Visual Analytics for Big Data – Theory Assignment

1. Introduction

The goal of this document is to describe the theory assignment of the course *Visual Analytics for Big Data*.

The assignment consists in the *study*, *analysis*, *interpretation*, and *refinement* of a visualization design aimed at solving a concrete real-world problem. The goal of the assignment is to show how students have understood the many technical, theoretical, and practical points pertaining to the construction and evaluation of (good) information visualization solutions. Students that successfully complete the assignment are, thus, provably able to operate as designers, evaluators, and end users of (good) information visualization solutions further in their career.

The assignment does not test *implementation-level* skills – that is, students are not supposed to be able to create running information visualization software tools. The set of skills required to do so falls within a different branch of studies (software engineering). The two aspects mentioned above – design and evaluation, and implementation – are complementary. The current assignment covers the former two, not the latter.

Central to the execution of this assignment is the concept of **motivated design and analysis**, which can be described as the *detailed discussion and presentation of all aspects of an existing visual design aimed to solve a concrete problem*. To be useful, such a discussion has to cover *all* aspects of the design of a visualization (starting with the application domain and questions to be answered, and ending with the actual use/interpretation of the resulting visualization); should discuss the obtained findings and insights in *detail*; should *motivate* the reached conclusions by using theoretical/technical evidence presented during the course; and should present ways to improve in a motivated way, highlighting the advantages and disadvantages of the proposal.

2. Assignment Overview

When we **design** a new visualization for a given problem, we are faced with many choices, ranging from which questions we want to solve, who are our typical users, what kind of data we have, how we want to encode data into visual variables, how we fine-tune all the visual design elements (e.g. labels, legends, annotations, background), and how we evaluate the quality of the produced solution. This process works top-down – we start with the problem, we end with the evaluation of the solution. The design process is summarized by the following diagram (for details, see Module 6 and "A Nested Model for Visualization Design and Validation" (T. Munzner, IEEE TVCG vol. 15, no. 6, 2009, pp. 921-928).

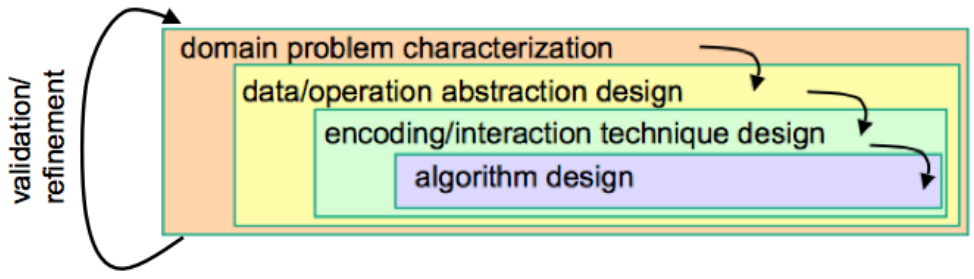


Figure 1: Visualization design process

The result of this process is the creation of a visualization application (or visual design) that works following the well-known visualization pipeline shown in the image below. At each pipeline step, specific operations are done on the data, and specific design choices are made. As such, the quality of the entire solution can be decomposed in terms of the suitability of the design choices having been made at each step. Additionally, the quality of the entire visualization can be

measured in terms of how well can users answer questions on the input of the pipeline (data) by examining the output of the pipeline (images). This process is described during the course in terms of *inverse mapping*. As discussed there, the overall *quality* of any visualization is strongly linked to the completeness and ease of performing this inverse mapping by typical users.

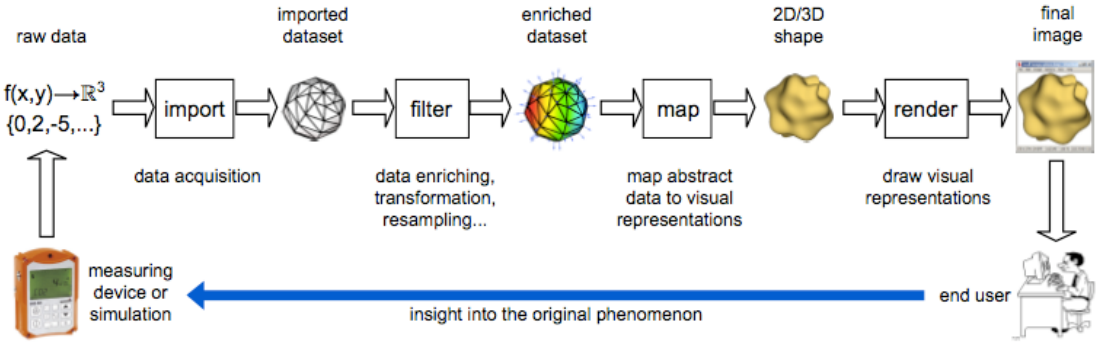


Figure 2: Visualization pipeline constructed by visual design process

As Figure 1 shows, the design of any visualization is an **iterative** process: Once a first design is obtained, it is evaluated and then incrementally refined to improve its quality. This assignment can be understood in terms of executing iterations 2 (and next) of the design process:

- We start with an existing visual design, executed by other people.
- We next evaluate the design (in terms of the underlying problem), discuss the design's motivations, discuss its strong and weak points, and propose improvements.

