



MONASH University

**Interactive and dynamic visualization
of high-dimensional data via animated
linear projections, their efficacy, and
their application to local explanations
of nonlinear models**

Nicholas Spyris

B.Sc. Statistics, Iowa State University

A thesis submitted for the degree of Doctor of Philosophy at
Monash University in February 2022
Department of Human-Centred Computing

February 2022

Contents

Copyright notice	v
Abstract	vii
Declaration	1
Publication as part of this thesis	1
Publications and papers during candidature not part of this thesis	1
Acknowledgments	3
Preface	5
1 Introduction	7
1.1 Research questions	10
1.2 Methodology	11
1.3 Contributions	12
1.4 Thesis structure	14
2 Background	15
2.1 Motivation	15
2.2 Multivariate visualization	17
2.3 Linear projections	22
2.4 Evaluating multivariate data visualization	24
2.5 Black-box models	25
2.6 Local explanations	26
2.7 Conclusion	27
3 A User-Controlled Radial Tour for Animated Linear Projections	29
3.1 Algorithm	30
3.2 Oblique cursor movement	37
3.3 Package structure	38
3.4 Use cases	41
3.5 Discussion	45
4 A Study on the Benefit of a User-Controlled Radial Tour for Variable Importance for Structure in High-Dimensional Data	47

CONTENTS

4.1 User study	49
4.2 Results	57
4.3 Conclusion	63
4.4 Accompanying tool: radial tour application	63
4.5 Extended analysis	65
5 Exploring Local Explanations of Nonlinear Models Using the Radial tour	71
5.1 SHAP and tree SHAP local explanations	72
5.2 The cheem viewer	74
5.3 Case studies	82
5.4 Discussion	92
6 Conclusion	95
6.1 Contributions	95
6.2 Limitations and future work	97
7 Appendix	101
.1 Glossary	101
.2 Animated tour links	101
.3 Supplementary material	102
Bibliography	103

Copyright notice

© Nicholas Spyrisson (2022).

I certify that I have made all reasonable efforts to secure copyright permissions for third-party content included in this thesis and have not knowingly added copyright content to my work without the owner's permission.

Abstract

Visualizing data space is a crucial aspect of exploratory data analysis, checking assumptions, and validating model performance. However, visualization quickly becomes complex as the dimensionality of the data or features increases. Traditionally, linear projections have viewed discrete pairs of components to mitigate this complexity. Data visualization *tours* are a class of dynamic linear projections that animate many linear projections over small changes to the projection basis. The permanence of observations between nearby frames potentially conveys more information than discrete orthogonal frames alone.

Tours are categorized by the path of their bases. *Manual tours* uniquely allow for user-controlled steering of a path of bases, where the contributions of individual variables can be changed. The *radial tour* is a specific case of the manual tour, that freezes the angle of movement but allows the magnitude to be varied. This is used in the work reported here. Chapter 3 defines and clarifies the theoretical basis of the radial tour. The details of implementation and illustration of its use are provided. It introduces an open-source R package, that facilitates creating these tours with a variety of display choices and user controls.

Allowing the analyst to steer a path in the radial tour should enable a better understanding of the variable importance to any structure revealed in a projection. Chapter 4 discusses a within-participant user study comparing the radial tour with current practices: principal component analysis (PCA) and the original type of tour that has no steering. The $n = 108$ crowdsourced participants performed a variable attribution task describing the separation between clusters. The results find that the radial tour lead to more accurate variable

attribution. Participants also reported that the radial tour was their preferred visualization method.

Nonlinear modeling techniques are sometimes referred to as black-box models due to the uninterpretable nature of model terms. Recent research in Explainable Artificial Intelligence (XAI) tries to bring these models interpretability through *local explanations*. Local explanations are a class of techniques that approximate the linear-variable importance for prediction at one point in the data. Chapter 5 provides a new approach for exploring the variable sensitivity of local explanations using radial tour. Local explanations can be considered projection bases. The radial tour is used to vary the contribution of a feature to assess the importance that feature for any particular prediction. This is illustrated using four contemporary data examples covering classification and regression. An accompanying R package provides a graphical user interface (GUI) for conducting this analysis.

Declaration

I hereby declare that this thesis contains no material which has been accepted for the award of any other degree or diploma at any university or equivalent institution and that, to the best of my knowledge and belief, this thesis contains no material previously published or written by another person, except where due reference is made in the text of the thesis.

Publication as part of this thesis

The work corresponding to Chapter 3 has been accepted for publication in *The R Journal* (Spyrison and Cook, 2020). While respective studies discussed in Chapters 4 and 5 have not yet been submitted.

I have renumbered and updated sections of the published paper to generate a consistent presentation within the thesis. The code illustrated has been adjusted to highlight the new ggproto API. Introductory and background content has been moved to their respective thesis chapters.

Publications and papers during candidature not part of this thesis

In addition to the research discussed in the thesis, other notable contributions during my candidature include:

- *The state-of-the-art on tours for dynamic visualization of high-dimensional data* (Lee et al., 2021). A WIREs Computational Statistics Review of current tour methods. I contributed writing and visuals discussing manual tours.

CONTENTS

- “*Is IEEE VIS that good?*” *On key factors in the initial assessment of manuscript and venue quality* (Spyrison, Lee, and Besançon, 2021). A survey IEEE VIS authors, how they source articles, decide which to read, and evaluate venue quality. We find low evidence that sentiment changes across academic positions for these topics and provide commentary and discussion on the effects of “publish or perish” environment, standard author and journal metrics, and the need to publish “null” findings and replication studies.
- *Intraday effect of COVID-19 restrictions on Melbourne electricity consumption* (Barrow, Chong, and Spyrison, 2020). We corroborate that the Victorian interday effect on energy consumption did not change and novelly find that the intraday distribution of energy consumption does change. Namely, we find a statistically significant change in the height of the morning and evening peak, energy consumption that we posit is due to less strict schedules associated with working from home, absence of commute time, and other employment changes. We were awarded 1st place of hundreds of entries in the insights category of the Melbourne 2020 Datathon.

Student name: Nicholas Spyrison

Student signature:



Date: 2021-12-12

Acknowledgments

I would like to express my sincere gratitude to my supervisors, Professor Dianne Cook and Professor Kimbal Marriott, for their support of my Ph.D studies and research, their subject expertise, and their careers of supervising and teaching in addition to performing research. Thank you for continuously pushing me and my research to new levels. I have enjoyed teaching data visualization, which has been strongly shaped by Di's practical, data-first, visualization-often approach. I will continue to hear Kim's persistent question, "what do we learn from this?" as a reminder not to get lost in the details of implementation and regularly step back and analyze if this is the correct object to change.

Special thanks to Jieyang Chong and Julie Holden for their proofreading and suggested language clarifications. I thank my fellow Ph.D students, lab members, and Pomodoro partners, primarily on the occasion of stimulating discussions and the positive peer pressure of knowing others are around and working. Thanks immensely to those who empathized with me and others, especially through the hardships of studies and COVID-19. Gratitude to Ying Zhou for her enduring support through the thick of my studies and wavering mental health.

Last but not least, I would like to thank my parents, Doug and Terry, for their support and concerns at odd hours of the day. Thanks to Alan and Claire for their companionship and support now and in our more formative years.

This research was supported in part by an Australian Government Research Training Program (RTP) Scholarship.

Preface

This thesis has been written using R Markdown with the bookdown package (Xie, 2016). All materials required to compile the thesis are available at github.com/nspyrison/thesis_monash_phd. Versions are made available as html and pdf at nspyrison.github.io/thesis_ns/ and github.com/nspyrison/thesis_ns/blob/master/docs/thesis_ns.pdf, respectively.

Chapter 1

Introduction

Exploratory Data Analysis (EDA) is the process of the initial summarization and visualization of a dataset. EDA is a critical first step of checking for realistic values, finding improper data formats, and revealing insights (Tukey, 1977). Wickham and Grolemund (2017) describes the analyst workflow into discrete steps. The data is imported and cleaned before entering into an iterated cycle of transformations, visualization, and modeling. This work aids the visualization step, focusing specifically on quantitative multivariate data.

As modern datasets have grown in complexity, multivariate data has become ubiquitous. Multivariate data is found in physics, biology, social sciences, and manufacturing (Wang et al., 2018; Huber et al., 2015; Brown, 2015; Evans and Boreland, 2017). As the number of variables in the data increases, however, it becomes increasingly difficult to visualize the relationships between the variables and discern the structure of the data set. This thesis addresses the visualization of multivariate data and the analysis of multivariate data using the above-iterated workflow.

One of the most common and successful approaches to visualize multivariate data is to project p -dimensional data onto 1-3 display dimensions. In the same way that 3D object casts a 2D shadow, these projections cast a profile orientation of the data onto a 2D plane.

Our starting point is linear projections, which use a linear combination of variables, called a basis, to define each dimension of the projected space. Each dimension combines multiple

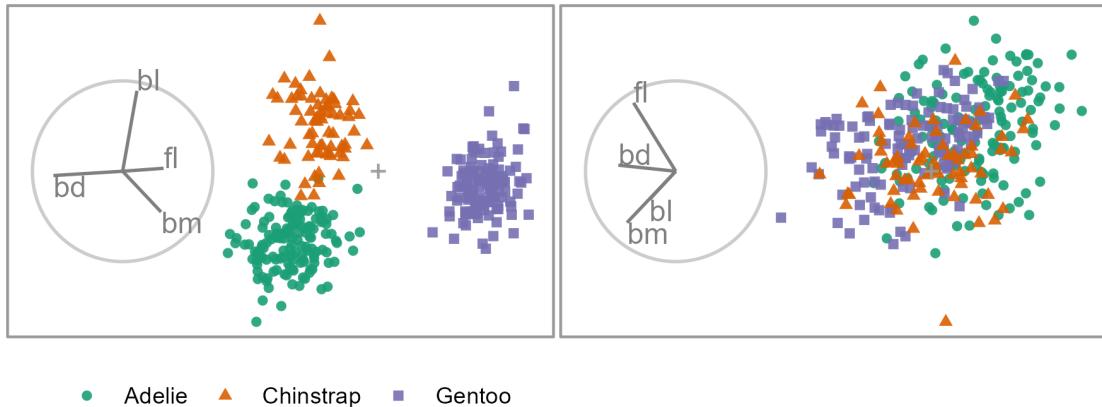


Figure 1.1: Two linear projections of penguins data. Biplot circles depict the basis, the direction and magnitudes of the contributions of the variables. In the left frame, the direction of separation of the orange cluster is mainly in the direction that *bl* contributes, meaning that the variable is sensitive to the separation of this cluster. The purple cluster's separation is attributed primarily to *bd* and *fl* and *bm* to a smaller extent. Many other linear orientations do not resolve structures of interest, such as cluster separation (right frame). The Palmer Penguin data (Gorman, Williams, and Fraser, 2014; Horst, Hill, and Gorman, 2020) measures four physical variables: bill length (*bl*), bill depth (*bd*), flipper length (*fl*), and body mass (*bm*) for three species of penguins.

variables. This is in contrast to looking at each pair of variables in isolation. The basis of linear projections is frequently illustrated with a biplot (Gabriel, 1971). The biplot shows the magnitude and angle each variable contributes to the resulting display dimensions inscribed in a unit circle, such as in Figure 1.1.

There are many features that an analyst may be interested in when analyzing multivariate data. Shape and spread, identifying clusters, outliers, and irregularities are the most common. While linear projections can show these features, not all projections will reveal the feature. Thus the choice of projection is important. Furthermore, the analyst is interested in understanding which combination of variables indicate the feature. For instance, Figure 1.1 shows two linear projections of penguin data. The color and shape of the data points distinguish the three penguin species. The linear projection used in the left frame does not distinguish the three species, while that used in the right frame shows a significant separation.

It, therefore, makes sense for an analyst to explore different linear projections. However, it can be difficult to understand the effect of the projection and to keep track of data points

if the analyst is shown several projections with no relationship to one another. The *tour* is a dynamic visualization that overcomes this difficulty (Cook et al., 2008; Lee et al., 2021). It is a class of linear projections that animate over small changes to the projection basis. A key feature of the tour is the visual permanence and trackability of the points through the frames. In the shadow analogy, an object such as a barstool will cast a circular shadow if the light is directly above the seat. However, such a shadow does not give the observer sufficient information as it could arise from any number of shapes that contain profiles of spheres, cylinders, or circles. However, if the stool was rotated, its legs would show in the shadow quickly giving an intuitive interpretation of the object. Similarly, the rotation of a data object yields information about its structure.

A crucial component for *human-in-the-loop* analysis (Karwowski, 2006) is the ability for the analyst to manually steer the tour. This allows, for example, the analyst to explore what happens to the structure when one variable is removed or the contribution of another is increased. Cook and Buja (1997) introduced the *manual tour*, offering user control over the basis. By selecting a variable and initializing an additional manipulation dimension to an embedding, the contribution of the variable can be controlled.

The manual tour allowed the analyst to control the angle and magnitude that a variable contributes to the projection. However, controlling the magnitude is generally more meaningful as angular manipulations effectively rotate the projection, changing the relative position but not the contribution of a variable. Because of this, this work focuses on a specific manual tour, the *radial tour*, where the angle of the manipulated variable is fixed, but the analyst can vary its magnitude, changing its radius. Figure 1.2 shows a radial tour of the penguins data varying the contribution of bill length. In frames 1 and 2 there is a significant contribution from bill length, and all species are distinguished. In frame 3, however, bill length is removed, and the separation between orange and green clusters has collapsed; we say this variable is sensitive to the separation of these two clusters.

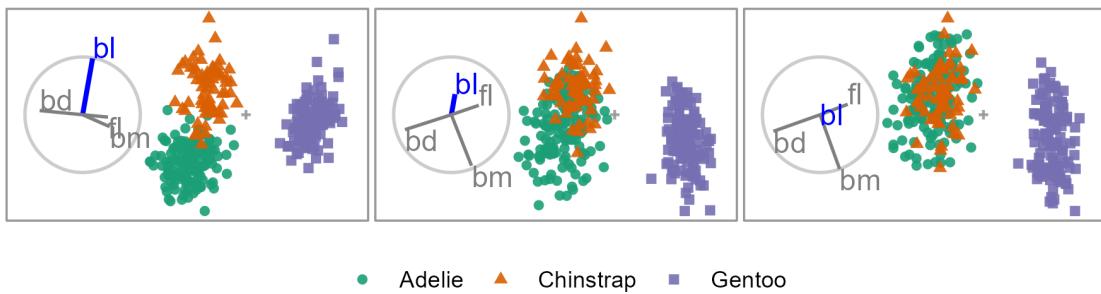


Figure 1.2: A radial tour changing the contribution of bill length (bl). When bill length has a considerable contribution, the clusters of orange and green are separated (left and middle). When its contribution is removed, the clusters overlap (right). Because of this, we say that bill length is sensitive to the separation of these two species. An animated version can be viewed at vimeo.com/676723431.

1.1 Research questions

Discerning variable sensitivity to the structure is crucial in understanding the contribution of a variable to features or differences of interest to the analyst. We conjecture that the user interaction afforded by the radial tour should allow for a more precise exploration of this structure by testing the variable sensitivity to that structure. The over-arching question of interest can, therefore, be stated as:

Can the radial tour, with user control of the basis, help analysts understand the variable sensitivity of structure in the projection?

While Cook and Buja (1997) sketched the theoretical basis for the manual (and hence radial) tour, some details are missing. Furthermore, we lack a publicly available implementation, fully-featured interface design, implementation notes, and have no evaluation of its performance over alternatives.

RQ 1. How do we define and implement a user interface and interactions for the radial tours to add and remove variables smoothly from a 2D linear data projection?

At present, the radial tour is not used by analysts. Instead, they would use a single projection to understand the structure, almost always the principal components analysis (PCA), which chooses the basis that shows the most variation. Another approach is to use a *grand tour*. This randomly selects target bases and then generates an animation

that interpolates between these target bases. Neither PCA nor the grand tour provides a means for manually manipulating a desired variable's contribution to the basis. We wish to investigate if the ability for the analyst to steer the basis in the radial tour facilitates a better understanding of variable sensitivity to the structure.

RQ 2. Does the use of the interactive radial tour improve analysts understanding of the relationship between variables and structure in 2D linear projections compared to existing approaches?

Complex nonlinear models are also being applied more frequently to predict or classify from many predictors. While these models lead to increased accuracy over linear models, they suffer from a loss of the interpretability of their variables. One aspect of eXplainable Artificial Intelligence (XAI, Adadi and Berrada, 2018; Arrieta et al., 2020) tries to preserve the interpretability of such models through local explanations. These explanations are essentially linear variable importance in the vicinity of one observation of a model. That is, the extent that variables help the model explain the difference between the observed means and this observation's prediction. The user control from the radial tour potentially allows an analyst to better understand the model and the support of these local explanations.

RQ 3. Can the radial tours be used in conjunction with local explanations to improve the interpretability of black-box models?

1.2 Methodology

The research corresponding with RQ 1 entails *algorithm & software design* (Kleinberg and Tardos, 2006) adapts the algorithm from Cook and Buja (1997).

To address RQ 2, we use *experimental design* (Winer, 1962). We must define a task and measure suitable to evaluate the radial tour against alternatives. The experimental factors and their levels must be selected and randomly assigned to explore the efficacy of user-controlled radial tours compared with two benchmark methods.

The research responding to RQ 3 involves *design science* (Hevner et al., 2004). It is not obvious how to combine a radial tour with a nonlinear model. A local explanation approximates the linear variable importance in the vicinity of one observation. We must

develop a novel interactive visualization that accommodates two aspects. First, it should visually facilitate the selection of observations to explore. Secondly, extend the biplot to show the distribution of the local explanations and examine the variable sensitivity to the structure identified in the local explanation with the radial tour.

1.3 Contributions

The contributions resulting from the research to address these research questions can be split into scientific knowledge and software contributions:

1.3.1 Scientific knowledge

- Radial tour algorithm
 - Refined and clarified the steps to producing a radial tour based on the use of the Rodrigues' rotation formula extends the approach from Rodrigues (1840).
 - The algorithm makes radial tour in the animation and interactive systems simpler.
 - Provides new examples of usage.
- A user study comparing the radial tour's efficacy against two alternatives—PCA and the grand tour, the first empirical evaluation of the radial tour.
 - Creation of supervised classification task to assess the variable attribution to the separation of two clusters.
 - As tested over experimental factors: location, shape, and dimensionality.
 - Definition of an accuracy measure to evaluate this task.
 - Results: strong evidence that the radial tour increases the accuracy of this task by a sizable amount and minor evidence to suggest a moderate increase in accuracy of the grand tour over Principal Component Analysis (PCA).
 - Mixed model regression helps to attribution the source of the error accounting for the variability of participant's skills and the difficulty from random simulation.
- A novel interactive visualization using the radial tour to explore the variable sensitivity of local explanations from nonlinear models, part of XAI.

- A global view approximates the variable space, attribution space, and model information side-by-side serves to identify a primary observation of interest.
- This observation’s normalized variable attribution is used as a projection basis.
- Explore the support of the local explanation; using the radial tour, the variable sensitivity to the structure identified test the range of contributions supporting the explanation.

1.3.2 Software

- **spinifex**, an **R** package for transforming data, performing radial tours, and the layered composition of any tours.
 - Facilitates the transformation of numeric variables in the data.
 - Identify various bases finding various features of the data.
 - Creation of manual tours allows analyst steering of the basis to explore the variable sensitivity to the structure.
 - Layered composition of tour displays that mirrors the approach in **ggplot2** (Wickham, 2016), interoperable with tours made from **tourr** (Wickham et al., 2011).
 - Interactive **shiny** (Chang et al., 2021) application to preprocess data and explore. Users can choose from six supplied datasets or upload their own.
 - Vignettes and code examples help users get up to speed.
 - Introduces an interactive application to preprocess data and explore. Users can choose from six supplied datasets or upload their own.
- **cheem**, an **R** package that facilitates the exploration of local explanations of nonlinear models:
 - Preprocessing: given a tree-based model, calculate the tree SHAP local explanation (via **treeshap** (Kominsarczyk et al., 2021)) of all observations, and find statistics to accent the separability of this space.
 - Visualization of approximations of the data space, attribution space, and model residual information side-by-side with linked brushing, hover tooltips, and tabular display facilitates the selection of observations to explore.

- Use of the radial tour varies variable contribution to test the support of the variable contribution in agreement with the explanation.
- Interactive application facilitates this analysis for several prepared datasets or user preprocessed data.
- A vignette and code examples help users get up to speed.

1.4 Thesis structure

The remainder of the thesis is organized as follows: Chapter 2 covers various visualizing techniques before introducing related studies and then nonlinear models and their interpretability issues. Chapter 3 discusses the theory and implementation of the manual tour in the package **spinifex**. Chapter 4 discusses a user study evaluating the radial, manual tour’s efficacy compared with PCA and the grand tour. Chapter 5 extends the use of radial tours to improve the interpretability of nonlinear models. Lastly, Chapter 6 concludes with some takeaways and a discussion of possible extensions.

Chapter 2

Background

In the last chapter, we discussed the importance of data visualization despite the complexity of viewing high-dimensional data and outlined the questions we wish to address with the user-control of the radial tour. This chapter first motivates data visualization in general and the importance of interaction. Then we cover orthogonal and observation-liken visuals before turning to dimension reduction. We eventually turn to linear dimension reduction, including further discussion of tours. Then we discuss empirical evaluations of multivariate visuals. Lastly, it concludes with nonlinear models and extends their interpretation with local explanations.

2.1 Motivation

Visualization is much more robust than numerical summarization alone (Anscombe, 1973; Matejka and Fitzmaurice, 2017). Several datasets have the same summary statistics in these studies yet contain obvious visual shapes that could go utterly unheeded if plotting is foregone. Figure 2.1 illustrates this, where data is allowed to drift toward different patterns provided that the mean and standard deviations stay within some tolerance of the original data.

Human-Computer Interaction is the field of study that focuses on the interface between people and computing technology. It encompasses not only the computing resources and hardware input, but also, crucially, the human perception and conception. The term



Figure 2.1: The *datasaurus dozen* is a modern data set illustrating Anscombe’s point that summary statistics cannot always adequately summarize the content of data. The data patterns shown in the scatterplots have the same summary statistics. Unless the data is plotted, one would never know that there were such radical differences.

interaction covered a broad range of uses, purposes, and contexts, making it overburdened and potentially ambiguous (Dimara and Perin, 2019). Interaction facilitates accessing increasing amounts of data and amplifies cognition through control and input. Interaction is known to be an essential aspect of modern data visualization (Batch et al., 2019; Card, Moran, and Newell, 1983; Marriott et al., 2018).

More narrowly, we focus on interactions common to quantitative multivariate data visualization. Linked brushing (Becker and Cleveland, 1987) colors selected observations in one view or frame of an animation allowing selections to be tracked and correlated across other views and frames. Linked brushing has proved to be helpful in animated tours (Arms, Cook, and Cruz-Neira, 1999; Laa, Cook, and Valencia, 2020; Lee, Laa, and Cook, 2020). Tooltip displays upon the cursor hovering over observations can also aid in point identification and displaying associated values.

Any of the above interactions will help a single visual, but can result in an increased effect when combined with coordinated and multiple views (Roberts, 2007) (also known as ensemble graphics Unwin and Valero-Mora (2018)). These views use several types of

visuals to connect observations giving a more comprehensive understanding than anyone visual portrays.

2.2 Multivariate visualization

Before jumping into visualization, we cover the context of the data in mind. Consider the case where data is a complete numeric matrix $X_{n \times p}$ containing n observations of p variables. Ideally, values are continuous numeric (not ordinal levels) and $n > p$ with many more observations than variables. While written as though always operating on the original variables, the visualization discussed could similarly be applied to component spaces or feature decomposition of data not fitting this format.

Grinstein, Trutschl, and Cvek (2002) illustrate many visualization methods in one piece, while Liu et al. (2017) give a good classification of subfield groupings. Below covers a more guided discussion while we generally consider the question, “How can an analyst visualize arbitrary p -dimensions?”. We continue to use the Palmer penguins data in illustrations. This data contains 333 observations of three penguin species across four physical measurements: bill length, bill depth, flipper length, and body mass. Observations were collected between 2007 and 2009 near Palmer Station, Antarctica. This is a good substitute for the over-used iris data.

