Master’s Programme in Visual Communication Design
Aalto University

BIVARIATE HUE BLENDING

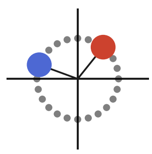
Color scale design for bivariate choropleth maps with a custom tool

Jonatan Hildén

Master’s Thesis 2022

BIVARIATE
HUE
BLENDING

Color scale design for bivariate choropleth maps with a custom tool



Aalto University
School of Arts, Design and Architecture

Aalto University, P.O. BOX 31000, 00076 AALTO
www.aalto.fi

Master of Arts thesis abstract

Author Hans Jonatan Hildén

Title of thesis Bivariate hue blending – Color scale design for bivariate choropleth maps with a custom tool

Department Department of Media

Degree programme Visual Communication Design

Year 2022 **Number of pages** 142+16 **Language** English

Abstract

Bivariate maps are a type of map visualization where two related data series are displayed at once for each data point. They can answer questions of how two variables interrelate in a geographical context using several kinds of encodings —visual variables — such as shape or color. The most common types are choropleth maps that use color hue and lightness to encode data and symbol-based maps that use shape size for both data series. Bivariate maps have seen a minor surge in popularity with new software tools but remain an understudied visualization type with a lack of clear usage recommendations.

The thesis consists of a theoretical and a practical part. The purpose was to collate existing recommendations about the design of bivariate maps and determine whether they are considered a useful type of visualization. The theoretical part was a literature survey of relevant visualization and cartography literature, including empirical studies. I also sought to see whether bivariate choropleths are considered more effective than other types.

The practical part was building a web tool prototype for bivariate color scale creation limited to choropleth maps, the Bivariate hue blender. The tool uses the Hue-Chroma-Lightness (HCL) color space for scheme design. By rotating the hue angle of an input color by a user-defined amount, a new color can be created. Intermediate colors are generated by blending these two with each other and a light secondary input color. The primary purpose of the tool was to improve color scheme creation and the building process used the framework of research-based design. It involved building the tool, using it to evaluate seven existing palettes, and creating three new palettes. These were applied to four different bivariate maps using statistical data from Finland in two different geographical divisions. Test data was selected using contingency table visualizations to ensure that all classes contain values. In addition to the color scales, a bivariate ordinal texture design was created.

Bivariate maps were found to be grouped in categories using the concept of integral and separable dimensions. Bivariate choropleth maps were found to be a relevant visualization type, provided that the data is suitable, and that the number of classes is no larger than 9. An issue pertaining to color contrast was identified — accessibility guidelines stipulate a lightness difference between adjacent hues that require the use of strokes in most choropleth maps. Questions concerning effectiveness of other types, how bivariate symbols interact and how viewers can use bivariate maps for analytical tasks remain unresolved. The tool was subjectively found to enable better control over bivariate color scale creation than other similar software. The evaluated bivariate palettes had issues in lightness uniformity and separation of colors, which could be resolved in the three new palettes. These were found to be at least as practical as the seven initial palettes.

This work has concluded that bivariate maps can be considered useful in special cases with the right data, which should encourage visualization designers to employ them. It has contributed a prototype tool that aids the creation of new perceptually uniform color scales for bivariate choropleth maps. Three new colorblind-safe 3×3 palettes are an addition to the limited set of schemes in active use. The method of selecting data using contingency tables can aid in creating bivariate maps.

Keywords bivariate maps, choropleth maps, color scales, visualization, textures, visual variables

Aalto-yliopisto
Taiteiden ja suunnittelun korkeakoulu

Aalto-yliopisto, PL 11000, 00076 AALTO
www.aalto.fi

Taiteen maisterin opinnäytteen tiivistelmä

Tekijä Hans Jonatan Hildén

Työn nimi Kahden muuttujan sävyjen sekoitus – väriasteikoiden suunnittelu kahden muuttujan koropleettikartoille räätälöidyllä työkalulla

Laitos Median laitos

Koulutusohjelma Visual Communication Design

Vuosi 2022 **Sivumäärä** 142 + 16 **Kieli** Englanti

Tiivistelmä

Kahden muuttujan tietokartat ovat visualisointityyppi, jossa kaksi toisiinsa liittyvää tietosarjaa näytetään kunkin datapisteen kohdalla. Niillä voidaan tutkia kuinka kaksi muuttujaa ovat yhteydessä toisiinsa maantieteellisessä kontekstissa, käyttämällä useita erilaisia visuaalisia muuttujia – kuten muotoa tai väriä. Yleisimpiä tyyppejä ovat koropleettikartat, joissa käytetään värin sävyä ja vaaleutta tietojen esittämiseen, sekä symbolikartat, joissa käytetään muodon kokoa molemmille datasarjoille. Kahden muuttujan karttojen suosio on kasvanut uusien ohjelmistotyökalujen myötä, mutta ne ovat edelleen vähän tutkittu visualisointityyppi, josta puuttuvat selkeät käyttösuositukset.

Opinnäytetyöni koostuu teoreettisesta ja käytännön osasta. Tarkoituksena on ollut koota olemassa olevia suosituksia kahden muuttujan kartoista ja selvittää, pidetäänkö niitä hyödyllisenä visualisointityyppinä. Teoriaosuus on kirjallisuuskatsaus visualisointi- ja kartografiakirjallisuuteen, mukaan luettuna myös empiiriset tutkimukset. Pyrin myös selvittämään, pidetäänkö kahden muuttujan koropleettikarttoja tehokkaampina kuin muita kahden muuttujan karttatyyppejä.

Käytännön osuus on verkkotyökalun prototyyppi, *Bivariate hue blender*, joka on tehty kahden muuttujan väriasteikkojen luomista varten. Työkalu käyttää *Hue-Chroma-Lightness* (HCL; sävy, kromaattisuus, vaaleus) -väriavaruutta. Kun syötetyn värin sävykulmaa kääntää, syntyy uusi väri. Alkuperäisestä ja uudesta väristä luodaan kaksi erillistä väriasteikkoa vaaleasta aloitussävystä ja näitä yhdistämällä muodostetaan asteikon välivärit. Työkalun ensisijaisena tarkoituksena on ollut helpottaa väriasteikkojen luomista. Sen kehittämisessä on sovellettu tutkimukseen perustuvaa suunnittelua. Työkalun avulla on arvioitu seitsemän palettia ja luotu kolme uutta. Näitä on sovellettu neljään erilaiseen kahden muuttujan karttaan, joissa on käytetty tilastotietoja Suomesta kahden eri maantieteellisen jaon mukaan. Väriasteikkojen lisäksi on luotu kuviotekstuuri.

Tutkimuksessa todetaan, että kahden muuttujan kartat voidaan jakaa luokkiin käyttäen kokonaisten ja eroteltavien ulottuvuuksien käsitettä. Koropleettikarttojen todetaan olevan toimiva laji, kunhan aineisto on sopiva ja luokkia enintään yhdeksän. Työssä tunnistettiin värikontrastiin liittyvä ongelma – esteettömyysohjeissa määrätyt vierekkäisten sävyjen vaaleuserot edellyttävät ääriviivojen käyttöä useimmissa kartoissa. Tutkimuksessa auki jäävät kysymykset koskevat muiden tyyppien tehokkuutta, kaksimuuttujaisten symbolien vuorovaikutusta ja sitä, kuinka katsoja lukee ja käyttää näitä karttoja. Työkalun voidaan todeta subjektiivisesti mahdollistavan paremman hallinnan kaksimuuttujaväriasteikkojen luomisessa vastaaviin ohjelmiin verrattuna. Arvioiduissa paleteissa oli ongelmia vaaleuden tasaisuudessa ja värien erottelussa, jotka nyt voitiin ratkaista kolmessa uudessa paletissa. Näiden todetaan olevan ainakin yhtä käytännöllisiä kuin seitsemän alkuperäistä palettia.

Työn loppupäätelmä on, että kaksimuuttujaisia karttoja voidaan pitää hyödyllisinä tietyissä tapauksissa ja niihin soveltuvalla datalla, mikä voi kannustaa visualisointisuunnittelijoita käyttämään niitä. Työn tuloksena on prototyyppityökalu, joka auttaa luomaan uusia tasajakoisia väriskaaloja kahden muuttujan koropleettikarttoja varten. Kolme uutta palettia on lisäys aktiivisessa käytössä olevien kaksimuuttujaisten palettien rajalliseen joukkoon. Kontingenssitaulukoihin perustuva aineiston valintamenetelmä voi auttaa suunnittelijoita kahden muuttujan karttojen luomisessa.

Avainsanat kahden muuttujan kartat, koropleettikartat, väriasteikot, visualisointi, tekstuurit, visuaaliset muuttujat

Contents

Acknowledgements

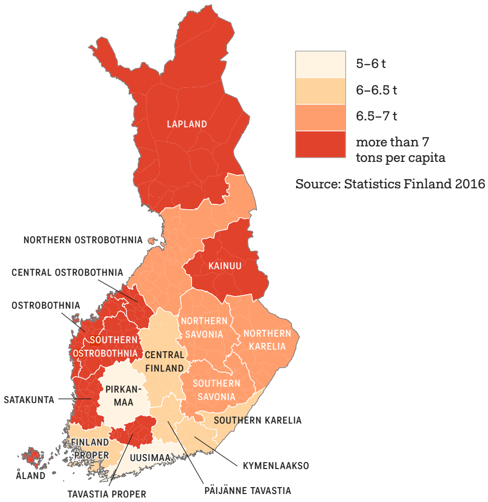
I want to thank my family for putting up with me during this project and especially to Tiina in supporting me in a long process. Thank you to my supervisor and advisor Professor Masood Masoodian for the precise comments. Thank you to Frederik de Bleser and the Experimental Media Research Group in Antwerp for showing the way towards generative design. Finally I also want to thank colleagues and friends for useful pointers and being a source of infographic inspiration.

Helsinki, 18 December 2022
Jonatan Hildén

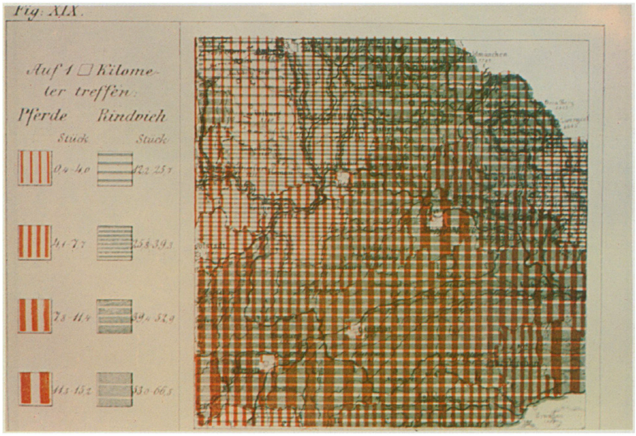
1. Introduction

Choropleth map

Koponen and Hildén: Greenhouse gas emissions in Finland in 2013.Tons CO_2e per capita, not including EU ETS traded emissions or land use, land-use change, and forestry (LULUCF)

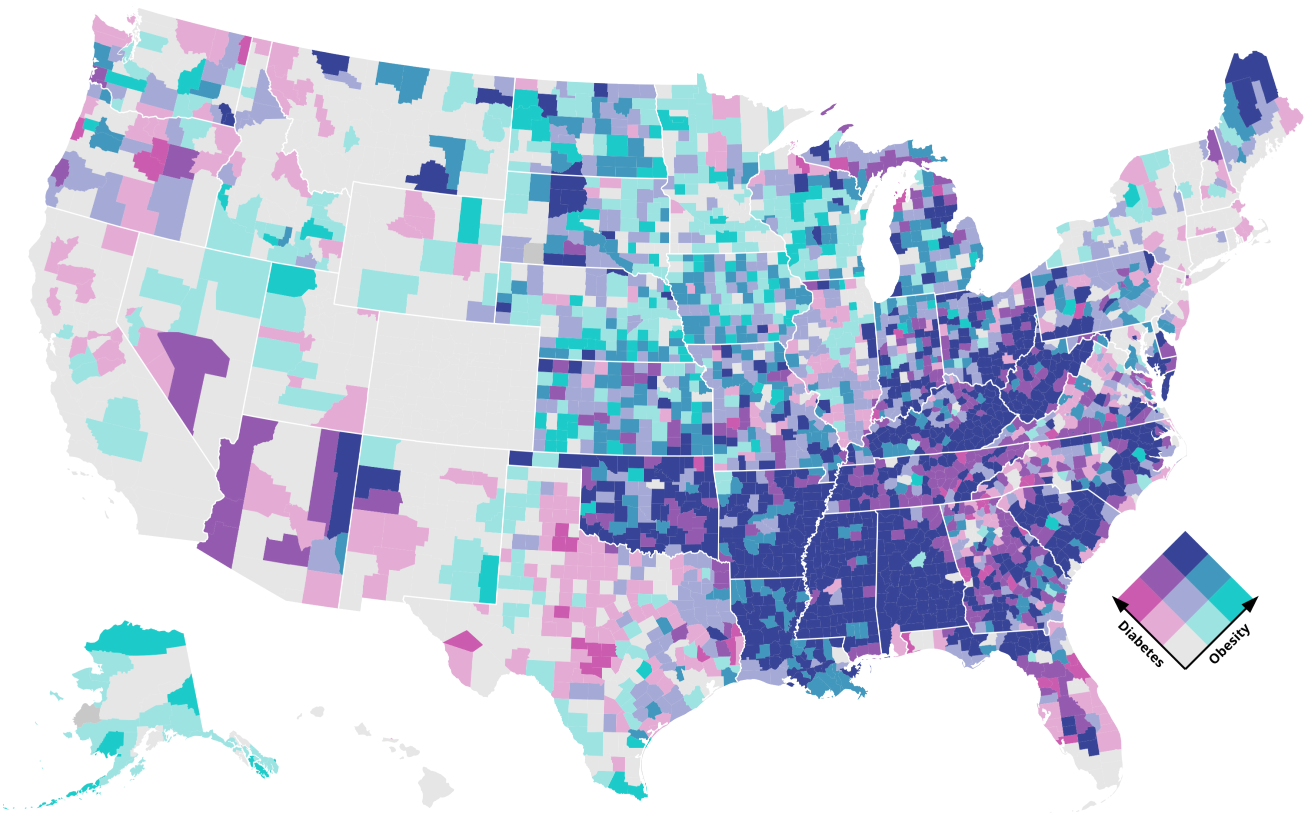


Early bivariate map



Mayr: A Two-Variable Color Map Showing the Joint Distribution of Horses (Pferde) and Cattle (Rindvich) in Eastern Bavaria Done According to Scheme 1

Modern bivariate choropleth map



Bostock: Diabetes and obesity prevalence by county in the United States, 2013.
Colors: Stevens; Data: CDC

Figure 1.1. Top left: Example of a choropleth map adapted from Koponen and Hildén (2019); Top right: Early bivariate map using textures by von Mayr (top right) from Friendly and Denis (2001); Bottom: A modern bivariate choropleth map by Bostock (2019).

> *A heightened awareness of reality as such is something mystics may dream about, but cannot realize. The number of stimuli that impinge on us at every moment — if they were countable — would be astronomical. To see at all, we must isolate and select.*
>
> —E. H. Gombrich (1982)

1.1 Background and motivation

I have been curious about the possibilities of bivariate maps since I learned about them. One important inspiration was Stevens' blog post "Bivariate Choropleth Maps: A How-to Guide" (2015), which step by step describes how to create a bivariate choropleth map. The choropleth map is a common kind of thematic map, where statistical regions (such as municipalities, census blocks or other administrative divisions) are colored based on their data values. The somewhat odd name comes from Greek χῶρος, *khōra*, meaning "place," and πλῆθος, *plēthos*, "multitude"[1]. Choropleth maps for quantitative data typically show a single data series and how its values fall into different classes for the areas on the map (Figure 1.1, top left shows an example of such a choropleth map of Finland). "Bivariate" refers to how instead of one just one data series *two different* data series are shown at once for each mapped area. They are most commonly of the choropleth type — which means that they use colors to encode values (as in Figure 1.1, bottom) — but can employ other symbolizations as well.

I do not remember exactly how and where I encountered it first, but I used Stevens' instructions to create a bivariate grid map of children compared to total population as a part of the *Data Atlas of Finland*, illustrated in Figure 1.2. It was part of a project launched together with Juuso Koponen in 2019[2] and consists of a zoomable map using Finnish 1km² population grid data, which shows how relatively more children tend to live in regions outside the densest urban centers. The map has an interactive legend so that the viewer can choose to hide different population levels. Creating it suggested to me that this type of visualization might be a meaningful way to approach geographical data. In a sense, one could say that the bivariate map translates a scatterplot — a chart which shows the relation between two data series for each data item by encoding them as points in a coordinate system, where one series defines the y position and the other the x position — into a geographical context.

1 "Choropleth definition and meaning" (2022)

2 The project has not been abandoned, but we have not managed to publish any new maps since the launch in May 2019. It is available online as it stands at tietokartasto.fi/Suburban-children/ (Hildén, 2019).

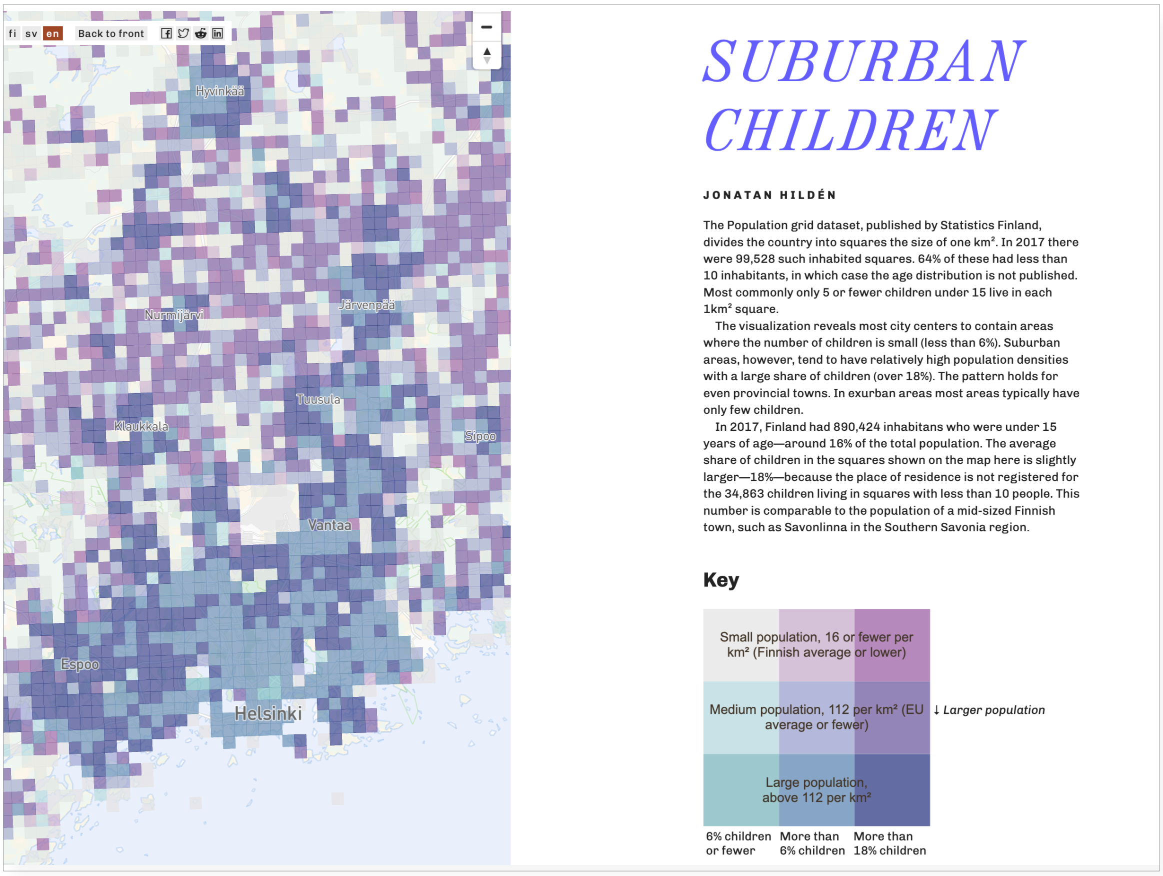


Figure 1.2. Screenshot of the *Suburban children* bivariate visualization (Hildén, 2019).

Looking into the background of this visualization type I found that bivariate maps have a long history with the earliest known example by Georg von Mayr coming out of the climate of innovation that marked 19th century visualizations (Friendly and Denis, 2001, Figure 1.1, top right). They eventually became more widely known and discussed with a series of U.S. Census maps in the 1970s (see Chapter 2., p. 70). These maps were also widely criticized as being hard to understand — Edward Tufte dismissively called them “puzzle graphics” that are experienced “verbally, not visually” (2001, p. 153). But are bivariate maps fundamentally flawed, or can they be designed so that they function effectively? In recent years there seems to have been a renewed surge in interest around bivariate maps, accompanied by instructions and software tools for creating them. Arguably bivariate maps, while visually complex, should make it possible to show relations between two different variables over geographical areas in a way that is hard to achieve with other types of visualizations.

Bivariate maps can show interesting intersections in the data, like how the example in Figure 1.1. highlights that the counties around Mississippi have high prevalence of both obesity and diabetes. Counties in Alaska on the other hand have mostly low rates of diabetes, despite high or medium levels of obesity. For me as a designer having worked with different data visualizations over many years bivariate maps are also in themselves fascinating, because they approach the limits of what can be clearly expressed in visualizations. This is due to the sheer number of data levels that they attempt to translate into visual form.

While it is interesting to discuss extreme visualizations that push the boundaries of what can be interpreted by readers, the discussion about accessibility in visualizations has also become increasingly active and urgent in recent years. This means on one hand stricter formal requirements being set and required and on the other hand a growing sentiment among visualization designers that these issues matter and need addressing. There is a recognition that visualization design — like other fields of design — actively should consider the varying abilities of its audience. Accessibility strictly refers to design for computer software, websites and visualizations that considers people with disabilities, but it can also benefit people with different situational limitations (Henry, Abou-Zahra and Brewer, 2014).

Guidelines for visualization design seldom consider accessibility questions, something which is pronounced in the case of maps. The cartography textbooks that I have surveyed do not deal directly with or even mention design considerations for disabled audiences. Official accessibility guidelines on the other hand offer few useful guidelines for map creation — the Web Content Accessibility Guidelines (WGAG2.1, Accessibility Guidelines Working Group, 2018) define the requirements for accessible websites, but include no detailed and standardized requirements for data visualization at the moment of writing. Color scale design is not the only nor the most important accessibility consideration, but attempting to improve it is a necessary part of more accessible map design.

The WCAG does contain specific requirements for color and could even hastily be interpreted as ruling out the creation of common thematic maps where color encode data, because the adjacent colors in a map with more than three data classes mostly do not have sufficient contrast to each other. Looking at bivariate maps now means discussing at the very least whether the color choices used can be made in a way that satisfies contrast requirements and accounts for color vision deficiencies. The limitations of design tools means that this is not as easy as it could be. Good tools for color evaluation do exist, but they are mostly not an integrated part of visualization design workflows — especially not in desktop software.

In information design the lack of a defined "one size fits all" toolkit or software package means that a designer often is required to create novel software tools or at least modify existing ones to achieve desired results. This approach of tool creation is an underlying principle of generative design as described by Reas and Fry (2007). I was myself first introduced to this way of working through learning the Nodebox software (de Bleser, 2016, p. 43). Having participated in a Nodebox workshop in 2008 I then did an Erasmus exchange with the Experimental Media Group at Sint Lucas Antwerpen in 2009. There I learned more about generative design and Python programming by doing small script experiments and building a constrained drawing application. This established programming and tool-building as an important part of my design practice that continues to inform my work in information design.

I am using this thesis as a way of attempting to explore further the creation of purpose-made small tools to solve design problems using the case of a particularly complicated category of charts, the bivariate map.

The purpose of a bivariate map ought to be that it allows the viewer to see relations between mapped data dimensions more easily than from a pair of separate graphics. To discuss the mapping of data to visual shapes, I use the concept of visual variables first introduced by Bertin (2011). The visual variables are built on the basic visual channels — color, shape, and motion. These are fundamental properties of visual objects that are processed in some extent separately in human visual perception (Ware, 2013, p. 150).

There are multiple visual variables that can be employed in static visualizations to *encode* data — the most important ones ones being location, size, shape, orientation, color (hue, lightness and saturation), and texture density. A good map must always use the most effective visual variable (location in space) to show spatial distribution of a phenomenon like population density — where the studied areas are located. Therefore, only less effective variables like color remain for showing additional data variables that describe the phenomenon in question — the actual population density values. Contrast this with a simple bar chart, where the length and the position of the bars show population values — but not the locations of the areas studied. Introducing the location therefore always involves a tradeoff in visualization.

Visual encodings are also inherently more complex than simply translating numbers into visual form that then can be more or less accurately retrieved by a viewer. There is an ongoing discussion about how situational factors and different task requirements can affect the ordering of visual variables — color saturation and hue can for instance be more effective variables for appraising averages, even if they are much less accurate than position when reading individual values (see Franconeri *et al.*, 2021). Even so there are significant differences between visual variables in how effectively information can be retrieved by the reader and qualitative differences in the visual channels available. These differences would appear to set fairly strict limits to the number of data variables that can be encoded into a visualization in a way that actually can be read and understood.

As visualization researcher Robert Kosara (2022) notes in a recent commentary, the visual encodings that are specified by design are not necessarily the encodings that a reader actually uses to extract information from a graphic. The observed encodings may be more or less emergent properties. Kosara uses the common pie chart as an example — by design a pie chart uses angles to encode information, but research has shown that viewers may use some combination of the resulting visual properties (area, arc length, chord length, and shape) to determine the visualized values.

In a bivariate choropleth map the data dimensions are combined using the visual channels of color. The resulting color scale would seem to function more as a qualitative color scale denoting differences in kind with a secondary quantitative association (darker colors imply larger values).

Thus, the number of colors that can be separated effectively in a map limits the number of variables and steps in the scale. There is empirical support (Lee, Reilly and Butavicius, 2003) for restructuring of data to accommodate human visual cognition — reducing the number of dimensions displayed appears to lead to easier interpretation. Bivariate maps are likely to achieve the goal of being more effective than two separate maps only if they are well designed and show a limited number of data series, and they may well be worse for some tasks. Other kinds of bivariate maps also exist, but they are less common and less studied than the bivariate choropleth.

1.2 Research questions

This thesis combines a literature survey with *research-based design* (Leinonen, 2010, p. 56), where the result is the creation of a set of prototype tools and methods informed by the surfaced theoretical concerns.

I will survey existing research and literature on the effectiveness of bivariate maps and discuss how well the results align with findings on visual perception and how they support the design of bivariate choropleth maps in particular. My working assumption is that previous visualization research, even while not strictly done on bivariate maps, sets clear limits to the number and type of categories that can be effectively read from a bivariate data map. As noted, the usefulness of bivariate choropleths has also been called into question. This leads to two research questions:

1. **Whether bivariate choropleth maps are currently recognized as an effective visualization type**, and
2. **What benefits are bivariate choropleths considered to have over other bivariate maps?**

I also posit that it should be possible to improve the currently common color scales for bivariate choropleth maps with a color scale tool that allows creation and adjustment of bivariate color palettes with immediate visual feedback and charting of hue and lightness values. This should also help in solving accessibility questions. The production part of this thesis consists mainly of the creation of a prototype tool for this purpose. An undercurrent in this work is how tool creation and modification is an essential part in bringing insights from research into functional design and visualization tools. The third research question is therefore:

3. **Can a custom-built interactive tool improve the design of color scales for bivariate maps?**

Using the tool, I attempt to apply existing findings and recommendations to the design of a small collection of bivariate palettes. It may be possible to further enhance the separation of colored areas using additional visual variables such as textures. I survey some literature on texture encodings and experiment with the application of textures to alleviate the accessibility issues of bivariate maps and improve discrimination of individual classes.

The resulting color and texture schemes are finally used to create a set of maps with real-world bivariate map data, which are discussed and assessed using a basic task analysis.

1.3 Thesis overview and structure

The theoretical background in Chapter Two starts out with outlining fundamental features of visualizations and how they apply to bivariate maps based on available theoretical literature: *visual variables*, color and color scale design, textures, and symbol dimensionality. This chapter includes a discussion of how bivariate maps have been defined in a selection of the cartographical literature.

Chapter Three is a description of the practical part of the thesis. It deals with test data selection and used methods for this, the selection and assessment of color scales and the choices that went into building the prototype tool.

Chapter Four documents the details of the created color scale web tool and discussion of its functionality.

Chapter Five starts with an analysis of a set of existing bivariate choropleth palettes and a description of the creation of three novel palettes and a texture palette. The palettes are then applied to the example maps which are then discussed.

In Chapter Six I attempt to answer the research questions, summarize the learning outcomes of this process, discuss the limitations, and give some suggestions for further research.

2. Theoretical background and methodology

In this chapter I describe and discuss theory related to the design of bivariate maps. I start from a short overview of visual variables and some relevant aspects of visual perception. Because the focus of this thesis is on choropleth maps particular attention is paid to the visual variables of color and texture. Relevant specifics of thematic mapping is then discussed. Finally, I describe the theoretical framework of research-based design used to inform the practical part of the thesis.

2.1 Visual channels

Ware (2013, p. 145) observes that research into early-stage visual processing has determined that different visual properties are processed in what is termed to be separate *channels*. The basic channels are color, shape, and motion. The color and form of visual elements are thus treated as separate to some degree in human visual perception. From this follows that it is possible to attend to them in separation — it is for instance possible to look for visual elements of a certain color while disregarding their shapes, or vice versa. Further, there exists a subdivision of the shape channel into orientation, form, contrast, and size. These can be thought of as so-called *spatial frequency channels*. According to Ware they should not be thought of as entirely independent — different channels do interact, but they still represent visual information that is processed separately to some extent. Research has shown that visual distinctness on these fundamental levels is the key explanation for why something stands out — what is termed the *pop-out effect* (Ware, 2013, p. 152).

The concept of *preattentive processing* was devised to account for this effect (Ware, 2013, p. 152), but a more current view is that all stages of visual processing is guided by attention, which makes the *pre* in the term preattentive misleading, as perception and visualization researcher Steve Haroz notes on the *Data Stories* podcast ("Visual Perception and Visualization with Steve Haroz," 2018).

Things that stand out can nevertheless be focused on using selective attention: we can ignore a particular class of objects and find only those with specific visual characteristics, provided these characteristics are sufficiently different. Ware (2008, p. 29) proposes to call these *tunable* rather than pop-out features, to reflect that they can be *tuned* for selectively when planning eye movements. Figure 2.1. shows a selection of known tunable features with practical relevance for design.

Ware (2013, p. 384) notes that the selectivity of attention isn't perfect: there will be some crossover from confounding elements that also get processed and strong stimuli (like blinking or movement) might break the focus of attention. Despite this, what we see is mostly what our attention is focused on.

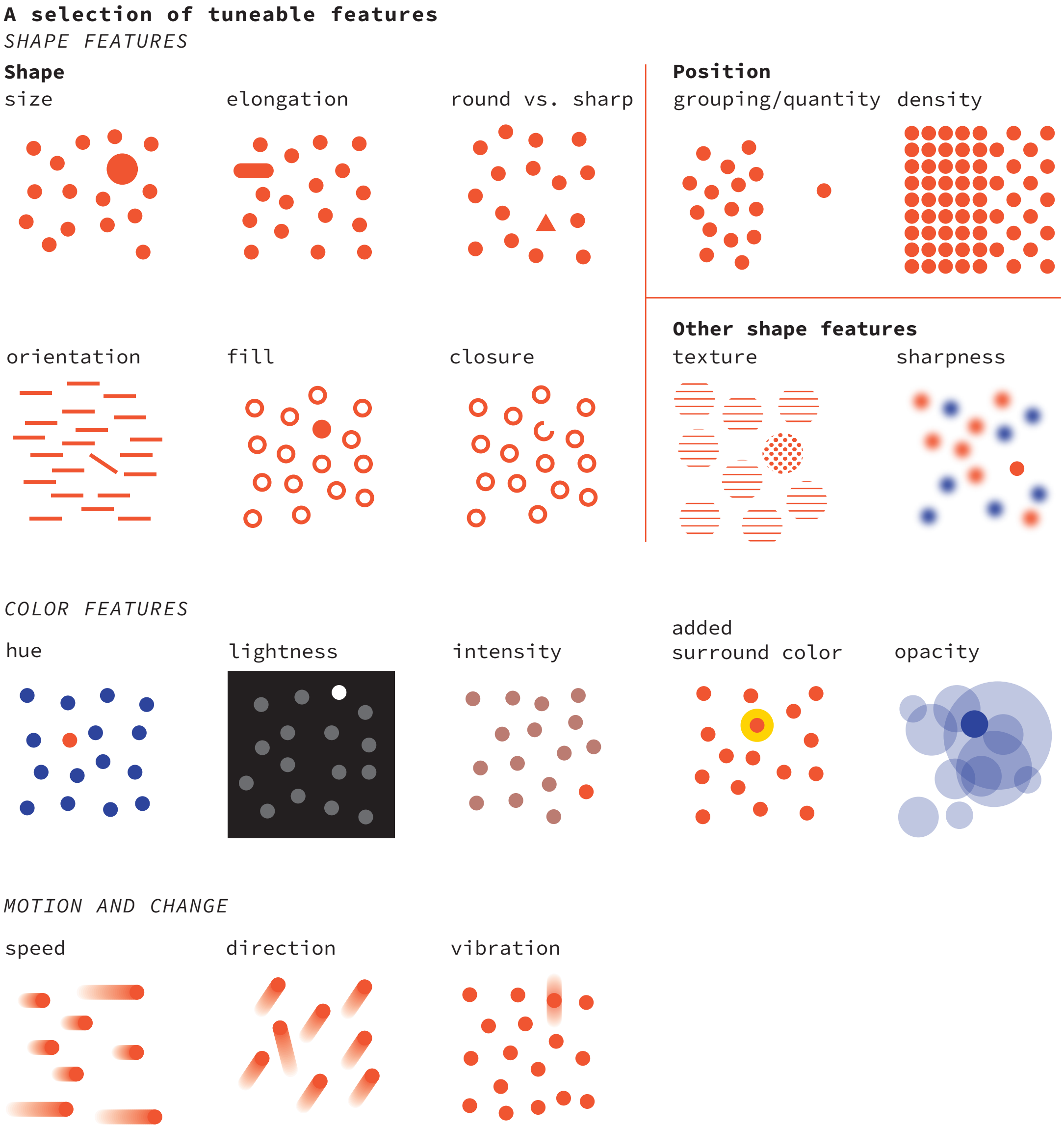


Figure 2.1. Examples of tuneable features after Koponen and Hildén (2019) using the terminology of Ware (2008).

From experiments it has been determined that clear differences in distinctness on the level of basic visual channels mean that tasks like the following can be performed effectively and quickly (Koponen and Hildén, 2019, p. 52):

- finding an individual element
- identifying edges between groups or separating groups
- tracking elements that move
- assessing numbers of elements shown

Whether something stands out or not has been determined to depend on two factors. The first is how different the feature a test person is looking for (i.e., the target) is from the surrounding non-targets. The second is how different the non-target items are from each other. Figure 2.2. (p. 22) shows an example of this. The lone red circle is easy to find when surrounded by blue non-targets (left), but much harder to spot if the non-targets come in many colors (right).

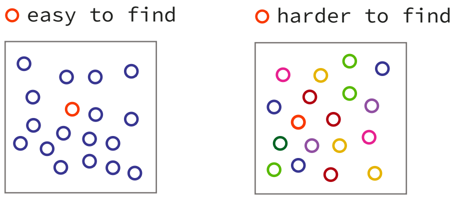


Figure 2.2. The appearance of surrounding non-targets affects how easy it is to find a target feature. After Koponen and Hildén (2019).

Visual elements that are distinct from each other and from the background on multiple channels will be easier to find than ones that are more similar, and something that is alone in using a particular channel will be very easy to find — like a colored item in an otherwise grayscale graphic. (Ware, 2013, p. 157)

Using *redundant coding* can make objects more distinct still, for instance by applying clear differences in both shape and color (see Fig. 2.3). Adding something to a feature typically makes it more distinct than removing something from it. It is easier to spot a single area on a map that has an added additional symbol than finding an individual area which lacks a symbol, as shown in Figure 2.4. (Ware, 2013, pp. 148–149, 157)

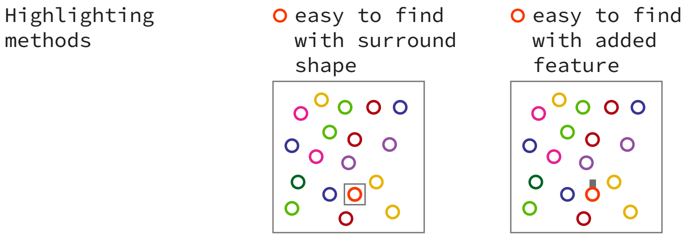


Figure 2.3. Example of added features making a target object more distinct (after Ware, 2013, p. 158).

2.1.1 Conjunction searches: combinations that can be found

More complex visual patterns result in visual tasks where rapid searches are impossible. While searches for clear differences based on one or several channels that reinforce each other are very rapid, searches based on combinations of visual properties become much harder. If the targets are coded visually into two or more intersecting groups searching for a specific target becomes a *conjunction search* — the viewer must find a specific combination of, e.g., color and shape attributes (Treisman and Gelade, 1980).

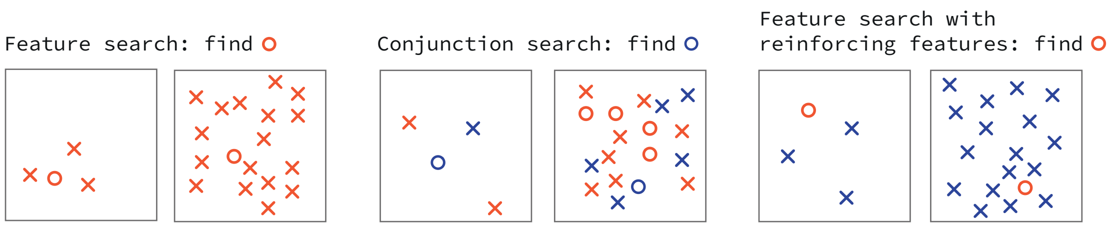


Figure 2.4. Example of feature search (left) compared to conjunction search (center) and redundant coding (right) after Treisman and Gelade (1980).

In the illustrated example in Figure 2.4 there are both blue and red crosses and circles. Looking up only red objects or *either* circles or crosses are single feature searches that do not become significantly slower even when the number of objects in the visual image is increased. In comparison, finding just the object that is blue *and* circle, if there are both red and blue circles as in the middle panel takes longer time and is strongly impacted by the total number of objects. Referring to Treisman's work, Ware (2013, pp. 159–160) notes that this is a fundamental limitation — even extensive training will only modestly improve the speed of conjunction searches. Quickly identifying complex patterns in visualizations is therefore hard and relying on conjunction encodings ought to be avoided.

According to Ware (2013, pp. 160–170) there are some important exceptions to the limitation of conjunction searches relating to the combination of spatial codings such as position, stereoscopic depth, shape from shading or motion with another attribute like color, size, or shape. In such cases rapid conjunction searches appear to be possible. Of practical interest is spatial grouping — if items are grouped visually into clusters on the XY plane, it is possible to perform rapid searches for the conjunction of color and a particular cluster, by searching within each cluster separately. Movement also appears to allow segmentation in a similar way. Moving and non-moving targets can be searched rapidly in separation — a red moving target is visually distinct from a static red non-target. Movement could therefore be useful as a separating dimension in on-screen geospatial displays. The application of spatial grouping in map design is more limited, as the XY dimension is required to show the locations of the mapped regions.

2.1.2 Visual variables: definition and use in maps

Visual channels can be further grouped into visual variables which are used to encode information into visual shapes. Visual variables as the building blocks of visualization were systematically defined and described by Jaques Bertin using the term *retinal variables* in *Semiology of graphics*, published in French in 1967 (2011). This work will use the current established term *visual variables* used by MacEachren (1995) and Tyner (2010) among others. In the original cartographical use, they refer to the visual means by which the point, line, and area symbols employed in maps can be further differentiated based on the values or characteristics that they represent (Tyner, 2010, p. 136).

There is a limited number of useful visual variables for encodings that can be rapidly understood. Table 1 is quoted from Ware (2013, p. 171) and presents such low-level graphical attributes that can be used to create glyphs that are practical to separate visually. This can be compared with the list of cartographical visual variables illustrated in Figure 2.5.

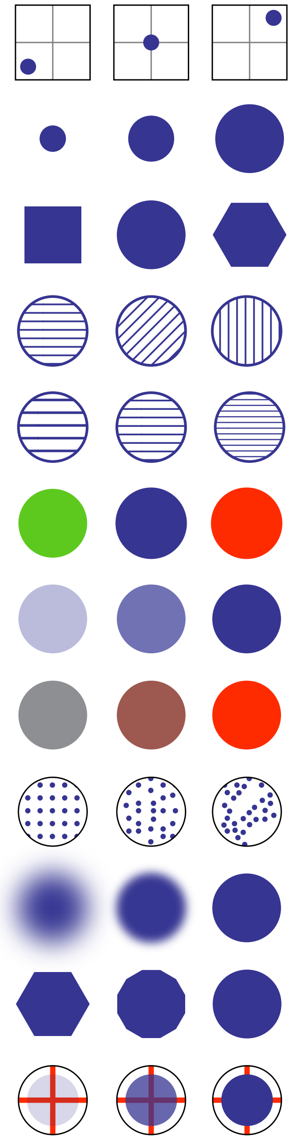
Visual variable	Dimensionality	Comment
Spatial position	Three dimensions: X, Y, Z	
Color	Three dimensions: defined by color opponent theory (luminance, red-green, yellow-blue)	Luminance contrast is needed to specify all other graphical attributes
Shape	Size and orientation are basic but there may be more usable dimensions	The dimensions of shape that can be processed rapidly are unknown; however, the number is certainly small
Surface texture	Three dimensions: orientation, size and contrast	Surface texture is not independent of shape or orientation; uses at least one color dimension
Motion coding	Approximately two to three dimensions; more research is needed, but phase is critical	
Blink coding	One dimension	Motion and blink coding are highly interdependent

Table 1. Separable graphical attributes of glyphs. After Ware (2013, p. 171).

Maps and map-like visualizations are characterized by the fact that the visual variables generally considered to be most effective — the vertical and horizontal position in space — are required to display absolute (or in the case of, e.g., cartograms relative) geographical locations (Koponen and Hildén, 2019, p. 59). Thus, the designer of a data map is forced to employ secondary and less effective variables for the data dimension(s). The most commonly used are the color variables: density (lightness/value), hue and saturation; area (in the case of linear features width), shape and texture (pattern) (Tyner, 2010, pp. 136–137).

In theory any number and combinations of the remaining visual variables could be used to *encode* multiple data dimensions in the graphical marks on a map. In practice research into visual variables shows that certain combinations are more effective than others. A foundational work on this is the study by Cleveland and McGill (1984) that determined that the most accurate encoding for numerical values is spatial position, with length coming second. Color saturation is particularly ill suited for representing exact values. These should not be taken as being consistent across all tasks and contexts: Szafir *et al.* (2016) posit that, e.g., color encodings may be more effective for appraising average values in a graphic by virtue of being less accurate in isolation.

MacEachren (1995) proposed the classification of visual variables by suitability for different encoding tasks: what in semiotics is termed as their *syntactics*. These are presented in Figure 2.5.



Suitability for encoding data by type:

Visual variable variations	**Variable name**	Nominal (non-ordered)	Ordinal (ordered)	Numerical (quantitative)
	Location	GOOD	GOOD	GOOD
	Size	GOOD	GOOD	GOOD
	Shape	GOOD	POOR	POOR
	Orientation	GOOD	MARGINAL	MARGINAL
	Texture (density)	GOOD	MARGINAL	MARGINAL
	Color hue	GOOD	MARGINAL	MARGINAL
	Color lightness (value)	POOR	GOOD	MARGINAL
	Color saturation (chroma)	GOOD	MARGINAL	MARGINAL
	Arrangement	MARGINAL	POOR	POOR
	Crispness	POOR	GOOD	POOR
	Resolution	POOR	GOOD	POOR
	Transparency	MARGINAL	GOOD	POOR

Figure 2.5. Visual variables in maps and their syntactics redrawn from a graphic by Roth (2017). Bertin's associative and selective categories have been left out.

Nominal variables without a natural ordering (non-ordered) are in MacEachren's classification useful for qualitative distinctions — identifying different categories. Shapes are associated with nominal differences but can not be sorted in an unambiguous way. Ordinal variables again, such as color lightness or transparency, are easy to see as sorted while not necessarily being suited for quantitative encoding. Of these only position and size have a direct association with quantitative values. Here size is understood to include both length and area but reading exact numerical values from areas is fairly inaccurate (Koponen and Hildén, 2019, p. 60).

While the number of data items that can be encoded in a visualization is limited only by the designer's imagination, there would appear to be quite strict limits to what information a reader actually can retrieve from a visualization accurately and in a timely fashion due to limits on visual processing (Haroz and Whitney, 2012). The difficulty of interpreting any graphic increases with its complexity, although not in a linear way. MacEachren (1982a, 1982b) employs the term *visual complexity* to refer to the level of structural and organizational intricacy in maps.

A *glyph* is a general term for a graphical object where one or several visual variables such as width, area or color represent data values (Ward, 2008). In visualization research the term is especially used for symbols that attempt to encode multiple quantitative variables (Koponen and Hildén, 2019, p. 32).

2.1.3 Color perception, color spaces and color scale design

Bivariate choropleth maps use color variables for all encodings, which motivates a detailed discussion of color in the context of this thesis. Of the commonly used visual variables, the ones relating to color are arguably among the most complex. Because most humans are trichromats, meaning that we typically have three basic color receptors, any color can in principle be defined as the combination of three primary colors. All color spaces can hence be understood as three-dimensional volumes. (Ware, 2013, pp. 97–98). This is termed the opponent process theory and the three perceptual color channels are *red-green*, *yellow-blue*, and *black-white* or luminance (Ware, 2008, p. 68). This is illustrated as a simplified schematic in Figure 2.6. Short wavelength S cones also contribute to the luminance channel, but only to a small extent and in particular conditions (Ripamonti *et al.*, 2009). The rods (not illustrated) have a limited role in color vision as they are mostly inactive in brightly lit conditions but can contribute to the luminance channel.

Color opponent channels

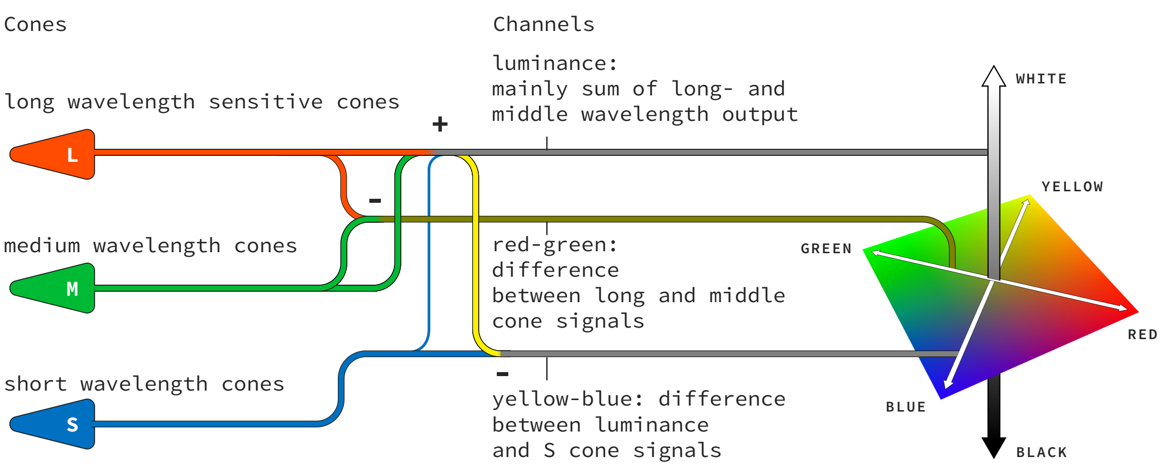


Figure 2.6. Schematic of the interaction of color opponent channels after Ware (2008).

Brewer (1994, pp. 124–126) notes that there is some variation with regards to color terminology in the cartographical literature. According to Brewer the least confusing and most generally accessible terms for the perceptual dimensions of color are *hue*, *lightness*, and *saturation*. These three dimensions the make up a familiar three-dimensional color space and are understandable to lay persons.

- **Hue** corresponds to the named color, such as green, yellow, or red and is formed from the red-green and yellow-blue perceptual color channels.
- **Lightness** refers to how light or dark a surface color with a particular *luminance* appears[3]. Referring to the perception of self-luminous screen colors *brightness* would be more accurate (Ware, 2013, p. 80), but following Brewer's example I will generally prefer lightness in this thesis. The term *value* is often used to refer to lightness in cartographical literature, for instance by MacEachren (1995).
- **Saturation** or chroma generally refers to how colorful or vivid a color appears. White, grays and blacks are colors with very low or zero saturation[4].

The established standard for accurate measurement of colors uses the *Commission International de l'Eclairage* (CIE) system which defines three abstract receptors or *virtual primaries* based on color perception experiments to determine how humans respond to color and termed X, Y and Z (Smith and Guild, 1931). Together they form an abstract 3-dimensional space, where all perceivable colors can be defined by XYZ coordinate values (The Y value is also the same as luminance). Light-based colors are straightforward to specify in this way with three colored lights — computers use additive red, green and blue primaries (RGB). Surface color measurement and specification is far more complex, since it is dependent on lighting conditions and pigment interactions. (Ware, 2013, pp. 101–103)

Ware (2013, pp. 102–103) notes that since three-dimensional CIE XYZ coordinates are challenging to use practically, they are usually converted to a different representation that separates the lightness component, giving in two cartesian *chromaticity coordinates* (x and y), and luminance (Y) — see Figure 2.7 for an illustration.

Because of technical limitations, not all colors that can be seen can be created on a particular physical display device. This limitation is termed the *device* or *monitor gamut* (Ware, 2013, p. 102). A color that can be displayed on

3 Brightness, lightness and luminance are often used interchangeably, but technically luminance refers to the absolute, measured amount of visible light emitted or reflected (weighted by the effect of each wavelength on the human visual system), brightness to perceived luminance, and lightness to the perceived reflectance (amount of light as reflected by a surface) relative to the brightness of a similarly illuminated white surface. Since luminance is perceived in a non-linear way, perceptually uniform color spaces try to adjust brightness values accordingly. (Ware, 2013, pp. 80, p.89)

4 If the saturation or chroma is zero, hue is not defined at all in a cylindrical color coordinate space, since the angle is undefined for a ray without length (Verou, 2020).

a display device is said to be within the gamut. For display on a computer screen, colors have to be converted to a device color space such as *sRGB* (Ware, 2013, pp. 104–105).

The gamut of a RGB color space is defined by the red, green, and blue primaries and the *white point*. Basically, this means the appearance of the brightest white in the given color space, and it is set to correspond to particular lightning conditions. The most widely used white point is D65, which is designed to correspond to daylight from a cloudy sky. (Ware, 2013, p. 104; Atkins, Lilley and Verou, 2022)

Notably, device gamut is often in practice much larger than the gamut of the widely used sRGB color space. Many computer monitors can display a wider range of colors than sRGB can represent — the P3 color space, which corresponds to many high-gamut displays has 50% more colors (Verou, 2020).

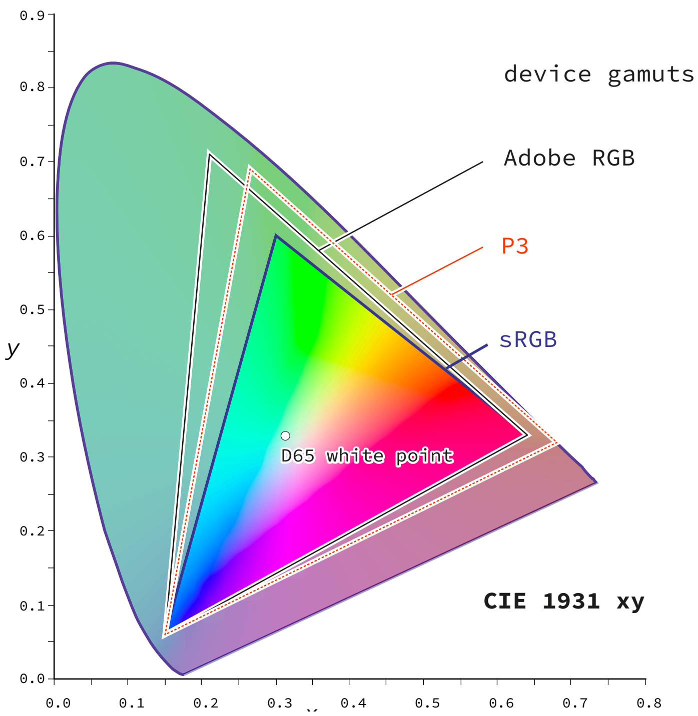


Figure 2.7. Gamut comparisons on a diagram of the CIE 1931 xy color space redrawn based on Myndex (2022).

Figure 2.7 shows the CIE 1931 chromaticity diagram overlaid with the gamuts of the P3, Adobe RGB and sRGB color spaces (Colors outside sRGB are dimmed as they cannot be reliably represented). The corners of the gamut triangles represent the primary colors of each color space. As the diagram shows, the biggest differences are in the available range of green, yellow, and turquoise hues.

To be practically useful for creating color scales a color space should be *perceptually uniform* (Ware, 2013, p. 105). This means that in theory, an equal distance anywhere in the color space (i.e., equal changes in values of color coordinates) represents an equal difference in perceived color. The color spaces *CIELAB* or *L*a*b** and *CIELUV* or *L*u*v* are transformations of the

CIE XYZ space that are intended to be perceptually uniform (Ware, 2013, pp. 105–106). They retain the property of being device-independent, which means that they correspond to a theoretical model of a “standard observer” rather than colors displayed on a particular device, and must therefore be converted for displaying (Ware, 2013, p. 101).

In both of these standards, L* stands for perceptual lightness, while the chroma and hue are mapped to the other axes. The *u* and *v*, or *a* and *b*, coordinates both signify *chromacity coordinates* — positions on green–red and blue–yellow axes specified somewhat differently. L*u*v has a more even distribution of colors, but can have more clipping issues (Somers, 2020).

It should be noted that while these color spaces are fairly uniform in lightness, neither are fully uniform in hue and chroma (Somers, 2020). A particularly noticeable problem in CIELAB is that a range of blues with equal hue but decreasing chroma will shift into purple (Atkins, Lilley and Verou, 2022, ch. 9.1). CIELAB was developed for reflective colors and is primarily used in the printing industry, while CIELUV is better suited for describing and creating self-luminant screen colors. According to Somers (2020) L*u*v is more uniform in color distribution and better suited for choosing colors, but L*a*b* is more widely known and directly available in applications like Adobe Photoshop™. OKlab (Ottoson, 2022) is a recent alternative perceptually uniform colorspace which improves on CIELAB by being more uniform and more consistent in hue and chroma[5].

