Dynamic visualization of high-dimensional data via low-dimension projections across 2D and 3D display devices

Candidature confirmation report

Nicholas S Spyrison

Supervisors:

Prof. Kimbal Marriott,

Prof. Dianne Cook,

Prof. German Valencia

Faculty of Information Technology

Monash University

Australia

March 2019

Contents

Abstract

Visualizing data space is crucial to exploratory and general data analysis yet doing so quickly becomes difficult as the dimensionality of the data increases. Traditionally, static, low-dimensional linear embeddings are used to identify clustering, outliers, and structure. Observing one such embedding often misses a significant amount of variation, and hence, information held within the data. Tours are a class of dynamic linear projections that animates many linear projections as the orientation in data space changes. This maintains transparency to the original variables, while preserving information in the data. User-controlled steering (UCS) of the original dimensionality offers fine control of the local structure of projections.

Excitement and hype about virtual reality preceded hardware's progress in the 1980's and 90's for a false start with data visualization. Despite this and more recently, studies have regularly shown increased accuracy of perception of visuals displayed in 3D over 2D, including in projected subspaces.

Multivariate data is ubiquitous and viewing it in data-space is a crucial aspect of data analysis and consumption. This research involves scaling data visualization as data dimensionality increases, improving the perception of dynamic linear projections via application in VR, and implements novel dynamic projections in 3D space, and explores the benefits of doing so.

Chapter 1

Introduction

1.1 Exploratory data analysis

The term exploratory data analysis was coined in Tukey (1977). Tukey purposefully leaves the term to broadly encompass the phase of exploring, understanding the structure of the data and validating assumptions made by prospective methodology. Data is everywhere and understanding the shape and distribution of the data is an early and fundamental task. Visualization is crucial to a clear understanding of the data. Things can go awry when data is summarized via numeric statistics alone (Anscombe, 1973), demonstrated in figure 1.1 (Matejka and Fitzmaurice, 2017). In these studies, bivariate data have the same summary statistics, such as mean and standard deviation, yet contain obvious visual trends and shapes that could go completely unheeded if graphing the data is foregone. Because there are inhearant dangers to relying on statistics alone, this requirement for looking at visuals necessitates *human-in-the-loop* analysis, defined as a model that requires human interaction.

It is clear that data-space visualization is needed but becomes complex as data dimensionality increases. Embedding (or projecting) $p-$dimensional data on to a lower, d-dimensional subspace is a common dimension reduction approach to visualize multivariate data spaces. Traditionally single static projection is used to summarize a space, which necessarily shows a subset of the variation of the data. Asimov (1985) suggested the use

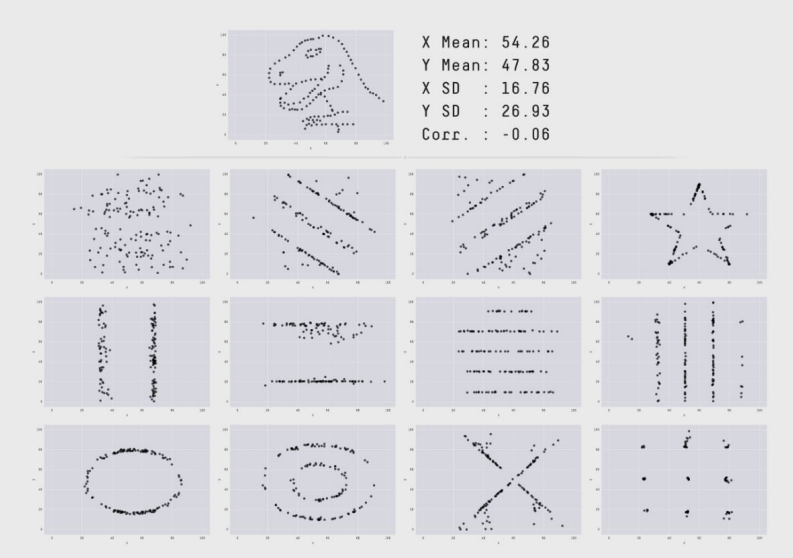


Figure 1.1: *12 data sets created from the datasaurus by simulated annealing. Each is restrained to the the same summary statistics, but given a shapes with visual peculiarity to mutate into (Matejka and Fitzmaurice, 2017).*

of viewing projections dynamically across a changing projection basis allows for more variation to be contained and viewed temporally. This dynamic view of many changing projections is known as *tours*. While, there are different methods of generating tour paths, human-in-the-loop user-controlled steering (UCS) offers the finest control for navigating local structure and is particularly useful in exploration after an interesting feature has been identified.

Hardware improvement has made VR more affordable and available to wider audiences, at ever increasing resolutions of display. The literature points to improved perception when viewing multivariate data in 3D VR as compared with traditional 2D monitors. Yet tours are often viewed on 2D monitors while the selected geometric display dictates the dimensionality of graphic, $d = \{1, 2, p\}$. With the exception Nelson, Cook, and Cruz-Neira (1998), where $d = 2$ tours were viewed in 3D head-tracked VR. Bringing dynamic projections and UCS into VR should improve perception of $d = 2$ embeddings, allow for more accurate and intuitive perception of 3D surfaces, and enable dynamic projections $d = 3$ spaces.

1.2 Research objectives

Data and models are typically high-dimensional, with many variables or parameters. Developing new methods to visualize high dimensions has been a pursuit of statisticians, computer scientists and visualization researchers for decades. As technology evolves examining, extending, and assessing current techniques, in new environments, for new data challenges, is an important endeavor. The primary goal of this PhD research can then be summarized as:

Research objectives (RO):

1. **How can UCS be implemented in 1- and 2D projections?** (Work in progress, chapter 3.) Following the theory laid out in Cook and Buja (1997), a new UCS algorithm is devised for use in animation specific implementations. This sets a frame work to be used in the remaining objectives.
2. **Does 2D UCS provide benefits over alternatives?** (Future work, chapter 4.)   The quality and effectiveness of the previous algorithm will be compared to benchmark datasets against commonly used alternatives of static, single, linear and non-linear projection techniques.
3. **How can UCS be extended to 3D?** (Future work, chapter 4.) This involves extending the UCS algorithm to abrbitary, or at least, a third dimension and the rendering of 3D tours for use across mixed reality display devices.
4. **Does UCS in 3D displays provide perception benefits over 2D displays?** (Future work, chapter 4.) Building on the work from the previous objective the efficacy of 3D tours should be explored using modern hardware across display devices.

1.3 Methodology

RO #1 is completed **applied research** following work in Cook and Buja (1997) to allow for UCS. This outcome of this is an *R* package, `spinifex`, which will be submitted to to CRAN and for hosting and distribution. This forms the foundation for future work in the remaining objectives.

The second objective is a **case study** comparison between dynamic linear projections and alternatives (static linear and static non-linear projections such as Principal Component Analysis, Multi-Dimensional Scaling, and t-distributed neighbor embeddings, described in more detail in chapter 4). Measures comparing between the techniques will include: variation explained, transparency to the original variable space, clustering, and outlier identification.

RO #3 involves **experimental design** applying contribution from RO #1 with the immersive analytics toolkit (IATK) in Cordeil (2019), an extensible toolkit for visualizing data in VR. First the work from RO # 1 will be extend into $d = 3$ projections an surface projections. This will be called from the IATK and used for a standardized interface across display devices.

The response to RO #4 is a *randomized full factorial design* **empirical study** to explore the efficacy of bringing UCS into 3D as compared across various display devices. In this design every participant will complete every task on every display device. Quantitative measurements include participant speed and accuracy of tasks, biometric readings, and subjective Likert survey of participants. A lineup-type model as outlined in Hofmann et al. (2012) may also be employed for assessing quality of display types.

In this report, the related literature is discussed in chapter 2. A brief overview of the research is given in chapter 1.5, followed by the completed work and future work in chapters 3 and 4 respectively. A prospective timeline is listed in chapter 5. Notation for dynamic touring and VR data visualization can be found in appendix A, and an excerpt of a paper to be submitted to the R Journal can be found in appendix B.

1.4 Workflow and reproducibility

Figure 1.2 depicts general data analysis workflow (Wickham and Grolemund, 2016). Where data first must be imported into a tool, the structure of the data must be tidied and ordered neatly into the correct use format. After the data enters a repeating cycle, where values maybe transformed, visualized, and modeled with communication going to the appropriate recipients.

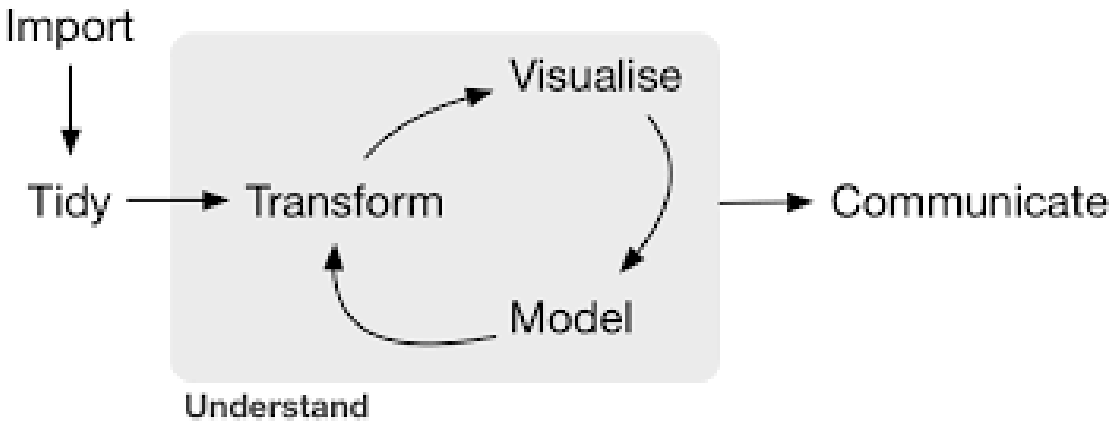


Figure 1.2: *Data analysis workflow (Wickham and Grolemund, 2016). This research aids visualization in exploratory data analysis and in the workflow.*

The programing language, *R*, is used in the work described below to import, tidy, and transform data. It can be use directly to visualize 2D tour (RO #1 & 2) or be consumed into the game engine *Unity* to visualize 3D tours (RO #3 & 4). Doing analysis and writeup in such programmatic ways allow work to be done reproducibly. All work completed and discussed is done with reproduction principles in mind, where data, analysis, and code are stored in the same directory in order to make the work as transparent and reproducible as possible. Transparency and reproduction of work is a key feature to a mature analysis workflow as they validate and defend the claims and methodology held within a work. Directories of completed and planned work are hosted publicly on GitHub, including this report. Accessing the source files for this report is discussed in section 6.

1.5 Project overview

The research gaps in the literature review leave room for the research objectives outlined in the introduction. Figure 1.3 depicts a schematic flow chart that the research objectives will be executed in. RO #1, application of 2D user-controlled steering (UCS), sets the foundation for which the other objectives can be researched. RO #3, the application of 3D UCS, must precede RO #4, exploring the efficacy of 3D UCS across display devices. RO #2, the comparison of 2D UCS vs alternatives, must come after RO #1, but is of lower priority to RO #3 & 4, and so will be conducted last, in the event of a time crunch.

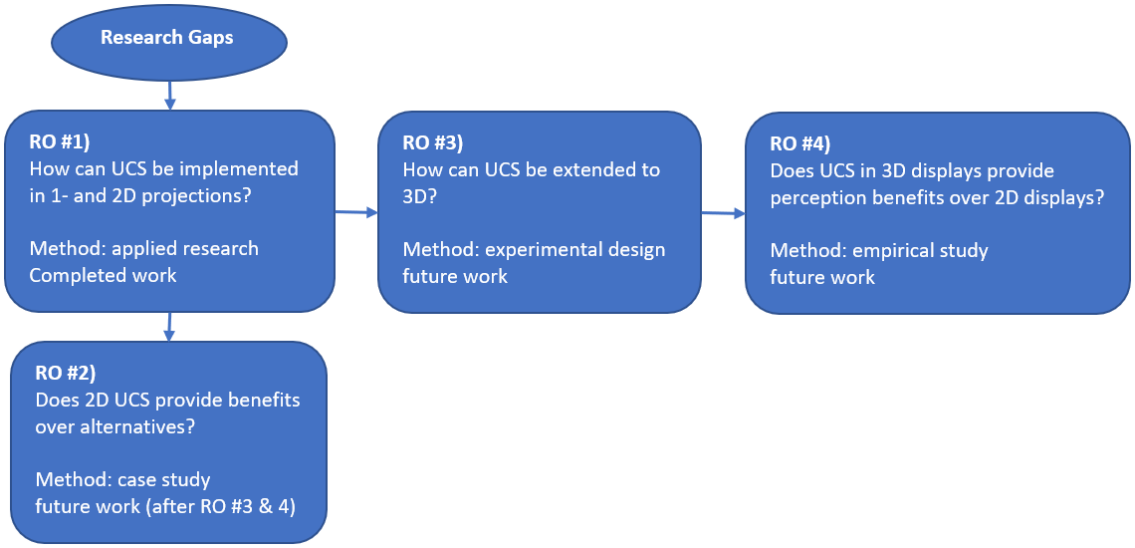


Figure 1.3: *Flow chart of research objective (RO) dependencies, work order, and methodology.*

Chapter 2

Literature review

In the following chapter we discuss the academic research in two primary areas: that of dynamic linear projection (collectively known as tours) followed by multivariate data visualization in stereoscopic 3D. After each section we highlight the research gaps and show how they relate to the research objectives.

2.1 Dynamic linear projections of multivariate data (tours)

2.1.1 Overview

The introduction established that visualizing data-space is an important aspect of exploratory data analysis and data analysis in general. Yet, there is an inherent difficulty as the dimensionality of data increases. In univariate data sets histograms, or smoothed density curves are employed to visualize data. In bivariate data $x - y$ scatterplots and contour plots (2D density) can be employed. In three dimensions the two most common techniques are 3D scatterplot[1] or 2D scatterplot with the 3rd variable as an aesthetic (such as color, size, or height). Such variable mappings can afford another dimension or 2, but this is not a sustainable solution.

[1] Graphs depicting 3 dimensions are typically printed on paper, or rendered on a 2D monitor, they are intrinsically 2D images of monocular 3D spaces, sometimes referred to as 2.5D data visualization, more on this in section A.2.