2.2.1 Scatterplot matrices

Following the thread of coordinated views, an analyst could look at p -univariate histograms or density curves. Extending this idea, exhaustive pairs of variable could be viewed. Using all of these together brings us to scatterplot matrices, also known as SPLOM (Chambers et al., 1983). In a scatterplot matrix, variables are arranged across the columns and rows. The diagonal elements show univariate densities, while off-diagonal positions show scatterplot pairs, as Figure 2.2. This is useful for getting a handle on the range of the variables but is not going to scale well with dimension and is not a suitable audience-ready display.

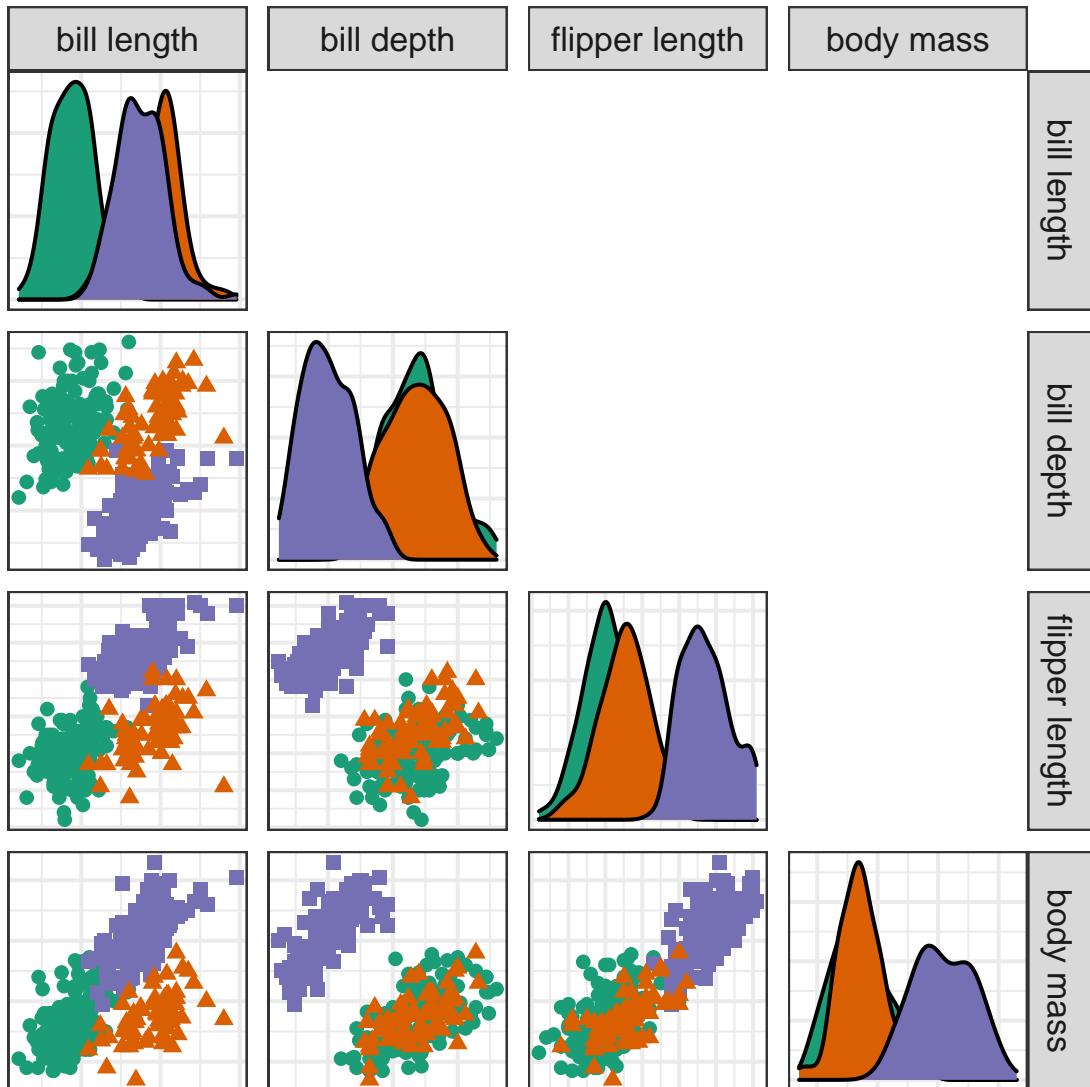


Figure 2.2: Scatterplot matrix views univariate densities and all pairs of bivariate scatterplots. The panels show partial cluster separation indicating these variables contain some cluster discerning information. This approach is good for quickly exploring the range of the data but will not resolve features in more than two dimensions. This is a good exploratory visual but does not scale well with the number of variables.

Alternatively, as n increases, scatterplots displays suffer from dense points that start to occlude each other. This is typically addressed in a few ways. One method would decrease points' opacity, allowing more layers to be seen. Less interestingly, visualization can be performed on a representative subset of the data. Another would change the geometric display to an aggregated heatmap or 2D density contours. These aggregations typically render faster and scale better with increasing observations. Moreover, these methods to alleviate observation density generally work on any scatterplot display (illustrated in Figure 2.5).

2.2.2 Parallel coordinate plots

Alternatively, we could consider a class of observation-based visuals. In parallel coordinate plots (Ocagne, 1885), variables are arranged horizontally, and lines connect observations after being transformed to a common scale quantile or z-value (standard deviations away from the mean). Figure 2.3 illustrates this method. This scales much better with dimensions but poorly with observations. It also suffers from an asymmetry with the variable order. That is, changing the order of the variables will lead people to very different conclusions. The x-axis is also used to display variables rather than the values of the observations. This restricts the amount of information that can be interpreted between variables. Munzner asserts that position is the more human-perceptible channel for encoding information; we should prefer to reserve it for the values of the observations rather than distinguishing variables.

The same issues persist across other displays that separate observations by position, creating n glyph visuals. Examples of these include star plots (Chambers et al., 1983), pixel-based visuals (Keim, 2000), and Chernoff faces (Chernoff, 1973). Parallel coordinate plots and these other visuals scale quite poorly with observations, though they scale well with variables. They maybe candidate visualizations for low n , high p data.

2.2.3 Dimension reduction

Ultimately, we will need to turn to dimension reduction to visualize multivariate spaces. This uses a function to map p -spaces onto a lower d -dimensional space. Dimension reduction

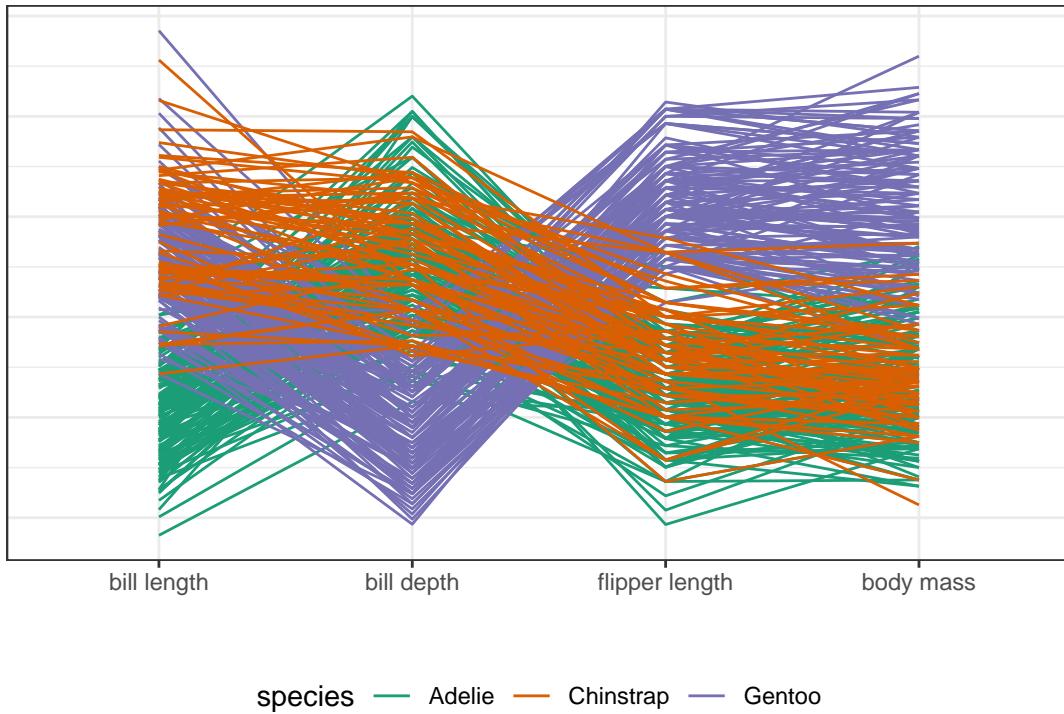


Figure 2.3: Parallel coordinate plots put variables on a common scale and position them side-by-side with lines connecting observations. Some variation between the clusters can be seen. This approach scales relatively well with the number of variables but poorly with the number of observations. It is more difficult to perceive summary statistics (correlation, for example) and suffers from asymmetry with the order of variables. Changing the order of the variables can result in different interpretations. Further, the perception channel of position is used for distinguishing between variables rather than values within the variable.

is separated into two categories, linear and nonlinear. The linear case spans all affine mathematical transformations, essentially mappings where parallel lines stay parallel. Nonlinear transformations complement the linear case, think transformations containing exponents or interacting terms. Examples in low dimensions are relatable. For instance, shadows are an example of linear projections where a 3-dimensional object casts a 2D projection, the shadow. Linear perspective drawings are an example. An example of a nonlinear transformation is that of 2D maps of the globe. The most common maybe the Mercator projection, a rectangular display where the area is proportionally distorted with the distance away from the equator (Snyder, 1987). Other distortions are created when the surface is unwrapped into an elongated ellipse. Yet others create non-continuous gaps on land or oceans to minimize the distortion of targeted areas. wikipedia (2021) lists over

75 different projections to distort the surface to display as a map, each with their unique properties distorting the 3D surface.

Nonlinear techniques often have hyperparameters (absent from linear methods). Hyperparameters are inputs that affect how the spaces are distorted to fit into fewer dimensions. Various computational quality metrics, such as Trustworthiness, Continuity, Normalized stress, and Average local error, describe the distortion of the space (Espadoto et al., 2021; Gracia et al., 2016; Maaten, Postma, and Van den Herik, 2009; Venna and Kaski, 2006). To quote Panagiotelis (2020), “All nonlinear projections are wrong, but some are useful”, a play on George Box’s quote about models (“All *models* are wrong, but some are useful”, Box (1976)). The distortions are hard to interpret and, thus, hard to communicate.

Opinions differ on how to best deal with hyperparameters. Probst, Wright, and Boulesteix (2019) discusses tuning strategies. Others compare implementation defaults (Gijsbers et al., 2021; Pfisterer et al., 2021). Probst, Boulesteix, and Bischl (2019) look at the sensitivity of hyperparameters to the performance of a model. Automated machine learning takes a programmatic approach to hyperparameter tuning (Feurer et al., 2015; Hutter, Kotthoff, and Vanschoren, 2019; Yao et al., 2019). Due to the difficulty of interpreting and communicating nonlinear mappings and the added subjectivity of hyperparameter selection, this thesis continues with linear techniques.

Unfortunately, there is no free lunch here. An increase in the original data dimensions will lead to exponentially larger viewing space in the linear case or an exponentially perturbed and distorted space in nonlinear techniques. Neither is a panacea for visualizing multivariate spaces. However, we will continue with only linear projections due to the opaque distortions of nonlinear methods.

2.2.4 Intrinsic data dimensionality

The intrinsic dimensionality of data is the number of variables needed to minimally represent the data (Grinstein, Trutschl, and Cvek, 2002). Intrinsic data dimensionality is an essential aspect of dimension reduction that does not have to end in visual but is also a standard part of factor analysis and preprocessing data. Consider a Psychology survey consisting of 100 questions about the Big Five personality traits. The data consists of 100 response variables,

while the theory would suggest the intrinsic dimensionality is five. Regardless, reducing the disparity of these volumes would be necessary to gate the exponentially increasing space to view.

2.3 Linear projections

Linear projections map or embed a higher p -dimensional space onto a smaller d -space with an affine mapping (essentially parallel lines stay parallel). A projection is determined by the data multiplied by a basis $Y_{n \times d} = X_{n \times p} \cdot A_{p \times d}$. This basis is an orthonormal matrix that explains the orientation and size of the variable that contribute to the resulting space. This basis is often illustrated as a biplot (Gabriel, 1971), where variable contributions are inscribed in a unit circle showing the direction and the magnitude that they contribution to the resulting space.

A common linear projection is principal component analysis (PCA, Pearson, 1901), which creates a component space ordered by descending variation. It uses eigenvalue decomposition to identify the basis reorientation. These components are typically viewed as discrete orthogonal pairs, commonly approximated in fewer components than the original dimensionality. The exact number of components to keep is subjective but typically guided by the use of a screeplot (Cattell, 1966). Scree plots illustrate the decreasing variation contained in subsequent components. The analyst then identifies an elbow in this plot. This is also commonly used in preprocessing and model initialization when there are many more variables than the intrinsic data dimensionality, such as the Big Five personality example. These reduced spaces can be interpreted as the original variables, albeit with the added abstraction of another linear mapping.

2.3.1 Tours, animated linear projections

In one static linear projection, the resulting projection is defined by one basis. In contrast to one such projection, a data visualization tour animates many such projections through small changes in the basis. In the shadow analogy, structural features of an object is gained by watching its shadow change due to its rotation. An analyst similarly gains information about the data object by watching continuous changes to the basis (the orientation of the

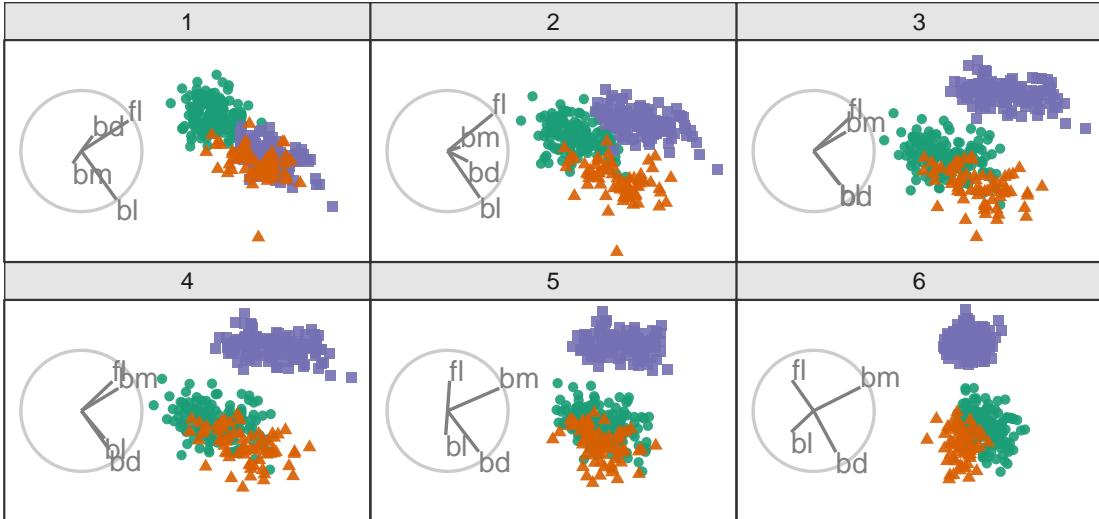


Figure 2.4: Frames from a grand tour. Biplots (grey circles) illustrate the direction and magnitude that variables contribute. In the grand tour, target bases are selected randomly. Tours animate linear projections over small changes in the basis. The observation permanence between frames is an essential distinction in tours. An animation can be viewed at vimeo.com/676723441.

data). There are various types of tours that are classified by the generation of their basis paths. We enumerate a few related to this work. A more comprehensive discussion and review of tours can be found in the works of Cook et al. (2008) and Lee et al. (2021).

Regardless of the type of tour target basis identified, tours must interpolate frames between distant generated target bases. This interpolation is performed on the geodesic path of the bases. Geodesic refers to the shortest path (on a p -sphere of possible bases). This ends up being slightly curved in 2D representation in the same way that flight paths appear curved on 2D representations of the globe. The interpolated frames at small enough angles is essential for the trackability of observations between frames.

Originally in a *grand tour* (Asimov, 1985), several target frames are randomly selected. Figure 2.4 illustrates six frames from a grand tour. The grand tour is good for EDA in that it will show frames with widely varying contributions but lacks a destination, objective function, or means of steering.

The grand tour has no input or control over the bases selected. In contrast, the *manual tour* (Cook and Buja, 1997) allows the analyst to change the contribution of a selected variable. It does so by initializing a manipulation dimension on a 1- or 2D projection which

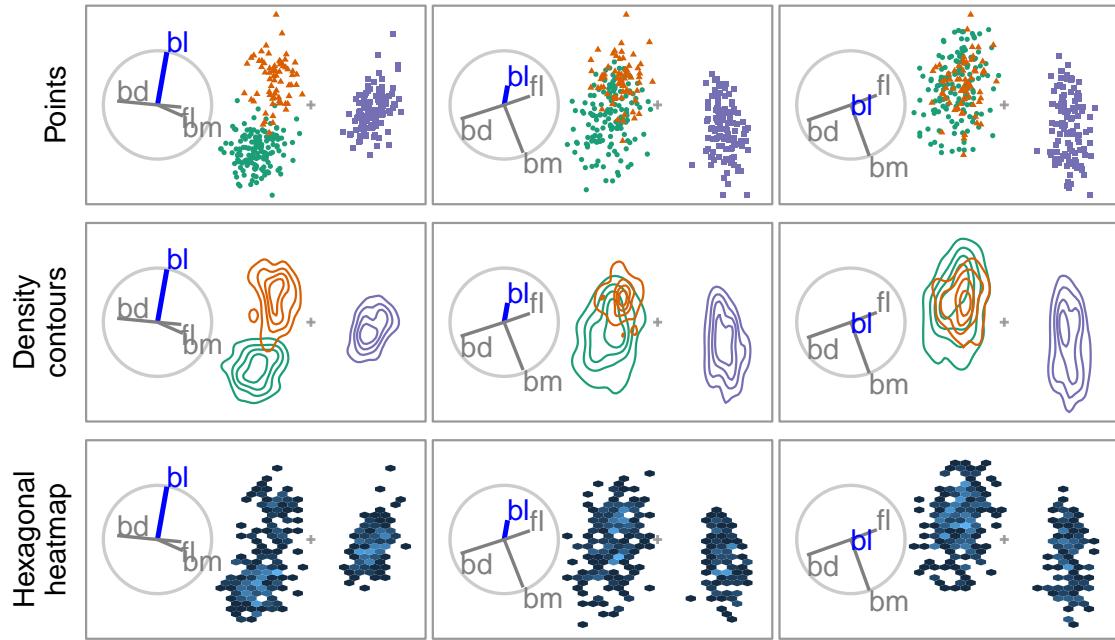


Figure 2.5: *Radial tour across three geometric displays. The contribution of bl is being removed from the frame and with it the separation between the orange and green clusters is also removed. Changing to density contours or hexagonal heatmap display are common ways to mitigate occlusion caused by dense observations. Heatmap display does not perform well with supervised cluster separation, but can be useful to see structure in dense data.*

can then be rotated to alter the contributions of that projection. This work focuses on the *radial tour* sub-variant, where the angle of contribution is fixed and the magnitude of contribution along the radius is controlled. Figure 2.5 illustrates the same radial tour shown in the previous chapter across three geometric displays. The use of density contours and aggregated heatmap displays is useful to mitigate the occlusion of dense observations and is compatible with any bivariate scatterplot display.

2.4 Evaluating multivariate data visualization

Grinstein, Trutschl, and Cvek (2002) provide collected illustrations from many visualization methods. While Liu et al. (2017) provide a good classification of visualization techniques. Definitions and survey of quality metrics is given in Bertini, Tatu, and Keim (2011). The latest, most comprehensive empirical evaluations are given in (Espadoto et al., 2021; Nonato and Aupetit, 2018). While the former studies include discussion of tours, they are absent from empirical evaluations.

Gracia et al. (2016) conducted an $n = 40$ user study comparing between 2D and 3D scatterplots on traditional 2D monitors. Participants perform point classification, distance perception, and outlier identification tasks. The results are mixed and primarily have small differences. There is some evidence to suggest a lower error in distance perception from 3D scatterplot. Wagner Filho et al. (2018) performed an $n = 30$ within participants using scatterplot display between 2D, 3D displays on monitors and 3D display with a head-mounted display on PCA reduced spaces. None of the tasks on any dataset lead to a significant difference in accuracy. However, the immersive display reduced effort and navigation while resulting in higher perceived accuracy and engagement. Sedlmair, Munzner, and Tory (2013) instead uses two expert coders to evaluate 75 datasets and four dimension reduction techniques for 2D scatterplots, interactive 3D scatterplots, and 2D scatterplot matrices. They suggest a tiered guidance approach finding that 2D scatterplots are often ‘good enough’ to resolve a feature. If not, try 2D scatterplots on a different dimension reduction technique before going to scatterplot matrix display or concluding a true negative. They find that interactive 3D scatterplots help in very few cases.

Empirical studies scarcely compare tours. However, Nelson, Cook, and Cruz-Neira (1998) compare scatterplots of grand tours on a 2D monitor with 3D (stereoscopic, not head-mounted) over $n = 15$ participants. Participants perform clusters detection, dimensionality, and radial sparseness tasks on six-dimensional data. They find that stereoscopic 3D leads to more accuracy in the cluster identification, though interaction time much increased in the 3D case. In chapter Chapter 4, we extend the evaluation of tours which novelly compares the radial tour as benchmarked against the grand tour and discrete pairs of principal components.

2.5 Black-box models

There are different reasons and emphases to fit a model. Breiman (2001), reiterated by Shmueli (2010), taxonomize modeling based on its purpose; *exploratory* modeling is done for some inferential purpose, while *predictive* modeling focuses more narrowly on the performance of some objective function. The intended use has important implications for model selection and development. In explanatory modeling, interpretability is vital for

drawing inferential conclusions. While use of black-box models is almost exclusively used in predictive modeling. However, the prevalence of nonlinear models is not without controversy (O’Neil, 2016; Kodiyan, 2019), and the loss of interpretation presents a challenge.

Interpretability is vital for exploring and protecting against potential biases (e.g., sex – Dastin (2018); Duffy (2019), race – Larson et al. (2016), and age – Díaz et al. (2018)) in any model. For instance, models regularly pick up on biases in the training data that have observed influence on the response (output) feature, which is then built into the model. Variable-level (feature-level) interpretability of models is essential in evaluating such biases.

Another concern is data drift, where a shift in the range of the explanatory variables (features or predictors). Nonlinear models are typically more sensitive and do not extrapolate well outside the support of the training data. Maintaining variable interpretability is similarly essential to address issues arising from data drift.

2.6 Local explanations

Explainable Artificial Intelligence (XAI) is an emerging field of research that tries to increase the interpretability of black-box models. A common approach is to use *local explanations*, which attempt to approximate linear variable importance at the location of each observation (instance). That is a prediction at a specific point in the data domain. Because these are point-specific, the challenge is to visualize them to understand a model comprehensively.

Consider a highly nonlinear model. It can be hard to determine whether small changes in a variable’s value will make a class prediction change group or identify which variables contribute to an extreme residual. Local explanations shed light on these situations by approximating linear variable’s importance in the vicinity of a single observation. Figure 2.6 motivates local explanations where the analyst wants to know the variable attribution for a particular observation close to the classification boundary in a nonlinear model.

A comprehensive summary of the taxonomy and literature of explanation techniques is provided in Figure 6 of Arrieta et al. (2020). It includes a large number of model-specific explanations such as deepLIFT (Shrikumar et al., 2016; Shrikumar, Greenside, and Kundaje,

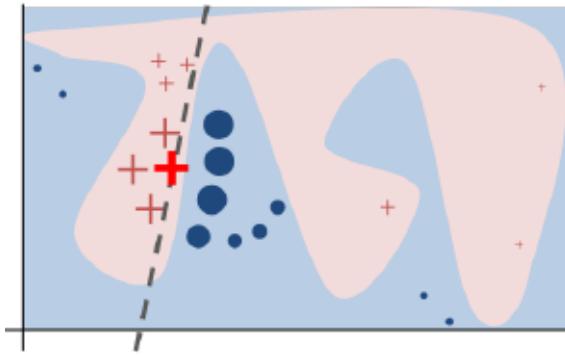


Figure 2.6: Illustration of a nonlinear classification model. An analyst may want to know the variable importance at the vicinity of the highlighted red cross. Understanding this attribution elucidates how the variable influence this point that is precariously close to the classification boundary. Local explanations approximate this linear attribution in the vicinity of one observation. Figure from Ribeiro, Singh, and Guestrin (2016).