According to Ihaka (2003) human color *understanding* seems to follow a polar coordinate representation, even if the cartesian opponent channel model works well to describe how color vision works. This means that instead of defining hue and colorfulness on a plane with x and y coordinates, a more intuitive representation uses some kind of color wheel, where the different hues are placed in a circular arrangement. This can be achieved by a cylindrical transformation of a cartesian color space. Either CIELAB or CIELUV can be used in a cylindrically transformed version termed *CIE LCh*[6] or the hue-chroma-lightness (HCL) space, a later development created specifically for information visualization (Zeileis *et al.*, 2020). In HCL hue is given as an angle and colorfulness as chroma, defined by the radial distance on the given hue angle from a neutral color of the same lightness in the center of the coordinate system. They retain the differences of the original cartesian versions — cylindrical L*u*v thus has more even hue differences, consistent chroma and opponent colors are found at 180° opposite angles (Somers, 2020). I will use the term HCL in this thesis following Zeileis, instead of the alternative LCH.

5 The *CSS Color Module Level 4* document by Atkins, Lilley and Verou (2022) that describes how color is handled in Cascading Style Sheets (CSS) (Atkins, Etemad and Rivoal, 2022) gives a good summary of different color spaces and color representations, and how colors can be specified for web documents. Latest version available at https://www.w3.org/TR/css-color-4/

6 Sometimes also HSLuv specifically for cylindrical CIELUV; OKlab is called OKlch in its cylindrically transformed version

A stepped three-dimensional schematic of the HCL (LCH) color space

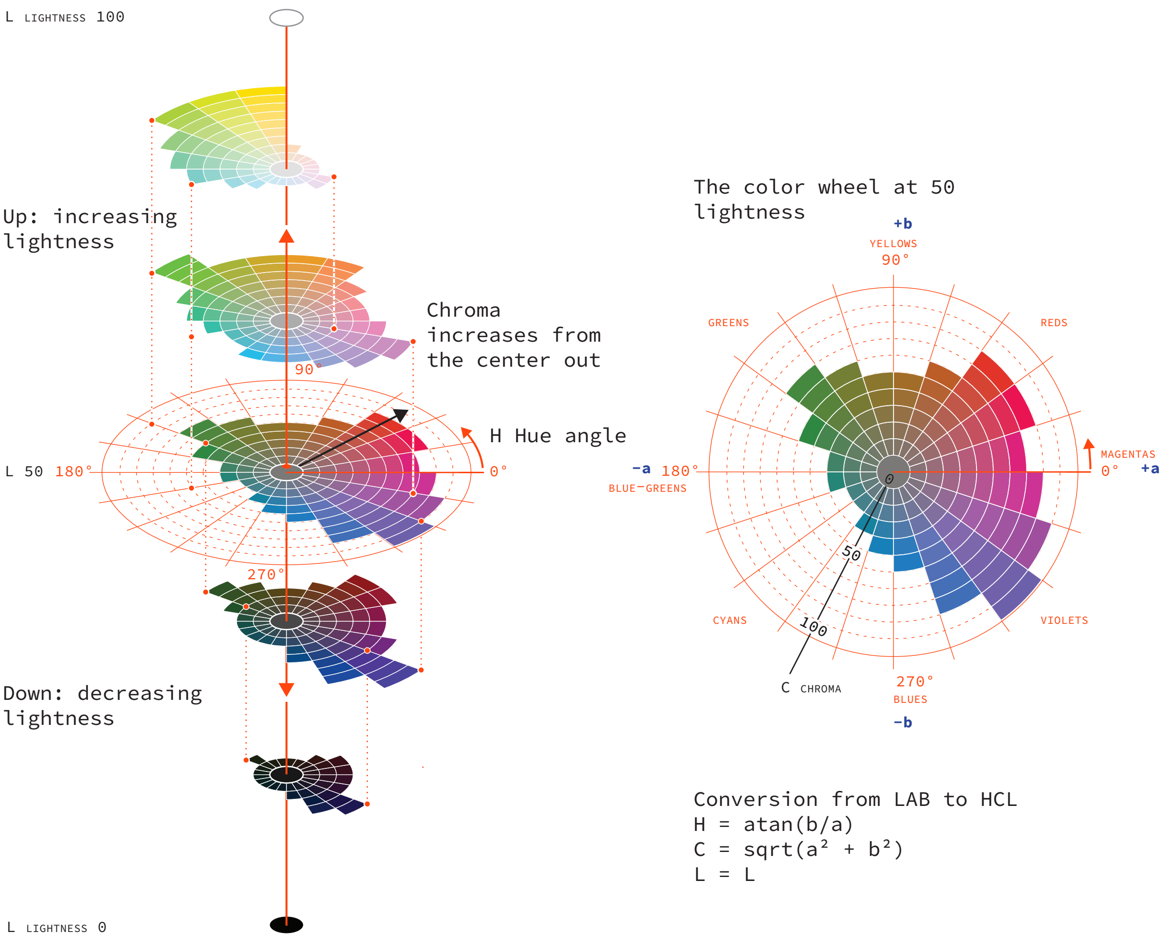


Figure 2.8. Stepped three-dimensional representation of the cylindrical hue-chroma-luminance (HCL) color space.

Figure 2.8 shows a stepped view of the HCL color space based on CIELAB and the color wheel at 50 lightness with the formulas for converting L*a*b values to HCL. The hue is the rotation angle (0 to 360°), the color intensity (chroma) increases from the center outwards, and colors are lighter towards the top. L for lightness in HCL is the relative luminance, which is precisely the same as lightness in CIELAB on a 0 to 100 range where 0 is black and 100 is white[7]. Undefined or out-of gamut colors are not shown.

Since the hue-chroma-lightness (HCL) color space is fairly established, in this thesis I will also use chroma instead of saturation when referring to exact values in the color space.

Not all HCL values are perceptually defined, since the possible range of chroma varies depending on hue and lightness — hence the asymmetrical color wheel in Figure 2.8. From a HCL model it can be observed that colors such as “dark yellow” or “bright violet” do not exist. Saturated yellows always appear light, while saturated blues or violets are much darker (Koponen and Hildén, 2019, pp. 66–67). Note that this is different from colors being out of gamut, although the practical result is the same.

7 The numbers for lightness are usually given as unitless, but the CSS specification uses percentages for compatibility reasons (Atkins, Etemad and Rivoal, 2022, ch. 9.1)

The common Hue-Saturation-Value (HSV) and Hue-Saturation-Lightness (HSL) color spaces are also cylindrical, but as they are simply alternative representations of the RGB color space they are not even somewhat perceptually uniform. For this reason, they can not be used to design equidistant color scales. (Brewer, 1999; Koponen and Hildén, 2019, pp. 66–67)

This can be demonstrated with visual comparisons. In a perceptually uniform color space hue values can be changed while lightness and chroma remain constant, while a non-uniform color space will show changes in lightness when hue changes.

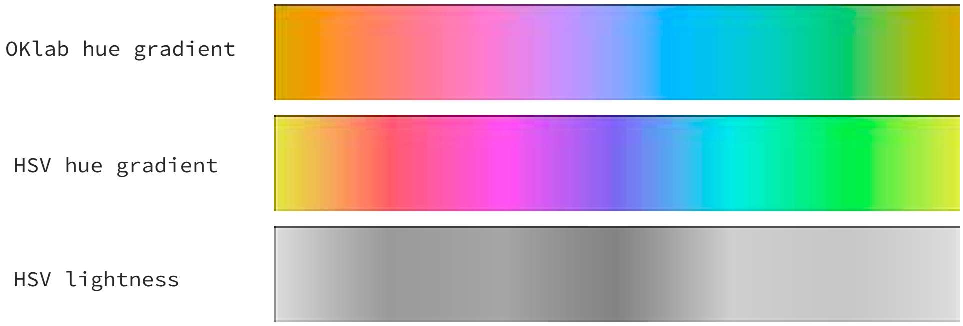


Figure 2.9. Comparison of a color gradient with constant lightness and saturation in OKlab with one created using HSV with changing hue and constant saturation and value. Image adapted from Ottoson (2022).

As Figure 2.9 shows, the Hue-Saturation-Value (HSV) representation of the RGB color space is not uniform, because changes in hue causes clearly visible shifts in both lightness and chroma that appear as bands and more saturated regions. The uneven lightness variation in the HSV gradient is apparent when plotted separately.

Lightness is essentially meaningless as a separate value in a non-uniform color space like HSL. In Figure 2.10 a saturated blue (left) has the same HSL lightness value as a saturated yellow, but they are not even close in apparent lightness. To the right the same blue is contrasted with a "yellow" of the same hue, but the actual lightness is the same for both — resulting a brownish color[8]. Because dark saturated yellows do not exist, its saturation value is much lower in the uniform HCL color space.

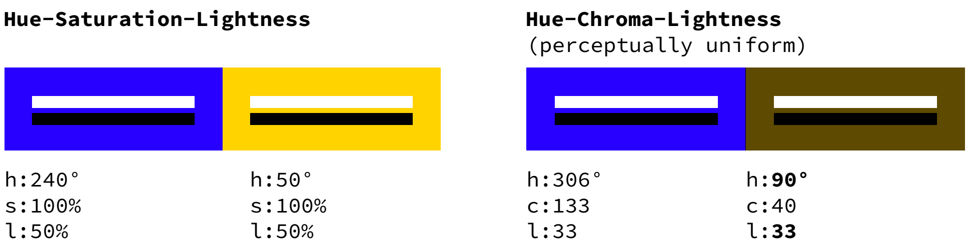


Figure 2.10. Comparison of HSV colors and HCL colors with similar values.

8 The hue angles are different, because HSL angles do not correspond exactly to HCL (Verou, 2020). Values were calculated with Chroma.js.

The issue with undefined and out-of-gamut colors complicates the practical use of perceptually uniform color spaces like HCL as such colors easily can be created when inputting values (Zeileis *et al.*, 2020). This is not something that can be solved simply by using wide gamut color spaces. As noted previously, many computer monitors can indeed display a wider range of colors than what the common sRGB color space can represent. However, this does not mean that wide gamut color spaces like P3 necessarily give any real advantage when designing visualizations. Such color spaces are not yet consistently supported for display outside specialist imaging applications, and especially not on the web (Lilley and Needham, 2022). Common projectors used for presentations also typically have smaller gamuts than screens, and for printed graphics the available range of colors tend to be much smaller still. An additional complication is that the colors available in wider gamuts are unequally distributed across the range of possible hues, as shown in Figure 2.7. While the increase in the number of possible colors in high-gamut spaces sounds impressive relative to sRGB, the practical differences are perhaps not that large in most contexts[9].

Designing color scales that use the full range of colors in wide gamut color spaces has limited utility at least until these are reliably supported across web browsers and even then the advantage may be small. For these reasons sRGB remains the standard color space for screen use. While out-of-gamut colors can be mapped to the available gamut in a particular display color space, this process is fraught and unreliable and frequently introduces changes that can affect perceptual uniformity (Ware, 2013, p. 138). Therefore, a safer and more reliable approach may for now be to work within the limitations of sRGB and avoid colors that it cannot represent when designing colors for visualizations.

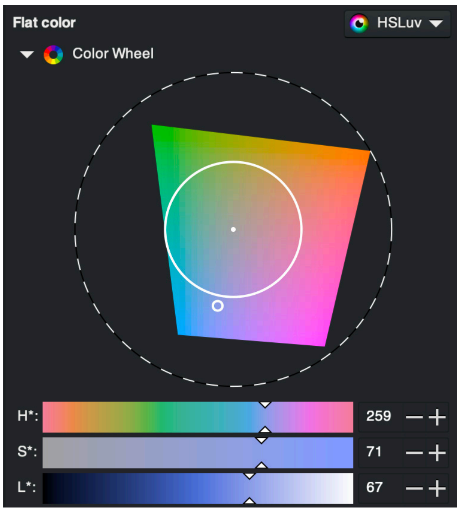


Figure 2.11. The Inkscape HSLuv color wheel (2022). Screenshot.

9 This can be tested in, e.g., Adobe Photoshop™ by comparing a green color input with the RGB values `0, 255, 0` with one with the maximum L*a*b values `100, -128, 127` in a document using L*a*b color mode and trying different display color profiles. I was unable to perceive the difference between these hues on a 2016 MacBook Pro using the Color LCD (default), Adobe RGB, or Display P3 color profiles, but with Wide Gamut RGB it became clearly visible.

Figure 2.11 shows an example of the HSLuv (cylindrical transform of CIELUV) color picker in the open-source vector drawing program Inkscape, demonstrating one possible solution to deal with undefined colors in an interactive tool. The quadrilateral shape represents the available color gamut for the current lightness value and input values are clipped to fall within it.

Undisplayable colors have to be considered when designing perceptually uniform color scales as well, where each step ought to represent a monotonic uniform change in perceived color value. Without adjustments intermediate hues are often undefined, even if the end and start points are valid colors. Figure 2.12 shows an example of this in the *HCL color picker*[10] (Brown, 2022). The uniform linear interpolation between a light green and a dark blue with equal chroma shows a gap of undefined colors.

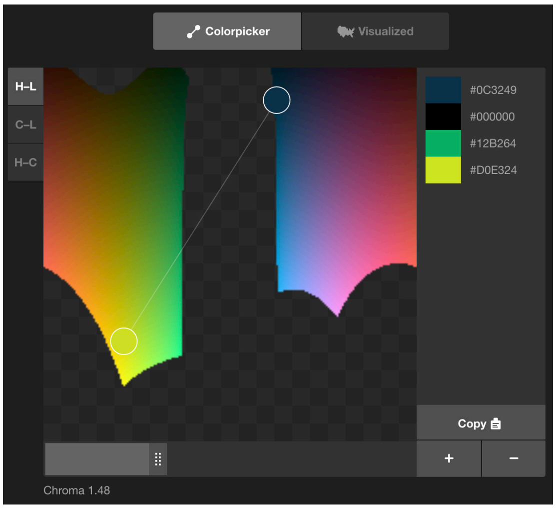


Figure 2.12. Undefined colors as visualized in the HCL color picker (Brown, 2022). Screenshot.

The HCL color space is currently not available in most desktop design software, but it exists as software implementations such as Gregor Aisch's *Chroma.js* (2022a) JavaScript library, the *Culori* JavaScript library (Burzo, 2022) or the `colorspace` R package by Zeileis *et al.* (2020)[11]. These allow interactive creation of colors using the HCL color space which are then converted to displayable sRGB but share the fundamental usability challenge of undefined and out-of gamut colors.

According to the v2.4.0 documentation Chroma.js uses the CIELAB space for HCL colors (Aisch, 2022a). Because of its easy availability, I use the definition in Chroma.js in this thesis, despite some disadvantages compared to CIELUV-based HCL (see p. 29). Ardov's (2022) *Huetone* is one example of a color picker displaying HCL or LCH, implemented with chroma.js[12]. Figure 2.13 shows an example of how a selected color varies in lightness, chroma and hue as displayed in the Huetone app. The grey areas in the charts represent undisplayable or non-existent colors.

10 Available at http://tristen.ca/hcl-picker/
11 The colorspace package can be used online at http://hclwizard.org/
12 https://huetone.ardov.me

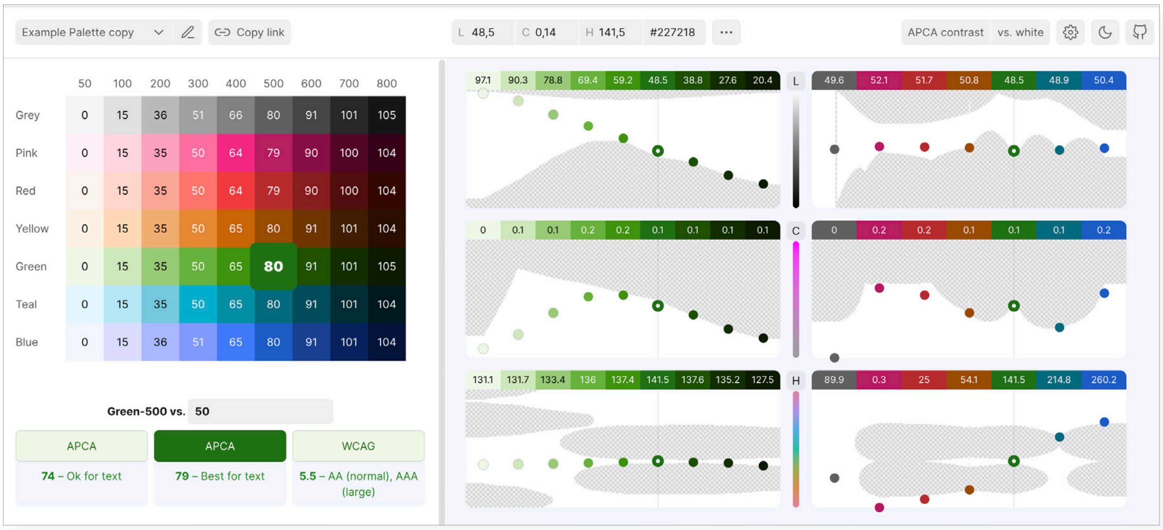


Figure 2.13. The Huetone web app displaying multiple palette colors (Ardov, 2022). Screenshot.

Ware (2013, pp. 106–107) notes that while perceptually uniform color spaces are helpful for color scale design, color interactions can lead to significant differences in the perception of adjacent colors that they cannot account for. The size of a colored area is an important factor: colors are perceived more accurately for large samples, while color differences very nearly can vanish for tiny patches, especially on the yellow–blue axis. From this follows that color scales need to be designed with larger difference between colors the smaller the colored areas in a graphic are (Szafir, 2018).

Lightness differences are critically important for making out shapes and detecting small details: a grayscale sequence allow a much better perception of form than a sequence that varies only in hue, saturation or both (Ware, 2013, p. 129). Monochrome grayscale sequences are however very susceptible to contrast effects, where the appearance of a colored area is affected by surrounding colors — simultaneous contrast. From this follows a recommendation to design quantitative color scales so that they combine a continuous change in hue with a change in lightness, creating what Ware describes as a “spiral upward in color space” (Ware, 2013, p. 131). Devising such color scales that are both effective and aesthetically pleasing can require careful manual work even with color tools that account for lightness changes. Smart, Wu and Szafir (2019) describe an algorithmic approach that can emulate such manually crafted quantitative color scales and show that it is possible to in this way create quantitative palettes that outperforms colors generated by common mathematical approaches.

2.1.3.1 *Contrast and separation of colors*

Quantifying the difference between individual colors is necessary for generating color scales with recognizably different colors. The difference is termed ∆E — (Delta E, dE) and represents a measure of change in visual perception between two colors. The scale of Delta E generally ranges from 0 to 100, with values under 1 representing differences that are not perceptible. The current standard for accurate color difference algorithms is Delta E 2000, proposed by the CIE organization (Schuessler, 2019). In the 2.4.0 version Chroma.js uses an implementation of this algorithm (Aisch, 2022a).

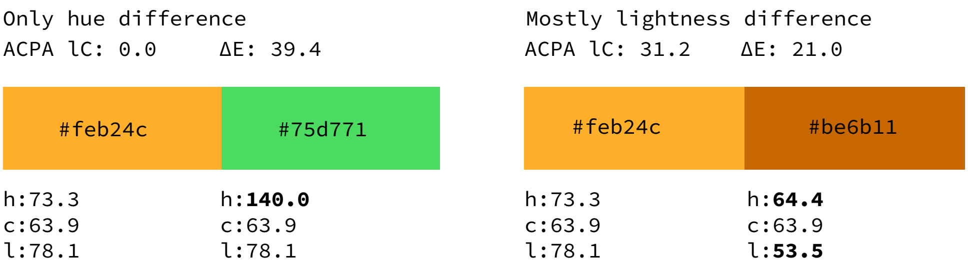


Figure 2.14. Comparison of lightness contrast and Delta E (ΔE).

Here it is necessary to clarify that contrast and color difference are two related, but separate concepts. Contrast refers to differences in lightness between two colors, while Delta E also accounts for differences in hue and chroma. A pair of colors can exceed the Just Notifiable Difference (JND) threshold clearly while having identical lightness, i.e., zero contrast. This is illustrated in Figure 2.14. The pair on the left have a fairly large Delta E difference, while their lightness contrast (lC) is zero (see p. 36 for further details on the contrast calculation). The colors on the right have higher contrast than Delta E.

For practical use differences in color must exceed the just notifiable difference (JND) threshold by a fair degree. What the exact JND value is and how much it should be exceeded depends on the context and how Delta E is calculated, which means that given values should be taken as general guidelines (Schuessler, 2019). Schuessler (2019) argues that values from 2–10 are "perceptible at a glance" based on self-conducted tests.

The *Leonardo* online applications[13] (Baldwin, 2022) correspondingly uses a Delta E of 11 as the lower cut-off value for a difference that will be safely distinguished. Such a small difference is however below any practical contrast requirements, as Figure 2.15 demonstrates. The first `#fee8c8` and the last color `#e34a33` in a three-class ColorBrewer *OrRd* scale[14] have a Delta E of 39.6 between them, but still just barely exceed the Web Content Accessibility Guidelines (WCAG 2 2018) AA level for UI components and graphics. A grey color with the hex value `#757575` just passing WCAG criteria AA for regular text (24px/19px bold and below) has a Delta E of 37.2 against white.

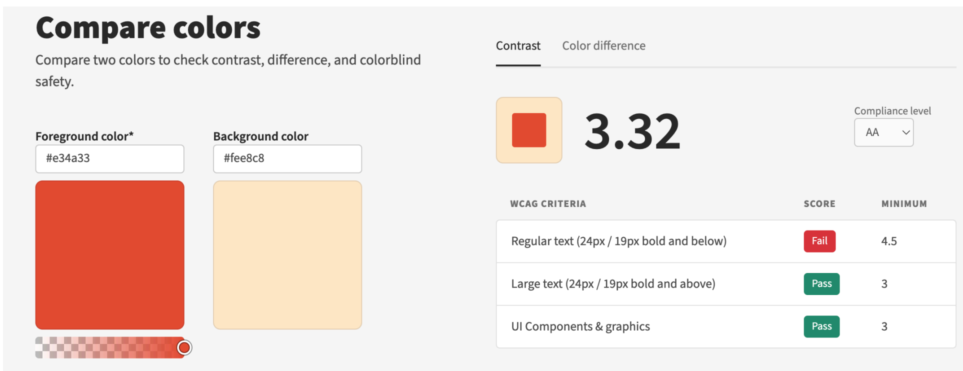


Figure 2.15. Comparison of the first and last color in a three-step ColorBrewer OrRd scale in Leonardo (Baldwin, 2022). Screenshot.

13 Leonardo uses Chroma.js with modifications for color calculations. https://leonardocolor.io/
14 https://colorbrewer2.org/#type=sequential&scheme=OrRd&n=3; Brewer (2013)

In the case of text, color contrast differences need to be much larger. Somers (2021b) argues that the difference should exceed the JND at least by 10 times, referring to work by Christen and Abegg (2017), in which increasing contrast was found to support higher reading speeds under simulated low vision conditions for text in different shades of grey. The lowest tested value was a barely discernible light grey with a Delta E of 2.4 while reading speeds rapidly improved up to a Delta E of 39.9 — some 16 times higher[15]. Without simulated vision impairments, contrast had only little effect on the ease of reading (Christen and Abegg, 2017).

2.1.3.2 *Issues with color contrast in visualization design*

An issue with applying mathematical color science models to design and visualization work is that they do not control for important real-world factors that affect color perception. These include differences in viewing conditions such as lightning and display means (e.g. monitor type, printing techniques) in addition to confounding factors such as surrounding colors and varying sizes of the colored targets (Szafir, 2018).

In an attempt to handle some of these practical factors Stone, Szafir and Setlur (2014) created a simple model for computing color differences in relation to size using Euclidean distance in the L*a*b* color space for △E adjusted, through crowd-sourced data from Mechanical Turk users. A survey was used to study perceived color differences at different sizes. Using data from the 624 participants they created a parameterized noticeable difference (ND) as a linear function of distance in CIELAB space for each studied target size. While the Euclidean distance model in itself is considerably less accurate than Delta E 2000 for color differences the authors argue that their parametrization makes their model useful for practically assessing color differences, which also outperforms traditional models (Szafir, 2018).

In the context of this thesis, the lower bound for differences in color that can be recognized is not of major importance, since any bivariate palette design should exceed those limits by a significant amount for all colors. Another consideration is that high contrast color scales may be undesirable in some cases. For instance, Careri (2022) notes that readers with dyslexia suffering from scotopic sensitivity syndrome may find strong color contrasts stressful and prefer color scales that are limited in hue variation.

Currently the Web Content Accessiblity Guidelines (WCAG 2 2018) do not define a specific level of contrast required for colors used maps and data visualizations[16], but an established approach is applying the WCAG AA contrast criteria of 3:1 for graphical objects[17] in these cases (Elavsky, 2021c).

15 Text luminance was given in 8-bit greyscale where 0 is defined as black and white as 255. The highest tested luminance was 243 and the second-lowest 110 (Christen and Abegg, 2017), i.e., corresponding to `rgb(243, 243, 243)` and `rgb(110, 110, 110)`.

16 Accessibility Guidelines Working Group (2021), G111: Using color and pattern

17 Accessibility Guidelines Working Group (2018); Success Criterion 1.4.11 Non-text Contrast

The forthcoming update to the Web Content Accessibility Guidelines, WCAG 3 (Spellman *et al.*, 2021), is proposed to use an improved algorithm for calculating contrast called the Accessible (or Advanced) Perceptual Contrast Algorithm[18] or ACPA (Somers, 2022) building on the work of Stone, Szafir and Setlur (2014) and others. It is perceptually uniform. If the APCA contrast value for two different color pairs is the same, it represents similar perceptual contrast, which is not the case for WCAG 2.x contrast calculations. ACPA also involves taking spatial frequency into account, which means different contrast values for different sizes — small objects (texts) require higher scores than large colored surfaces.

Using APCA the minimum value for usable color contrast is 15. The APCA algorithm or a version of it is likely what will be practically used to assess colors in a web environment in the future WCAG 3 (Somers, 2021a), but for general robustness a difference corresponding to the old WCAG2.1 contrast of 3:1 will be used is this thesis as a cut-off value. Different web tools exist for easily calculating contrast values[19].

To further ensure separation of colors a stroke is commonly added to colored features in a chart (Shixie *et al.*, 2020; see e.g., Organ, 2021; Elavsky, 2021a). It is also a technique recommended by the WCAG[20]. This is generally necessary, as the differences between sequential colors in well-known schemes such as the yellow-orange-red *YlOrRd* from ColorBrewer[21] do not satisfy even the lowest 3:1 contrast requirement between all the individual color steps, despite using the minimum three classes, as seen in Figure 2.16. The lightest yellow has too low contrast both to a white background and to the middle color. Light starting colors of typical color scales also tend to fall below the recommended APCA score of 30 for graphical objects when compared against a light background. If a dark background is used the opposite problem generally occurs, with dark colors being insufficiently different from the background.

18 A in name changed from Advanced to stand for Accessible in april 2022. (Somers, 2022)

19 WCAG 2 contrast values can be determined for example with the previously mentioned Leonardo or WebAIM's contrast checker utility: https://webaim.org/resources/contrastchecker/; WebAIM (2021). The APCA contrast algorithm also has a web utility available at www.myndex.com/APCA/

20 "Provide sufficient contrast at the boundaries between adjoining colors," www.w3.org/WAI/WCAG21/Techniques/general/G209

21 https://colorbrewer2.org

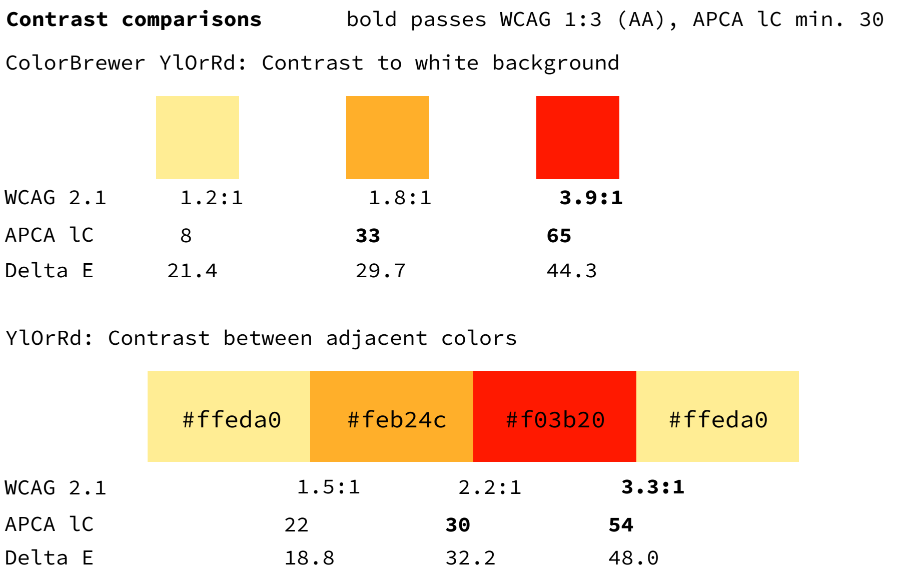


Figure 2.16. Contrast calculated with WCAG 2 and ACPA methods between sequential color steps in a three-step YlOrRd ColorBrewer scale.

This is a challenge for color encoding information in visualizations. The WCAG 2.1 Success Criterion 1.4.1 ("Color is not used as the only visual means of conveying information [...] or distinguishing a visual element.") can be satisfied, if the used colors differ in *both* hue and lightness, but as Figure 2.15 shows, this will be limited to only at most a few colors without adding strokes. Notably the requirement only applies to adjacent colors and is satisfied if they are separated with strokes. No value for minimum color differences in other situations is given in WCAG2.1, even though this in some cases would be necessary to ensure that color legends are legible[22].

2.1.4 Visually distinct textures

Texture as a visual variable also has particular relevance to area-based mapping, as it can be used as a qualitative and to some degree as an ordinal encoding when applied to shapes. A texture or pattern[23] is here defined as being created by repeating a shape or *tile* with a design on it in that is repeated in a periodic tiling. While many kinds of tilings exist, tilings for pattern design in visualization and cartography generally tend to use regular, uniform rectangular tiles[24]. Textures or patterns are visually distinct if they differ sufficiently in their main three fundamental perceptual channel components (Ware, 2013, p. 202) — orientation, scale, and contrast. Figure 2.17. shows some monochrome sedimental lithology textures with different characteristics used for maps or charts where this can be observed. Texture 667 differs from 669 in orientation but is similar in scale and contrast.

22 Accessibility Guidelines Working Group (2018); Success Criterion 1.4.11 Non-text Contrast, www.w3.org/WAI/WCAG21/Understanding/non-text-contrast
23 The terms are often used interchangeably. Here I prefer to use *texture* referring to particular, designed textures and pattern as a more general term.
24 see e.g., Jones (2011) for an example of designing tiles for use on maps in the ArcGIS software.

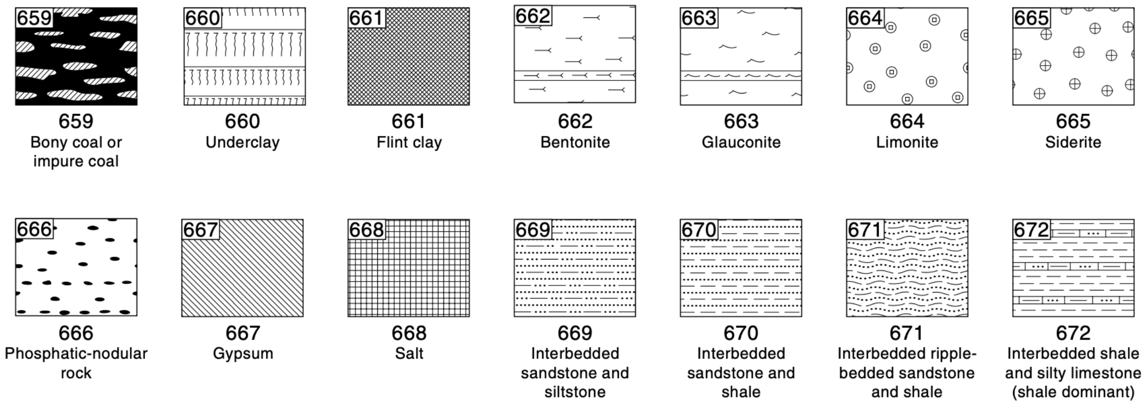


Figure 2.17. Examples of patterns or textures used for geologic map symbolization (Federal Geographic Data Committee, 2017).

Textures 664 and 665 have larger repeating elements, so their scale is larger compared to the others. All textures use black figures on white, so their contrast is the same. Ware notes that textures have other important dimensions as well — randomness being particularly relevant. The regular grid texture 668 is clearly different from the more organic 671.

Without employing color or contrast changes, scale and orientation become the most important factors for differentiation of textures. The apparent scale of a pattern is dependent on the spatial frequency, which means how often the pattern repeats per degree of visual angle. For this reason, it is also directly dependent on the viewing distance (Ware, 2013, pp. 59–60). One degree of visual angle at a viewing distance of 60 cm is about 1 cm[25]. Figure 2.18 (right) shows how the visual angle is calculated and two sine wave patterns with different spatial frequency (left).

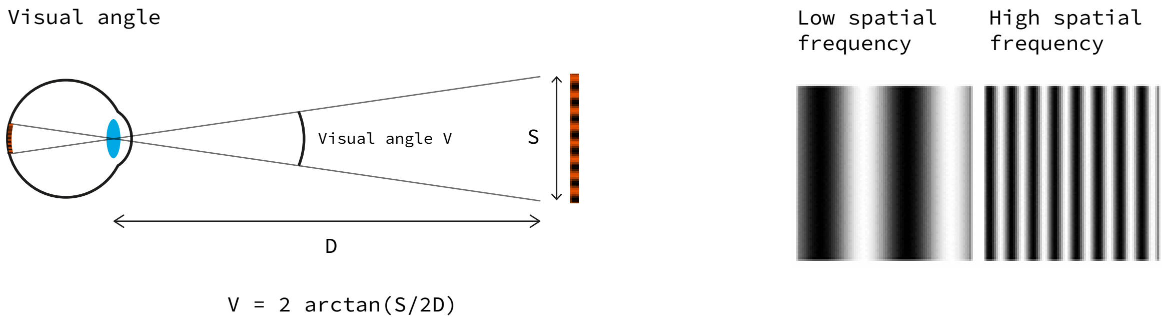


Figure 2.18. Illustration of visual angle with formula (left) and two patterns of different spatial frequency but same contrast (right).

Texture orientation and texture size variations in adjacent textures can also cause contrast effects — like optical illusions where a texture appears more fine-grained on a coarser texture background or where line orientations appear to change. These limitations lead Ware (2013) to suggest that visually distinct textures should differ by a factor of 3 in overall spatial frequency and by at least 30 degrees in orientation. In the paper *Quantitative Texton Sequences for Legible Bivariate Maps* Ware (2009) proposes a bivariate

25 See https://elvers.us/perception/visualAngle/ for a visual angle calculator.

display scheme using a combination of a color sequence together with repeated texture symbols designed to be legible and perceptually ordered — termed quantitative texton sequences or QTonS. These symbols (two variations are shown in Figure 2.19) are designed to be applied to a grid (rather than as a shape fill), which solves issues related to how textures interact with borders, but also limits their use.

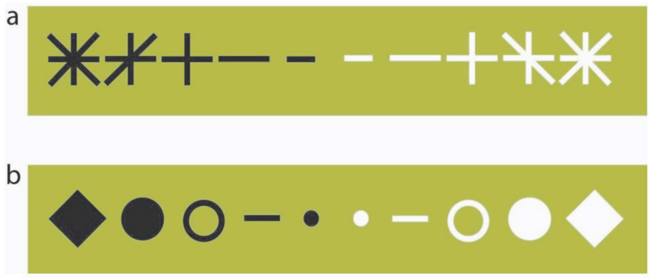


Figure 2.19. The ordered texture symbols or QTonS proposed by Ware (2009).

As Shixie *et al.* (2020) describe, textures can also be used as an alternative to colors in response to the accessibility issues that rise from adjacent colors in graphics — i.e., for satisfying the "no use of color alone" success criterion 1.1.4 in the Web Content Accessibility Guidelines[26]. However, the use of textures in visualizations can be perceived as disturbing, especially if the textures are high-contrast, and possibly may even trigger epileptic seizures in viewers at risk (Elavsky, 2021b). There is empirical evidence that "stressful" high-contrast striped lines even affect brain activity, particularly if the spatial frequency is about 3 cycles per degree of the viewing field (Huang and Zhu, 2017). At a normal screen viewing distance of around 60 cm this corresponds to three cycles per centimeter — in other words three dark and light bands per centimeter. According to Huang and Zhu (2017) this is specifically a property of textures with straight stripes — a checkered pattern with similar spatial frequency is not perceived as equally stressful. Presumably the effect is also weaker for random textures with organic shapes, like zebra stripes.

There are significant conflicting demands concerning the application of textures to visualizations. Elavsky (2021b) notes that much testing and research is needed to establish effective guidelines for use of texture that not only technically satisfies requirements but also creates practical and visually satisfactory results.

Franconeri *et al.* (2021) discuss the practice of using textures as additional redundant encodings and suggest that adding them inadvertently may lead to graphics that have no additional benefit or become confusing even for users without vision impairments. This may mean that following and formally satisfying the "no use of color alone" criterion can result in visualizations

26 Accessibility Guidelines Working Group (2018), Success Criterion 1.4.1 Use of Color: "Color is not used as the only visual means of conveying information, indicating an action, prompting a response, or distinguishing a visual element."

that are less understandable than ones that only employ color for encodings. As a counterpoint Careri (2022) writes that neurodivergent readers might find patterns pleasing and helpful if applied to visualizations in a way that reinforces hierarchies.

One approach to handling these possibly conflicting demands is allowing users to toggle textures at will, as Elavsky (2021b) suggests. This approach does leave the use of textures in non-interactive contexts as an unresolved question.

2.1.5 Integral and separate visual dimensions

The theory of *integral and separable dimensions* proposed by Garner (1974) is according to Ware (2013, p. 162) a useful additional framework for approaching graphical encodings and glyph designs. The possible visual variables of a glyph are recognized to be interrelated and affect each other. Carswell and Wickens (1990) summarize: “Two physical attributes that correspond to a single perceptual code are integral, whereas two physical attributes that are each associated with distinct perceptual codes are separable.”

Integral and separable dimensions have been studied by speeded classification tasks, where test subjects are asked to rapidly sort objects (glyphs) by particular visual criteria (e.g. pick out all objects with the same size) (Ware, 2013, p. 164). According to Ware the notion that visual variables fall into two clearly different categories is too simplistic. In reality visual variables will fall somewhere on a range between very integral to very separable, as illustrated in Figure 2.20. The lightness-hue pairing is missing from Ware’s graphic (2013) but based on work by Burns (2014) it may belong below the red-green and yellow-blue pairs. Group location cannot be used for encodings on thematic maps, although it is involved in intermediate-level map reading tasks.

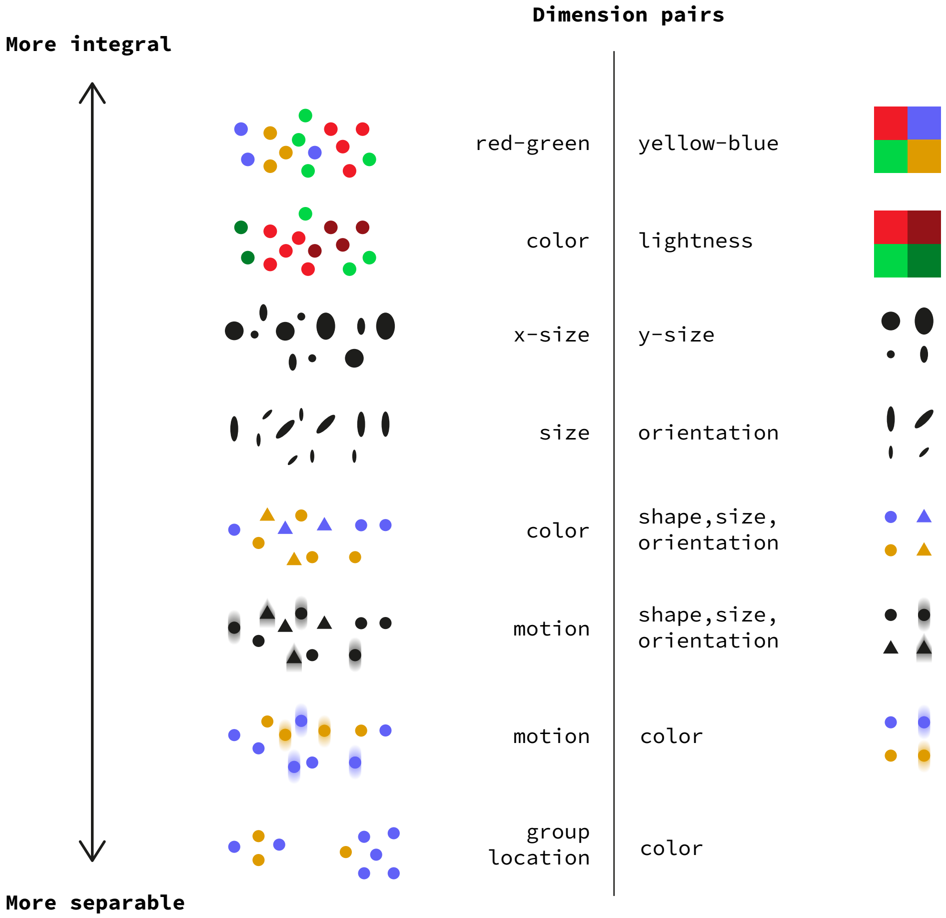


Figure 2.20. Demonstration of glyph coding pairs sorted from most integral at the top to the most separable at the bottom (after Ware, 2013, p. 167).

According to Ware this ordering and distinction should still be taken as roughly indicative, since there are many exceptions. Ware concludes that while the integral–separable description essentially is the same as the channel theory, which has a firmer experimental foundation, it has the advantage of providing practical design recommendations.

Integral display dimensions tend to be perceived as a whole and are hard to separate. For instance, width and height as components of a shape tend to be seen together, so that two rectangles or ellipses of different sizes that have similar proportions will be interpreted as belonging together. Color hue is strongly integral — it is hard, but possible to perceive the degree of redness in a blue color.

In the case of *separable display dimensions*, it is relatively easy for a viewer to attend to one dimension of a visual shape in isolation: size and color is a typical example. This can be referred to as *analytical processing*. Ware (2013, p. 163) notes that most empirical research has been conducted on pairwise combinations of graphical qualities, while the interaction of three or more dimensions has received very little attention. Burns (2014) notes that there are many unknowns related to how dimension pairs interact depending on situational effects from the tasks assigned. Adding redundant encodings can sometimes lead to unexpected interactions.

The conditions for selectivity between dimension pairs have been expanded to include two additional terms in addition to the integral and separable: the intermediate conditions of *configural* (Carswell and Wickens, 1990) and *asymmetrical* combinations (Nelson, 2000b, 2000a).

Configural combinations are described by Carswell and Wickens (1990) and MacEachren (1995) as being of optional separability. The viewer can attend either to the combination of visual dimensions, or to one of them in separation. According to Nelson (2000b) typical configurational symbols (see Figure 2.21) employ the same variable twice in one glyph — size/size (left) or lightness/lightness (right). In this case, it is possible to attend to either of the variables in such symbols, but they also interact strongly within each symbol forming *emergent dimensions* such as the overall lightness or shape of individual glyphs.

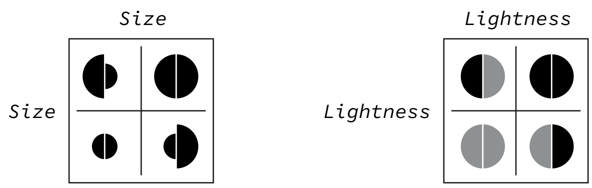


Figure 2.21. Glyphs using configurational combinations: size/size and lightness/lightness.

Carswell and Wickens (1990, pp. 166–167) did not find examples of true integrality of visual dimensions in their study and hence argue, that the visual dimensions employed in most visualizations should be characterized as having configurable characteristics.

In asymmetrical combinations one of the paired variables is visually more distinct than the other. According to Nelson (2000b) asymmetry means that classification becomes easier when two symbol dimensions are correlated in one direction, while correlation in the opposing direction does not have the same effect. Emergent properties created by the two visual dimensions thus improve the ease of sorting, but only when they correlate positively. Elmer (2012) gives numerousness and size as a primary example: when symbol count and size increase in tandem they effectively reinforce each other, while combinations of low symbol count and large symbol size can be hard to distinguish from the opposite of high symbol count and small symbol size, as both combinations result in similar total coverage (the emergent property). Texture and hue can also function in this manner: the hue is harder to distinguish in a texture that has a low total coverage.

Ware (2013, p. 168) notes that the main usefulness of the theory of integral and separate dimensions lies in its apparent simplicity and ease of application in design. According to Ware there are multiple irregularities involving asymmetric combinations and exceptions in the theory, and a lack of an underlying underlying explanation mechanism. Interestingly, neither Elmer (2012) nor Nelson (2000b) refer to Treisman and Gelade's (1980) work on conjunction searches.

As discussed under Visual variables: definition and use in maps (p. 23) there is a limited number of such variables that are practically useful. Ware (2013, pp. 171–172) suggests that the maximum number of dimensional data that may be displayed clearly is eight: this would employ "color, shape, spatial position and motion to create the most differentiated set possible." Further, the number of individual steps that can be discerned in each dimension has strong limitations. Color steps are limited to a maximum of around 12 (Ware (2013), p.126). Concerning size, no more than four steps can be reliably separated (see also Robinson, 1995, p. 413). The number of easily separated orientation steps is about four — the maximum being around six since the difference in orientation needs to be at least 30 degrees and opposing orientations are easily confused (Ware, 2013, p. 204). Ware ends up suggesting at most 32 easily distinguishable alternatives for shapes assuming no difficult conjunction searches. As an additional complicating issue is the semantic appropriateness of the visual dimensions which make some mappings more appropriate than others, e.g., size has a natural association to amounts and orientation to directions.

2.2 Definition of maps

Before discussing types of maps in more detail, a definition of maps is needed. Several such definitions have been proposed. Kraak and Ormeling quote this rather inclusive one by Board (1990): "[a map is] a representation or abstraction of geographic reality. A tool for presenting geographic information in a way that is visual, digital or tactile." (cited in Kraak and Ormeling, 2010, p. 41)

A somewhat more narrow definition of a map would be a "measurement-based miniature image of a geographical area". (Koponen and Hildén, 2019, p. 137)

A detailed discussion of cartography is outside the scope of this thesis. As accessible introductory resources for further reading on cartography one may recommend the books *Principles of map design* by Tyner (2010) or *Making Maps: A Visual Guide to Map Design for GIS* by Krygier and Wood (2016).

2.3 Types of maps: thematic maps and general use maps

There are several ways to divide and classify different types of maps. Thematic cartography — the creation of thematic maps — first began to emerge as a separate branch of cartography with the maps showing trade winds and the strength of the Earth's magnetic field published by astronomer Edmond Halley from the 1680s onward (Fig. 2.20).

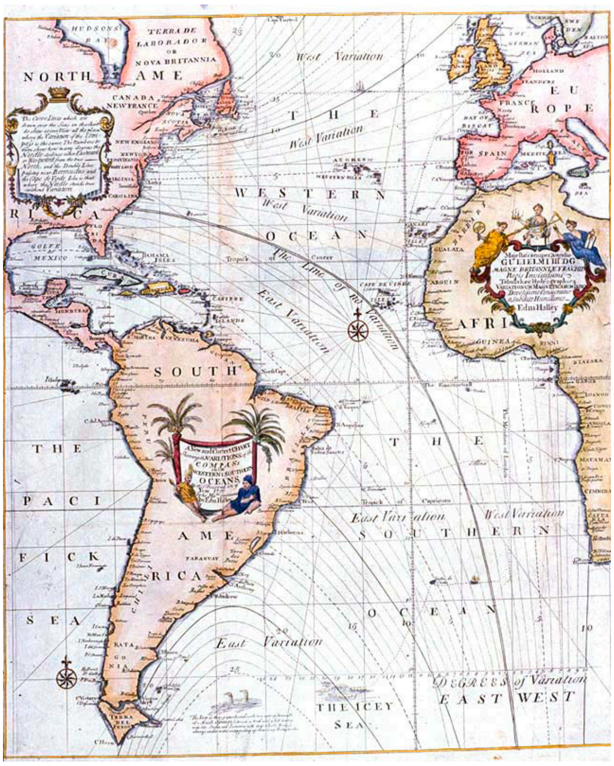


Figure 2.20. Halley's map of the magnetic field: an early contour or isoline map (Halley, 1700).

Despite the existence of early examples like this, it was only in the 19th century that thematic maps became commonplace. Tyner considers thematic maps to now be the main type of map used in for example newspapers, journals and textbooks. (2010, p. 7)

Tyner (2010) employs three categories based on map function: *general-purpose maps*, *special-purpose maps*, and *thematic maps*. General-purpose maps or reference maps[27] serve the purpose of representing general geographical features of an area, both man-made and natural. In Tyner's classification special-purpose maps typically show a detailed view of a small area and are created for a specific use case (such as cadastral maps, geologic maps, or maps for route finding). According to Tyner the term *thematic maps* is now

27 Slocum et al. note (2014, p. 2) that distinguishing thematic maps from general reference maps largely is a convenience of categorization: a reference map could be seen as a *multivariate thematic map* that display many attributes at once, and spatial patterns of attributes can be observed in topographical maps, in addition to locations.

widely accepted as the umbrella name for maps displaying quantitative or qualitative data against a spatial background that provides the reference for locating the mapped distribution. (Tyner, 2010, p. 7)

Tyner uses the terms *cartogram* to refer to “a geographic representation on which the size or distance isn scaled to a variable other than earth size or distance units” (2010, p. 189). Tyner further identifies *diagrams* as general term for schematic graphics that for instance show idealized climate zones or route connections such as subway maps (2010, p. 198).

Koponen and Hildén propose the term *data maps* (2019, p. 139) as a more general term that in addition to thematic maps also includes cartograms and other map-like visualizations. *Data maps* are then all maps or map-like visualizations that allow the study of a phenomenon with a geographical distribution — such as population density — by displaying geographic attributes, usually in addition to some background topographic information. By this definition thematic maps refer specifically to geographic representations of data that also include more or less geographically accurate dimensions.

In this thesis I will employ the following narrowed-down definition of a data map, which covers both geographically accurate thematic maps and cartograms:

A data map is a map or map-like visualization which displays geographic distribution of a phenomenon using quantitative or qualitative data that is encoded in an explicitly determined systematic manner using one or more visual variables.

2.4 Univariate and multivariate data maps

All data maps are in a sense multivariate, as they display at least three dimensions: horizontal and vertical location plus some additional data dimension. Here univariate and multivariate maps will refer to data maps that display either one or several additional data variables in addition to location variables.

In a univariate data map *one* data dimension is displayed in addition to geographical location using some visual variable. In the case of a choropleth map this is typically done by coloring the mapped regions by a data dimension such as population density. A univariate graduated symbol map uses symbol size to denote a data dimension. A map is not considered multivariate when two or more different visual variables are used to display the same data dimension — such as using both a color scale and symbol size to display values on a symbol map. This is what Roth (2017) calls redundant symbolization, which can be employed to strengthen the visual encoding of the mapped dimension.

Tyner considers all thematic maps, which display more than one additional data variable as being multivariate. A simple multivariate map uses two different symbols for two variables to compare them on a single map, such as adding proportional symbols to a choropleth map (Tyner, 2010, p. 178). Multivariate maps can show qualitative or quantitative data or a combination thereof. This thesis is concerned with the case of bivariate quantitative displays. Further, the discussion will be limited to bivariate data maps along with the addition of the special case of trivariate cartograms where two additional variables are displayed on shapes sized by population.

Here I also make a distinction between contextualizing background map information and thematic data: a choropleth map of postal districts with roads, bodies of water and names of neighborhoods is not in this sense multivariate, even if it does contain multiple layers of data. A data map is here considered univariate, if it displays exactly one data dimension (such as population density) using graphical marks at some geographical locations, even when all background information is taken out.

I focus on bivariate data maps that displays two data dimensions for each individual mapped region by using different visual variables in a single graphic. Therefore, *small multiples* (Tufte, 1990, p. 67) are also not here considered, even though they can be used to display multivariate geospatial data as shown in Figure 2.22.

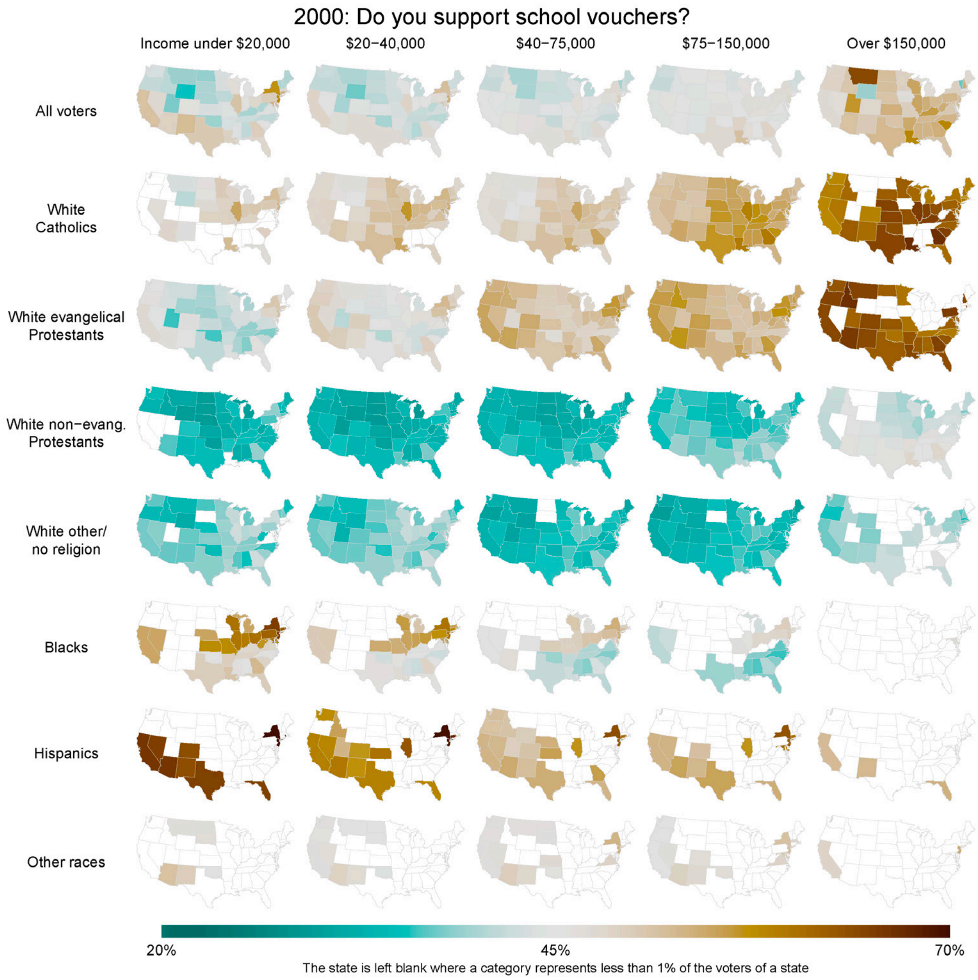


Figure 2.22. Small multiples map visualization showing support for school vouchers by states for different social and income groups in the United States in 2000. From Gelman (2011).