Visualization of multivariate data for modest p numeric dimension, even 6 or 8, quickly becomes complex in. It's far too common that visualizing in data-space is dropped altogether in favor of modeling parameter-space, model-space, or worse: long tables of statistics without visuals (Wickham, Cook, and Hofmann, 2015). A solution that better scales with the dimensionality of data is needed; this is where dimensionality reduction comes in. This work will focus on a group of dynamic linear projection techniques collectively known as *tours*. Broader review of dimensionality reduction techniques is discussed in Grinstein, Trutschl, and Cvek (2002), and Heer, Bostock, and Ogievetsky (2010). Tours are used for a couple of salient features: use of linear projections maintaining transparency back to the original variable space (which non-linear projections lose) and keeps all components and their information in tack (which static linear projections lose). Employing the breadth of tours extends the dimensionality visualization, and with it, the intrinsic understanding of structure and distribution of data that is more succinct or beyond the reach of summary statistics alone.

Let p be the dimensionality of the data, and d be the dimension of the projection space. Tours perform linear dimensionality reduction, orthogonally projecting p-space down to $d(\leq p)$ dimensions. Many such projections are interpolated, each making small rotations in p-space. These frames are then viewed in order as an animation of the lower dimensional embedding changing as the original variable space is manipulated. Shadow puppets offer a useful analogy to aid in conceptualizing touring. Imagine a fixed light source facing a wall. When an object is introduced it projects a 2D shadow onto the wall. This is a physical representation of a simple projection, that from $p = 3$ down to $d = 2$. If the object rotates then the shadow correspondingly changes. Observers watching only the shadow are functionally watching a $d = 2$ tour as the 3D object is manipulated. Some orientations explain more information about the shape of the object than others but watching an animation of the shadow changing gives a more robust understanding than looking at any one frame. More complex structures generally require more time to comprehend the nature of the geometry. These features hold true in touring as well.

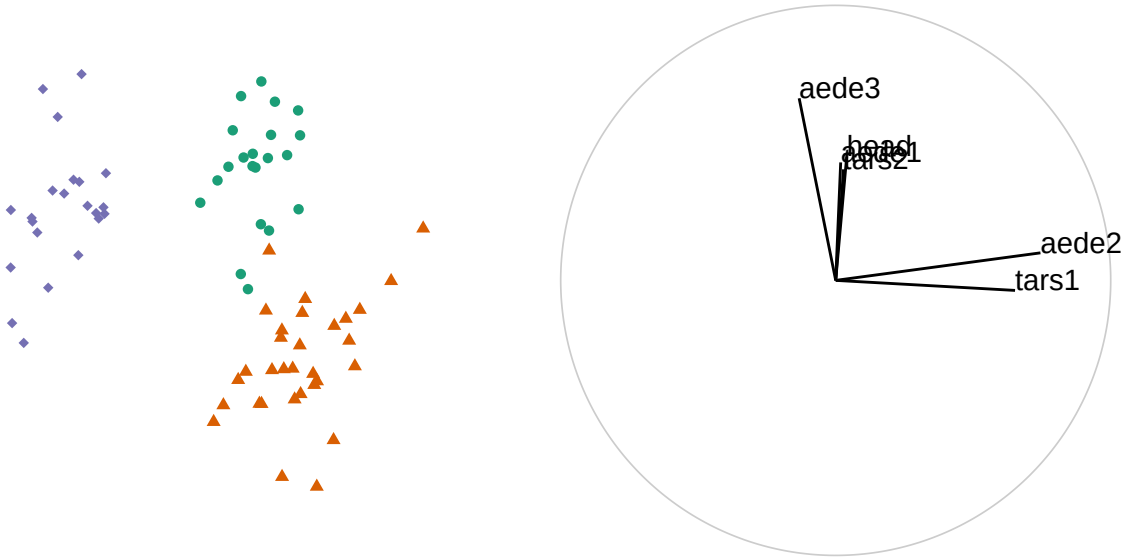


Figure 2.1: *(left) one frame of a x-y scatterplot projection, the results of holes tour with a corresponding reference frame indicating the direction and magnitude the variables contribute to the 2D projection.*

2.1.2 Example illustrations

A list of tour notation is given in the appendix section A.1.

This sections a toy data set of 74 observations of flea beetles across 6 numeric variables corresponding to physical measurements. Each observation belongs to one of 3 species as shown in the color and shape of the data points. Figure 2.1 shows one frame of a tour (left) and the corresponding reference frame (right) showing the linear combination of the variables onto the projection-space, a visual representation of the basis.

Tours view many such frames in sequence, by identifying some target frames and then interpolating between them as shown schematically in figure 2.2, captured from figure 1 of Buja et al. (2005). For illustration figure 2.3 lays out frames of a tour, A html version of a user-controlled steering can be found at `https://nspyrison.netlify.com/thesis/flea_manualtour_mvar5/`. The sections below will outline path generation and enumerate display geoms.

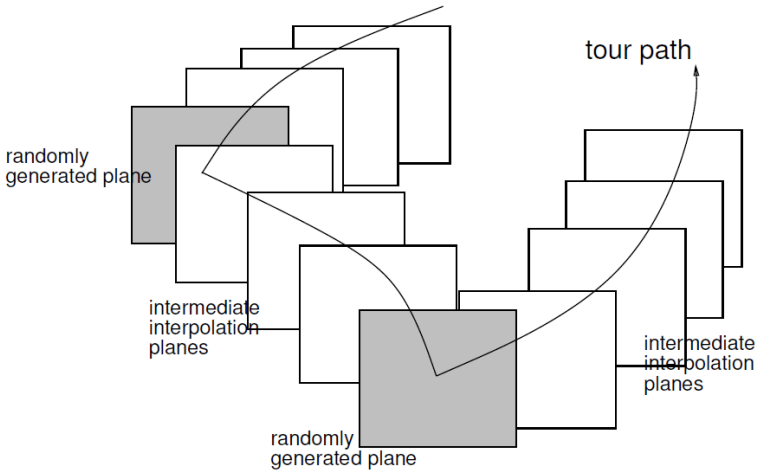


Figure 2.2: *Screen capture of figure 1 from Buja et al. (2005). A schematic representation of randomly generated planes (from a grand tour) and intermediate interpolation planes.*

2.1.3 History

Touring was first introduced by Asimov (1985) with his purposed *grand tour* at the Stanford Linear Accelerator, Stanford University. In which, Asimov suggested three types of grand tours: torus, at-random, and random-walk. The original application of tours was performed on high energy physics on the PRIM-9 system.

Before choosing projection paths randomly, an exhaustive search of $p-$space was suggested by McDonald (1982), also at the Stanford Linear Accelerator. This was later coined *little tour*.

Friedman and Tukey (1974) and later Huber (1985) purposed projection pursuit (also referred to as PP). Projection pursuit involves identifying "interesting" projection, remove a single component of the data, and then iterates in this newly embedded subspace. Within each subspace the projection seeks for a local extremum via hill climbing algorithm on an objective function. This formed the basis for *guided tours* suggested by Hurley and Buja (1990).

The grand and little tour have no input from the user aside from the starting basis. Guided tours allow for an index to be selected, but the bulk of touring development since has largely been around dynamic display, user interaction, geometric representation, and application. The details are expounded on in the following sections.

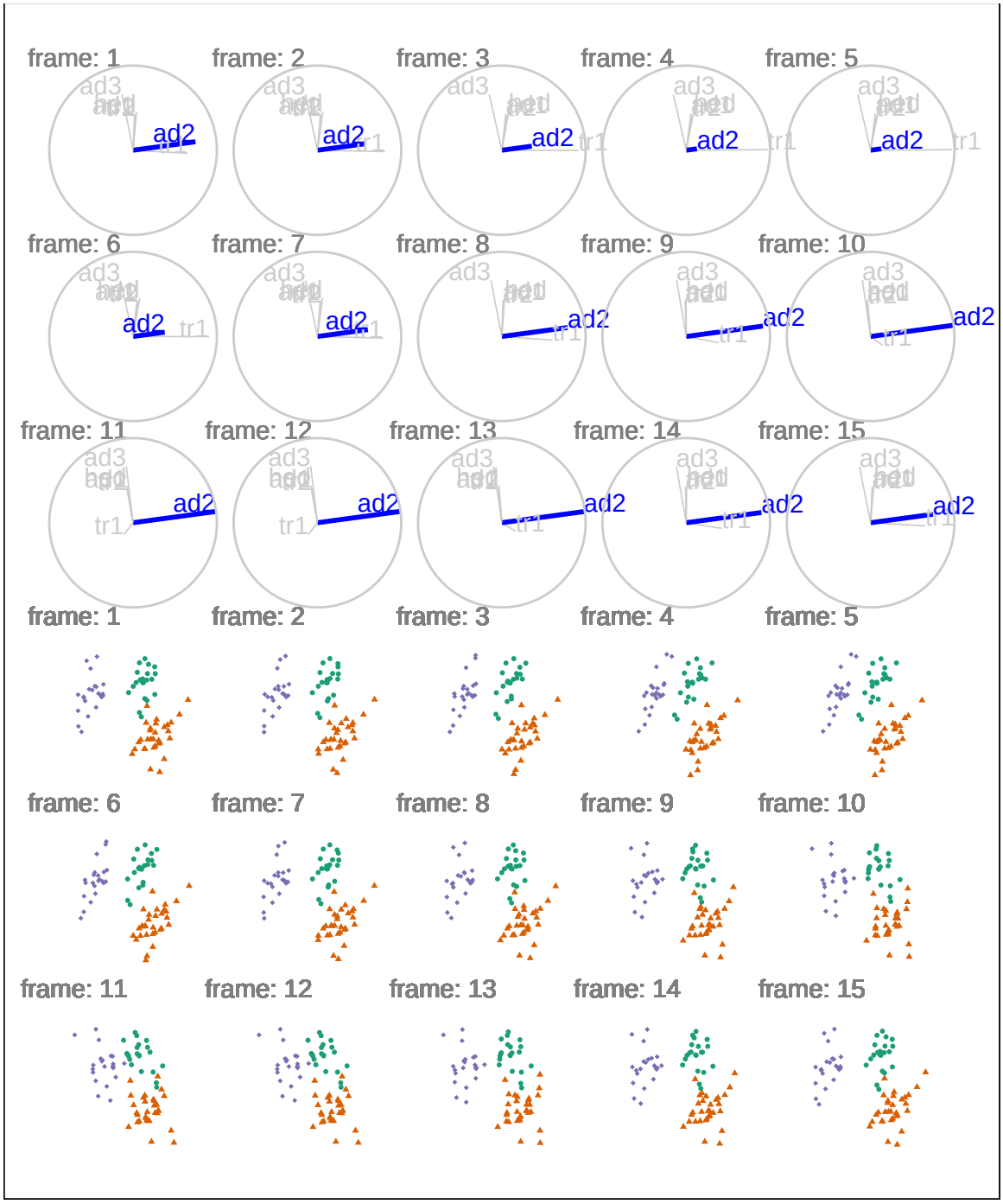


Figure 2.3: *Illustration of a radial manual tour (RO #1), a dynamic version can be viewed at* https://nspyrison.netlify.com/thesis/flea_manualtour_mvar5/.

2.1.4 Path generation

A fundamental aspect of touring is the path of rotation. Of which there are four primary distinctions (Buja et al., 2005): random choice, data driven, precomputed choice, and manual control.

- Random choice, *grand tour*, constrained random walks p-space. Paths are constrained for changes in direction small enough to maintain continuity and aid in user comprehension
  - torus-surface (Asimov, 1985)
  - at-random (Asimov, 1985)
  - random-walk (Asimov, 1985)
  - *local tour* (Wickham et al., 2011), a sort of grand tour on a leash, such that it goes to a nearby random projection before returning to the original position and iterating to a new nearby projection.
- data driven, *guided tour*, optimizing some objective function/index within the projection-space, called projection pursuit (PP) (Hurley and Buja, 1990), including the following indexes:
  - holes (Cook, Buja, and Cabrera, 1993) - moves points away from the center.
  - cmass (Cook, Buja, and Cabrera, 1993) - moves points toward the center.
  - lda (Lee et al., 2005) - linear discriminant analysis, seeks a projection where 2 or more classes are most separated.
  - pda (Lee and Cook, 2010) - penalized discriminant analysis for use in highly correlated variables when classification is needed.
  - convex (Laa and Cook, 2019) - the ratio of area of convex and alpha hulls.
  - skinny (Laa and Cook, 2019) - the ratio of of the perimeter distance to the area of the alpha hull.
  - stringy (Laa and Cook, 2019) - based on the minimum spanning tree (MST), the diameter of the MST over the length of the MST.
  - dcor2D (Grimm, 2017; Laa and Cook, 2019) - distance correlation that finds linear and non-linear dependencies between variables.

  - splines2D (Grimm, 2017; Laa and Cook, 2019) - measure of non-linear dependence by fitting spline models.
  - other user-defined objective function can be implemented with the *tourr* package Wickham et al. (2011).

- Precomputed choice, *planned tour*, in which the path has already been generated or defined.

  - *little tour* (McDonald, 1982), where every permutation of variables is stepped through in order, analogous to a brute-force or exhaustive search.
  - a saved path of any other tour, typically an array of basis targets to interpolate between.

- Manual control, *manual tour*, a constrained rotation on selected manipulation variable and magnitude (Cook and Buja, 1997). Typically used to explore the local area after identifying an interesting feature, perhaps via guided tour.
- *dependence tour*, combination of n independent 1D tours. A vector describes the axis each variable will be displayed on. *ie* $c(1, 1, 2, 2)$ is a 4- to 2D tour with the first 2 variables on the first axis, and the remaining on the second.

  - *correlation tour* (Buja, Hurley, and McDonald, 1987), a special case of the dependence tour, analogous to canonical correlation analysis.

2.1.5 Path evaluation

Consider $d = 2$, then each projection is called a 2-frame (each spanning a 2-plane). Mathematically, a Grassmannian is the set of all possible unoriented 2-frames in p-space, $\mathbf{Gr}(2,\ p)$. Asimov (1985) pointed out that the unique 2-frames of the grand tour approaches $\mathbf{Gr}(2,\ p)$ as time goes to infinity. The *density* of a tour is defined as the fraction of the Grassmannian explored. Ideally an exploring tour will be dense, but the time taken to become dense vastly increases as variable space increase dimensionality. *Rapidity* is then defined as how quickly a tour encompasses the Grassmannian. Due to the random selection of a grand tour it will end up visiting homomorphisms of previous 2-frames, leading sub-optimal rapidity.