Several points are essential to understand here (in the text below, 'we' refers to students executing this assignment):

- **Application domain:** We are not the 'owners' of the application domain, nor are we experts in that domain. As such, the first step of the assignment is to choose an application domain that we understand reasonably well, so we can identify ourselves as much as possible with typical users and specialists in that domain. This includes understanding the data and questions typically used in the respective domain.

- **Visualization:** We are not the original 'creators' of the visualization(s) used for the above-mentioned application domain. Our role is to understand how these were created, why they were created in that way, which are their strong and weak points, and propose how to improve the latter. Since we did not build the visualization(s) ourselves, we need to do some amount of 'reverse engineering' work – that is, understand what happened during the actual design process, and why. Note that this way of working is typical for visualization designers: In many cases, the original designers of a visualization are not available any longer; the design decisions are not documented; and we still want to improve the visualization. We then do precisely the kind of work covered by this assignment.

- **Materials:** We usually do not have access to the actual design documents or implementation of the visualization we study. We do have access to the presentation of the visualization results (images), brief comments on these (in terms of explanations of the results provided by the design authors or third-party commentators), and usually the running application software. As such, we have to base all our analysis and work on the information contained in these materials.

3. Steps of the Assignment

The assignment contains five steps, as follows:

- Step 1: Choose an Application Domain and Visualization Solution
- Step 2: Problem Description
- Step 3: Data Description
- Step 4: Visual Design Study
- Step 5: Description of Findings

These steps are best executed in the order listed above. Of course, the entire process can be done iteratively – produce a first raw draft of the five steps, then refine the level-of-detail and explanations in each step.

The five steps are detailed below. As supporting example to explain each step, this document uses an existing problem and visual solution: The 'map of the market' visualization provided online at https://finviz.com/map.ashx .

4. Step 1: Choose an Application Domain and Visualization Solution

The first step of the assignment is to choose a *problem domain* and existing *visualization solution* for that problem domain. Both elements are needed: We need a problem domain so we can determine which are typical users, what are the questions posed by these users, and what is the data under study. We need an existing visualization *solution* so we can analyze its design, identify how well it answers the above-mentioned questions, where are improvement points, and how we think we can improve upon these.

You can choose *any* pair of problem domain and existing visualization solution from the very wide set of information visualization applications out in the public or scientific arena. However, the problem (and solution) have to be of a minimal complexity to pose interesting challenges from a visual design perspective. To make sure this happens, follow the next checklist:

- **Data complexity:** The visualized data should be of minimal size to warrant a complex visualization (a few hundreds of data points, and several attributes per data point). Datasets having attributes of different types (e.g., quantitative, categorical, text) are preferred, since they create more visualization challenges. Similarly, datasets which are time-dependent are preferred, since they add the extra complexity of visualizing how data structure changes.

- **Problem complexity:** The questions posed by the application domain should be of a reasonable minimal complexity to warrant the use of visualization. For example, asking "what is the largest value in a set of data values" is a simple question, that basically any visualization (even a poor one) would likely answer very easily, so there is no point in discussing or improving visual design here. Typical complex questions include finding the correlation of several attributes, finding trends (how data varies in time), and finding groups of data points that share different properties.

- **Access:** You of course need to have a visual solution that can be executed on actual data, i.e., a visualization tool implemented in a system that runs on your actual computer.

Good starting points for choosing an application domain and visualization solution:

- **Envisioning Information** (E. Tufte, Graphics Press, 1990): This book contains about 100 examples of historical information visualization examples. The examples are only discussed briefly (in terms of a few of their strongest advantages/limitations), which leaves enough space for this assignment (detailed discussion). For instance, there is, in general, no discussion in the book of the exact questions to be answered (by a visualization), the data type being treated, and only a very limited discussion of the visual design. Note that only a subset of the examples are actually suitable for this assignment – specifically, those which present a reasonable *amount* of *quantitative* data. The full book is provided in PDF in the online materials.

- **Visual Display of Quantitative Information** (E. Tufte, Graphics Press, 2001): We provide subsets of the pages of this book in the online materials (about 50% of the full book). The visualization examples in here are very similar, in terms of level-of-detail of the description, to the ones in Envisioning Information. As such, the same rules for selecting a good example apply.

- **Financial Visualization** (online resource, https://finviz.com/map.ashx): This online tool allows exploring various aspects of the financial markets. The tool visualizes a quite large amount of data (thousands of data points, several time periods), covers many types of attributes (quantitative, categorical, time-related, labels, hierarchy), and features several visualization techniques (treemaps, color coding, labeling, time series, interactive selections and explorations). It is a great example of easy-to-use, though reasonably complex, visualization application. This is also the application used to illustrate the rest of the assignment. If you use this application, make sure that your work will go in *significantly* more details than presented in this document next (when we use the same application as an illustration to explain the assignment steps).

