simply think of a table listing for each node its parent (if not the root node) and all of its other data attributes (if any). Converting a table to a tree however requires user decisions in (1) the order in which to map the columns to the tree levels and (2) how to group or bin cell values to create children nodes. (b) is strictly speaking not correct – the tree depends on the order of treating (mapping) the data columns, not on the order of the data columns themselves in the table (this order is actually irrelevant, a table is a set of named columns, not an orderd sequence of named columns). (c) is not correct as the material presented during the lecture showed (we can use binning to create trees from quantitative data tables). (d) is not correct, as already explained above (the order of mapping columns to tree levels will yield different trees).

8. Parallel coordinate plots (PCPs) are one of the most important tools for visualizing multidimensional data. However, despite their power, they have several limitations. Which of the following limitations is true?

a. □ PCPs cannot show the distribution of values along a specific coordinate axis (dimension).

b. □ PCPs do not allow one to study the precise values of a specific observation (data sample) along all its dimensions.

c. □ The result displayed by a PCP depends on the order in which we draw the observations.

d. □ PCPs require user interaction if we want to clearly compare any pair of dimensions.

The correct answer here is (d). (a) is not correct as shown during the lecture (the 1D set of points representing the intersection of the polylines with some coordinate axis is precisely this distribution of *values along that axis). (b) is not correct – the shape of a polyline representing a data sample is giving precisely all the values of that observation. (c) is not correct, at least not in the case of typical, generic, PCP plots which will not use color coding to map additional data to the polylines. In that case, the drawing order will not change anything, even if transparency is used to convey the density of observations. (d) is correct: if we want to compare two dimensions which are placed very far apart from each other in the PCP, we'd best rearrange the PCP so that these two dimensions become neighbors.*

9. Projections are another key tool for visualizing multidimensional data. In particular, they are used to detect the presence of groups (clusters) of similar observations in a given dataset. For this task, which of the following is true?

a. □ If we see a clear segregation of points in a 2D projection, then a clear segregation of data values must exist in the original high-dimensional space.

b. □ If points are clearly segregated in the original high-dimensional space, we will see a clear segregation of their projections in a 2D projection.

c. □ Neither (a) nor (b) are true.

d. □ Both (a) and (b) are true.

Key to answering this correctly is to first understand that, when referring to '2D projections', we refer to projections which are implemented and applied reasonably well. That is, the question does not cover the case of someone applying a projection algorithm wrongly. In this case, the correct answer is (a): Indeed, consider the poorest projection (PCA) in terms of preserving, on average, the distances between data points. If we see two clearly separated clusters in a PCA plot, it is clear that something makes these data values very different from each other in n dimensions. More generally, any projection aims to preserve the data structure (distances or neighborhoods). While the entire data structure may not be preserved, no projection will 'create' new data structure (like clusters) out of no data structure. Hence, (a) is the correct answer. (b) is not correct, as explained above: Some low quality projections (e.g. PCA) may not fully preserve the high-dimensional data structure; therefore, we may see a 2D projection with no clear clusters, whereas these clusters do exist in nD. By implication, (c) and (d) are both wrong.

10. Projections are subject to various errors. Understanding such errors is important for being able to utilize a projection to make inferences about the structure of the high-dimensional data. Consider a projection of such a dataset which delivers an undiferentiated 'blob' of 2D points. If we encounter such a situation, which is the best next step to follow in our visual analysis?

a. □ An undiferentiated blob tells us that the high-dimensional data is not strongly segregated into clusters. We don't need any subsequent analysis to confirm this.

b. □ We should fine-tune the parameters of the projection algorithm, or alternatively change the projection algorithm, until we obtain a clear segregation of the projected points into groups. Only then can we next reason about the structure of the data.

c. □ We should use fewer features (dimensions) to project the data so as to obtain a clearer segregation into clusters.

d. □ We should check the presence of false neighbors and missing neighbors in the projection. If these are numerous, we cannot trust the projection and should perform (b). Else, we can trust the projection and we can follow the outcome of (a).

This question addresses the typical run of a visual analytics pipeline where we see something (the blob), we form a hypothesis (data is undifferentiated), and then we try to validate the hypothesis. (a) is not correct since, as explained for question 9, a projection only aims, but cannot guarantee, to preserve fully the high-dimensional data structure. Hence, when we see a blob, it may be that (1) the projection lost a lot of the data structure, or (2) the data is indeed undifferentiated in nD. (b) is also not a correct answer: Doing (b) implies we already know that the data is separated into clusters in nD, so all we want is to tweak the 2D view to reflect that; but we do not know this a priori, we actually want to use the visualization to confirm if data is segregated or not. (c) has the same problems, if not more, as (b): Using fewer dimensions can indeed force the appearance of some clusters in the projection, but this is not our main aim in the general case (that is, in the case we don't simply know that data is strongly segregated). (d) is the correct answer: Before doing anything with the projection, we should first of all see if we can trust it by looking at error metrics. If we trust it, then the data is indeed undifferentiated (a blob) and we're done. If we see many errors, then we can indeed follow (b) i.e. tweak the projection parameters aiming to reduce its errors, and next repeat the assessment.