The little tour introduced in McDonald (1982), on the other hand is necessarily both dense and rapid, performing essentially an exhaustive search on the Grassmannian. However, this path uninteresting and with long periods of similar projections strung together. The Grassmannian is necessarily large and increases with the square of p. Viewing of the whole Grassmannian is time consuming, and interesting projections are sparse, there was a clear space for computers to narrow the search space.

Guided tours (Hurley and Buja, 1990) optimize an objective function generating path will be relatively small subset of the Grassmannian. As such, density and rapidity are poor measures, however interesting projections are quickly identified. Recently, Laa and Cook (2019), compared projection pursuit indices with the metrics: *smoothness, squintability, flexibility, rotation invariance* and *speed*.

2.1.6 Geometric display by dimensionality

Up to this point this document has discussed 2D scatterplots, which offer a logical display for viewing embeddings of high-dimensional point clouds. However, other geometrics offer perfectly valid projections as well.

- 1D geometrics (geoms)
  - 1D densities: such as histogram, average shifted histograms (Scott, 1985), and kernel density (Scott, 1995).
  - image (pixel): (Wegman, Poston, and Solka, 2001).
  - time series: where multivariate values are independently lagged to view peak and trough alignment. Currently no package implementation, but use case is discussed in (Cook and Buja, 1997).
- 2D geoms
  - 2D density (available on GitHub at `https://github.com/nspyrison/tourr`)
  - $x - y$ scatterplot
- 3D geoms - these geoms do not perform projections in 3 dimensions, but rather are $d = 2$ projections that utilize the manipulation dimension to give depth perception cues.

  - Anaglyphs, sometimes called stereo, where (typically) red images are positioned for the left channel and cyan for the right, when viewed with corresponding filter glasses give the depth perception of the image.
  - Depth, which gives depth cues via aesthetic mappings, most commonly size and/or color of data points.

- d-dimensional geoms

  - Andrews curves (Andrews, 1972), smoothed variant of parallel coordinate plots, discussed below.
  - Chernoff faces (Chernoff, 1973), variables linked to size of facial features for rapid cursory like-ness comparison of observations.
  - Parallel coordinate plots (Ocagne, 1885), where any number of variables are plotted in parallel with observations linked to their corresponding variable value by polylines.
  - Scatterplot matrix (Becker and Cleveland, 1987), showing a triangle matrix of bivariate scatterplots with 1D density on the diagonal.
  - Radial glyphs, radial variants of parallel coordinates including radar, spider, and star glyphs (Siegel et al., 1972).

2.1.7 Tour software implementations

Tours have not yet been widely adopted, this is likely due in part, to the fact that print and static .pdf output does not accommodate dynamic viewing. Conceptual abstraction and technically density have also hampered user growth. Due to small adoption and rapid advancement of technology support and maintenance of such implementations give them a particularly short life span. Despite the small user base, there have been a fair number of software implementing touring in some degree, including:

- spinifex github.com/nspyrison/spinifex – R package, all platforms.
- tourr (Wickham et al., 2011) – R package, all platforms.
- CyrstalVision (Wegman, 2003) – for Windows.
- GGobi (Swayne et al., 2003) – for Linux and Windows.

- DAVIS (Huh and Song, 2002) – Java based, with GUI.
- ORCA (Sutherland et al., 2000) – Extensible toolkit build in Java.
- VRGobi (Nelson, Cook, and Cruz-Neira, 1998) – for use with the C2, tours in stereoscopic 3D displays.
- ExplorN (Carr, Wegman, and Luo, 1996) – for SGI Unix.
- XGobi (Swayne, Cook, and Buja, 1991) – for Linux, Unix, and Windows (via emulation).
- XLispStat (Tierney, 1990) – for Unix and Windows.
- Explor4 (Carr and Nicholson, 1988) – Four-dimensional data using stereo-ray glyphs.
- Prim-9 (Asimov, 1985; Fisherkeller, Friedman, and Tukey, 1974) – on an internal operating system.

2.1.8 Research gaps

Currently there is no compiling software that offers UCS (**RO #1**). This leaves the class of dynamic linear projections without the most precise, fine scale control of rotating p-space. This should be reimplemented with an eye on extensibility and maintainability,

A comparative study outlining the benefits of UCS vs alternatives is also absent from the literature (**RO #2**). The benefits of dynamic linear projections withhold in theory, but a direct comparison with popular alternatives should be made. Barriers to adoption should also be addressed such as: dynamic display is not easy on print and in .pdf documents.

2.2 Multivariate data visualization in 3D

As this research pertains to numeric multivariate data, a wider overview of 3D data visualization is discussed in chapter 2 of Marriott et al. (2018). Terminology for 3D visuals is in the glossary section A.2.

2.2.1 A rocky start

Scientific visualization has readily adopted mixed realities as a large amount of the science exist in 3 spatial dimensions, lending itself well to virtual immersion (Marriott et al., 2018). Data visualization, on the other hand, has been slow to utilize graphics above 2.5D, (and

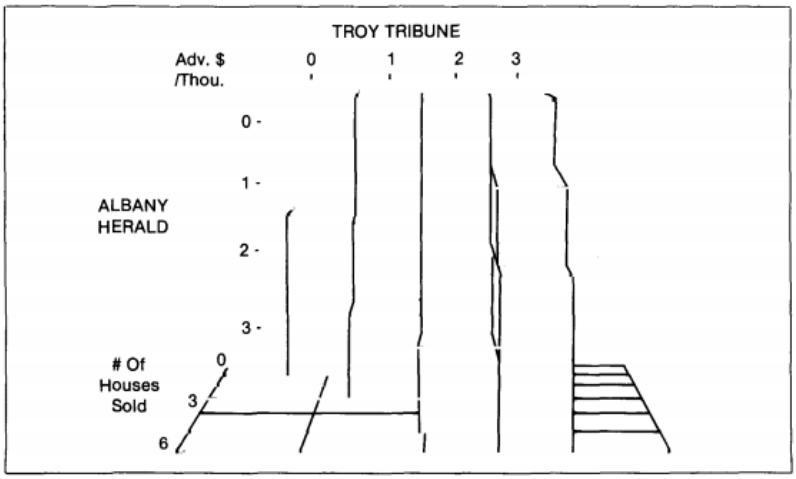


Figure 2.4: *Screen capture of "Figure 7. 3-D Block Model" from Lee, MacLachlan, and Wallace (1986).*

haptic interaction) primarily due to the mixed results of over-hyped of 3D visuals from the 1980's and 90's (Munzner, 2014). However, since then there have been several promising studies suggesting that it is time for data visualization to revisit and adopt 3D visuals for specific combinations of visuals and depth cues.

2.2.2 3D rotated projections vs 3 2D orthogonal projections

3D shapes can be represented by 3 orthogonal 2D views, or rather 3 pairwise projections. When 3D representations are used with binocular cues, they are found to have more accurate perception than 2D counterparts (Lee, MacLachlan, and Wallace, 1986, depicted in figure 2.4).

Between 3D and split view 2D of 3-dimensional economics data Wickens, Merwin, and Lin (1994), depicted in figure 2.5, asked participants integrative questions, finding that participants were faster to answer when questions involved three dimensions, while performance was similar when questions involved fewer dimensions.

Using 3D rotated projection gives more accurate perception (relative to 2D) of a ball suspended above complex box shapes, while combinations of 2D and 3D give the most precise orientation and positioning information (Tory et al., 2006, depicted in figure 2.6).

Sedlmair, Munzner, and Tory (2013), depicted in figure 2.7, tasked users with cluster separation across 2D scatterplot, 2D scatterplot matrices (SPLOMs) and interactive 3D

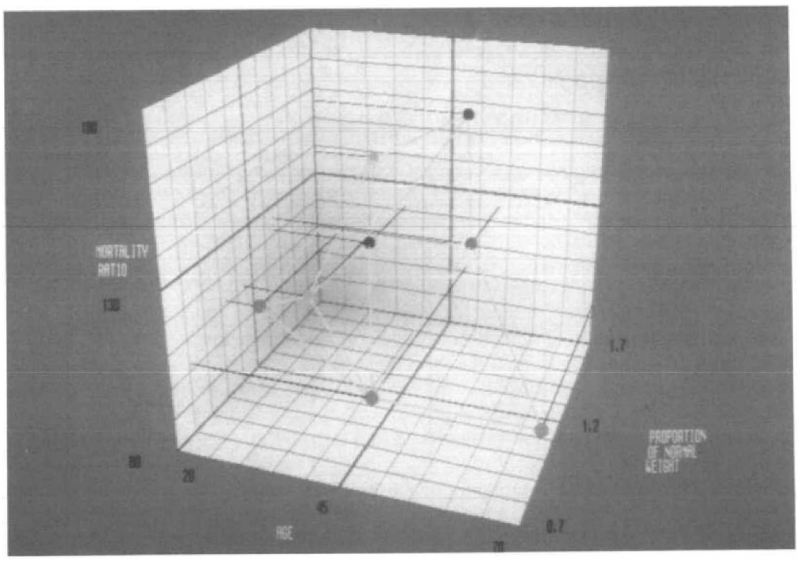


Figure 2.5: *Screen capture of "Figure 5. Example of a mesh display" from Wickens, Merwin, and Lin (1994).*

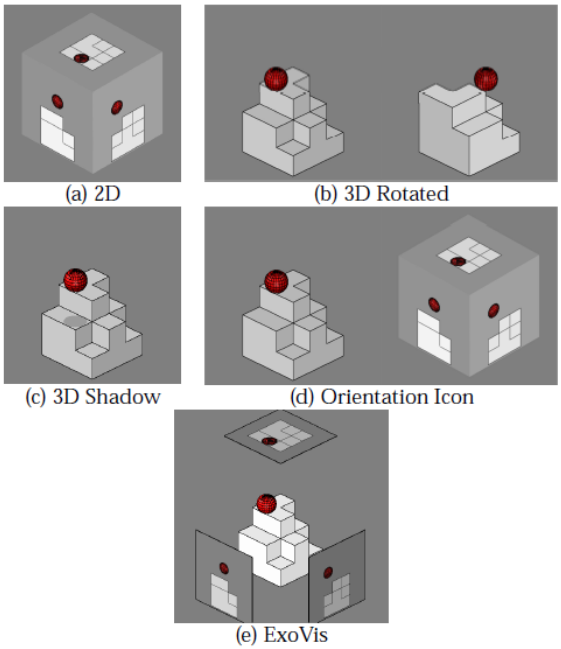


Figure 2.6: *Screen capture from Tory et al. (2006): "Fig. 1 (a) 2D, (b) 3D Rotated, (c) 3D Shadow, (d) Orientation Icon, and (e) ExoVis displays used in Experiment 1 (position estimation). Participants estimated the height of the ball relative to the block shape. In this example, the ball is at height 1.5 diameters above the block shape."*

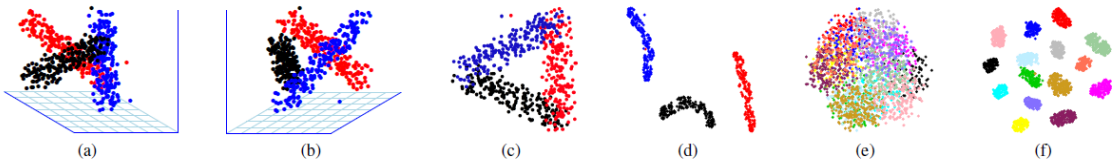


Figure 2.7: *Screen capture of "Figure 5. Example of a mesh display" from Sedlmair, Munzner, and Tory (2013): "Fig. 5. (a)-(d): Screenshots of the entangled dataset* `entangled1-3d-3cl-separate` *designed to show the most possible benefits for i3D. (a),(b) two viewpoints of the same i3D PCA scatterplot. An accompanying video shows the full 3D rotation. (c) 2D PCA projection. (d) t-SNE untangles this class structure in 2D. (e)-(f): 2D scatterplots of the reduced* `entangled2-15d-adjacent` *dataset which we designed to have a ground truth entangled class structure in 15D. (e) Glimmer MDS cannot untangle the classes, neither can PCA and robPCA (see supplemental material). (f) t-SNE nicely untangles and separates the ground truth classes in 2D."*

scatterplots as viewed in monocular 3D from a standard monitor. They conclude that interactive 3D scatterplots perform worse for class separation. This result is surprisingly as the extra dimension theoretically allows for clustering structure to be seen and explored more clearly.

2.2.3 Comparing 3D and 2D embeddings of multivariate data

Nelson, Cook, and Cruz-Neira (1998), depicted in figure 2.8, had $n = 15$ participants perform brushing and touring tasks (identification of clusters, structure, and data dimensionality) in 3D with head-tracked binocular VR. 3D proved to have substantial advantage for cluster identification and some advantage in identifying shape. Brushing did take longer in VR, perhaps due to the lower familiarity of manipulating 3D spaces.

Another study, Gracia et al. (2016), depicted in figure 2.9, performed dimensionality reduction down to 2- and 3D scatterplots, both displayed in monocular 3D on a standard monitor. Users were found to more accurately compare distances between points and identify outliers on 3D scatterplots. However, both tasks were performed slower with use of the 3D scatterplots and statistical significance was not reported.

Wagner Filho et al. (2018), depicted in figure 2.10, performed an $n = 30$ empirical study of PCA embedded projections, and perception error across 4 tasks and 3 display types: 2D, 3D, and immersive. Overall task error was less in 3D and immersive relative to 2D.