In a typical small multiple each component graphic employs the same visual variables. Figure 2.22 is a complex example, where varying support for school vouchers are shown for the combination of 8 respondent groups and 5 income groups on 40 choropleth maps of the US using the same color scale and positioned in a tabular grid. Interactive maps where layers can be toggled or animations that go through different data sets are by the same reasoning also not considered.

There is no comprehensive literature on bivariate maps, but typically textbooks on thematic cartography will include a number of examples. However, apart from including the bivariate choropleth map there is little agreement between texts on the number and types of bivariate maps presented. Elmer (2012) summarizes bivariate maps discussed in six different textbooks (Dent, Torguson, and Hodler 2009; Fisher, 1982; Krygier and Wood, 2011; Robinson, 1995; Slocum et al., 2003; Tyner, 2010). To this summary in Table 2 I add Kraak and Ormeling (2010) and correct the omission of the bivariate choropleth from Robinson (1995) and the ray glyph from Tyner (2010):

Map type	*Tyner*	*Dent, Torguson, and Hodler*	*Fisher*	*Krygier and Wood*	*Slocum et al.*	*Kraak and Ormeling*	*Robinson (3rd ed.)*
Bivariate choropleth	●	●			●		●
Graduated pie charts (segmented point symbols)	●					●	●
Multiseries dot density	●	●	●				
Ray glyph	●				●		
Choropleth with graduated symbol	●		●				
Shaded cartogram		●		●			
Shaded graduated symbols		●		●			
Bar graph (or composite diagram map)			●			●	
Isoline with graduated symbols				●			
Rectangle map (height/width)					●		
Multiseries graduated symbol			●				

Table 2. Summary of bivariate maps mentioned in 7 cartography textbooks.

As Elmer (2012, p. 12) notes there is fairly little agreement between the textbooks on the map types mentioned, and none offer a comprehensive typology of bivariate maps. The bivariate choropleth is the most common type described in 4 out of the 7 books. Three types are mentioned only once. A more detailed assessment would require a survey of published maps that is beyond the scope of this thesis, but as Figure 2.23. illustrates the bivariate choropleth also appears to be one of the most popular types of bivariate map in general use at the moment of writing.

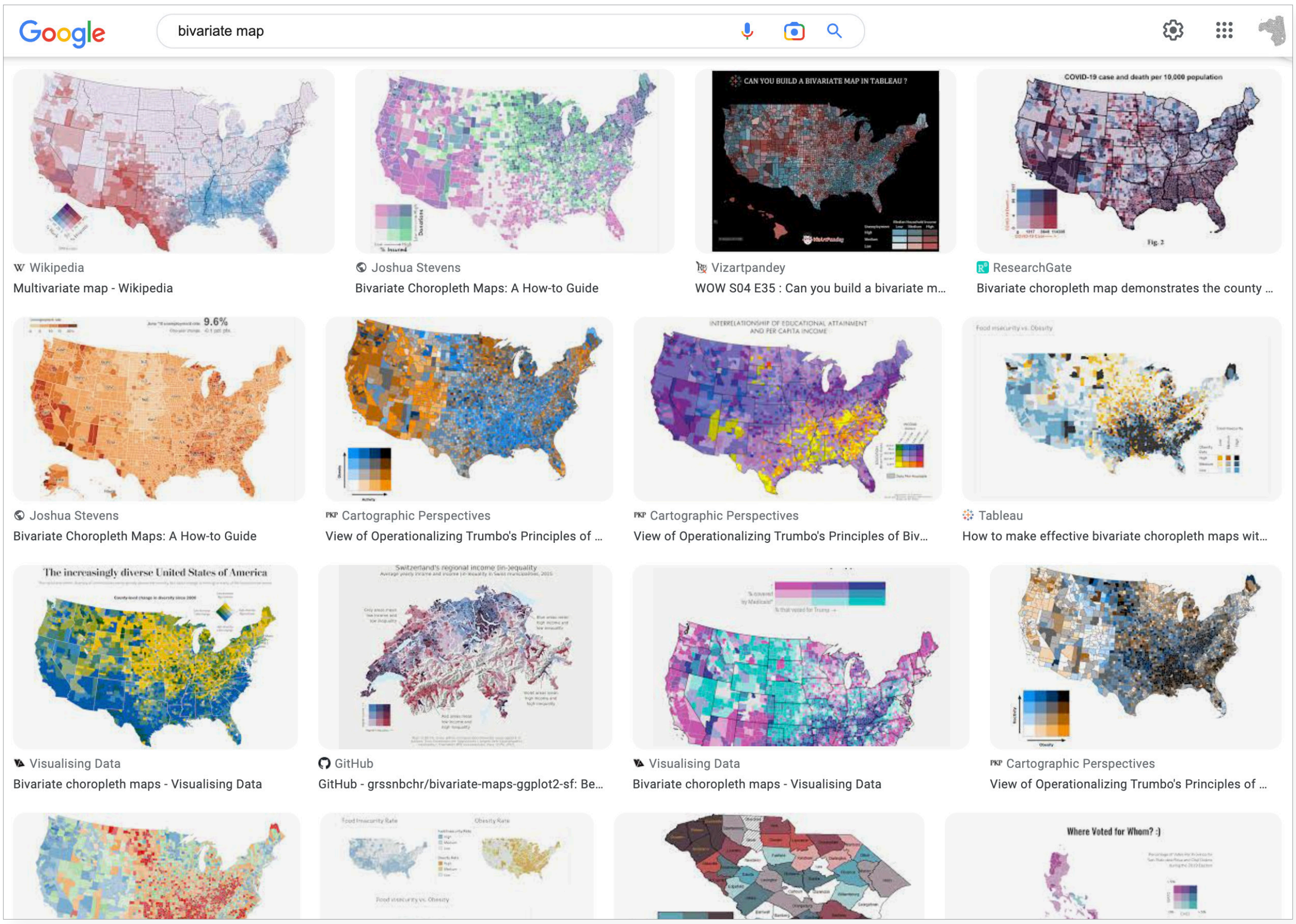
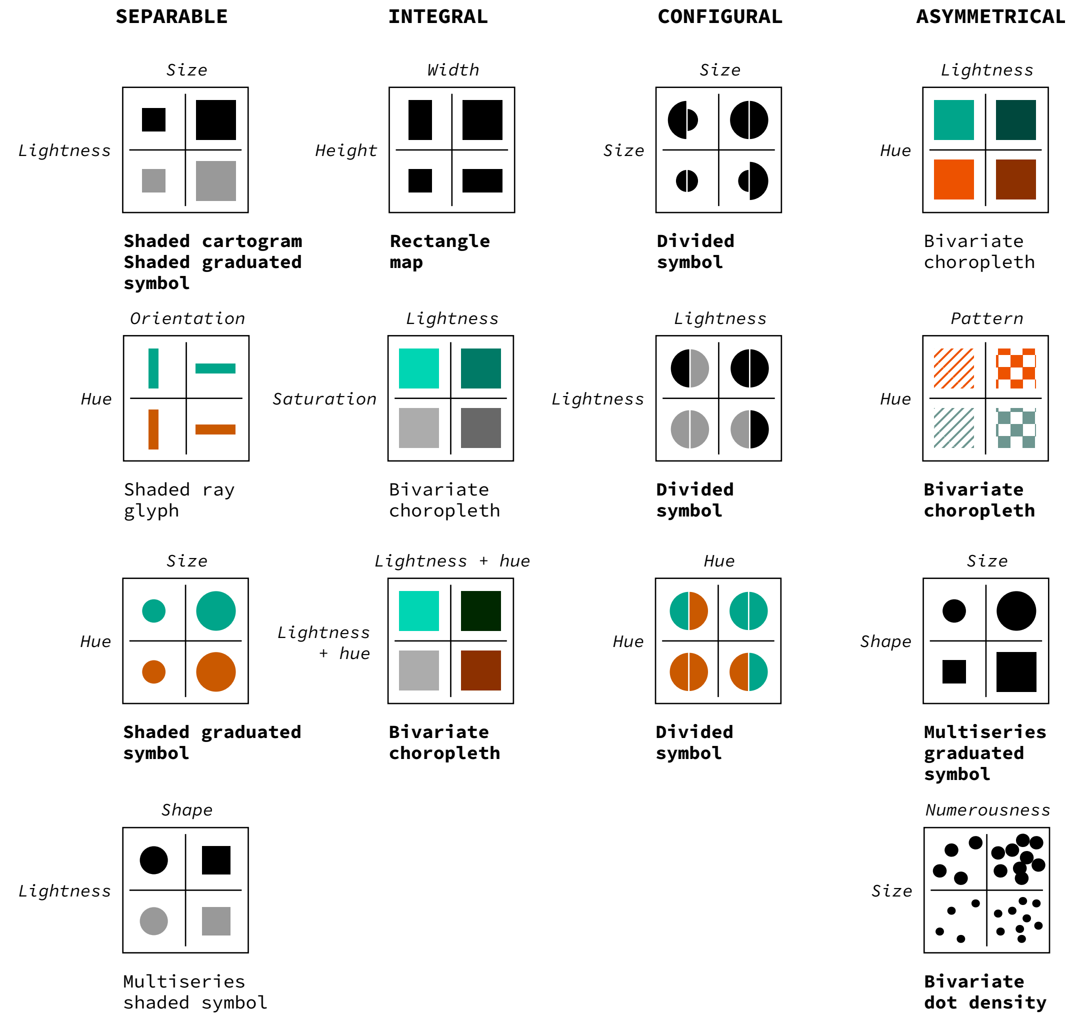


Figure 2.23. The bivariate choropleth is the dominant result when using Google to search for bivariate map images (November 2022).

The possible design space of bivariate maps can also be more systematically explored through a framework of combinations of the *visual variables* that were discussed previously. The number of theoretically possible configurations becomes exponentially large, but fortunately not all combinations are practical or tenable.

A more systematic attempt at a taxonomy of bivariate maps based on visual variables is included in Nelson (2000b), here reproduced in modified form based on the additions made by Elmer (2012) as Figure 2.24. The bivariate maps are illustrated with small 2×2 legend graphics and arranged into four columns based on four different relations between the variable pairings — **separable**, **integral**, **configural** and **asymmetrical.** Separable pairings of visual variables should enable relatively effortless study of either variable in isolation, integral ones tend to be seen together, configural variable pairings create a new combined dimension, and in asymmetrical pairings one visual variable is more salient than the other. I return to discuss these combinations in more detail in the section “Main and emergent visual variables” on p. 62. As Elmer (2012) notes, this categorization does not attempt to catalogue all possible bivariate combinations, but it covers many bivariate types that are in common use. These are labeled with bold type. The integral combination of hue and lightness on both axes is the most common bivariate choropleth type but is an addition to Nelson’s graphic (2000b). *Value* has



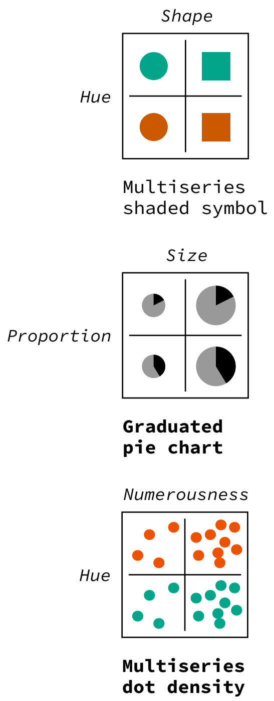


Figure 2.24. Visual variables in bivariate maps arranged by configurality after Elmer (2012).

been renamed lightness for consistency with the terminology used in this thesis.

I have left out the combinations of linear pattern and size from Figure 2.24. as bivariate linear symbols are outside of the scope of this thesis. Nelson's two examples of typeface combinations are also left out here, since these are essentially duplicates of shape/shape (typeface/style) and shape/size (typeface/size).

What follows is a more detailed description of some important types of data maps, how they are adapted to display bivariate data and some design considerations relating to them.

2.4.1 Areal unit maps

Areal unit maps take polygons representing geographical areas and encode quantitative or qualitative values using colors hues, shades or in some cases patterns applied to these polygons. Classified areal unit maps are generally preferable to the unclassified type, since only a limited number (usually around 7) of different hues can be reliably separated (Koponen and Hildén, 2019, p. 147).

Figure 2.25. shows example coloring schemes for univariate areal unit maps. Qualitative scale for qualitative data, binary schemes for binary (yes/no) data, or a single-hue, multi-hue, or diverging color scale for numerical data. Diverging scales are employed when the data range has a natural midpoint. (Brewer, 1994)

All areal unit maps essentially differ in how the areas are divided and are thus at least in principle adaptable to representing bivariate data. This is discussed further under Bivariate color scales for choropleth maps, p. 70.

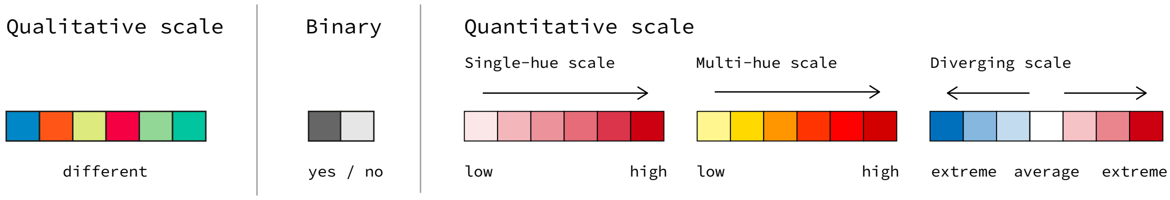


Figure 2.25. Examples of types of univariate color schemes used in choropleth maps.

Choropleth maps (Figure 2.26, left) use pre-existing regional divisions like municipalities or postal districts as statistical units and display aggregate values that are collected for each of these subdivisions using color or sometimes textures. According to Tyner (2010, p. 160) this is the most common and familiar kind of thematic map. Choropleth maps cannot show variation within enumeration areas, since each is colored uniformly based on its aggregate value. Choropleth maps are very sensitive to changes in enumeration areas, so in most cases data should be shown using the areal divisions that were used for data collection. Since areas vary in size, absolute values should not be visualized on a choropleth (Koponen and Hildén, 2019, pp. 148–150).

Dasymetric maps (Figure 2.26, middle) are similar to choropleth maps, but additional information is used to adjust the statistical areas and give a more realistic view of variations within the measured areas. In practice this can for instance mean that in a map dealing with population data regional divisions are modified by removing parts that are known to be uninhabited (Tyner, 2010, pp. 163–164).

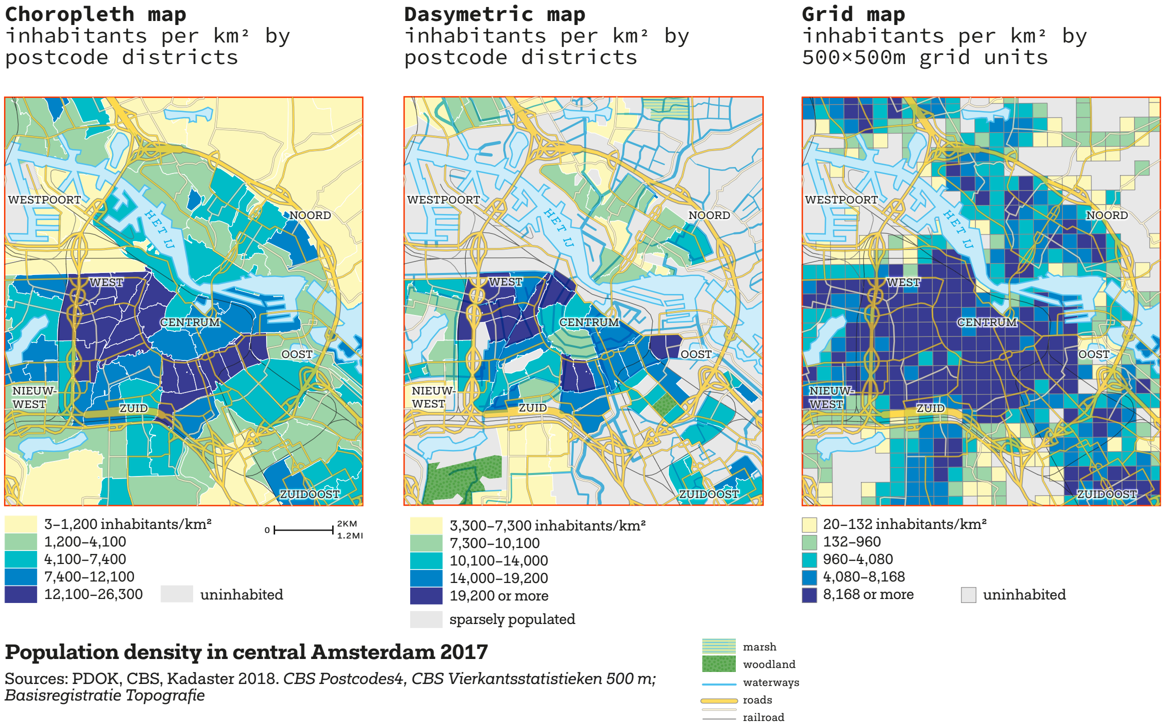


Figure 2.26. Examples of a choropleth map, a dasymetric map and a grid map representing the same region. After Koponen and Hildén (2019).

Grid maps (Figure 2.26, right) resemble choropleth maps where the visualized area is divided into cells of uniform size which are colored by the data dimension. Since grid cells are uniformly sized, this map type can display absolute values (Koponen and Hildén, 2019, pp. 150–151). Thanks to the uniform sizing of grid units grid maps may be more suitable to different bivariate displays than other areal unit maps. Especially symbol-based visual variables could be easier to apply to a grid map, although the advantage conceivably concerns color as well, since the issue of colors appearing different due to the varying sizes of mapped areas is avoided.

Contour maps or **isarithmic maps** connect points that have similar values with lines (Tyner, 2010, p. 171). An isoplethic or isometric map where the areas are colored based on the data values can be called an **area-class** or **zone map**. Figure 2.27 shows an example of a bivariate area class map of hours of sunshine compared with precipitation in the United Kingdom. Area-class maps may sometimes employ color scales with significantly more steps than a typical choropleth map, since successive values always are located next to each other (Koponen and Hildén, 2019, pp. 151–152).

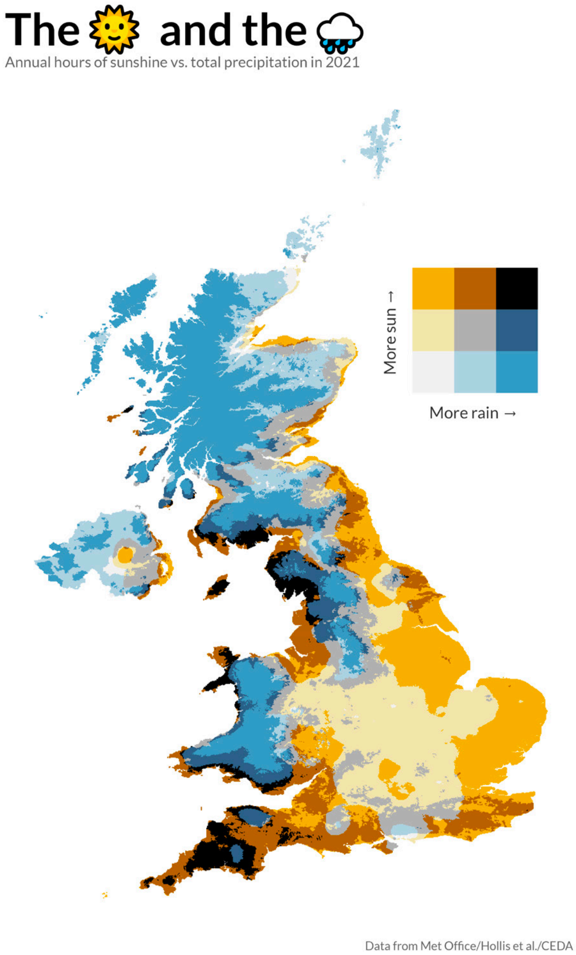


Figure 2.27. A bivariate area class map created by Colin Angus (2022). *The sun and the rain. Annual hours of sunshine vs. total precipitation in 2021.*

2.4.2 Other types: Cartograms and multidimensional glyph designs

Cartograms or anamorphic maps are diagrams that in some way resemble maps but distort geographical dimensions to communicate additional data variables. According to Tyner (2010, p. 198) a cartogram by definition employs distortions that are controlled and systematic. The two main types of anamorphic map are the **area** or **value-by area cartogram** and the **central point** or **distance cartogram**. Distance cartograms distort geographical distances from a predefined point based on travel times. Area cartograms are more common and generally useful. Because cartograms involve significant geographical distortion conventional chart types like a bar chart may be more effective in cases where no clear geographic patterns can be discerned (Koponen and Hildén, 2019, p. 155). Cartograms are not included in the typologies of multivariate maps discussed above, but have frequently been used for bivariate data, e.g., by showing an additional variable with color on a population cartogram. Tyner (2010, p. 194) calls them dramatic and eye-catching, but considers that there is a need for more empirical research on their effectiveness.

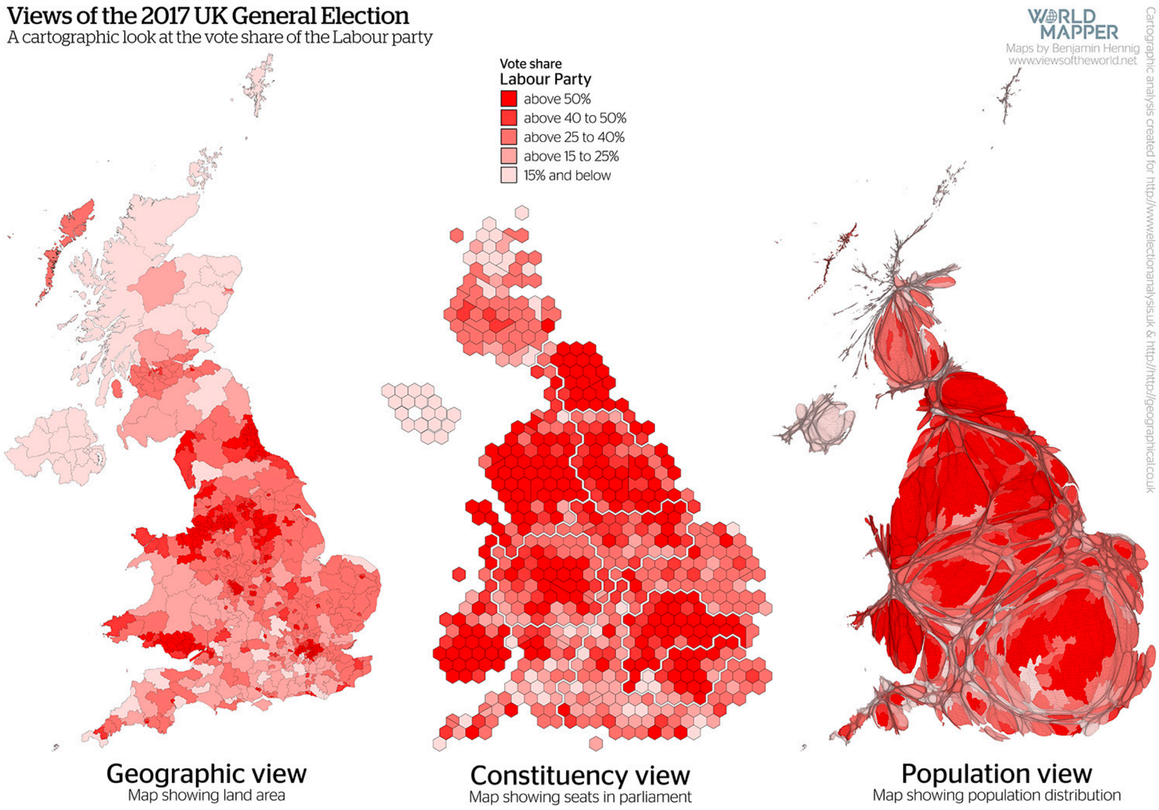


Figure 2.28. *Views of the 2017 UK General Election. A cartographic look at the vote share of the Labour Party.* A choropleth map (left) contrasted with a grid cartogram (middle) and contiguous cartogram (right) created by geographer Benjamin Henning (2017).

A **contiguous** area cartogram (Figure 2.28, right) renders geographical regions in their correct relative positions and maintaining border relationships, but with their surface areas programmatically distorted to match a data variable, such as population. Since the distortion always is relative to the actual sizes of the mapped regions, recognizing them is essential for interpretation, but this is in practice difficult (Tyner, 2010, p. 191).

Grid or **mosaic cartograms** are made by dividing regions into a number of uniform units (Figure 2.28, middle). The number of units may correspond to — for instance — population or to political constituencies, as in the illustrated example. These units are then positioned to render an approximate resemblance of the mapped regions. Constructing them is more difficult and usually involves manual composing, compared to contiguous cartograms which can be easily generated by software. The House of Commons Library (2022) has published a non-contiguous grid cartogram of the UK. (Koponen and Hildén, 2019, p. 154)

Non-contiguous cartograms (not illustrated) scale the areas of regions in place without distorting their geometries. Including the non-distorted outlines of the regions can provide a frame of reference, which makes them easier to interpret than contiguous cartograms (Tyner, 2010, p. 192).

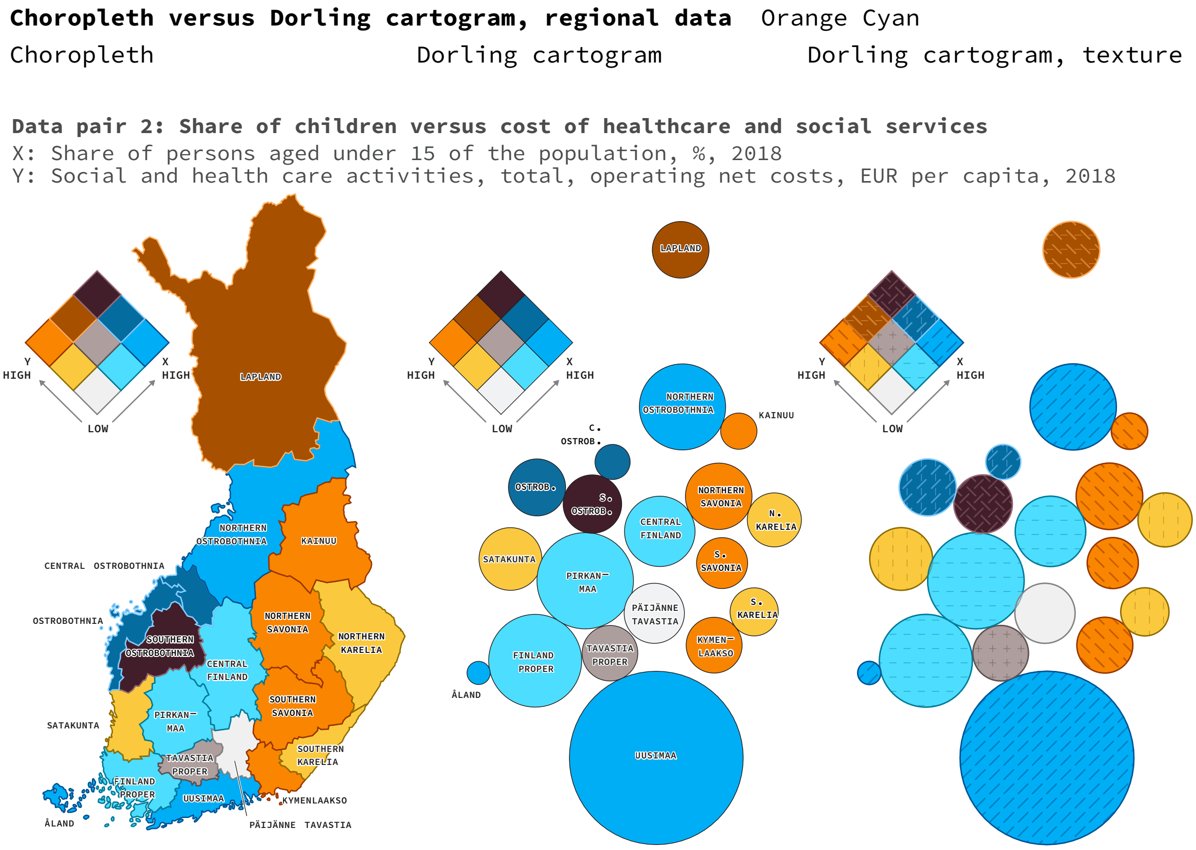


Figure 2.29. Two Dorling cartograms with bivariate color schemes compared with choropleth of same data set (left). The areas of the regions are sized by total population. Created as part of this thesis – see Chapter 5 for discussion.

The **Dorling cartogram** (Figure 2.29) is a variation of the non-contiguous cartogram where the mapped regions are represented by uniform shapes — usually circles[28] — with their areas derived from the mapped data dimension and placed in approximately correct relative locations (Tyner, 2010, p. 193). The shapes should generally not overlap but the resulting diagram is often far removed from the underlying geography.

In a cartogram the coloring can be the main visual variable, while the size provides secondary information, such as in a cartogram using color variables to display party vote share in districts sized by population. Applying a bivariate color scheme to a Dorling cartogram thus effectively creates a trivariate map.

Nusrat *et al.* (2018) demonstrate an alternative solution for bivariate Dorling cartograms that is illustrated in Figure 2.30. Areas are based on the combination of two data series (Y and X), whose relative contributions are indicated for each mapped region by size, line width and a color scheme with three categories. When both variables are in balance, the line is thin and rendered in grey, when the Y value is large, one color is used and another for X. Larger Y or X values are mapped to line width. From a small experiment with 23 participants they found this type of visualization to be viable.

28 The Demers cartogram is a variation that uses squares placed edge to edge instead of circles (Koponen and Hildén, 2019, p. 155).

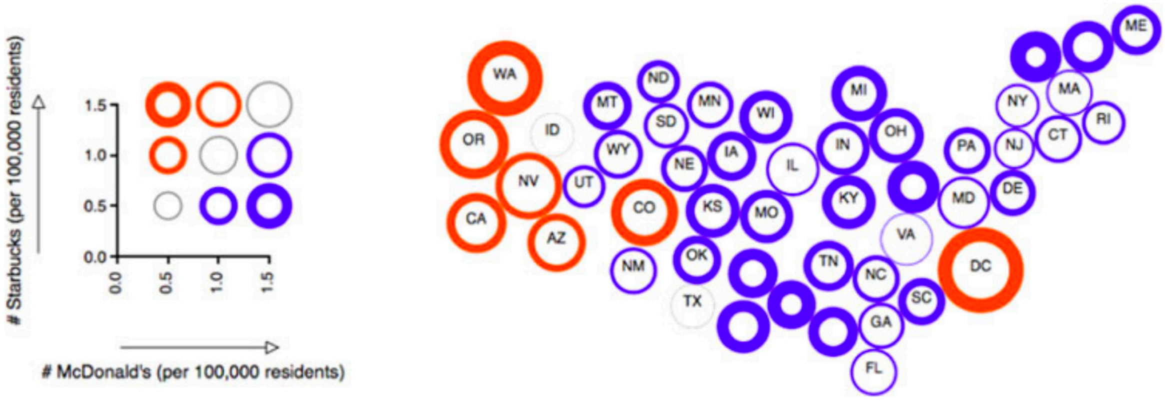


Figure 2.30. Example of novel bivariate Dorling cartogram created by Nusrat et al.: “A per-capita bivariate cartogram showing the distribution of Starbucks and McDonald’s shops per 100,000 residents in the US” (2018).

Statistical symbol maps (Figure 2.31, left), sometimes called diagram maps are proportional symbol maps that employ traditional statistical charts as point symbols. The most common type uses pie charts, also referred to as *segmented circles* or *segmented proportional circles*. Statistical symbol maps generally display data for at least three variables per location. Typically the circles vary in size based on total values like in a proportional symbol map while the wedges show percentages or fractions of the total (Tyner, 2010, p. 180).

Statistical symbol maps are considered confusing and hard to read compared to both proportional symbol maps and other chart types (Koponen and Hildén, 2019, p. 147). Kraak and Ormeling (2010) also actively discourages their use for this reason.

Chernoff faces (Figure 2.31, middle) are not listed in any of the discussed textbooks but may perhaps the best-known type of glyphs used on maps beside statistical charts. Herman Chernoff introduced these stylized face-like glyphs in 1973 and claimed that they would allow representing as many as 18 different variables using dimensions such as size, curvature and width of the mouth, angle of the eyebrows and so forth (Tyner, 2010, p. 182). Kosara (2007) argues that the particularities of human face perception which is very specialized is likely to make these glyphs difficult to understand. Different facial features are not perceived in a linear or differentiated fashion and cannot therefore easily be used as mappings for arbitrary data dimensions.

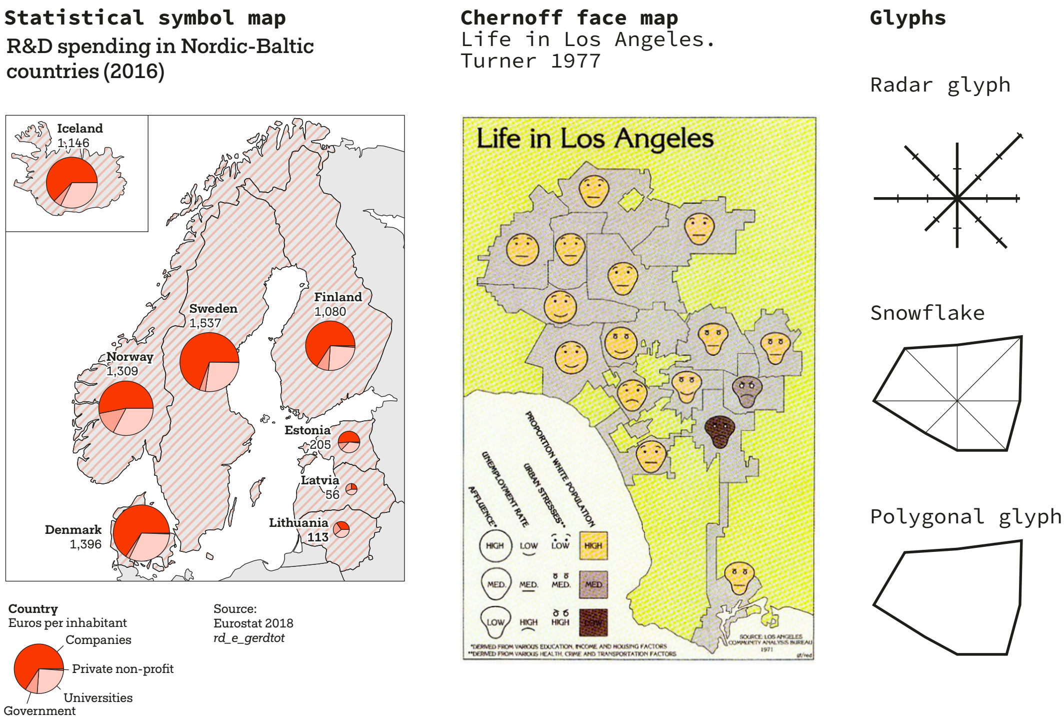


Figure 2.31. Statistical symbol map (left) (Koponen and Hildén, 2019), Chernoff face map (middle) (Turner, 1977), and radar graph variations (right; after Tyner, 2010).

Tyner (2010, p. 183) gives variations on *radar graphs* as an additional example of point-based symbology using glyphs that could be placed on a map (Figure 2.31, right). These are based on three or more rays starting from a central point. Data is encoded using the length of the ray, or in the case of polygonal glyphs, the shape — not the area — of the outline. These are functionally similar to standalone radar graphs — also called spider or star graphs. This chart type is considered to be mostly ineffectual and will not be considered further here — for details see e.g., Few (2005).

Empirical evaluations of Chernoff faces and star graphs — see, Lee, Reilly and Butavicius (2003), Morris and Ebert (2000) — suggests that such visualizations are both difficult and time-consuming to read, and lead to a high number of reading errors. Using them on a map, where the placement is determined by geographical features is likely to only make these issues even more severe. While other glyphs may be designed to be more contextually relevant and might conceivably avoid the particular problems associated with Chernoff faces and star graphs, it is reasonable to assume that the number of variables that can be easily appraised from glyph dimensions is rather limited (Kosara, 2007).

2.4.3 Symbol dimensionality

Geographical information is understood to exist in three categories based on their *dimensionality*:

- *point* feature data,
- *linear* (or arc) data, and
- *polygonal* feature data.

These correspond to the common division of geographic phenomena occurring as points, lines and areas (Tyner, 2010, p. 134). Additionally, data can be represented as volumes. Point data represents discrete data points corresponding to *point phenomena* recorded without additional spatial dimensions at defined geographical locations, such as the position of individual trees. Linear data is a one-dimensional representation of *linear phenomena* where the significant dimension is length, being either physical such as roads or streams, intangible like political borders or conceptual data aggregations such as traffic volumes. Polygonal data represents the two-dimensional areas and shapes of *areal phenomena* such as lakes, administrative regions or building footprints or conceptual data such as land use. (Tyner, 2010, p. 134; Dempsey, 2017)

Volume phenomena also cover geographical areas but include a quantitative third dimension. In the case of elevation, this is directly observable, but the third dimension can also be conceived as being conceptual in nature, in which case the representation forms a *statistical surface*. This means that data is collected with a *z* dimension in addition to the spatial latitude and longitude or x and y dimensions. A population map can be conceptualized as a statistical surface consisting of areas where the z-dimension (i.e., height) represents population density. (Tyner, 2010, p. 134)

The possible symbol dimensionalities corresponding to these phenomena are usually considered to consist of points (zero-dimensional), lines (one-dimensional) and polygons (2d). In addition to these, some authors (Stefan *et al.*, 2007; e.g., Tyner, 2010; Kraak and Ormeling, 2010) include rendered virtual volumes or surfaces (referred to as 2.5d) and actual 3-dimensional shapes (in the case of maps as physical 3d objects).

In thematic mapping, data is often encoded with a symbol that has a different dimensionality than the actual phenomenon displayed, as when areal data is displayed using point symbols placed on the center of the mapped region. An example is representing cities as circles with areas corresponding to their total population placed at their geographical center points. (Tyner, 2010, p. 142)

Based on this, all bivariate maps can be understood as the combination in one map of two visual variables (size, shape, color properties etc.) and two symbol dimensionalities. An example of this is shown in Figure 2.32, where the visual variables of area and lightness are combined into a bivariate symbology that uses point and polygon features.

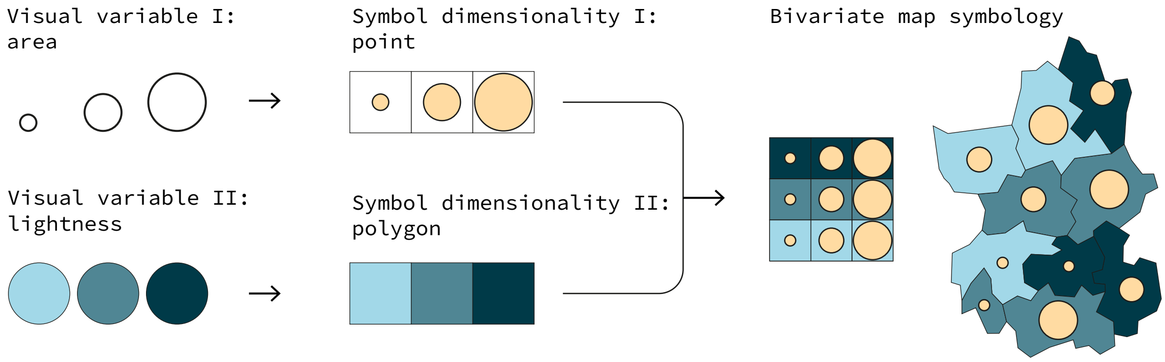


Figure 2.32. Combining visual variables and symbol dimensionality into bivariate map symbols after Elmer (2013).

Elmer (2012, pp. 21–22) shows that this excluding 2.5d and 3d yields three possible combinations of symbol dimensionality in bivariate maps displaying area-based information, illustrated in Figure 2.33:

- **polygon/polygon**: both data dimensions are applied to different visual variables of the same area symbol
- **polygon/point**: one visual variable is varied on an area symbol, another on a superimposed point symbol
- **point/point**: a map using a bivariate glyph where two data dimensions are mapped to different visual variables, e.g., a statistical symbol map.

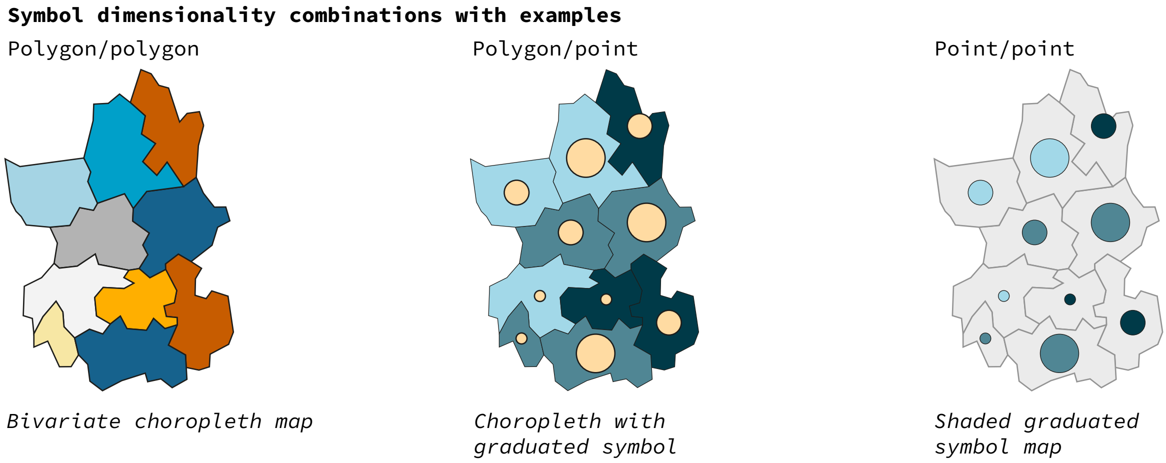


Figure 2.33. Combinations of symbol dimensionality with example bivariate map types, modified from Elmer (2012).

Adding a rendered 3d volume (2.5d) to the repertoire would add the additional possible combinations of polygon volume/polygon volume (e.g. extruded map polygons), polygon/point volume (area symbols with overlaid volume symbols that vary in height) and point volume/point volume (empty base map with volume symbols that vary in two dimensions). Many of these are likely of questionable utility as perceived volume is known to be very hard to judge correctly (Ware, 2013, pp. 168–169), but 2.5d polygon volumes with a second color variable as applied specially to grid maps could in some cases be interesting — such as those demonstrated by Stefan *et al.* (2007)

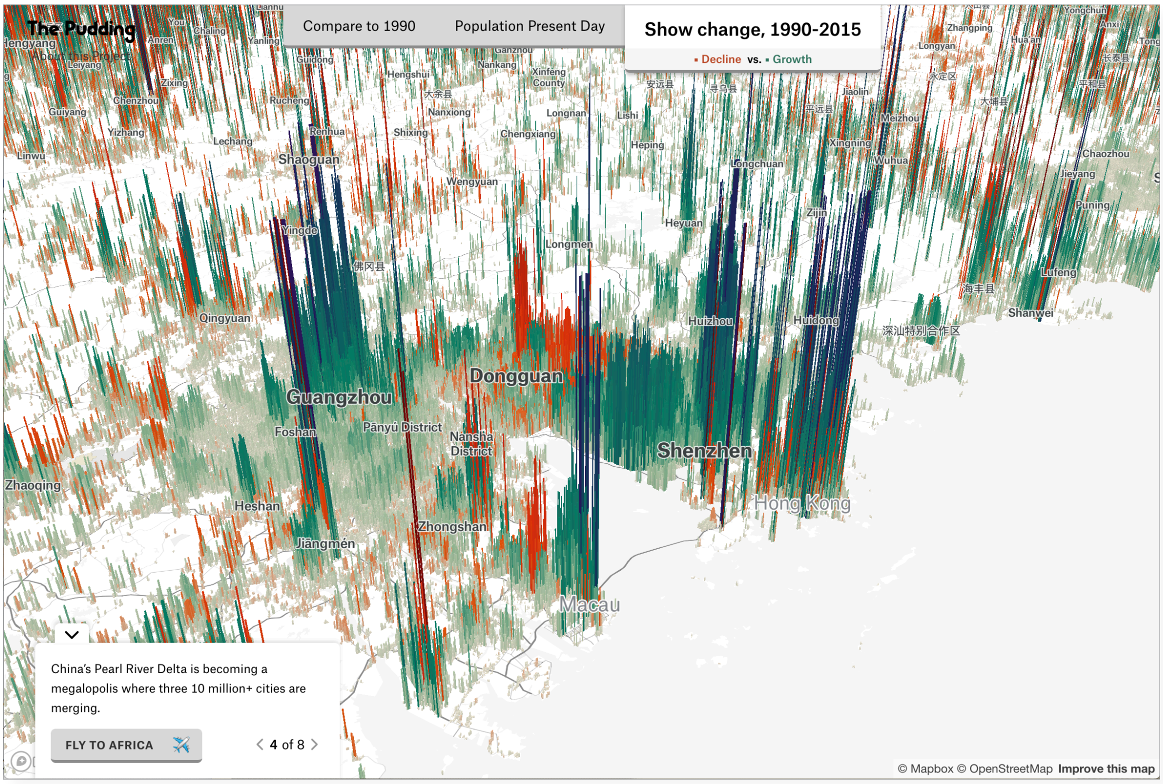


Figure 2.34. Bivariate view in the online project *Human terrain*, where population is mapped to volume symbol height and change magnitude and direction to a diverging color scheme (Daniels, 2018).

or *Human terrain* by Daniels (2018) shown in Figure 2.34[29]. Their potential usefulness is improved by the fact that the perceived dimension for practical purposes is height (the length of "pillars" of uniform width), rather than volume. A similar display based on irregular areal divisions is much harder to interpret.

The variable of transparency poses specific challenges. For transparency to be perceived at all in a static visualization — as opposed to lightness differences or color mixing — there must be some perception of overlapping shapes, but interferences can easily lead to confusion as to which objects belong together (Ware, 2013, p. 211). Interactive maps can avoid this issue, as the toggling of map layers allows transparency to be understood with relative ease. This can be used to study the interaction between symbols on different layers as shown by Luz and Masoodian (2014).

MacEachren (1995) argues that when transparency is used as a visual variable, only three levels can be reliably identified — nearly clear, intermediate, and nearly or entirely opaque. Woodruff (2010) notes that their solution actually does not use MacEachren's transparency variable as such, which is related to overlapping layers and shapes creating an illusion of transparency. They instead represent graded adjustments of lightness and saturation by a data variable mapped to the alpha channel to mix colors of regions with a neutral (black or white) background. As such, the

29 Interactive article: https://pudding.cool/2018/10/city_3d/

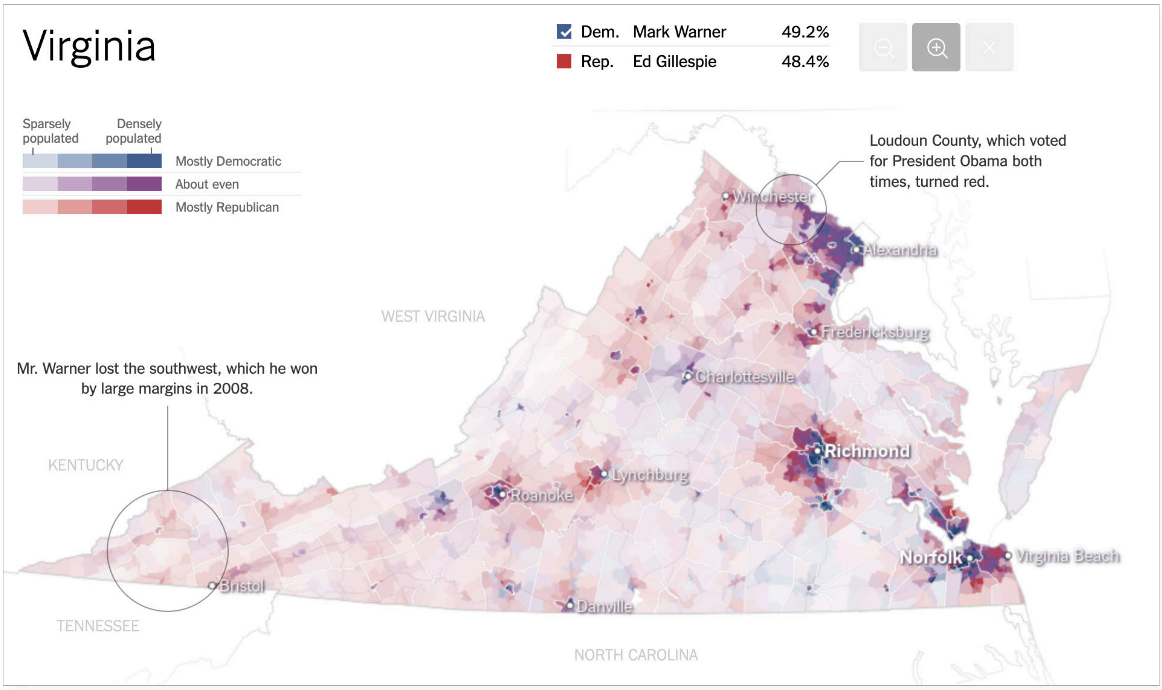


Figure 2.35. Value-by-alpha map of Virginia midterms results. Screenshot from web article (Cox *et al.*, 2014).

alpha channel is used as a convenient proxy. Since Roth et al.'s (2010) Value-By-Alpha (VBA) method actually employs a variation on ordinary lightness-hue-saturation color scales somewhat more steps are available than by MacEachren's (1995) definition. Figure 2.35 shows the method applied to a midterms election map in an article by the New York Times, where the alpha channel is used to divide three color categories into four distinct levels (Cox *et al.*, 2014).

On the other hand, this also means that bivariate symbols that use overlapping layers with data encoded in *different* transparencies become graphically untenable with the VBA method. Value-by-alpha methods can by definition only de-emphazise the visual impact of regions, which is a further limitation (Woodruff, 2010).

2.5 Tasks in map reading

Franconeri *et al.* (2021) note that the complexity of real-world tasks and how they interact with different visualization types poses significant research challenges. The picture is further complicated by variations in design choices and in the visualized data. There is a large number of possible tasks that readers may accomplish with visualizations and the work of creating task taxonomies and and studying these is ongoing. A detailed discussion of the specific literature on tasks in visualization and map reading is beyond the scope of this thesis, but a short aside on some simple tasks is warranted. The visual queries discussed in the section visual channels, p. 20, are somewhat analogous with tasks that people might use to study thematic maps in real-world situations.

According to Elmer (2012) the simplest and most widely used taxonomy of thematic map reading tasks is one that involves two distinct objectives: *identifying* and *comparing*. The identifying task is simply about using the thematic map to get information about a single area, while the comparison task involves collating information across two or more different units on a map.

Bertin (2011) describes a typology of visualization reading tasks with three levels: the *elementary*, the *general*, and the *global* levels. Elementary tasks involve single mapped units (finding the values of a particular unit), general tasks involve groups of units (comparing one region consisting of multiple units with another) while global tasks deal with large visual patterns that occur on the entire map (i.e., seeing a north-south divide in the mapped data).

2.6 Selectivity and effects of symbol combinations

Selectivity affects how a reader can focus their attention on one class of visualized objects while disregarding other, confounding objects. As discussed previously in the chapter on integral and separable visual dimensions (p. 41) the different possible combinations of symbols for bivariate maps can interact in varying ways depending on their design. Roth (2017) classifies these conjunctions as either *homogeneous*, where the same visual variable is used twice to map the different dimensions, or as *heterogenous* if two different variables are used. The employed symbol dimensionalities can also be either homogenous or heterogenous (e.g. point/point or point/area).

Based on their selective characteristics bivariate maps can be classed according to the concept of integral and separable dimensions as belonging to one of the four categories: separate, integral, configural or asymmetrical (Roth, 2017).

2.6.1 Main and emergent visual variables

Bivariate maps should have a matrix of symbols that function as the map legend or key, as illustrated in Figure 2.36. Olson (1981) found that a clear legend and accompanying explanation of how to read the map is particularly important for bivariate maps to be considered understandable. The established form for the legend is a square grid, where the horizontal X and the vertical Y axes show the two main variables (Trumbo, 1981; Eyton, 1984; Dunn, 1989). The cells representing the intersections between them form what Elmer (2013) terms emergent visual variables. These fall on the orthogonal axes that run diagonally across the matrix and are called the Plus (+) and

Minus (-) axes. The selectivity is in direct relation to the map reading tasks: a functional bivariate encoding ought to enable selectively attending to different groups of symbols that occur on the map (i.e., looking for variation on a single variable alone, or the combination of two variables).

Not many empirical studies of bivariate map reading have been conducted. Halliday (1987) investigated how well four different asymmetrical 4×4 bivariate scales applied to a map with 43 regions were interpreted. The scales used — shown reconstructed in Figure 2.37 from the black and white originals — were a hue or spectrum range (red, orange, green, blue), and a lightness range (light green to dark green) versus a black and a white texture. It can be noted that the lightness scale is somewhat uneven and that the Y axis on the hue palette likely ought to be considered qualitative, even though all schemes were used for quantitative data. The experiment was a between-subject study with 120 student participants and all scales were according to Halliday found to be roughly equally effective.

Nelson (2000a) studied individual bivariate symbol combinations were studied with speeded classification tasks. In a follow-up study the Nelson (2000b) bivariate symbolizations grouped by dimensional interactions were tested on maps with real data. This was done with a group of 150 students and found that both rectangle height–width and bivariate choropleth maps using lightness and saturation encoding were effective in showing correlation between the data variables. This was interpreted as supporting that these visualizations indeed are integral.

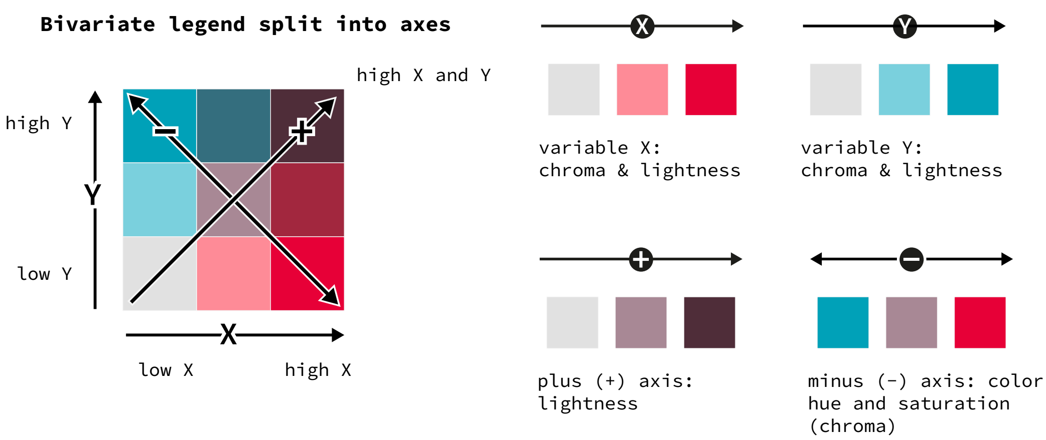


Figure 2.36. Illustration of bivariate choropleth legend decomposed into X/Y and +/- axes (after Elmer, 2013).