- **Gapminder World** (online resource, http://www.gapminder.org/world): This online tool allows exploring over hundred different datasets related to various indicators measured over the entire world and over different time periods. The available datasets are organized in several categories (child health, disasters, economy, …) – see the *Open Graph Menu* option in the tool. While the proposed visualizations are relatively simple (scatterplots, bubble charts, and geographical maps), the interaction options and number of available datasets let users create hundreds (if not thousands) different visualizations. If you use this data source, make sure to study the online tutorial of the tool first, and also to explore *several* datasets.

- **Museum Artifacts** (online resource, http://historywired.si.edu): This online tool allows exploring hundreds of historical artifacts present at the National Museum of American History. Artifacts can be organized per type (transportation, home, military, …), period of manufacturing, and allow being viewed in terms of images. The visualization is similar to the Financial Visualization and Gapminder World, though has several different design points. If you use this data source, it is very useful to also study either the Financial Visualization or Gapminder World, so you can compare the different design decisions.

- **Music Timeline** (online resource, http://music-timeline.appspot.com): This online tools allows exploring hundreds of musical genres, covered by (tens of) thousands of artists, in terms of millions of songs. Artifacts (songs, genres, artists) can be organized in various ways, with a visual design centered on a timeline metaphor. The visualization serves to answer questions concerning the varying popularity of genres/artists over time, how these relate to each other, and also to discover less known artists/genres/songs that are related to a given artifact (artists/genres/songs) of interest. This example is very interesting as it covers a quite large database, and uses a mix of visualization techniques (timelines, flows, relational data, animation).

- **Tableau Public** (online resource, http://public.tableau.com): Tableau is discussed in detail during the practical assignment. As explained there, a myriad of visualizations and their accompanying datasets are shared online via Tableau's public cloud. Pick from here an interesting one – also pay attention for example to the so-called 'featured' visualizations that are shown in Tableau Public, once you open the tool on your desktop computer.

After you choose a problem and visualization (either from the above sources, or other ones), inform the lecturer (prof. A. C. Telea) about your choice, so this can be *validated*.

Example of a good choice

Why is the Financial Visualization a good choice? Several reasons are listed below:

- The application domain (while a bit technical) is quite easy to understand for everyone: It shows the ups and downs of stocks of various companies, organized in industry sectors, over

several geographical regions and periods of time. Even for a non-specialist, it is clear why seeing such information is important (e.g. find out well-performing stocks or sectors, comparing how different stocks perform, finding trends in stock prices, etc).

- There is a lot of data being visualized (thousands of stocks, information recorded for several years). Data is of various types (quantitative, time series, categorical, hierarchical, and text). Many visualization techniques are used (scatterplots, timelines, treemaps). As such, the visual design is indeed challenging and complex.
- The application is interactive – you can actually test different scenarios, and try to answer different questions, simply by using the controls of the tool.
- The application is easy to use – you only need a recent web browser to start exploring.

Below we see a typical snapshot of this application:

5. Step 2: Problem Description

The second step of the assignment is to describe the *problem* that the visualization selected at step 1 actually tries to solve. Note that, in most cases, a visualization does not try to solve a *single* problem, nor does it try to solve that problem (or problems) *completely*. Indeed: Any application domain is concerned with several (typically inter-related) problems; and there is no single solution that covers all such problems completely.

5.1. Questions

Hence, the best way to approach this step, is to use a *bottom-up strategy*:

- Try to list a set of different *concrete questions* that you see that the visualization tries to answer. These are simple to find, as they focus on low-level tasks (e.g: find the stock with highest growth during a day);
- Next, group related questions into *higher-level questions;*
- Finally, try to summarize all these higher-level questions into a *single* phrase or question.

For example, for the Financial Visualization, a (simplified) execution of the above procedure gives us:

- **Concrete questions**
  - Which is the price variation of a given stock on a given day? Which is the market capitalization of a given stock on a given day?
  - Which are the largest (market cap) stocks on a given day? Which are the strongest gainers (or loosers) on a given day?
  - How are stocks in a given industry sector comparing to each other?
  - Which industry sector is doing best (worst) on a given day?

  - Which are the strongest gainers (loosers) during the last year?
  - How are different stock markets across the world comparing to each other?
  - How is a given stock doing over an entire time period (e.g. a year)?
- **Higher-level questions**
  - How are stocks grouped per industry sector, or worldwide?
  - How can we compare individual stocks (or groups of stocks) from the perspective of price change and market cap?
  - How can we find outlier stocks (which perform differently than a selected group)?
  - How can we see trends (price variations in time)?
- **Summary**
  - How can we create a 'map of the market' which lets us compare many stocks from the perspective of price, market capitalization, sector, and time?