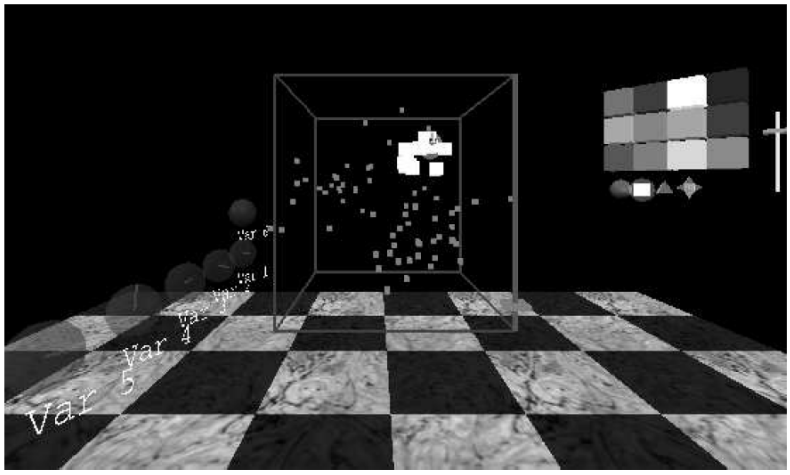


Figure 2.8: *Screen capture from Nelson, Cook, and Cruz-Neira (1998): "Figure 4: This is a picture of a 3-D room, running VRGobi. Data is plotted in the center, with painting tools to the right and variable spheres to the left. In the viewing box the data can be seen to contain three clusters, and one is being brushed."*

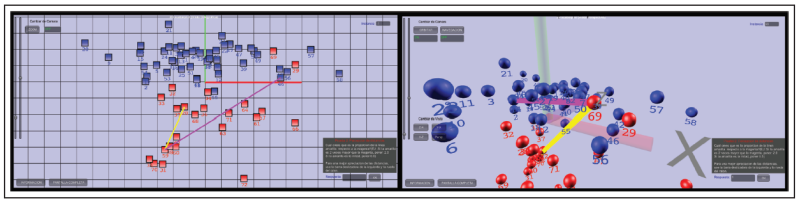


Figure 2.9: *Screen capture from Gracia et al. (2016): "Figure 5. Distance perception test. Left-hand image: 2D version. Here, the yellow line could be perceived as roughly twice the length of the magenta line, thus the value to be introduced should be approximately 2.0. Right-hand image: 3D version. Here, the inclusion of an extra dimension could provide new information about the relation, in terms of distances, between both lines."*

According to the user Likert-scale survey 2D is slightly easier to navigate and slightly more comfortable, while, 3D and immersive display are slightly easier to interact and moderately easier to find information.

2.2.4 Immersive analytics platform in VR

Immersive analytics is a emerging technology where data visualization and analysis is facilitated in an intuitive, immersive virtual reality environment. An example of which is shown in Cordeil et al. (2017) introduces a collaborative space for immersive data analysis. Where axes are displayed and intuitively interacted with while respond to proximity to other variable axes and popping into place changing the resulting geometric display. For example, three variables can be placed as the x, y, z axis for a 3D scatterplot or stood up

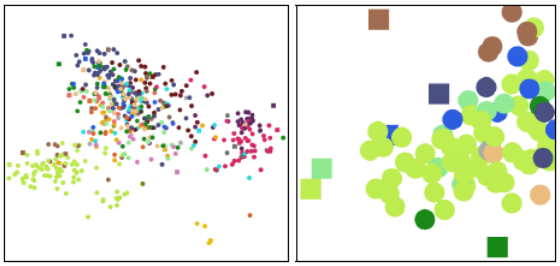


Figure 4: In the *2D* condition, data points are distributed along screen space (left), and the user is allowed to zoom and pan (right).

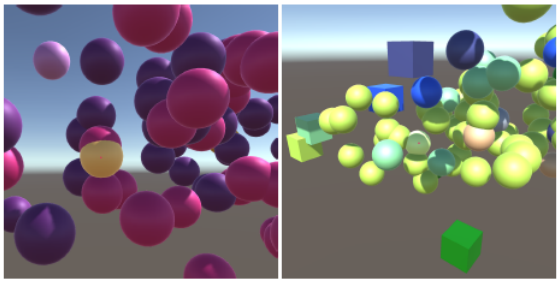


Figure 5: In the *3D* conditions, the user is allowed to freely navigate through the data, which is distributed along a 3D virtual environment.

Figure 2.10: *Screen capture from Wagner Filho et al. (2018), original captions contained in capture.*

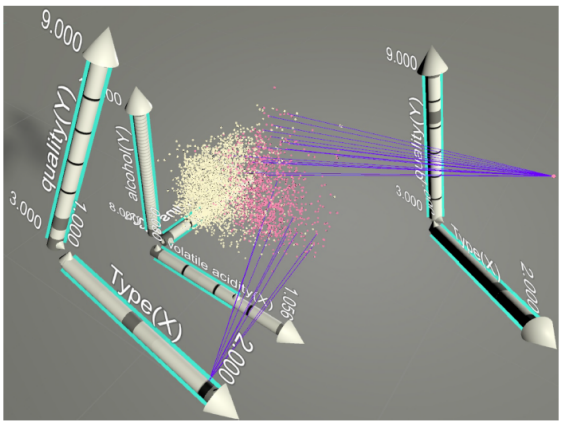


Figure 2.11: *Screen capture of figure 15 from Cordeil et al. (2017).*

right next to each other for a parallel coordinate plot. The subsequent work in Cordeil (2019) builds from the previous reference, and refines it for the next iteration in interactive, scalable data visualization in virtual spaces.

2.2.5 Research gaps

When comparing between 3D and 2D orthogonal views studies in general show that perception accuracy is better in 3D, though manipulation speed is generally slower. The

speed discrepancy is confounded by the difference in users familiar with manipulating 2D vs 3D spaces (Lee, MacLachlan, and Wallace, 1986; Wickens, Merwin, and Lin, 1994; Tory et al., 2006; counter example Sedlmair, Munzner, and Tory, 2013).

Similar results have been shown in static, 3D projected spaces (Gracia et al., 2016; Wagner Filho et al., 2018) and in dynamic $d = 2$ embedded spaces depicted in immersive 3D (Nelson, Cook, and Cruz-Neira, 1998). The literature stops short of the dynamic $d = 3$ linear projections. With Modern VR hardware advancement, so too does the, quality, resolution, and prevalence of VR advance, making VR more easily accessible than ever. It's time to view dynamic 3D projections with immersive spaces and quantify the corresponding benefits (**RO #3 & 4**).

Chapter 3

Work in progress

Implementing UCS in low dimensions is fundamental to the extension of the UCS into 3D space. The implementation of such, forms the foundation for future work addressed in the remaining research objectives.

3.1 RO #1) How can UCS be implemented in 1- and 2D projections?

This section covers the work done in the last year implementing USC via *manual tours*. Following experimental design methodology this work implements the manual tour as described in Cook and Buja (1997) and allows users to rotate a selected variable into and out of a 2D projection of high-dimensional space for user-controlled steering (UCS). The primary use is to determine the sensitivity of structure visible in a projection to the contributions of a variable. This is particularly powerful for exploring the local structure once a feature of interest has been identified by a guided tour (Cook et al., 1995) for example. The algorithm for a manual tour allows rotations in horizontal, vertical, oblique, angular and radial directions. In the algorithm below focuses on radial rotation.

Section 3.2 explains the algorithm using a toy dataset. A wider discussion of implementation as an *R* package and application to high energy physics data (Wang et al., 2018;

Cook, Laa, and Valencia, 2018) is attached in appendix B formatted as a paper that will be submitted to The R Journal.

3.2 Algorithm

Creating a manual tour animation requires these steps:

1. Provided with a 2D projection, choose a variable to explore. This is called the "manip" variable.
2. Create a 3D manipulation space, where the manip variable has full contribution.
3. Generate a rotation sequence which zeros coefficient and increases it to 1 before returning to the initial position.

These steps are described in more detail below.

3.2.1 Notation

Start with multivariate data in an $n \times p$ numeric matrix, and an orthonormal d-dimensional basis set is describing the projection from $p-$ to $d-$space.

$$\mathbf{X}_{[n,\,p]} = \begin{bmatrix} X_{1,\,1} & \dots & X_{1,\,p} \\ X_{2,\,1} & \dots & X_{2,\,p} \\ \vdots & \ddots & \vdots \\ X_{n,\,1} & \dots & X_{n,\,p} \end{bmatrix}$$

$$\mathbf{B}_{[p,\,d]} = \begin{bmatrix} B_{1,\,1} & \dots & B_{1,\,d} \\ B_{2,\,1} & \dots & B_{2,\,d} \\ \vdots & \ddots & \vdots \\ B_{p,\,1} & \dots & B_{p,\,d} \end{bmatrix}$$

The algorithm is primarily operating on the projection basis and utilizes the data only when making a display for computational efficiency. A more comprehensive list of tour notation is given in A.1.

3.2.2 Toy data set

The flea data from Lubischew (1962) will be used as an example dataset. The data contains 74 observations across 6 variables, corresponding to physical measurements of the insects. Each observation belonging to one of three species.

A guided tour on the flea data is conducted by optimizing on the `holes` index (Cook, Swayne, and Buja, 2007). In a guided tour the data the projection sequence is shown by optimizing an index of interest. The `holes` index is maximized by when the projected data has a lack of observations in the center. Figure 3.1, shows an optimal projection of this data. The left plot displays the projection basis, while the right plot shows the projected data. The display of the basis has a unit circle with lines showing the horizontal and vertical contributions of each variable in the projection. Here is primarily `tars1` and `aede2` contrasting the other four variables. In the projected data there are three clusters, which have been colored, although not used in the optimization. The question that will be explored in the explanation of the algorithm is how important is `aede2` to the separation of the clusters.

The left frame of figure 3.1 shows the reference frame for the basis. It describes the X and Y contributions of the basis as it projects from the 6 variable dimensions down to 2. The right side shows how the data looks projected through this basis. You can project a single basis at any time through the matrix multiplication $\mathbf{X}_{[n,\ p]} * \mathbf{B}_{[p,\ d]} = \mathbf{P}_{[n,\ d]}$ to such effect.

3.2.3 Step 1 Choose variable of interest

Select a manipulation variable, k. Initialize a zero vector e and set the k-th element set to 1.

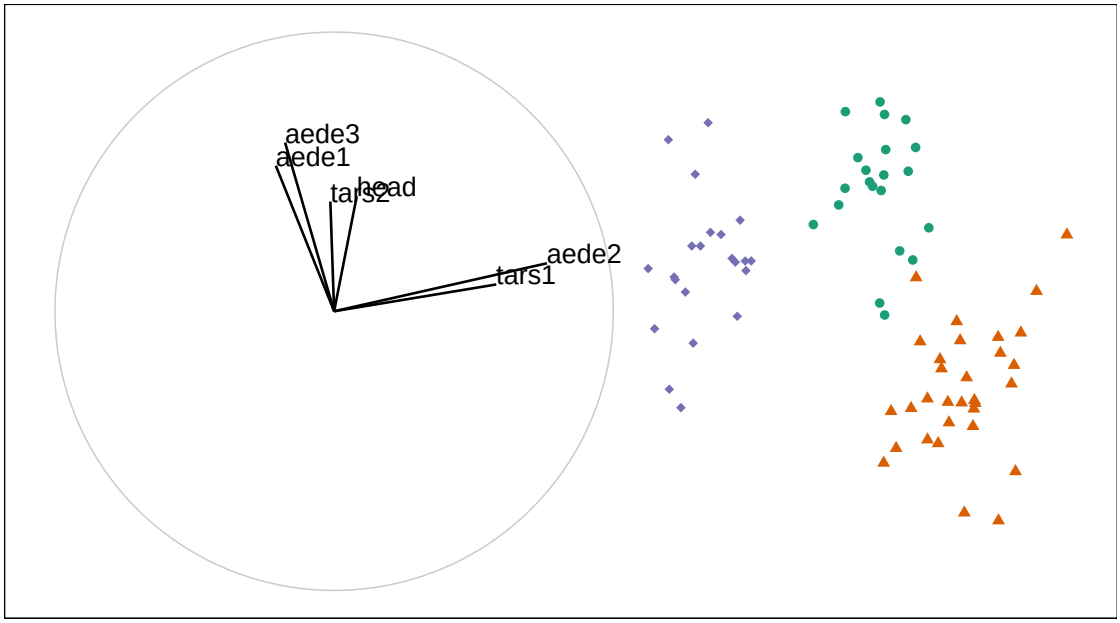


Figure 3.1: *Basis reference frame (left) and projected data (right) of standardized flea data. Basis identified by holes-index guided tour. The variables 'aede2' and 'tars1' contribute mostly in the X direction, whereas the other variables contribute mostly in the Y direction. Select 'aede2' as the manipulation variable to see how the structure of the projection changes.*

$$\mathbf{e}_{[p,\,1]} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix}$$

In figure 3.1, above, notice that the variables `tars1` and `aede2` are almost orthogonal to the other 4 variables and control almost all of the variation in the x axis of the projection. `Aede2` has a larger contribution and is select it as the manip variable.

3.2.4 Step 2 Create the manip space

Use the Gram-Schmidt process to orthonormalize the concatenation of the basis and e yielding the manip space.

$$\begin{aligned}\mathbf{M}_{[p,\,d+1]} &= Orthonormalize_{GS}(\mathbf{B}_{[p,\,d]}|\mathbf{e}_{[p,\,1]}) \\ &= Orthonormalize_{GS}\left(\begin{bmatrix} B_{1,1} & \dots & B_{1,d} \\ B_{2,1} & \dots & B_{2,d} \\ \vdots & \ddots & \vdots \\ B_{k,1} & \dots & B_{k,d} \\ \vdots & \ddots & \vdots \\ B_{p,1} & \dots & B_{p,d} \end{bmatrix} \middle| \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix}\right)\end{aligned}$$

Adding an extra dimension to our basis plane allows for the manipulation of the specified variable. Orthonormalizing rescales the new vector, while leaving the first d variables identical to the basis. An illustration of such can been seen below in figure 3.2.

Imagine being able to grab hold of the red axis and rotate it changing the projection onto the basis plane. This is what happens in a manual tour. For a radial tour, fix θ, the angle within the blue plane, and vary the ϕ, the angle orthogonal to the blue projection plane. The user controlling these angles changes the values of the coefficient the manip variable and performs a constrained rotation on the remaining variables.

3.2.5 Step 3 Generate rotation

Define a set of values for ϕ_i, the angle of out of plane rotation, orthogonal to the projection plane. This corresponds to the angle between the red manipulation axis and the blue plane in figure 3.2.

For i in 1 to n_slides:

For each ϕ_i, post multiply the manipulation space by a rotation matrix, producing, **RM**, the rotated manip space.