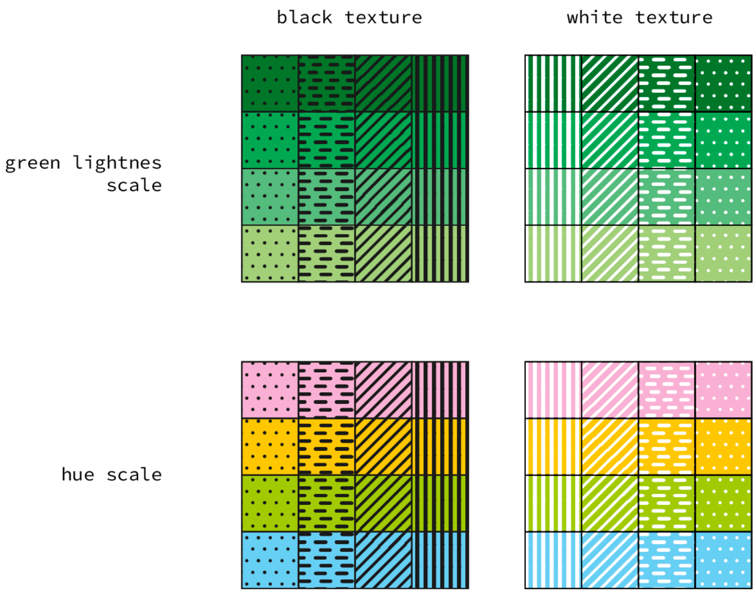


Figure 2.37. Palettes from Halliday's study reconstructed in color from provided CMYK values (1987).

Leonowicz (2006) did an empirical study with 138 university students where a bivariate choropleth map with nine classes was compared to two separate univariate choropleth maps with three classes each. The bivariate map was found to outperform the separate maps in showing the relationships between the data series, but to be clearly less effective in showing the overall spatial distribution of the individual series.

Elmer (2013) conducted an experiment with 8 bivariate map types and visual variable combinations to study selectivity with different symbolizations. This is a continuation of the work of Nelson (2000b) and the types are shown in Figure 2.38:

- Shaded cartogram: separable visual variables (color/area)
- Choropleth w/ graded symbol: separable (color/area)
- Bivariate choropleth: integral data dimensions (lightness/hue and saturation)
- Rectangle map: integral (width/height)
- Shaded texture: asymmetrical (lightness/texture density)
- Value by alpha: asymmetrical (lightness/transparency)
- Spoke glyph: configural (angle/angle)
- Bar chart: configural (area/area)

The different map types were used by Elmer (2013) to visualize to the same fictional data for "chicken consumption and pizza consumption" using a three-step ordinal scale (low, medium, and high) on a map of 36 areas of roughly equal sizes. The 55 participants completed eight questions per map, divided into *elementary* and *general* levels of reading (after Bertin, 2011). Elementary tasks consisted of looking up values of individual units on the map, while general tasks involved questions related groups of mapped units. *Global* tasks that concern the visual distributions visible on the entire map were not studied.

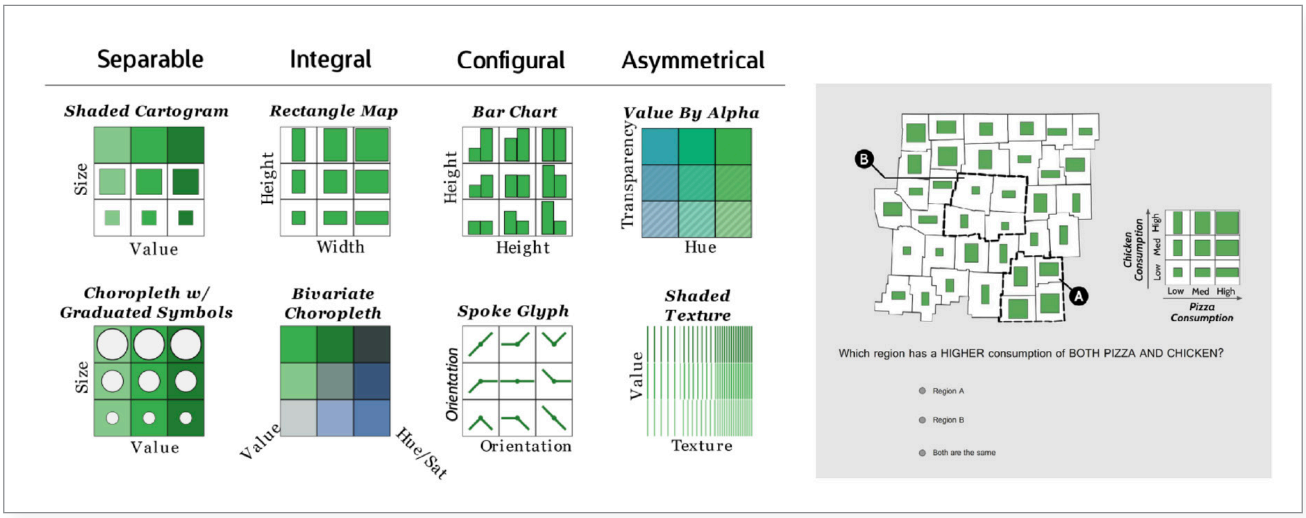


Figure 2.38. Legends of Elmer's (2013) tested bivariate designs; screenshot of test interface showing rectangle map.

Of the studied symbolizations the spoke glyph and the value by alpha maps had the worst performance based on reaction time. The bar chart performed poorly on general-level tasks. Accuracy rates exhibited only small variations with the spoke glyph and shaded texture performing worse than the other symbolizations. (Elmer, 2013)

Elmer (2013) concludes that the results were somewhat unexpected with regards to symbol selectivity and that further research into cognitive strategies for interpreting bivariate maps would be needed to explain the observed effects.

It is unclear how well if at all these results generalize to a situation with a larger number of mapped regions or regions of less uniform shapes. Halliday (1987) suggests that the number of regions may be a deciding factor for what can be interpreted by the reader. Halliday notes that a limitation of the previous research surveyed was the large variation in statistical units displayed — from U.S. census maps with around 1,000 displayed regions to maps with only 9.

It should be noted that the point encodings in particular require the statistical units to be visually fairly large and of uniform size. If the size variation of the mapped areas is large, symbolizations such as the choropleth with graduated symbol or the rectangle map become practically untenable. Franconeri *et al.* (2021) also argue that encodings that use color hues are more powerful for showing groupings than shape-based ones.

Roth (2017) makes a useful breakdown of the four types of available conjunctions of bivariate symbols and their interactions illustrated by one map each and its legend, building on the work of Elmer (2013). These are here reproduced in slightly modified form in Figure 2.39 on the following page. The green-blue color scale used by Roth has been switched to a more distinct gray-pink palette. Arrows on the map legends indicate an interpretation of how easily the different axes can be selectively attended to according to Elmer.

The types are:

- **A: Separable**, represented by a choropleth with graduated symbol,
- **B: Integral**, represented by a choropleth with graduated symbol,
- **C: Configural**, represented by a split graduated symbol map, and
- **D: Asymmetrical**, a value-by-alpha map.

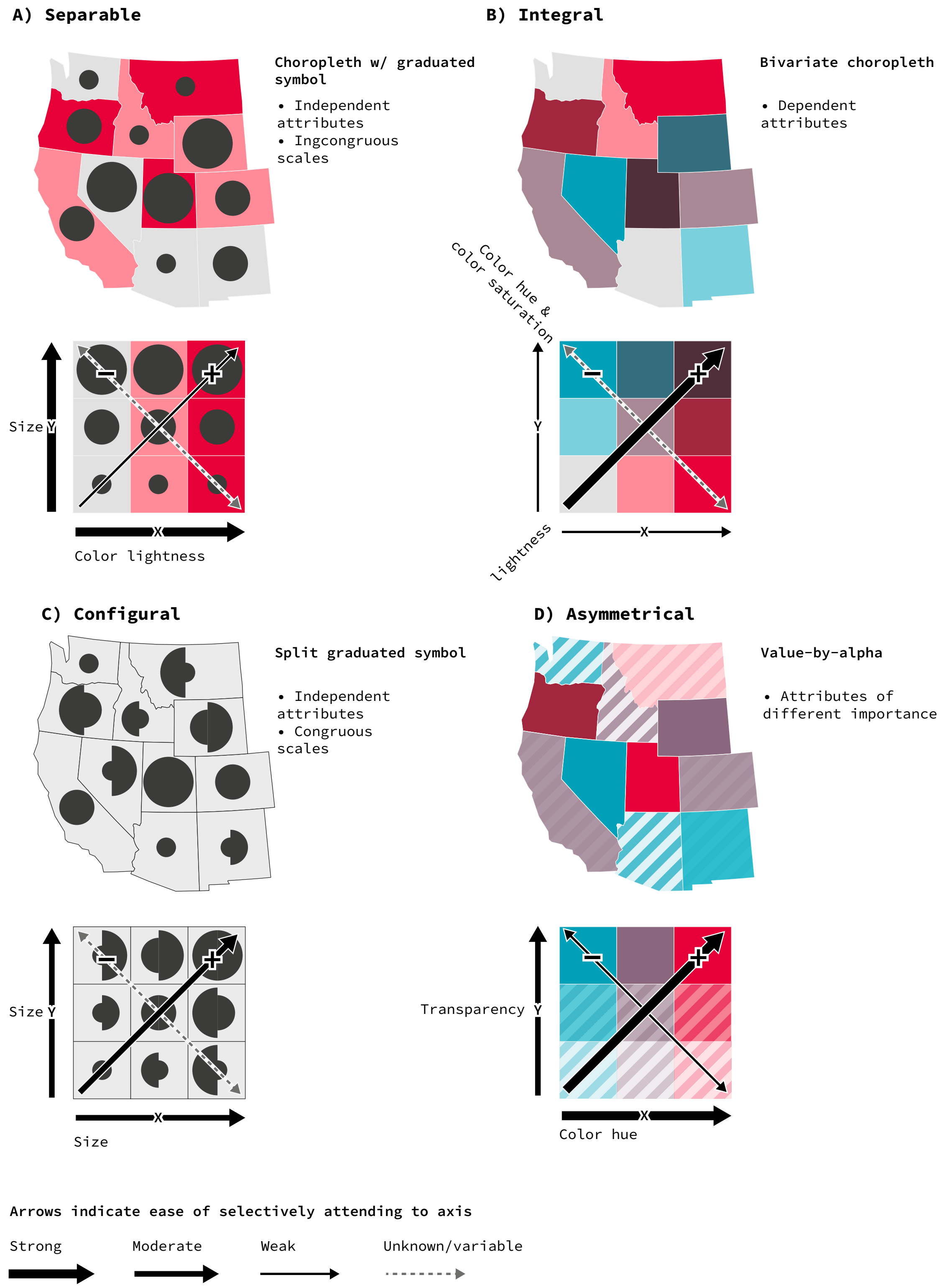


Figure 2.39. Conjunctions of bivariate symbols after Roth (2017).

A map with **separable** dimensions like Figure 2.39 **A** — a choropleth with overlaid graduated symbols dimensions — supports studying the distributions of either variable in separation, as these are single-feature searches (e.g. find large circles or find light areas). Studying the conjunctions (+ and - axes in the illustration) can presumed to be perceptually demanding as this requires conjunction searches, like finding a large circle on a light background. Elmer (2013) notes that this type of display is suited for independent attributes that are not assumed to be in correlation and use different, incongruous scales, different classifications or so forth.

By the same logic, shaded proportional symbol maps and shaded cartograms are both separable despite using homogenous symbol dimensionalities, since the visual variables (shape and color) are separable (Roth, 2017).

With an **integral** conjunction like Figure 2.39 **B** (bivariate choropleth) Elmer (2013) posits that attending to the emergent Plus (+) visual dimension is effortless. This is due to the color scheme having decreased lightness along the Plus axis — seeing where X and Y values are high or low together is therefore easy. The original X and Y dimensions are on the other hand harder to interpret in separation. Roth (2017) notes that looking for regions that have the same value for X while ignoring differences in Y values is difficult — it is for instance hard to focus on all regions that are in the middle X column. It does however seem that color scale choice affects this. The palette used in Roth's example has fairly small hue differences. Using a more distinct palette can arguably make the dimensions easier to separate — in Figure 2.40 the high X values appear to form a clear group of three colors from saturated red for high X - low Y to a very dark purple hue where both X and Y are high.

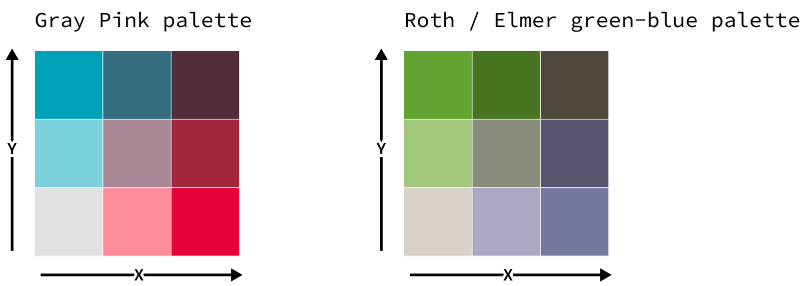


Figure 2.40. Comparison of gray pink palette used in the redrawn maps in Figure 2.39 with the green-blue palette in the examples by Roth (2017).

Elmer (2013) and Trumbo (1981) recommend bivariate choropleth maps specifically for studying the correlation of two variables. The emergent Minus (-) axis that typically is represented by a difference in hue is assumed to be of unknown or variable selectivity. Contrary to the interpretation of Roth (2017) and Elmer (2013) the Minus axis can in some cases have fairly good separation with a suitable color palette — it is essentially a bidirectional scale with a neutral middle point. Strode *et al.* (2020) argues that the bivariate choropleth, when designed to use complementary colors on the X and Y axes (what they term the diagonal model after Trumbo, 1981) also effectively highlights regions where either series has large values in isolation. Therefore, it can be effective also in cases where the the two mapped series mostly are not correlated. Stevens (2015) offers a similar interpretation (see Figure 2.51). This gives three apparent visual categories: correlated data is shown as a lightness sequence on the + axis, while the Y and X axes show non-correlated data in two complementary color and lightness gradients. Hence, the type can also be used for explorative purposes to visually determine the absence of correlation on a map.

Configural combinations like Figure 2.39 **C** (split graduated symbol) have a clear emergent (+) dimension, but the X and Y dimensions remain more easily separable than in an integral display, as a viewer should be able to separately focus on either half of the symbol. In a configural combination the data for X and Y should share a common scale. When the series are in agreement this is apparent from the symmetrical symbol that emerges on the + axis. (Elmer, 2013)

Empirical studies would be needed to determine this in more detail, but it is reasonable to assume that shape-based encodings can lead to displays where conjunction searches become increasingly difficult as the number of mapped regions increases — it is hard to attend to just large circles on a dark background. In the case of split graduated symbols some combinations (large circles, both axes in agreement) may be easier to find than others. It might be possible to combine other visual variables to avoid conjunction searches, as Nusrat *et al.* (2018) demonstrate in their cartograms (see Figure 2.30).

While looking up values of individual sides of symbols work for elementary tasks involving single areas, the configural combination used in Elmer's (2013) experimental study (a pair of bar charts) exhibited long reaction times for general-level tasks. This appears to give some reason to suspect that searching for one part of a configural combination requires a conjunction search that takes a lot of effort and focused attention — it may for instance not be easy to selectively attend just to symbols where the left side is small (Figure 2.39, A). It is even possible, that shapes encoding opposite values could be grouped together due to their visual similarity (Figure 2.39, B).

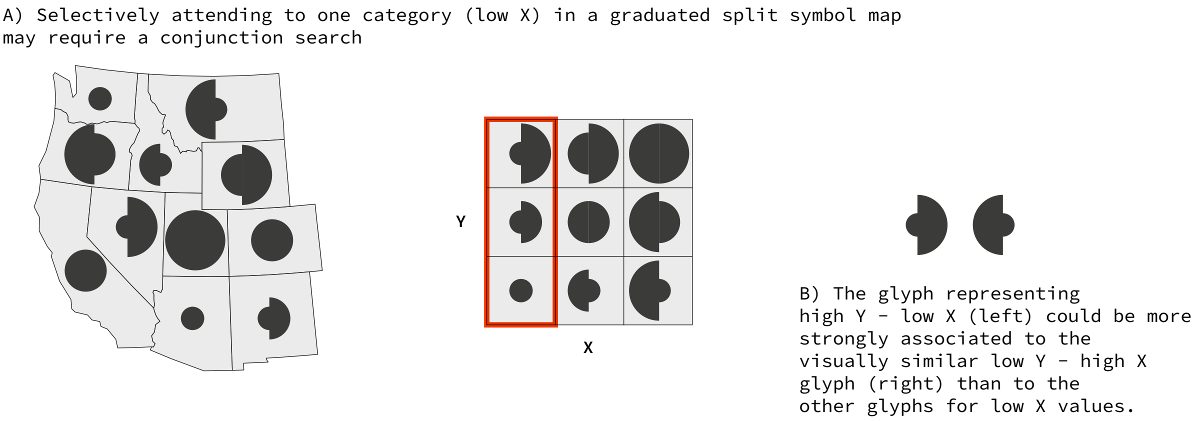


Figure 2.39. Some possible issues with configural bivariate symbols.

An **asymmetrical** combination like Figure 2.39 **D** (Value-by-alpha) basically creates a situation where one channel (Y) inhibits the other (X) by making it stand out less for certain combinations. In the example regions with low Y values are visually de-emphasized. This could be useful for adjusting the appearance of a variable of interest by a secondary variable like population as proposed by Roth et al (2010). Elmer's (2013) examples include a complementary graduated pattern variable to reinforce the effect. Another example of an asymmetrical combination is using size for one dimension and color for the other.

In addition to the four examples discussed in detail here there are many more possible types of bivariate maps that visualize numerical information aggregated to map regions. Elmer (2013) created a catalogue of possible bivariate combinations, illustrated with clarifying images. This categorization is basically an extension of the classification in Figure 2.24 that extends it with the three dimensionality combinations — polygon/polygon, point/polygon and point/point. Based on the assessments of MacEachren (1995) some variable combinations — like orientation applied to both data series for polygonal symbols — were deemed "non-functional or graphically untenable" and thus left out. This results in a table with a total of 42 possible combinations of bivariate symbolizations.

I have chosen to leave asymmetrical point/polygon features outside the scope of further discussion in this thesis. While point/point symbolization — such as split graduated symbols — may conceivably be effective in some cases, they pose challenges to map composition and can be hard to place if the mapped areas vary significantly in size.

Of these 42 combinations, 13 possible bivariate map types belong to the polygon/polygon category in Elmer's figure (2013). These are listed here in Table 3. Bolded names refer to bivariate map symbolization that according to Elmer are in common use. Empty cells represent non-applicable combinations. Orientation has been removed from the original table.

Polygon/polygon features	*Size*	*Color (hue, saturation, lightness)*	*Transparency*	*Fill size*	*Fill density*
Size					
Color (hue, saturation, lightness)	**Shaded cartogram**	**Bivariate choropleth**			
Transparency	Value-By-Alpha cartogram	**Value-By-Alpha choropleth**			
Fill size	Cartogram w/ texture	Shaded texture	VBA w/ texture	Bivariate texture	
Fill density	Cartogram w/ dot density	Shaded dot density	VBA dot density	Graduated dot density	Multiseries dot density

Table 3. Visual variable combinations of polygon/polygon solutions for numerical data aggregated to polygonal features.

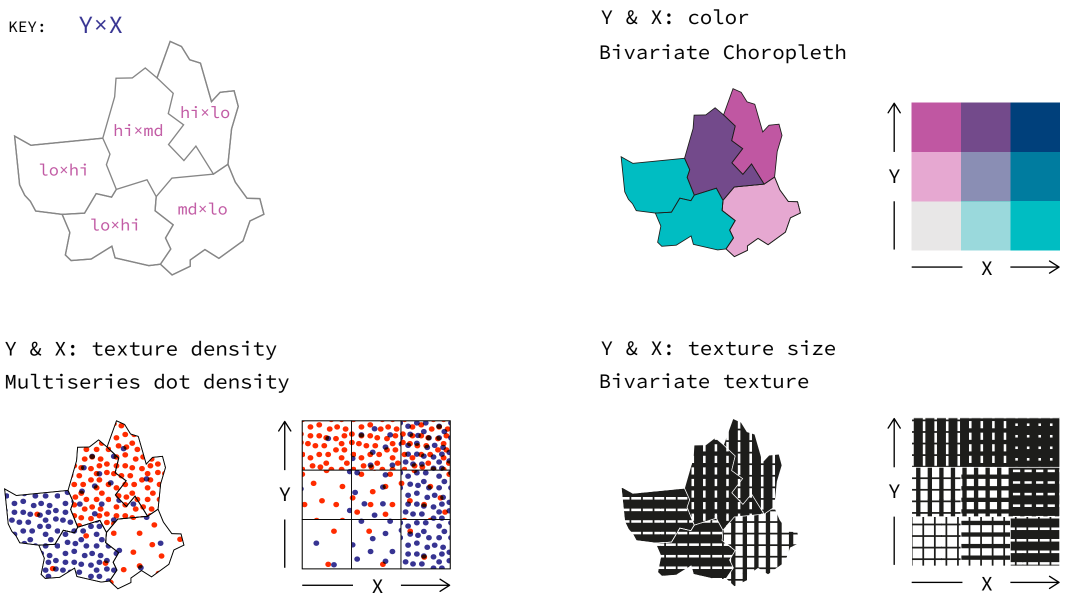


Figure 2.41. The three homogenous bivariate symbolizations of map areas after Elmer (2013).

To further narrow the focus I concentrate on homogenous combinations, where the same visual variables are applied to the area of the mapped areas on both the X and the Y axis. These are all integral combinations; in that they focus on the Plus axis and are suited to represent phenomena that are assumed to have a positive correlation (more X means more Y). This leaves only three combinations applied to the map features themselves, illustrated in Figure 2.41. The bivariate choropleth map, the bivariate texture map, and the multiseries dot density map. Of these three, the choropleth appears least ambiguous, while the multiseries dot density maps seem to pose such significant design challenges as to make them largely inpractical.

The bivariate texture map — while less problematic — is very sensitive to the texture designs used, in addition to the scale of presentation as discussed previously. This means that of these three alternatives the bivariate choropleth likely makes it easiest to design a functional bivariate map that achieves effective separation of the classes. The following section will discuss bivariate color scales in more detail.

2.6.2 Bivariate color scales for choropleth maps

The history of bivariate choropleth maps goes back at least to the 19th century and the work of Georg von Mayr[30] (Figure 2.42). Mayr's map uses texures and is thus not strictly a choropleth — Strode *et al.* (2020) claim that the type was invented for the U.S. Census maps as late as 1974, but this cannot be taken as conclusive. While guidelines for color use in maps generally has been limited to using hue for visualizing qualitative distinctions and

30 Friendly and Denis (2001): 1874. Georg von Mayr.

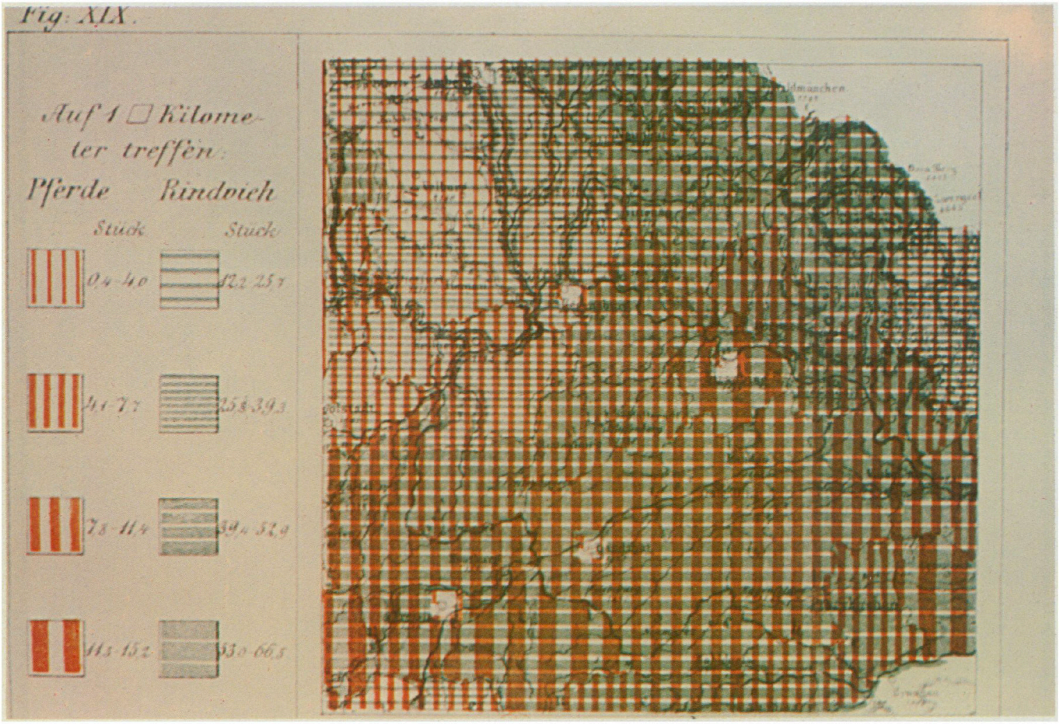


Figure 2.42. 1874 bivariate map by Georg von Mayr as reproduced in Wainer and Francolini (1980): *A Two-Variable Color Map Showing the Joint Distribution of Horses (Pferde) and Cattle (Rindvich) in Eastern Bavaria Done According to Scheme 1.*

lightness for quantitative values (Brewer 1994, p. 123), some theoretical and practical guidelines for multivariate choropleth maps have been suggested by different authors such as Trumbo (1981), Eyton (1984), (Brewer 1994) and more recently Stevens (2015). Multivariate color schemes may be created by mixing univariate color scales or using transparency in addition to hue and lightness (e.g., Gao, Li and Qin, 2019), and thereby making it possible to create multivariate choropleth maps. This is effectively limited to two or perhaps three variables — *bivariate* and *trivariate* choropleth maps (Tyner, 2010, pp. 183–185).

Brewer (1994) provides a systematic account of the use of color schemes based on previously published literature on choropleth maps and other displays where a data dimension is visualized using colored areas. The use of transparency and combining color with pattern are not discussed by Brewer. Brewer states that her proposals are based on literature, cartographic conventions and experience and that they are "not yet thoroughly tested."

- **One-variable color schemes**, including the following four basic types of color scheme (see Figure 2.25.):
  - qualitative
  - binary
  - sequential
  - diverging

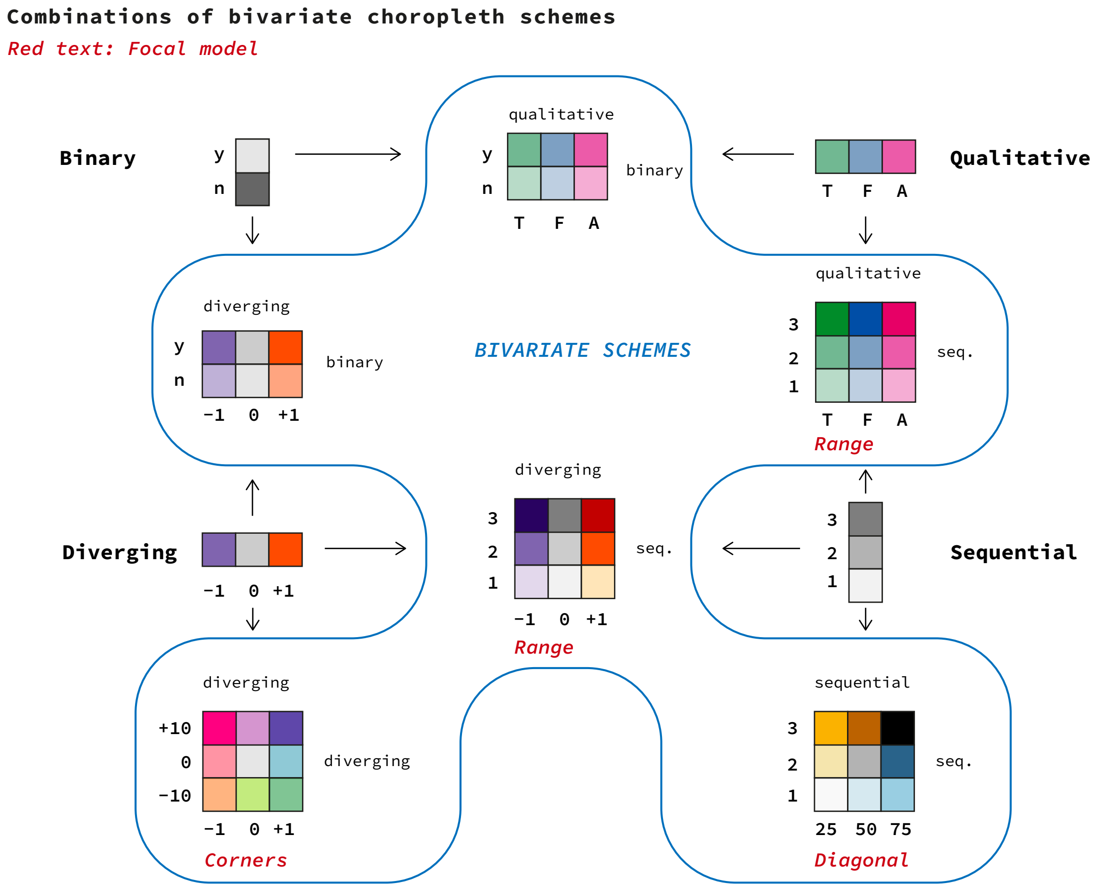


Figure 2.43. A diagram based on Brewer's (1994) categorization of possible bivariate choropleth color schemes and how they interrelate. Schemes have been labeled with their focal models from Strode *et al.* (2020).

- **Two-variable color schemes** comprising the following combination schemes, illustrated in Figure 2.43:
  - qualitative/binary
  - qualitative/sequential
  - sequential/sequential (with balance scheme as a special case)
  - diverging/sequential
  - diverging/diverging
  - diverging/binary

Since I deal with the representation of quantitative maps, this thesis is concerned in particular with the sequential/sequential scheme.

In *Information Visualization* Colin Ware (2013, pp. 134–135) briefly discusses bivariate color sequences and considers them "notoriously difficult to read." As the main underlying issue Ware identifies the difficulty of reading color dimensions like lightness and hue in a separable way. Ware refers to empirical research carried out by Wainer and Francolini (1980) on the readability of a particular bivariate choropleth design for U.S. census data using a color scheme employed by the U.S. census bureau. One of the evaluated maps is illustrated in Figure 2.44. In this study, Wainer and Francolini looked at how well a group of 16 test participants were able to read bivariate maps on what Bertin (2011) terms the *elementary level*. This involved reading quantitative values for particular areas using the map colors — answering questions like “what is the median family income of this statistical unit?” The exact questions used were variations on “what is happening at this place” and the performance of reading a bivariate map was compared with reading two univariate maps (Wainer and Francolini, 1980, p. 84).

Interestingly, they state that for determining locations for a bivariate event, i.e., a combination of two events “... the superiority of the bivariate maps to the two univariate ones in this task seems so obvious that no further testing is required.”

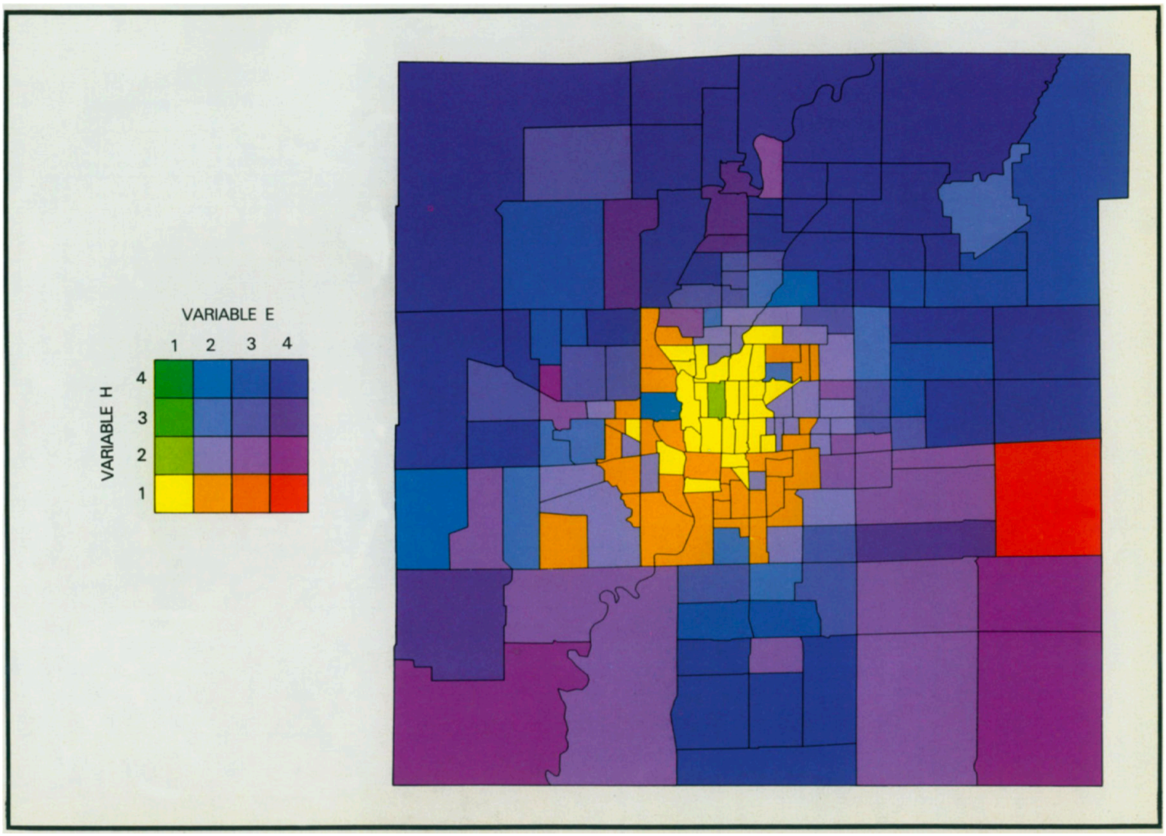


Figure 2.44. The map that was evaluated in the paper by Wainer and Francolini (1980): *H. Two-Variable Color Map Crossing Variables H and E (U.S. Bureau of the Census).*

The performance of the test participants in interpreting the bivariate maps was mostly poor and based on this the authors conclude that there is little practical use for bivariate maps. However, this cannot be seen as conclusive, since the experiments were conducted using only a single color scheme which furthermore had four steps yielding 16 color combinations. The color classes were constructed by overlaying two univariate color schemes.

Wainer and Francolini (1980) argue that a possible use for these maps would be identifying aggregations of regions that belong together (e.g., “these are all blue”) but deem this use case to be limited. They appear to overlook their own conclusion that a bivariate map can be clearly superior to two separate maps in answering questions concerning bivariate events. They are on the other hand more positive with regards to a scheme combining texture and color, as in Mayr’s early example from 1874 (1980, pp. 88, 92, illustrated in Figure 2.42).

Olson (1981) did a study with multiple experiments of the US census maps that came to more positive conclusions about their utility, provided that the test participants were instructed how to read the maps and how the legend works. In the open-text answers the subjects were able to describe aspects on the map, and most did not find them too complex to read. This is notable, as the 4×4 color scales in the maps can be considered less than ideal. An interesting suggestion by Olson is that bivariate choropleth maps should be accompanied by univariate maps of both distributions in separation. An example of this is shown in the map by Leonowicz (2006) in Figure 2.53. Olson's study also involved a test map with a similar legend, where the bin divisions are illustrated.

As alternatives to bivariate scales where both dimensions are encoded with color, (Ware, 2013, p. 134) proposes using dimensions that are easier to separate perceptually from color, such as visual texture or height difference, for the second variable. Ware is skeptical of the the possibility to design generally applicable bivariate color scales that display quantities effectively and without distortion. Halliday's study (1987) discussed previously lends some support for the notion that texture can be effective.

Stevens (2015) notes that creating a bivariate choropleth map should rest on the assumption that there is some meaningful connection between the mapped variables. Eyton (1984) indeed directly equates bivariate maps with scatterplots that show the correlation between two variables. An important additional observation by Slocum (2014, p. 252) is that bivariate choropleth maps are most effective when the geographical distribution of the mapped data correlates with the bivariate data values, so that similar areas can be seen forming clear regional groups. If there is little connection between locations and data values, the map looks speckled and chaotic, especially when the mapped statistical units are small in size.

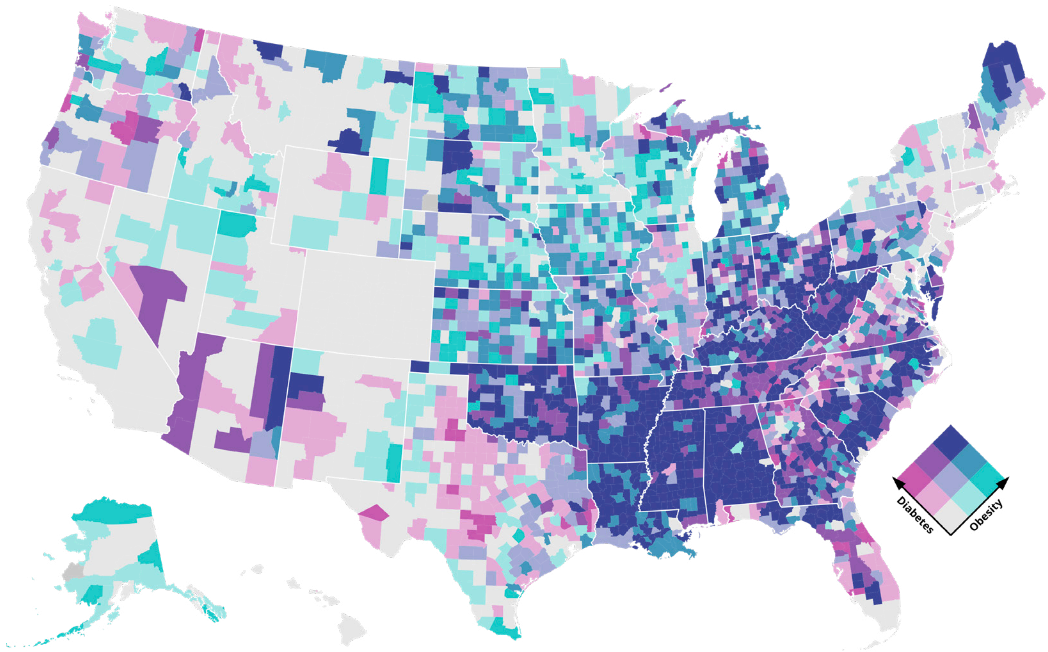


Figure 2.45. Bivariate choropleth map of diabetes and obesity levels by county in the United States by Bostock (2019).

The bivariate map of diabetes compared to obesity in the United States in Figure 2.45. shows aspects of both these conditions — the states south of the Great Lakes are somewhat checkered in appearance, while the variation is smoother around the South states between Texas and Florida.

2.6.2.1 *Designing bivariate choropleth maps for different use cases: focal models*

For map publication and design Strode *et al.* (2020) extend on the work of Trumbo (1981) and provide three categories for bivariate choropleth schemes that they term *focal models*, illustrated in Figure 2.46 — **Corners**, **Range**, and **Diagonal**.

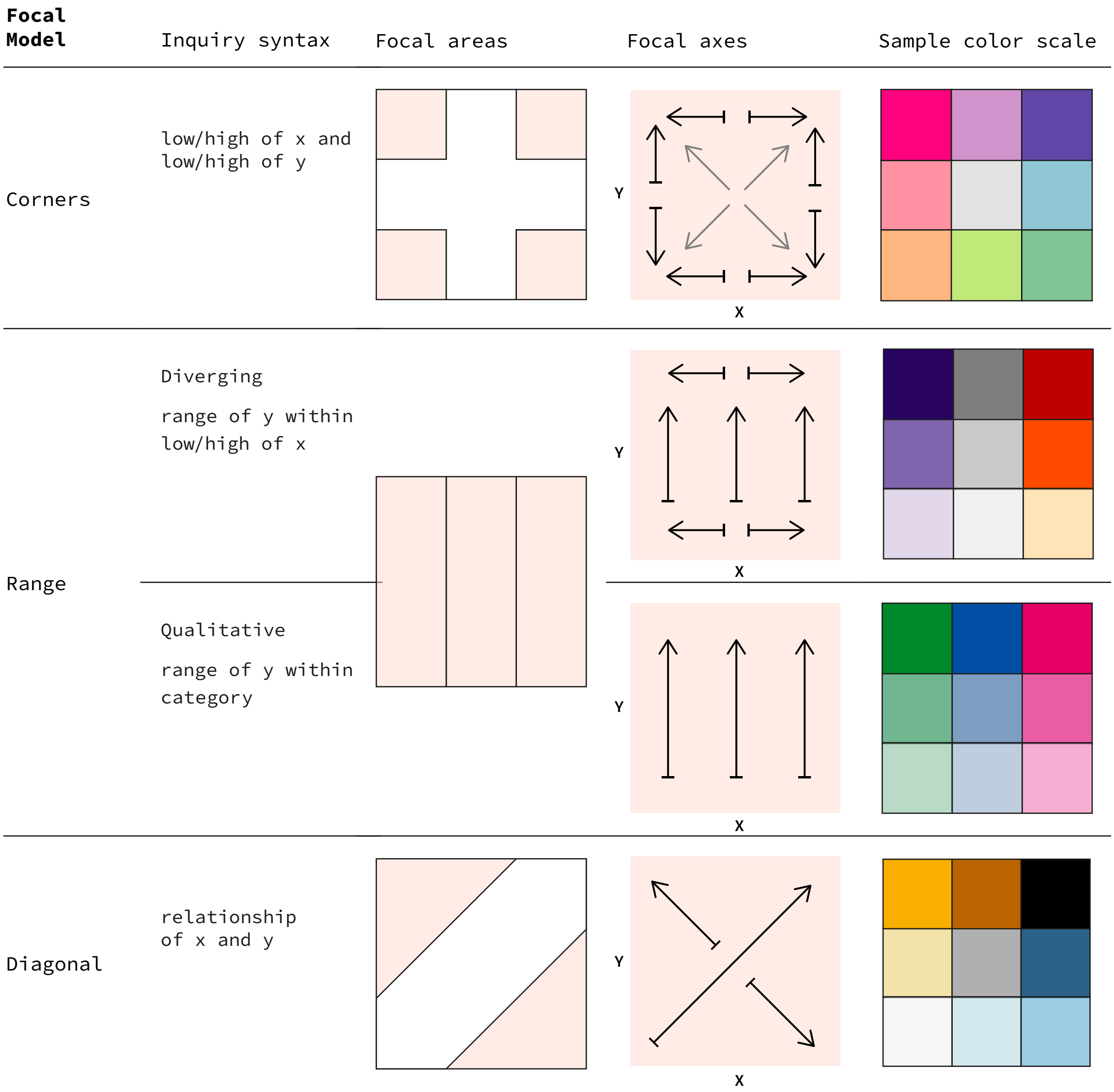


Figure 2.46. Summary of focal models with their attributes after Strode *et al.* (2020). Colors scales have been replaced with the 3×3 examples used by Brewer (1994). Strode *et al.*'s diverging palette does not use lightness change along the X axis.

These define palette arrangements specifically for bivariate choropleth maps that can be chosen depending on the data used and what relationships the map designer want to focus on. For consistency I have redesigned Figure 2.46 so that each scale uses only 9 categories and the colors of Brewer (1994) illustrated in Figure 2.43. Each focal model has an example inquiry syntax (the type of questions about the data it supports), focal areas that are visually prominent and focal axes along which variation in the data is mainly appraised. The focal axes are similar to the emergent dimensions in the typology by Roth (2017).

The **Corners** focal model is essentially a diverging/diverging color scale where the main interest is in the four extreme values around a neutral middle point.

The **Range** focal model includes two subcategories — diverging and qualitative. The Diverging category is an asymmetrical configuration that looks at different ranges for a variable Y, where X varies between two extremes. This is similar to the value-by-alpha map shown in Figure 2.35 — although the axis there is turned 90°, so that the modifying variable is on the X axis. The Qualitative category simply compares ranges of Y values within separate categories that do not have a numerical relation on the X axis.

The **Diagonal** focal model is essentially the base case of the bivariate choropleth map that shows an integral conjunction between two data series with a strong focus on the Plus (+) axis. Trumbo (1981) describes the requirement that Strode *et al.* (2020) subsequently has formalized as the diagonal focal mode. "If display of positive association is a goal, scheme elements should resolve themselves visually into three classes: those on or near the principal diagonal, those above it, and those below." Trumbo states that a minimum requirement is that the data on both axes is ordinal, which means that the color scheme should be visually ordered as well. Because the diagonal focal model arguably is the most established and generally applicable of the three it will be the focus of this thesis. It also offers somewhat more flexibility in the design of color scales than the range and especially the corners models, which makes it an interesting subject for experimentation.

2.6.3 Practical guidelines for the design of sequential bivariate color schemes

A diagonal focal model or sequential/sequential map essentially combines the classes of two separate univariate maps that both have the same N number of classes. The resulting bivariate map does therefore have N^2 classes that need to be identified by color. Figure 2.47. shows an example map created with the R package `biscale` (Prener, 2022), and using the "DkBlue" color scheme by Stevens (2015). (No additional styling applied; note that the category low emissions×medium value added does not occur on the map).

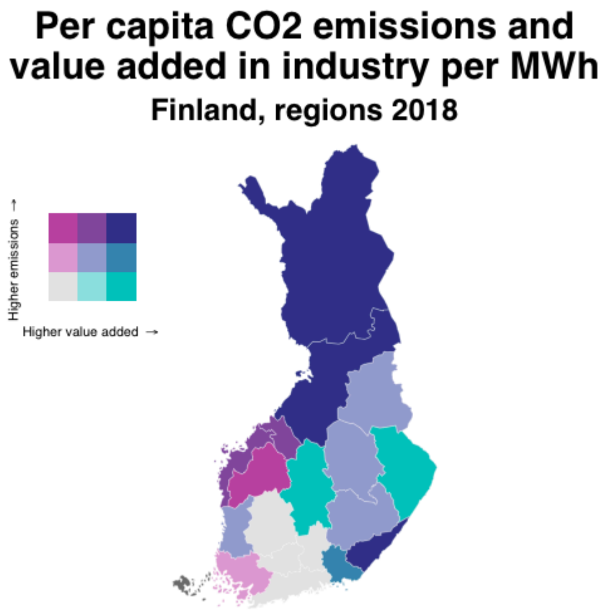


Figure 2.47. Example of sequential/sequential or diagonal focal model bivariate map with 3×3 classes.

The number of different colors in a palette that reliably can be distinguished under different conditions is around 12 according to Arnkil (2013, p. 143). From this, it follows that the effective number of unique and distinct hues is smaller for sequential color scales. Bertin (2011, p. 325) states that "[c]olor variation alone yields only about six selective degrees. Beyond that, one must use schemes involving monochrome figurations."

Because colored regions in a choropleth map often are of a wide range of sizes, depending on the mapped areas, this can make small areas hard to reliably distinguish (Dorling-like cartograms face a similar issue depending on the data values). One might use the approach Szafir (2018) suggests and algorithmically boost contrast if small regions occur, but this can create visual inconsistencies between maps that are shown together. A more robust alternative would appear to be that color scales for use in bivariate choropleth maps (and other visualizations where the colored areas vary in size) are designed with colors that to begin with are different enough that the smallest areas reliably can be told apart.

Another compounding issue is that any color scale is difficult to design so that sufficient contrast between adjacent colors *and* between the lightest color and the background is achieved, as discussed on p. 36 in the section Issues with color contrast in visualization design.

According to conventional recommendations (e.g., Koponen and Hildén, 2019, p. 77) the number of easily discernible color steps in a sequential scale is 5–7 for visualizations such as choropleth maps where any color in the scale may appear next to any other. Tyner (2010, p. 186) removes Eyton's (1984) recommendation on the number of categories from a list of guidelines for bivariate choropleth maps but goes on to state that a map with a 3×3 legend is easier to comprehend than one with a 4×4 or 5×5 legend. In contrast, Strode *et al.* (2020) shows scales with up to 4×4 classes, while Stevens (2015) maintains that 3×3 steps is the practical maximum. Visualizations of continuous variables, where only mostly related colors appear together may allow for as many as 20 different colors, since they do not need to be identified in isolation. However, this is unlikely to be the case in bivariate maps. The 25 colors of a 5×5 legend is obviously outside of any conceivable scope of reliably separable colors.

An additional limit is placed in the case of bivariate schemes by the requirement that the colors should be visually related (Eyton, 1984). A sequential/sequential scheme is constructed from the intersection of two lightness schemes based on different hues — high values in both correspond to dark colors and low values to light colors. This puts rather clear limits on the number of sensible combinations. (Trumbo, 1981) states that good schemes for a diagonal models will have a neutral diagonal, like the combination orange and cyan. Brewer (1994, p. 141) agrees in recommending a pairing of schemes that are approximately but not precisely complementary such as blue and orange-yellow to produce neutral diagonals. In the case of two subtractive primaries the mixing will result in new combination hues. A common bivariate scheme for sequential/sequential data uses magenta and cyan sequences that combine to create purple hues. Figure 2.48 shows four such color schemes created by Stevens (2015).

Figure 2.48. Examples of bivariate schemes with hex color codes employing color mixing after Stevens (2015).

Trumbo (1981) provides recommendations for color scale designs. However, while the underlying principles, like selecting colors by sections through a uniform color space, are relevant in theory they are in practice fairly difficult to apply, for instance due to referring to color models that are no longer in active use. Strode *et al.* (2020) do not provide a detailed discussion of color scale design in their paper but quote personal communication with Trumbo where he recalls encountering issues in actually testing and creating the color schemes with the computer systems available at the time.

Similarly, Robertson and O'Callaghan (1986) describe an approach using the uniform CIELAB color space to generate bivariate color scales following Trumbo's example. They note limitations in the uniformity of the colors created but argue that this in practice might not matter — an eventual lack of uniformity between colors could well be more than offset by the effect of induction (i.e., simultaneous contrast). Interference of adjacent colors on the map creates perceived differences that are larger than the flaws in uniformity. They argue that for this reason the colors used in a map do not have to be strictly uniformly spaced, or even the color space used for color scale design to be exactly uniform. In addition, they also conclude that four classes per series is a practical upper limit for bivariate maps.

The blog article “Bivariate Choropleth Maps: A How-to Guide” by Stevens (2015) is a rare recent example of a detailed explanation of how to create bivariate color schemes. Stevens suggests starting from two separate sequential color schemes that begin with a light, neutral hue for lows and a darker, saturated hue for highs. The “high” hues should be roughly but not exactly complementary. Overlaying the two color schemes in design software and using either “darken” or “multiply” blending modes provides the additional color combinations as illustrated in Figure 2.49. Stevens further notes that the resulting intersecting colors can benefit from manual adjustment: the color representing high values for both variables should be increased in saturation while the middle color should be brought closer to the high-by-high or 3-by-3 color.

Dark Blue palette, component color scales

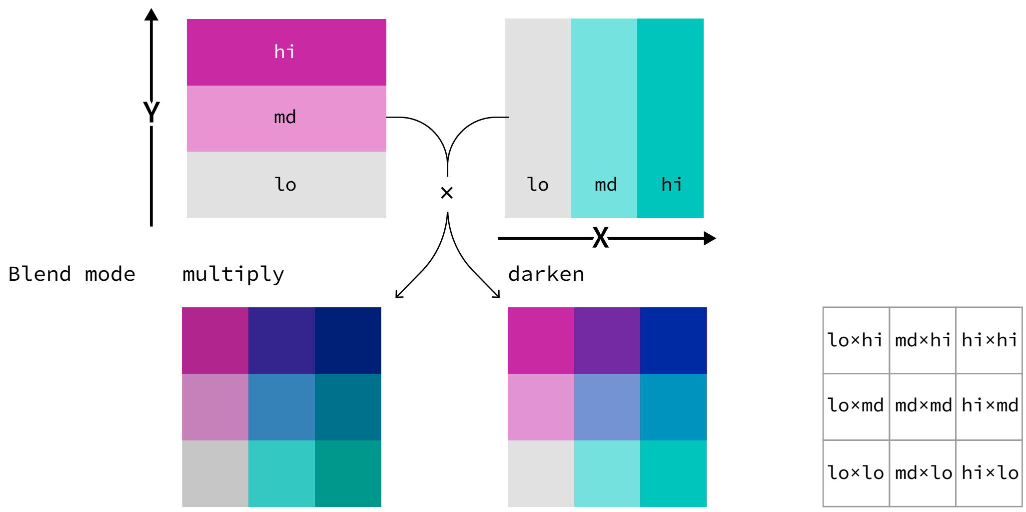


Figure 2.49. Variations of the Dark Blue bivariate scheme recreated in Adobe Illustrator™ using Stevens’ method by overlaying two color scales and blending them with either multiply or darken, without additional manual adjustments.

Blend modes are mathematical transformations that use the RGB color channels. Technical documents like *Compositing and Blending Level 2* (CSS Working Group, 2022) deal with blending between *source* and *destination* images that produce a *result*. Since only blending of individual colors is of interest here, I will instead refer to two source colors (1 and 2) that by blending produce a result color (1×2). Of the commonly available blending modes, only multiply and darken tend to result in practically viable results when mixing color scales. Multiply can be likened to the effect of overlaying two colored films. Darken results in a color that appears approximately intermediate, if the blended colors are complementary and differ in lightness. Figure 2.50. illustrates the Darken and Multiply blending modes and their respective functions for calculating the blended color values with the same pair of source colors.

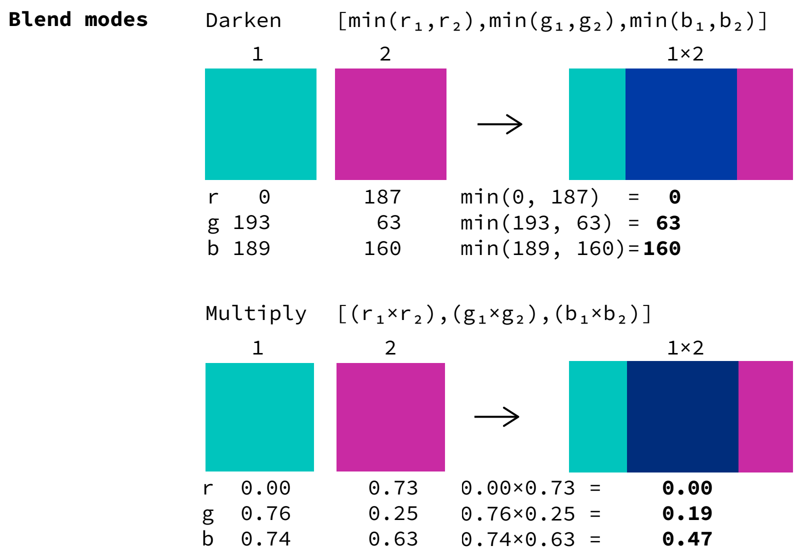


Figure 2.50. Darken and multiply blend modes and their corresponding functions as applied to combinations of two colors.