Besides describing the questions addressed by the visualization for the chosen application domain, you should also characterize the domain along several other dimensions:

5.2. Users: Which are the typical users in this application domain who would benefit from this visualization?

In the case of the Financial Visualization, the answer will include financial experts, traders, but also the general public which has some interest in the stock exchange.

5.3. Purpose: What is the main purpose of the visualization?

Recall the purposes of visualization (Module 1): Confirm the known / discover the unknown; and analysis / presentation. Which of these apply to your chosen visualization? In the case of the Financial Visualization, the purposes seem to be somewhere at the intersection of *analysis* (see the questions listed above) and *presentation* (get a general idea of 'how the market does'). Of course, you should refine the discussion further in your assignment.

6. Step 3: Data Description

The third step of the assignment describes and characterizes the *data* being used in the visualization. To complete this step, study Module 2.

The purpose of describing the data being used in the visualization under study is to understand which are the constraints with which the visual design has next to work. For instance: Consider that a visualization has to display a single set of 1000 temperature measurements. Clearly, data size does not pose a major problem here, so many visual designs can be used. Also, data is quantitative, so we have to use next techniques that are suitable for quantitative data (like, for example, encode values into position, use continuous colormaps, etc). Consider, in contrast, a visualization that has to display 100 different types of measurements, each of them having 100 measurement points. Clearly, data size does pose now a problem (we have many data points and many dimensions). Also, if the data types are of various attribute kinds (continuous, ordinal, categorical), we have to use different kinds of visual encodings for each.

Example

The Financial Visualization is a very interesting case: While it appears simple, it uses a quite sophisticated dataset. Data is basically organized into stocks. Each stock has many dimensions (name, price, price variation, market capitalization, industry sector, stock exchange). These dimensions, or attributes, are of several different kinds (text, continuous, categorical). Moreover, each stock has measurements defined over many time moments. Also, stocks are organized in a hierarchical fashion (stocks are grouped under industry sectors, and sectors are grouped under different stock exchanges).

The size of this dataset is also quite large. Try to estimate the size, by considering

- How many stock exchanges does the visualization cover?
- How many industry sectors does a stock exchange have (on average)?
- How many stocks does a sector have (on average)?
- How many time moments does the visualization cover (on average)?

Giving a concrete estimate of all above tells us what is the total data size that the visualization aims to present. In turn, this will be very useful to explain the visual design choices (see next step).

After the above, give a formal description of the *data structure* used in the visualization. Simply put, this means to describe a model, in terms of observations, attributes (dimensions), attribute types, attribute value ranges, and relations between the above, which would capture the entire data in the application. In (yet) other words, this is equivalent to proposing a (minimal) database schema that would capture the data at hand.

The challenge here is to find the *appropriate* data model: Since you don't have access to the internals of your chosen application, you don't know how data is actually represented/stored; and, of course, the same data can be represented in several ways. However, in most cases, there are typically only a small number of schemas that truly make sense for representing a given dataset.

Hints on how to proceed:

- Try to think of the data as a set of *tables*. Define the columns (dimensions) in each table and the observations (records) as well. Define the relations between tables (equivalent to foreign keys in database terminology).
- When doing the above, try to minimize *redundancy*. That is, prefer a design with a few tables with many columns and few replicated data entries (values) between them to a design with many tables, replicated data values between tables, and many foreign keys.
- If your data is *time-dependent* and *multidimensional*, a typical schema consists of several tables, one per time-stamp.
- If your data contains *relational* information, model that separately as a graph. You can also model relational data as tables, but it is far less intuitive and helpful for the next steps.
- Describe the *attribute types* in terms of the material discussed during the course (quantitative, categorical, ordinal, integral, text, relational). Where applicable, you may introduce new attribute types (e.g. image, sound).

7. Step 4: Visual Design Study

The fourth step of the assignment discusses the visual design being used in the visualization under study. The purpose of this step is threefold:

- Explain how the data discussed at step 3 is actually *mapped* (encoded) to visual variables (at what does one look to see specific data attributes)
- Explain how the *questions* determined at step 2 are actually addressed by the visualization (what does one need to do / where does one need to look to answer a specific question)
- Find strong and weak points of the visual *design*, and propose *improvements* to the latter.

To study a visual design, cover the following elements:

7.1. Type of visualization

First, we have to decide which kind of visualization we are looking at: SciVis, InfoVis, or Infographics. As discussed in Module 1, each such visualization type is optimal for data having different characteristics and different kinds of users. Use the material from steps 2 and 3, and the actual visualization under study, to determine the visualization kind.

For example, the Financial Visualization is clearly of an InfoVis type (it treats non-spatial data; attribute types are not only quantitative but also categorical and relational).