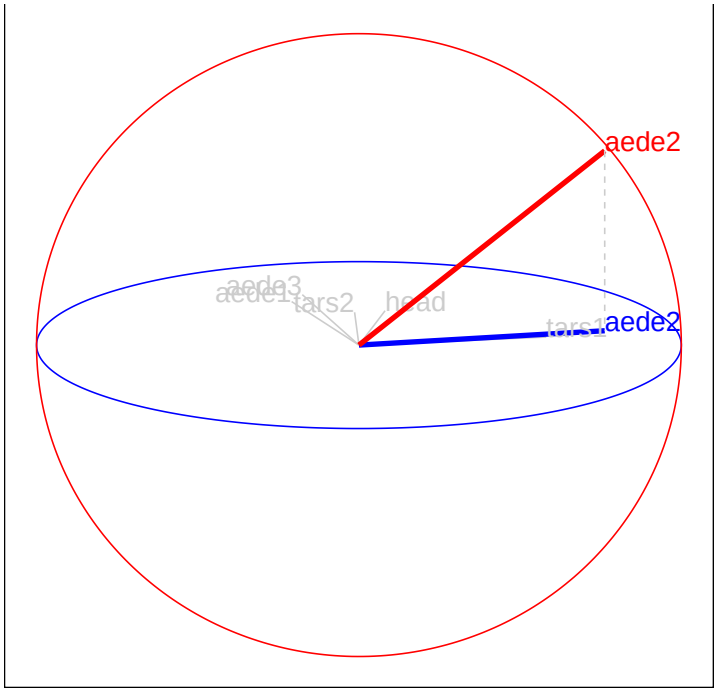


Figure 3.2: *Manipulation space for controlling the contribution of aede2 of standardized flea data. Basis was identified by holes-index guided tour. The out of plane axis, in red, shows how the manipulation variable can be rotated, while other dimensions stay embedded within the basis plane.*

$$\mathbf{RM}_{[p,\,d+1,\,i]} = \mathbf{M}_{[p,\,d+1]} * \mathbf{R}_{[d+1,\,d+1]}$$

$$= \begin{bmatrix} M_{1,1} & \dots & M_{1,d} & M_{1,d+1} \\ M_{2,1} & \dots & M_{2,d} & M_{2,d+1} \\ \vdots & \ddots & \vdots & \\ M_{p,1} & \dots & M_{p,d} & M_{p,d+1} \end{bmatrix}_{[p,\,d+1]} * \begin{bmatrix} c_\theta^2 c_\phi s_\theta^2 & -c_\theta s_\theta (1-c_\phi) & -c_\theta s_\phi \\ -c_\theta s_\theta (1-c_\phi) & s_\theta^2 c_\phi + c_\theta^2 & -s_\theta s_\phi \\ c_\theta s_\phi & s_\theta s_\phi & c_\phi \end{bmatrix}_{[3,\,3]}$$

For the $d = 2$ case:

Where:

θ is the angle that lies on the projection plane (*ie*. on the xy plane)

ϕ is the angle orthogonal to the projection plane (*ie*. in the z, direction)

c_θ is the cosine of θ

c_ϕ is the cosine of ϕ

s_θ is the sine of θ

s_ϕ is the sine of ϕ

In application: compile the sequence of ϕ_i and create an array for each rotated manipulation space. ϕ is the angle of change relative to the ϕ_1, the transformation ϕ_i - ϕ_1 to useful to think about ϕ relative to the basis plane. If the manip variable doesn't move as expected this is the first place to check.

Figure 3.3 illustrates the sequence with 15 projected bases and highlight the manip variable on top, while showing the corresponding projected data points on the bottom. A dynamic version of this tour can be viewed online at https://nspyrison.netlify.com/thesis/flea_manualtour_mvar5/, (may take a moment to load).

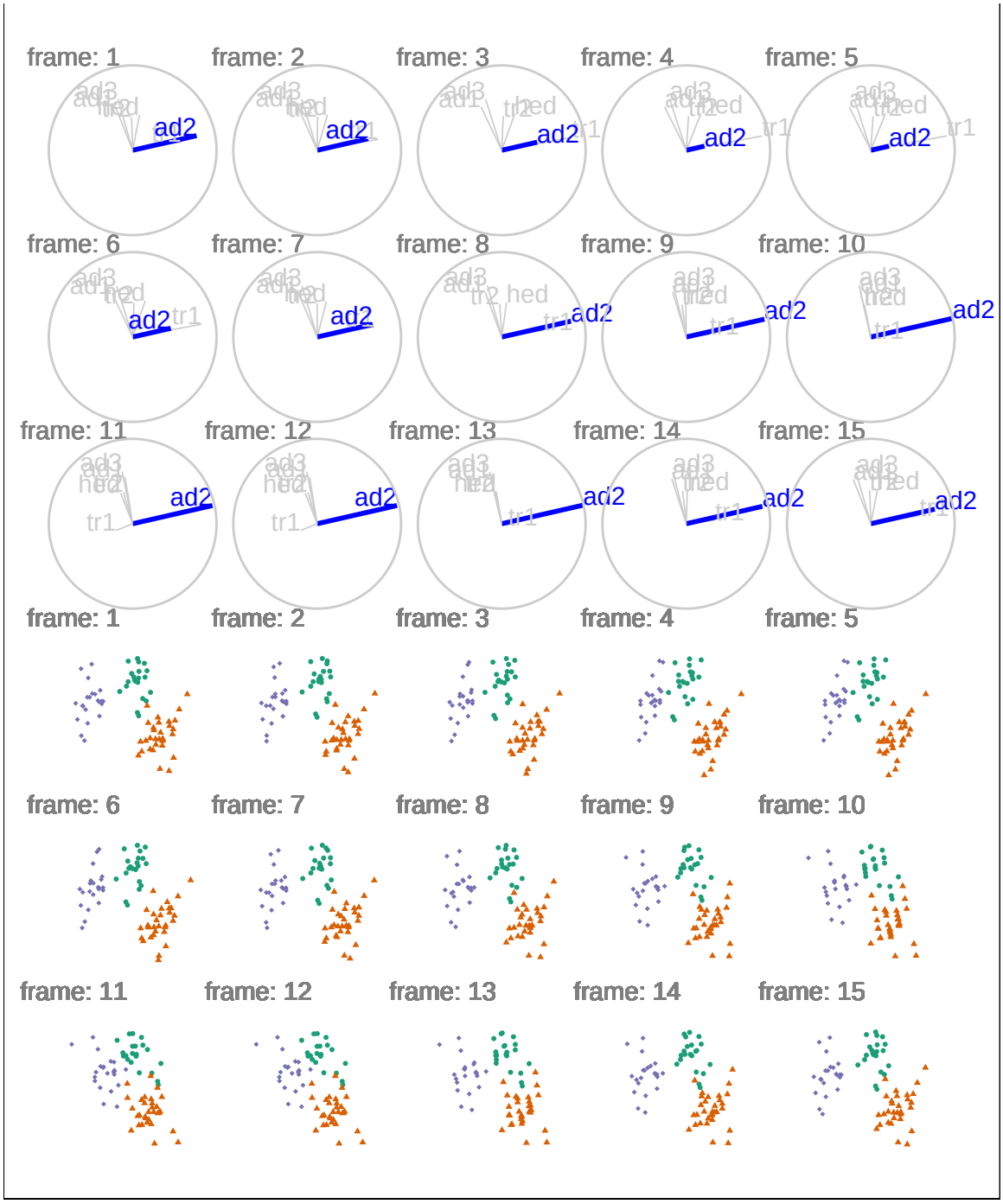


Figure 3.3: *Rotated manipulation spaces, a radial manual tour manipulating* `aded2` *of standardized flea data. The manipulation variable,* `aede2`*, extends from its initial contribution to a full contribution to the projection before decreasing to zero, and then returning to its initial state. A dynamic version can be viewed at* `https://nspyrison.netlify.com/thesis/flea_manualtour_mvar5/`.

3.3 Display projection sequence

To get back to data-space pre-multiply the rotated manip space by the data for the projection in data-space.

$$\mathbf{P}_{[n,\, d+1]} = \mathbf{X}_{[n,\, p]} \ * \ \mathbf{RM}_{[p,\, d+1]} \tag{3.1}$$

$$= \begin{bmatrix} X_{1,\,1} & \dots & X_{1,\,p} \\ X_{2,\,1} & \dots & X_{2,\,p} \\ \vdots & \vdots & \vdots \\ X_{n,\,1} & \dots & X_{n,\,p} \end{bmatrix}_{[n,\, p]} * \begin{bmatrix} RM_{1,\,1} & RM_{1,\,2} & RM_{1,\,3} \\ RM_{2,\,1} & RM_{2,\,2} & RM_{2,\,3} \\ \vdots & \vdots & \vdots \\ RM_{p,\,1} & RM_{p,\,2} & RM_{p,\,3} \end{bmatrix}_{[p,\, d+1]} \tag{3.2}$$

Plot the first $d = 2$ variables from each projection in sequence for an XY scatterplot. The remaining variable is sometimes linked to a data point aesthetic to produce depth cues used in conjunction with the XY scatterplot.

3.3.1 Storage and sharing

Storing each data point for every frame with the overhead dynamic graphics is very inefficient. In the same way that efficiency is gain by performing math on the bases, the same approach suggested for storage and sharing tours. Consider a radial manual tour, the salient features can be stored in 3 bases, where ϕ is at its starting, minimum, and maximum values. The frames in between can be interpolated later by supplying angular speed.

Chapter 4

Future work

4.1 RO #2) Does 2D UCS provide benefits over alternatives?

High dimensional data and models are ubiquitous but viewing them in data space is not trivial. This work quanifies various measurements of 2D UCS implemented in RO #1, and commonly used alternatives. All comparison groups are unsupervised (agnostic of clustering) static, single embeddings in lower dimension of the data, and would include:

- **Principal Component Analysis (PCA)**, is a linear transformation that orients linear combinations of the variables into basis components and orders them according to amount of variation. The first principal component is the linear combination that explains the most variation with the second explaining the most of the remaining variation and is orthogonal to the all previous components, and so on.
- **Multi-Dimensional Scaling (MDS)**, non-linear dimension reduction that compares pairwise distances between observations.
- **t-distributed neighbor embeddings (tSNE)**, a nonlinear technique that iterates epochs of: 1) constructing a probability distributions for selecting neighboring data and 2), minimizing Kullback-Leibler divergence (a measure of relative entropy).

Unfortunately, static linear projections necessarily cut variation in the components not shown, while non-linear techniques lose transparency back to the original variable-space. Tours preserve this transparency to variable-space and keeps variation in tack. Providing user-controlled steering of tour should allow for finer structural exploration than the alternatives.

The tentative methodology for this future work is a **case study** between UCS and leading alternatives. This will be a sufficient comparison if there are enough quantifiable measurements across the different techniques. However, if enough measurements are not comparable across technique an empirical study analogous to the study suggested in RO # 4 will be considered. Design space includes data sets, techniques, and measures of comparison.

4.2 RO #3) How can UCS be extended to 3D?

The literature has shown positive results for improved accuracy and precision for 3D displays. Dynamic linear projections should have similar gains in $d = 2$ projections, and the additional dimension should allow for improved perception of surfaces and dynamic viewing of $d = 3$ projections. The work done in RO #1) will be extended to these uses.

The work done in Cordeil et al. (2017) creates a collaborative space for people to engage in immersive data analysis. The subsequent Cordeil (2019) created the immersive analytics toolkit (IATK), which generalized data visualization in the Unity game engine. By integrating dynamic linear touring in 3D with the IATK offers a consolidated user interface that can be used across various display devices in RO #4.

This is an **exploratory design**, first the *R* package spinifex will be extended to 3D, and then calling it via the *Unity* package IATK for rendering in 3D VR and offers a compatible front end to be used across display devices.

4.3 RO #4) Does UCS in 3D displays provide perception benefits over 2D displays?

The bulk of past touring endeavors have existed whole in 2D, with the exceptions of Nelson, Cook, and Cruz-Neira (1998) and Arms, Cook, and Cruz-Neira (1999) whom performed a small ($n = 15$) experimental study comparing tasks performed across 2D and 3D touring displays. The XGobi interface was used on a standard 2D monitor while VRGobi (on the C2 setup) was used with head-tracked binocular VR. The 3 accuracy tasks: clustering, intrinsic data dimensionality, and radial sparseness were recorded along with the speed of a brushing data. Accuracy was the same for the dimensionality task, while 3D display outperformed 2D on clustering, and even more so on the radial sparsity. However, time taken to brush a cluster was less than half the time in 2D display as compared with 3D.

The results of Wagner Filho et al. (2018), Nelson, Cook, and Cruz-Neira (1998) and, Arms, Cook, and Cruz-Neira (1999) cast positive light on 3D spaces improving the perception of embeddings of high-dimensional data, while others have found the same for data already in three dimensions. After implementing touring and UCS in 3D spaces (RO #3), the next step is to quantify the effects across display type.