Multiply as applied to two colors takes the values for each RGB color channel for color 1 [0…1.0] and multiplies it with the corresponding values from color 2. Darken compares each channel of the source colors by value and takes the darker (smaller) RGB value for each channel to use in the resulting color. If neither color is white or black multiply always results in a color that is at least as dark as the source colors. Both modes are commutative in that the blended result color is not affected by the order of the source colors. (CSS Working Group, 2022)

In contrast with Elmer's (2013) and Roth's (2017) division into main and emergent visual variables Stevens' conceptualization of the components of a bivariate color scale emphasizes that the "edges" of the legend are important, as illustrated in Figure 2.51. Stevens' "agreement" is referred to as the Plus (+) axis in this thesis. The categories that reflect mainly either of the two variables are more visually salient than intermediate combinations. This is similar to the conceptualization of the diagonal focal model by Strode *et al.* (2020), but without their focus on the opposite or Minus - diagonal between high Y, low X and low Y, high X values.

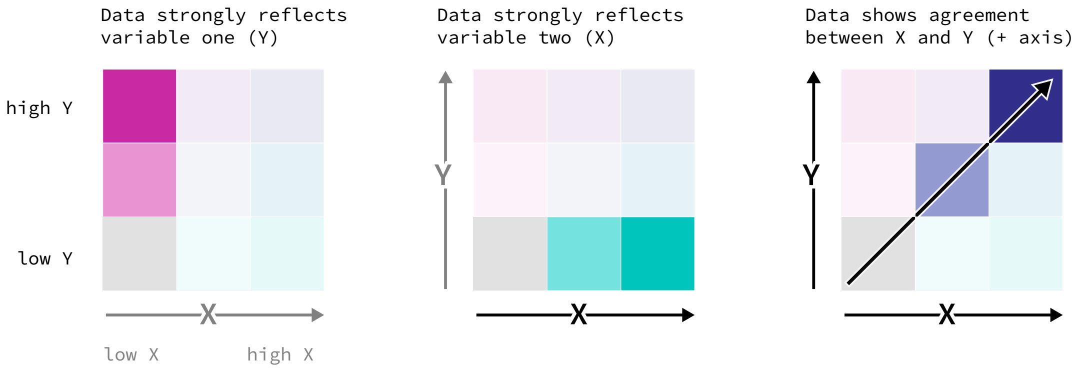


Figure 2.51. Data relations in a bivariate color scale after Stevens (2015).

The sources surveyed in this thesis do not explicitly refer to how much the lightness should change on the Plus or agreement axis, nor exactly how the colors should interrelate in lightness, except that lightness should increase for higher values. Strode *et al.* (2020) state that "correlated data appear in a grayscale sequence, while non-correlated data are shown using gradients of complementary colors". It is reasonable to apply the general recommendations of uniform lightness changes in quantitative color scales (see e.g., Ware, 2013, p. 131; Liu and Heer, 2018) to bivariate choropleth palettes as well, at least in the diagonal focal mode as defined by Strode et al. (2020). Using Stevens' design principles does result in at least a a mostly uniform decrease in lightness from the low×low color along the diagonal towards the highest 3×3 value, which can be emphasized by turning the palettes 45° so that 1×1 is on the bottom. This is illustrated in Figure 2.52. Note uneven variations in lightness in the Brown palette (left) compared to the Dark Violet (right) palette, where the lightness change is more equal from low to high and across the diagonal.

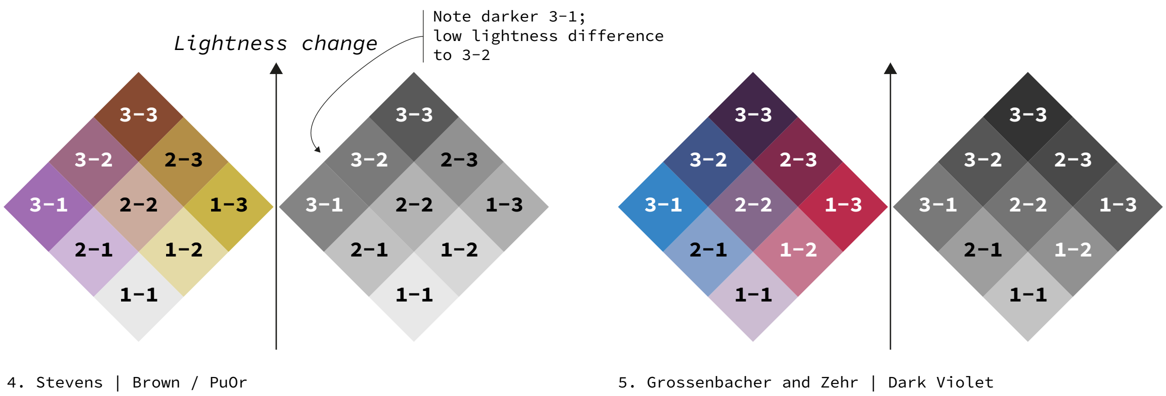


Figure 2.52. Bivariate palettes rendered in original colors and greyscale. The lightness change on the Plus axis in the Dark Violet (right) palette is more uniform.

Following Robertson and O'Callaghan (1986), it could be questioned whether colors in a bivariate scale strictly need to be of roughly equal lightness across the diagonal Plus axis — i.e., that the high Y, low X and low Y, high X values have similar lightness. However, the relative appearance of uniformity seems to be a reasonable default design approach to follow, and in accordance with the general principles of color scale design, described earlier in this chapter.

2.6.4 Classification

In the design of any data map, the classification of quantitative data into brackets or classes is an important design choice that defines the visual end result and what it can communicate (see e.g., Schwabish, 2017; Koponen and Hildén, 2019, pp. 96–100). Kraak and Ormeling (2010, p. 128) describe classification as a three-step process in which a map type is chosen first, then the number of classes is selected, and finally the class limits is set. As I have argued previously, the number of classes is effectively limited to three per series in a bivariate map.

The legend of a map should ideally make visible the used classification (Schwabish, 2017). Therefore, the question of classification is closely related to the design of map legends. While a detailed discussion of the methods of data classification for maps is outside the scope of this thesis, the six common classification methods used for choropleth maps as outlined by Koponen and Hildén (2019, p. 97) are:

- **Round numbers**: A series of round numbers within the data domain are used to define class boundaries. This way of defining boundaries does not account for the data distribution and may lead to issues like empty classes.
- **Equal interval**: the domain or range between the minimum and maximum values is divided into a given number of regular intervals that form the class boundaries. Has similar issues as using round numbers.
- **Quantiles**: The data is divided into a given number of categories with the boundaries defined so that each category contains roughly the same number of data points. While this obscures the shape of the distribution, quantiles are useful as a default choice for showing where values falling into different categories are located on a map.
- **Natural breaks** or Jenks: Gaps in the distribution are identified using a mathematical optimization method and used to locate a given number of breaks. Similar values are grouped together but the classes may be very different in sizes. Is not practical if no apparent gaps exist in the data.
- **Standard deviations**: Class breaks are placed at given standard deviations from the median or mean.
- **Compromise**: Breaks are deliberately modified from values derived using one of the above methods to emphasize particular features of the data.

Leonowicz (2006) shows an example of a bivariate legend design in Figure 2.53, where the classification is based on standard deviations and the legend additionally includes the ranges and a scatter plot of the data.

Alternative methods of classifications can also be considered and specifically adapted to the characteristics of bivariate maps. Eyton (1984) proposes a scheme and an accompanying legend (Figure 2.54, A) in which values near the identified normal distribution of the two variables are drawn as an ellipse at the center, while the corner values are assigned their own complementary colors.

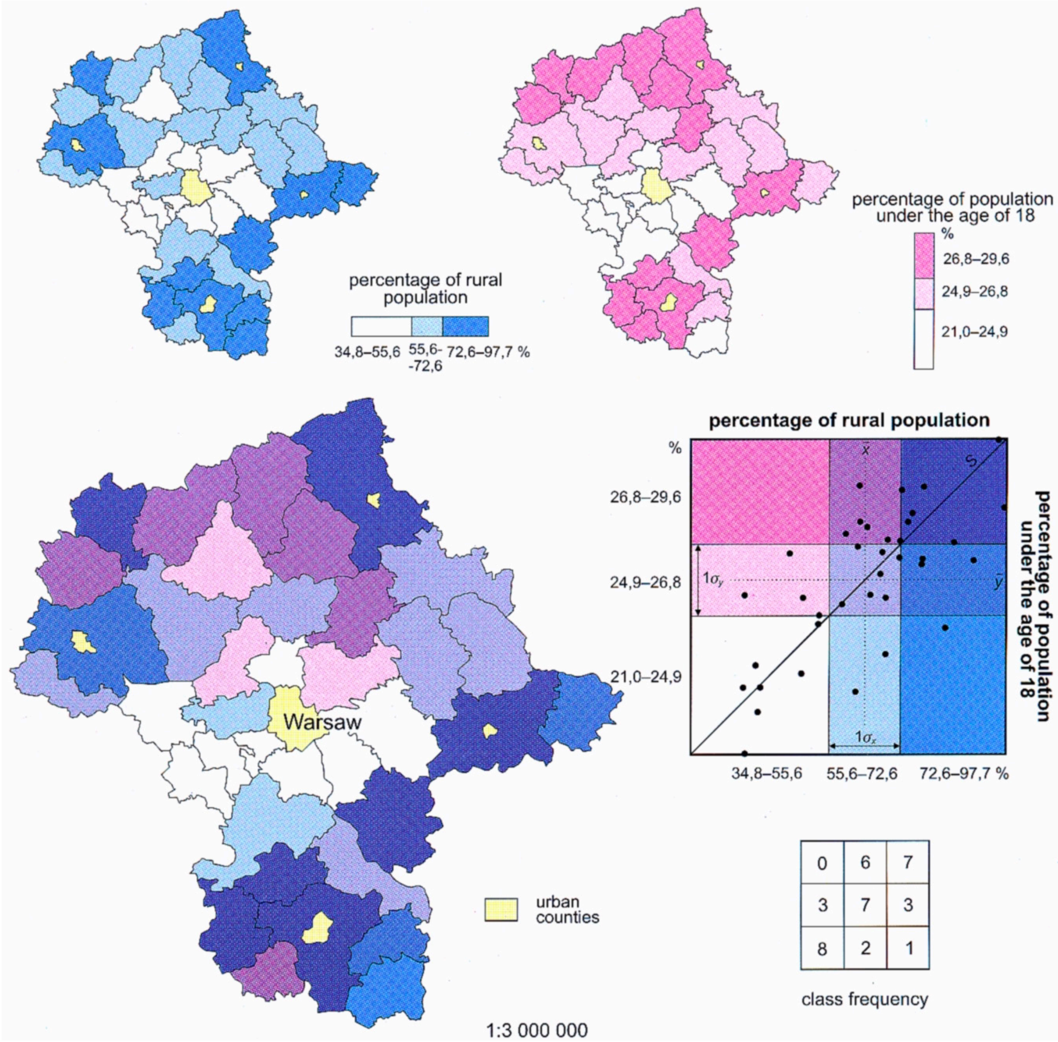


Figure 2.53. 3×3 bivariate choropleth of Mazowsze Region (Województwo Mazowieckie) in Poland, showing the percentage of rural population and the percentage of population under the age of 18 in 37 rural counties. From Leonowicz (2006).

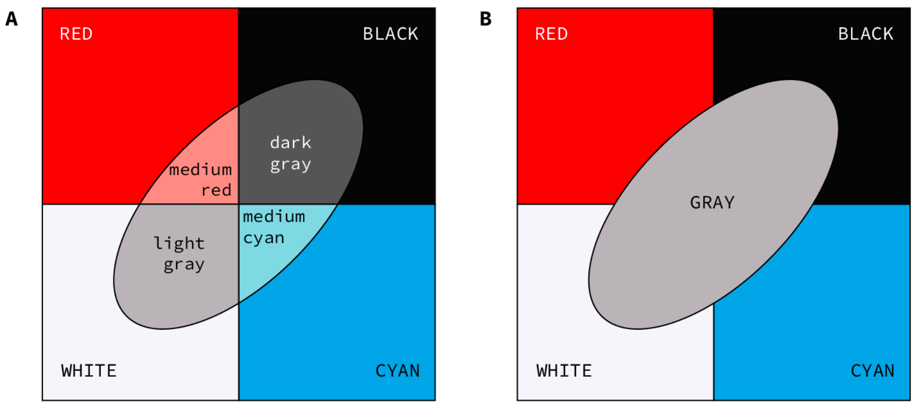


Figure 2.54. A. The classification and legend scheme proposed by Eyton (1984), and B, a simplified version by Dunn (1989). From Strode *et al.* (2020).

Dunn (1989) argues that the customary method of dividing each variable into quantiles and mapping their resulting intersection based on the distribution of values leads to unsatisfactory results. The majority of the data points end up in a few classes while others may be nearly or indeed completely empty. If there is a clear linear correlation this division create what Dunn terms a "'staircase effect' on the main diagonal." As a possible improvement Dunn recommends a simplified version of the scheme by Eyton (1984) (Figure 2.54, B). Additionally, Dunn proposes instead the use of different interactive classifications where the user can decide how the bivariate distribution will be split into classes. Dunn's "high interaction" model is rather complex and is presented as a novel tool for exploratory data analysis by skilled users rather than as a methodology for designing static bivariate maps.

Despite the relevant questions concerning classifications and legend design posed by Dunn (1989), it seems that these alternative solutions in practice remain rare and the simple bivariate map created by the intersection of two classified data series remains the dominant type for which software tools and instructions are readily available (see, e.g., Figure 2.23; the tutorial by Grossenbacher and Zehr (2019), or the bivariate choropleth example created by Bostock (2019))

One design variation that is fairly common to the legends of bivariate maps it to show them rotated at an angle of 45 degrees. This may have been inspired by Bergstrom and West (2018) who proposed a similar modification to scatterplots as a way of emphasizing that there is no particular relation to one axis being considered a predictor and the other being a predicted variable.

In this master's thesis, a simple three-category quantile classification of the diagonal focal model will be used for demonstration purposes in the example visualizations.

2.7 Research-based design and tool creation as method

The practical part of this thesis is an attempt to adapt the theoretical concerns outlined in this chapter to the design of prototype tools. This kind of approach has been described by Leinonen (2010, pp. 56–57) as a *research-based design process*. This approach differs from the similar-sounding *design-based research*, where specific designed interventions are conducted and assessed in different contexts. In research-based design specific tools or *artifacts* are created in an iterative process and based on gathered knowledge, with the purpose of offering affordances to affect some particular problem (in this case making effective bivariate color scales). According to Leinonen this process is by definition iterative.

This approach can alternatively be framed through the concept of generative design, where tool building plays a central role. de Bleser (2016, p. 23) posits that a *generative design approach* can be instrumental in improving the efficiency of designers by allowing them to work around the limitations of ready-made tools. A generative approach also allows for more rapid testing of the generated concepts. An important enabler is tools or software environments that themselves enable creating novel tools — such as Processing by Reas and Fry (2007), Paper.js by Lehni and Puckey (2021), or Nodebox by De Bleser et al. (2015). These types of tools have for instance made programming more viable as an approach, especially to designers without a professional education in computer programming. Recent developments in web design with frameworks like Svelte (Svelte, 2022) have also lowered the threshold for creating and publishing small purpose-built tools without extensive technical knowledge.

While designer tools often are created and used by a single individual a similar approach with more focus on sharing and collaboration is echoed in the rationale behind Jupyter Notebooks as described by Kluyver *et al.* (2016). By having a portable format for publishing live computer code along with rich documentation and images, knowledge sharing can become more effective and interactive, and also more accessible to non-professional coders. This has been a significant enabler of open science and data analysis. Observable (Observable, Inc., 2022) is a more recent example of a notebook approach. While programming notebooks may not typically be considered as belonging to generative design, they integrate program code closely with the visual output that they produce, in a way that is very similar to how Nodebox or Processing have been designed.

Data visualization requires an approach that involves programming when the data complexity hits a certain threshold: as Victor (2013) argues in his lecture *Drawing Dynamic Visualizations* programming allows for flexible visualizations *and* extensive creative control of the visual end result.

Bivariate choropleth maps are a good fit for a research-based or generative approach involving tool creation, as they are sufficiently complex visualizations, and working with them using manual methods often get very tedious rapidly. They are popular but remain rare enough that ready-made creating tools for them are missing from many software packages. The design problem of bivariate color scales is also specific enough to allow the possibility of improvement with custom tools.

In this framework my goals for the production part of this master's thesis could be understood as striving towards four separate outcomes:

1. Making the actual tool(s) that can be used to design and assess bivariate color scales — this is described in Chapter 4, p. 101,
2. how said tool(s) facilitate color scale creation,
3. the actual maps and individual color scales that are made with the tools,
4. insights and ideas for improvement gained from the process of using the tool(s).

3. Methods and analysis model

This chapter describes the methods for the production part of the thesis including data collection, tool creation and the final visualizations. It concludes with the analysis model explaining how the end results are discussed.

3.1 Methods

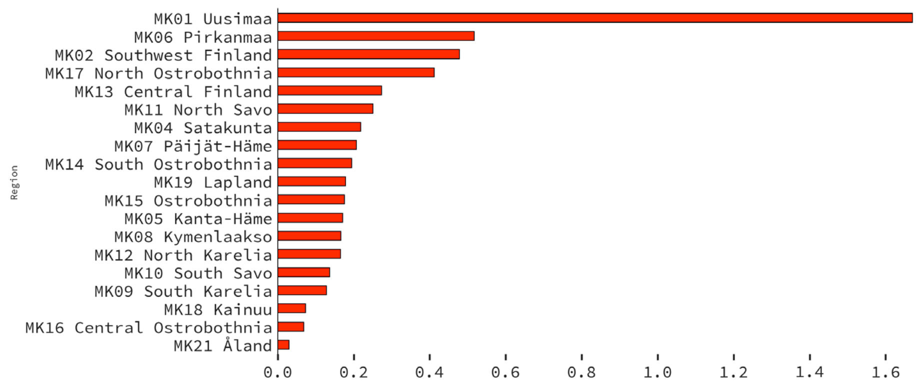
Using 8 different correlated data sets, bivariate choropleth maps of different regional divisions were created to demonstrate and design alternative graphical treatments of the color scales using patterns in addition to hue and lightness.

An interactive tool was developed to aid in the design of additional 3×3 color scales. This work focuses on bivariate choropleth displays with integral dimensions and the diagonal focal model, the use of which assumes that studying two variables in correlation is desirable.

3.1.1 Selection of test data

To demonstrate visually different outcomes with real-world data on different levels of aggregation, two commonly used geographical divisions of Finland were utilized: the regional and municipal levels. Finland has 19 regions (Finnish: maakunta; Swedish: landskap; Statistics Finland, 2020) — this division was used to demonstrate sub-national aggregate data for larger areas. The regions have a skewed distribution in both size and population, with the largest region (Uusimaa) having a population of 1,671,024 and the smallest region (Åland) with just 29,789 inhabitants in 2018. In that year, the population of Uusimaa was 3.2 times that of the second-most populous region, Pirkanmaa. By surface area the largest region (Lapland) is 2.5 times larger than the second-largest, North Ostrobothnia.

By surface area the largest region (Lapland) is 2.5 times larger than the second-largest, North Ostrobothnia.



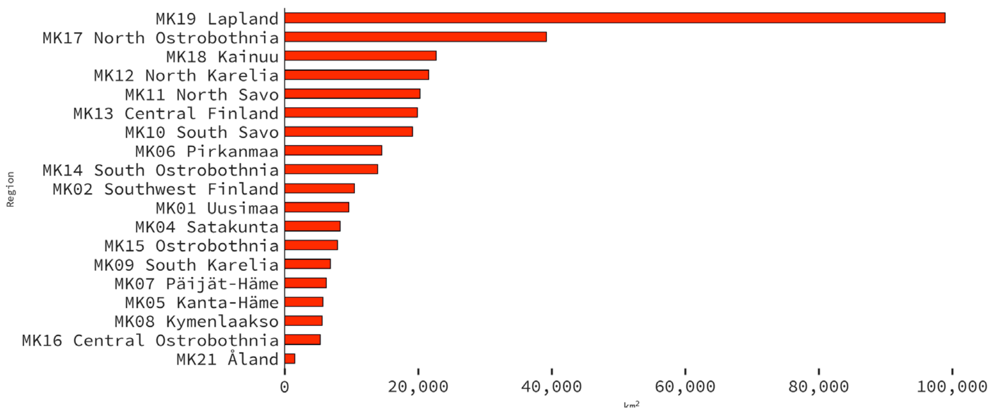


Figure 3.1. Regions of Finland by population size, millions.

Figure 3.2. Regions of Finland by area in km²

The municipal level represents the smallest administrative units of Finland, of which there were 309 in 2018. They vary considerably in both area and population size. According to the Official Statistics of Finland (OSF, 2022), the median population size for the municipalities was 6,134 in 2018. While the largest municipality (Helsinki) had a population of 648,042 — nearly 106 times above the median — the lowest 25 % of municipalities by population had only 2,780 inhabitants or less, with the smallest (Sottunga) having only 91 inhabitants.

According to National Land Survey of Finland (2018), the largest municipality by area is Inari (17,333.65 km²) and the smallest Kauniainen (6 km²), with the median area being 760 km². As can be seen from Figures 3.3. and 3.4, while both population and area distributions skew heavily towards low values, the distribution of areas is slightly more even than that of population.

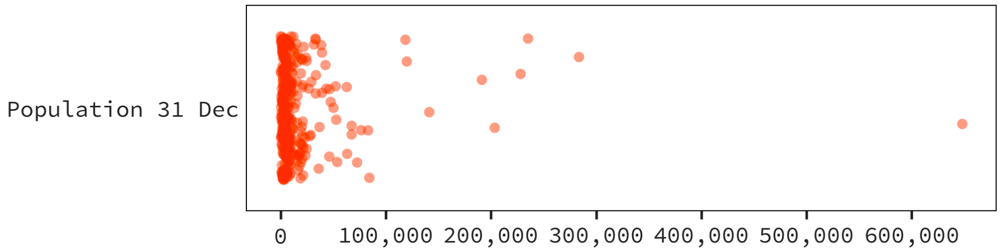


Figure 3.3. Municipalities of Finland by population size (OSF, 2022).

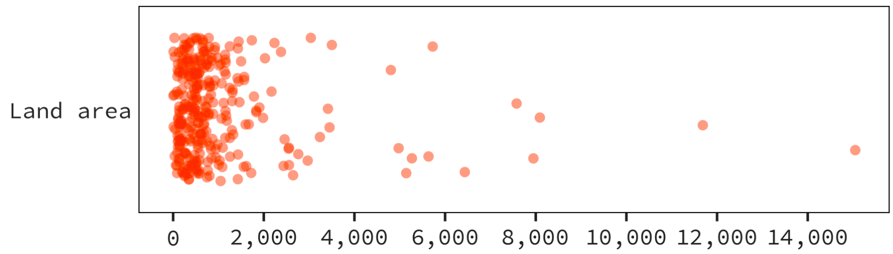


Figure 3.4. Municipalities of Finland by area in km² (National Land Survey of Finland, 2018).

Due to this large range in areas, many smaller municipalities will not be uniquely recognizable in typical choropleth maps. Since the population numbers skew even more unevenly, similar issues occur in population proportional visualizations like Dorling cartograms — although they do solve the visual underrepresentation of small but heavily populated municipalities.

In this thesis, two pairs of data variables were visualized for each level of areal division. The selection criteria employed was to use data sets that presumably are meaningfully correlated, but at the same time have geographical differences in distribution that are visible when represented on a choropleth map.

3.1.1.1 *Statistical data*

Statistical data series were selected for visualization based on exhibiting systematic and discernible regional variation. The data was retrieved from Statistics Finland and visualized preliminarily using the thematic mapping functionality in the *Paikkatietoikkuna* geodata portal (National Land Survey of Finland, 2022).

The pairings of data series were selected based on plausible correlations. To ensure suitability for bivariate mapping, candidate data was analyzed in the Jupyter Lab environment (Project Jupyter, 2019) with the data analysis toolkit *Pandas* (Pandas development team, 2021).

Paired data sets were classified using three quantile groups (both series are divided into three roughly equal-size bins) and visualized with heatmaps based on the resulting 3×3 *contingency tables* (Glen, 2013). An issue that became apparent when exploring model data was that pairings of regional data frequently leads to combinations, where one or more intersections are without data, as shown in Figure 3.5. The contingency table heatmap shows that three out of nine intersecting categories are missing: no region has either high or low shares of both elderly and young people and a high share of children does not occur with medium numbers of elderly.

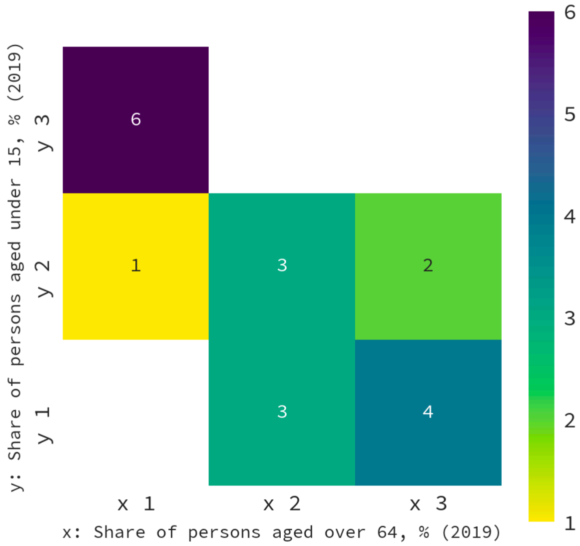


Figure 3.5. Contingency table heatmap of a candidate pair of data sets with three empty categories.

A limitation with the bivariate choropleth encoding in the diagonal focal mode is that it is unsuitable for visualizing data in which both negative and positive values occur. While possible, the resulting encoding would be unintuitive, since zero as a natural middle point will fall arbitrarily in one of the bins in the low to high range. This meant that otherwise applicable data sets had to be discarded. In the typology of Strode *et al.* (2020) the Range variant of the Diverging focal model can be used when one data series has negative and positive values and the Corners focal model could conceivably be adapted for cases where both data series have negative and positive values, but this was deemed beyond the scope of this thesis. Based on these considerations the following data sets were selected. Bin edge values are reported for each of the three categories in both data series.

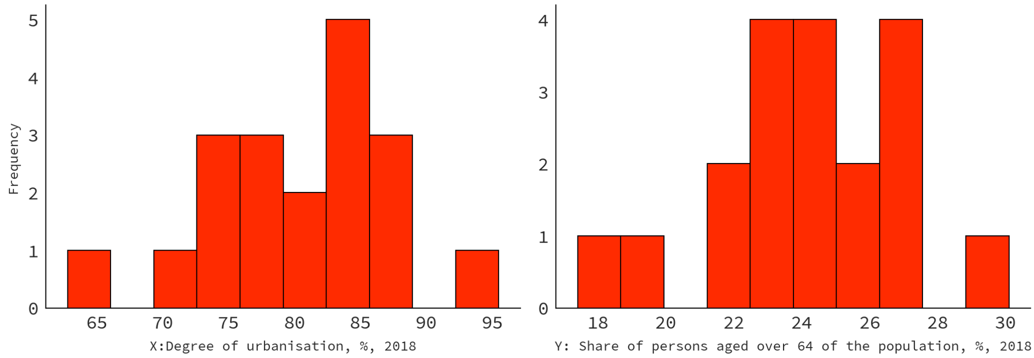


Figure 3.6. Histogram of variables 1.X and 1.Y for regional data.

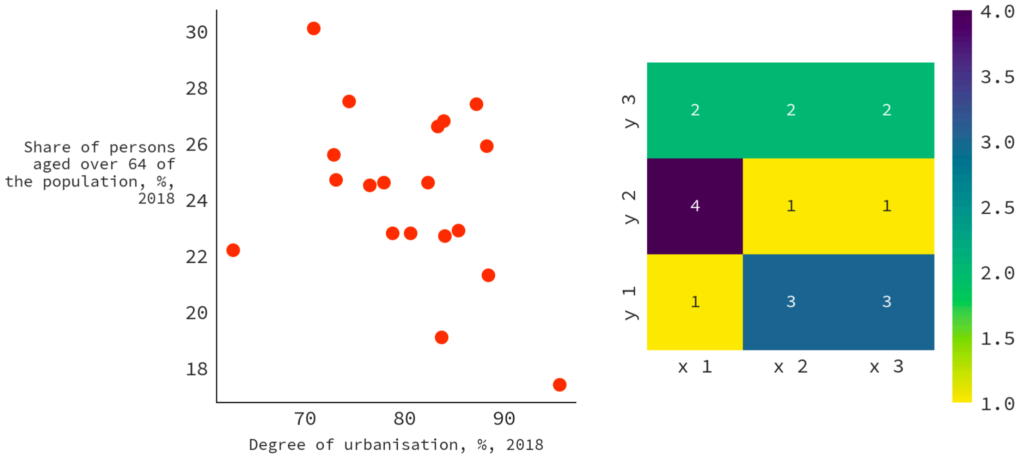


Figure 3.7. Scatterplot and contingency table heatmap of 1.X and 1.Y for regional data.

Regional data, **Pair 1** (Figures 3.6 and 3.7):

- 1.X: **Degree of urbanisation**. Using the table *Degree of urbanisation, %, 2018* from the 2018 data provided by Statistics Finland. "Degree of urbanisation means the proportion of people living in urban settlements among the population whose location is known. Urban settlements are all groups of building with at least 200 inhabitants, where the distance between buildings usually is no more than 200 metres." (Statistics Finland, 2019).
- 1.Y: **Share of persons aged over 65 years**: Using the table *Share of persons aged over 64 of the population, %* from the 2018 data provided by (Statistics Finland, 2019).

Pearson's correlation **r** for 1.X and 1.Y is -0.37.

Bin edges X: minimum 62.8, 77.9, 83.9, maximum 95.5;
Bin edges Y: minimum 17.4, 22.8, 25.6, 30.1

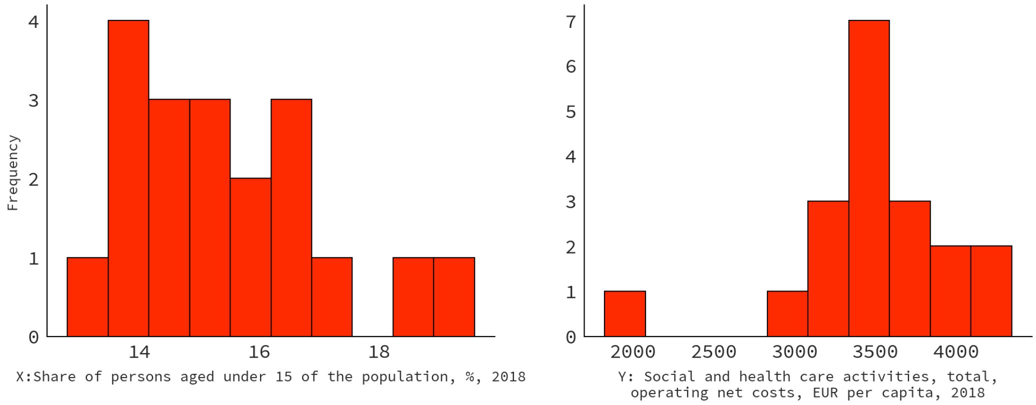


Figure 3.8. Histogram of variables 2.X and 2.Y for regional data.

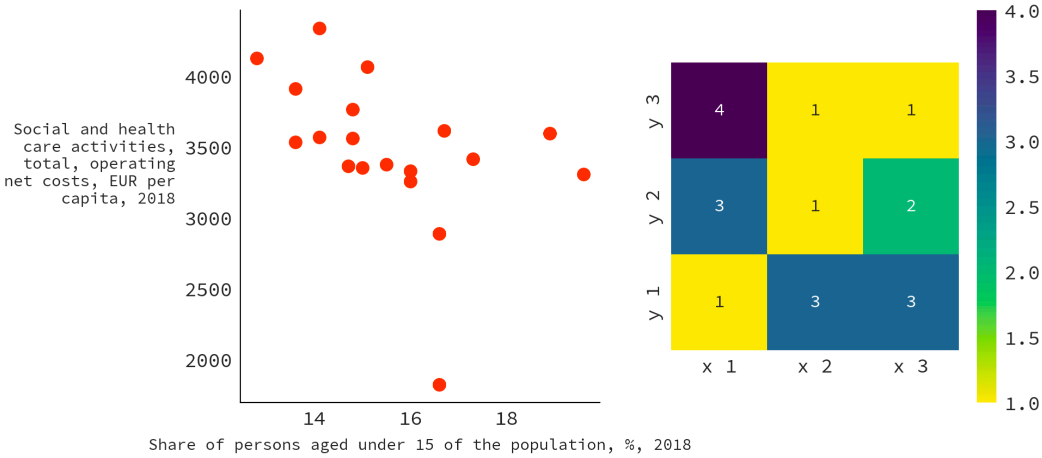


Figure 3.9. Scatterplot and contingency table heatmap of variables 2.X and 2.Y for regional data. The low outlier is Åland.

Regional data, **Pair 2**. (Figures 3.8 and 3.9):

- 2.X: **Share of persons aged under 15**. Using the table *Share of persons aged under 15 of the population, %,* from the 2018 data provided by (Statistics Finland, 2019).
- 2.Y: **Social and health care activities, operating net costs per capita**. Using the table *Social and health care activities, total, operating net costs, EUR per capita* from the 2018 data provided by Statistics Finland. "Operating net costs = operating expenses - operating income. Operating costs = operating expenses total + depreciation and devaluation + allocated common expenses" (Statistics Finland, 2019).

Pearson's correlation **r** for 2.X and 2.Y is -0.43.

Bin edges 2.X: minimum 12.8, 14.8, 16.0, maximum 19.6;
Bin edges 2.Y: minimum 1824.0, 3365.9, 3598.2, maximum 4341.1

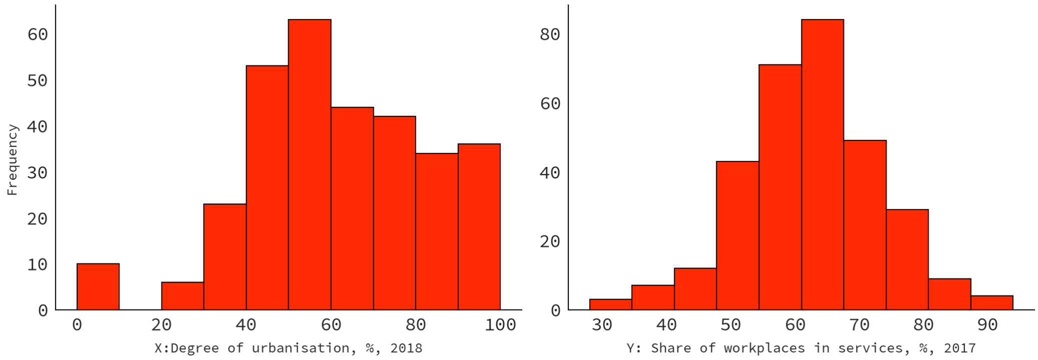


Figure 3.10. Histogram of variables 1.X and 1.Y for municipal data, 1.X is somewhat uneven.

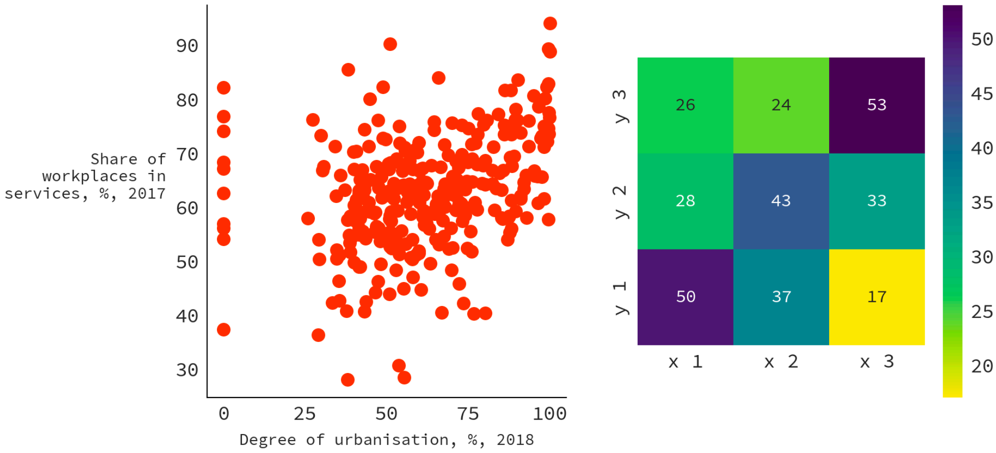


Figure 3.11. Scatterplot and contingency table heatmap of 1.X and 1.Y. for municipal data. A group of municipalities with no urbanization is visible.

Municipal data, **Pair 1**. (Figures 3.10 and 3.11):

- 1.X: **Degree of urbanization**. Using the same table as for regional data (Statistics Finland, 2019).
- 1.Y: **Share of workplaces in services**: Using table *Share of workplaces in services, %* from the 2017 data provided by Statistics Finland (2019, latest available data). In this statistic, every person employed corresponds to one workplace in the region in question. Services consists of categories G–H in the Standard Industrial Classification TOL 2008, including service work both in the private and public sector.

Pearson's correlation **r** for 1.X and 1.Y is 0.33.

Bin edges 1.X: minimum 0.0, 51.1, 71.3, maximum 100.0;
Bin edges 1.Y: minimum 28., 58.4, 66.8, maximum 94.0

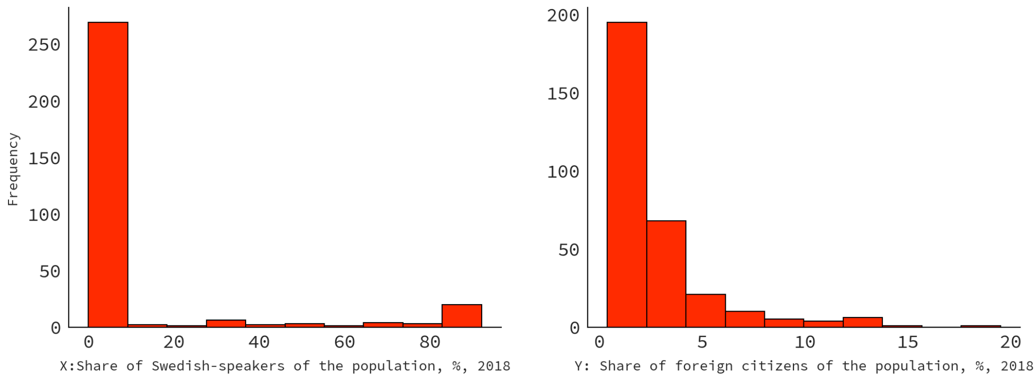


Figure 3.12. Histogram of variables 2.X and 2.Y for municipal data. The distribution of Swedish-speakers (X) has a lump near the high end of the distribution.

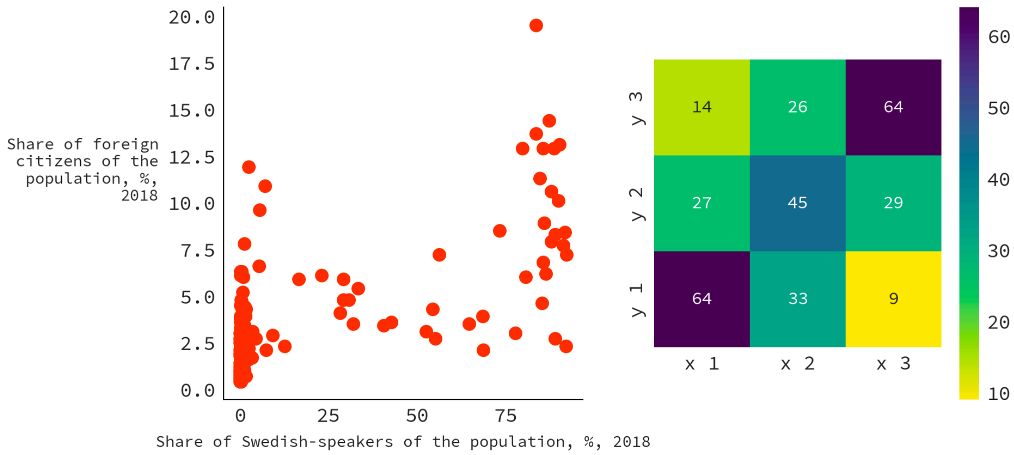


Figure 3.13. Scatterplot and contingency table heatmap of 2.X and 2.Y for municipal data.

Municipal data, pair 2. (Figures 3.12 and 3.13):

- 2.X: **Share of Swedish-speakers**. Using the table *Share of Swedish-speakers of the population, %* from the 2018 data provided by Statistics Finland (2019).
- 2.Y: **Share of foreign citizens**. Using table *Share of foreign citizens of the population, %* from the 2018 data provided by Statistics Finland (2019).

Pearson's correlation **r** for 1.X and 1.Y is 0.71. Both distributions are strongly skewed, with many very low values.

Bin edges 2.X: minimum 0.0, 0.1, 0.4, maximum 92.1;
Bins edges 2.Y: minimum 0.4, 1.3, 2.6, maximum 19.5

3.1.1.2 *Geospatial data*

The geographical areas defining the mapped municipalities are from *Municipalities (1:4,5 M)* (Statistics Finland, 2018a). Regions are derived from the *Regions (1:4,5 M)* data set (Statistics Finland, 2018b).

Figure 3.14. Palettes discussed in the thesis: all 3 by 3 palettes included in the initial release of the biscale package and two example schemes by Brewer.

3.1.2 Some existing color scales for bivariate maps

The R package `biscale` (Prener, 2022) originally included five palettes: Dark Blue (`'DkBlue'`), Dark Cyan (`'DkCyan'`), Brown (`'Brown'`), Gray Pink (`'GrPink'`) and Dark Violet (`'DkViolet'`)[31]. The first four were created by Joshua Stevens (2015) who did not name the palettes. Bostock's (2019) bivariate map notebook uses Stevens' palettes, but with a slightly different naming convention. The Dark Violet palette is an addition by Grossenbacher and Zehr (Grossenbacher and Zehr, 2019). These are similar to, but not identical with the two sequential/sequential example schemes given by Brewer (1994, p. 141). The discussed palettes are illustrated in Figure 3.14. The color values for Brewer's palettes are not given in writing in the original source. They are instead taken from the R package `pals` (Wright, 2021), which also includes a number of other palettes not listed here.

3.1.3 Manipulating and analyzing colors

In this thesis the main tool chosen to manipulate and analyze colors is the JavaScript framework Chroma.js by Gregor Aisch (2022a). It is mature, widely used[32] and offers sufficient functionality for practical purposes, while being an easy-to-use implementation as well. Additionally it allows for convenient generation of color scales with some perceptual corrections for lightness. Chroma.js uses CIELAB internally for color manipulation. I use the R package `colorblindcheck` (Nowosad, 2019) to simulate color vision deficiencies for the created palettes.

---

31 As of the v1.0.0 release 2nd June 2022 a total of 17 palettes are included, although 10 of these are pairs that correspond to the initial five to retain compatibility with previous versions (Prener, 2022).

32 i.e., 292,480 weekly downloads on NPM, https://www.npmjs.com/package/chroma-js

3.1.4 Contrast and separation of colors in bivariate maps

As discussed in Chapter 2 a three-step sequential lightness color scale where all color combinations satisfy a 3:1 contrast requirement with each other and the background is not possible within a standard computer monitor gamut. However, using the updated requirements suggested by Somers (2021a) to construct three-step color scales with a minimum ACPA contrast of 30 between each step becomes possible.

Despite this, it is still clear that a bivariate 3×3 scale with 9 unique colors is impossible to design in such a way that it is both visually uniform and that every color would achieve minimum contrast with every other color in the scale. Indeed attempting this is not even desirable, as a good quantitative bivariate scale is expected to have a more or less uniform change in lightness (Trumbo, 1981; see e.g., Brewer, 1994; Stevens, 2015) along the *X*, *Y* and + axes (see Chapter 2). Thus, the previously mentioned approach of using strokes to ensure separation between colors has been used in this thesis, employing the standard WCAG2.1 contrast criterion of 3:1.

In a bivariate choropleth map it is not sufficient that colors next to each other are distinguishable as separate. In addition, color pairs of similar lightness but representing values on the different axes axis also need to identifiable as different hues. This is assessed visually and using the Delta E calculation in Chroma.js. As a reasonably robust absolute minimum the value of 11 for Delta E used in Leonardo[33] is employed in this thesis (Baldwin, 2022).

3.1.5 Assessing contrast of bivariate schemes

The online tool created for this thesis allows comparison and assessment of color scales for contrast. It uses both the current WCAG and the ACPA algorithm as of 1 January 2022. In Figure 3.15 the Contrast grid visualization is shown displaying contrast values between all 9 color combinations + white in the DkBlue bivariate color scale.43 of the 90 combinations (excluding colors paired against themselves) exceed the APCA 30 minimum without strokes.

33 leonardocolor.io

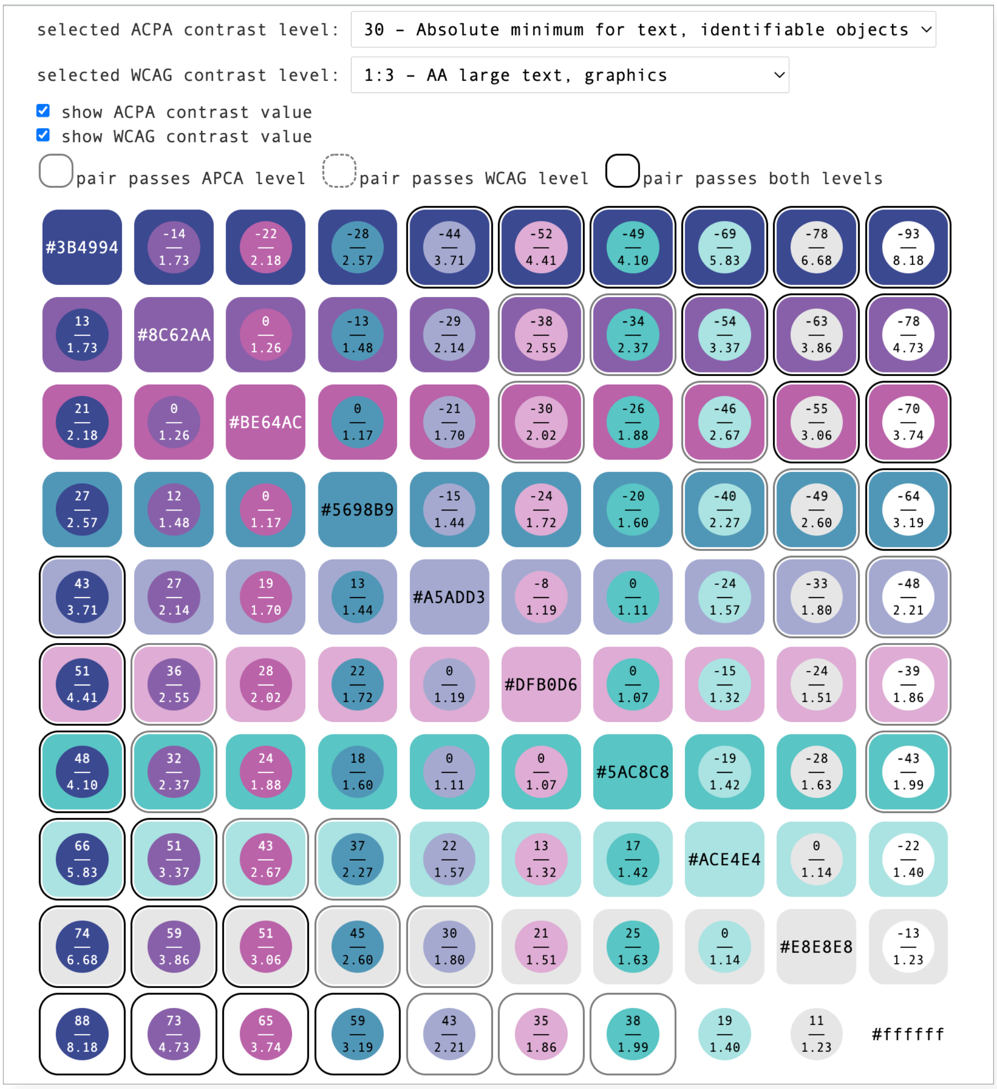


Figure 3.15. Contrast grid visualization showing contrast values for all possible color pairs in DkBlue bivariate color scale.

3.1.6 Color blindness

Nowosad (2020) has assessed 12 bivariate palettes in the `pals` package using the `colorblindcheck` package (Nowosad, 2019). The Delta E distances between the colors in each palette was compared with appearances simulated for three classes of color vision deficiencies (deuteranopia, protanopia and tritanopia). About 8% of males and 0.5% of females have some color vision deficiency, mostly affecting differentiation of red and green: protanomalia/protanopia and deuteranomalia/deuteranopia. Tritanopia causing difficulty distinguishing blue and yellow is much rarer, affecting only 0.008% of both males and females (Kalloniatis and Luu, 2007). Nowosad's conclusion is that `brewer.seqseq2` (1994a), `stevens.greenblue` (Dark Cyan), and `stevens.purplegold` (Brown) perform best in this regard, using a minimum cutoff value of 6 for Delta E. The new palettes created as a part of this thesis will be assessed with the `colorblindcheck` package. A further improvement would be to integrate colorblindness checking and simulation directly in the bivariate matrix tool, similar to the example provided by Aisch (2018).

3.1.7 Creation of textures

There are many implementations of patterns or textures for different visualization environments. Matplotlib (Matplotlib development team, 2022) has a pattern functionality called *hatch* with a few ready-made textures that can be used in Geopandas (Jordahl *et al.*, 2020). For `ggplot2` in R, there is the package `ggpattern` (FC and Davis, 2022), which is more flexible — but not integrated with the `biscale` package for building bivariate maps. `Ggpattern` is a useful example insofar as it offers programmatic control of multiple relevant texture properties, including angle, density or the proportion of area filled, spacing (meaning the scale of the pattern), etc.[34]

The open-source geographic information software QGIS also supports customized texture designs, but with the limitation that the output is rasterized. Because textures created in one visualization environment cannot easily be ported or used in another, and instead they have to be re-implemented, I deemed implementing ready-to-use new textures for a specific environment to be outside the scope of this thesis. Instead, the demonstration texture palette was created as vector graphics in Adobe Illustrator™, which offers a flexible and practical tool for the design of repeating texture patterns.

3.1.8 Making custom color scale tools

The color scale tools have been developed as a web application called Colorgridder[35] with Svelte.js (Svelte, 2022). Svelte was chosen based on its straightforward development experience, easy hosting and smooth client-side interactivity — all code runs as JavaScript in the browser. This is unlike, e.g., Shiny (2022) for R, which needs to be hosted on a web server. Chart axes and map rendering uses functions from the JavaScript library D3 (2022).

Additional color scale testing and comparisons was again largely done in the JavaScript-based Observable (2022) notebook environment.

The application uses Chroma.js[36] for color manipulation functionality throughout. It has two different modes: one for analyzing color lists by contrast (*Color Scale Grid*) and the other for creating novel bivariate palettes (*Bivariate Hue Blender*). The application displays colors in the common and widely supported sRGB color space. Color values can be input as hexadecimal RGB color codes — hex strings for short, where the value for each of the three color channels (R, G, and B) is represented by the two digits of a hexadecimal byte (Web Colors Explained, 2006).

3.1.9 Previous art

The design and content of the color scale tools in the application have been particularly influenced by Aisch's *Chroma.js color palette helper*[37] which also is

34 coolbutuseless.github.io/package/ggpattern/articles/patterns-stripes.html
35 Hosted as a Github repository at github.com/hjhilden/svelte-colorgridder
36 Documentation: https://gka.github.io/chroma.js/
37 Aisch (2022b), gka.github.io/palettes

built in Svelte.js (Figure 3.16). Another influence was *Hcl wizard*[38] by Zeileis *et al.* (2020) which served as a demonstration of the direct manipulation of hue, chroma and lightness of visualization palette colors using sliders, with accompanying map preview (Figure 3.17).

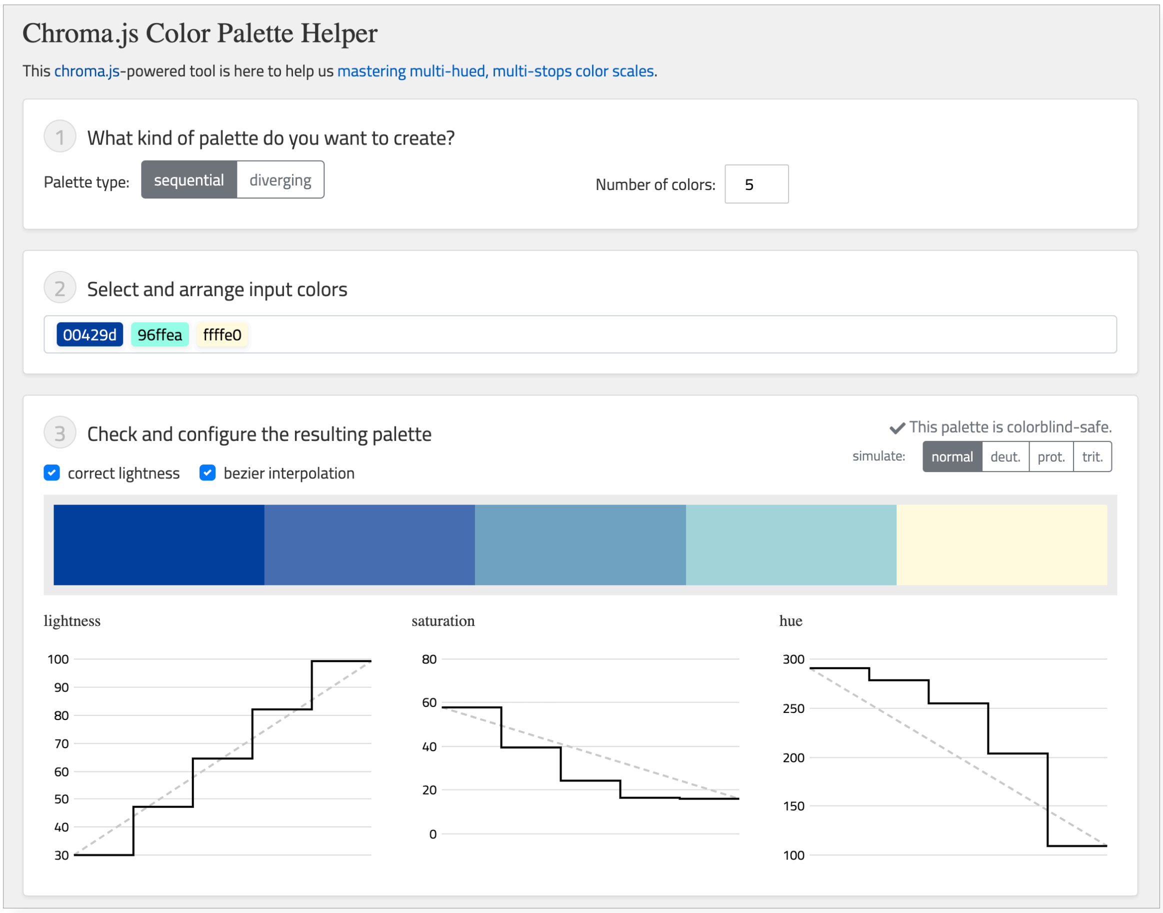


Figure 3.16. The Chroma.js color palette helper (Aisch, 2022b). Screenshot.