2017), a popular recursive method for estimating importance in neural networks. There are fewer model-agnostic explanations, of which LIME, (Ribeiro, Singh, and Guestrin, 2016) SHAP, (Lundberg and Lee, 2017), and their variants are popular.

These observation-level explanations are used in various ways depending on the context. In image classification, a saliency map indicate important pixels for the resulting classification (Simonyan, Vedaldi, and Zisserman, 2014). For example, snow is regularly highlighted when distinguishing if a picture contains a wolf or husky (Besse et al., 2019). In text analysis, word-level contextual sentiment analysis can be used to highlight the sentiment and magnitude of influential words (Vanni et al., 2018). In the case of numeric regression, they are used to explain variable additive contributions from the observed mean to the observation’s prediction (Ribeiro, Singh, and Guestrin, 2016).

2.7 Conclusion

We have discussed the motivation to extend data visualization as dimensionality increases and the importance of user interaction. We discussed several visuals before turning to linear dimension reduction. Empirical evaluations, black-box models, and their local explanations were discussed. There is an absence of studies comparing animated tours with alternative visualization. XAI and extending the interpretability of black-box models are sought-after.

In the following chapters, we address the three stated research questions, respectively. Chapter 3 discusses the implementation of a package that facilitates the creation radial tours and extends the display and exporting of tours in general. Chapter 4 covers the first user-study evaluating the radial tour compared with two common alternatives. Chapter 5, introduces a novel analysis that explores local explanations of nonlinear models with the radial tour.

Chapter 3

A User-Controlled Radial Tour for Animated Linear Projections

This chapter introduces manual tours that allows analysts to influence the contributions to a projection. This feature is unique from previous linear embeddings and not facilitated by compiling software.

Dynamic low-dimensional linear projections of multivariate data known as *tour* provide an essential tool for exploring multivariate data and models. The **R** package **tourr** provides functions for several types of tours: grand, guided, little, local, and frozen. Each of these can be viewed in a development environment, or their basis array can be saved for later consumption. This chapter describes a new package, **spinifex**, which provides a manual tour of multivariate data. In a manual tour an analyst controls the contribution of a variable to the projection. Controlled manipulation is important to explore a variables sensitivity to structure of an identified feature. The use of the manual tour is applied to particle physics data to illustrate the sensitivity of structure in a projection to specific variable contributions. Additionally, we create a ggproto API for composing any tour that mirrors the layered additive approach of **ggplot2**. Tours can then be animated and exported to various formats with **plotly** or **ganimate**.

In the Chapter 2 we introduce linear projections and *tours*, dynamic linear projections animated over small changes to the basis. The *manual tour* (Cook et al., 1995) novelly

allows an analyst control the contribution of a variable to the basis. On the theoretical side of the contribution we fill in previously absent details to solve at the 3D rotation matrix used in 2D manual tours. This proves a scaffolding for the extension for solving for a 4D rotation matrix that could be used for a 3D manual tour. After that we turn our attention to the package and implementation before illustrating use of the manual tour with meta analysis on high energy particle physics data.

The chapter is organized as follows. Section 3.1 describes the algorithm used to perform a radial manual tour implemented in the package **spinifex**. Section 3.3 discussed the functions. Package functionality and code usage following the order applied in the algorithm follows in section 3.3.1. Section 3.4 illustrates how this can be used for sensitivity analysis applied to multivariate data collected on high-energy physics experiments (Wang et al., 2018). Section 3.5 summarizes this chapter.

3.1 Algorithm

The types of manipulations of the manual tour can be thought of in several ways:

- *radial*: fix the direction of contribution, and allow the magnitude to change.
- *angular*: fix the magnitude, and allow the angle or direction of the contribution to vary.
- *horizontal, vertical*: allow rotation only around the horizontal or vertical axis of the current 2D projection.
- *oblique*: paths deviating from these movements such as being captured from the movement of a cursor.

Angular manipulations are homomorphic, in that they show the same information while rotating the frame. More interesting a change in the magnitude of the contribution, changing the radius along the original angle of contribution. For this reason we implement the radial tour as the default values for the manual tour. Below we describe the manual tour illustrated in detail. After that we also include summaries of the algorithms oblique cursor movements for in the 1D and 2D instances.

3.1.1 Notation

The notation used to describe the algorithm for a 2D radial manual tour is as follows:

- \mathbf{X} , the data, an $n \times p$ numeric matrix to be projected.
- \mathbf{A} , any orthonormal projection basis, $p \times d$ matrix, describing the projection from $\mathbb{R}^p \Rightarrow \mathbb{R}^d$.
- k , is the index of the manipulation variable or manip var for short.
- \mathbf{e} , a 1D basis vector of length p , with 1 in the k -th position and 0 elsewhere.
- \mathbf{R} , the $d + 1$ -D rotation matrix, for performing unconstrained 3D rotations within the manip space, \mathbf{M} .
- θ , the angle of in-projection rotation, for example, on the reference axes; c_θ, s_θ are its cosine and sine.
- ϕ , the angle of out-of-projection rotation, into the manip space; c_ϕ, s_ϕ are its cosine and sine. The initial value for animation purposes is ϕ_1 .
- \mathbf{U} , the axis of rotation for out-of-projection rotation orthogonal to \mathbf{e} .
- $\mathbf{Y} = \mathbf{X} \times \mathbf{A}$, the resulting projection of the data through the manip space, \mathbf{M} , and rotation matrix, \mathbf{R} .

The algorithm operates entirely on projection bases and incorporates the data only when making the projected data plots in light of efficiency.

3.1.2 Steps

Step 0) Setup

The flea data (Lubischew (1962)), available in the **tourr** package (Wickham et al., 2011), is used to illustrate the algorithm. The data contains 74 observations of six variables, physical measurements of flea beetles. Each observation belongs to one of three species.

An initial 2D projection basis must be provided. A suggested way to start is to identify an interesting projection using a projection pursuit guided tour. Here the holes index is used to find a 2D projection of the flea data, which shows three separated species groups. Figure 3.1 shows the initial projection of the data. The left panel displays the projection basis (\mathbf{A}) and

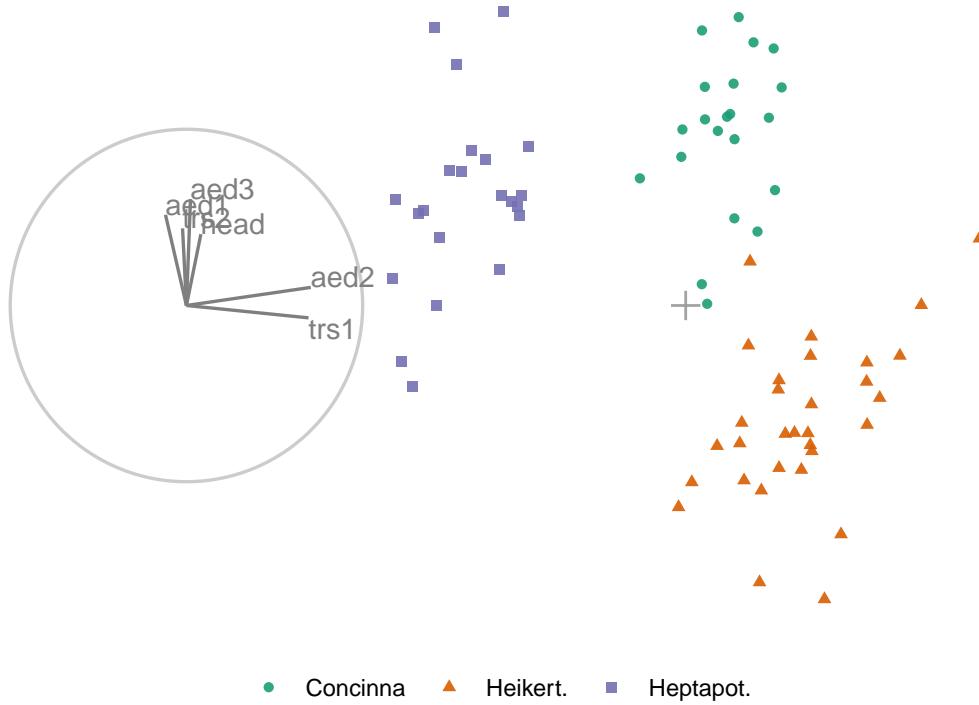


Figure 3.1: Biplot of the initial 2D projection: representation of the basis (left) and resulting data projection (right) of standardized flea data. The color and shape of data points are mapped to the species of flea beetle. The basis was produced by a projection pursuit guided tour with the holes index. The contribution of the variables *aede2* and *tars1* approximately contrasts the other variables. The visible structure in the projection are the three clusters corresponding to the three species.

can be used as a visual guide of the magnitude and direction that each variable contributes to the projection. The right panel shows the projected data, $\mathbf{Y}_{[n, 2]} = \mathbf{X}_{[n, p]} \mathbf{A}_{[p, 2]}$. The color and shape of points are mapped to the flea species.

Step 1) Choose manip variable

In figure 3.1 the contribution of the variables *tars1* and *aede2* mostly contrast the contribution of the other four variables. These two variables combined contribute in the direction of the projection where the purple cluster is separated from the other two clusters. The variable *aede2* is selected as the manip var, the variable to be controlled in the tour. The question that will be explored is: how important is this variable to the separation of the clusters in this projection?

Step 2) Create the 3D manip space

Initialize the coordinate basis vector as a zero vector, \mathbf{e} , of length p , and set the k -th element to 1. In the example data, aede2 is the fifth variable in the data, so $k = 5$, set $e_5 = 1$. Use a Gram-Schmidt process to orthonormalize the coordinate basis vector on the original 2D projection to describe a 3D manip space, \mathbf{M} .

$$e_k \leftarrow 1$$

$$\mathbf{e}_{[p, 1]}^* = \mathbf{e} - \langle \mathbf{e}, \mathbf{A}_1 \rangle \mathbf{A}_1 - \langle \mathbf{e}, \mathbf{A}_2 \rangle \mathbf{A}_2$$

$$\mathbf{M}_{[p, 3]} = (\mathbf{A}_1, \mathbf{A}_2, \mathbf{e}^*)$$

The manip space provides a 3D projection from p -dimensional space, where the coefficient of the manip var can range completely between $[0, 1]$. This 3D space serves as the medium to rotate the projection basis relative to the selected manipulation variable. Figure 3.2 illustrates this 3D manip space with the manip var highlighted. This representation is produced by calling the `view_manip_space()` function. This diagram is purely used to help explain the algorithm.

Step 3) Defining a 3D rotation

The basis vector corresponding to the manip var (red line in Figure 3.2), can be operated like a lever anchored to the origin. This is the process of the manual control, that rotates the manip variable into and out of the 2D projection (Figure 3.3). As the variable contribution is controlled, the manip space turns, and the projection onto the horizontal projection plane correspondingly changes. This is a manual tour. Generating a sequence of values for the rotation angles produces a path for the rotation of the manip space.

For a radial tour, fix θ , the angle describing rotation within the projection plane, and compute a sequence for ϕ , defining movement out of the plane. This will change ϕ from the initial value, ϕ_1 , the angle between \mathbf{e} and its shadow in \mathbf{A} , to a maximum of 0 (manip

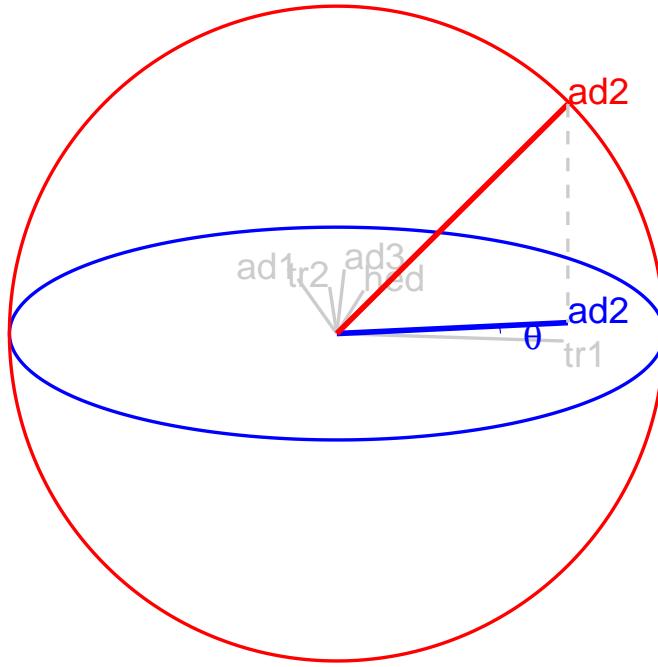


Figure 3.2: Illustration of a 3D manip space, the projection plane is shown as a blue circle extending into and out of the display. A manipulation direction is initialized, the red circle, orthogonal to the projection plane. This allows the selected variable, *aede2*, to change its contribution back to the projection plane. The other variables contributions rotate into this space as well, preserving the orthogonal structure, but are omitted in the manipulation dimension for simplicity.

var fully in projection), then to a minimum of $\pi/2$ (manip var out of projection), before returning to ϕ_1 .

Rotations in 3D can be defined by the axes they pivot on. Rotation within the projection, θ , is rotation around the Z -axis. Out-of-projection rotation, ϕ , is the rotation around an axis on the XY plane, \mathbf{U} , orthogonal to \mathbf{e} . Given these axes, the rotation matrix, \mathbf{R} , can be written as follows, using Rodrigues' rotation formula (originally published in Rodrigues (1840)):

$$\mathbf{R}_{[3, 3]} = \mathbf{I}_3 + s_\phi \mathbf{U} + (1 - c_\phi) \mathbf{U}^2$$

$$\begin{aligned}
 &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 & 0 & c_\theta s_\phi \\ 0 & 0 & s_\theta s_\phi \\ -c_\theta s_\phi & -s_\theta s_\phi & 0 \end{bmatrix} + \begin{bmatrix} -c_\theta(1 - c_\phi) & s_\theta^2(1 - c_\phi) & 0 \\ -c_\theta s_\theta(1 - c_\phi) & -s_\theta^2(1 - c_\phi) & 0 \\ 0 & 0 & c_\phi - 1 \end{bmatrix} \\
 &= \begin{bmatrix} c_\theta^2 c_\phi + s_\theta^2 & -c_\theta s_\theta(1 - c_\phi) & -c_\theta s_\phi \\ -c_\theta s_\theta(1 - c_\phi) & s_\theta^2 c_\phi + c_\theta^2 & -s_\theta s_\phi \\ c_\theta s_\phi & s_\theta s_\phi & c_\phi \end{bmatrix}
 \end{aligned}$$

where

$$\begin{aligned}
 \mathbf{U} &= (u_x, u_y, u_z) = (s_\theta, -c_\theta, 0) \\
 &= \begin{bmatrix} 0 & -u_z & u_y \\ u_z & 0 & -u_x \\ -u_y & u_x & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & -c_\theta \\ 0 & 0 & -s_\theta \\ c_\theta & s_\theta & 0 \end{bmatrix}
 \end{aligned}$$

Step 4) Creating an animation of the radial rotation

The steps outlined above can be used to create any arbitrary rotation in the manip space. To use these for sensitivity analysis, the radial rotation is built into an animation where the manip var is rotated fully into the projection, completely out, and then back to the initial value. This involves allowing ϕ to vary between 0 and $\pi/2$, call the steps ϕ_i .

1. Set initial value of ϕ_1 and θ : $\phi_1 = \cos^{-1} \sqrt{A_{k1}^2 + A_{k2}^2}$, $\theta = \tan^{-1} \frac{A_{k2}}{A_{k1}}$. Where ϕ_1 is the angle between \mathbf{e} and its shadow in \mathbf{A} .
2. Set an angle increment (Δ_ϕ) that sets the step size for the animation, to rotate the manip var into and out of the projection. Uses of angle increment, rather than a number of steps to control the movement is consistent with the tour algorithm as implemented in the **tourr**.

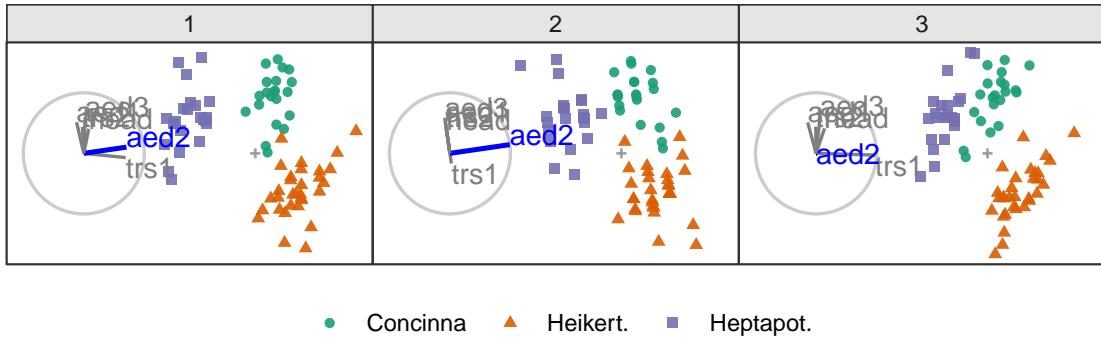


Figure 3.3: Select frames highlight the animation of a radial manual tour manipulating $aede2$: (1) original projection, (2) full contribution, (3) zero contribution, before returning to the original contribution.

3. Step towards 0, where the manip var is entirely in the projection plane.
4. Step towards $\pi/2$, where the manip variable has no contribution to the projection.
5. Step back to ϕ_1 .

In each of the steps 3-5, a small step may be added to ensure that the endpoints of ϕ ($0, \pi/2, \phi_1$) is reached.

Step 5) Projecting the data

The operation of a manual tour is defined on the projection bases. Only when the data plot needs to be made the data projected into the relevant basis.

$$\mathbf{Y}_{[n, 3]}^{(i)} = \mathbf{X}_{[n, p]} \mathbf{M}_{[p, 3]} \mathbf{R}_{[3,3]}^{(i)}$$

where $\mathbf{R}_{[3,3]}^{(i)}$ is the incremental rotation matrix, using ϕ_i . To make the data plot, use the first two columns of \mathbf{Y} . Show the projected data for each frame in sequence to form an animation.

Tours are typically viewed as an animation. The animation of this tour can be viewed online on [GitHub](#). The page may take a moment to load.

3.2 Oblique cursor movement

In a move abbreviated way we can think about the algorithm for 1D and 2D oblique manual tours as:

Algorithm 1 1D oblique manual tour from cursor movement

Require:

- \mathbf{A} is a 1D basis defining the current projection;
- $\mathbf{e} = e_k$ is a p -dimensional vector of zeros with a 1 in the k -th position, the manipulation variable;
- dist_x = horizontal distance of the cursor movement;
- dist_y = vertical distance of the cursor movement;

Ensure: $\|\mathbf{A} - \mathbf{e}\| > \text{tolerance}$

Initialize the manipulation space, Gram-Schmidt process orthonormalizing \mathbf{e} on \mathbf{A} :

$\mathbf{M} \leftarrow \mathbf{e} - \langle \mathbf{e}, \mathbf{A} \rangle \mathbf{A}$

Initialize change in magnitude, ϕ , as a function of the horizontal cursor movement:

$$\phi \leftarrow \frac{\text{dist}_x}{\text{size of plot}_x}, \quad c_\phi = \cos(\phi), \quad s_\phi = \sin(\phi)$$

for ϕ_i in $[0, \phi]$ by an interpolation increment step size **do**

 Rotate the manipulation space:

$$\mathbf{M} \leftarrow \mathbf{M} \cdot \begin{pmatrix} c_{\phi_i} & s_{\phi_i} \\ -s_{\phi_i} & c_{\phi_i} \end{pmatrix}$$

 Append an array, \mathbf{Y} , premultiplying the data by each interpolated basis of the manipulation space:

$$\mathbf{Y}_{npi} \leftarrow \mathbf{X}_{np} \cdot \mathbf{M}_{p1:d}$$

 (Alternatively, store the array of the interpolated bases for more compact format and premultiply data when needed.)

end for

Algorithm 2 2D oblique manual tour from cursor movement

Require:

- \mathbf{A} is a 1D basis defining the current projection;
- $\mathbf{e} = e_k$ is a p -dimensional vector of zeros with a 1 in the k -th position, the manipulation variable;
- dist_x = horizontal distance of the cursor movement;
- dist_y = vertical distance of the cursor movement;

Ensure: $\|\mathbf{A} - \mathbf{e}\| > \text{tolerance}$

Initialize the manipulation space, Gram-Schmidt process orthonormalizing \mathbf{e} on \mathbf{A} :

$$\mathbf{M} \leftarrow \mathbf{e} - \langle \mathbf{e}, \mathbf{A}_1 \rangle \mathbf{A}_1 - \langle \mathbf{e}, \mathbf{A}_2 \rangle \mathbf{A}_2$$

Initialize change in magnitude, ϕ :

$$\phi \leftarrow \frac{\sqrt{\text{dist}_x^2 + \text{dist}_y^2}}{\text{size of plot}_x}, \quad c_\phi = \cos(\phi), \quad s_\phi = \sin(\phi)$$

Initialize change in angle, θ :

$$c_\theta = \cos(\theta) \leftarrow \frac{\text{dist}_x}{\sqrt{\text{dist}_x^2 + \text{dist}_y^2}}$$

$$s_\theta = \sin(\theta) \leftarrow \frac{\text{dist}_y}{\sqrt{\text{dist}_x^2 + \text{dist}_y^2}}$$

for ϕ_i in $[0, \phi]$ by an interpolation increment step size **do**

 Rotate the manipulation space:

$$\mathbf{M} \leftarrow \mathbf{M} \cdot \begin{pmatrix} c_\theta^2 c_\phi + s_\theta^2 & -c_\theta s_\theta(1 - c_\phi) & -c_\theta s_\phi \\ -c_\theta s_\theta(1 - c_\phi) & s_\theta^2 c_\phi + c_\theta^2 & -s_\theta s_\phi \\ c_\theta s_\phi & s_\theta s_\phi & c_\phi \end{pmatrix}$$

 Append an array, \mathbf{Y} , premultiplying the data by each interpolated basis of the manipulation space:

$$\mathbf{Y}_{npi} \leftarrow \mathbf{X}_{np} \cdot \mathbf{M}_{p1:d}$$

 (Alternatively, store the array of the interpolated bases for more compact format and premultiply data when needed.)

end for

3.3 Package structure

In addition to facilitating the manual tour the other primary function is to facilitate the layered composition of tours, interoperably with tours from **tourr**. This package tries to abstract away the complexity of dealing with a varying number of frames and replicating the length of arguments. We use a layered composition approach to tours stemming from **ggplot2** (Wickham, 2016), which can then be animated animation by **plotly** (Sievert, 2020) or **ganimate** (Pedersen and Robinson, 2020). This section describes the functions available in the package, their usage, and how to install and get up and running.

3.3.1 Usage

Using the penguins data , available in the package, to illustrate a manual tour, we will illustrate generating a manual tour to explore the sensitivity of a variable separating two clusters. The composition of the tour display echos the additive layered approach of **ggplot2**, while abstracting away the complexity of dealing with changing number of frames and their animation.

```
## Process penguins data
dat      <- scale_sd(penguins[1:4])
clas     <- penguins$species
bas      <- basis_olda(data = dat, class = clas)

## A manual tour tour path
mt_path <- manual_tour(basis = bas, manip_var = 1, data = dat)

## Composing the display of the tour
ggt <- ggtour(mt_path, angle = .15) +
  proto_point(aes_args      = list(color = clas, shape = clas),
               identity_args = list(alpha = .8, size = 1.5)) +
  proto_basis() +
  proto_origin()

## Animating
animate_plotly(ggt, fps = 5)

## A 1D grand tour from tourr
gt_path <- save_history(data = dat, tour_path = grand_tour(d = 1), max_bases = 10)

## Composing the display of the tour
```

Table 3.1: *Summary of primary functions.*

Family	Function	Related to	Description
processing	scale_01/sd	-	scale each column to [0,1]/std dev away from the mean
processing	basis_pca/olda/...	Rdimtools::do.*	basis of orthogonal component spaces
processing	basis_half_circle	-	basis with uniform contribution across half of a circle
processing	basis_guided	tourr::guided_tour	silently return the basis from a guided tour
tour path	manual_tour	-	basis and interpolation information for a manual tour
tour path	save_history	tourr::save_history	silent, extended wrapper returning other tour arrays
display	ggtour	ggplot2::ggplot	canvas and initialization for a tour animation
display	proto_point/text	geom_point/text	adds observation points/text
display	proto_density/2d	geom_density/2d	adds density curve/2d contours
display	proto_hex	geom_hex	adds hexagonal heatmap of observations
display	proto_basis/1d	-	adds adding basis visual in a unit-circle/-rectangle
display	proto_origin/1d	-	adds reference mark in the center of the data
display	proto_default/1d	-	wrapper for proto_* point + basis + origin
display	facet_wrap_tour	ggplot2::facet_wrap	facets on the levels of variable
display	append_fixed_y	-	add/overwrite a fixed vertical position
animation	animate_plotly	plotly::ggplotly	render as an interactive hmtl widget
animation	animate_gganimate	gganimate::animate	render as a gif, mp4, or other video format
animation	filmstrip	-	static ggplot faceting on the frames of the animation

```

ggt2 <- ggtour(gt_path, angle = .15) +
  proto_default(aes_args      = list(color = clas, fill = clas)) +
  proto_basis1d() +
  proto_origin1d()

## Animating

animate_plotly(ggt2, fps = 5)

```

3.3.2 Functions

Table 3.1 lists the primary functions and their purpose. These are grouped into four types: processing the data, production of tour path, the composition of the tour display, and its animation.