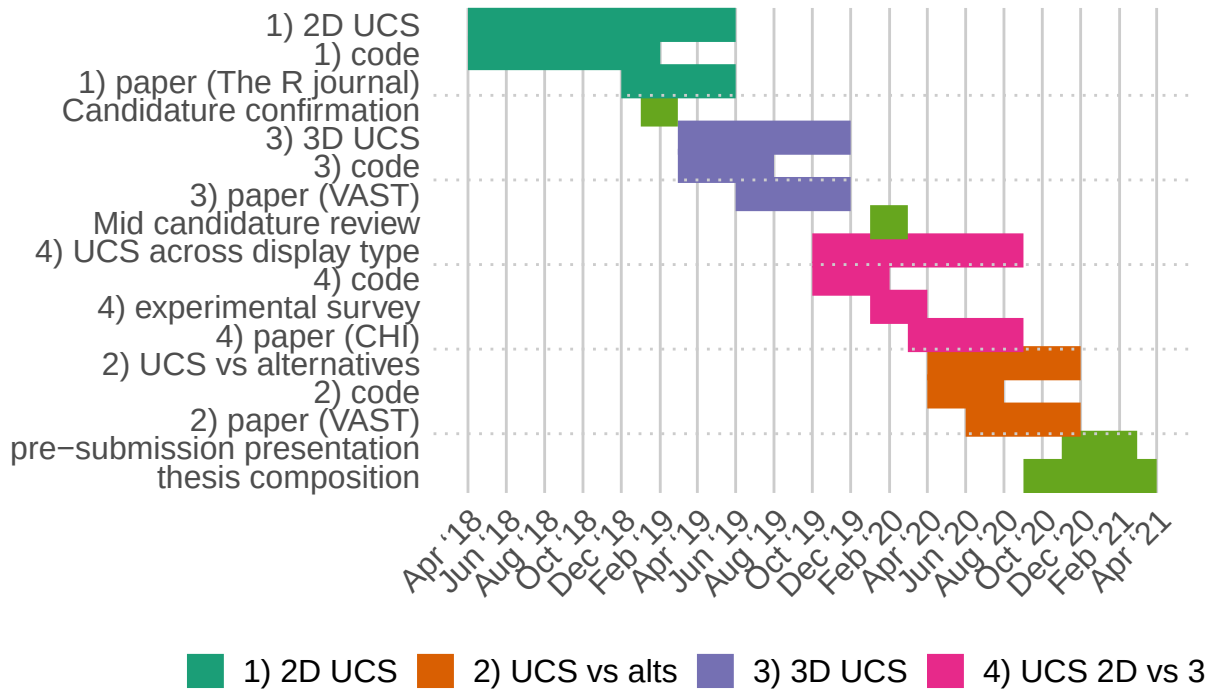
I plan to test the efficacy of doing so with the following **empirical study**: *randomized full factorial design*, where every participant will complete every task on every display device. Task order and display device will be randomly assigned to minimize learning bias. Correctness and speed of tasks will be recorded alongside demographic data and subjective 5-point Likert scale survey. A lineup-type model as outlined in Hofmann et al. (2012) may be employed to quantify "best" display device.

Tasks will test perception of structure and surface, varying across manipulation variable. All tasks will be conducted across at least three display devices: standard 2D monitor, stereoscopic 3D monitor (on a zSpace 200), and head-mounted VR goggles (HTC VIVE). User interface will be standardized across display device. The data explored will be of high energy physics experiments already being discussed in publication (Wang et al., 2018; Cook, Laa, and Valencia, 2018) and looked at in 2D UCS in appendix B.

Chapter 5

PhD schedule

5.1 Timeline



Note RO #2 logically would fit before RO #3 & 4, but is of lower impact and retrospective case study. I move this research to the end to give the other research objectives the priority.

5.2 Accompanying documents

- FIT 5144 hours
  - >120 hours **Tracked, awaiting mandatory events**, due at mid-candidature review
- WES Academic record
  - FIT6021: 2018 S2, **Completed** with Distinction
  - FIT5144: 2019 S1+2, **Upcoming**, due at mid-candidature review
  - FIT5113: 2018 S2, **Exemption submitted**, forwarded 14/02/2019
- myDevelopment - IT: Monash Doctoral Program - Compulsory Module
  - Monash Graduate Research Student Induction: **Completed**
  - Research Integrity - Choose the Option most relevant: **Completed** (2 required of 4)
  - Faculty Induction: **Content unavailable** (25/02/2019: "Currently being updated and will be visible in this section soon")

Chapter 6

Source code

This article was created in `R` (R Core Team, 2018), using `bookdown` (Xie, 2016) and `rmarkdown` (Xie, Allaire, and Grolemund, 2018), with code generating the examples inline.

Following good practice of version control and reproduction, the source files can be found at: github.com/nspyrison/confirmation/.

Appendix A

Glossary

A.1 Tour notation

Terminology varies across articles. In my work, I use the following:

- n, number of observations in the data.
- p, number of numeric variables, the dimensionality of data space.
- d, dimensionality of projection space.
- $\mathbf{X}_{[n,\ p]}$, a data matrix in variable-space, $\mathbf{X} \in \mathbb{R}^p$. Typically centered, scaled, and optionally sphered.
- $\mathbf{B}_{[p,\ d]}$, orthonormal basis set (d linear combinations of p variables, each at a right angle, to the others, with a norm of 1), defining the axes directions for the projection from $p-$ to $d-$space.
- $\mathbf{Y}_{[n,\ d]}$, projected data matrix in projection-space, $\mathbf{Y} \in \mathbb{R}^d$.
- For projections down to 1- and 2D, it is common to display each variables contribution and direction on its own axis (1D) or relative to a unit circle (2D), this is referred to as basis axes or sometimes the reference frame.
- Geometric objects are referred to in generalized dimensions; the use of plane is not necessarily a 2D surface, but a hyper-plane in the arbitrary dimensions of the projection space.

A.2 Data visualization terminology

- 2D - representation of data in 2 dimensions, without use of depth perception cues and minimal aesthetic mapping (such as color, size, and height) to data points.
- 2.5D - Following the definition given in Ware (2000): visualizations that are essentially 2D but select depth cues are used to provide some suggestion of 3D. However, the term 2.5D is commonly used for several meanings *due to the ambiguous use of 2.5D, this document errs on the side stating 3D with descriptions of depth cues used*.
- 3D - visualizations of 3 dimensions with a liberal use of depth cues unless otherwise qualified.
- Depth perception cues - an indication that indicates depth to an observer, including:
  - linear perspective - the property of parallel lines converging on a vanishing point.
  - aerial perspective - objects that are far away have lower contrast and color saturation due to light scattering in the atmosphere.
  - occulation (or interposition) - where closer objects partially block the view of further objects.
  - motion perspective/parallax - closer objects, move across the field of view faster than further objects.
  - accommodation - the change of focal length due to change in the shape of the eye. Effective for distances of less than 2 meters.
  - binocular stereopsis/disparity - the use of 2 images of slightly varied angles from the horizontal distance of the eyes. The disparity for distant objects is small, but it is significant for nearby objects.
  - binocular convergence - The ocular-motor cue due to stereopsis focusing on the same objects. Convergence is effective for distances up to 10 meters.
- Virtual reality (VR) - computer generated display of virtual spaces in place of physical vision.
- Augmented reality (AR) - computer generated display of information overlaid on a physical space.

- Mixed reality (MR) - any degree of virtual or augmented reality.
- Scatterplot matrices (SPLOM) - matrix display of pair-wise 2D scatterplots with 1D density on the diagonal.

Appendix B

Using animation to explore sensitivity of structure in a low-dimensional projection of high-dimensional data with user controlled steering

The content contained in this appendix document is work done in the last year of my research and currently formatted as a paper to be submitted to the R Journal.

B.1 Abstract

The tour algorithm, and its various versions provide a systematic approach to viewing low-dimensional projections of high-dimensional data. It is particularly useful for understanding multivariate data, and useful in association with techniques for dimension reduction, supervised and unsupervised classification. The *R* package *tourr* provides many methods for conducting tours on multivariate data. This paper discusses an extension package which adds support for the manual tour, called *spinifex*. It is particularly usefully

for exploring the sensitivity of structure discovered in a projection by a guided tour, to the contribution of a variable. *Spinifex* utilizes the animation packages *plotly* and *gganimation* to allow users to rotate the selected variable into and out of a chosen projection.

Keywords: grand tour, projection pursuit, manual tour, high dimensional data, multivariate data, data visualization, statistical graphics, data science, data mining.

B.2 Introduction

A tour is a multivariate data analysis technique in which is a sequence of linear (orthogonal) projections into a lower subspace in which $p-$space is rotated across time. Each frame of the sequence corresponds to a small change in the projection for a smooth transition to persevere the object continuity.

Multivariate data analysis can be broken into 2 groups: linear and non-linear transformations. Like PCA and LDA, touring uses linear dimension reduction that maintain transparency back to the original variable-space. PCA and LDA are typically represented with single static projection as a 2- or 3D scatterplot, inherently losing the variation held with the high components, whereas touring keeps the information in tack by showing the other components across time. Non-linear transformations such as tSNE (t-distributed stochastic nearest neighbor embeddings), MDS (multi-dimension scaling), and LLE (local linear embedding) distort the parameter-space which lacks transparency back to the original parameter-space. They show more extreme separation in embeddings, but the variable opacity can be a non-starter for many uses.

There are many ways that a tour path can be generated, we will focus on one, the manual tour. The manual tour was described in Cook and Buja (1997) and allows a user to rotate a variable into and out of a 2D projection of high-dimensional space. This will be called user-controlled steering (UCS). The primary purpose is to determine the sensitivity of structure visible in a projection to the contributions of a variable. Manual touring can also be useful for exploring the local structure once a feature of interest has been identified, for example, by a guided tour (Cook et al., 1995). The algorithm for a manual tour allows rotations in horizontal, vertical, oblique, angular and radial directions. Rotation in a radial

direction, would pull a variable into and out of the projection, which allows for examining the sensitivity of structure in the projection to the contribution of this variable. This type of manual rotation is the focus of this paper.

A manual tour relies on user input, and thus has been difficult to program in R. Ideally, the mouse movements of the user are captured, and passed to the computations, driving the rotation interactively. However, this type of interactivity is not simple in R. This has been the reason that the algorithm was not incorporated into the *tourr* package. Spinifex utilizes two new animation packages, *plotly* (Sievert, 2018) and *gganimate* (Pedersen and Robinson, 2019), to display manual tours or other saved tours. From a given projection, the user can choose which variable to control, and the animation sequence is generated to remove the variable from the projection, and then extend its contribution to be the sole variable in one direction. This allows the viewer to assess the change in structure induced in the projection by the variable's contribution.

The paper is organized as follows. Section 3.2 explains the algorithm using a toy dataset. Section B.4 illustrates how this can be used for sensitivity analysis. The last section, B.6 summarizes the work and discusses future research.

B.3 Algorithm

Algorithm and example is discussed above in section 3.2 and is purposefully removed here.

–>

B.4 Application

In a recent paper, Wang et al. (2018), the authors aggregate and visualize the sensitivity of hadronic experiments. The authors introduce a new tool, PDFSense, to aid in the visualization of parton distribution functions (PDF). The parameter-space of these experiments lies in 56 dimensions, $\delta \in \mathbb{R}^{56}$, and are presented in this work in 3D subspaces of the 10 first principal components and non-linear embeddings.

The work in Cook, Laa, and Valencia (2018) applies touring for discern finer structure of this sensitivity. Table 1 of Cook et. al. summaries the key findings of PDFSense & TFEP

(TensorFlow embedded projection) and those from touring. The authors selected the 6 first principal components, containing 48% of the variation held within the full data when centered, but not sphered. This data contained 3 clusters: jet, DIS, and VBP. Below pick up from the projections used in their figures 7 and 8 (jet and DIS clusters respectively) and apply manual tours to explore the local structure with finer precision.

B.4.1 Jet cluster

The jet cluster is of particular interest as it contains the largest data sets and is found to be important in Wang et al. (2018). The jet cluster resides in a smaller dimensionality than the full set of experiments with 4 principal components explaining 95% of its variation (Cook, Laa, and Valencia, 2018). We subset the data down to ATLAS7old and ATLAS7new to narrow in on 2 groups with a reasonable number of observations and occupy different parts of the subspace. Below, we perform radial manual tours on various principal components within this scope. In PC3 and PC4 are manipulated in figure B.1 and figure B.2 respectively. Manipulating PC3, where varying the angle of rotation brings interesting features in-to and out of the center mass of the data, is interesting than the manipulation of PC4, where features are mostly independent of the manip var.

Jet cluster manual tours manipulating each of the principal components can be viewed from the links: PC1, PC2, PC3, and PC4.

B.4.2 DIS cluster

We perform a manual tour on this data, manipulating PC6 as depicted in figure B.3. Looking at several frames we see that DIS HERA lie mostly on a plane. When PC6 has full contributions, we see the dimuon SIDIS in purple is almost orthogonal to the DIS HERA (green). Yet the contribution of PC6 is zeroed the dimuon SIDIS data occupy the same space as the DIS HERA data. A dynamic version of this manual tour can be found at: `https://nspyrison.netlify.com/thesis/discluster_manualtour_pc6/`. The page takes a bit to load, as the animation is several megabytes.

This is different story than if we had selected a different variable to manipulate. In figure B.4 we manipulate PC2.

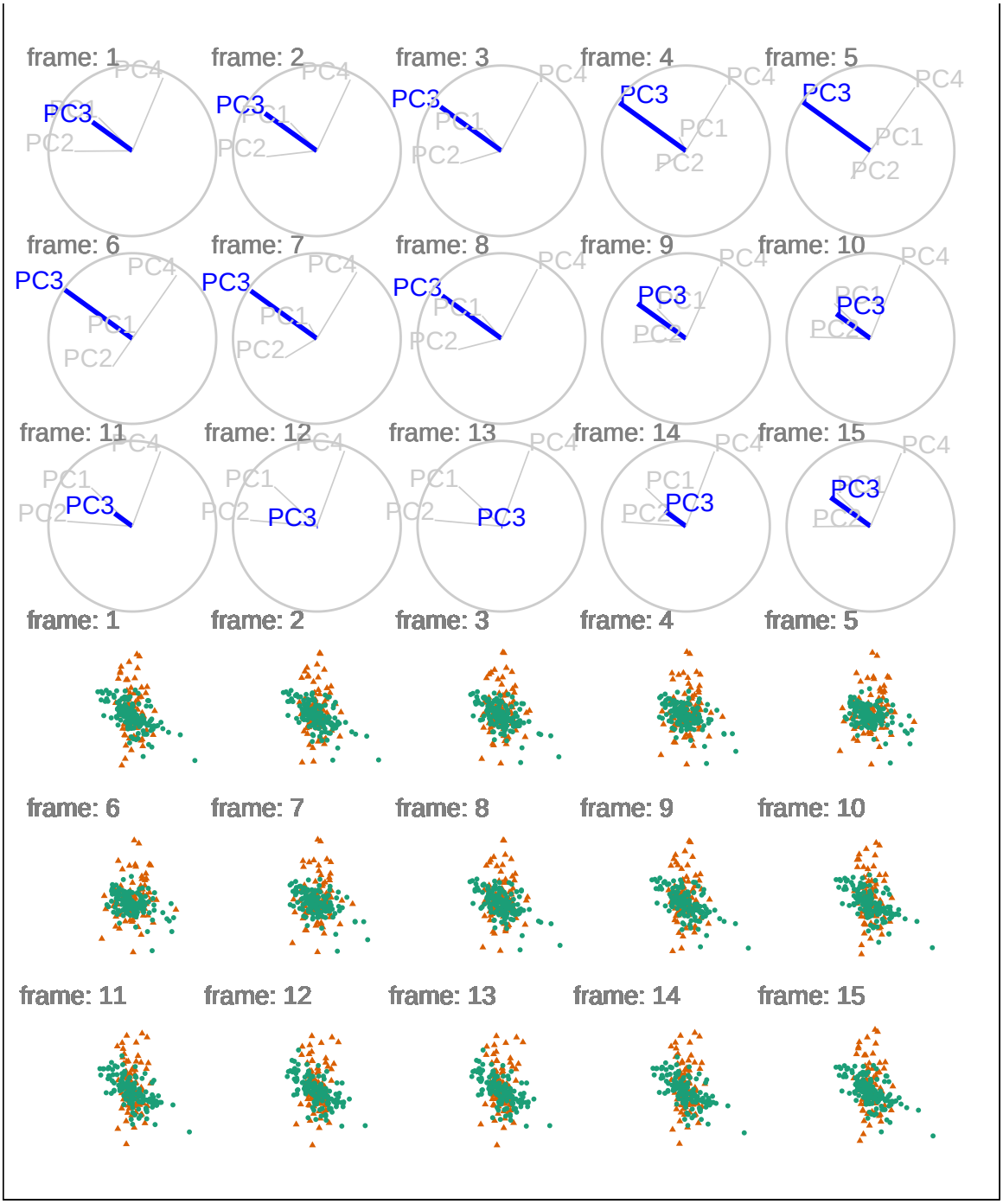


Figure B.1: *Jet cluster, radial manual tour of PC3. Colored by experiment type: 'ATLAS7new' in green and 'ATLAS7old' in orange. When PC3 fully contributes to the projection ATLAS7new (green) occupies unique space and several outliers are identifiable. Zeroing the contribution from PC3 to the projection hides the outliers and indeed all observations with ATLAS7new are contained within ATLAS7old (orange). A dynamic version can be viewed at* `https://nspyrison.netlify.com/thesis/jetcluster_manualtour_pc3/`.