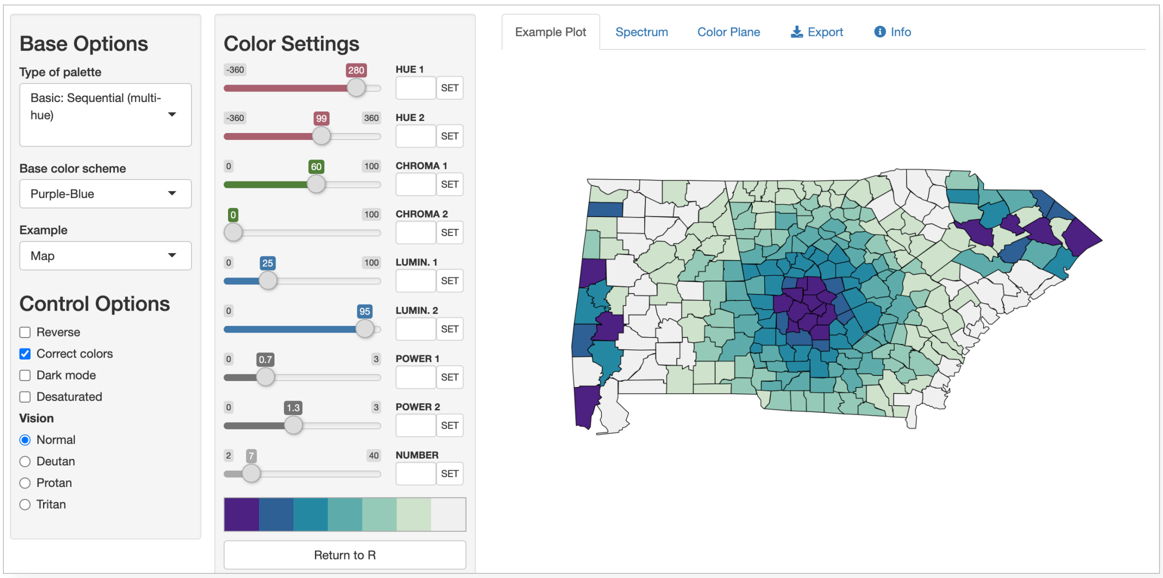


Figure 3.17. The Hcl wizard web tool (Zeileis *et al.*, 2020). Screenshot.

38 Available online at: http://hclwizard.org:3000/hclwizard/

The *Bivariate Choropleth Color Generator* by Brooke (2019) on Observable[39] was an important influence on the construction and layout of the palettes in the Bivariate Hue Blender.

Other significant influences were the *Huetone* online app by Ardov (2022)[40]; Meeks' and Lu's *Viz palette* (2018)[41] ; the previously mentioned *Leonardo* by Baldwin (2022)[42] and lastly the *Color picker for data* by Brown (2022)[43], which gave the original idea of a color picker that indicates out-of-gamut colors (illustrated in Figure 2.12). The color scale grid layout was influenced by the web tool *Accessible Brand Colors*[44] (Use All Five, 2019).

3.1.10 Creation of bivariate maps

Bivariate maps were made as vector graphics output with the Python library Geopandas (Jordahl *et al.*, 2020) using custom code loosely based on the R package `biscale` (Prener, Grossenbacher and Zehr, 2020). The color scales created use the same formatting and also be used in the `biscale` package. The Jupyter notebooks are available as attachments.

Testing and exploration were done in RStudio, even if the intention of creating the maps in the R environment was abandoned during the working process. The rationale for using geopandas rather than `biscale` for the final maps was that this turned out to be more straightforward, as both data pre-processing and visualizing could be done in one environment that was more familiar to me.

Some additional testing of geometry was further done in the graphical Python environment Drawbot (van Rossum, van Blokland and Berlaen, 2021). QGIS was used to handle and retrieve map data and Adobe Illustrator™ was used for graphics compositing and drawn illustrations.

3.2 Analysis model

- The color palettes created are assessed by color distances (Delta E) and compared with pre-existing palettes
- Color palettes are applied to maps using real-world statistical data
- A texture design is applied to the created maps
- Map and palette combinations are discussed through subjective observation, partly based on the task categories of Bertin (2011): *elementary*, *intermediate* and *global*.

39 Available at https://observablehq.com/@benjaminadk/bivariate-choropleth-color-generator
40 huetone.ardov.me
41 projects.susielu.com/viz-palette
42 leonardocolor.io
43 http://tristen.ca/hcl-picker
44 abc.useallfive.com

4. Description

The production part consists of bivariate maps made using four pairs of data sets, two on the regional level and two on the municipality level. Using the color scale tool, three new bivariate color scales were created. The bivariate maps demonstrate both versions combining textures and colors to separate categories and versions with color differences alone.

A summary of the map creation process is as follows, illustrated in Figure 4.1:

1. Candidate data from Paikkatietoikkuna was explored and saved by copying from displayed tables into csv files. These were read into a Jupyter notebook and analyzed with Matplotlib to determine the selected data.
2. The selected data was joined with tables of relevant geographical codes.
3. Geographical data for regions was loaded using Geopandas and joined with with the tabular data by municipality or regional codes.
4. Bivariate map outputs were rendered using custom Python code for Geopandas and saved to vector pdf files. Color palettes were copied from the Biscale hue blender web application.
5. Vector files were imported and collated in Adobe Illustrator. In the case of the texture demonstrations, textures were applied manually to the texture examples using the *select similar* functionality and example texture palettes.

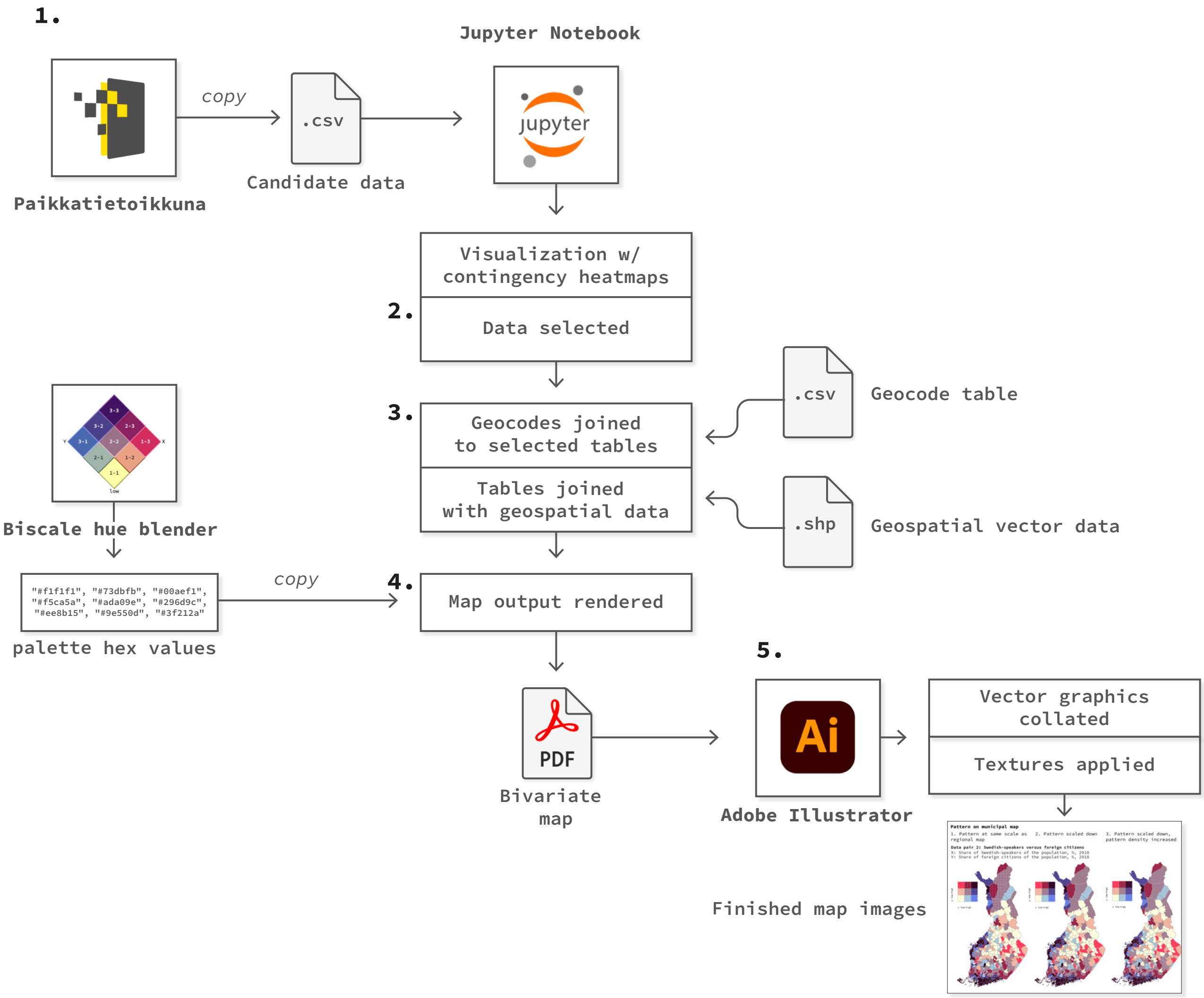


Figure 4.1. Diagram of the map creation process from data to final example images.

4.1 The web application

The web tool or application is a standalone website that has two modes in separate tabs: the color scale assessment mode **Contrast grid** and the palette creation mode **Bivariate hue blender**[45].

The **Contrast grid** mode is shown in Figure 4.2. It allows for a gridded comparison and analysis of a supplied color scale provided as a comma-separated list of hexadecimal RGB strings (hex colors like `#00ff00`). The main purpose of this tool is to quickly assess relative contrasts between all colors within a color palette. For this purpose different contrast level requirements can be set interactively, and the desired assessments can be toggled.

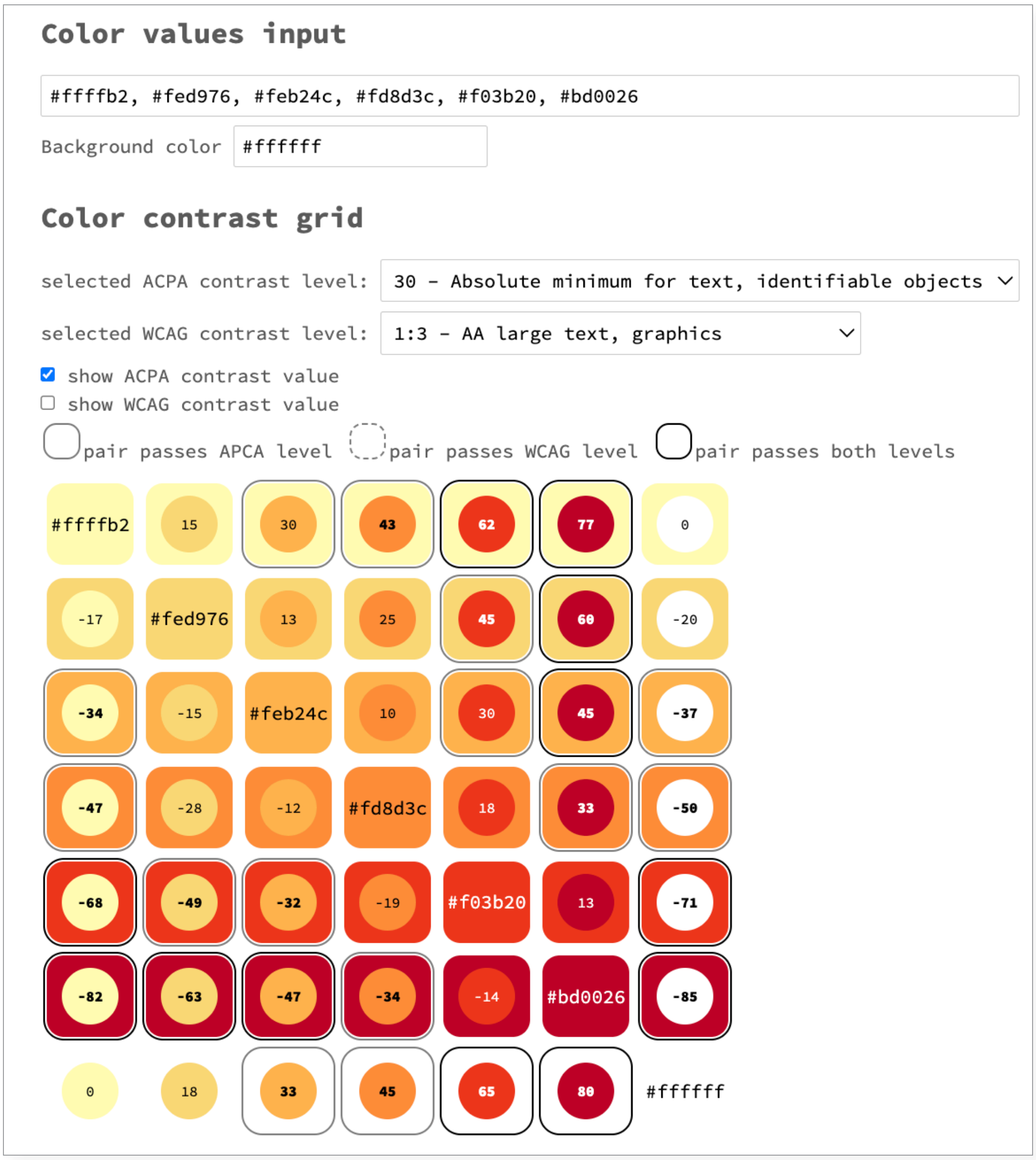


Fig 4.2. Main view of contrast grid mode, October 2022 version.

45 The application is online at https://demo.koponen-hilden.fi/colorgridder and available as a public repository at https://github.com/hjhilden/svelte-colorgridder

With a given list of colors, it is easy to check whether the pairings exceed a given contrast requirement (either APCA and/or WCAG2) for every combination in the list. This grid visualization can be exported as a SVG vector graphic for reuse in visual design software. Additionally, it generates a set of derived border colors that can be used between colors that otherwise have insufficient contrast.

Figure 4.3. Early version of bivariate color matrix tool using Chroma.js to construct a lightness-adjusted color scale from two provided input hues (y), May 2022. The second (x) scale is created by offsetting the hue of the y axis hues 180°. The + axis colors are created from x and y with the “multiply” blending mode.

An early version of **Bivariate hue blender** is shown in Figure 4.3. This version used a HTML grid to show the color palette. The current version from December 2022 renders the color palette as a SVG graphic and is shown as an overview in Figure 4.4.

By default the Bivariate hue blender takes the input of two colors as RGB hex strings (`#ffffb0`) from which a bivariate palette is generated. Colors are named with paired numbers corresponding to their position in the matrix, borrowing the convention used for palette definitions in `biscale`.

The Bivariate hue blender allows for adjusting and interactively creating bivariate color scales with 2×2, 3×3 or 4×4 colors (the 4×4 option is included mostly to demonstrate the difficulty of reading a palette with so many colors),

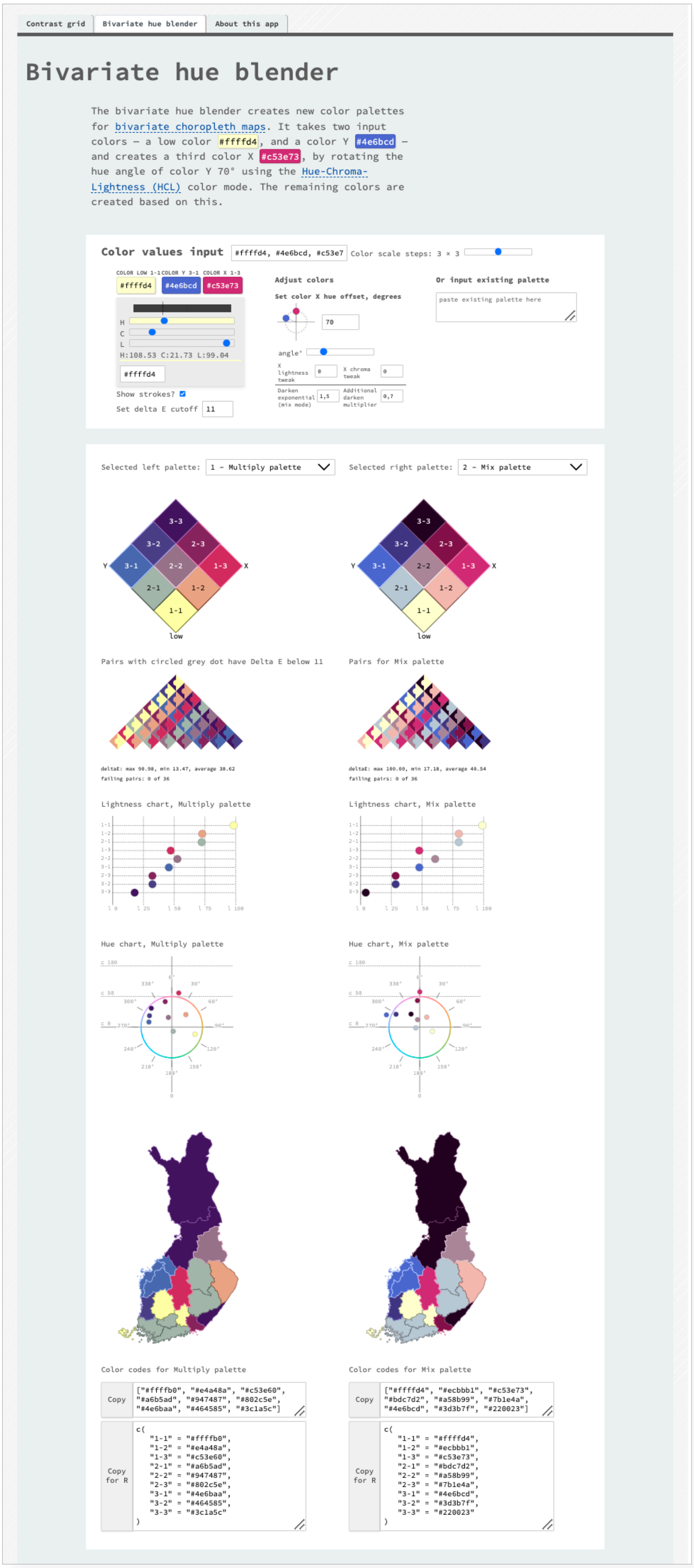


Figure 4.4. Overview of the Bivariate hue blender, December 2022 version. Demonstration palette with input colors #ffffd4, #4e6bcd.

starting from two initial colors (*low 1-1* and *Y 3-1*). The inputs and the palettes are shown in detail in Figure 4.5. The inputs are divided in three columns. The first shows the three main colors as buttons and a Hue-Chroma-Lightness (HCL) color picker. The individual colors can be adjusted with the picker by clicking the corresponding button. The second column has the angle adjustment input that controls the hue of color X and a small hue-chroma visualization of the colors X and Y. Underneath it are numeric inputs where the lightness or chroma of color X can be increased or reduced. Underneath the dividing line are two variables controlling the mix palette mode. Alternatively complete bivariate color scales generated elsewhere can be inputted by pasting the color values as comma-separated hex codes into a text field in the third, rightmost column. The tool visualizes the created palettes as color legends rotated 45°, with the color corresponding to the low×low or 1-1 value pointing down. The tool creates two parallel colors scales using two different mixing methods, which are visualized in two identical columns. The column assignment can be switched using a drop-down menu. Each palette color swatch can also be edited manually.

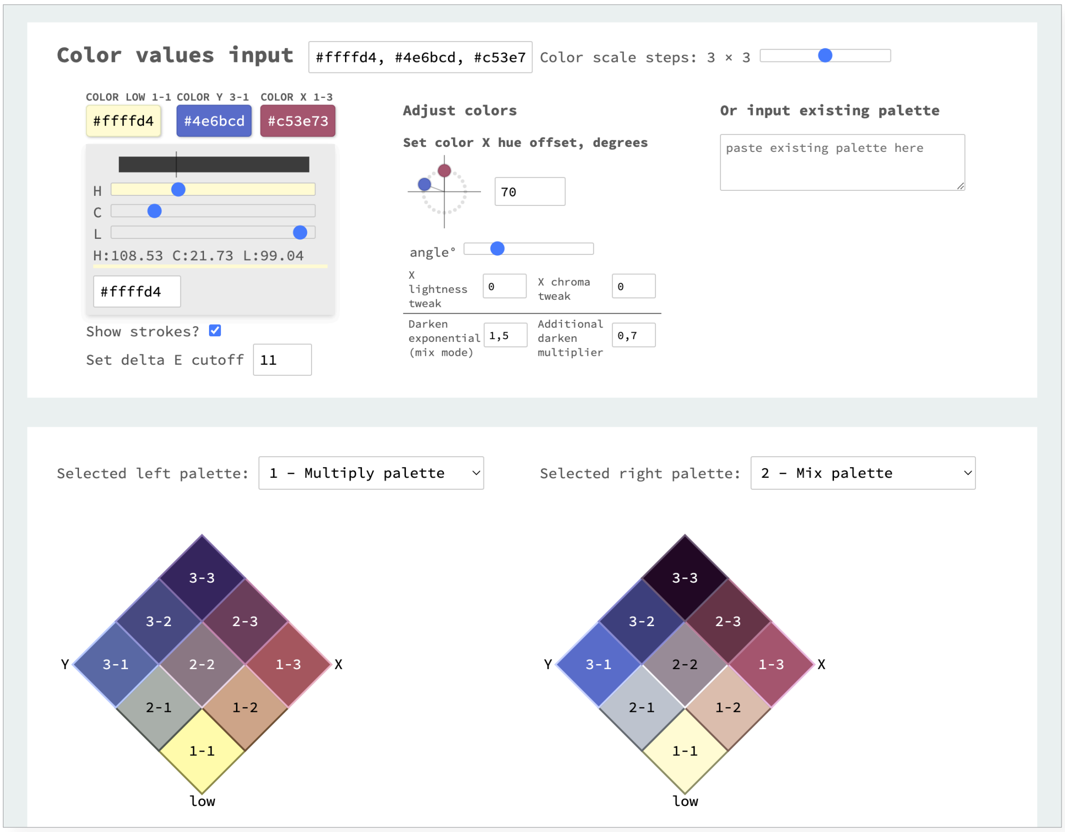


Figure 4.6. View of the input and controls of the Bivariate hue blender with two palette versions using the different mixing modes.

Under the color legends is a pairwise comparison that visualizes how every possible color pair looks and gives a summary of minimum, maximum and average contrast values (Fig. 4.6). Color combinations that have a Delta E difference below a set user-adjustable minimum value (default 22) are indicated with a grey dot marker.

Figure 4.6. Pairwise comparison visualization of color pairs in two palettes.

Below the pairwise comparisons there is a lightness chart and a hue chart for the color palettes (Fig. 4.7). with lightness on the X axis, so that the pairs with corresponding levels are close together. This is contrasted with a hue chart that shows the palette swatches as dots in a radial layout, positioned by hue angle and chroma as a function of distance from the center. Additionally, a small map preview (Fig. 4.8) using the regions of Finland is included.

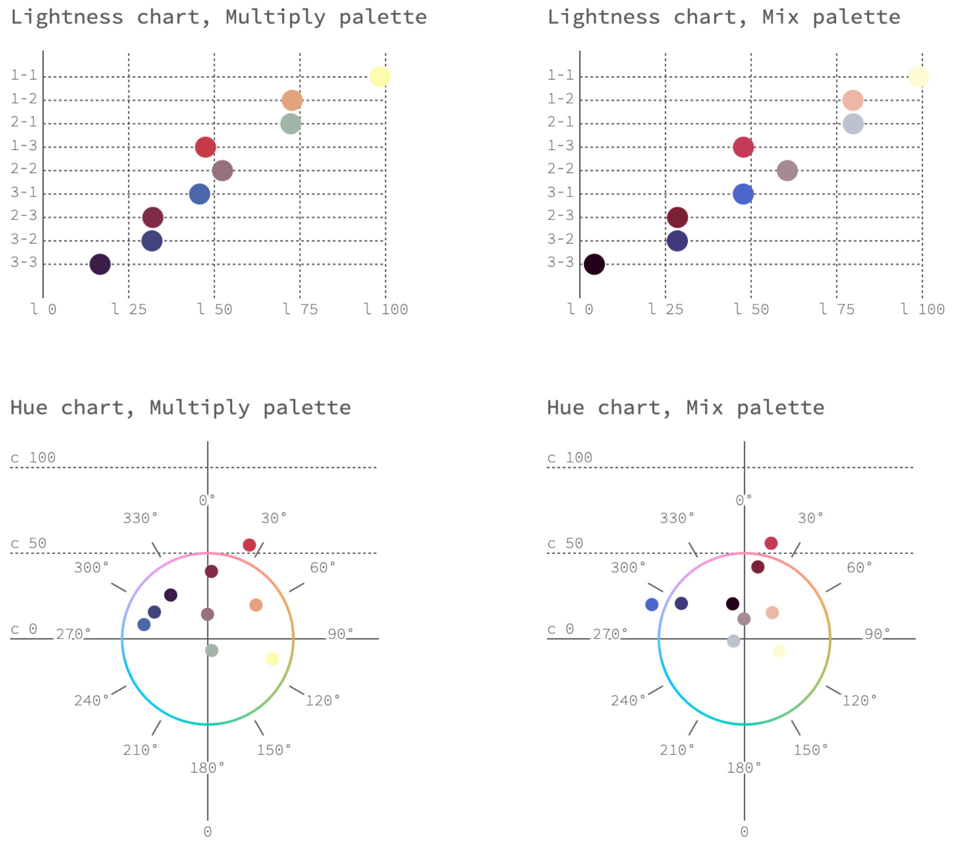


Figure 4.7. Lightness (above) and hue charts (below) for two palettes, multiply and mix color blending modes.

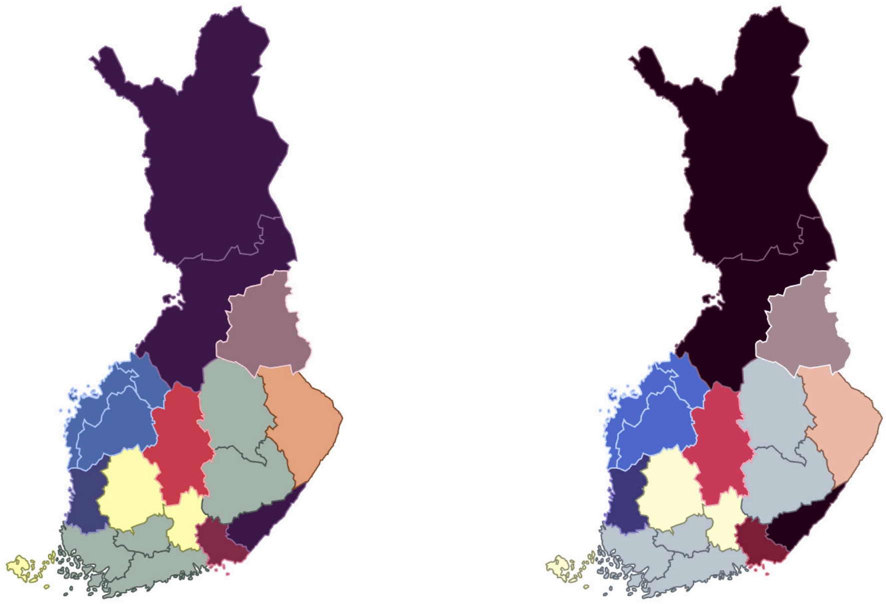


Figure 4.8. Map preview of two palettes, multiply (right) and mix (left) color blending modes.

Generated palettes can be copied as hex strings, formatted as either lists of values usable in Python or JavaScript environments or as R vectors in the format used by the `biscale` module (Prener, Grossenbacher and Zehr, 2020).

4.1.1 Palette creation with the Bivariate hue blender

The principle of palette generation is illustrated in Figure 4.9. The 1-1 or shared low×low *Color low* and the 3-1 or high×low color *Y* are the default inputs — a common lightest color and the *Y* color. The *X* color is by default created by shifting the hue of *Y* by 85 degrees. Chroma (distance from the center in the chart) is kept constant if this is possible within the gamut. The intermediate colors (2-1 and 1-2) are generated with uniform lightness steps using Chroma.js interpolation[46]. The remaining colors are derived by mixing the hues of the opponent Y and X color axes. The chroma and lightness of the generated X color can be tweaked manually via two sliders.

46 In the early version shown in Figure 4.3, the lightest color was also created by hue rotation and blending, but this led to less control over the palette appearance.

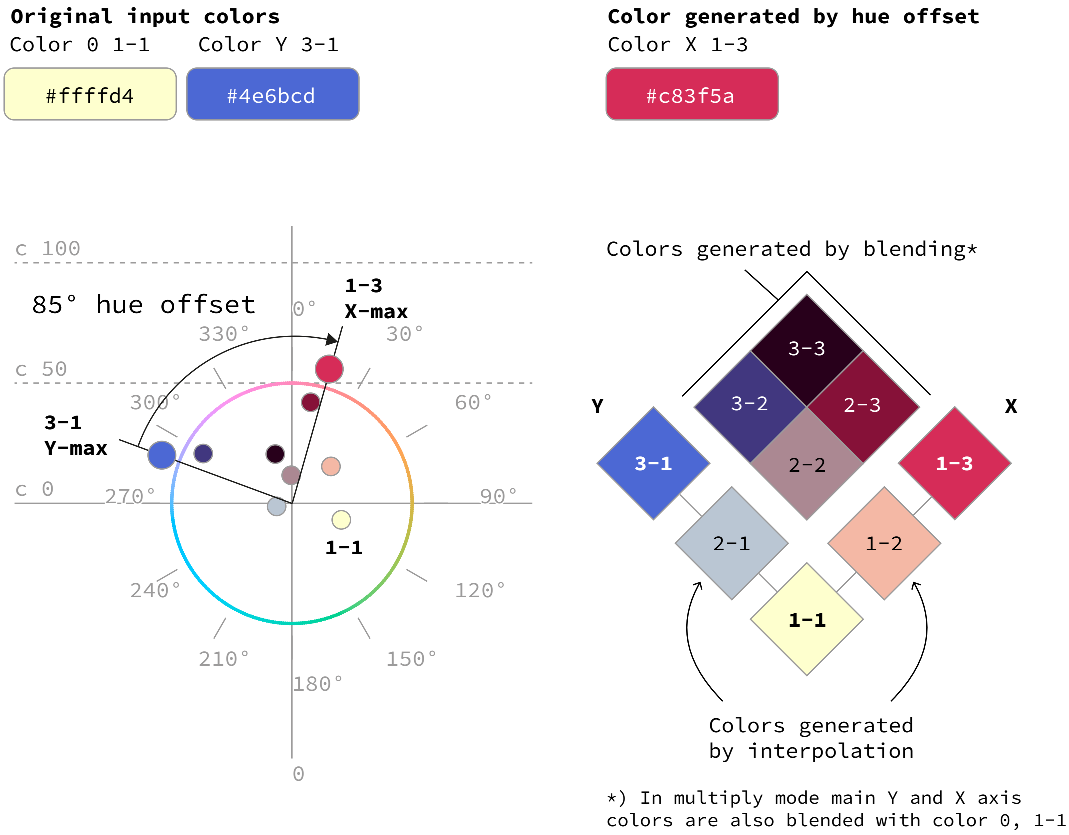


Figure 4.9. Schematic explaining the creation of a palette using a hue offset, interpolation and blending. Input angles over 360° wrap around, so when the starting hue angle for 3-1 was 290°, a blue color, the resulting hue angle with an 85° offset is 15° for 1-3, resulting in a red color.

In addition to the fixed color picker that allows editing of the three main colors, every generated color can also be edited manually by clicking the palette visualization, which opens up a secondary pop-up color picker, as seen in Figure 4.10 (NB! in the December 2022 version, this functionality is only available in the left column). The available hue range for the selected chroma and lightness is displayed as a gradient in the color picker.

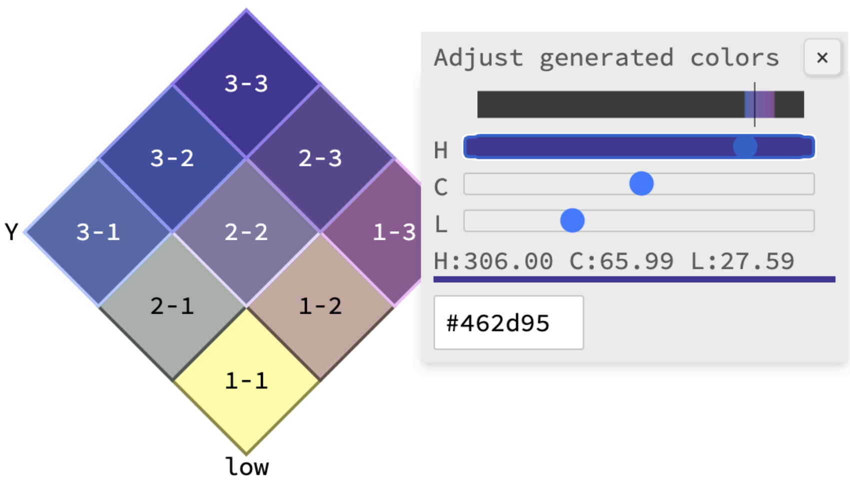


Figure. 4.10. Editing the palette colors with the color picker.

Color picking was influenced by the *Huetone* online tool Ardov (2022). The approach used in my color picker design is more numbers-oriented than the color picking functionalities in Aisch's *Color palette helper* or Meeks' and Lu's *Viz palette*. These both uses a collection of "preview" swatches in different hues to indicate adjustment directions. The color picker used here may be somewhat less intuitive but has the advantage of explicitly showing out-of-gamut regions of the hue scale (black sections in the hue range in Figure 4.10).

4.1.1.1 *Alternatives for mixing colors*

Based on Steven's recommendations (Stevens, 2015) and Brooke's *Bivariate Choropleth Color Generator* on Observable (2019) *darken* and *multiply* color blending modes were first implemented as the methods for creating blended color scales in the bivariate matrix tool. *Darken* leads to blended colors that are lighter and more saturated, but trials showed that it results in poor to no differentiation between colors if colors *X* and *Y* are kept identical in lightness along either edge of the palette (high X or high Y) due to how the RGB channels are mixed. This can be seen clearly in the comparison in Figure 4.11, where the colors 2-3 and 1-3 are identical when using the darken mode (right). This is due to the color 1-3 having lower values than 2-1 on all three RGB channels.

This issue can be somewhat overcome by changing the lightness as well as the hue of *X* relative to *Y*. However, such adjustments also make the color scale less visually uniform. The basic assumption is that colors on the X axis should correspond reasonably well in lightness to their opposites on the Y axis (2-1 is the same lightness as 1-2, etc.). Hence the darken mode was discarded.

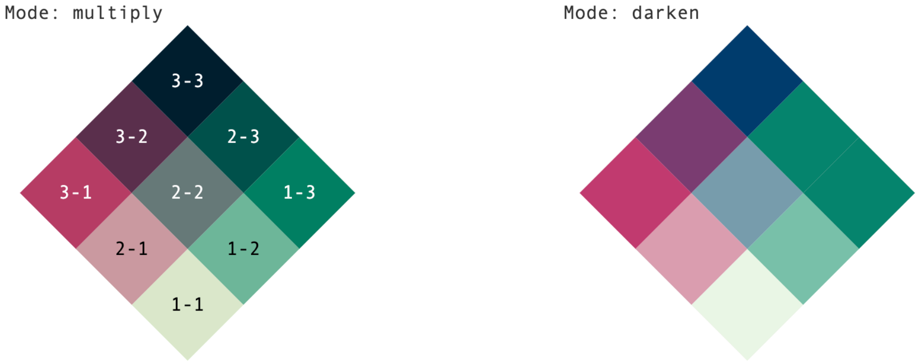


Figure 4.11. Illustration of case where applying darken blending mode results in no differentiation between colors 2-3 and 1-3 (right).

The advantage of the multiply blending mode is that it works intuitively by resembling the appearance of overlaid colored films. However, because the multiply blending mode creates colors by mixing the two series, the extremes of the X and Y series are not the pure input colors, but rather mixes with the lightest color[47].

47 This follows directly from implementing the palette creation as described by Stevens (2015).

Therefore a third method for creating color scales named *mix* was devised using the `chroma.mix` function of Chroma.js. The mix function works by interpolating between two colors in a given color space, thus offering more control over the result and pure, unmixed colors at the X and Y extremes.

Since the mix function attempts to leave lightness unaffected it was coupled with the `darken` method in Chroma.js (not to be confused with the blending mode of the same name) to make the hues corresponding to higher values darker — i.e., reduce their lightness. To adjust the amount of lightness reduction applied to each color a common easing function is used (exponential ease in, from Sitnik and Solovev, 2022) based on the multiplied index of the X and Y series. Only the mixed colors for values above the middle of the Plus axis (X and Y are equivalent) are darkened. The rate and coefficient can be modified by the user to affect to what extent lightness is reduced.

4.1.2 Ensuring contrast and difference between colors

Originally my intention was to examine bivariate palettes using the Contrast grid tool, but this turned out to be unnecessary. This because due to the design of the scale and the available dynamic range, colors in a 3×3 bivariate scale can not have sufficient contrast in every possible pairing to satisfy even a low APCA 30 contrast level. This is illustrated in Figure 4.12, where a majority of the possible color pairs from the palette in the early palette tool seen in Figure 4.3 can be seen to fail the set contrast requirement. Based on this observation I determined that it is irrelevant to analyze each individual color scale in this mode.

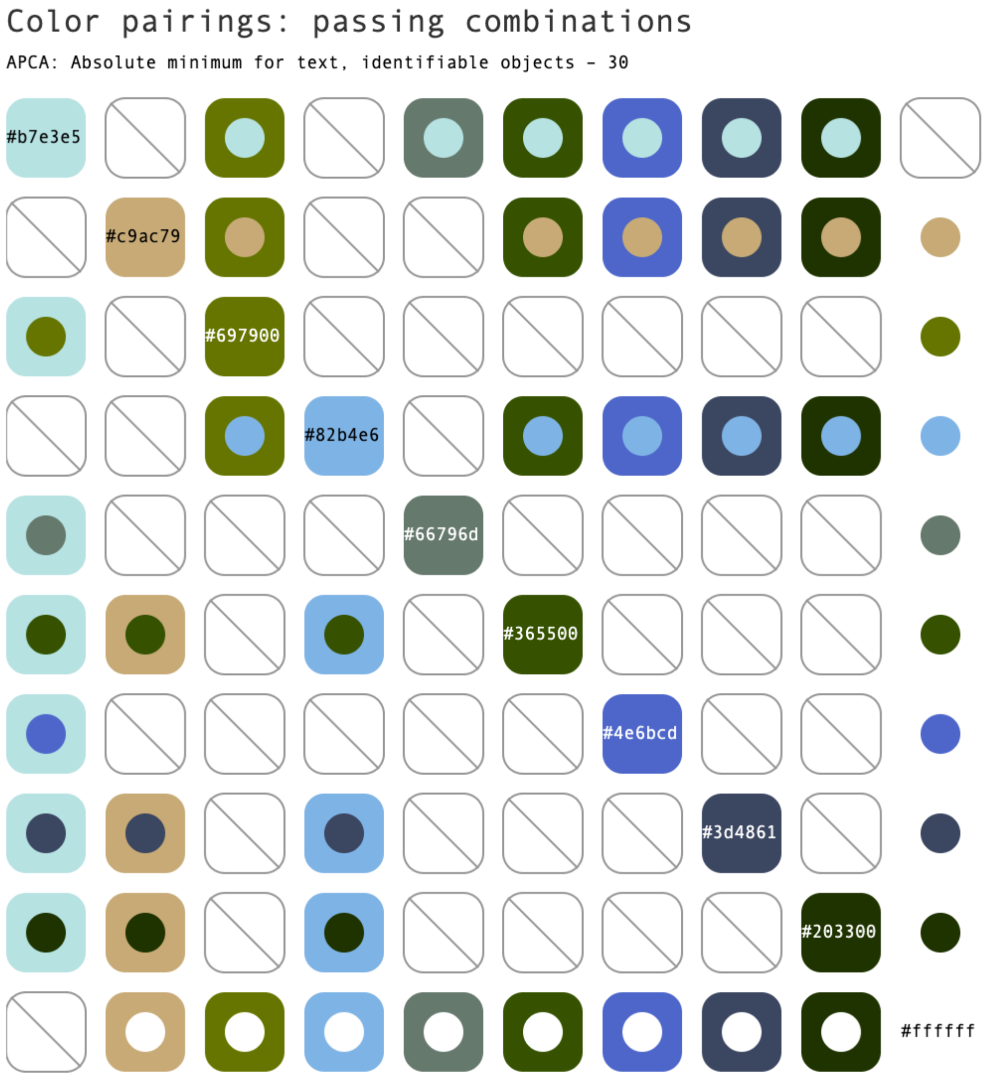


Figure 4.12. Cross-comparison of the early trial bivariate color scale shown in Fig. 4.3 with white added as background color. Each color is a swatch along the diagonal with corresponding hex code. The cut-off is APCA 30, only color pairs above this contrast level are shown.

Colors can be provided with lighter or darker outlines as necessary to ensure differences in contrast between adjacent areas, since any color in the palette can appear next to any other color (including itself) on a choropleth map. A simple way to systematically generate outlines that separates areas is to check whether a color has sufficient contrast to the background or not. If not, the stroke color will be darkened from the original color until a minimum contrast level is exceeded between it and the original color[48]. If the color is distinct from the background, the stroke will instead be brightened until it is sufficiently different from the input color itself. The strokes may also be rendered in grey using the same principle. Figure 4.13. provides a comparison of these methods applied to the same palette.

48 for instance 3:1, as specified in WCAG 2.1

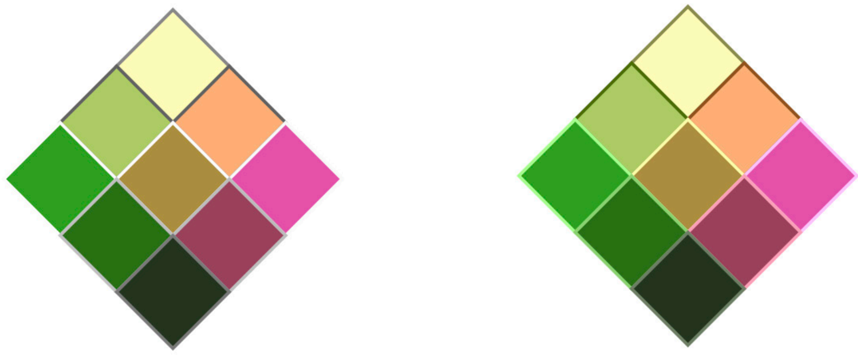


Figure 4.13. A 3×3 bivariate scale created with multiply mode, where a stroke is applied to each color. Left with desaturated (grey) strokes, right with strokes lightened or darkened from original color. Because the strokes are thin, the difference is not pronounced.

A drawback of having multicolored strokes is that the proper stroke ordering presents a problem: strokes of different colors will overlap depending on drawing order. A topological rendering could in principle be used to identify unique boundaries between regions and color them with intermediary hues[49]. This would lead to a combinatorial explosion of unique stroke colors becoming impractical in palette design. While this approach could be implemented in a map visualization itself, it is not obvious that the result would be an improvement over simpler approaches.

The colors in a bivariate map must also be clearly discernible as different hues. In the tool this is quantified using the Delta E value (see Chapter 2: Contrast and separation of colors, p. 34). Delta E is calculated with Chroma.js and the default minimum value of 11 is used, same as in the Leonardo tool (Baldwin, 2022). A higher minimum value of 22 was tested[50], but eventually found to be very hard to satisfy for most palettes — Figure 4.14 on the following page shows a palette where very high color contrast was attempted while still keeping the lightness differences uniform.

49 see *Command-Line Cartography, Part 3* by Bostock (2017) for an example of using topological analysis to identify and render internal borders separately. medium.com/@mbostock/command-line-cartography-part-3-1158e4c55a1e

50 The version of Chroma.js originally used in the Bivariate hue blender had an error in the default Delta E calculation resulting in too high values throughout. This was corrected by updating to 2.4.0.

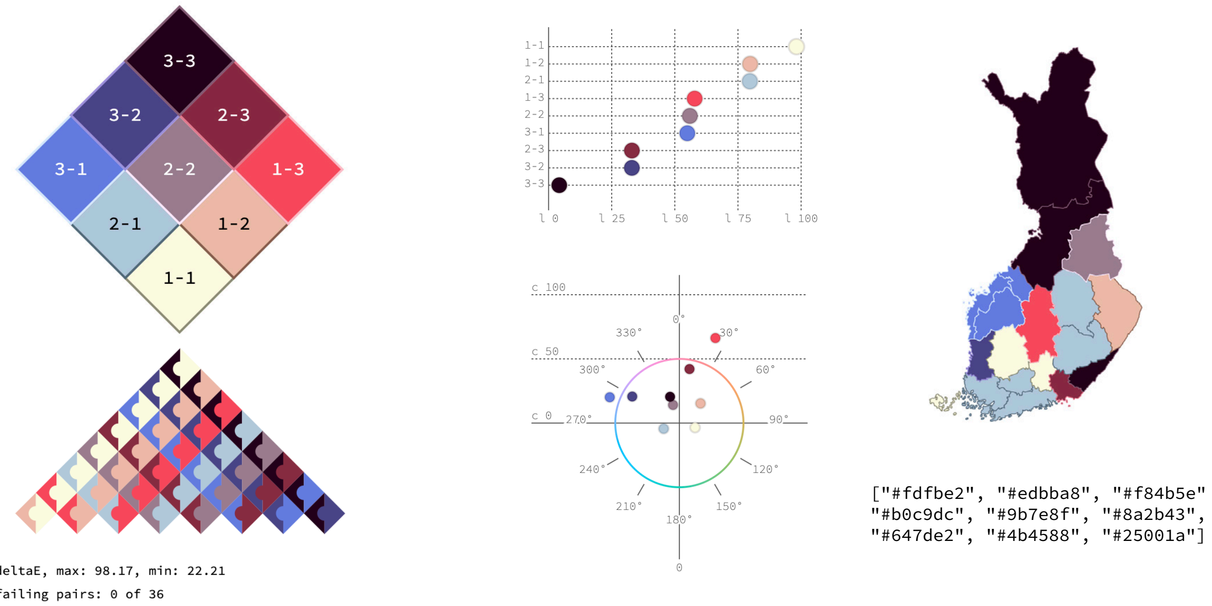


Figure 4.14. Custom palette demonstrating an attempt to maximize color differences in Delta E while attempting to keep lightness differences uniform. The lowest Delta E is 22.21 between classes 3-1 and 2-2.

Calculated Delta E values do not necessarily in all cases correspond fully to experienced visual differences. In Figure 4.15 below the passing pair of colors (2-1 and 1-2, bottom right) appears equally or even less different than the example failing pair (2-2 and 1-3). This may be an issue with color rendering, monitor fidelity or color perception — or a combination of all of these factors. There are also some known issues in the Delta E formula for certain color combinations (Schuessler, 2019).

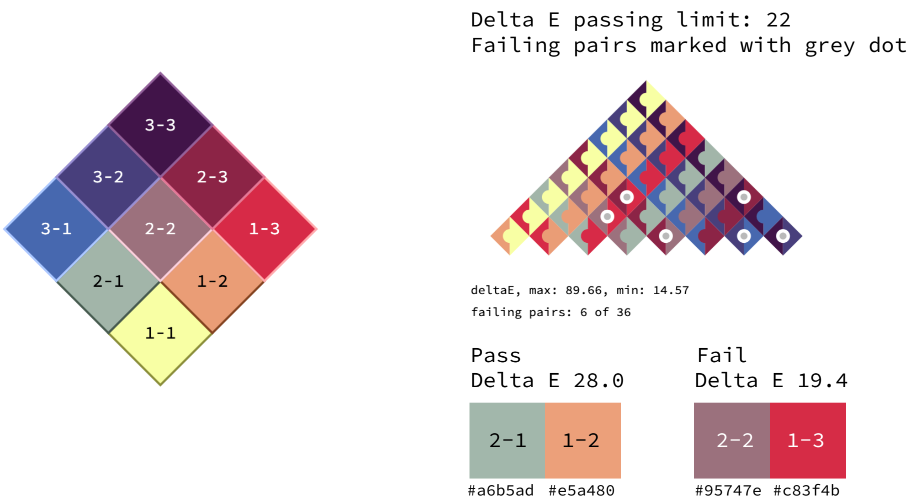


Figure 4.15. Examples of color pairs where Delta E values seem to contradict perceived differences. The combination passing the chosen Delta E limit of 22 appears less distinct than the darker and more saturated combination that fails it.

5. Results and analysis

In this chapter, the existing palettes are analyzed, and the new palettes created as part of this thesis are described. The application of the palettes and the pattern to maps with real-world data is also presented. The created maps are then discussed and subjectively assessed.

5.1 Summary analysis of existing palettes

By inputting the seven bivariate color palettes listed in Chapter 3 in the Bivariate hue blender it was possible to assess and compare their characteristics.

A number of observations concerning the palettes can be made from the charts in Figure 5.1. Most of the palettes have a Delta E of higher than 11 for all their color combinations. Palettes 1. and 2. both have one pair under 11 each. Palette 3 (Dark Cyan) is worst in this regard, with two failing color pairs and some further combinations that pass based on value but appear hard to discern reliably. It also has the lowest average color difference of the set.

Palette 6. (sequential/sequential 1994a by Brewer) has the biggest average color differences. Unlike the others, it uses black instead of a blended color for the high×high 3-3 class. The second-highest Delta E average is for Palette 7., which is also by Brewer.

Palette 5. (Dark Violet) is notable for having a darker starting color than the other palettes, but its average color difference is still roughly similar to the others.

From the lightness charts it can be observed that the majority of the palettes have quite pronounced differences in lightness between colors on the same level. In Palette 1. (Gray Pink) the color 1-3 is very close in lightness to 3-2. Palettes 5. and 6. have the strongest segmentation by lightness levels, so that the colors form groups readily apparent from the charts. The black swatch in 6. breaks the uniformity of the lightness scale and also gives this palette the by far biggest lightness range. Palette 4 (Brown) has the narrowest range, with a difference in lightness of somewhat over 50 between the darkest and the lightest swatch.

In the hue/chroma charts Palette 6. stands out with a bright yellow 3-1 swatch that is considerably more saturated (high chroma value) than any other color in the palette. This gives the palette an unequal appearance. Palette 7. also has highly saturated colors for high Y values (3-1, 3-2, 3-3). The other palettes are more uniform in hue and chroma, but many of them have some inequalities in the chroma of opposing X–Y color pairs. Palettes 3. and 5. have the most symmetrical chroma and hue distributions.

Out of the seven palettes, four pass the color vision deficiency check if using a cutoff value of 6 for Delta E as suggested by Nowosad (2020) — Palettes 3, 4, 5 and 6. Palette 5 (Dark Violet) is an addition to Nowosad's list.

5.1.1 Creation of additional color palettes

Using the color scale tool, I created three additional palettes, shown in Figure 5.2. The design goals were that the palettes should have mostly uniform lightness across the Plus diagonal (i.e., the Y series and X series would have

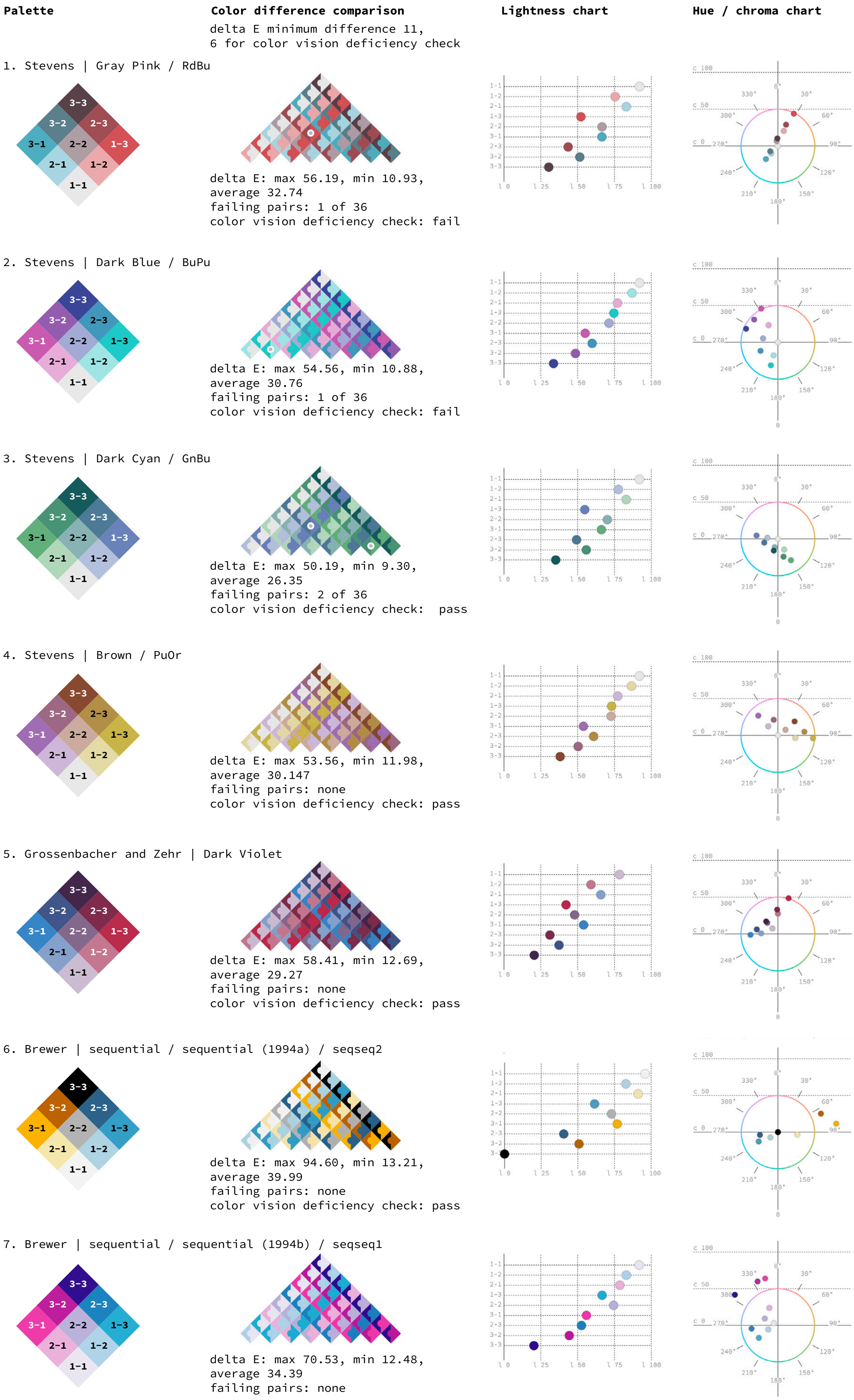


Figure 5.1. Illustration of the seven discussed color palettes, visualizing color difference comparison, lightness charts and hue/chroma chart for the different palette color.

corresponding lightness), color differences over 11 Delta E for all the combinations, and that each palette passes the colorblind check. The three palettes are:

1. *Orange Cyan*: Using Palette 6. (sequential/sequential 1994a by Brewer) as a base a new palette was made with the goal to create a visually similar palette with more uniform color distribution. The new palette has somewhat less saturated yellows and more saturated blues/cyans. The darkest hue is adjusted to a dark brown rather than complete black.
2. *Blue Red high saturation*: The primary goal of this palette was to have a high minimum Delta E but still keep the colors mostly uniform in lightness by level and evenly distributed in chroma. This palette uses a light yellow for the low×low category.
3. *Green Violet*: A new palette that is loosely based on Palette 3. (Stevens' Dark Cyan) but more saturated. Simulating color vision deficiencies has very little impact on color separations in this palette.