3.3.3 Installation

The **spinifex** is available from CRAN, the following code will help to get up and running:

```

# Installation:

install.package("spinifex") ## Install from CRAN

library("spinifex") ## Load into session

```

```
# Getting started:  
## Shiny app for visualizing basic application  
run_app("intro")  
## View the code vignette  
vignette("getting_started_with_spinifex")  
## More about proto_* functions  
vignette("ggproto_api")
```

3.4 Use cases

Wang et al. (2018) introduce a new tool, PDFSense, to visualize the sensitivity of hadronic experiments to nucleon structure. The parameter-space of these experiments lies in 56 dimensions, and are approximated as the ten first principal components.

Cook, Laa, and Valencia (2018) illustrates how to learn more about the structures using a grand tour. Tours can better resolve the shape of clusters, intra-cluster detail, and better outlier detection than PDFSense & TFEP (TensorFlow embedded projections) or traditional static embeddings. This example builds from here, illustrating how the manual tour can be used to examine the sensitivity of structure in a projection to different parameters. The specific 2D projections passed to the manual tour were provided in their work.

The data has a hierarchical structure with top-level clusters; DIS, VBP, and jet. Each cluster is a particular class of experiments, each with many experimental datasets which each have many observations of their own. In consideration of data density, we conduct manual tours on subsets of the DIS and jet clusters. This explores the sensitivity of the structure to each of the variables in turn and we present the subjectively best and worst variable to manipulate for identifying dimensionality of the clusters and describing the span of the clusters.

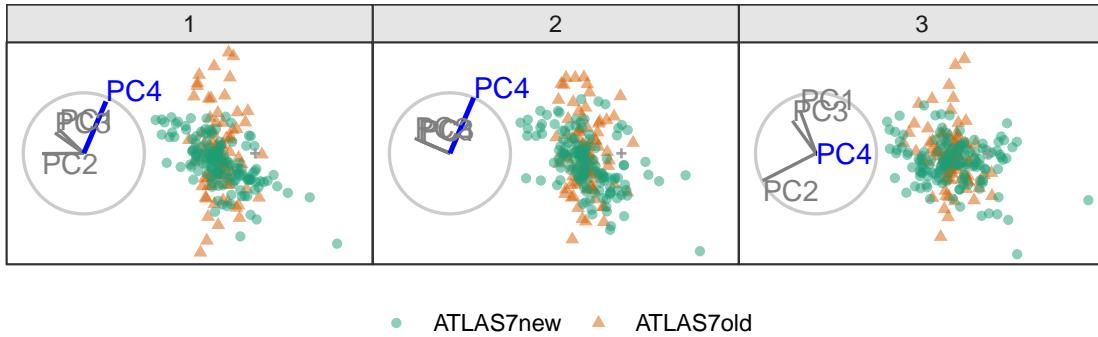


Figure 3.4: Select frames from a radial tour of PC4 within the jet cluster, with color indicating experiment type: ATLAS7new (green) and ATLAS7old (orange). When PC4 is removed from the projection (frame 10), there is little difference between the clusters, suggesting that PC4 is important for distinguishing the experiments.

3.4.1 Jet cluster

The jet cluster resides in a smaller dimensionality than the full set of experiments, with four principal components explaining 95% of the variation in the cluster (Cook, Laa, and Valencia, 2018). The data within this 4D embedding is further subset to ATLAS7old and ATLAS7new, to focus on two groups that occupy different parts of the subspace. Radial manual tours controlling contributions from PC4 and PC3 are shown in Figures 3.4 and 3.5, respectively. The difference in shape can be interpreted as the experiments probing different phase spaces. Back-transforming the principal components to the original variables can be done for a more detailed interpretation.

When PC4 is removed from the projection (Figure 3.4), the difference between the two groups is removed, indicating that PC4 is essential for the separation of experiments. However, eliminating PC3 from the projection (Figure 3.5) does not affect the structure, meaning PC3 is not important for distinguishing experiments. Animations for the remaining PCs can be viewed at the following links: [PC1](#), [PC2](#), [PC3](#), and [PC4](#). It can be seen that only PC4 is important for viewing the difference in these two experiments.

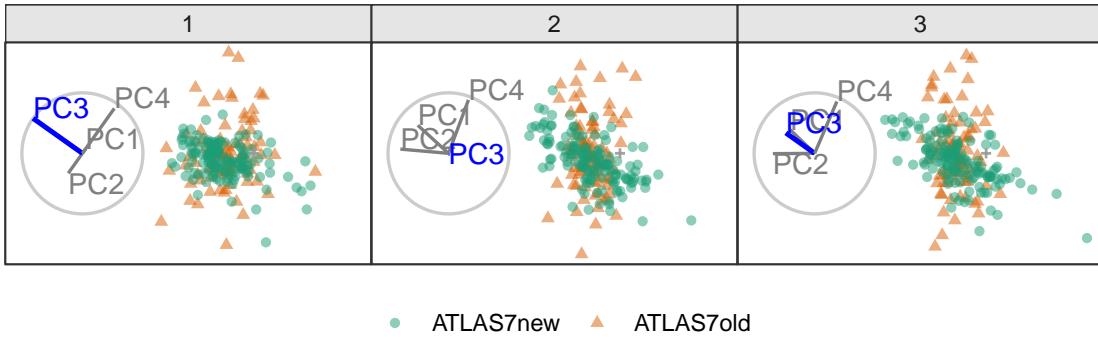


Figure 3.5: Frames from the radial tour manipulating PC3 within the jet cluster, with color indicating experiment type: ATLAS7new (green) and ATLAS7old (orange). When the contribution from PC3 is changed, there is little change in the separation of the clusters, suggesting that PC3 is not important for distinguishing the experiments.

3.4.2 DIS cluster

Following Cook, Laa, and Valencia (2018), to explore the DIS cluster, PCA is recomputed and the first six principal components, explaining 48% of the full sample variation, are used. The contributions of PC6 and PC2 are explored in Figures 3.6 and 3.7, respectively. Three experiments are examined: DIS HERA1+2 (green), dimuon SIDIS (purple), and charm SIDIS (orange).

Both PC2 and PC6 contribute to the projection similarly. When PC6 is rotated into the projection, variation in the DIS HERA1+2 is greatly reduced. When PC2 is removed from the projection, dimuon SIDIS becomes more distinct. Even though both variables contribute similarly to the original projection their contributions have quite different effects on the structure of each cluster, and the distinction between clusters. Animations of all of the principal components can be viewed from the links: [PC1](#), [PC2](#), [PC3](#), [PC4](#), [PC5](#), and [PC6](#).

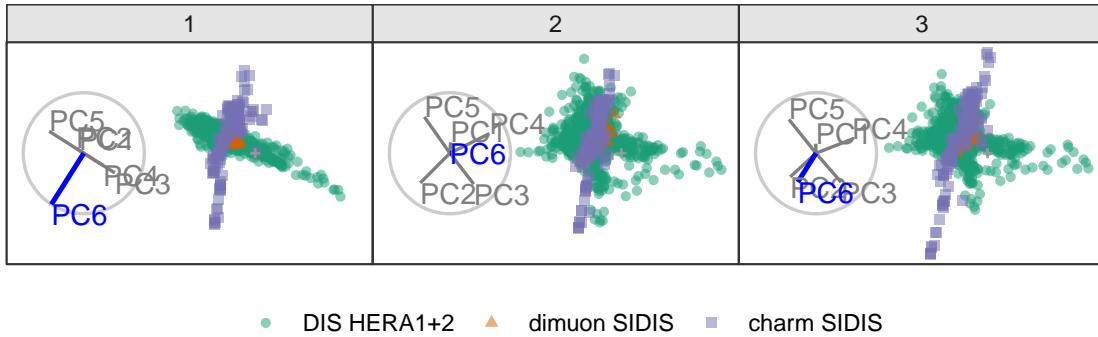


Figure 3.6: Select frames from a radial tour exploring the sensitivity that PC_6 has on the structure of the DIS cluster, with color indicating experiment type: $DIS_{HERA1+2}$ (green), dimuon SIDIS (purple), and charm SIDIS (orange). $DIS_{HERA1+2}$ is distributed in a cross-shaped plane, and charm SIDIS occupies the center of this cross, and dimuon SIDIS is a linear cluster crossing $DIS_{HERA1+2}$. As the contribution of PC_6 is increased, $DIS_{HERA1+2}$ becomes almost singular in one direction (frame 5), indicating that this cluster has very little variability in the direction of PC_6 .

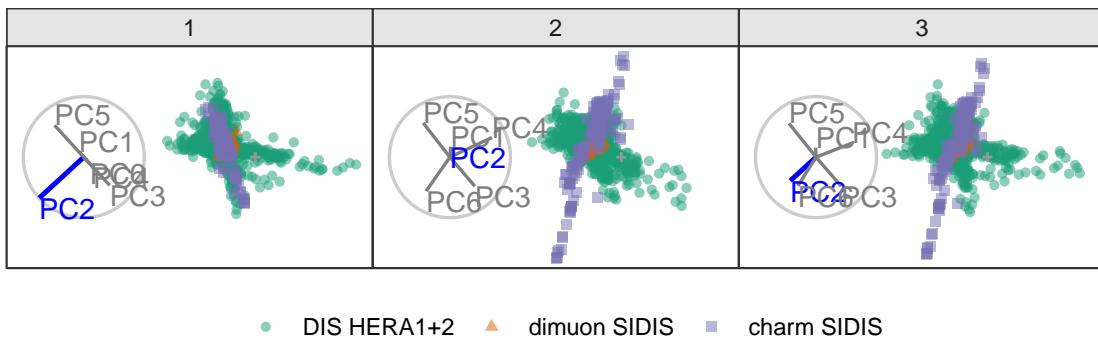


Figure 3.7: Frames from the radial tour exploring the sensitivity PC_2 to the structure of the DIS cluster, with color indicating experiment type: $DIS_{HERA1+2}$ (green), dimuon SIDIS (purple), and charm SIDIS (orange). As the contribution of PC_2 is decreased, dimuon SIDIS becomes more distinguishable from the other two clusters, indicating that in the absence of PC_2 is important for separating this cluster from the others.

3.5 Discussion

Dynamic linear projections of numeric multivariate data, tours, play an important role in data visualization; they extend the dimensionality of visuals to peek into high-dimensional data and parameter spaces. This research has taken the manual tour algorithm, specifically the radial rotation, used in GGobi (Swayne et al., 2003) to interactively rotate a variable into or out of a 2D projection, and modified it to create an animation that performs the same task. It is most useful for examining the importance of variables and how the structure in the projection is sensitive or not to specific variables. This functionality is made available in the package **spinifex**. Which also extends the geometric display and export formats interoperable with the **tourr** package.

This work was motivated by problems in physics, and thus the usage was illustrated on data comparing experiments of hadronic collisions to explore the sensitivity of cluster structure to different principal components. These tools can be applied quite broadly to many multivariate data analysis problems.

The manual tour is constrained in the sense that the effect of one variable is dependent on the contributions of other variables in the manip space. However, this can be useful to simplify a projection by removing variables without affecting the visible structure. Defining a manual rotation in high dimensions is possible using Givens rotations and Householder reflections as outlined in Buja et al. (2005). This would provide more flexible manual rotation but more difficult for a user because they have the choice (too much choice) of which directions to move.

Chapter 4

A Study on the Benefit of a User-Controlled Radial Tour for Variable Importance for Structure in High-Dimensional Data

The previous chapter introduced the package **spinifex** which gave us the means to perform radial tours. Now we have the means to perform radial tours, we want now investigate whether we should be confident that this method with user control will lead to better analysis than traditional alternatives. Therefore, this chapter discusses the user study to elucidate the efficacy of the radial tour.

In Chapters 1 and 2 have introduced PCA, the grand tour, and radial tour. This chapter describes a within-participants user study evaluating efficacy of these techniques. We devise a supervised classification task where participants evaluate variable attribution of the separation between two classes. We define an accuracy measure as the response variable. Data were collected from 108 crowdsourced participants, who performed two trials of each visual for 648 trials in total.

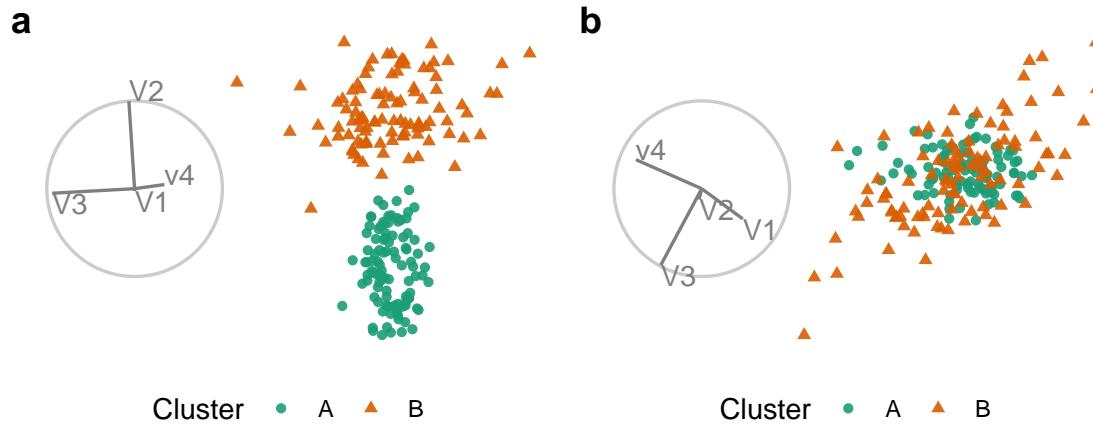


Figure 4.1: Illustration of cluster separation. Panel (a) shows clear separation in V_2 and no separation in the direction of V_3 . While V_1 and v_4 have relatively small contributions to the frame. Panel (b) has a random basis with a minimal contribution from V_2 , and no separation between the cluster means is resolved.

The user influence over a basis, uniquely available in the radial tour, is crucial to testing variable sensitivity to the structure visible in projection. If the contribution of a variable is reduced and the feature disappears, then we say that the variable was sensitive to that structure. For example, Figure 4.1 shows two frames of simulated data. Panel (a) has identified separation between the two clusters. The contributions in panel (b) show no such cluster separation. The former has a large contribution to V_2 in the direction of separation, while it is negligible in the right frame. Because of this we say that V_2 is sensitive to the separation of the clusters.

Knowing which variables to use is also important for statistical modeling and their interpretations. Models are becoming increasingly complex, and the nonlinear interactions of the terms cause an opaqueness to model interpretability. Exploratory Artificial Intelligence (XAI, Adadi and Berrada, 2018; Arrieta et al., 2020) is an emerging field that extends the interpretability of such black-box models. Multivariate data visualization is essential for exploring features spaces and communicating interpretations of models (Biecek, 2018; Biecek and Burzykowski, 2021; Wickham, Cook, and Hofmann, 2015).

The chapter is structured as follows. Section 4.1 describes the experimental factors, task, and accuracy measure used. The results of the study are discussed in Section 4.2.

Conclusions and potential future directions are discussed in Section 4.3. The software used for the study is described in Section 4.4.

4.1 User study

An experiment was constructed to assess the performance of the radial tour relative to the grand tour and PCA for interpreting the variable attribution contributing to separation between two clusters. Data were simulated across three experimental factors: cluster shape, location of the cluster separation, and data dimensionality. Participant responses were collected using a web application and crowdsourced through prolific.co, (Palan and Schitter, 2018) an alternative to MTurk.

4.1.1 Objective

PCA will be used as a baseline for comparison as it is the most common linear embedding. The grand tour will act as a secondary control that will help evaluate the benefit of animation without influencing its path. Lastly, the radial tour should perform best as it benefits from animation and user control.

Then for some subset of tasks, we expect to find that the radial tour performs most accurately. In the appendix, section 4.5, we also regress on the last response time. Due to the absence of inputs, we expect the grand tour to perform faster than the alternatives since users can focus all of their attention on interpreting the fixed path. Conversely, we are less sure about the accuracy of such limited grand tours as there is no objective function in selecting the bases; it is possible that the random selection of the target bases altogether avoids bases showing cluster separation. However, given that the data dimensionality was modest, it seems plausible that the grand tour coincidentally regularly crossed frames with the correct information for the task.

We measure the accuracy and response time over the support of the discussed experimental factors. The null hypothesis can be stated as:

H_0 : task accuracy does not change across visualization method

H_α : task accuracy does change across visualization method

4.1.2 Experimental factors

In addition to visual factor, we simulate the data across three aspects. First, the *location* of the difference between clusters by mixing a signal and a noise variable at different ratios, we vary the number of variables and their magnitude of cluster separation. Secondly, the *shape* of the clusters reflects varying distributions of the data. And third, the *dimension-ality* of the data. Below we describe the levels within each factor, while Figure 4.2 gives a visual representation of the levels.

The *location* of the separation of the clusters is at the heart of the measure. It would be good to test a few varying levels. To test the sensitivity, we mix a noise variable with the signal-containing variable. The difference in the clusters is mixed at the following percentages: 0/100% (not mixed), 33/66%, 50/50% (evenly mixed).

In selecting the *shape* of the clusters, we follow the convention given by Scrucca et al. (2016), where 14 variants of model families containing three clusters are defined. The name of the model family is the abbreviation of its respective volume, shape, and orientation of the clusters, the levels of which are either `_E_equal` or `_V_ary`. We use the models EEE, EEV, and EVV. For Instance, in the EEV model, the volume and shape of clusters are constant, while the shape's orientation varies. The latter model is further modified by moving four-fifths of the data out in a “V” or banana-like shape.

Dimension-ality is tested at two modest levels, namely, in four dimensions containing three clusters and six dimensions with four clusters. Such modest dimensionality is required to bound the difficulty and search space to keep the task realistic for crowdsourcing.

4.1.3 Task and evaluation

With our hypothesis formulated and data at hand, let us turn our attention to the task and how to evaluate it. Regardless of the visual method, the elements of the display are

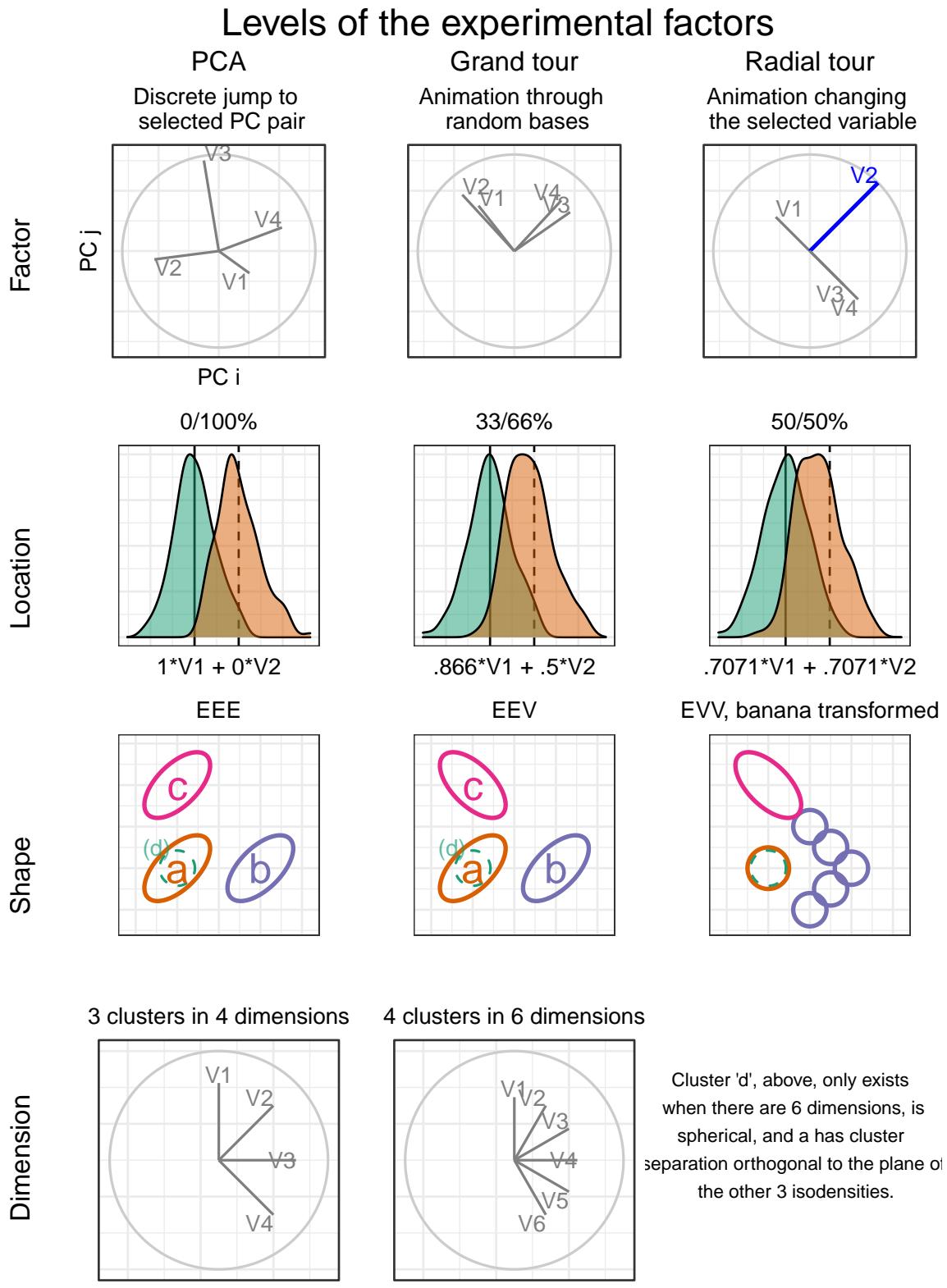


Figure 4.2: Illustration of the experimental factors, the parameter space of the independent variables, the support of our study.

held constant, shown as a 2D scatterplot with an axis biplot to its left. Observations were supervised with the cluster level mapped to color and shape.

Participants were asked to ‘check any/all variables that contribute more than average to the cluster separation green circles and orange triangles,’ which was further explained in the explanatory video as ‘mark any and all variable that carries more than their fair share of the weight, or one quarter in the case of four variables’.

The instructions iterated several times in the video was: 1) use the input controls to find a frame that contains separation between the clusters of green circles and orange triangles, 2) look at the orientation of the variable contributions in the gray circle (biplot axes orientation), and 3) select all variables that contribute more than uniformly distributed cluster separation in the scatterplot. Independent with experimental level, participants were limited to 60 seconds for each evaluation of this task. This restriction did not impact many participants as the 25th, 50th, 75th quantiles of the response time were about 7, 21, and 30 seconds respectively.

The evaluation measure of this task was designed with a few features in mind: 1) the sum of squares of the individual variable weights should be one, 2) symmetric about zero, that is, without preference to under- or over-guessing 3) heavier than linear weight with increasing distance from a uniform height. We define the following measure for evaluating the task with these in mind.

Let a data $\mathbf{X}_{n, p, k}$ be a simulation containing clusters of observations of different distributions. Where n is the number of observations, p is the number of variables, and k indicates the cluster an observation belongs. Cluster membership is exclusive; an observation cannot belong to more than one cluster.