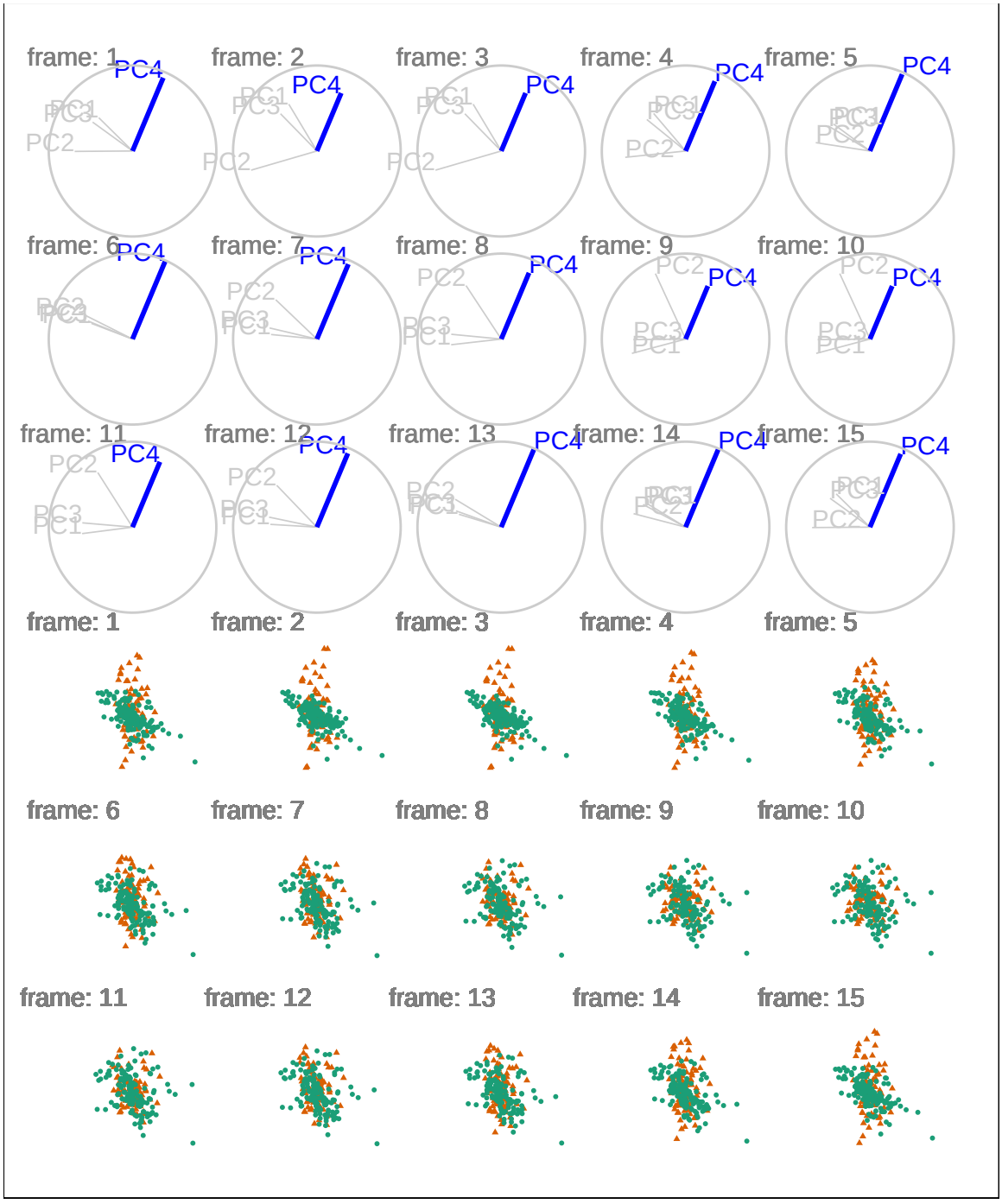


Figure B.2: *Jet cluster, radial manual tour of PC4. Colored by experiment type: 'ATLAS7new' in green and 'ATLAS7old' in orange. This tour contains less interesting information ATLAS7new (green) has points that are right and left of ATLAS7old, while most points occupy the same projection space, regardless of the contribution of PC4. A dynamic version can be viewed at* `https://nspyrison.netlify.com/thesis/jetcluster_manualtour_pc3/`.

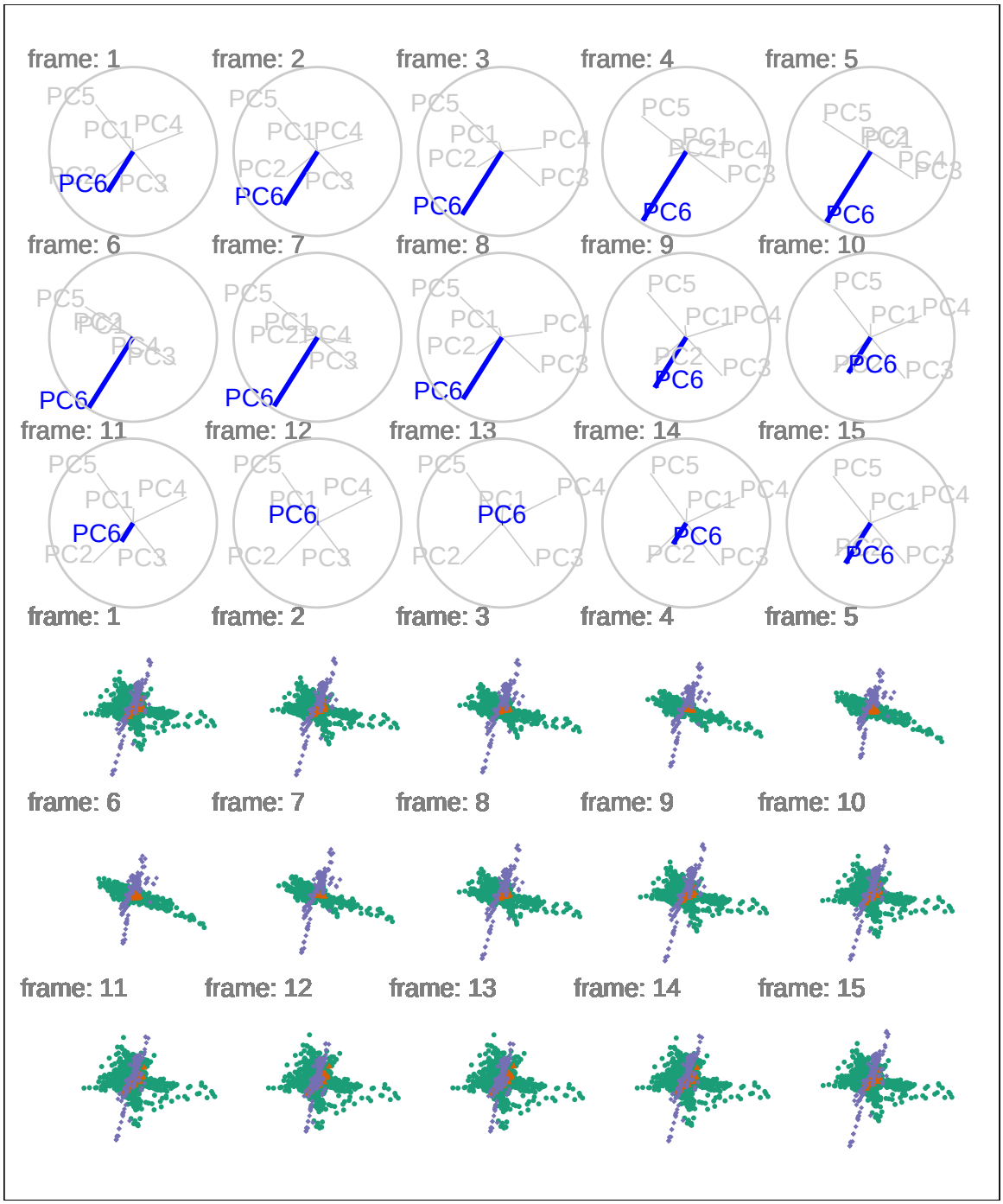


Figure B.3: *DIS cluster, radial manual tour of PC6. colored by experiment type: 'DIS HERA1+2' in green, 'dimuon SIDIS' in purple, and 'charm SIDIS' in orange. When the contribution PC 6 is large we see that dimuon SIDIS (purple) data are nearly orthogonal to DIS HERA (green) data. As the data is rotated, we can also see that DIS HERA (green) practically lie on a plane in this 6-d subspace. When the contribution of PC6 is near zero, dimonSIDIS (purple) occupies the same space as the DIS HERA data. A dynamic version can be viewed at* `https://nspyrison.netlify.com/thesis/discluster_manualtour_pc6/`.

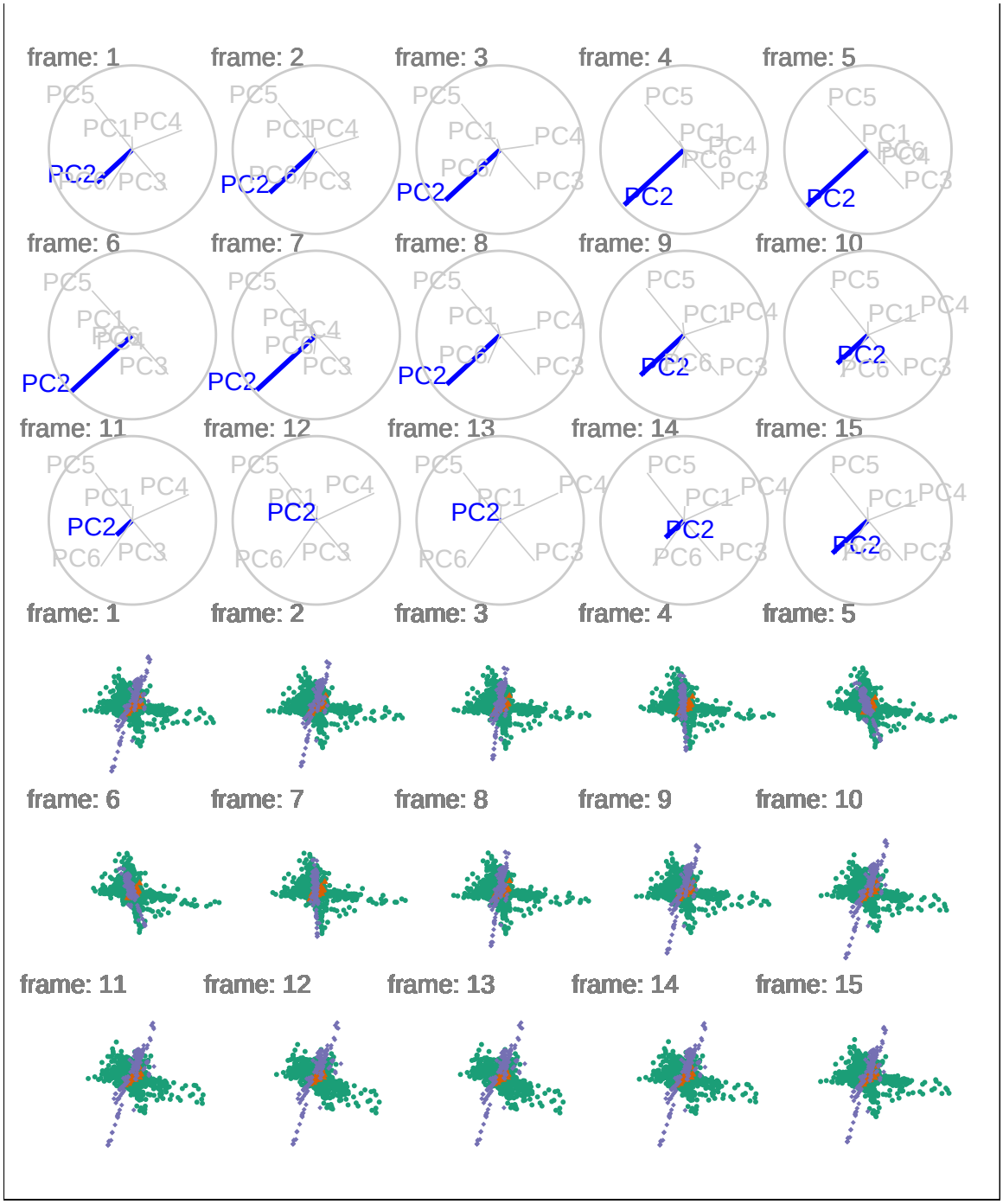


Figure B.4: *DIS cluster, radial manual tour of PC2. Colored by experiment type: 'DIS HERA1+2' in green, 'dimuon SIDIS' in purple, and 'charm SIDIS' in orange. The structure of previously described plane of DIS HERA (green) and nearly orthogonal dimuon SIDIS (purple) is present, however the manipulating PC2 does not give a head-on view of either, a less useful manual tour than that of PC6. A dynamic version can be viewed at* `https://nspyrison.netlify.com/thesis/discluster_manualtour_pc2/`.