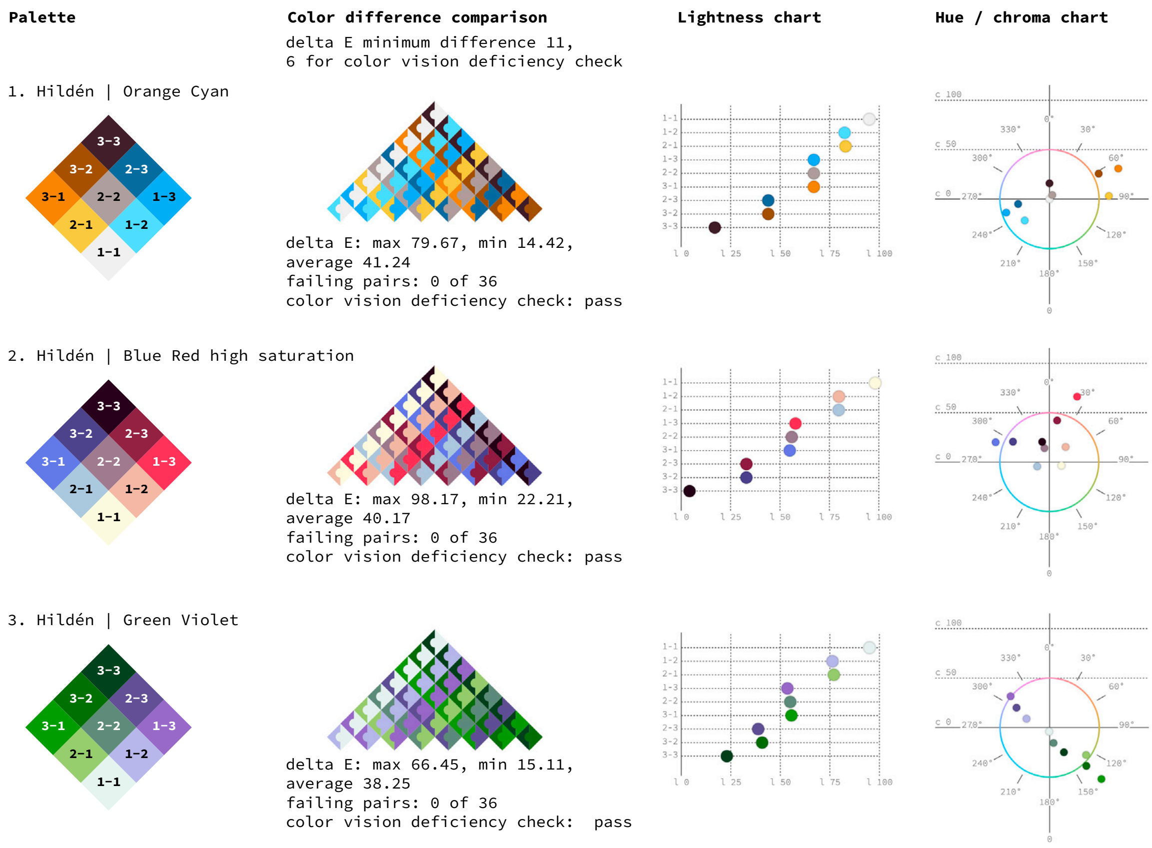


Figure 5.2. Table of the three new color palettes, with corresponding color difference comparisons, lightness charts and hue/chroma charts.

5.2 Creation of texture palette

A texture for the 3×3 combinations required for bivariate maps was created in Adobe Illustrator based on roughly uniform and additive visual differences, so that the patterns corresponding to higher values appear denser and darker. The different patterns are further designed have obvious differences in orientation and contrast. The intention of the design was to create a texture that results in a bivariate palette with integral dimensions and a strong conjunction on the diagonal Plus axis — the Y and X textures therefore have similar progressions in lightness and density. Instead of using unique textures for each of the 9 possible combinations two series of textures corresponding to values low–medium–high are designed in such a way that the intersecting combination textures are formed by interlacing, i.e., overlapping, as shown in Figure 5.3. For the low categories no texture is used. The Y textures consists of vertical dashed lines for medium values and dashed lines with heavier stroke rotated 45° for high values. The X textures are the same but rotated 90°. The textures created by interlacing X and Y are indicated with a cyan background.

To ensure good overlapping, the repeat remains the same for all textures, so changes in spatial frequency depend on the size of the repeating elements alone. This early test was created using the line dash property in Illustrator, which led to some rendering issues in rasterized exports.

Adobe Illustrator cannot create textures ("pattern swatches") that contain other textures, so each texture was constructed manually as 8 separate pattern swatches using a visual template. Although this gives improved

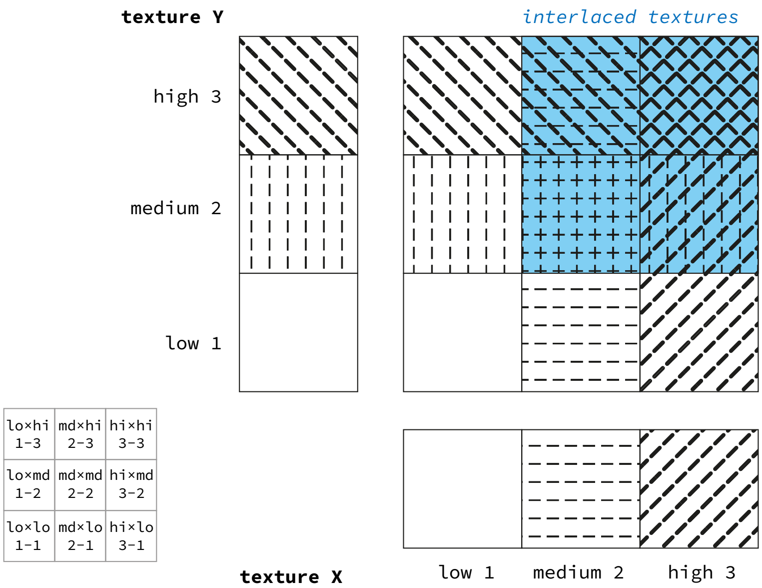


Figure 5.3. Early test of bivariate "lacy" texture construction; interlaced textures highlighted in blue.

control over the appearance of each texture, it makes the design process more painstaking. The textures have to be manually positioned after creation to ensure that the interlacing and tiling is uniform. Here a simple 2×2 square grid was used to draw the textures, which were then finalized within the Pattern Options view as shown in Fig 5.4.

texture construction template and Adobel Illustrator™ pattern tool

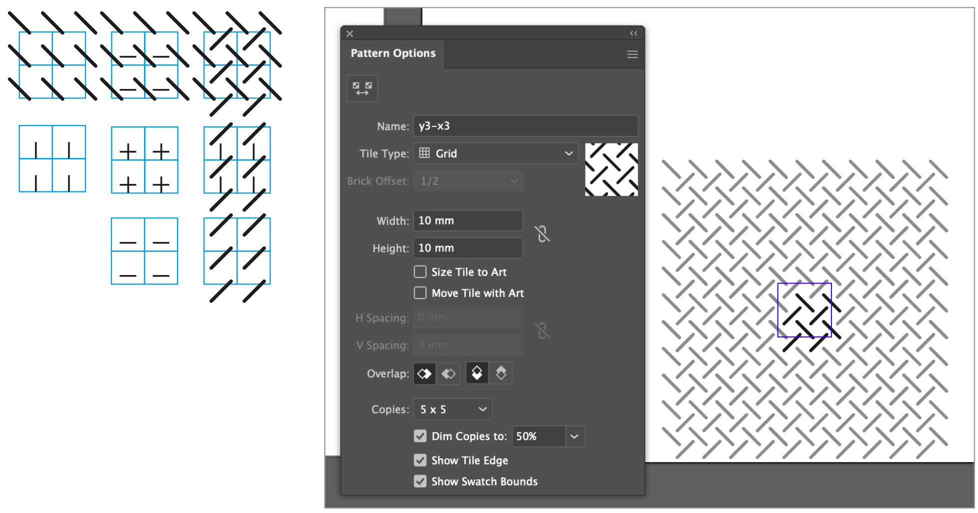


Figure 5.4. Texture construction template (left) and Pattern Options in Adobe Illustrator (right).

The improved texture demonstrated in Figure 5.5 was created using manually defined 10×10 mm grid units with separate stroked path segments that renders more reliably than the earlier example using dashed strokes shown in Figure 5.3. A dedicated interactive tool for creating these texture palettes could improve the process significantly and allow greater flexibility.

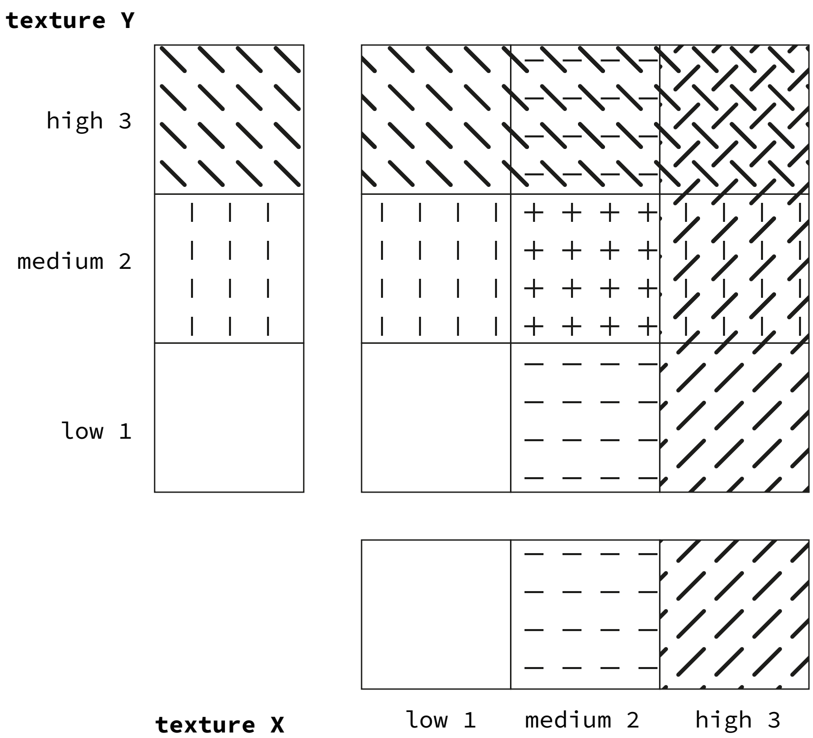


Figure 5.5. Improved example of “lacy” bivariate texture.

5.2.1 Hybrid palettes combining texture and color

By adding the textures to the bivariate palettes it is possible to fulfill the WCAG2.1 Success Criterion 1.4.1 “Color is not used as the only visual means of conveying information”[51] *and* create colors that are more easily identified from the map legend, despite the contrast ratios between multiple palette hues being lower than 3:1. The textures could either use a blending mode or the distinctive stroke colors calculated in the bivariate matrix tool. Figure 5.6 shows these differences. The first row shows the pattern superimposed on each of the three new palettes using the overlay blending mode. The second row shows the patterns using outline colors. The third row shows the patterns in outline colors, but with the contrat reduced by applying 50% opacity. The spatial frequency (visual density of the pattern) can be adjusted by scaling the textures, which can be done without resizing shapes by toggling the option *Transform pattern only* and inputting a value in the Transform palette in Illustrator.

A practical issue encountered was that Adobe Illustrator lacks the option to directly change colors of textures by applying a color. Using the “Recolor artwork” feature texture colors can be edited somewhat more easily, but this is still a multi-step manual process, where each swatch has to be manually edited.

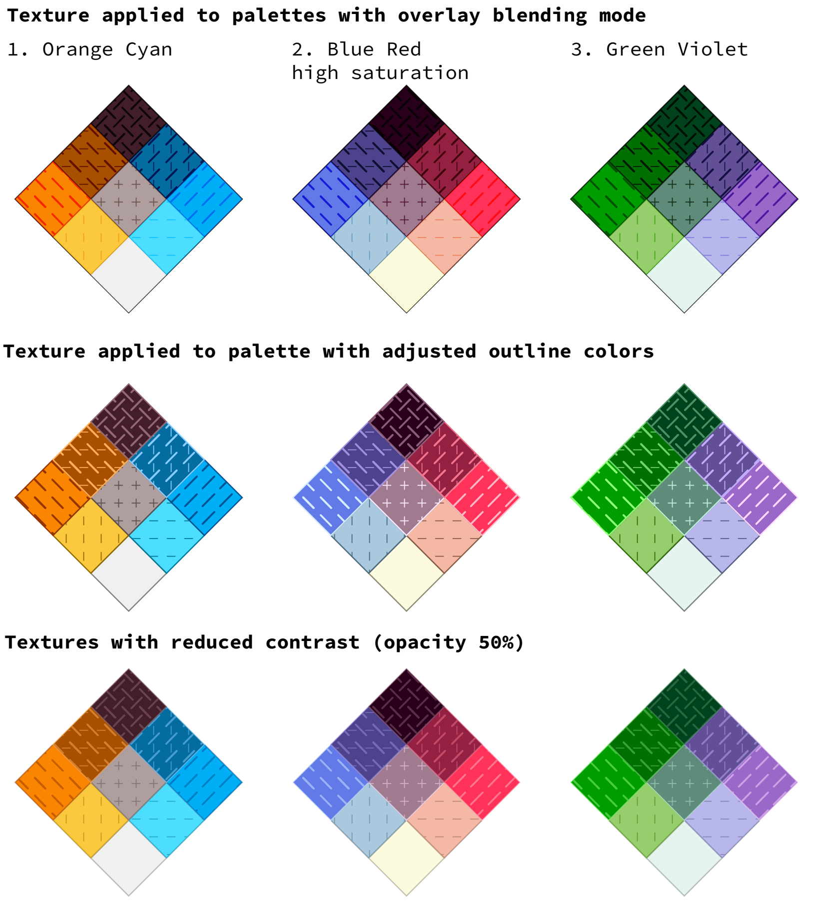


Figure 5.6. Three model palettes combined with the patterns in different ways. By reducing the contrast between pattern and color visual dazzle can be decreased.

51 Understanding Success Criterion 1.4.1: Use of Color www.w3.org/WAI/WCAG21/Understanding/use-of-color.html

5.3 Creation and assessment of example bivariate maps

The maps were created with Matplotlib and geopandas as vector output and then edited in Adobe Illustrator by manually applying textures to regions according to the scheme outlined above. The legends were rotated and relabelled in Illustrator. Since the function of the individual maps is to demonstrate the coloring schemes, therefore, not all the map details were included. A publication-ready map would also include bin limits and other explanatory content.

Figure 5.7. shows the three model palettes (1. Orange Cyan, 2. Blue Red high saturation and 3. Green Violet) applied to bivariate choropleth maps showing data for the 19 regions of Finland. Maps for Data pair 1. (first row) shows urbanization compared to share of elderly population. Maps for Data pair 2. (second row) shows share of children (% of population) compared to cost of healthcare and social services.

The different categories are fairly distinguishable for both Data pairs 1. and 2. and all three palettes in Fig. 5.7, making elementary tasks easy. Even the cases where only a single region belongs to a category are identifiable in both data pairs. Due to the small number of geographical units intermediate task are not as salient, but it is possible to notice certain regional aspects such as the four regions of Ostrobothnia (along the Northwest coast) all falling in the high X category (high operating costs for social and health care activities) for Data pair 2 (lower row).

For both data pairs some clear global patterns can be distinguished (such as the low share of children in the East in Data pair 2.), even though the Finnish regions are so large that those patterns by necessity are somewhat ambiguous. Due to the interrelated colors, it also appears possible to visually group all high X separately from all high Y values, thus satisfying the concept of three interrelated but somewhat independent axes of appraisal. The high×high and low×low categories stand out visually.

Semantically it can be noted that especially the Blue Violet scheme can be read as interpretative: the saturated red hue indicating a large share of children in data pair 2 could appear as an unintended negative association. It would be a convenient addition to bivariate mapping tools if the palettes could be easily mirrored across the Plus axis.

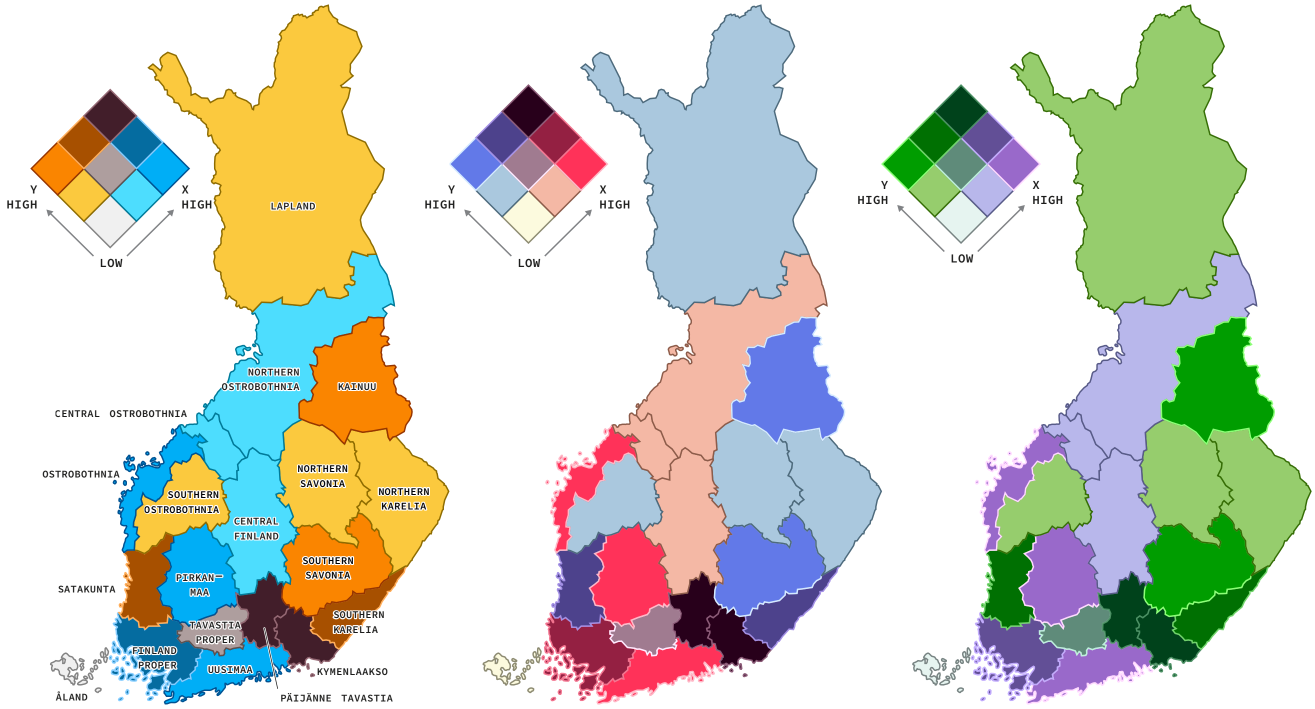
Figure 5.8. on p. 124 is the same as Figure 5.7, but with the texture applied at 50% opacity for each map. The textures appear relatively unobtrusive and could conceivably aid in disambiguating the categories. A critical question is the overall spatial frequency of the texture, which ultimately is dependent on the intended publication size and the minimum area of the mapped regions. Any practical tool for applying patterns should therefore allow for a flexible scaling of the patterns and perhaps interactive density adjustment.

Bivariate maps with color scales: Regional maps

1. Orange Cyan
2. Blue Red high saturation
3. Green Violet

Data pair 1: Urbanization versus share of elderly

X: Degree of urbanisation, %, 2018
Y: Share of persons aged over 64 of the population, %, 2018



Data pair 2: Share of children versus cost of healthcare and social services

X: Share of persons aged under 15 of the population, %, 2018
Y: Social and health care activities, total, operating net costs, EUR per capita, 2018

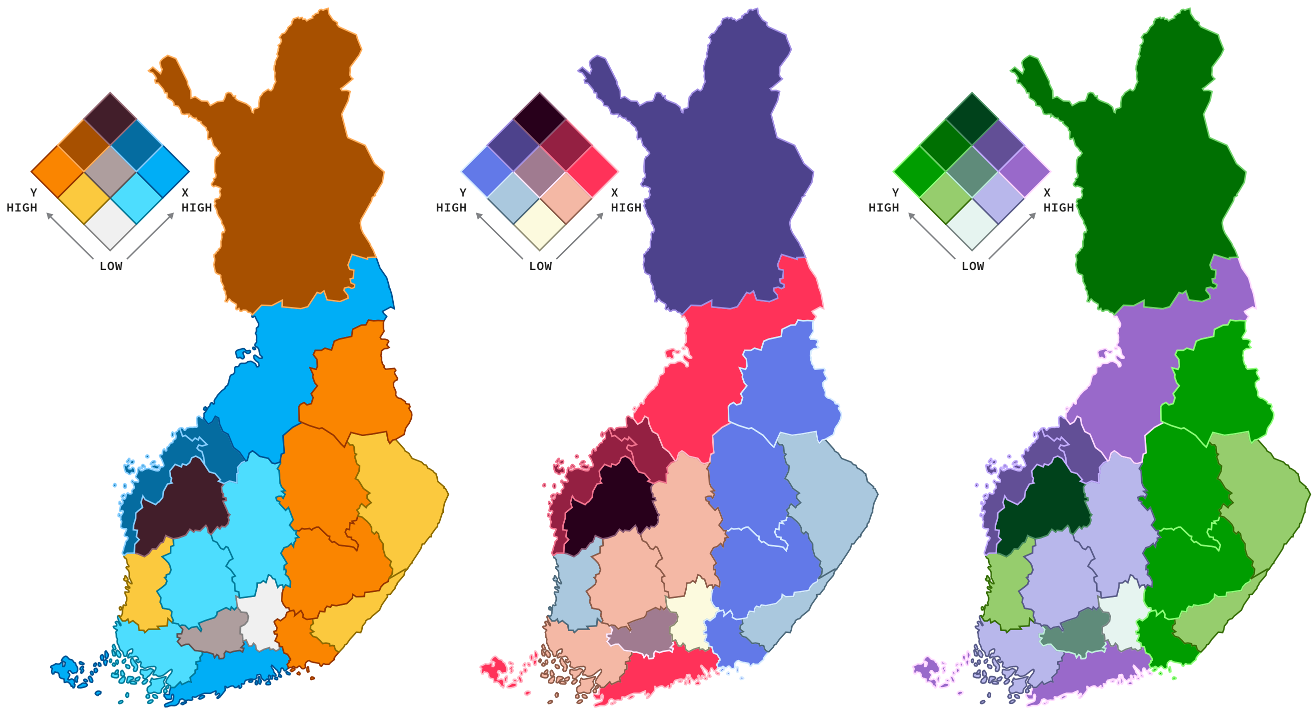
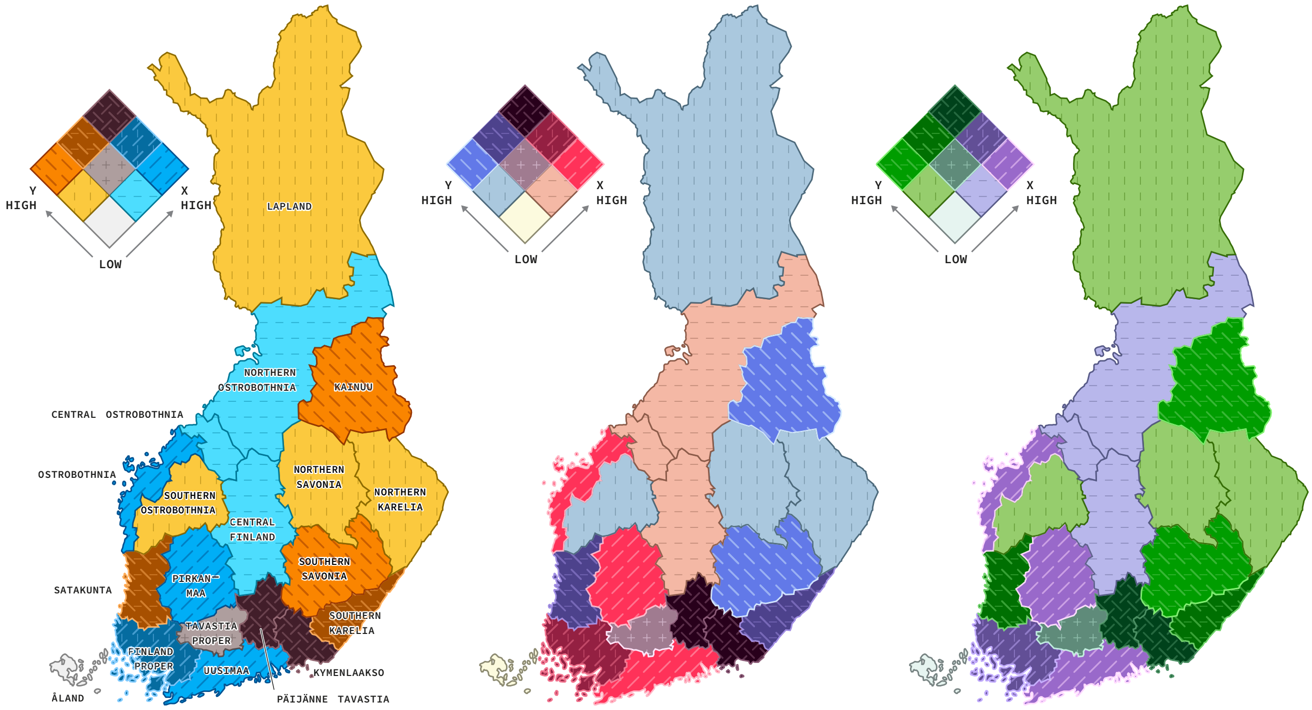


Figure 5.7. Color alone: bivariate choropleth maps of regional Data pairs 1. and 2 with the three model palettes applied. Regions are labelled for reference purposes in the first map.

Bivariate maps with color scales and textures: Regional maps

1. Orange Cyan
2. Blue Red high saturation
3. Green Violet

Data pair 1: Urbanization versus share of elderly
X: Degree of urbanisation, %, 2018
Y: Share of persons aged over 64 of the population, %, 2018



Data pair 2: Share of children versus cost of healthcare and social services
X: Share of persons aged under 15 of the population, %, 2018
Y: Social and health care activities, total, operating net costs, EUR per capita, 2018

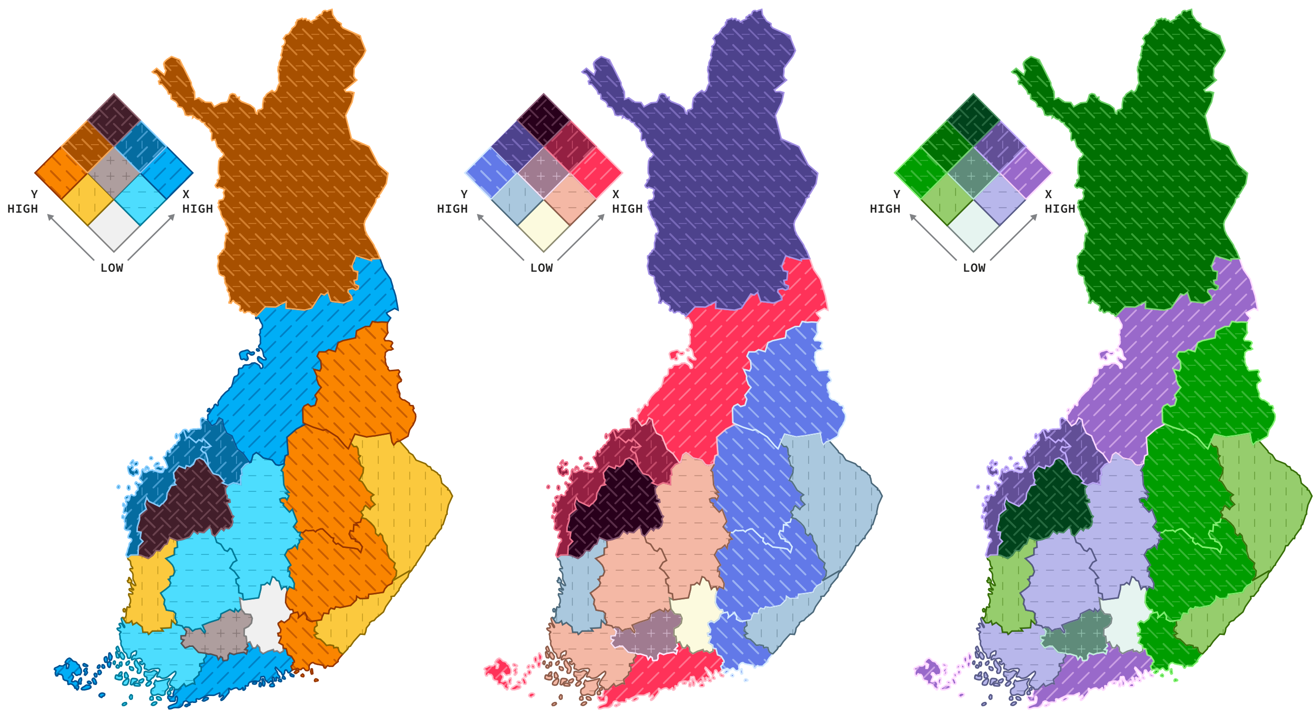
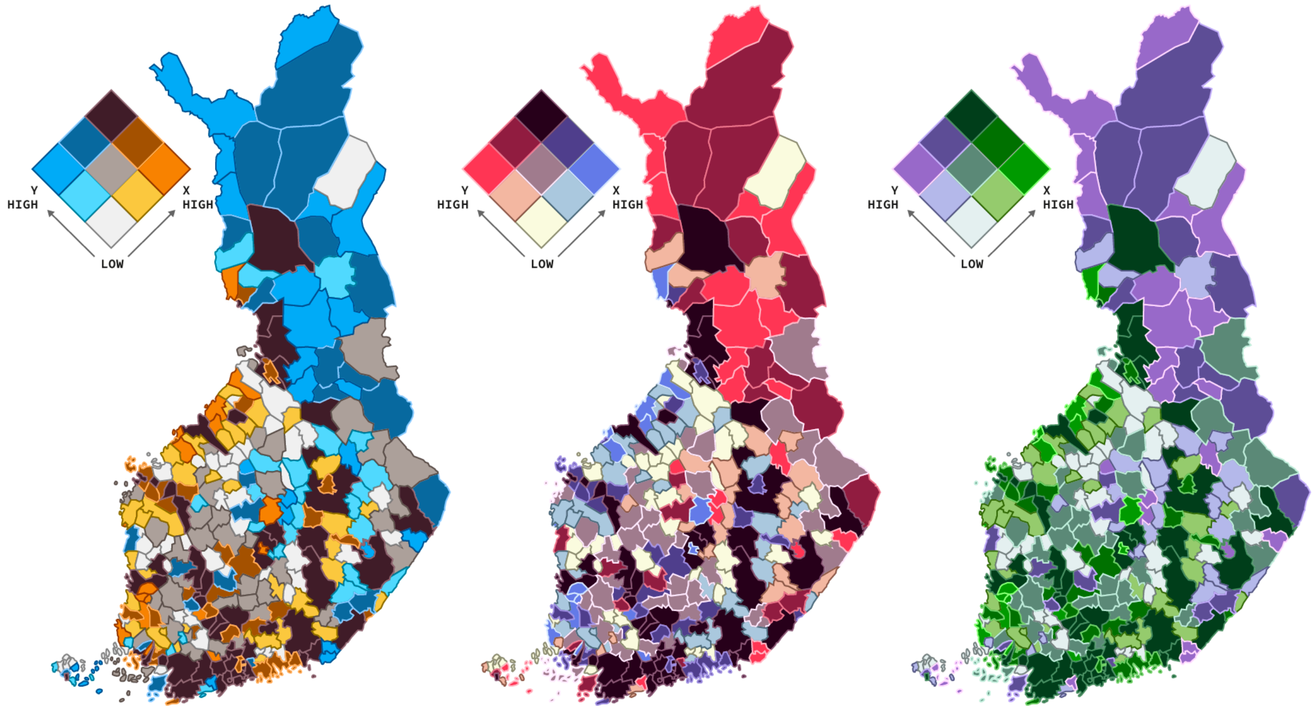


Figure 5.8. Color and texture palettes, with texture added at 50% opacity.

Bivariate maps with color scales: Municipal maps

1. Orange Cyan
2. Blue Red high saturation
3. Green Violet

Data pair 1: Urbanisation versus workplaces in service
X: Degree of urbanisation, %, 2018
Y: Share of workplaces in services, %, 2017



Data pair 2: Swedish-speakers versus foreign citizens
X: Share of Swedish-speakers of the population, %, 2018
Y: Share of foreign citizens of the population, %, 2018

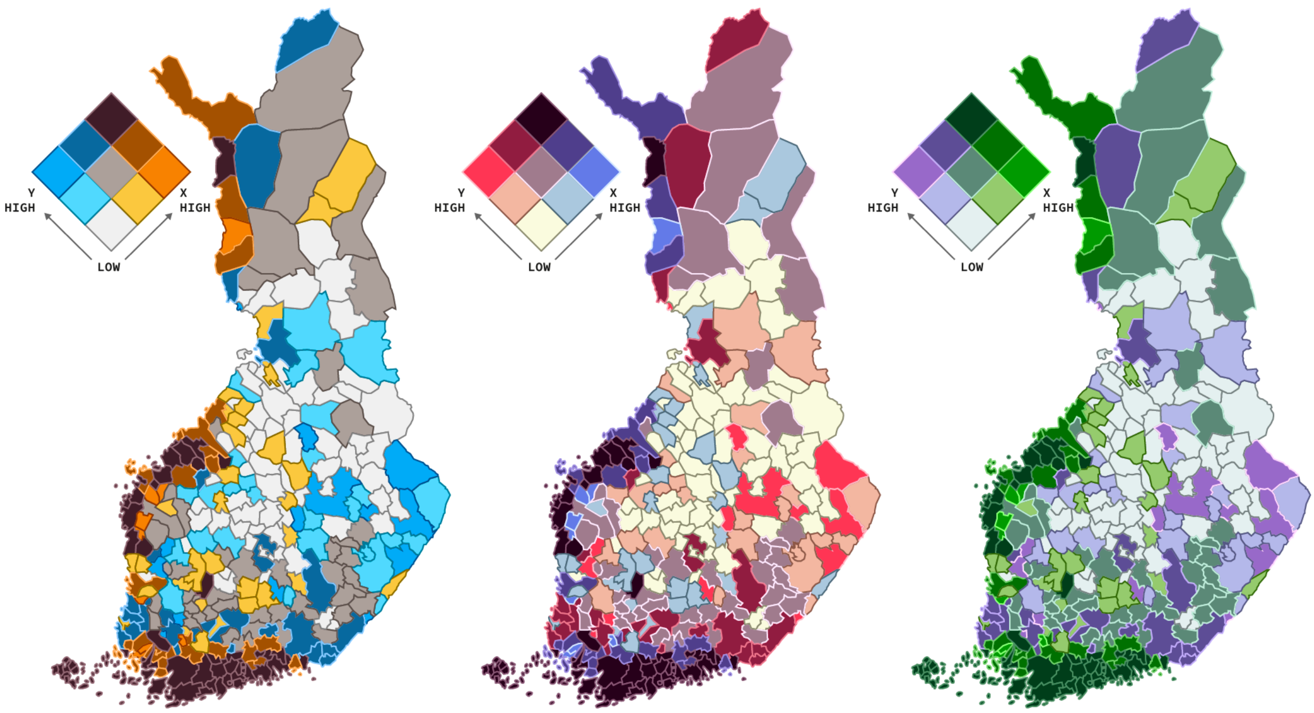


Figure 5.9. Color, municipal data: bivariate choropleth maps of municipal Data pairs 1. and 2 with the three model palettes applied.

Figure 5.9. (previous page) shows the three model palettes (1. Orange Cyan, 2. Blue Red high saturation and 3. Green Violet) applied to bivariate choropleth maps showing data for the 309 municipalities of Finland in 2018. Maps for Data pair 1. (first row) shows urbanization versus shares of workplaces in services. Maps for Data pair 2. (second row) show share of Swedish-speakers versus share of foreign citizens.

Due to the large number and big range of sizes of the mapped regions the municipal data series result in maps that are somewhat harder to interpret. Elementary tasks are particularly challenging in a non-interactive setting, since many of the individual statistical units are very small. The two saturated palettes (1. and 2.) appear more visually restless, but individual units are easier to identify. Palette 1. (Orange Cyan) would seem to have the most effective separation of colors. Palette 2. shows that the overall dark and saturated appearance leads to poorer color separation between the three darkest categories for small areas. This is evident especially when comparing the regions on the Southern Coast and Lower West Coast in Data pair 2. between Palettes 1. and 2. In Palette 3. the middle hue may actually be easier to distinguish, if it were slightly lighter.

Data pair 1. (Fig. 5.9. upper row, urbanization versus workplaces in services) leads to a map where the Southern part of Finland has a piebald appearance. On an intermediate reading level the North and East municipalities with mostly low to medium urbanization and high share of service workplaces stand out. Many municipalities in the medium×medium category are also easy to notice. The Northwest coastal region (Bothnian coast) has some high urban vs. low service areas while municipalities with larger cities tend to fall in the expected high×high category.

For Data pair 2. (Fig. 5.9. lower row, Swedish speakers versus foreign citizens) the low×low inland municipalities is the global standout feature. globally stand out. Some low Swedish and high foreign citizen municipalities in the East and Central parts are easily noticed. Concerning this data, it should be reminded that the distribution is very skewed in both series, and hence, not necessarily well suited to the applied classification here. The medium category for both data series consists of municipalities with a very low proportion of the respective groups. A more effective map for this data may be one with just two categories per series (low–high) or alternatively using manually or algorithmically adapted class divisions instead of quantiles (Koponen and Hildén, 2019, pp. 97, 99).

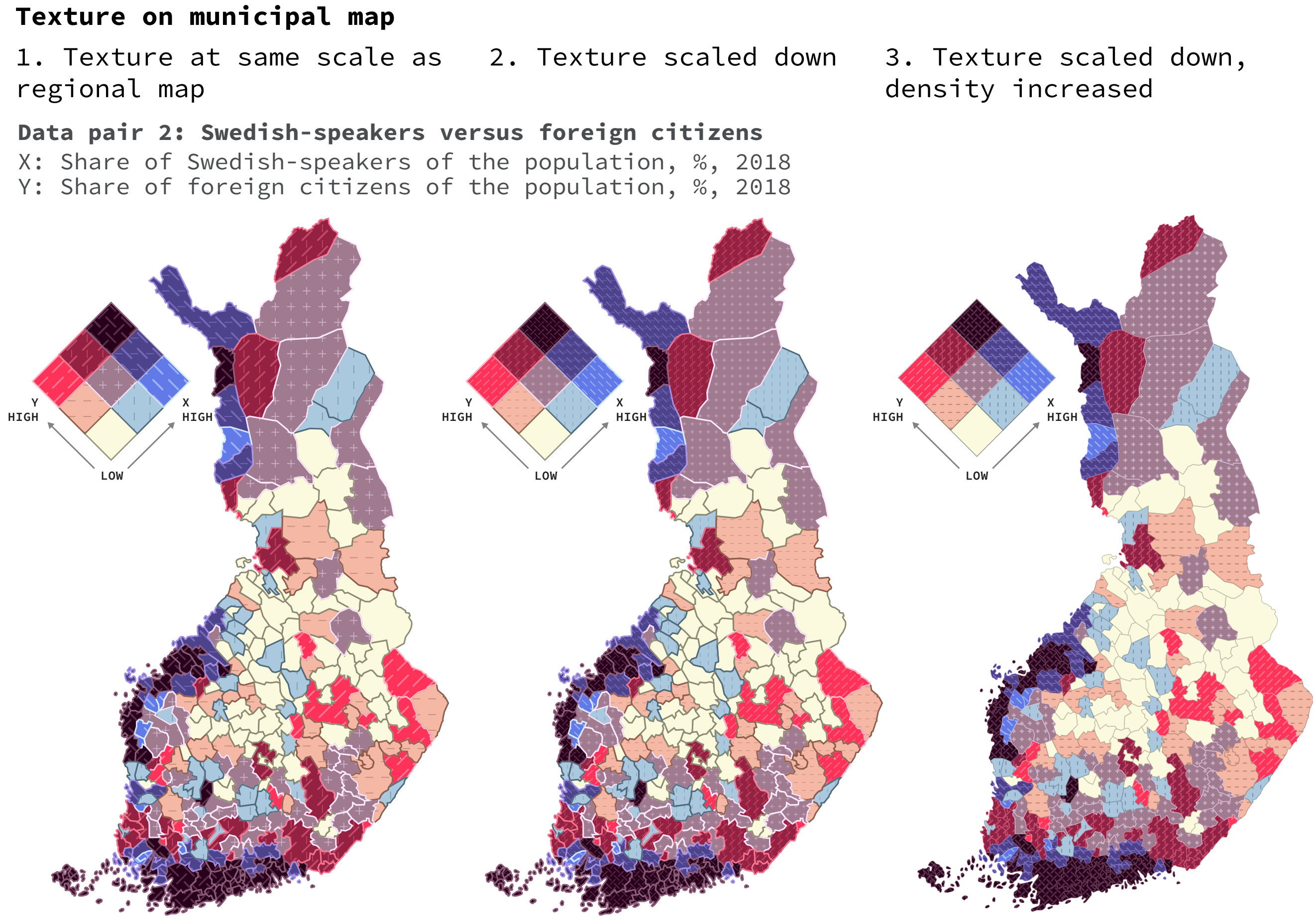


Figure 5.10. Texture combined with Red Violet color scale at different sizes.

Figure 5.10 shows Data pair 2. with the Red Violet palette and a supporting texture in three variations:

1. The same texture as on the regional maps in Figure 5.8.,
2. the texture with reduced scale, and
3. the texture with reduced scale but increased density (thicker details).

Applying the pattern designs to the municipal map is more fraught, as the fine details (high spatial frequency) of the municipal borders interacts with the textures and creates a possibly confusing visual effect. This is particularly obvious in variation 1. This effect can somewhat be alleviated by scaling down the texture and increasing the density, but the risk is that the map becomes visually restless. Textures are likely to require much adjustment to find a satisfying appearance for a particular map.

Two example Dorling cartograms (Figures 5.11 and 5.12) with areas scaled by population sizes were created with geopandas using default parameters in code by Daniel Lewis and collaborators (Lewis, 2021). These are not visually ideal examples, but serve to demonstrate that a bivariate color scheme can be applied to a Dorling cartogram without the result necessarily

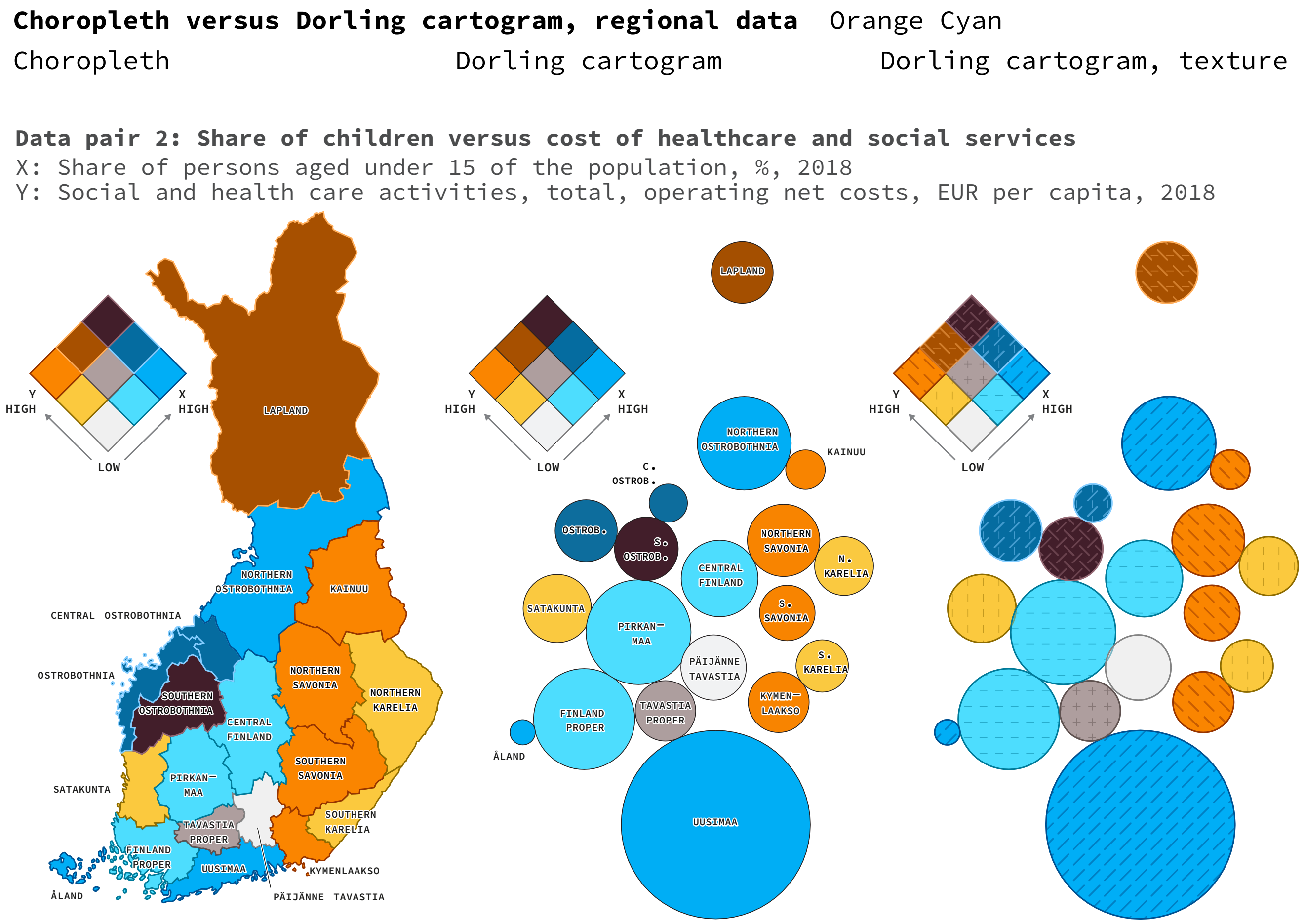


Figure 5.11. Dorling population cartogram with bivariate regional Data pair 2. visualized using color scale, with and without pattern, choropleth included for comparison.

being harder to interpret than the choropleth version, even though an additional third data dimension is introduced. Figure 5.11 shows the choropleth map for regional Data pair 2., a Dorling cartogram of the same data, and a Dorling cartogram with textures. All use the Orange Cyan color scale.

A main area of improvement for the cartograms concerns the relative sizes and positions of the circles. By reducing the maximum size, the positions can be made to fall closer to the actual centroids of the regions. As Nusrat *et al.* (2018) note, some of the issues inherent with cartograms can be alleviated by including more labelling. A scale for sizes should also be incorporated, despite this being secondary information in this case.

Figure 5.12 (p. 129) contrasts the municipal choropleth map of Data pair 1. (Urbanization versus workplaces in service) with a Dorling cartogram of the same data, both using the Orange Cyan color scale. It is interesting to note how scaling the municipalities by population significantly changes the emphazis of the visualization. What could be guessed at from the choropleth with some background knowledge (that municipalities with large populations mostly are urbanized and have high share of service workplaces) is made immediately obvious: every municipality with high population is in the high×high class. The Dorling cartogram makes elementary and intermediate

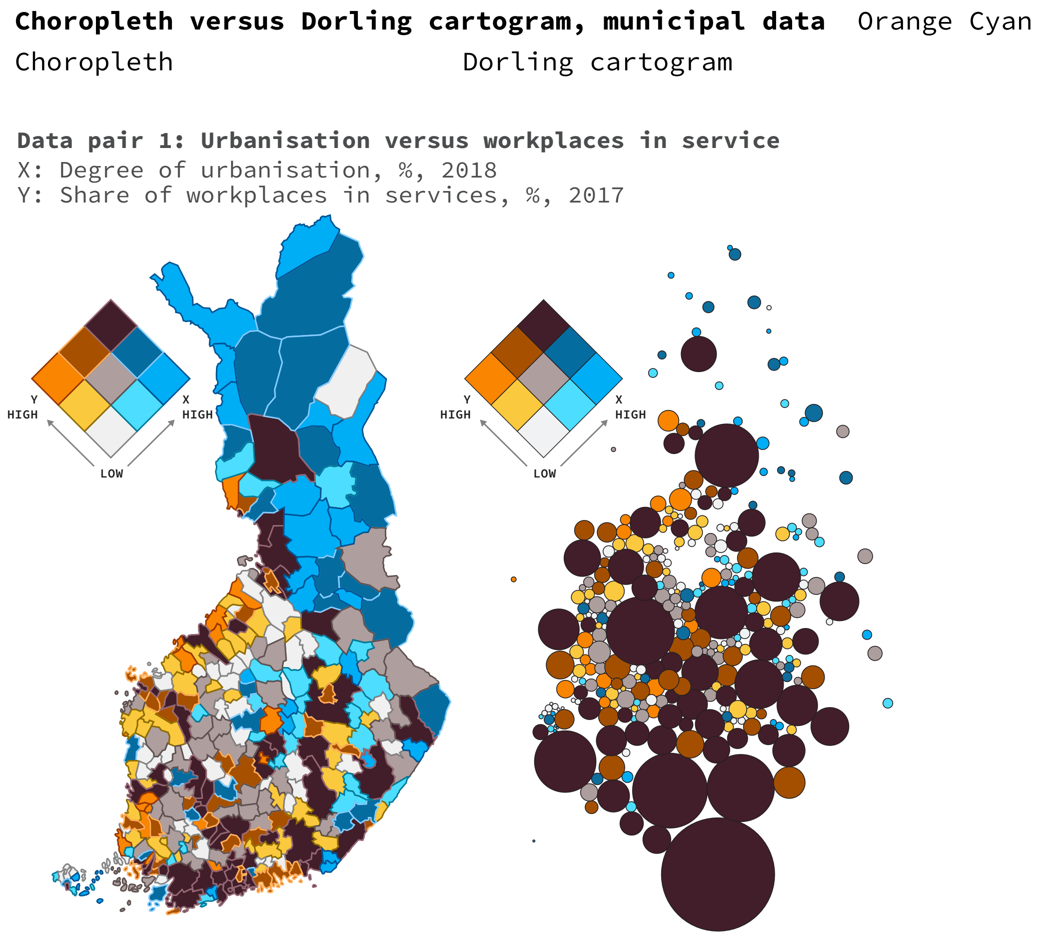


Figure 5.12. Bivariate Dorling cartogram for municipalities.

tasks easier for regions with large population. It is also very apparent that the other combinations of the X and Y values occur mainly in units with much smaller populations but answering detailed intermediate or global questions about these becomes challenging. Because of the de-emphasizing of sparsely populated municipalities in the Dorling cartogram the global task of attending to low or medium urban and high service units is considerably harder than in the choropleth map.

Calculating the layout in Figure 5.12 took 19 m. 54 sec. and the placement of the circles is still unsatisfactory, with some near Central Finland overlapping and others being located far from the main shape. Using alternative software or better optimized code to create Dorling cartograms would offer better control over this process. Nusrat *et al.* (2018) employed custom cartogram software using a force-directed layout, where the balance between topology (whether mapped areas are situated next to their actual neighbors), and locality (the geographical center points of the areas) can be interactively adjusted by the viewer.

6. Conclusion

Here I answer the research questions posed in Chapter 1. I discuss what was learned from building the Bivariate hue blender tool and how it might be improved, limitations to the work and offer some sprawling suggestions for further research.

6.1 Answering the research questions

The first research question posed — whether bivariate choropleth maps are currently recognized as an effective visualization type — can at least tentatively be answered affirmatively. The second question concerning what benefits they are considered to have over other bivariate maps is harder to gauge. Bivariate choropleth maps are not necessarily the only or even most effective bivariate maps, but clearly there is support for them being functional as long as they use carefully designed color scales and usually have no more than 3×3 classes. Generally the surveyed literature and examples did confirm my preconception that color scales used in practice often exceed perceptual limits.

That Halliday (1987) even found empirical support for printed bivariate choropleth maps being readable despite testing maps with 4×4 classes is interesting and indicates that more categories in some cases may work. Notably Halliday's maps used a combination of graduated textures and color. When bivariate choropleths have been dismissed as ineffective as by Tufte (2001, p. 153), it may be that too wide-reaching conclusions were drawn about the entire visualization type from specific, flawed examples and research performed on these. Unlike for instance statistical symbol maps, the surveyed cartographical reference literature generally did not outright dismiss bivariate choropleth maps. A consensus view appears to be that they indeed can be effective in certain use cases. (In fairness it should be noted that statistical symbol maps are not universally dismissed either).

Abstract studies of the symbolization alone like Nelson (2000a) may miss important interactions and confounding effects. On the other hand, as the theory of integral and separable dimensions according to Ware (2013, p. 168) leaves open questions and bivariate visual variables are less understood than visual variables applied to one dimension in isolation, it may be that the classification of symbolizations still needs reappraising. This is hinted at by empirical results not neatly conforming to the integral/separable division, as noted by Elmer (2013).

Because bivariate maps are complex visualizations, it is challenging empirically to disentangle the effectiveness of the visualization technique from other important factors such as choice of data and areas mapped. Studies with real-world data like Olson (1981) offer support that these maps do work, but how generalizable their results are is open for interpretation.