We define weights, w as a vector explaining the variable-wise difference between two clusters. Namely, the difference of each variable between clusters, as a proportion of the total difference, less $1/p$, the expected cluster separation if it were uniformly distributed. Participant responses are a logical value for each variable - whether or not the participant thinks each variable separates the two clusters more than uniformly distributed separation.

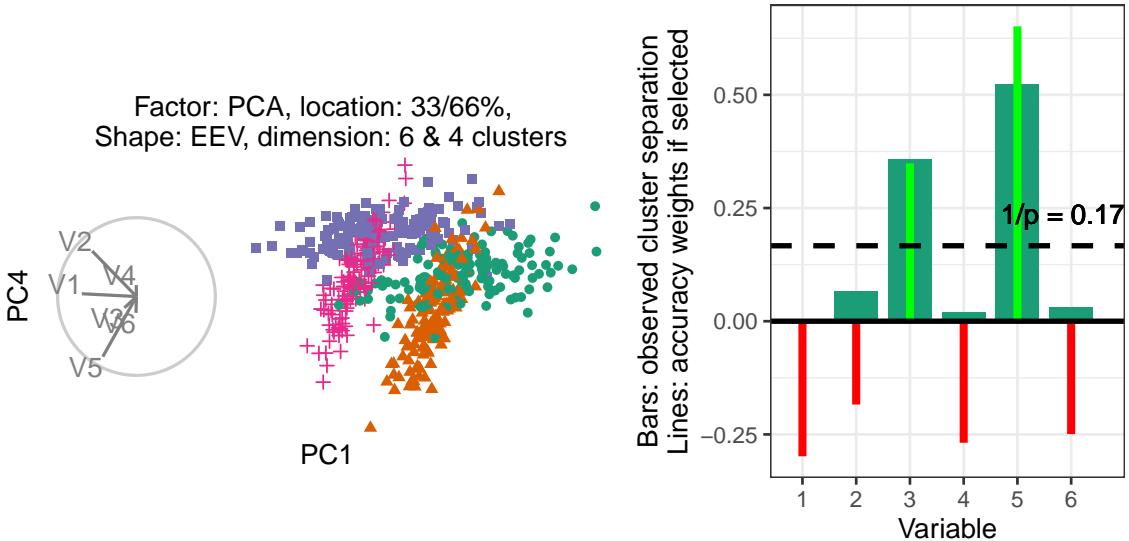


Figure 4.3: (L), PCA biplot of the components showing the most cluster separation with (R) illustration of the magnitude of cluster separation is for each variable (bars) and the weight of the variable accuracy if selected (red/green lines). The horizontal dashed line is $1/p$, the amount of separation each variable would have if evenly distributed. The weights equal the signed square of the difference between each variable value and the dashed line.

$$w_j = \frac{(\bar{X}_{\cdot,j=1,k=1} - \bar{X}_{\cdot,1,2}, \dots, \bar{X}_{\cdot,p,1} - \bar{X}_{\cdot,p,2})}{\sum_{j=1}^p (|\bar{X}_{\cdot,j,k=1} - \bar{X}_{\cdot,j,2}|)} - \frac{1}{p}$$

Where accuracy, A, is defined as:

$$A = \sum_{j=1}^p I(j) \cdot \text{sign}(w_j) \cdot w^2$$

Where $I(j)$ is the indicator function, the binary response for variable j . Figure 4.3 shows one frame of a simulation with its observed variable separation (wide bars), expected uniform separation (dashed line), and accuracy if selected (thin lines).

4.1.4 Visual design standardization

The factors are tested within-participant, with each visual being evaluated twice by each participant. The order that experimental factors are experienced is controlled with the assignment, as illustrated in Figure 4.4. Below we cover the visual design standardization and the input and display within each factor.

The visualization methods were standardized wherever possible. Data were displayed as 2D scatterplots with biplots (Gabriel, 1971), a visual with variable contributions inscribed on a unit circle. All aesthetic values (colors, shapes, sizes, absence of legend, and axis titles) were constant. Variable contributions were always shown left of the scatterplot embeddings with their aesthetic values consistent. What did vary between factors were their inputs.

PCA inputs allowed users to select between the top four principal components for both axes regardless of the data dimensionality (four or six). Data were simulated to have cluster separation within the 2nd to 4th components. Cluster separation was sampled to not bury signal in 5th and 6th components (not selectable in PCA input) in the interest of simplicity and time. There was no user input for the grand tour; users were instead shown a 15-second animation of the same randomly selected path. Participants could view the same clip up to four times within the time limit. Radial tours were also displayed at five frames per second with a step size of 0.1 radians between interpolated frames. Users were able to swap between variables. Selecting a new variable resets the animation where the new variable is manipulated to a full, zero, and then back to its initial contribution. The complete animation of any variable takes about 20 seconds and is almost entirely in the projection frame at around six seconds. The starting basis was initialized to a half-clock design, where the variables were evenly distributed in half of the circle. This design was created to be variable agnostic while maximizing the independence of the variables.

4.1.5 Data simulation

Each dimension is originally distributed as $\mathcal{N}(0, 1)$, given the covariance set by the shape factor. Clusters were originally separated by a distance of two before location mixing. Signal variables had a correlation of 0.9 when they had equal orientation and -0.9 when their orientations varies. Noise variables were restricted to zero correlation. Each cluster is simulated with 140 observations and is offset in a variable that did not distinguish previous variables.

Clusters of the EVV shape are transformed to the banana-chevron shape (illustrated in figure 4.2, shape row). Then location mixing is applied by post-multiplying a (2x2) rotation matrix to the signal variable and a noise variable for the clusters in question. All variables

are then standardized by standard deviation. The rows and columns are then shuffled randomly. The observation's cluster and order of shuffling are attached to the data and saved.

Each of these replications is then iterated with each level of the factor. For PCA, projections were saved for each of the 12 pairs of the top four principal components. We first save two basis paths of differing dimensions for the grand tour before each replication is projected through the common basis path. Each simulation's variable order was previously shuffled, effectually randomizing cluster separation shown. The resulting animations were saved as .gif files. The radial tour starts at either the four or six variable "half-clock" basis, where each variable has a uniform contribution in the right half with no variable contributing in the opposite direction. This acts to minimize the dependence between variable contributions. A radial tour is then produced for each variable and saved as a .gif.

4.1.6 Randomized factor assignment

Now, with simulation and their artifacts in hand, we explain how the experimental factors are assigned and illustrate how this is experienced from a participant's perspective.

We section the study into three periods. Each period is linked to a randomized level of factor visualization and the location. The order of dimension and shape are of secondary interest and are held constant in increasing order of difficulty; four then six dimensions and EEE, EEV, then EVV-banana, respectively.

Each period starts with an untimed training task at the simplest remaining experimental levels; location = 0/100%, shape = EEE, and four dimensions with three clusters. This serves to introduce and familiarize participants with input and visual differences. After the training, the participant performs on two trials with the same factor and location level across the increasing difficulty of dimension and shape. The plot was removed after 60 seconds, though participants rarely reached this limit.

The order of the factor and location levels is randomized with a nested Latin square where all levels of the visual factor are exhausted before advancing to the next level of location. That means we need $3!^2 = 36$ participants to evaluate all permutations of the experimental

Consider a new participant, the 63rd participant,

- 1) Set the factor order:

$1 + (63 - 1) \bmod 6 =$
permutation 4;
grand, PCA, & radial

- 2) Set location order:

$1 + \text{floor}((63 - 1) / 6) \bmod 36 =$
permutation 3; 33/67, 50/50,
& 0/100 % noise/signal mix

Fixed factors:

- 3) Variance-covariance shape
increments with each period:
EEE, EEV, EVV-banana
4) Data dimension is fixed
within each period: 4 then 6

factor order permutation	Period 1			Period 2			Period 3		
	1	2	3	1	2	3	1	2	3
1 P G R	P	G	R	1 1 2 3	1	2	3		
2 P R G	P	R	G	2 1 3 2	2	1	3		
3 G R P	G	R	P	3 2 3 1	3	2	3	1	
4 G P R	G	P	R	4 2 1 3	4	2	1	3	
5 R P G	R	P	G	5 3 1 2	5	3	1	2	
6 R P G	R	P	G	6 3 2 1	6	3	2	1	

Increment through all permutations of factor, before incrementing 1 permutation of location

Period	Evaluation order	Factor ¹	Location ²	Shape ³	Dimensions ⁴
1	Train 1	Grand	0/100	EEE	4 (3cl)
1	1	Grand	33/67	EEE	4 (3cl)
1	2	Grand	33/67	EEE	6 (4cl)
2	Train 2	PCA	0/110	EEE	4 (3cl)
2	3	PCA	50/50	EEV	4 (3cl)
2	4	PCA	50/50	EEV	6 (4cl)
3	Train 3	Radial	0/100	EEE	4 (3cl)
3	5	Radial	0/100	Ban	4 (3cl)
3	6	Radial	0/100	Ban	6 (4cl)

Figure 4.4: Illustration of how a hypothetical participant 63 is assigned experimental factors. Each of the six factor order permutations is exhausted before iterating to the next permutation of location order.

factors once. This randomization controls for potential learning effects the participant may receive. Figure 4.4 illustrates how an arbitrary participant experiences the experimental factors.

Through pilot studies sampled by convenience (information technology and statistics Ph.D. students attending Monash University), we predict that we need three full evaluations to properly power our study; we set out to crowdsource $N = 3 \cdot 3!^2 = 108$ participants.

4.1.7 Participants

We recruited $N = 108$ participants via prolific.co (Palan and Schitter, 2018). We filtered participants based on their claimed education requiring that they have completed at least an undergraduate degree (some 58,700 of the 150,400 users at the time); we apply this filter under the premise that linear projections and biplot displays will not be regularly used for consumption by general audiences. There is also the implicit filter that Prolific participants must be at least 18 years of age and implicit biases of timezone, location, and language. Participants were compensated for their time at £7.50 per hour, whereas the mean duration of the survey was about 16 minutes. We cannot preclude previous

knowledge or experience with the factors but validate this assumption in the follow-up survey asking about familiarity with the methods. The appendix contains a heatmap distribution of age and education paneled across preferred pronouns of the participants that completed the survey, who are relatively young, well educated, and slightly more likely to identify as males.

4.1.8 Data collection

Data were recorded in **shiny** application and written to a Google Sheet after each third of the study. Especially at the start of the study, participants experienced adverse network conditions due to the volume of participants hitting the application with modest allocated resources. In addition to this, API read/write limitations further hindered data collection. To mitigate this, we throttled the number of participants and over-collect survey trials until we received our target three evaluations of all permutation levels.

The processing steps were minimal. First, we format to an analysis-ready form, decoding values to a more human-readable state, and add a flag indicating whether the survey had complete data. We filter to only the latest three complete studies of each experimental factor, which should have experienced the least adverse network conditions. The bulk of the studies removed were partial data and a few over-sampled permutations. This brings us to the 108 studies described in the paper, from which models and aggregation tables were built. The post-study surveys were similarly decoded to human-readable format and then filtered to include only those 84 associated with the final 108 studies.

The code, response files, their analyses, and the study application are publicly available at on GitHub; https://github.com/nspyrison/spinifex_study.

4.2 Results

To recap, the primary response variable is task accuracy, as defined in section 4.1.3. The parallel analysis of the log response time is provided in the appendix. We have two primary data sets; the user study evaluations and the post-study survey. The former is the 108 participants with the explanatory variables: visual factor, location of the cluster

separation signal, the shape of variance-covariance matrix, and the dimensionality of the data. Experimental factors and randomization were discussed in section 4.1.2. The survey was completed by 84 of these 108 people. It collected demographic information (preferred pronoun, age, and education), and subjective measures for each factor (preference, familiarity, ease of use, and confidence).

Below we build a battery of mixed regression models to explore the degree of the evidence and the size of the effects from the experimental factors. Then, we use Likert plots and rank-sum tests to compare the subjective measures between the visual factors.

4.2.1 Accuracy regression

To more thoroughly examine explanatory variables, we regress against accuracy. All models have a random effect term on the participant and the simulation. These terms explain the error attributed to the individual participant's effect and variation due to the random sampling data.

In building a set of models to test, we include all single term models with all independent terms. We also include an interaction term for factor and location, allowing for the slope of each location to change across each level of the factor. For comparison, an overly complex model with all interaction terms is included. The matrices for models with more than two terms is rank deficient; there is not enough varying information in the data to explain all interacting terms.

Fixed effects	Full model
α	$\hat{Y} = \mu + \alpha_i + \mathbf{Z} + \mathbf{W} + \epsilon$
$\alpha + \beta + \gamma + \delta$	$\hat{Y} = \mu + \alpha_i + \beta_j + \gamma_k + \delta_l + \mathbf{Z} + \mathbf{W} + \epsilon$
$\alpha \cdot \beta + \gamma + \delta$	$\hat{Y} = \mu + \alpha_i \cdot \beta_j + \gamma_k + \delta_l + \mathbf{Z} + \mathbf{W} + \epsilon$
$\alpha \cdot \beta \cdot \gamma + \delta$	$\hat{Y} = \mu + \alpha_i \cdot \beta_j \cdot \gamma_k + \delta_l + \mathbf{Z} + \mathbf{W} + \epsilon$
$\alpha \cdot \beta \cdot \gamma \cdot \delta$	$\hat{Y} = \mu + \alpha_i \cdot \beta_j \cdot \gamma_k \cdot \delta_l + \mathbf{Z} + \mathbf{W} + \epsilon$

Table 4.1: Model performance of random effect models regressing accuracy. Each model includes a random effect term of the participant explaining the individual's influence on accuracy. Complex models perform better in terms of R^2 and RMSE, yet AIC and BIC penalize their large number of fixed effects in favor of the much simpler model containing only the visual factor. Conditional R^2 includes the random effects, while marginal does not.

Fixed effects	No. levels	No. terms	AIC	BIC	R2 cond.	R2 marg.	RMSE
a	1	3	-71	-44	0.303	0.018	0.194
a+b+c+d	4	8	-45	4	0.334	0.056	0.194
a*b+c+d	5	12	-26	41	0.338	0.064	0.193
a*b*c+d	8	28	28	167	0.383	0.108	0.19
a*b*c*d	15	54	105	360	0.37	0.222	0.185

where α_i , fixed term for factor $| i \in (\text{pca, grand, radial})$

β_j , fixed term for location $| j \in (0/100\%, 33/66\%, 50/50\%)$ % noise/signal mixing

γ_k , fixed term for shape $| k \in (\text{EEE, EEV, EVV banana})$ model shapes

δ_l , fixed term for dimension $| l \in (4 \text{ variables \& 3 cluster, 6 variables \& 4 clusters})$

μ is the intercept of the model including the mean of random effect

$\mathbf{Z} \sim \mathcal{N}(0, \tau)$, the error of the random effect of participant

$\mathbf{W} \sim \mathcal{N}(0, v)$, the error of the random effect of simulation

$\epsilon \sim \mathcal{N}(0, \sigma)$, the remaining error in the model

Table 4.1 compares the model summaries across increasing complexity. We select the $\alpha \cdot \beta + \gamma + \delta$ model to explore in more detail. Table 4.2 looks at the coefficients for this model.

We also want to visually explore the conditional variables in the model. Figure 4.5 explores violin plots of accuracy by factor while faceting on the location (vertical) and shape (horizontal). Use of the radial visual, on average, increases the accuracy, and especially so when the location of signal mixing is not 33/66%.

4.2.2 Subjective measures

The 84 evaluations of the post-study survey also collect four subjective measures for each factor. Figure 4.6 shows the Likert plots, or stacked percentage bar plots, alongside violin plots with the same non-parametric, ranked sum tests previously used. Participants preferred to use radial for this task. Participants were also more confident of their answers

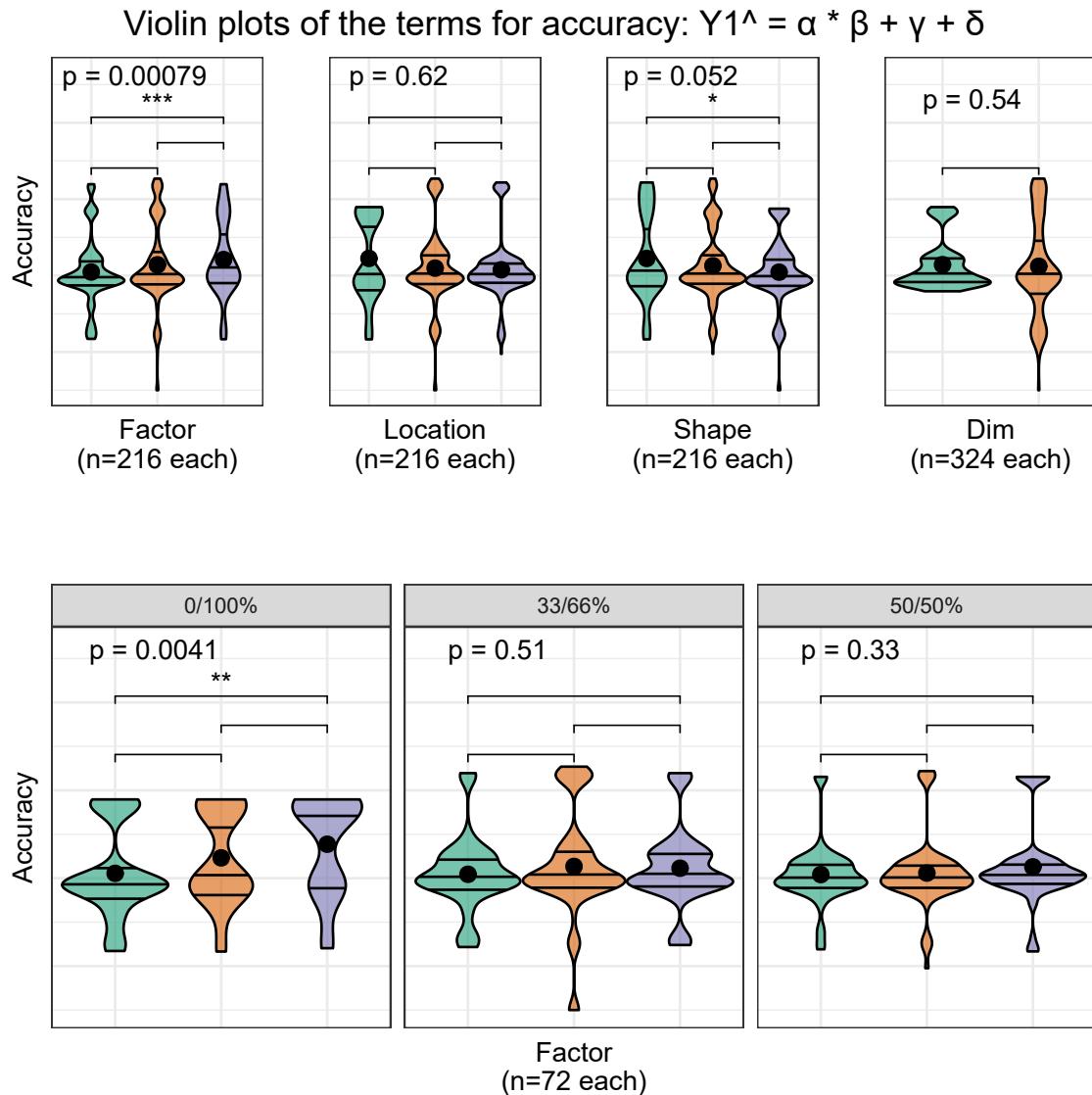


Figure 4.5: Violin plots of terms of the model $\hat{Y} = \alpha \cdot \beta + \gamma + \delta$. Overlaid with global significance from the Kruskal-Wallis test and pairwise significance from the Wilcoxon test, both are non-parametric, ranked-sum tests suitable for handling discrete data. Participants are more confident and find the radial tour easier to use than the grand tour. Participants claim low familiarity, as we expect from crowdsourced participants. Radial is more preferred compared with either alternative for this task.

Table 4.2: The task accuracy model coefficients for $\hat{Y} = \alpha \cdot \beta + \gamma + \delta$, with factor = pca, location = 0/100%, shape = EEE, and dim = 4 held as baselines. Factor being radial is the fixed term with the strongest evidence supporting the hypothesis. Interacting with the location term there is evidence suggesting radial performs worse with 33/66% mixing.

	Estimate	Std. Error	df	t value	Pr(> t)
(Intercept)	0.10	0.06	16.1	1.54	0.143
factor					
Factorgrand	0.06	0.04	622.1	1.63	0.104
Factorradial	0.14	0.04	617.0	3.77	0.000 ***
fixed effects					
Location33/66%	-0.02	0.07	19.9	-0.29	0.777
Location50/50%	-0.04	0.07	20.0	-0.66	0.514
ShapeEEV	-0.05	0.06	11.8	-0.82	0.427
Shapebanana	-0.09	0.06	11.8	-1.54	0.150
Dim6	-0.01	0.05	11.8	-0.23	0.824
interactions					
Factorgrand:Location33/66%	-0.02	0.06	588.9	-0.29	0.774
Factorradial:Location33/66%	-0.12	0.06	586.5	-2.13	0.033 *
Factorgrand:Location50/50%	-0.03	0.06	591.6	-0.47	0.641
Factorradial:Location50/50%	-0.06	0.06	576.3	-1.16	0.248

and found radial tours easier than grand tours. All factors have reportedly low familiarity, as expected from crowdsourced participants.

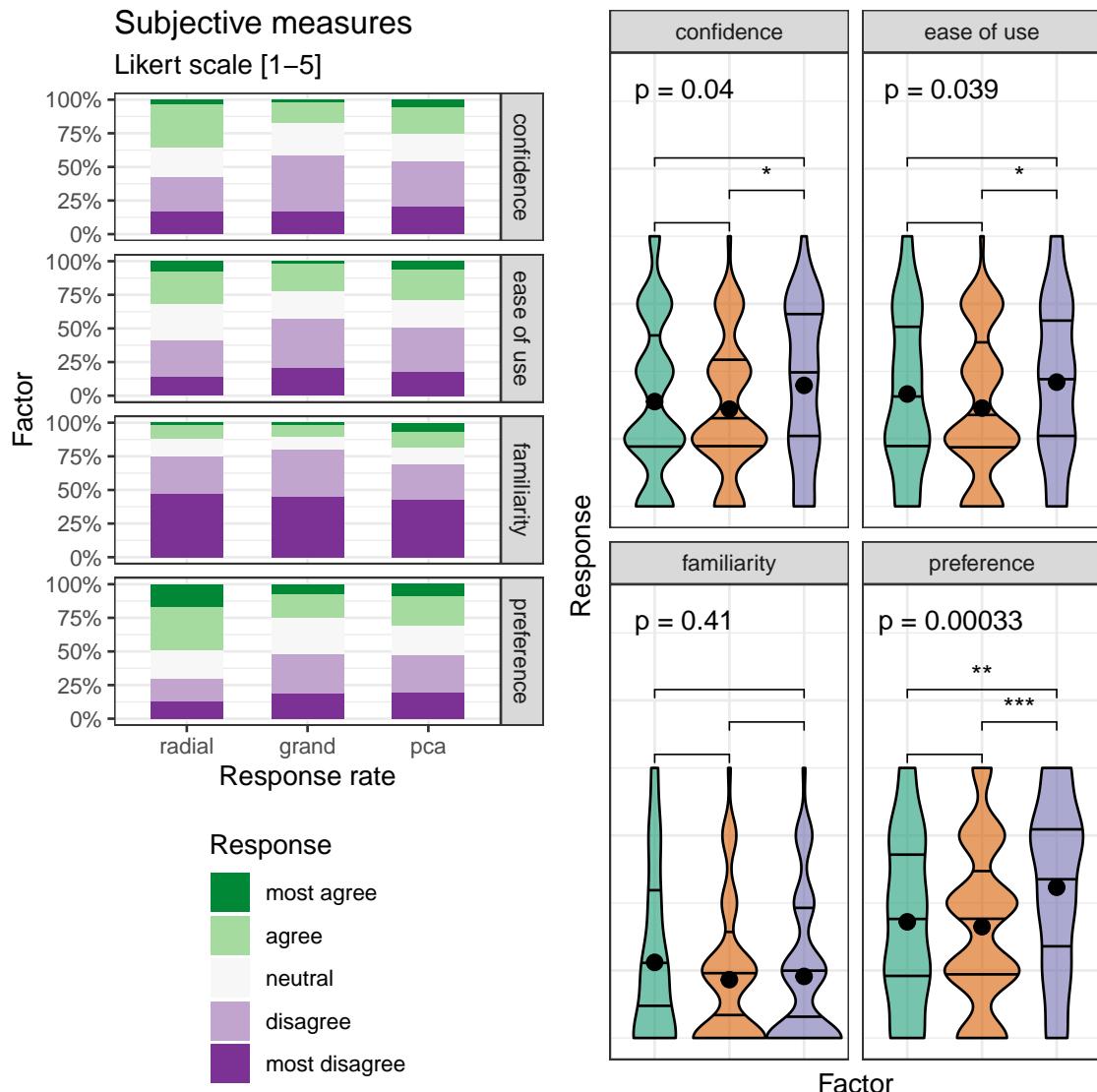


Figure 4.6: The subjective measures of the 84 responses of the post-study survey, five discrete Likert scale levels of agreement (L) Likert plots (stacked percent bar plots) with (R) violin plots of the same measures. Violin plots are overlaid with global significance from the Kruskal-Wallis test and pairwise significance from the Wilcoxon test, both are non-parametric, ranked sum tests.