DIS cluster manual tours manipulating each of the principal components can be viewed from the links: PC1, PC2, PC3, PC4, PC5, and PC6.

B.5 Source code and usage

This article was created in `R` (R Core Team, 2018), using `bookdown` (Xie, 2016) and `rmarkdown` (Xie, Allaire, and Grolemund, 2018), with code generating the examples inline. The source files can be found at: github.com/nspyrison/confirmation/.

The source code for the `spinifex` package can be found at github.com/nspyrison/spinifex/. To install the package in R, run:

```
# install.package("devtools")
devtools::install_github("nspyrison/spinifex")
```

B.6 Discussion

This work has described an algorithm and package for exploring conducting a manual tour, from a 2D projection, to explore the sensitivity of structure to the contributions of a variable.

Future work on the algorithm and package would include developing it to work with arbitrary projection dimension, enabling the method to operate on other displays like parallel coordinates, and implementing the unconstrained manual control, called oblique in Cook and Buja (1997).

The Givens rotations and Householder reflections as outlined in Buja et al. (2005) may provide a way to conduct higher dimensional manual control. In a Givens rotation, the x and y components ($ie. \theta = 0,\ pi/2$) of the in-plane rotation are calculated separately and would be applied sequentially to produce the radial rotation. Householder reflections define reflection axes to project points on to the axes and generate rotations.

The *tourr* package provides several d-dimensional graphic displays including Andrews curves, Chernoff faces, parallel coordinate plots, scatterplot matrix, and radial glyphs. Having manual controls available for these types of displays would require a dimensionally-generalized rotation matrix.

Development of a graphical user interface, e.g. *shiny* app, would make the *spinifex* package more flexible. The user could easily switch between variables to control, adjust the step size to make smoother rotation sequences, or save any state to continue to continue to explore the contributions of other variables.

Bibliography

Andrews, DF (1972). Plots of High-Dimensional Data. *Biometrics* **28**(1), 125–136. (Visited on 12/19/2018).

Anscombe, FJ (1973). Graphs in Statistical Analysis. *The American Statistician* **27**(1), 17–21. (Visited on 12/19/2018).

Arms, L, D Cook, and C Cruz-Neira (1999). The benefits of statistical visualization in an immersive environment. In: *Virtual Reality, 1999. Proceedings., IEEE*. IEEE, pp.88–95.

Asimov, D (1985). The grand tour: a tool for viewing multidimensional data. *SIAM journal on scientific and statistical computing* **6**(1), 128–143.

Becker, RA and WS Cleveland (1987). Brushing Scatterplots. *Technometrics* **29**(2), 127–142. (Visited on 01/10/2019).

Buja, A, D Cook, D Asimov, and C Hurley (2005). "Computational Methods for High-Dimensional Rotations in Data Visualization". en. In: *Handbook of Statistics*. Vol. 24. Elsevier, pp.391–413. http://linkinghub.elsevier.com/retrieve/pii/S0169716104240147 (visited on 04/15/2018).

Buja, A, C Hurley, and JA McDonald (1987). A data viewer for multivariate data. In: *Colorado State Univ, Computer Science and Statistics. Proceedings of the 18 th Symposium on the Interface p 171-174(SEE N 89-13901 05-60)*.

Carr, DB and WL Nicholson (1988). 'Explor4: A Program for Exploring Four-Dimensional Data Using Stereo-Ray Glyphs, dimensional constraints, rotation, and masking. *Cleveland and McGill (1988)*, 309–329.

Carr, D, E Wegman, and Q Luo (1996). ExplorN: Design considerations past and present. **129**.

Chernoff, H (1973). The Use of Faces to Represent Points in K-Dimensional Space Graphically. *Journal of the American Statistical Association* **68**(342), 361–368. (Visited on 01/05/2019).

Cook, D and A Buja (1997). Manual Controls for High-Dimensional Data Projections. *Journal of Computational and Graphical Statistics* **6**(4), 464–480. (Visited on 04/15/2018).

Cook, D, A Buja, and J Cabrera (1993). Projection Pursuit Indexes Based on Orthonormal Function Expansions. *Journal of Computational and Graphical Statistics* **2**(3), 225–250. (Visited on 01/07/2019).

Cook, D, A Buja, J Cabrera, and C Hurley (1995). Grand Tour and Projection Pursuit. en. *Journal of Computational and Graphical Statistics* **4**(3), 155. (Visited on 05/27/2018).

Cook, D, U Laa, and G Valencia (2018). Dynamical projections for the visualization of PDFSense data. *Eur. Phys. J. C* **78**(9), 742.

Cook, D, DF Swayne, and A Buja (2007). *Interactive and Dynamic Graphics for Data Analysis: With R and GGobi*. en. Google-Books-ID: 34DL7lR_4CoC. Springer Science & Business Media.

Cordeil, M (2019). *Immersive Analytics Toolkit*. original-date: 2017-02-16T05:25:32Z. https://github.com/MaximeCordeil/IATK (visited on 02/04/2019).

Cordeil, M, A Cunningham, T Dwyer, BH Thomas, and K Marriott (2017). ImAxes: Immersive axes as embodied affordances for interactive multivariate data visualisation. In: *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*. ACM, pp.71–83.

Fisherkeller, MA, JH Friedman, and JW Tukey (1974). PRIM-9: An Interactive Multidimensional Data Display and Analysis System.

Friedman, J and J Tukey (1974). A Projection Pursuit Algorithm for Exploratory Data Analysis. en. *IEEE Transactions on Computers* **C-23**(9), 881–890. (Visited on 06/22/2018).

Gracia, A, S González, V Robles, E Menasalvas, and T von Landesberger (2016). New insights into the suitability of the third dimension for visualizing multivariate/multidimensional data: A study based on loss of quality quantification. en. *Information Visualization* **15**(1), 3–30. (Visited on 08/20/2018).

Grimm, K (2017). *mbgraphic: Measure Based Graphic Selection*. https://CRAN.R-project.org/package=mbgraphic (visited on 02/07/2019).

Grinstein, G, M Trutschl, and U Cvek (2002). High-Dimensional Visualizations. en, 14.

Heer, J, M Bostock, and V Ogievetsky (2010). A tour through the visualization zoo. en. *Communications of the ACM* **53**(6), 59. (Visited on 05/31/2018).

Hofmann, H, L Follett, M Majumder, and D Cook (2012). Graphical tests for power comparison of competing designs. *IEEE Transactions on Visualization and Computer Graphics* **18**(12), 2441–2448.

Huber, PJ (1985). Projection Pursuit. en. *The Annals of Statistics* **13**(2), 435–475.

Huh, MY and K Song (2002). DAVIS: A Java-based Data Visualization System. en. *Computational Statistics* **17**(3), 411–423. (Visited on 01/06/2019).

Hurley, C and A Buja (1990). Analyzing High-Dimensional Data with Motion Graphics. *SIAM Journal on Scientific and Statistical Computing* **11**(6), 1193–1211. (Visited on 11/27/2018).

Laa, U and D Cook (2019). Using tours to visually investigate properties of new projection pursuit indexes with application to problems in physics. *arXiv:1902.00181 [physics, stat]*. arXiv: 1902.00181. (Visited on 02/04/2019).

Lee, EK and D Cook (2010). A projection pursuit index for large p small n data. en. *Statistics and Computing* **20**(3), 381–392. (Visited on 02/13/2019).

Lee, EK, D Cook, S Klinke, and T Lumley (2005). Projection Pursuit for Exploratory Supervised Classification. *Journal of Computational and Graphical Statistics* **14**(4), 831–846. (Visited on 01/07/2019).

Lee, JM, J MacLachlan, and WA Wallace (1986). The effects of 3D imagery on managerial data interpretation. *MIS Quarterly*, 257–269.

Lubischew, AA (1962). On the use of discriminant functions in taxonomy. *Biometrics*, 455–477.

Marriott, K, F Schreiber, T Dwyer, K Klein, NH Riche, T Itoh, W Stuerzlinger, and BH Thomas (2018). *Immersive Analytics*. en. Google-Books-ID: vaVyDwAAQBAJ. Springer.

Matejka, J and G Fitzmaurice (2017). Same Stats, Different Graphs: Generating Datasets with Varied Appearance and Identical Statistics through Simulated Annealing. en. In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems - CHI '17*. Denver, Colorado, USA: ACM Press, pp.1290–1294. http://dl.acm.org/citation.cfm?doid=3025453.3025912 (visited on 12/19/2018).

McDonald, JA (1982). Interactive Graphics For Data Analysis.

Munzner, T (2014). *Visualization analysis and design*. AK Peters/CRC Press.

Nelson, L, D Cook, and C Cruz-Neira (1998). XGobi vs the C2: Results of an Experiment Comparing Data Visualization in a 3-D Immer- sive Virtual Reality Environment with a 2-D Workstation Display. en. *Computational Statistics* **14**(1), 39–52.

Ocagne, Md (1885). *Coordonnées parallèles et axiales. Méthode de transformation géométrique et procédé nouveau de calcul graphique déduits de la considération des coordonnées parallèles, par Maurice d'Ocagne, ...* French. OCLC: 458953092. Paris: Gauthier-Villars.

Pedersen, TL and D Robinson (2019). *gganimate: A Grammar of Animated Graphics*. http://github.com/thomasp85/gganimate.

R Core Team (2018). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Scott, DW (1985). Averaged shifted histograms: effective nonparametric density estimators in several dimensions. *The Annals of Statistics*, 1024–1040.

Scott, DW (1995). Incorporating density estimation into other exploratory tools. In: *ASA Proceedings of the Section on Statistical Graphics', American Statistical Association, Alexandria, VA*. Citeseer, pp.28–35.

Sedlmair, M, T Munzner, and M Tory (2013). Empirical guidance on scatterplot and dimension reduction technique choices. *IEEE Transactions on Visualization & Computer Graphics* (12), 2634–2643.

Siegel, JH, EJ Farrell, RM Goldwyn, and HP Friedman (1972). The surgical implications of physiologic patterns in myocardial infarction shock. English. *Surgery* **72**(1), 126–141. (Visited on 01/05/2019).

Sievert, C (2018). *plotly for R*. https://plotly-book.cpsievert.me.

Sutherland, P, A Rossini, T Lumley, N Lewin-Koh, J Dickerson, Z Cox, and D Cook (2000). Orca: A Visualization Toolkit for High-Dimensional Data. *Journal of Computational and Graphical Statistics* **9**(3), 509–529. (Visited on 01/10/2019).

Swayne, DF, D Cook, and A Buja (1991). *Xgobi: Interactive Dynamic Graphics In The X Window System With A Link To S*.

Swayne, DF, DT Lang, A Buja, and D Cook (2003). GGobi: evolving from XGobi into an extensible framework for interactive data visualization. *Computational Statistics & Data Analysis*. Data Visualization **43**(4), 423–444. (Visited on 12/19/2018).

Tierney, L (1990). *LISP-STAT: An Object Oriented Environment for Statistical Computing and Dynamic Graphics*. eng. Wiley Series in Probability and Statistics. New York, NY, USA: Wiley-Interscience.

Tory, M, AE Kirkpatrick, MS Atkins, and T Moller (2006). Visualization task performance with 2D, 3D, and combination displays. *IEEE transactions on visualization and computer graphics* **12**(1), 2–13.

Tukey, JW (1977). *Exploratory data analysis*. Vol. 32. Pearson.

Wagner Filho, J, M Rey, C Freitas, and L Nedel (2018). Immersive Visualization of Abstract Information: An Evaluation on Dimensionally-Reduced Data Scatterplots. In:

Wang, BT, TJ Hobbs, S Doyle, J Gao, TJ Hou, PM Nadolsky, and FI Olness (2018). Visualizing the sensitivity of hadronic experiments to nucleon structure. *arXiv preprint arXiv:1803.02777*.

Ware, C (2000). Designing with a 2$1/2$D attitude. *Information Design Journal* **10**(3), 258–265.

Wegman, EJ (2003). Visual data mining. en. *Statistics in Medicine* **22**(9), 1383–1397. (Visited on 12/19/2018).

Wegman, E, W Poston, and J Solka (2001). *Pixel Tours*. University of Minnesota. https://ima.umn.edu/2001-2002/W11.12-15.01/18492 (visited on 01/10/2019).

Wickens, CD, DH Merwin, and EL Lin (1994). Implications of graphics enhancements for the visualization of scientific data: Dimensional integrality, stereopsis, motion, and mesh. *Human Factors* **36**(1), 44–61.

Wickham, H, D Cook, and H Hofmann (2015). Visualizing statistical models: Removing the blindfold: Visualizing Statistical Models. en. *Statistical Analysis and Data Mining: The ASA Data Science Journal* **8**(4), 203–225. (Visited on 03/16/2018).

Wickham, H, D Cook, H Hofmann, and A Buja (2011). **tourr** : An *R* Package for Exploring Multivariate Data with Projections. en. *Journal of Statistical Software* **40**(2). (Visited on 11/23/2018).

Wickham, H and G Grolemund (2016). *R for Data Science: Import, Tidy, Transform, Visualize, and Model Data*. en. Google-Books-ID: I6y3DQAAQBAJ. "O'Reilly Media, Inc.".

Xie, Y (2016). *bookdown: Authoring Books and Technical Documents with R Markdown*. Boca Raton, Florida: Chapman and Hall/CRC. https://github.com/rstudio/bookdown.

Xie, Y, JJ Allaire, and G Grolemund (2018). *R Markdown: The Definitive Guide*. Boca Raton, Florida: Chapman and Hall/CRC. https://bookdown.org/yihui/rmarkdown.