Elmer's (2013) study found surprisingly small differences between the different types of bivariate maps studied, but one possible explanation is the relatively simple maps and questions employed in the survey. Basically it is not entirely convincing that all the surveyed types would be nearly equally effective in many different real applications. While glyph-based maps may work as well as bivariate choropleth in some cases, they have the obvious disadvantage of being more challenging to design due to issues relating to placing symbols correctly within statistical units and without overlap. They may also be sensitive to the numbers of mapped areas. Despite the fact that integral size-size symbolizations performed well in the experiments by Elmer (2013) and Nelson (2000b) this type of map has remained rare compared to bivariate choropleths. It could be speculated that the relative flexibility of bivariate choropleths also adds to their popularity — the possibility to use different color schemes allows for more variations than encodings using size.

Asymmetrical combinations using encodings like graduated symbols on colored regions are unlikely to support visual searches involving symbol combinations (such as "small circles on dark red areas") due to these being conjunction searches. For this reason their advantage compared to two separate maps of the same data series could be assumed to be limited. Looking at intersections between the bivariate series for single data points ("in this area, the value for X is large while Y is small") is of course possible. Research done on chart types other than maps also appear to suggest that symbolizations like spoke glyphs are unlikely to be effective in real-world cases. Apart from the asymmetrical value-by-alpha map (which is a special type of bivariate choropleth map) there is much less literature on how to practically apply separable and configural symbolizations and in what situations they are suitable, compared to the integral bivariate choropleths.

6.1.1 Outcomes from the practical part

The third research question — can a custom-built interactive tool improve the design of color scales for bivariate maps — is here answered subjectively, based on the four outcomes defined for the production part:

The first outcome — 1. Making the actual tool(s) that can be used to design and assess bivariate color scales — was described in detail in Chapter 4. An important learning outcome from that process was developing some understanding about how to use the Svelte framework for building web apps and combining it with D3 and Chroma.js. Here I proceed to discuss the remaining outcomes:

2. how said tool(s) facilitated color scale creation,
3. the actual maps and individual color scales that were made with the tools (the maps and color scales are also described in Ch. 4),
4. insights and ideas for improvement gained from the process of using the tool(s).

6.1.1.1 *How the Bivariate hue blender facilitates color scale creation*

The ability to directly adjust all the individual colors with a palette tool that shows the available color range and simultaneously view the result both on a preview map and as hue and lightness visualizations make the color scale design process more deliberate and controlled. The visually indicated contrast assessments also assist in avoiding the easy pitfall of bivariate color scale design where different colors are too similar. Other good color tools exist, but the Bivariate hue blender integrates the entire process so that it is possible to design and validate color scales in one place and then immediately use them.

From the results the three new palettes seem to be at least as usable as the existing palettes that were used as reference. The tool made the creation process fairly straightforward and using it to apply tweaks to pre-existing palettes was also found to be effective. The direct chart feedback and visualization in a realistic context with a map preview makes the tool, while unpolished, superior to this specific purpose compared to design software like Adobe Illustrator. Arguably the new color maps are more perceptually uniform than the studied palettes, but it may be questioned if this matters and to what degree.

Using and testing the tool gives the impression that the number of bivariate color scales that are visibly different but also functionally practical, i.e., by satisfying minimum color difference requirements and being colorblind safe is likely to be fairly limited. This raises the question of how much relevance further technical improvement to a tool for creating bivariate palettes can have. On the other hand working in a constrained design space benefits from a tool that gives detailed control over color generation in the actual use context.

6.1.1.2 *Discussion of color and texture scales*

The texture palette designed in this thesis should be considered a quick demonstration of how a sequential bivariate texture or pattern may be designed. Following the constraints described in Chapter 2. (Visually distinct textures, p. 38), different textures of better artistic quality could be created. An interesting alternative would be to use a raster-based approach which could enable blurred, low-contrast textures that may be less obtrusive particularly on maps with small regions. Lowered contrast and smooth variation would reduce the interference with the necessarily crisp boundaries of the geographical regions. Reducing the texture contrast by using transparency as I did here go some way to alleviate excessive contrast issues.

A big limitation for texture creation is the lack of practical tools and the issue of easy use of texture definitions across software implementations. Creating textures would be significantly easier with a tool that allows a hybrid approach of drawing and mathematically defining the textures and where textures can be “laced” together by design. By this I mean that a texture A×B could be created by overlaying two textures A and B in such a way that any changes made to A automatically are reflected also in the combination A×B. This effect can be simulated in Adobe Illustrator to some extent simply by overlaying objects with applied textures, but not directly converted to usable texture swatch definitions.

6.1.1.3 *Discussion of created maps*

Finding example map data turned out to pose a challenge as it was hard to know whether a particular data set would be suitable for the purpose. The lack of charting tools included in the Finnish national statistics database that would enable rapid prototype visualization of candidate data made this process more difficult than it needed to be — the Paikkatietoikkuna site contains only a small subset of all potential data sets with geographical dimensions. The contingency table visualization created for cross-checking data series was an indispensable aid. Still, the example maps created in this thesis are mostly less informative than for instance the bivariate grid map of children and population density discussed in the introduction (Figure 1.2, p. 14).

The choice to use contingency table visualizations to pre-evaluate candidate bivariate data appears to be a novel approach, although it obviously cannot be excluded that someone would have described something similar (Leonowicz's (2006) study includes a similar table, but as a part of a finished map). The advantage of using contingency tables compared to the alternative of directly generating bivariate maps from candidate data is that it immediately allows the designer to see whether certain classes are empty or contain very few data points, which is hard to do on a map if it contains more than a handful of regions. Furthermore it postpones the issue of spatial data coding — data is often not available in a format that can be linked to geographical regions without some intermediate processing.

In hindsight it might have been more straightforward to use established examples from international data, but I saw using Finnish data as having a certain novelty value as very few bivariate maps to my knowledge have been created here. It may also be that bivariate maps just in general are harder to make for Finnish data due to Finland's unequal geographical distributions of both population and municipality sizes.

6.1.1.4 *Insights from using the tool and possible improvements*

An important usability improvement to the Bivariate hue blender would be to directly integrate color vision deficiency previews, so that the palettes would not have to be evaluated separately for this. The color editing process in the Bivariate hue blender could also be made more dynamic by making the accompanying hue and lightness visualizations themselves interactive — for instance so that the lightness of a color could be modified by directly moving the chart component representing it. For this to work well there would need to be some deliberate handling of undisplayable colors, though. In the current version cycling through undisplayable colors lead to some erratic changes in hues. Allowing some properties such as overall lightness to be adjusted for all colors at once may also be useful.

Using the tool I noticed that the color creation and adjustment interface could be improved by making it "sticky," so that it would be available when scrolling down to look at the maps. But this created a new problem where the main interface covered the palette views, making it difficult to adjust individual colors while viewing the lightness charts. A better solution would involve further adjustments, like a toggle to collapse the interface or have a narrow version of it. I found it helpful to have the color picker permanently visible rather than always hidden behind a popup, but the color picker itself is a very preliminary design and options for improving it should be explored. A more general problem with the Bivariate hue blender is that it now has multiple interfaces for doing essentially the same thing (adjusting colors). A more streamlined version would probably have just one color picker that handles all the roles, maybe in a collapsed or extended mode.

Negative or inverted color schemes, while certainly having potential were left outside the scope of the tool design at this stage. They may be a useful future addition to the tool. The Bivariate hue blender likely would need further modifications than just a dark-mode background to create such color schemes effectively. It could also be extended to support the other schemes relating to the focal models of Strode *et al.* (2020) — corners and range. Minor tweaks include the support of inverting the color scheme (flipping X and Y colors).

There are a number of improvements that could be made to palette blending and the creation of intermediate colors. The parameters of the "Mix" mode are somewhat arbitrary, with just an adjustable exponent and additional multiplier constant. It could be improved by including the possibility of directly manipulating the lightness curve. The "Multiply" mode could also be modified, so that the pure colors are left unblended.

Even though it is unclear how much this would affect the visual end results there are options for improvements to the color generation itself. *Color.js* (Verou, 2022) which was significantly updated during the time of writing might offer a better model for mapping undisplayable colors to the available color range (gamut). Some easy improvements to color generation may be achieved by using CIELUV-based HCL or alternatively the polar transform OKlch of the OKlab color space, which solves issues relating to hue uniformity that especially affect blue colors in CIELAB (Ottoson, 2022). When I found that OKlab had been added to Chroma.js the Bivariate Hue blender was already at a stage where there was no time for the switch.

In hindsight creating the color tools as a stand-alone web app using Svelte may have been a less than ideal choice for future development, maintainability and distribution. Building the tools in Observable would make them easier to share and lower the threshold for modification, thanks to the design of the publishing platform, which directly enables sharing and reuse. Less time could have been spent dealing with interface components, since these can be repurposed from other notebooks. Observable offers less direct control over the design, but the design was not the main concern in creating the tools. Rewriting the existing tools in Observable may thus be worth considering.

6.2 Limitations

The discussion of the theoretical research focused on guidelines rather than being a comprehensive survey of the empirical research done on bivariate maps. The empirical studies discussed were fairly different and no systematic attempt was here made to appraise and compare their results in detail or to assess their validity.

The color tool and palettes were created and evaluated strictly in the context of on-screen displays. Thus it is easy to create palettes are impossible to use in print. This is also a limitation in much of the newer referenced empirical research, which tends to be done on on-screen displays for convenience. Older research again was done with paper maps, which then raises questions as to how well it applies to screen displays where the available color ranges are greater. The three palettes were here not evaluated according to how well if at all they may be converted to CMYK inks or other printing colors.

More subtle technical details of color reproduction and display modes were largely ignored, since I considered that the fundamental limitations of perception essentially mean that the color differences used in palette designs always need to be much larger than any eventual differences caused by display modes. For high-quality design and exacting control of color appearances color reproduction and display modes do matter, however.

The color scale tools are as they stand fairly limited and have a number of technical issues and limitations. Notably the setting of the browser hash is still broken and works unreliably. The map preview for 2×2 and 4×4 color scales also does not work yet. Interaction and interface design has not been in any way thoroughly tested with different users and it remains to be seen to what extent they are usable without additional work on the documentation.

The discussed existing color schemes were not cross-examined with the three new schemes by applying them to maps, and the latter were only evaluated together and alone for basic functionality. More work would be needed to assess whether they actually outperform or perform equally well as the existing schemes in different contexts.

I brought up the question of accessibility, but this only in the limited context of color scale design. Visualization accessibility is a much wider topic covering considerations such as text descriptions for screen readers and navigation (in interactive contexts). These important questions were not dealt with in this thesis and would require significant further work to expand upon in the context of bivariate maps.

Using specific real-world data posed some limitations on the creation of the maps: they cannot be considered a representative representation of possible cases for bivariate maps, as the data distributions mostly represent the idiosyncrasies of the data that was available and easy to find. More conclusive discussion and assessment of palette behavior under real-world conditions would need much more data with clearly different characteristics. The same holds for the regional data: end results may be very different for maps created of different countries or subdivisions. Comprehensive testing would also involve applying different and contextually suitable classification methods to the data visualized.

While I discussed them in the context of theory and past research, the practical part of this work did not deal with alternative solutions to bivariate maps. Therefore no additional contribution was made as to their practical applicability or whether they in some cases might outperform bivariate choropleth maps.

6.3 Directions for further research

While this thesis used an essentially manual process to generate a small number of color scales it would certainly be possible to use automated processing to generate novel palettes. This could be done by using iterative algorithms to optimize the Delta E distances between colors within given parameters for palettes of the desired size. Using procedural generation and/or machine learning it would be conceivable to explore the entire available bivariate palette color space and generate new palettes that satisfy set requirements, by rolling novel but sensible suggestions in the vein of projects like the anonymous *huemint.com* (2022) or from given source colors like Akveo's *Deep learning color generator* for their Eva design system[52] (2022). These tools are both oriented towards branding colors and as such not directly applicable to visualization.

The Color Crafting paper by Smart, Wu and Szafir (2019) is one example of analyzing existing palettes created by designers in order to model the subtle adjustments that humans make to palette characteristics. The model can be used to make novel quantitative color scales from given seed colors that according to the authors outperform palettes made using typical mathematical approaches, as demonstrated in a prototype tool[53]. However, to my knowledge no finished machine-learning-powered tool for generating palettes geared specifically towards quantitative data visualization colors has been published at the time of writing. From the perspective of a person with a design background such a tool should also include options to adjust the generated colors manually to be appealing to use.

Going forward there seems to be a choice between creating (and empowering the creation) of more different palettes or alternatively improving on a limited number of established bivariate palettes and doing empirical research on how they are perceived and functuion in different contexts. The

52 Available at colors.eva.design; due to the paucity of documentation, it is unclear how and in what way machine learning in fact is used in this tool.
53 Available at cu-visualab.org/ColorCrafter

enduring popularity of individual palettes like Viridis (van der Walt and Smith, 2015) or the ColorBrewer collection (Brewer, 2013) seem to indicate that there is an inherent value in using a limited set of particular, named palettes at least in data analytics and scientific publishing. A "tried and true" palette can be considered to get an advantage from familiarity as long as it is perceptually functional. It offers an easy choice when doing analytics and can be assumed to be familiar to at least specialist readers as well.

On the other hand the practical context of doing data visualizations that utilize particular brand colors leads to a need for a great number of unique palettes that also fulfill requirements of contrast and color differences. Since this process often may involve people with less experience in and knowledge of relevant literature there could be a real need for a practical tool like the Bivariate hue blender.

Perceptual uniformity in the context of bivariate maps seems to be somewhat poorly defined. It is not strictly obvious how the colors in a bivariate scale should interrelate by lightness. Particularly — should the middle (2-2) color in a 3×3 palette be approximately equal in lightness to the 3-1 and 1-3 colors, or should it be strictly in the middle between the lightest and the darkest swatch?

Preferences could be explored by user testing, but I expect that there is a certain flexibility available which allows some variation in different directions as long as the overall impression feels right to the viewer. Similar concerns apply to differences in chroma/saturation. Ultimately it might be of questionable utility to try to explore this particular detail empirically. A palette that by human inspection appears reasonably uniform and where colors can be reliably separated may just be good enough for most use cases — something suggested by the results of Olson (1981) where study participants found even maps with colorful 4×4 scales interesting. However, this does not rule out that more extensive empirical testing of bivariate color scales could surface interesting results.

A larger cross-comparison of the established bivariate color palettes might offer new insights. However, considering the relative lack of empirical research into bivariate map reading, even more important may be to study how variations in the number and sizes of statistical units affect the comprehension of these displays. To investigate this in a controlled way that produces actionable results likely should be done with just one color scale as to not introduce confounding variables. A major challenge is that variations in the distribution of the mapped phenomenon across the statistical units itself is likely to affect perception significantly — if there are clear geographical patterns evident in the data, then a bivariate map will be more comprehensible than one where the distribution appears haphazard as noted by Slocum (2014, p. 252).

Taking a wider view on color scale design, it would be relevant to introduce (more) perceptually uniform color spaces such as hue-chroma-lightness for color picking and color design work directly in design software. The extant tools and libraries are accessible to technically oriented designers and software and web developers, but cumbersome to use for visual designers. The open-source vector illustration software Inkscape shows a good example here by including a HSLuv color picking space in the 2022 version (Derriche, 2022).

As Franconeri *et al.* (2021) note there is a need for empirical research on how and in what ways the use of textures in addition to color encodings may affect the reading of charts and maps. The presumed interfering effect may be connected to particular features of the pattern such as contrast and density rather than invariant and hence possible to control or amend by thoughtful design choices. This is however hard to ascertain without empirical testing of these conditions. Such research should be carefully designed to actually allow the assessment of the texture properties rather than other contextual variations.

The questions of color scale design and textures relate to questions concerning visualization accessibility and how it can be realized in the context of bivariate maps specifically. Examining how bivariate maps can be made accessible and what toolings are needed to enable this is as a small part of the research necessary to develop more accessible visualizations. Because bivariate maps are particularly complex visualizations, the challenges in creating guidelines for accessibility features like text descriptions are formidable.

This thesis did not delve deeply into the question of bivariate choropleth legend design, but this appears to be a fruitful area for further improvement of the map type. A question to be resolved is how class limits best ought to be indicated. A related issue is the effect of classification. The direction of creating customized legends and classification systems specifically for bivariate choropleths as proposed by Eyton (1984) and furthered by Dunn (1989) may be worth revisiting in this context. A related topic is how bivariate maps work in an interactive context. It is conceivable that interactive maps where the user can switch between a bivariate visualization and one where either data series is visualized alone could be effective. Interactive legends could also enhance their usability.

The limited literature on bivariate map reading suggests the need for more empirical research on how they actually can be and are used for different tasks. Szafir et. al's (2016) paper on ensemble codings of visual information from multiple graphical objects may offer an interesting model to apply to bivariate maps. They defined four categories of visual aggregation tasks (identification, summary, segmentation, and structure estimation) and posited that different visual feature mappings will be more effective for some tasks than others. Identification tasks involve picking out outliers or particular values. Summary tasks relate to describing the visualized data as a whole (e.g. estimating a mean). Segmentation tasks concern seeing groups that form in the visualization based on a particular visual variable. In structure estimation tasks correlations or trends in the data are observed from a visualization. Of these at least the three first would appear to be directly applicable and testable on bivariate maps.

6.4 Finally

In all this thesis project has left more loose ends than clear conclusions. The process of writing this thesis and building the related applications has in itself been a good example of how the process of information design requires a complex negotiation of tools and methods and how it becomes necessary to build one's own when the existing software falls short. Luckily, this is now also easier than ever.

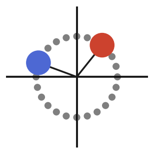
During the work process the color scale tool changed focus and appearance. In testing different palettes it became apparent that the original cross-comparison of lightness contrast (the Contrast grid mode) was relatively irrelevant for bivariate choropleth maps, since they fundamentally will not satisfy the given contrast requirements for all color combinations. I chose to retain the mode, as it has relevance for the analysis of color schemes in my visualization design work and no other existing tool has precisely similar functionality.

This has been a winding self-taught programmer's journey. At the same time the process has demonstrated how far existing visualization recommendations often are from the actual praxis of *creating* visualizations. The theoretical literature is frequently rather removed from daily chart-making practices and can be hard to approach. The spotty empirical recommendations that exist can be challenging to apply conclusively, although excellent recent efforts have been done by writers like Robert Kosara, or Franconeri *et al.* (2021) in their *The Science of Visual Data Communication: What Works* paper. Making charts is both science and art, but the blank areas on the metaphorical map are so large, that significant room for art appears to remain for the conceivable future.

Perception science also takes a long time to cross into visualization writing. Having been made aware of the problem of the pre-attentive concept during the writing process through a podcast interview with Steve Haroz (2018) I now notice how it still keeps recurring as a fact in articles discussing visualization. We who work in visualization have something to learn about the limits of our current knowledge and making too certain claims.

Another question that this work surfaced was the tension between web design and visualization. This interaction becomes particularly relevant in the context of web accessibility, which until now has been developed largely without detailed guidance for applications in visualizations. In some sense visualization is a field that keeps being reinvented in different contexts, which creates odd gaps and overlaps — one example being visualizations made in the context of journalism versus those made for scientific publications versus the traditions of cartography. Since more and more material is being published primarily for the web the practices of web design bleeds into all of this, for better or for worse. Meanwhile one can wonder what happens to visualization in the world of printed artifacts.

As an information designer without a very solid background in any particular field of science the process of negotiating tool creation and tool using inevitably involves hopscotching between different environments and methods and combining their outputs to reach a particular end result. Arguably this process may be more systematic and controlled for someone with a more sharply defined skillset — for instance doing all work in R. This thesis itself is a hybrid, written in Markdown but exported into InDesign with Pandoc for layout, with illustrations travelling through various software before reaching their final form. Using a multitude of tools and methods can sometimes confuse both author and reader. Then again it may also be an advantage when one tries to find the tool where a particular task is easiest or most familiar — instead of trying to hammer everything with the same implement. I hope this thesis might encourage someone else in building their own tools to approach information desing questions.



Appendix

Table of bivariate 3×3 palettes

Colors as hex codes listed starting from light 1-1 swatch. Delta E values refer to differences between palette colors. Color vision deficiency (CVD) safe calculated using 6 as minimum acceptable Delta E value.

	Author	Name	Delta E average	Delta E min	Delta E max	CVD safe?	Colors
1.	Stevens	Gray Pink; RdBu	32.74	10.93	56.19	No	'#e8e8e8', '#e4acac', '#c85a5a', '#b0d5df', '#ad9ea5', '#985356', '#64acbe', '#627f8c', '#574249'
2.	Stevens	Dark Blue; BuPu	30.76	10.88	54.56	No	'#e8e8e8', '#ace4e4', '#5ac8c8', '#dfb0d6', '#a5add3', '#5698b9', '#be64ac', '#8c62aa', '#3b4994'
3.	Stevens	Dark Cyan; GnBu	26.35	9.30	50.19	Yes	'#e8e8e8', '#b5c0da', '#6c83b5', #b8d6be', '#90b2b3', '#567994', '#73ae80', '#5a9178', '#2a5a5b'
4.	Stevens	Brown; PuOr	30.15	11.98	53.56	Yes	'#e8e8e8', '#e4d9ac', '#c8b35a', '#cbb8d7', '#c8ada0', '#af8e53', '#9972af', '#976b82', '#804d36'
5.	Grossen-bacher and Zehr	Dark Violet	29.27	12.69	58.41	Yes	'#CABED0', '#BC7C8F', '#AE3A4E', '#89A1C8', '#806A8A', '#77324C', '#4885C1', '#435786', '#3F2949'
6.	Brewer	sequential/ sequential (1994a); seqseq2	39.99	13.21	94.60	Yes	'#f3f3f3', '#b4d3e1', '#509dc2', '#f3e6b3', '#b3b3b3', '#376387', '#f3b300', '#b36600', '#000000'
7.	Brewer	sequential/ sequential (1994b); seqseq1	34.39	12.48	70.53	No	'#e8e6f2', '#b5d3e7', '#4fadd0', '#e5b4d9', '#b8b3d8', '#3983bb', '#de4fa6', '#b03598', '#2a1a8a'

New palettes

	Author	Name	Delta E average	Delta E min	Delta E max	CVD safe?	Colors
1.	Hildén	Orange Cyan; OrCy	41.24	14.42	79.67	Yes	'#f1f1f1', '#73dbfb', '#00aef1', '#f5ca5a', '#ada09e', '#296d9c', '#ee8b15', '#9e550d', '#3f212a'
2.	Hildén	Blue Red high saturation; BuRdHS	40.17	22.21	98.17	Yes	'#fdfbe2', '#edbba8', '#f84b5e', '#b0c9dc', '#9b7e8f', '#8a2b43', '#647de2', '#4b4588', '#25001a'
3.	Hildén	Green Violet; GnVi	38.25	15.11	66.45	Yes	'#e8f3f1', '#b7b9e7', '#936fc4', '#a2cb78', '#688b7c', '#5e5392', '#31991b', '#276e1d', '#0d4020'

Data and code

Live web tool available at: https://demo.koponen-hilden.fi/colorgridder/

Web tool source code: https://github.com/hjhilden/svelte-colorgridder

Jupyter Notebook and supplementary material:
https://github.com/hjhilden/bivariate-dataprocess

References

Accessibility Guidelines Working Group (2018) *Web Content Accessibility Guidelines (WCAG) 2.1*. Available at: https://www.w3.org/TR/WCAG21/ (Accessed: May 9, 2022).

Accessibility Guidelines Working Group (2021) *Techniques for WCAG 2.1*. Available at: https://www.w3.org/WAI/WCAG21/Techniques/#techniques (Accessed: November 8, 2021).

Adobe (2022a) *Illustrator™* [Software], (Version 26.3.1).

Adobe (2022b) *Photoshop™* [Software], (Version 23.4.2).

Aisch, G. (2018) *I wrote some code that automatically checks visualizations for non-colorblind safe colors. Here's how it works*, *vis4.net*. Available at: https://vis4.net/blog/2018/02/automate-colorblind-checking/index.html (Accessed: October 16, 2022).

Aisch, G. (2022a) *Chroma.js* [Software], (Version 2.4.0). Available at: https://gka.github.io/chroma.js/ (Accessed: May 9, 2022).

Aisch, G. (2022b) *Color palette helper*. Available at: https://github.com/gka/palettes (Accessed: October 30, 2022).

Akveo LLC (2022) *Eva Design System: Deep learning color generator*. Available at: https://colors.eva.design/ (Accessed: October 31, 2022).

Angus, C. (2022) *Maps* [Software]. Available at: https://github.com/VictimOfMaths/Maps/blob/f1da6bac535f39b2a1d4484e13ebdaa849b50a81/SunvsRain.png (Accessed: October 29, 2022).

Ardov, A. (2022) *Huetone*. Available at: https://huetone.ardov.me (Accessed: October 30, 2022).

Arnkil, H. (2013) *Colours in the visual world*. Helsinki: Aalto University School of Arts, Design and Architecture (Aalto University publication series. Art + design + architecture, 8/2013).

Atkins, T.Jr., Etemad, E.J. and Rivoal, F. (eds) (2022) "CSS Snapshot 2022." W3C. Available at: https://www.w3.org/TR/2022/DNOTE-css-2022-20221122/ (Accessed: December 1, 2022).

Atkins, T.Jr., Lilley, C. and Verou, L. (eds) (2022) "CSS Color Module Level 4. W3C Candidate Recommendation Draft." W3C. Available at: https://www.w3.org/TR/2022/CRD-css-color-4-20221101/ (Accessed: December 1, 2022).

Baldwin, N. (2022) *Lenoardo. Color tools for design systems*. Available at: https://leonardocolor.io/.

Bergstrom, C.T. and West, J.D. (2018) "Why scatter plots suggest causality, and what we can do about it." arXiv. Available at: https://doi.org/10.48550/arXiv.1809.09328.

Bertin, J. (2011) *Semiology of graphics: diagrams, networks, maps*. Translated by W.J. Berg. Redlands, CA: ESRI Press.

Bostock, M. (2017) *Command-Line Cartography, Part 3*, *Medium*. Available at: https://medium.com/@mbostock/command-line-cartography-part-3-1158e4c55a1e (Accessed: November 8, 2022).

Bostock, M. (2019) *Bivariate Choropleth / D3*, *Observable*. Available at: https://observablehq.com/@d3/bivariate-choropleth (Accessed: September 27, 2022).

Brewer, C. (1994) "Color Use Guidelines for Mapping and Visualization," in A.M. Maceachren and D.R.F. Taylor (eds) *Modern Cartography Series*. Academic Press (Visualization in Modern Cartography), pp. 123–147. Available at: https://doi.org/10.1016/B978-0-08-042415-6.50014-4.

Brewer, C. (1999) *Color Use Guidelines for Data Representation [ASA presentation]*. Available at: http://www.personal.psu.edu/cab38/ColorSch/ASApaper.html (Accessed: December 9, 2021).

Brewer, C. (2013) *ColorBrewer: Color Advice for Maps*. Available at: https://colorbrewer2.org/# (Accessed: November 9, 2021).

Brooke, B. (2019) *Bivariate Choropleth Color Generator*, *Observable*. Available at: https://observablehq.com/@benjaminadk/bivariate-choropleth-color-generator (Accessed: September 20, 2022).

Brown, T. (2022) *Color picker for data*. Available at: http://tristen.ca/hcl-picker/ (Accessed: October 30, 2022).

Burns, D.M. (2014) *The Many Faces of Garner Interference*. Indiana University.

Burzo, D. (2022) *Evercoder/culori* [Software]. Moqups. Available at: https://github.com/Evercoder/culori (Accessed: May 9, 2022).

Careri, W. (2022) "Designing for Neurodivergent Audiences, Nightingale," *Nightingale*, 8 February. Available at: https://nightingaledvs.com/designing-for-neurodivergent-audiences/ (Accessed: November 8, 2022).

Carswell, C.M. and Wickens, C.D. (1990) "The perceptual interaction of graphical attributes: Configurality, stimulus homogeneity, and object integration," *Perception & Psychophysics*, 47(2), pp. 157–168. Available at: https://doi.org/10.3758/BF03205980.

"Choropleth definition and meaning" (2022) *Collins English Dictionary*. HarperCollins Publishers. Available at: https://www.collinsdictionary.com/dictionary/english/choropleth (Accessed: December 10, 2022).

Christen, M. and Abegg, M. (2017) "The Effect of Magnification and Contrast on Reading Performance in Different Types of Simulated Low Vision," *Journal of Eye Movement Research*, 10. Available at: https://doi.org/10.16910/jemr.10.2.5.

Cleveland, W.S. and McGill, R. (1984) "Graphical Perception: Theory, Experimentation, and Application to the Development of Graphical Methods," *Journal of the American Statistical Association*, 79(387), pp. 531–554. Available at: https://doi.org/10.2307/2288400.

Cox, A. *et al.* (2014) "The Most Detailed Maps You'll See From the Midterm Senate Elections," *The New York Times*, 5 November. Available at: https://www.nytimes.com/interactive/2014/11/04/upshot/senate-maps.html (Accessed: December 1, 2022).

CSS Working Group (2022) *Compositing and Blending Level 2*. Available at: https://drafts.fxtf.org/compositing-2/ (Accessed: May 10, 2022).

D3: Data-Driven Documents [Software] (2022) [Software]. D3. Available at: https://github.com/d3/d3 (Accessed: December 6, 2022).

Daniels, M. (2018) *Human Terrain*, *The Pudding*. Available at: https://pudding.cool/2018/10/city_3d (Accessed: November 24, 2021).

de Bleser, F. (2016) *The Impact of Generative Design. The NodeBox Perspective*. Universiteit Antwerpen.

de Bleser, F., de Smedt, T. and Nijs, L. (2015) *NodeBox* [Software]. Available at: https://www.nodebox.net/ (Accessed: November 28, 2022).

Dempsey, C. (2017) *Types of GIS Data Explored: Vector and Raster*, *GIS Lounge*. Available at: https://web.archive.org/web/20210506173624/https://www.gislounge.com/geodatabases-explored-vector-and-raster-data/ (Accessed: November 3, 2021).

Derriche, M. (2022) *Updated the color selector (!3852) · Merge requests · Inkscape*, *GitLab*. Available at: https://gitlab.com/inkscape/inkscape/-/merge_requests/3852 (Accessed: November 24, 2022).

Dunn, R. (1989) "A Dynamic Approach to Two-Variable Color Mapping," *The American Statistician*, 43(4), p. 245. Available at: https://doi.org/10.2307/2685372.

Elavsky, F. (2021a) *Contrast and "No Use of Color Alone" in Stacked Bar Charts, Pie Charts, and Charts with Adjacent Element Placement*, *Observable*. Available at: https://observablehq.com/@frankelavsky/contrast-and-no-use-of-color-alone-in-adjacent-charts (Accessed: November 9, 2021).

Elavsky, F. (2021b) *Experimenting with "No Use of Color Alone" in Data Visualizations with Sequential, Diverging, and Circular Color Scales*, *Observable*. Available at: https://observablehq.com/@frankelavsky/experimental-color-scale-textures (Accessed: January 13, 2022).

Elavsky, F. (2021c) *Introduction to Accessible Contrast and Color for Data Visualization*, *Observable*. Available at: https://observablehq.com/@frankelavsky/chartability-contrast-series (Accessed: November 9, 2021).

Elmer, M.E. (2012) *Symbol Considerations for Bivariate Thematic Mapping (Master's thesis)*. University of Wisconsin–Madison.

Elmer, M.E. (2013) "Symbol Considerations for Bivariate thematic Maps," *Proceedings of 26th International Cartographic Conference*, p. 18.

Eyton, J.R. (1984) "Complementary-Color, Two-Variable Maps," *Annals of the Association of American Geographers*, 74(3), pp. 477–490. Available at: https://doi.org/10.1111/j.1467-8306.1984.tb01469.x.

FC, M. and Davis, T.L. (2022) *ggpattern: "ggplot2" pattern geoms* [Software], (Version 0.4.3-1). [Software]. Available at: https://coolbutuseless.github.io/package/ggpattern/index.html.

Federal Geographic Data Committee (2017) "Pattern chart." Available at: https://ngmdb.usgs.gov/fgdc_gds/geolsymstd/fgdc-geolsym-patternchart.pdf (Accessed: April 14, 2022).

Few, S. (2005) "Keep Radar Graphs Below the Radar - Far Below," *DM Review*, p. 5. Available at: https://www.perceptualedge.com/articles/dmreview/radar_graphs.pdf (Accessed: November 4, 2021).

Franconeri, S.L. *et al.* (2021) "The Science of Visual Data Communication: What Works," *Psychological Science in the Public Interest*, 22(3), pp. 110–161. Available at: https://doi.org/10.1177/15291006211051956.

Friendly, M. and Denis, D.J. (2001) *Milestones in the History of Thematic Cartography, Statistical Graphics, and Data Visualization*. Available at: https://www.datavis.ca/milestones/index.php?group=1850%2B&mid=ms146 (Accessed: November 8, 2021).

Gao, P., Li, Z. and Qin, Z. (2019) "Usability of value-by-alpha maps compared to area cartograms and proportional symbol maps," *Journal of Spatial Science*, 64(2), pp. 239–255. Available at: https://doi.org/10.1080/14498596.2018.1440649.

Gelman, A. (2011) "Irritating pseudo-populism, backed up by false statistics and implausible speculations," *Statistical Modeling, Causal Inference, and Social Science*, 11 April. Available at: https://statmodeling.stat.columbia.edu/2011/04/04/irritating_pseu/ (Accessed: November 29, 2022).

Glen, S. (2013) "Contingency Table: What is it used for?" *Statistics How To*, 31 July. Available at: https://www.statisticshowto.com/what-is-a-contingency-table/ (Accessed: September 26, 2022).

Gombrich, E.H. (1982) *The Image and the Eye: Further Studies in the Psychology of Pictorial Representation*. Oxford: Phaidon.

Grossenbacher, T. and Zehr, A. (2019) "Bivariate maps with ggplot2 and sf | Timo Grossenbacher," 19 April. Available at: https://timogrossenbacher.ch/2019/04/bivariate-maps-with-ggplot2-and-sf/ (Accessed: May 4, 2022).

Halley, E. (1700) "New and Correct Chart Shewing (sic) the Variations of the Compass in the Western & Southern Oceans, A." London.

Halliday, S.M. (1987) *Two-variable choropleth maps : an investigation of four alternate designs*. masters. Memorial University of Newfoundland. Available at: https://research.library.mun.ca/890/ (Accessed: November 11, 2022).

Haroz, S. and Whitney, D. (2012) "How Capacity Limits of Attention Influence Information Visualization Effectiveness," *IEEE Transactions on Visualization and Computer Graphics*, 18(12), pp. 2402–2410. Available at: https://doi.org/10.1109/TVCG.2012.233.

Henning, B. (2017) “Political Landscapes of the United Kingdom in 2017,” *Views of the World*, 13 June. Available at: https://www.viewsoftheworld.net/?p=5437 (Accessed: December 1, 2022).

Henry, S.L., Abou-Zahra, S. and Brewer, J. (2014) “The role of accessibility in a universal web,” in *Proceedings of the 11th Web for All Conference*. New York, NY, USA: Association for Computing Machinery (W4A ’14), pp. 1–4. Available at: https://doi.org/10.1145/2596695.2596719.

Hildén, J. (2019) *Esikaupunkien lapset – Förstädernas barn – Suburban children*, *Suomen tietokartasto – Finlands dataatlas – Data atlas of Finland*. Available at: http://tietokartasto.fi/kartasto_dev/Esikaupunkien-lapset/ (Accessed: December 9, 2022).

House of Commons Library (2022) *uk-hex-cartograms-noncontiguous* [Software]. House of Commons Library. Available at: https://github.com/houseofcommonslibrary/uk-hex-cartograms-noncontiguous (Accessed: November 11, 2022).

Huang, J. and Zhu, D.C. (2017) “Visually stressful striped patterns alter human visual cortical functional connectivity,” *Human Brain Mapping*, 38(11), pp. 5474–5484. Available at: https://doi.org/10.1002/hbm.23740.

Ihaka, R. (2003) “Colour for Presentation Graphics,” *Proceedings of DSC*, 2, p. 18.

Inkscape Developers (2022) *Inkscape* [Software], (Version 1.2.1).

Jones, W. (2011) “Making and using tiled pattern fill symbols,” *ArcGIS Blog*, 31 August. Available at: https://www.esri.com/arcgis-blog/products/product/imagery/making-and-using-tiled-pattern-fill-symbols/ (Accessed: November 1, 2022).

Jordahl, K. *et al.* (2020) *geopandas* [Software], (Version v0.8.1). Zenodo. Available at: https://doi.org/10.5281/zenodo.3946761.

Kalloniatis, M. and Luu, C. (2007) *The Perception of Color. Table 1, Prevalence of congenital color deficiencies - Webvision - NCBI Bookshelf*, *Webvision: The Organization of the Retina and Visual System [Internet]*. Available at: https://www.ncbi.nlm.nih.gov/books/NBK11538/table/ch28kallcolor.T1/ (Accessed: October 16, 2022).

Kluyver, T. *et al.* (2016) “Jupyter Notebooks—a publishing format for reproducible computational workflows,” in *Positioning and Power in Academic Publishing: Players, Agents and Agendas*, p. 4.

Koponen, J. and Hildén, J. (2019) *Data visualization handbook*. First edition. Helsinki: Aalto University School of Arts, Design and Architecture (Aalto University publication series. Art + design + architecture, 1/2019).

Kosara, R. (2007) “A Critique of Chernoff Faces,” *Eagereyes*, 25 February. Available at: https://web.archive.org/web/20210413102506/https://eagereyes.org/criticism/chernoff-faces (Accessed: November 4, 2021).

Kosara, R. (2022) “More Than Meets the Eye: A Closer Look at Encodings in Visualization,” *IEEE Computer Graphics and Applications*, 42(2), pp. 110–114. Available at: https://doi.org/10.1109/MCG.2021.3138608.

Kraak, M.-J. and Ormeling, F. (2010) *Cartography : visualization of geospatial data*. 3rd ed. Harlow: Prentice Hall.

Krygier, J. and Wood, D. (2016) *Making Maps: A Visual Guide to Map Design for GIS, 3rd Edition*. New York, NY: The Guilford Press.

Lee, M.D., Reilly, R.E. and Butavicius, M.A. (2003) “An Empirical Evaluation of Chernoff Faces, Star Glyphs, and Spatial Visualizations for Binary Data,” *Conferences in Research and Practice in Information Technology*, 24, p. 10.

Lehni, J. and Puckey, J. (2021) *Paper.js* [Software], (Version v0.12.15). Available at: http://paperjs.org/ (Accessed: November 28, 2022).

Leinonen, T. (2010) *Designing learning tools: methodological insights*. Helsinki: Aalto University School of Art and Design (Aalto University School of Art and Design. A, 111).

Leonowicz, A. (2006) “Two-variable choropleth maps as a useful tool for visualization of geographical relationship,” *Geografija*, 42, pp. 33–37.

Lewis, D. (2021) *Cartogrampy* [Software]. Available at: https://github.com/danlewis85/cartogrampy (Accessed: October 29, 2022).

Lilley, C. and Needham, C. (2022) “High Dynamic Range and Wide Gamut Color on the Web.” Available at: https://w3c.github.io/ColorWeb-CG/ (Accessed: December 1, 2022).

Liu, Y. and Heer, J. (2018) “Somewhere Over the Rainbow: An Empirical Assessment of Quantitative Colormaps,” in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. Montreal QC Canada: ACM, pp. 1–12. Available at: https://doi.org/10.1145/3173574.3174172.

Luz, S. and Masoodian, M. (2014) “Readability of a background map layer under a semi-transparent foreground layer,” in *Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces*. New York, NY, USA: Association for Computing Machinery (AVI ’14), pp. 161–168. Available at: https://doi.org/10.1145/2598153.2598174.

MacEachren, A.M. (1982a) “Map Complexity: Comparison and Measurement,” *The American Cartographer*, 9(1), pp. 31–46. Available at: https://doi.org/10.1559/152304082783948286.

MacEachren, A.M. (1982b) “The Role of Complexity and Symbolization Method in Thematic Map Effectiveness,” *Annals of the Association of American Geographers*, 72(4), pp. 495–513. Available at: https://doi.org/10.1111/j.1467-8306.1982.tb01841.x.

MacEachren, A.M. (1995) *How maps work: representation, visualization, and design*. New York, NY: Guilford Press.

Matplotlib development team (2022) *Matplotlib: Visualization with Python* [Software], (Version 3.6.1). Available at: https://matplotlib.org/ (Accessed: April 14, 2022).

Meeks, E. and Lu, S. (2018) *Viz Palette*. Available at: https://projects.susielu.com/viz-palette (Accessed: October 16, 2022).

Morris, C.J. and Ebert, D.S. (2000) “An Experimental Analysis of the Effectiveness of Features in Chernoff Faces,” *Proceedings of SPIE - The International Society for Optical Engineering*, p. 7.

Myndex (2022) *The CIE 1931 chromaticity diagram showing the P3 gamut, and the gamuts of some other common RGB color spaces*. Available at: https://commons.wikimedia.org/wiki/File:CIE1931xy_gamut_comparison_of_sRGB_P3_Rec2020.svg (Accessed: December 1, 2022).

N.A. (2022) *Huemint - About Huemint - Machine learning for graphic design colorization*. Available at: https://huemint.com/about/ (Accessed: October 30, 2022).

National Land Survey of Finland (2018) “Suomen pinta-ala kunnittain.” Available at: https://www.maanmittauslaitos.fi/sites/maanmittauslaitos.fi/files/attachments/2018/01/Suomen_pa_2018_kunta_maakunta_0.xlsx.

National Land Survey of Finland (2022) *Geodata portal Paikkatietoikkuna*. Available at: https://kartta.paikkatietoikkuna.fi/ (Accessed: September 22, 2022).

Nelson, E.S. (2000a) “Designing Effective Bivariate Symbols: The Influence of Perceptual Grouping Processes,” *Cartography and Geographic Information Science*, 27(4), pp. 261–278. Available at: https://doi.org/10.1559/152304000783547786.

Nelson, E.S. (2000b) “The impact of bivariate symbol design on task performance in a map setting,” *Cartographica*, 37(4), pp. 61–78. Available at: https://doi.org/10.3138/V743-K505-5510-66Q5.

Nelson, J. (2020) “Multivariate Mapping,” *Geographic Information Science & Technology Body of Knowledge*. Edited by Wilson, John P., 2020(Q1). Available at: https://doi.org/10.22224/gistbok/2020.1.5.

Nowosad, J. (2019) *Check Color Palettes for Problems with Color Vision Deficiency* [Software]. Available at: https://nowosad.github.io/colorblindcheck.

Nowosad, J. (2020) "How to choose a bivariate color palette?" *R-bloggers*, 25 August. Available at: https://www.r-bloggers.com/2020/08/how-to-choose-a-bivariate-color-palette/ (Accessed: October 15, 2022).

Nusrat, S. *et al.* (2018) "Cartogram Visualization for Bivariate Geo-Statistical Data," *IEEE Transactions on Visualization and Computer Graphics*, 24(10), pp. 2675–2688. Available at: https://doi.org/10.1109/TVCG.2017.2765330.

Observable, Inc. (2022) *Observable documentation*. Available at: https://observablehq.com/@observablehq/documentation (Accessed: November 9, 2022).

Olson, J.M. (1981) "Spectrally Encoded Two-Variable Maps," *Annals of the Association of American Geographers*, 71(2), pp. 259–276. Available at: https://doi.org/10.1111/j.1467-8306.1981.tb01352.x.

Organ, N. (2021) *An Incomplete Guide to Accessible Data Visualization*, *Medium*. Available at: https://towardsdatascience.com/an-incomplete-guide-to-accessible-data-visualization-33f15bfcc400 (Accessed: November 9, 2021).

OSF Official Statistics of Finland (2022) *Key figures on population by region, 1990-2020*, *Tilastokeskuksen PxWeb-tietokannat*. Available at: https://pxdata.stat.fi/PxWeb/pxweb/en/StatFin/StatFin__vaerak/statfin_vaerak_pxt_11ra.px/ (Accessed: January 28, 2022).

Ottoson, B. (2022) *A perceptual color space for image processing*. Available at: https://bottosson.github.io/posts/oklab/ (Accessed: November 24, 2022).

Pandas development team (2021) *Pandas* [Software], (Version 1.3.4). Available at: https://pandas.pydata.org/.

Prener, C. (2022) *biscale* [Software], (Version 1.0.0). Available at: https://github.com/chris-prener/biscale (Accessed: October 15, 2022).

Prener, C., Grossenbacher, T. and Zehr, A. (2020) *Bivariate Mapping with ggplot2*. Available at: https://slu-opengis.github.io/biscale/articles/biscale.html (Accessed: February 2, 2022).

Project Jupyter (2019) *Jupyter Lab* [Software], (Version 1.1.4). Available at: https://jupyter.org/install.

QGIS Association (2021) *QGIS* [Software], (Version 3.22.6-Białowieża). Available at: https://www.qgis.org.

Reas, C. and Fry, B. (2007) *Processing: a programming handbook for visual designers and artists*. Cambridge, MA: Mit Press.

Ripamonti, C. *et al.* (2009) "The S-cone contribution to luminance depends on the M- and L-cone adaptation levels: Silent surrounds?" *Journal of Vision*, 9(3), p. 10. Available at: https://doi.org/10.1167/9.3.10.

Robertson, P.K. and O'Callaghan, J.F. (1986) "The Generation of Color Sequences for Univariate and Bivariate Mapping," *IEEE Computer Graphics and Applications*, 6(2), pp. 24–32. Available at: https://doi.org/10.1109/MCG.1986.276688.

Robinson, et. al. (1995) *Elements of cartography*. 6. ed. New York, NY: Wiley.

Roth, R. (2017) "Visual Variables," in, pp. 1–11. Available at: https://doi.org/10.1002/9781118786352.wbieg0761.

Roth, R., Woodruff, A.W. and Johnson, Z.F. (2010) "Value-by-alpha Maps: An Alternative Technique to the Cartogram," *The Cartographic Journal*, 47(2), pp. 130–140. Available at: https://doi.org/10.1179/000870409X12488753453372.

RStudio, Inc. (2017) *RStudio* [Software], (Version 1.0.153).

Schuessler, Z. (2019) *Delta E 101*. Available at: http://zschuessler.github.io/DeltaE/learn/ (Accessed: May 11, 2022).

Schwabish, J. (2017) "Choosing Map Bins," *PolicyViz*, 2 November. Available at: https://policyviz.com/2017/11/02/choosing-map-bins/ (Accessed: September 27, 2022).

Shiny [Software] (2022) [Software]. RStudio. Available at: https://github.com/rstudio/shiny (Accessed: December 6, 2022).

Shixie *et al.* (2020) "Color palettes and accessibility features for data visualization," *_carbondesign*, 15 November. Available at: https://medium.com/carbondesign/color-palettes-and-accessibility-features-for-data-visualization-7869f4874fca (Accessed: May 6, 2022).

Sitnik, A. and Solovev, I. (2022) *Easing Functions Cheat Sheet*. Available at: http://easings.net/ (Accessed: October 11, 2022).

Slocum, T.A. (2014) *Thematic cartography and geovisualization*. Third edition., Pearson new international edition. Harlow, Essex: Pearson.

Smart, S., Wu, K. and Szafir, D.A. (2019) "Color Crafting: Automating the Construction of Designer Quality Color Ramps," *arXiv:1908.00629 [cs]* [Preprint]. Available at: http://arxiv.org/abs/1908.00629 (Accessed: May 6, 2022).

Smith, T. and Guild, J. (1931) "The C.I.E. colorimetric standards and their use," *Transactions of the Optical Society*, 33(3), p. 73. Available at: https://doi.org/10.1088/1475-4878/33/3/301.

Somers, A. (2020) *A comparative look at Lab and Luv colorspaces, and LCh.*, *Gist*. Available at: https://gist.github.com/Myndex/47c793f8a054041bd2b52caa7ad5271c (Accessed: December 9, 2021).

Somers, A. (2021a) *APCA Visual Contrast Calculator*, *Myndex™ Simple APCA Tool*. Available at: https://www.myndex.com/APCA/simple (Accessed: December 10, 2021).

Somers, A. (2021b) *Why the New Contrast Method APCA*. Available at: https://github.com/Myndex/SAPC-APCA/blob/7eda3c083fb2d7b066336ffc682744aca7797389/WhyAPCA.md (Accessed: December 10, 2021).

Somers, A. (2022) *APCA™ for W3C & WCAG_3* [Software]. Available at: https://github.com/Myndex/apca-w3 (Accessed: May 11, 2022).

Spellman *et al.* (eds) (2021) "W3C Accessibility Guidelines (WCAG) 3.0 W3C Working Draft." W3C. Available at: https://www.w3.org/TR/wcag-3.0/ (Accessed: December 5, 2022).

Statistics Finland (2018a) "Municipalities (1:4,5 M). Municipality-based statistical units." Available at: https://stat.fi/org/avoindata/paikkatietoaineistot/kuntapohjaiset_tilastointialueet_en.html (Accessed: February 3, 2022).

Statistics Finland (2018b) "Regions (1:4,5 M). Municipality-based statistical units." Available at: https://stat.fi/org/avoindata/paikkatietoaineistot/kuntapohjaiset_tilastointialueet_en.html (Accessed: February 3, 2022).

Statistics Finland (2019) *Municipal key figures by Region 2019, Information and Year*, *Tilastokeskuksen PxWeb-tietokannat*. Available at: https://pxdata.stat.fi/PxWeb/pxweb/en/Kuntien_avainluvut/Kuntien_avainluvut__2019/kuntien_avainluvut_2019_aikasarja.px/ (Accessed: September 26, 2022).

Stefan, H. *et al.* (2007) "Multivariate Mapping in High Quality Atlases," *Proc. of the 23th Int. Conference of the ICA* [Preprint].

Stefaner, M. and Bertini, E. (2018) "Visual Perception and Visualization with Steve Haroz" (Data Stories). Available at: https://datastori.es/128-visual-perception-and-visualization-with-steve-haroz/ (Accessed: December 1, 2021).

Stevens, J. (2015) *Bivariate Choropleth Maps: A How-to Guide*, *Joshua Stevens – blog*. Available at: https://web.archive.org/web/20210424085202/https://www.joshuastevens.net/cartography/make-a-bivariate-choropleth-map/ (Accessed: November 8, 2021).

Stone, M., Szafir, D.A. and Setlur, V. (2014) "An engineering model for color difference as a function of size," in *Color and imaging conference*. Boston, pp. 253–258.

Strode, G. *et al.* (2020) "Operationalizing Trumbo's Principles of Bivariate Choropleth Map Design," *Cartographic Perspectives* [Preprint]. Available at: https://doi.org/10.14714/CP94.1538.

Svelte (2022) *Sveltejs* [Software]. Available at: https://github.com/sveltejs/svelte (Accessed: December 2, 2022).

Szafir, D.A. *et al.* (2016) "Four types of ensemble coding in data visualizations," *Journal of Vision*, 16(5), p. 11. Available at: https://doi.org/10.1167/16.5.11.

Szafir, D.A. (2018) "Modeling Color Difference for Visualization Design," *IEEE Transactions on Visualization and Computer Graphics*, 24(1), pp. 392–401. Available at: https://doi.org/10.1109/TVCG.2017.2744359.

Treisman, A.M. and Gelade, G. (1980) "A feature-integration theory of attention," *Cognitive psychology*, 12(1), pp. 97–136.

Trumbo, B.E. (1981) "A Theory for Coloring Bivariate Statistical Maps," *The American Statistician*, 35(4), pp. 220–226. Available at: https://doi.org/10.2307/2683294.

Tufte, E. (1990) *Envisioning Information*. Cheshire, CT: Graphics Press.

Tufte, E. (2001) *The Visual Display of Quantitative Information*. 2. edn. Cheshire, CT: Graphics Press.

Turner, E. (1977) "Life in Los Angeles." Available at: https://mapdesign.icaci.org/2014/12/mapcarte-353365-life-in-los-angeles-by-eugene-turner-1977/.

Tyner, J.A. (2010) *Principles of map design*. New York, NY: Guilford Press.

Use All Five (2019) *Accessible Brand Colors*. Available at: https://abc.useallfive.com (Accessed: October 30, 2022).

van der Walt, S. and Smith, N. (2015) *Matplotlib colormaps* [Software]. Available at: http://bids.github.io/colormap/ (Accessed: October 31, 2022).

van Rossum, J., van Blokland, E. and Berlaen, F. (2021) *DrawBot* [Software], (Version 3.128). Available at: https://www.drawbot.com/ (Accessed: December 7, 2022).

Verou, L. (2020) "LCH colors in CSS: what, why, and how?" *Life at the bleeding edge (of web standards)*, 4 April. Available at: https://lea.verou.me/2020/04/lch-colors-in-css-what-why-and-how/ (Accessed: November 29, 2022).

Verou, L. (2022) *Color.js: Let's get serious about color* [Software]. Available at: https://github.com/LeaVerou/color.js (Accessed: October 30, 2022).

Victor, B. (2013) *Drawing Dynamic Visualizations*. Available at: https://vimeo.com/66085662.

Wainer, H. and Francolini, C.M. (1980) "An Empirical Inquiry Concerning Human Understanding of Two-Variable Color Maps," *The American Statistician*, 34(2), p. 81. Available at: https://doi.org/10.2307/2684111.

Ward, M.O. (2008) "Multivariate Data Glyphs: Principles and Practice," in C. Chen, W. Härdle, and A. Unwin (eds) *Handbook of Data Visualization*. Berlin, Heidelberg: Springer (Springer Handbooks Comp.Statistics), pp. 179–198. Available at: https://doi.org/10.1007/978-3-540-33037-0_8.

Ware, C. (2008) *Visual Thinking for Design*. Waltham, MA: Morgan Kaufmann.

Ware, C. (2009) "Quantitative texton sequences for legible bivariate maps," *IEEE Transactions on Visualization and Computer Graphics*, 15(6), pp. 1523–1530. Available at: https://doi.org/10.1109/TVCG.2009.175.

Ware, C. (2013) *Information visualization: perception for design*. Third edition. Waltham, MA: Morgan Kaufmann (Interactive technologies).

Web Colors Explained (2006) *Hexadecimal Numbers Guide*. Available at: https://web.archive.org/web/20060422004336/http://www.web-colors-explained.com/hex.php (Accessed: December 5, 2022).

WebAIM (2021) *WebAIM: Contrast Checker*. Available at: https://webaim.org/resources/contrastchecker/ (Accessed: November 9, 2021).

Woodruff, A. (2010) "Value-by-alpha maps," 8 July. Available at: https://web.archive.org/web/20210502100410/http://andywoodruff.com/blog/value-by-alpha-maps/ (Accessed: November 24, 2021).

Wright, K. (2021) *pals: Color Palettes, Colormaps, and Tools to Evaluate Them* [Software], (Version 1.7). Available at: https://CRAN.R-project.org/package=pals (Accessed: October 15, 2022).

Zeileis, A. *et al.* (2020) "colorspace: A toolbox for manipulating and assessing colors and palettes," *Journal of Statistical Software*, 96(1), pp. 1–49. Available at: https://doi.org/10.18637/jss.v096.i01.

List of figures

Source details under references. Figures with no mentioned source are original work.

List of tables