4.3 Conclusion

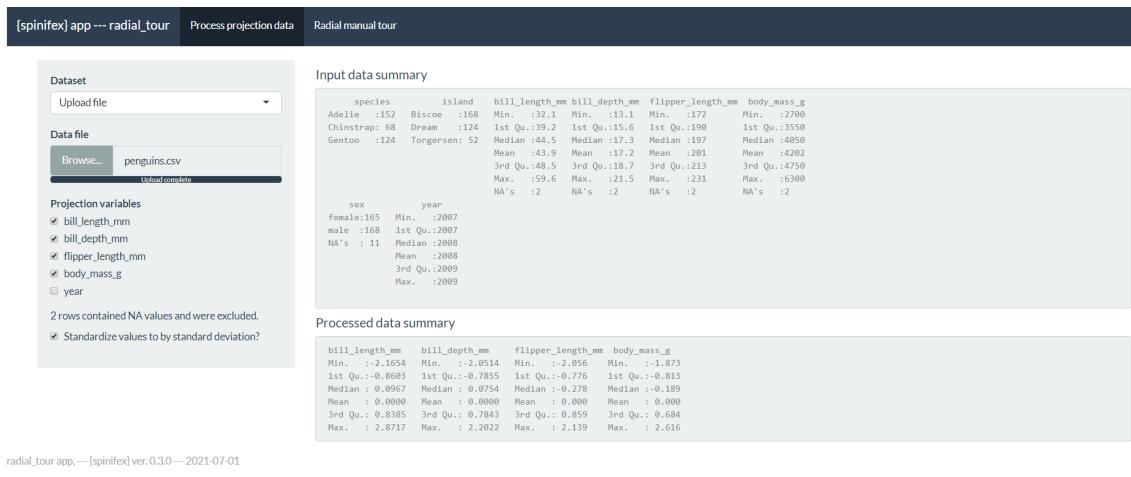
Data visualization is an integral part of EDA. However, thorough exploration of data in high dimensions becomes difficult. Previous methods offer no means for an analyst to impact the projection basis. The manual tour provides a mechanism for changing the contribution of a selected variable to the basis. Giving analysts such control should facilitate the exploration of variable-level sensitivity to the identified structure. We find strong evidence that using the radial tour improves the accuracy relative to PCA or the grand tour on the supervised cluster task assigning variable attribution to the separation of the two clusters.

This paper discussed a with-in participant user study ($n = 108$) comparing the efficacy of three linear projection techniques. The participants performed a supervised cluster task, explicitly identifying which variables contribute to separating two target clusters. This was evaluated evenly over four experimental factors. In summary, we find strong evidence that using the radial tour leads to a sizable accuracy increase. In the appendix, we find evidence for a small change in response time, with PCA being fastest, then the grand tour followed by the radial tour. The effect sizes on accuracy are large relative to the change from the other experimental factors, though smaller than the random effect of the participant. The radial tour was subjectively preferred, leading to more confidence in answers, and increased ease of use than the alternatives.

There are several ways that this study could be extended. In addition to expanding the support of the experimental factors, more exciting directions include: changing the task, visualizations used, and experience level of the target population. It is difficult to achieve good coverage given the number of possible permutations. Keep in mind the traffic volume and low effort of responses from participants when crowdsourcing.

4.4 Accompanying tool: radial tour application

We have produced an application to illustrate the radial tour to accompany this study. The **R** package, **spinifex**, (Spyrison and Cook, 2020) is free, open-source and now contains a **shiny** (Chang et al., 2021) application allowing users to apply various preprocessing



radial_tour app, ---(spinifex) ver. 0.3.0 --- 2021-07-01

Figure 4.7: Process data tab, interactively loads or select data, check which variables to project, and optionally scale columns by standard deviation.

tasks and interactively explore their data via interactive radial tour. Example datasets are provided with the ability to upload data. The .html widget produced is a more interactive variant relative to the one used in the user study. Screen captures and more details are provided in the appendix. Run the following **R** code will run the application locally.

```
## Download:
install.packages("spinifex", dependencies = TRUE)
## Run interactive app demonstrating the radial tour:
spinifex::run_app()
```

In the initial tab, Figure 4.7, users upload their own (.csv, .rds, or .rda) data or select from predefined data sets. The numeric columns appear as a list of variables to include in the projection. Below that, a line displays whether or not missing rows were removed. Scaling by standard deviation is included by default, as this is a common transformation used to explore linear projections of spaces. Summaries of the raw data and processed numeric data are displayed to illustrate how the data was read and its transformation.

The second tab, Figure 4.8 contains interaction for selecting the manipulation variable, and non-numeric columns can be used to change the color and shape of the data points in the projection. The radial tour is created in real-time, animated as an interactive **plotly**.html widget. The application offers users a fast, intuitive introduction elucidating what the radial tour does and some of the features offered.

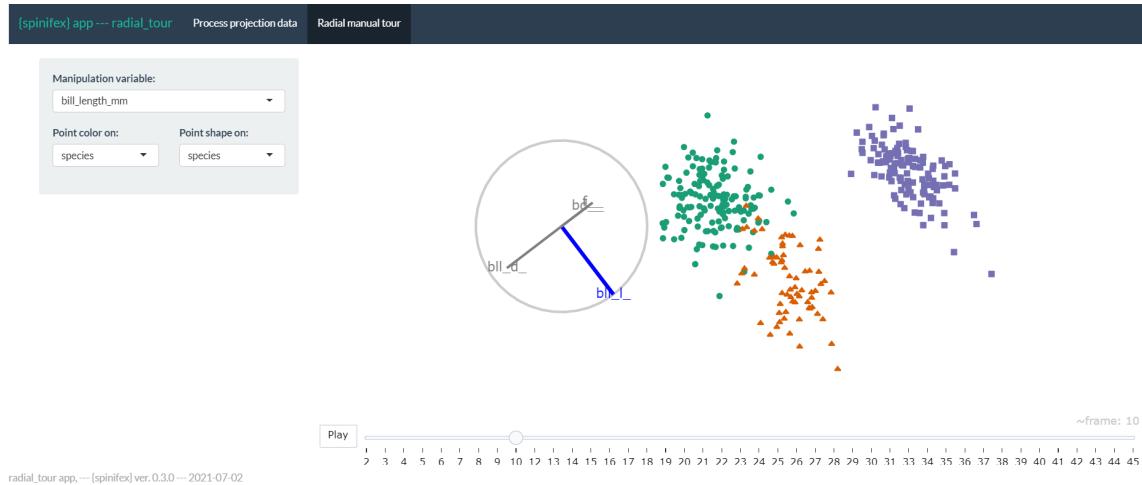


Figure 4.8: Radial tour tab, interactively create radial tours, changing the manipulation variable, color, or shape of the resulting manual tour. Here, the palmer penguins data is being explored, bill length was selected to manipulate as it is the only variable separating the green cluster from the orange. By mapping shape to island of observation, we also notice that the species in green live on all three islands, while the other species live only on one of the islands.

4.5 Extended analysis

This section covers peripheral and extended analysis. First, we explore the demographics of the participants. Then we extend a parallel modeling analysis on log response time. Lastly, we look at the effect ranges and marginal effects of the random effect of the participants and data simulation.

Survey participant demographics

The target population is relatively well-educated people, as linear projections may prove difficult for generalized consumption. Hence we restrict Prolific.co participants to those with an undergraduate degree (58,700 of the 150,400 users at the study time). From this cohort, 108 performed a complete study. Of these participants, 84 submitted the post-study survey, represented in the following heatmap. All participants were compensated for their time at £7.50 per hour, with a mean time of about 16 minutes. Figure 4.9 shows a heat map of the demographics for these 84 participants.

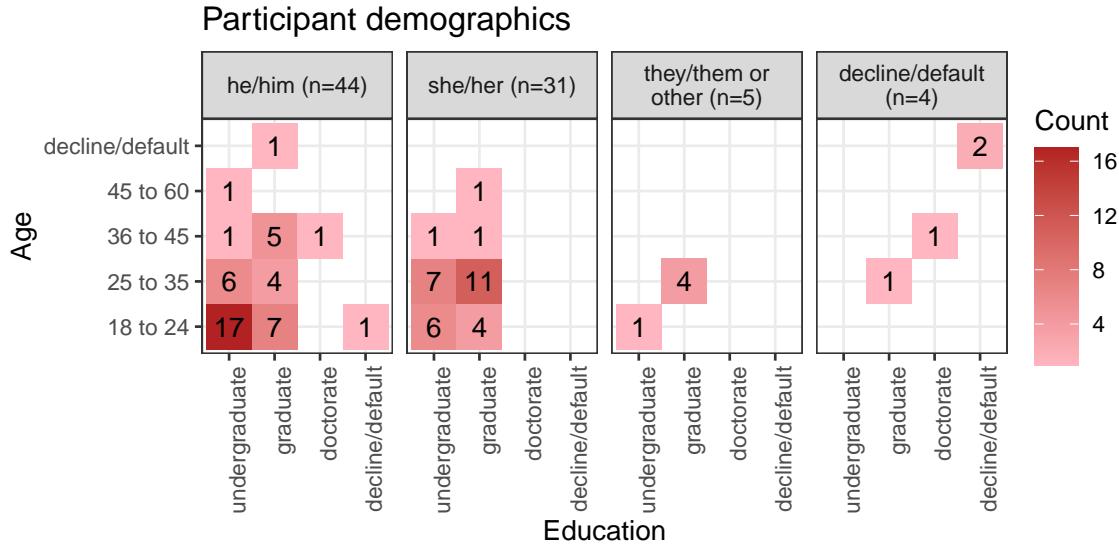


Figure 4.9: Heatmaps of survey participant demographics; counts of age group by completed education as faceted across preferred pronoun. Our sample tended to be between 18 and 35 years of age with an undergraduate or graduate degree.

Table 4.3: Model performance regressing on log response time [seconds], \widehat{Y}_2 random effect models, where each includes random effect terms for participants and simulations. We see the same trade-off where AIC/BIC prefer the simplest factor model, while R^2 and RMSE are the largest in the full multiplicative model. We again select the model $\alpha \cdot \beta + \gamma + \delta$ to explore further as it has relatively high marginal R^2 while having much less complexity than the complete interaction model. Conditional R^2 includes the random effects, while marginal does not.

Fixed effects	No. levels	No. terms	AIC	BIC	R2 cond.	R2 marg.	RMSE
a	1	3	1448	1475	0.645	0.007	0.553
a+b+c+d	4	8	1467	1516	0.647	0.017	0.552
a*b+c+d	5	12	1474	1541	0.656	0.024	0.548
a*b*c+d	8	28	1488	1627	0.673	0.054	0.536
a*b*c*d	15	54	1537	1792	0.7	0.062	0.523

4.5.1 Response time regression

As a secondary explanatory variable, we also want to look at time. First, we take the log transformation of time as it is right-skewed. We repeat the same modeling procedure: 1) Compare the performance of a battery of all additive and multiplicative models. Table 4.3 shows the higher level performance of these models over increasing model complexity. 2) Select the model with the same effect terms, $\alpha \cdot \beta + \gamma + \delta$, and examine its coefficients, displayed in Table 4.4.

Table 4.4: Model coefficients for log response time [seconds] $\widehat{Y}_2 = \alpha \cdot \beta + \gamma + \delta$, with factor = pca, location = 0/100%, shape = EEE, and dim = 4 held as baselines. Location = 50/50% is the fixed term with the strongest evidence and takes less time. In contrast, the interaction term location = 50/50%:shape = EEV has the most evidence and takes much longer on average.

	Estimate	Std. Error	df	t value	Pr(> t)	
(Intercept)	2.71	0.14	42.6	19.06	0.000	***
Factor						
Factorgrand	-0.23	0.12	567.6	-1.97	0.049	*
Factorradial	0.16	0.12	573.5	1.34	0.181	
Fixed effects						
Location33/66%	0.05	0.14	40.9	0.34	0.737	
Location50/50%	-0.05	0.14	42.1	-0.35	0.729	
ShapeEEV	-0.15	0.09	8.3	-1.61	0.145	
Shapebanana	-0.13	0.09	8.3	-1.42	0.192	
Dim6	0.14	0.08	8.3	1.90	0.093	
Interactions						
Factorgrand:Location33/66%	0.24	0.18	580.9	1.34	0.181	
Factorradial:Location33/66%	-0.24	0.18	582.4	-1.32	0.188	
Factorgrand:Location50/50%	0.12	0.18	578.6	0.69	0.491	
Factorradial:Location50/50%	0.05	0.18	584.4	0.25	0.800	

Random effect ranges

The random effect terms further clarify the source of the error. Below we compare the effect ranges attributed to the participant and to the simulations next to their marginal effect on the response, a sort of upper bound of the error they could explain. We look at models regressing accuracy and then log response time.

The residual plots have no noticeable nonlinear trends and contain striped patterns as an artifact from regressing on discrete variables. Figure 4.10 illustrates (T) the effect size of the random terms participant and simulation, or more accurately, the 95% CI from Gelman simulation of their posterior distribution. The effect size of the participant is much larger than simulation. The most extreme participants are statistically significant at $\alpha = .95$, while none of the simulation effects significantly deviate from the null of having no effect size on the marks. In comparison, (B) 95% confidence intervals participation and simulation mean accuracy, respectively.

Residual plots have no noticeable nonlinear trends and contain striped patterns as an artifact from regressing on discrete variables. Figure 4.10 illustrates (T) the effect size

of the random terms participant and simulation, or more accurately, the 95% CI from Gelman simulation of their posterior distribution. The effect size of the participant is much larger than simulation. The most extreme participants are statistically significant at $\alpha = .95$, while none of the simulation effects significantly deviate from the null of having no effect size on the marks. In comparison, (B) 95% confidence intervals participation and simulation mean accuracy, respectively.

Similarly, figure 4.11 shows the Gelman simulations and marginal effects of the simulation and participants for the model with the same terms regressing on log response time.

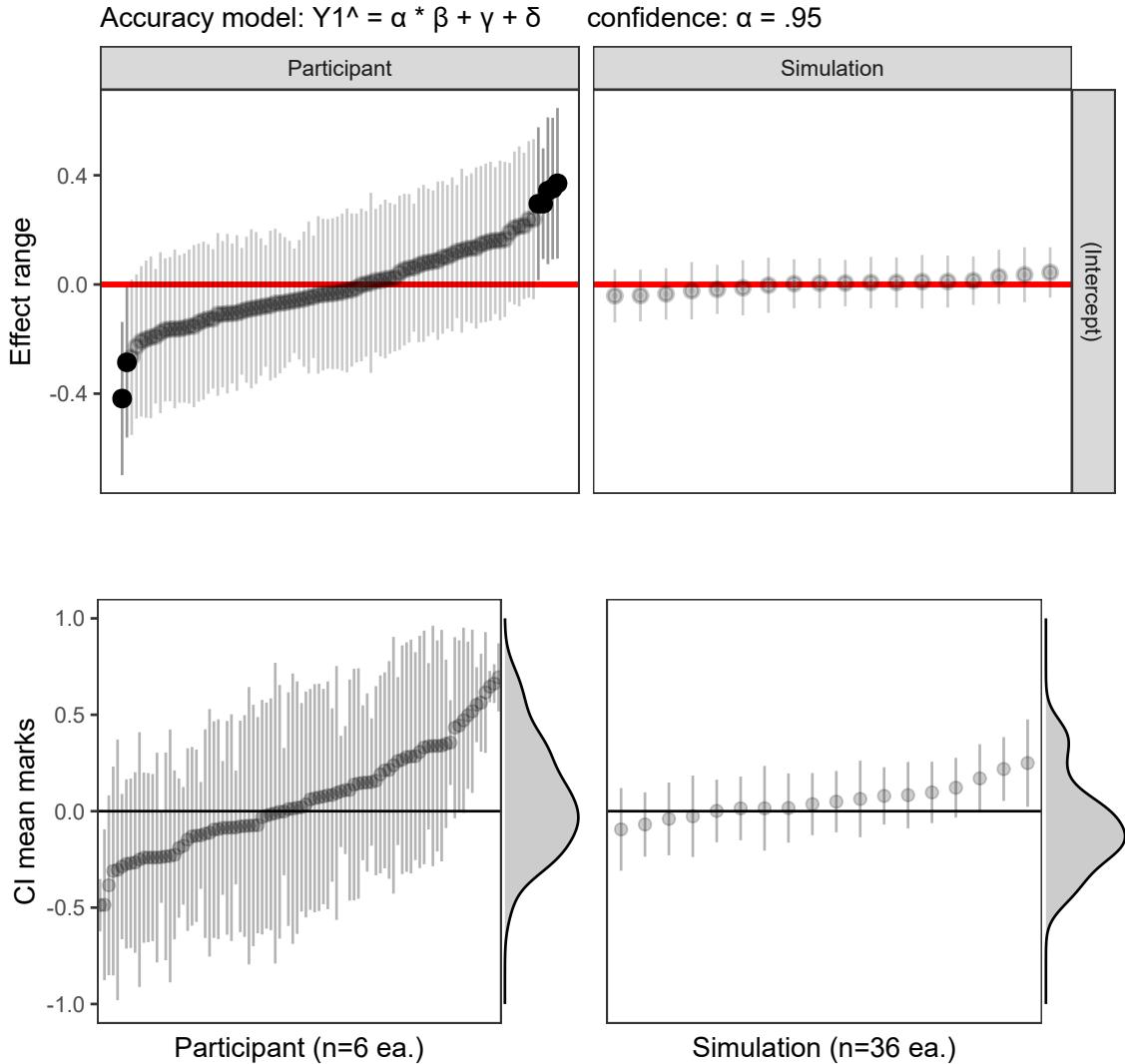


Figure 4.10: Accuracy model: (T) Estimated effect ranges of the random effect terms participant and data simulation of the accuracy model, $\widehat{Y}_1 = \alpha \cdot \beta + \gamma + \delta$. Confidence intervals are created with Gelman simulation on the effect posterior distributions. The effect size of the participant is relatively large, with several significant extrema. None of the simulations deviate significantly. (B) The ordered distributions of the CI of mean marks follow the same general pattern and give the additional context of how much variation is in the data, an upper limit to the effect range. The effect ranges capture about two-thirds of the range of the data without the model. All intervals for $\alpha = .95$ confidence.

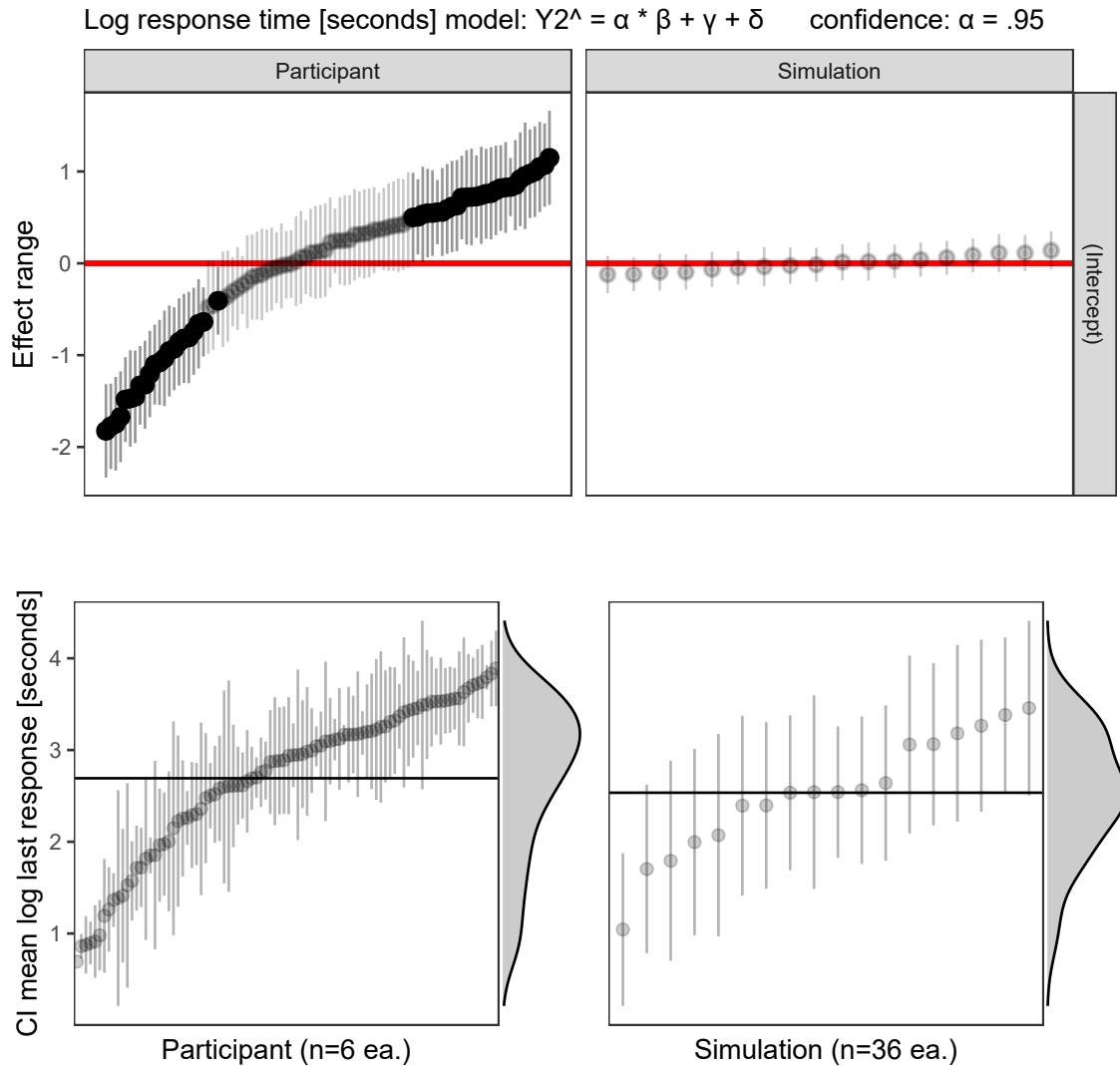


Figure 4.11: Log response time model: (T) The effect ranges of Gelman resimulation on posterior distributions for the time model, $\widehat{Y}_2 = \alpha \cdot \beta + \gamma + \delta$. These show the magnitude and distributions of particular participants and simulations. Simulation has a relatively small effect on response time. (B) Confidence intervals for mean log time by participant and simulation. The marginal density shows that the response times are left-skewed after log transformation. Interpreting back to linear time there is quite the spread of response times: $e^1 = 2.7$, $e^{2.75} = 15.6$, $e^{3.75} = 42.5$ seconds. Of the simulations on the right, the bottom has a large variation in response time, relative to the effect ranges which means that the variation is explained in the terms of the model and not by the simulation itself.

Chapter 5

Exploring Local Explanations of Non-linear Models Using the Radial tour

In the previous chapter discussed the within-participants user study comparing PCA, the grand tour, and the radial tour in a supervised variable attribution task. We found strong evidence that the radial tour leads to large increase in accuracy. Now we can be more confident that the radial tour leads to better analysis of variable-level attribution to features identified in a projection.

Given the interpretability crisis of nonlinear models, it would be interesting to see if the radial tour can help. Specifically, we will investigate using the radial tour to explore variable sensitivity to the structure identified in linear local explanations of nonlinear models. That is under what range of variable importance does an explanation make sense, and when does it fail to be supported by the data. This can provide insight into why an observation is misclassified or otherwise has an extreme residual. We can also test how susceptible a variable's contribution is to discerning the predictions of two observations.