7.2. Visual encoding

Any visualization encodes the different dimensions of its input data into different so-called visual variables (see Module 4). For the visualization under study, determine how the data dimensions you found at step 3 are encoded into the present visual variables. Argue, for each encoding, if it is indeed a good one (from the perspective of the encoding principles discussed in Module 4). Note that a single visualization can offer multiple encodings – for example, the same data attribute can be encoded simultaneously into color and size (see again Module 4). Also, in the case of an interactive visualization, users can choose how to encode data (they can for instance select which data attribute to map to color, or they can select whether to map a given data attribute to color and/or size).

Depending on the particular kinds of data attributes present in the visualization, the proposed encoding may be optimal or not. For instance, if we use color to encode data, a quantitative variable should be typically encoded with a continuous colormap, while a categorical variable should be encoded with a categorical colormap. A good encoding depends also on the number of different values that we want to show at the same time, and the total number of data attributes we want to show at the same time.

Specific questions you should answer in this part are listed below:

- Which are the visual variables that are used for data encoding in the visualization?
- How is each data attribute encoded into one or several visual variables?
- What are the color choices being used (if color is used in the visualization)?
- How are annotations being used (labels, markers, legends)?
- Is transparency used, and if so, is it used well?
- Is the background chosen well (if any)?
- Do we have ambiguities in the visualization (e.g. due to poor color choices, visual clutter, overdraw, 3D occlusion, bad texture choices, undesired perceptual effects)?
- Is the encoding clear (if I want to answer a specific question, is it clear where to look, and is there enough information being shown so I can indeed answer that question)?
- Are there inverse mapping problems? (several visual values indicate different data values; certain data values do not correspond to any visual value; perceptual distances between visual values do not reflect well actual distances between data values)

Overall, the answers to the above questions, when taken together, answer a single key question: *Is it easy to perform the inverse mapping on this visualization to go from the image back to the data?*

Example

Without being exhaustive, below are a few elements that discuss the visual design of the Financial Visualization:

- The visual variables being used are: position, size, color, and shapes (text labels).

- Position is used to map the industry sector, and size is used to map market capitalization. Together, the two are linked by the basic squarified treemap visualization technique. Position is actually used separately to show the variation in time of the price of stocks in an industry sub-sector (obtained when clicking on the map on the desired subsector). This is a classical example of a small multiples visualization using line charts as basic element. Separately, position is also used to show a scatterplot of two quantitative variables of choice (e.g. market capitalization *vs* gains) in a separate view (selected by the 'Bubbles' option top-left). Color is used to map price variation, using a divergent red-black-green colormap. Color

is also used to highlight specific elements of interest, when we select a stock or brush the map interactively.

- Annotations are used on several levels, to indicate the names of industry sectors and subsectors and individual stocks, and exact price variations (in percentages). A color legend, using 7 colors for 7 sub-intervals, is displayed.

- Transparency is used at several points. For example, the scatterplot (bubble plot) uses transparency to overlay the data points atop of the graph axes, so the latter are better visible. Also, transparency is used to blend items together in this plot (so the luminance is somehow indicative of the number of overlapping data points). However, the blending technique being used is not very clear: For instance, brightness in the resulting plot is not a clear indicator of the number of overlapping data points, since brightness is also used by the colormap employed to show data values atop of the data points in the scatterplot.

- The visualization uses a dark gray background. This is a good design choice, as all of the other visual elements are bright, so it maximizes contrast. Also, the dark background contrasts well with the used color maps, which use bright to indicate important values and dark to indicate neutral values. As such, the neutral values seem to fade into the background.

- There are some ambiguities in the visualization: For example, the axis scaling for the bubble plot does not always refresh upon range changes, which leads to poorly scaled plots. The transparency used by bubbles interferes by creating brightness patterns which clash with the use of brightness in color mapping. The color maps use a few discrete colors (one per range) – it is not clear why a continuous color map has not been used, since we have quantitative data in many cases. Also, large bubbles can obscure small ones, so we get a possibly wrong insight by looking at this bubble plot. The labels used in the treemap to indicate industry sectors, sub-sectors, and individual shares are quite similar – so it is not pre-attentively easy to tell which kind of category does a given label encode, or to quickly spot, for instance, all the subsector names among the total set of labels being shown. The treemap uses thin black borders to outline cells. When visualizing large amounts of data, this creates black blocks which resemble stocks having zero variation (encoded also in dark gray). Actually, these black blocks are no stocks, they are just very small treemap cells for which we only see the borders being drawn.

- The encoding is relatively clear, as different visual variables are used consistently for the different data attributes. However, there are limitations. For example, in the treemap, size encodes market capitalization. As we can compare sizes of rectangles relatively easy, we can tell which of 2 stocks has a larger capitalization (though, not precisely how much larger), or which are the stocks with highest capitalization across the whole market. However, we cannot tell the *exact* capitalization of a stock (in billions of USD, for example), using this encoding. Also, when changing time ranges, the sizes of the treemap cells stay the same. This can be seen as good, since it lets us compare easily different time moments (cells stay of the same sizes and places in a treemap). However, one expects the market capitalization of stocks to *change* over time. Where is this change shown? And, if the capitalization is assumed not to change, to which time moment do the values shown in the treemap refer?