The increased predictive power comes at the cost of interpretability, which has led to the emergence of eXplainable AI (XAI). XAI attempts to shed light on how models use predictors to arrive at a prediction with a point estimate of the linear feature importance in the vicinity of each instance. These can be considered linear projections and can be further explored interactively to understand better the interaction between features used

to make predictions across the predictive model surface. Here we describe interactive linear interpolation used for exploration at any instance and illustrate with examples with categorical (penguin species, chocolate types) and quantitative (football salaries, house prices) output. The methods are implemented in the **R** package **cheem**, available on CRAN.

In Chapter 2 we introduced predictive modeling, the interpretability crisis of nonlinear models, and local explanations — approximations of linear variable importance in the vicinity of one observation. The remainder of this chapter is organized as follows. The following Section, 5.1, induces the SHAP and tree SHAP local explanation. Section 2.3.1 explains the animations of continuous linear projections. Section 5.2 discusses the visual layout in the interactive interface, how they facilitate analysis, data preprocessing, and package infrastructure. Then Section 5.3 illustrates the application to supervised learning with categorical and quantitative output. We conclude with Section 5.4 of the insights gained and directions that might be explored in the future.

5.1 SHAP and tree SHAP local explanations

SHapley Additive exPlanations (SHAP) quantifies the feature contributions of one instance by examining the effect of other features on the predictions. The explanations of SHAP almost all refer to Shapley (1953)'s method to evaluate an individual's contribution to cooperative games by assessing the performance of this player in the presence or absence of other players. Strumbelj and Kononenko (2010) introduced the use of SHAP for local explanations in ML models. The attribution of feature importance depends on the sequence of the included features. The SHAP values are the mean contributions over different feature sequences. The approach is related to partial dependence plots (Molnar, 2020), used to explain the effect of a feature by predicting the response for a range of values on this feature after fixing the value of all other features to their mean. Partial dependence plots are a global approximation of the feature importance, while SHAP is specific to one instance. It could also be considered similar to examining the coefficients from all subsets regression, as described in Wickham, Cook, and Hofmann (2015), which helps to understand the relative importance of each feature in the context of all other candidate features.

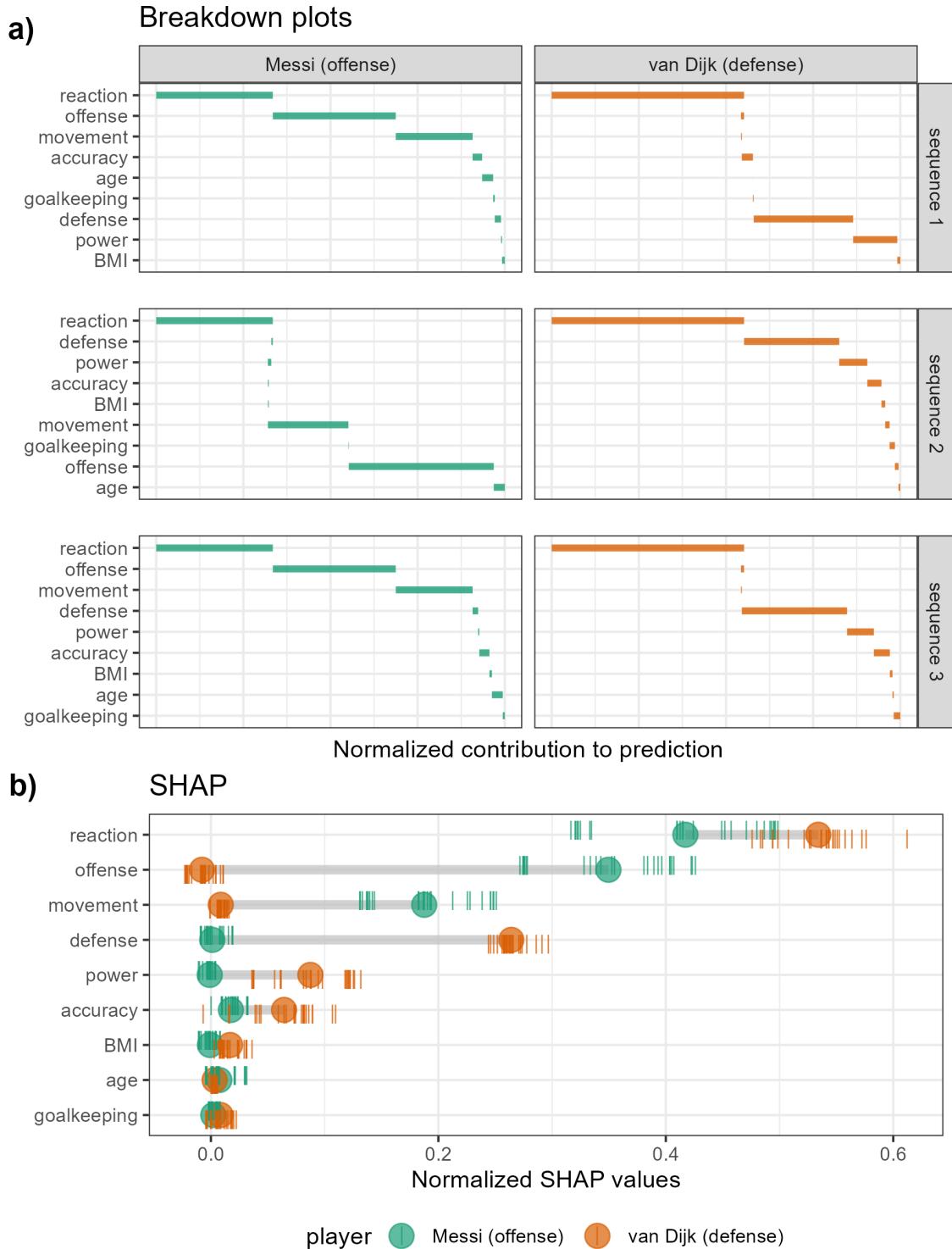


Figure 5.1: Illustration SHAP values for a random forest model FIFA 2020 player wages from nine skill predictors. A star offensive and defensive player are compared, L. Messi and V. van Dijk, respectively. Panel (a) shows breakdown plots of three sequences of the features, order, and magnitude change. Panel (b) shows the distribution of attribution for each feature across 25 sequences of predictors, with the mean displayed as a dot for each player. Offense and movement are important for Messi but not van Dijk, and conversely, defense and power are important for van Dijk but not Messi.

For our application, we use *tree SHAP*, a variant of SHAP that enjoys a lower computational complexity (Lundberg, Erion, and Lee, 2018). Instead of aggregating over sequences of the features, tree SHAP calculates instance-level feature importance by exploring the structure of the decision trees. Tree SHAP is only compatible with tree-based models; we illustrate random forests. The following section will use normalized SHAP values as a projection basis (call this the *attribution projection*) will have coefficients varied to further scrutinize the feature contributions.

Following the use case *Explanatory Model Analysis* (Biecek and Burzykowski, 2021), we use FIFA data to illustrate SHAP use. Consider soccer data from the FIFA 2020 season (Leone, 2020). There are 5000 instances of 9 skill measures (after aggregating highly correlated features). A random forest model is fit, regressing player's wages [2020 Euros] from their skill measurements. We then extract the SHAP values of a star offensive player (L. Messi) and defensive player (V. van Dijk). The results are displayed in Figure 5.1. We expect to see a difference in the attribution of the feature importance across the two positions of the players, which would be interpreted as how the player's salary depends on this combination of skill sets. Plot (b) is a modified breakdown plot (Gosiewska and Biecek, 2019) where the order of features is fixed, so the two instances can be more easily compared.

In summary, these plots highlight how local explanations bring interpretability to a model, at least in the vicinity of their instances. In this instance, two players with different positions receive different profiles of feature importance to explain the prediction of their wages.

5.2 The cheem viewer

To explore the local explanations, an ensemble of plots (Unwin and Valero-Mora, 2018) is provided, called the *cheem viewer*. There are two primary plots: the **global view** to give the context of all of the SHAP values, and the **radial tour view** to explore the local explanations with user-controlled rotation. In addition, there are numerous user inputs, including feature selection for the radial tour and instance selection for making comparisons. There are different plots used for the categorical and quantitative responses. Figures 5.2

and 5.3 are screenshots showing the cheem viewer for the two primary tasks: classification (categorical response) and regression (quantitative response).

5.2.1 Global view

The global view provides the context of all instances and facilitates the exploration of the separability of the data- and attribution-spaces. Both of these spaces are of dimension $n \times p$, where n is the number of instances and p is the number of predictors. The attribution space corresponds to the local explanations for each instance, each a vector of p values.

A visualization is provided by the first two principal components of the data (left) and the attribution (middle) spaces. These single 2D projections will not reveal all of the structure of higher-dimensional space, but they are useful visual summaries. In addition, a plot of the observed against predicted response values is also provided (Figures 5.2b, 5.3a), to help identify instances poorly predicted by the model. For classification tasks, misclassified instances are circled in red if applicable. Linked brushing between the plots is provided, and a tabular display of selected points helps to facilitate exploration of the spaces and the model (shown in Figures 5.2a,c).

While the comparison of these spaces is interesting, a main purpose of the global view is to enable the selection of instances to explore the local explanations. The projection attribution of the primary instance (PI) is examined and typically viewed with an optional comparison instance (CI). These instances are highlighted as asterisk and \times , respectively.

5.2.2 Radial tour

The local explanations for all observations are normalized (squared sum of values adds to 1), and thus, the relative importance of features can be compared across all instances. These are depicted as a vertical parallel coordinate plot, where each line connects one instance's feature attribution (Figures 5.2e and 5.3e). The attribution projections of the PI and CI are shown as dashed and dotted lines, respectively. From this plot, we would read the range of importances across all instances can be interpreted. For classification, one would look at differences between groups on any feature. For example, Figure 5.2e suggests that `b1` is important for distinguishing the green class from the other two. For

regression, one might generally observe which features have low values for all instances (not important), for example, `BMI` and `pwr` in Figure 5.3e, and which have a range of high and low values (e.g. `off`, `def`) suggesting important for some instances and not important for other instances.

The overlaid bars on the parallel coordinate plot represent the attribution projection of the PI. (Remember that the PI is interactively selected from the global view). The attribution projection is an approximation of the feature importance for the prediction of this instance. This combination of features best explains the difference between the mean response and an instance's predicted value. It is not an indication of the local shape of the model surface. That is, it is not some indication of the tangent to the curve at this point.

The attribution projection of the PI is the initial 1D basis in a radial tour, displayed as a density plot for a categorical response (Figure 5.2f), and as scatterplots for a quantitative response (Figure 5.3f). The PI and CI are indicated by vertical dashed and dotted lines, respectively. The user uses the radial tour to vary the contribution of the selected feature between 0-1. Doing so tests the sensitivity of structure (class separation or strength of relationship) to the feature's contribution. For classification, if the separation between classes diminishes when the feature contribution is reduced then this suggests that the feature is important for the class separation. For regression, if the relationship scatterplot weakens when the feature contribution is reduced then this suggests that the feature is important for accurately predicting the response.

5.2.3 Classification task

Selecting a misclassified instance as PI and a correctly classified point nearby in data space as CI makes it easier to examine the features most responsible for the error. The global view (Figure 5.2c) displays the model confusion matrix. The radial tour is 1D, plotted as a density where color indicates class. A scroll bar here enables the user to vary the contribution of each feature to explore the sensitivity of the separation to that feature.

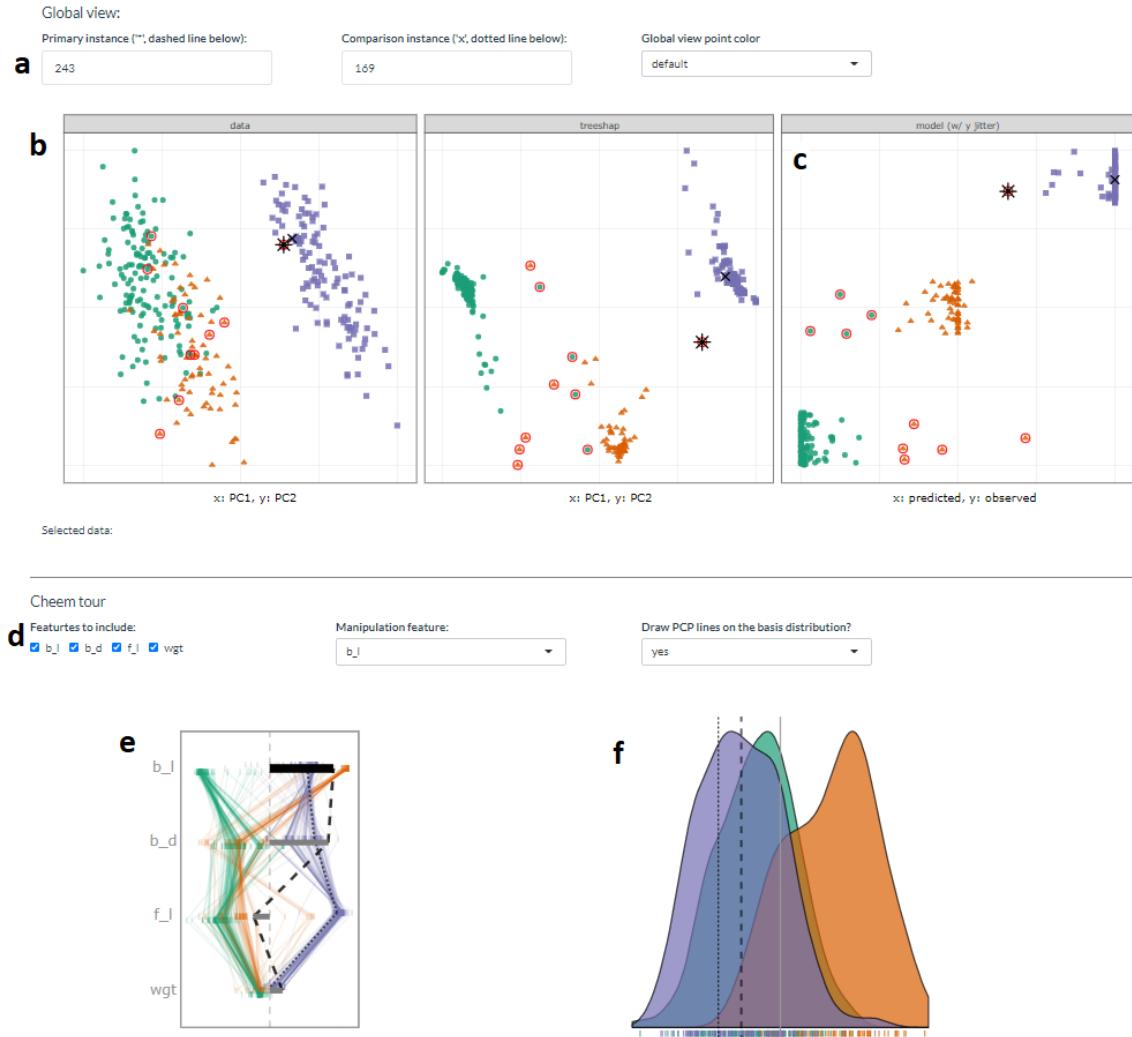


Figure 5.2: Overview of the cheem viewer for classification tasks. Global view inputs, (a), set the PI, CI, and color statistic. Global view, (b) PC1 by PC2 approximations of the data space and attribution space. (c) prediction by observed y (visual of the confusion matrix for classification tasks). Points are colored by predicted class, and red circles indicate misclassified instances. Radial tour inputs (d) select features to include and which feature is changed in the tour. (e) shows a parallel coordinate display of the distribution of the feature attributions while bars depict contribution for the current basis. The black bar is the variable being changed in the radial tour. Panel (f) is the resulting projection of the data indicated as density in the classification case.

5.2.4 Regression task

Selecting an inaccurately predicted instance as PI and an accurately predicted instance, with similar feature values, as CI is a useful way to understand how the model is failing or not. The global view (Figure 5.3a) shows a scatterplot of the observed vs predicted values, which should exhibit a strong relationship if the model is a good fit. The points can be colored by a statistic, residual, a measure of outlyingness (log Mahalanobis distance), or correlation, to help with understand where the model fits better or worse.

In the radial tour view, the observed response and the residuals (vertical) are plotted against the attribution projection of the PI (horizontal). The attribution projection can be interpreted similarly to the predicted value from the global view plot. It represents a linear combination of the features, and a good fit would be indicated when there is a strong relationship seen with the observed values. This can be viewed as a local linear approximation if the fitted model is nonlinear. As the contribution of a feature is varied, if the value of the PI does not change much, it would indicate that the prediction for this instance is NOT sensitive to that feature. Conversely, if the predicted value varies substantially, the prediction is very sensitive to that feature, suggesting that the feature is very important for the PI's prediction.

5.2.5 Interactive features

The application has several reactive inputs that affect the data used, aesthetic display, and tour manipulation. These reactive inputs make the software flexible and extensible. The application also has more exploratory interactions to help link points across displays and reveal structure found in different spaces.

A tooltip displays instance number/name and classification information while the cursor hovers over a point. Linked brushing allows the selection of points (left click and drag) where those points will be highlighted across plots. The information corresponding to the selected points is populated on a dynamic table. These interactions aid exploration of the spaces and, finally, identification of a primary and comparison instance.

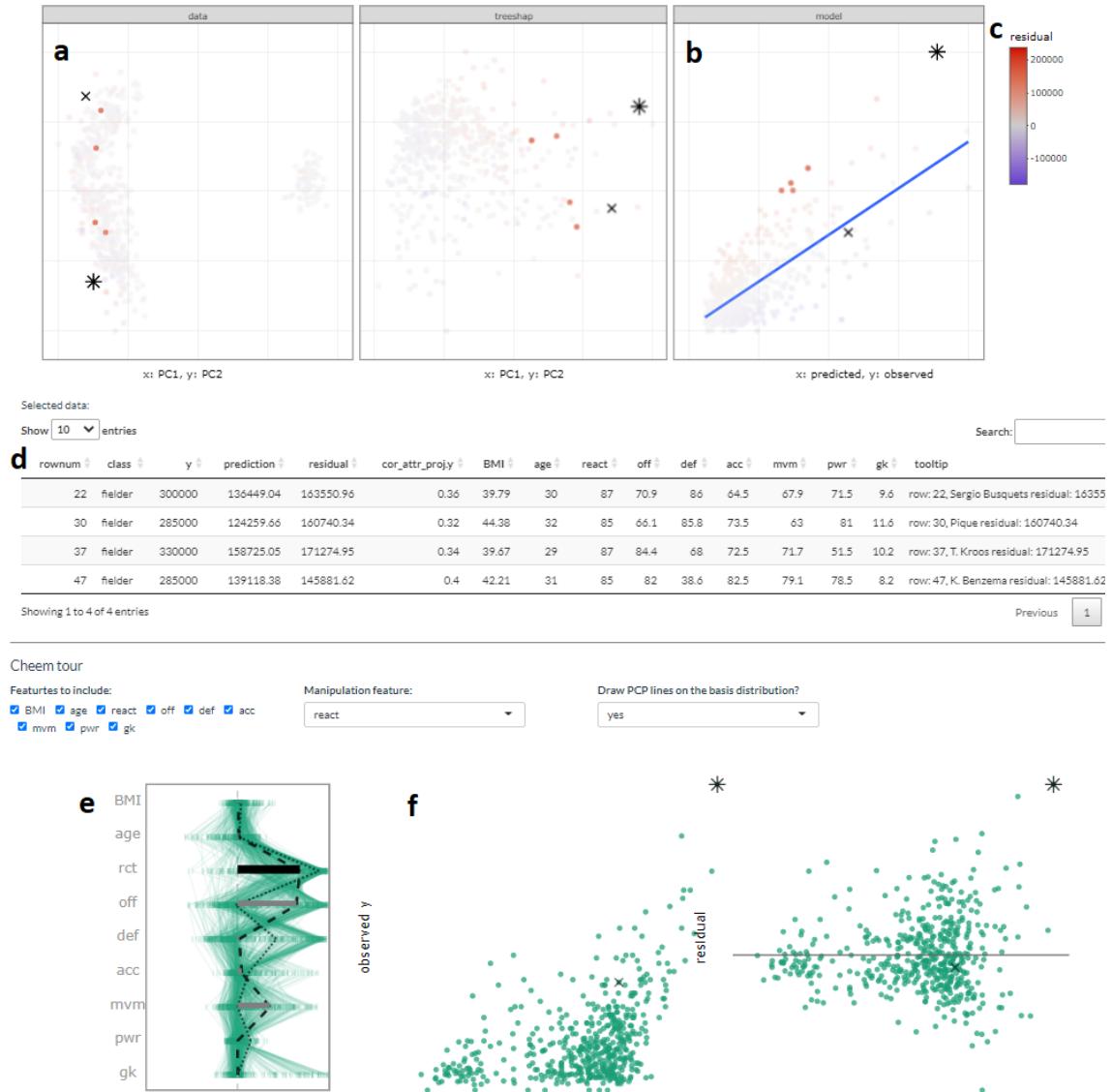


Figure 5.3: Overview of the cheem viewer for regression task highlighting the differences from the classification task and interactive features. Panel (a) PCA of the data and attributions spaces, (b), residual plot, predictions by observed values. Four points are selected points and highlighted in the PC spaces and tabularly displayed. Coloring on a statistic (c) highlights structure organized in the attribution space. Interactive tabular display (d) populates when instances are selected. Contribution of the 1D basis affecting the horizontal position (e) parallel coordinate display of the feature attribution from all observations, and horizontal bars show the contribution to the current basis. Regression projection (f) uses the same horizontal projection and fixes the vertical positions to the observed y and residuals (left and right).

5.2.6 Preprocessing

It is vital to mitigate the render time of visuals, especially when users may want to iterate many times. All computational operations should be prepared before runtime. The work remaining when an application is run solely reacts to inputs and rendering visuals and tables. Below we discuss the steps and details of the reprocessing.

- **Data:** predictors and response are unscaled complete numerical matrix. Most models and local explanations are scale-invariant.
- **Model and explanation:** any model can be used with this method. Currently, we apply random forest models via the package **randomForest** (Liaw and Wiener, 2002), compatibility tree SHAP. We use modest hyperparameters, namely: 125 trees, number features randomly sampled at each split, $mtry = \sqrt{p}$ or $p/3$ for classification and regression, and minimum size of terminal nodes $\max(1, n/500)$ or $\max(5, n/500)$ for classification and regression. Tree SHAP is calculated for *each* instance using the package **treeshap** (Kominsarczyk et al., 2021). This implementation aggregates exhaustively over all trees' attribution, and we opt not to fit interactions of features.
- **Cheem view:** after the model and full explanation space are calculated, we scale each feature by standard deviations away from the mean to achieve common support for visuals. Statistics for mapping to color are computed on the scaled spaces. Interactive tabular display reports the original values.

The time to preprocess the data will vary significantly with the model and local explanation. For reference, the FIFA data, 5000 instances of nine explanatory features, took 2.5 seconds to fit a random forest model of modest hyperparameters. Extracting the tree SHAP values of each instance took 270 seconds combined. PCA and statistics of the features and attributions took 2.8 seconds. These runtimes were from a non-parallelized R session on a modern laptop, but suffice to say that most of the time will be spent on the local attribution. An increase in model complexity or data dimensionality will quickly become an obstacle. Its reduced computational complexity makes tree SHAP a good candidate to start. (Alternatively, the package **fastshap** (Greenwell, 2020) claims extremely low

runtimes, attributed to fewer calls to the prediction function, partial implementation in C++, and efficient use of logical subsetting.)

5.2.7 Package infrastructure

The above-described method and application are implemented as an open-source **R** package, **cheem** available on [CRAN](#). Preprocessing was facilitated with models created via **randomForest** (Liaw and Wiener, 2002) and explanations calculated with **treeshap** (Kominsarczyk et al., 2021). The application was made with **shiny** (Chang et al., 2021). The tour visual is built with **spinifex** (Spyrison and Cook, 2020). Both views are created first with **ggplot2** (Wickham, 2016) and then rendered as interactive HTML widgets with **plotly** (Sievert, 2020). **DALEX** (Biecek, 2018) and the free ebook, *Explanatory Model Analysis* (Biecek and Burzykowski, 2021) were a huge boon to understanding local explanations and how to apply them.