Of course, many more design points can be analyzed for the Financial Visualization, and more details and insights can be found for each of the above-mentioned points.

7.3. Improvements

Based on the above analysis of the visual design of your selected visualization, you have identified several limitations. The last part of step 4 is to propose *improvements* in these directions.

To determine *what* to improve, consider your own analysis of the design limitations. To determine *how* to improve, consider the visual encodings and visualization techniques discussed during the lectures (Modules 4 and 5). For each proposed improvement, explain which (of the

identified limitations) it improves, how it does that, and argue why this proposal will not cause more problems (limitations) than it does remove.

Several points are important when judging whether an improvement makes sense:

- One can improve on a wide *range* – from small design details up to proposing a completely different visualization design. Both kinds of improvements are equally valuable: You should not try, for example, to propose only radically different designs, if small changes to the existing design do a good job in fixing problems. The key here is to clearly identify the problems, and then look for solutions *for those specific problems*. Hence, the reason to propose a specific technique is that it is effective in addressing a specific problem, and not just because it can visualize the data at hand in a different way. For example: We have identified the problem that treemap cells can create visual clutter, and we want to remove this problem. A simple way to do so is to use borderless treemaps, that is, encode the level of the cells in the treemap into shading (see the *shaded cushions* treemaps in Modules 3, 5). This fixes the issue, and does not introduce other problems. A wrong solution to the same problem would be to propose using *node-link tree layouts* to visualize the hierarchical data (see Module 5). Indeed, our data is hierarchical, so we can visualize it with a node-link tree design. But this visualization would not make much sense, as it only creates extra problems (lots of unused white space, low information density, clutter). These problems are much more serious than the limited issue of clutter created by small dark cell borders in the treemap.

- When presenting an improvement, support it by a *drawing*. To create such drawings, simply reuse snapshots of the various visualizations in the course slides, and indicate, with annotations, what would the elements in these drawings mean in the context of your current visualization application. You are not required to actually generate the 'correct' drawings showing your proposed visualization on the data in the studied problem. Indeed, this would not be possible, since you do not have access (in most cases) to that data, and this would also involve a completely different type of exercise (programming a visualization). When using an illustration to explain your new visualization proposal, try to be as specific and detailed as possible, in terms of how this will encode data, which will be its advantages (as compared to the existing visualization), and also its disadvantages. For example, using shaded cushions treemaps will remove the black-border problem discussed earlier, but will also remove the space used in the current visualization to place labels for the non-leaf nodes of the tree (industry sectors and subsectors). You can fix this by adding these labels using transparency (see example in Module 4) – but care should be taken here so you don't get visual clutter or undesired colors.

- When thinking of an improvement, consider again the original *questions* you identified in step 2. As explained there, the visualization you study can (a) answer them partially, or (b) answer a part of them. Both (a) and (b) present options for improvements. For (a), you consider using a different visual design that answers the *same* questions, only *better*. For (b), you consider using a different visual design that answers a *larger part* of the original set of questions. For example, the cushion treemap improvement suggested above is an improvement of type (a). Using a multidimensional projection (see Module 5) to show stocks as points in a scatterplot, and plot stocks that show similar gains/losses in time close to each other, is an improvement of type (b) – it answers the question "which stocks show similar behavior in time?" which is not a question really answered by the current treemap design.

8. Step 5: Description of Findings

This last step of the assignment simulates the usage of the visualization under study as if you were a *user* of it – that is, a receiver of the message it tries to communicate. The aim of this study is to concretely answer a few of the questions you already identified as being targeted by the visualization (step 2) by actually looking at the visualization generated interactively by the selected visualization tool.

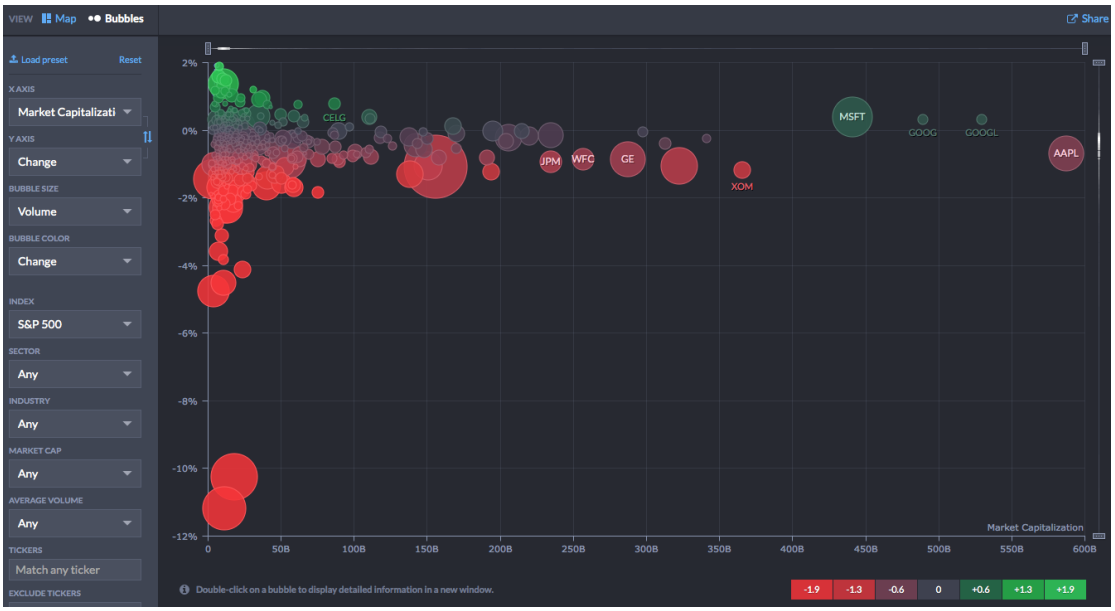
For this, act as an actual end user of the visualization:

- Select a few questions from the ones you listed in part 2
- Study the visualization, by running the tool, trying to find information that answers these questions
- Document the findings
    - Explain which are the answers you found for the questions
    - Explain the certainty you have that you found the correct answers
    - Explain how easy it was to obtain the answers, and which were the challenges

The above process is closely related to step 4: While, in step 4, you 'dissect' the visual design to show how it works, in the current step you actually 'use' the visual tool to get answers with it. Of course, elements that you found to work well in step 4 will be also found as effective tools when answering questions in step 5; and design problems you found in step 4 will also appear as difficulties in answering the questions in step 5.

Example

For the Financial Visualization, let us consider the question "Which stocks have a high market capitalization and high change, and are also traded intensively?" We can answer this by creating a bubble plot showing market capitalization (on the X axis), change (on the Y axis and encoded also in the color), and volume of trading (as bubble size). An example is shown in the figure below:



Now, to answer our question, we look for large bright green or large bright red bubbles placed right in the graph. We see two such bubbles: MSFT (Microsoft) and AAPL (Apple). Both bubbles are relatively large, indicating they are traded with high volume. Both bubbles are placed to the right in the graph, indicating that they have a large market capitalization. However, the bubbles are neither bright red nor bright green, indicating that the price change is actually low. This is actually not very surprising: If we look at the general shape of the scatterplot, we see that most large-market-capitalization stocks are located in a kind of horizontal band around the -1..0% change level. This is relatively expected in stock trading – very large companies do not have massive price increases or decreases of their stocks (they are more stable than small companies). Conversely, if we look at where the brightest red/green colors are, we see they are located to the left of the X axis. Hence, stocks that have a large price change have also a small market capitalization.

Getting the answer to the question is very easy (takes, say, 1..2 minutes) once we have created the above bubble plot. Of course, the difficulty here is to choose the right type of plot in this application. This is (always) a problem in interactive applications which offer many options to their users. For static infographics, this problem does not exist, as these show only one or several predefined visualizations.

A separate issue relates to the interpretation of the plot. For a trained statistician, this is easy. For the general-purpose spectator, this is much harder. One additional issue is that the plot shows overlaps between the bubbles, and these may hide interesting data (visual clutter). To reduce this, we can, for instance, encode the volume of trading in color, not bubble size. Try this visualization yourselves to see which advantages (and also disadvantages) it introduces!

9. Putting It All Together

To conclude the assignment description, here are a few general points about the structuring and writing of the final document:

9.1. Be timely:

Start early, by studying Tufte's books, provided papers, and the additional online resources of the course. The aim here is twofold: Gather more in-depth knowledge and understanding on how information visualization works, and selecting a good application example.

A list of the provided papers, with topics discussed in each one, is given below. All papers are available on the course's web page.

In Favor of Chart Junk (2010)
This paper presents a critical study of the disadvantages, but also advantages, of chart junk. This is in high contrast to Tufte's books, which strongly advocate for design minimalism (thus, against any form of chart junk). The truth seems to be (as always) in the middle: A certain kind and amount of chart junk *may* be actually useful, in infographics, to highlight certain data characteristics which are important for answering the questions at hand. However, the combination of too much chart junk and too complex data is almost surely leading to problems.

Is there Science in Visualization (2006)
This short and easily readable paper presents several viewpoints of known scientists on the actual 'hard science' content in visualization. It is useful to read this when you actually wonder, for your chosen visualization application, if the application formally succeeds in conveying hard evidence for a given problem.

Nested Model of Visualization Design and Validation (2009)
This key reference paper discusses how the process of designing and testing visualizations works. It is the material on which Module 6 is based. While quite technical, understanding the main elements of the iterative design (from application domain to problem, data, questions, tasks, visual encodings, and actually using the visualization) is fundamental to your assignment. Indeed, in this assignment, you basically perform an iteration of the above design-and-evaluation cycle; and the assignment is structured along the same parts (steps).

Rainbow Colormap Still Considered Harmful (2007)
A very easily readable paper about the problems of rainbow colormaps, and how these can be avoided by using other colormaps (also discussed in Module 4). Important to understand if your visualization application uses colormaps.

The Chartjunk Debate (2011)
A technical study of chart junk, with a description of the problem, causes, and possible solutions. Important to understand in detail when you wonder how much of your visualization under study is chart junk, how much is not, and what could be done to fix this.

Note: there are also other papers provided in the online material. Those are less important for this assignment.

9.2. Be complete:

When covering all the steps of your assignment, try to be as complete as possible. For instance, for step 2, try to find all types of different questions that the studied visualization tries to understand; and if the presented problem does generate other questions that the visualization does not cover; for step 4, try to cover all visual design elements present in the visualization and discussed during the course. As there are many such elements, and your time is limited, proceed by prioritizing: Make a list of all design elements you find in the visualization, sort the list by how important you find them for the success of the visualization, and next discuss them in this order and within the time you can allocate to the assignment (stop when you run out of time).

9.3. Be specific:

When discussing any point of your assignment, try to be as specific (exact) as possible. For instance, saying that "some value is encoded using color" is far less specific than saying "the price change of a stock, expressed in percentages, is encoded using a diverging red-black-green colormap, where red stands for losses, black for no change, and green for gains". Or, for instance, saying that "the bubble plot used in the Financial Visualization is sometimes hard to interpret because there is visual clutter" is far less specific than saying "the bubble plot used in the Financial Visualization is sometimes hard to interpret because there is visual clutter generated by overlapping sets of points, which are shown with disks of various sizes, colored by an additional attribute, and drawn using a certain amount of transparency".

When you wonder if your current text has the required level of specificity, ask yourself the question: Does my text contain enough information to explain not only that something is good or bad, but *why* this is actually good or bad, and *why* it happens? If the text answers this question, it is specific. If not, you probably need to add more information to explain the 'why' parts.

9.4. Be detailed:

When explaining any point in the assignment, try to use visuals (images, graphics) to make your point. It is often much easier to explain a certain problem, or solution, by simply drawing a sketch, or using a snapshot with annotations added to it, than writing long discourses in text. After all, this is another practical use of visualization (communication).

To create such annotated images or graphics, you can simply use Microsoft Word, Powerpoint or your favorite text or presentation editor. Next, save the drawings into some interchangeable graphics format and paste them in your final document.

9.5. Be critical:

Students often have the impression that if a visualization comes from some reputed source, it must be a good one and beyond critique; or, conversely, that if a reputed source has critiqued a visual design, that design must be bad and have no value. As shown by the many examples during the course, this is not always the case: We have visualizations created by reputed institutions which are, actually, close to chart junk. And, as the papers on the chart junk show, there can be value in using a certain amount of graphical freedom when designing charts, even if Tufte is against that.

The way to solve this issue, is to go back to the actual problem that visualization tries to solve. For your assignment, carefully re-read the questions you have identified yourself in step 2, which should be answered by the visualization under your study. Then, think critically if and how much the visualization answers these questions. If it does, then you have a good visual design. However, if there are problems here, you can (and should) be explicitly critical about them, no matter what theory or theorists say.

9.6. Work iteratively:

Start writing your essay iteratively: After having completed step 1 (choice of visualization), a good strategy is to quickly go through all points 2..5 described above, just to get an idea of the types of questions you will have to answer. With this in mind, do a first iteration on a few of the first steps (say steps 2..3 or steps 2..4), and produce a preliminary version of the report.

Send the report for feedback to the course lecturer (prof. A. C. Telea) as soon as you have a reasonable first version (say, something around 5..6 pages). Use this feedback to steer the execution of the remaining parts and to improve the already executed parts. Doing around 3..4 iterations in total will ensure that you will be on the right track early on and progress efficiently towards the completion of the entire assignment on time and with good results.

9.7. Final report:

The final report is used to grade your results. Below are listed a few important points on this report:

Size: There is no prescribed size for a good report. Given formatting styles, the use of smaller or larger images, and writing style, the same content can reach very different sizes. However, for this assignment, a typical report should be around 15..20 pages of typed text, 11-12 point size (including images). Larger reports are perfectly fine too. Shorter reports are likely going to have a too low level-of-detail or not cover all the aforementioned assignment points.

Structure: Structure your report precisely along the titles and content of steps 1..5 of the assignment. Within each step, use extra formatting (e.g. bulleted lists, or subsections) to emphasize the aspects discussed in there, similarly to the way this document is structured.

References: All references to existing published or web material should be documented in the text, either via footnotes or references at the end of the document.

Title information: Do not forget to provide a title page with clearly listed information on the course title, course code, student name, and student identification number.

Format: Please submit the intermediate reports and final report in Adobe PDF format. This ensures one can read the document with no formatting issues on any platform.

Language: You should write the assignment in English.