5.2.8 Installation and getting started

The package can be installed from GitHub using the following **R** code:

```
# Install remotes if absent
if(require("remotes") == FALSE) install.packages("remotes")
remotes::install.packages("cheem", dependencies = TRUE)
library("cheem")
run_app()
```

To process your own data, you will need to use the **treeshap** package, which can be installed from GitHub:

```
remotes::install_github('ModelOriented/treeshap')
```

Follow the examples provided with the package to compute the local explainers, (see `?cheem_ls`). The application expects the return of call from `cheem_ls()` which is then saved to an .rds file with `saveRDS()`. Alternatively, the cheem viewer shiny app can be directly accessed at ebsmonash.shinyapps.io/cheem_initial/.

5.3 Case studies

To illustrate the use of the cheem method, we apply it to modern datasets, two classification examples and then two of regression.

5.3.1 Palmer penguin, species classification

The Palmer penguins data (Gorman, Williams, and Fraser, 2014; Horst, Hill, and Gorman, 2020) was collected on three species of penguins foraging near Palmer Station, Antarctica. The data was publicly available to substitute for the overly-used iris data and is quite similar in form. After removing incomplete instances, there are 333 instances, and we will use the four physical measurements, bill length (`b1`), bill depth (`bd`), flipper length (`f1`), body mass (`bm`), for this illustration. A random forest model was fit with species as the response feature.

Figure 5.4 shows plots from the cheem viewer for exploring the random forest model on the penguins data. Panel (a) shows the global view, and panel (b) shows several 1D projections generated with the radial tour. Penguin 243, a Gentoo (purple), is the PI because it has been misclassified as a Chinstrap (orange).

There is more separation visible in the attribution space than the data space, as would be expected. The predicted vs observed plot reveals a handful of misclassified instances. We will explore why a Gentoo has been wrongly labeled as a Chinstrap for this illustration. The PI is a misclassified point (represented by the asterisk in the global view and a dashed vertical line in the tour view). The CI is a correctly classified point (represented by an \times and a vertical dotted line).

The radial tour starts from the attribution projection of the misclassified instance (b, left). The important features identified by SHAP in the (wrong) prediction for this instance are mostly `b1` and `bd` with small contributions of `f1` and `bm`. This projection is a view where the Gentoo (purple) looks much more likely for this instance than Chinstrap. That is, this combination of features is not particularly useful because the PI looks very much like other Gentoo penguins. To explore this, we use the radial tour to vary the contribution of flipper length (`f1`). (In our exploration, this was the third feature explored. It is typically

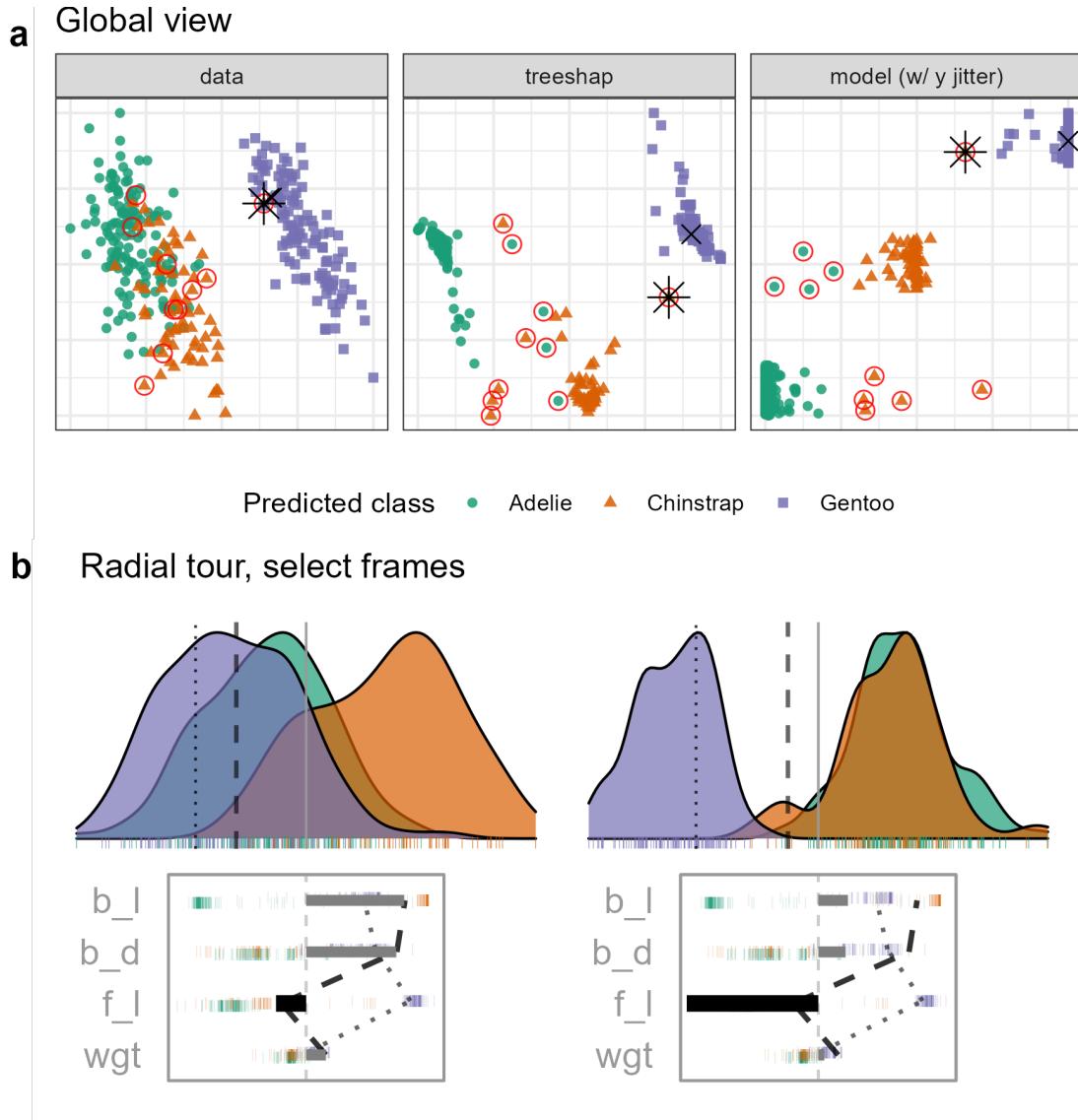


Figure 5.4: Examining the SHAP values for a random forest model classifying Palmer penguin species. The PI is a Gentoo (purple) Chinstrap (orange) penguin that is misclassified as a Chinstrap (orange), marked as an asterisk in (a), and the dashed vertical line in (b). The radial view shows varying the contribution of *fl* from the initial attribution projection (b, left), which produces a linear combination where the PI is more probably (higher density value) a Chinstrap than a Gentoo (b, right). (The animation of the radial tour is at vimeo.com/666431172.)

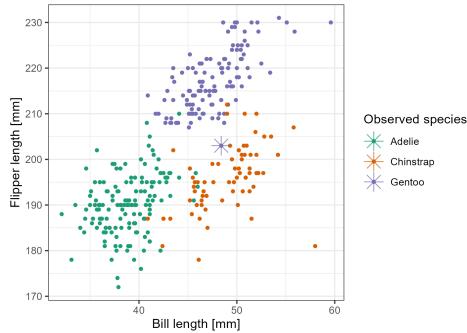


Figure 5.5: Checking what is learned from the cheem viewer. This is a plot of flipper length (f_l) and bill length (b_l), where an asterisk highlights the PI. A Gentoo (purple) misclassified as a Chinstrap (orange). The PI has an unusually small f_l length which is why it is confused with a Chinstrap.

useful to explore the features with larger contributions, here b_l and b_d , but when doing this, nothing was revealed about how the PI differed from other Gentoos). On varying f_l as it contributes more to the projection (b, right), we see that more, and more, this penguin looks like a Chinstrap. This suggests that f_l should be considered an important feature for explaining the (wrong) prediction.

Figure 5.5 confirms that flipper length (f_l) is important for the confusion of the PI as a Chinstrap. Here, flipper length and body length are plotted, and we can see that the PI is closer to the Chinstrap group in these two features, mostly because it has an unusually low value of flipper length relative to other Gentoos. From this view, it makes sense that its a hard instance to account for as decision trees can only partition only vertical and horizontal lines.

5.3.2 Chocolates, milk/dark chocolate classification

The chocolates dataset consists of 88 instances of ten nutritional measurements determined from their labels and labeled as either milk or dark. Dark chocolate is considered healthier than milk. The data was collected by students during the Iowa State University class STAT503 from nutritional information from the manufacturer's website and normalized to 100g equivalents. The data is available in the **cheem** package. A random forest model is used for the classification of chocolate type.

It could be interesting to examine the nutritional properties of any dark chocolates that have been misclassified as milk. A reason to do this is that a dark chocolate nutritionally more like milk should not be considered a healthy alternative. It is interesting to explore which of the nutritional features contribute most to misclassification.

This type of exploration is shown in Figure 5.6, where a chocolate labeled dark but predicted to be milk is chosen as the PI (instance 22). It is compared with a CI that is a correctly classified dark chocolate (instance 7). The PCA plot, and the tree SHAP PCA plots (a) show a big difference between the two chocolate types but with confusion for a handful of instances. The misclassifications are more apparent in the observed vs predicted plot, and can be seen to be mistaken in both ways: milk to dark and dark to milk.

The attribution projection for chocolate 22 suggests that Fiber, Sugars, and Calories are most responsible for its incorrect prediction. The way to read this plot is to see that Fiber has a large negative value while Sugars and Calories have reasonably large positive values. In the density plot, instances on the very left of the display would have high values of Fiber (matching the negative projection coefficient) and low values of Sugars and Calories. The opposite would be interpreting a point with high values in this plot. The dark chocolates (orange) are mostly on the left, and this is a reason why they are considered to be healthier: high fiber and low sugar. The density of milk chocolates is further to the right, indicating that they generally have low fiber and high sugar.

The instance of interest (dashed line) can be viewed against the comparison instance (dotted line). Now one needs to pay different attention to the parallel plot of the SHAP values, which are local to a particular instance, and the density plot, which is the same projection of all instances as specified by the SHAP values of the instance of interest.

We can quickly compare the feature contributions to the two different predictions from the parallel coordinate plot. The instance of interest differs with the comparison primarily on the Fiber feature, which suggests that this is the reason for the incorrect prediction.

From the density plot, which is the attribution projection corresponding to the instance of interest, both instances are more like dark chocolates. If we vary the contribution of Sugars

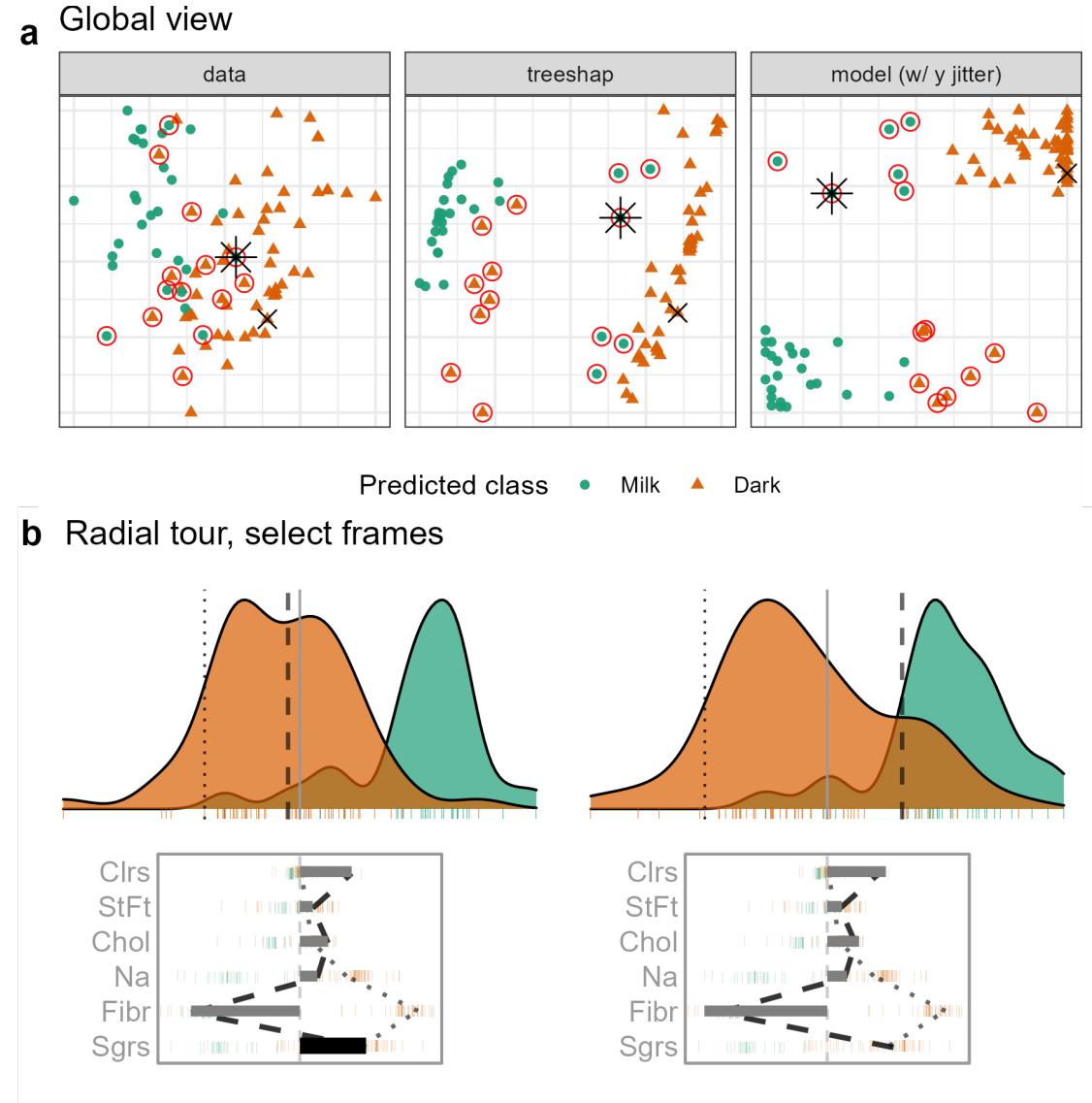


Figure 5.6: Examining the local interpretation for a PI which is dark (orange) chocolate incorrectly predicted to be milk (green). From the attribution projection, this chocolate correctly looks more like dark more than milk, which suggests that the local explanation does not help understand the prediction for this instance. So, we vary the contribution of Sugar – reducing it corresponds primarily with increasing Fiber. When Sugar is zero, Fiber is contributing strongly towards the left. In this particular view, the PI is closer to the bulk of the milk chocolates, suggesting that the prediction put a lot of importance on Fiber. This chocolate is a rare dark chocolate without any Fiber leading to it being mistaken for a milk chocolate. (A video of the tour animation can be found at vimeo.com/666431143.)

and altogether remove Sugars from the projection, this is where the difference becomes apparent. When primarily Fiber is examined, instance 22 looks more like a milk chocolate.

It would also be interesting to explore the inverse case. This would describe which features lead to a milk chocolate being misclassified as dark and how those attributions differ from the previous misclassification. Chocolate 84 is just this case, and we compare it with a correctly predicted milk chocolate (instance 71). This exploration is shown in Figure 5.7.

The difference of position in the tree SHAP PCA with the previous case is quite large; this gives an approximate feel that the attribution should be quite different. Looking at the initial contribution, we find this to be the case. Previously Fiber was very important while it is absent from the attribution in this case. Conversely, Calories from Fat and Total Fat are highly attributed here, while unimportant in the preceding case.

Comparing the attribution with the CI (dotted line), discrepancies in Sodium and Fiber are identified. We opt to vary Sodium. The instance looks slightly more like its observed milk than predicted dark chocolate even in the initial projection. The misclassification appears least supported when the basis reaches sodium attribution of typical dark chocolate.

5.3.3 FIFA, wage regression

The 2020 season FIFA data (Leone, 2020; Biecek, 2018) contains many skill measurements of soccer/football players and wage information. From a correlation matrix plot, nine higher-level skill groupings were identified and aggregated. A random forest model is fit from these aggregations and regress player wages [2020 euros]. The model was fit from 5000 instances before being thinned to 500 players to mitigate occlusion and render time. Continuing from the exploration in section 2.6, we are interested to see the difference in attribution based on the exogenous player position. That is, the model should be able to use multiple linear profiles to better predict the wages from different field positions of players. We compare a leading offensive fielder (L. Messi) with that of a top defensive fielder (V. van Dijk). The same instances were used in figure 5.1.

With figure 5.8, we will test the support of the local explanation. Offensive and reaction skills (`off` and `rct`) are both crucial to explaining a star offensive player. If either of them

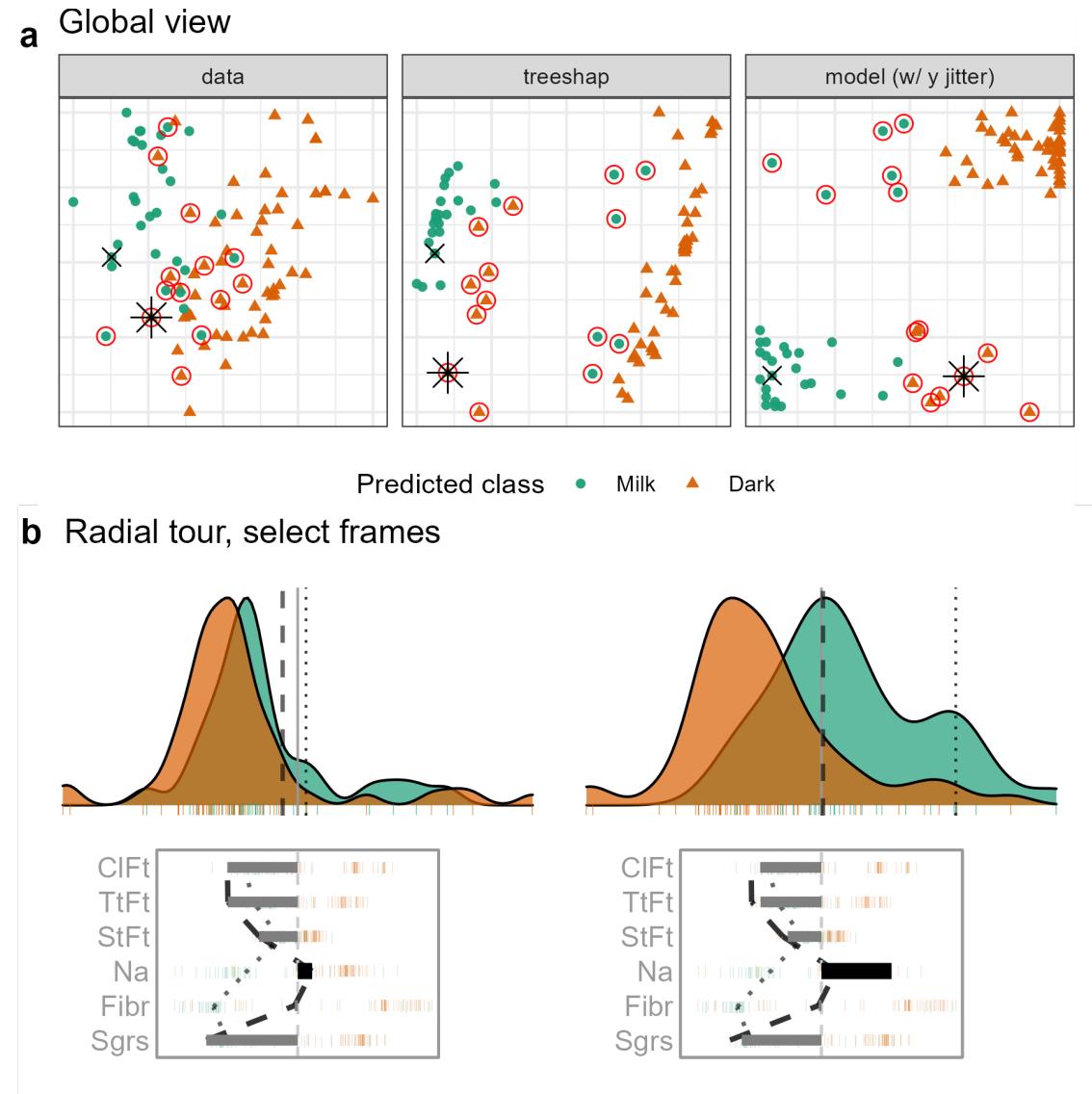
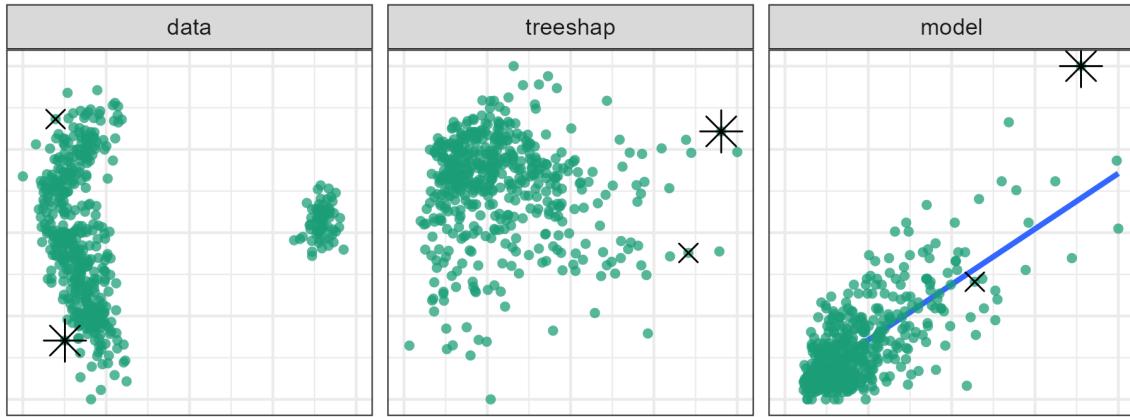


Figure 5.7: Examining the local interpretation for a PI which is milk (green) chocolate incorrectly predicted to be dark (orange). In the attribution projection, the PI could be either milk or dark. Sodium and Fiber have the largest differences in attributed feature importance, both with low values relative to other milk chocolates. We suspect the lack of use of these variables is what has contributed to the mistake, so we vary the contribution of Sodium. If Sodium had a larger contribution to the prediction (like in this view), the PI would look more like other milk chocolates. (A video of the tour animation can be found at vimeo.com/666431148.)

a

Global view



b Radial tour, select frames

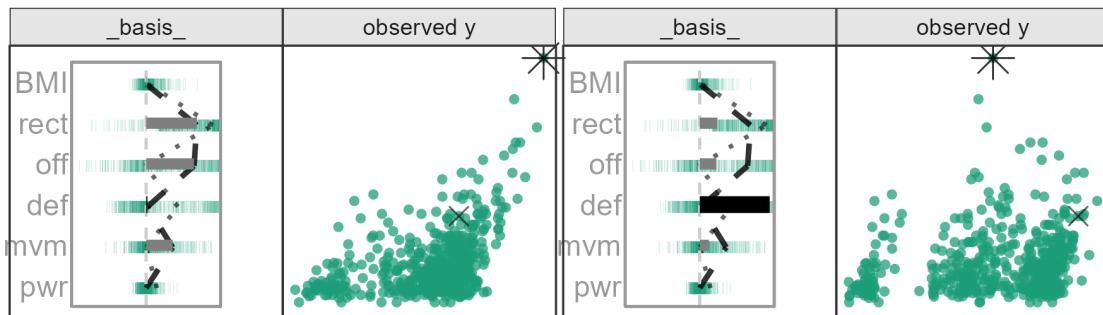


Figure 5.8: FIFA 2020 data, a random forest model, regresses wages [2020 Euros] from nine aggregated skill measurements. The PI is a star offensive player (L. Messi) compared with a top defensive player (V. van Dijk). We remove three features with low attribution from both players. The attribution projection starts with the selected instance on the right. We vary the contribution from defense (`def`). The star offensive player is not distinguished in the horizontal direction. At this point, defensive players have been rotated to the highest horizontal value. (A video of the animated radial tour can be found at vimeo.com/666431163.)

were rotated out, the other will be rotated into frame, maintaining a far-right position. However, when we vary the defensive skills, the other skills are rotated out of the frame.

As the contribution of defensive skills increases, Messi's is no longer separated from the group. Players with high values in defensive skills are now the rightmost points. In terms of what-if analysis, the difference between the data mean and his predicted wages would be halved if Messi's tree SHAP attributions at these levels.

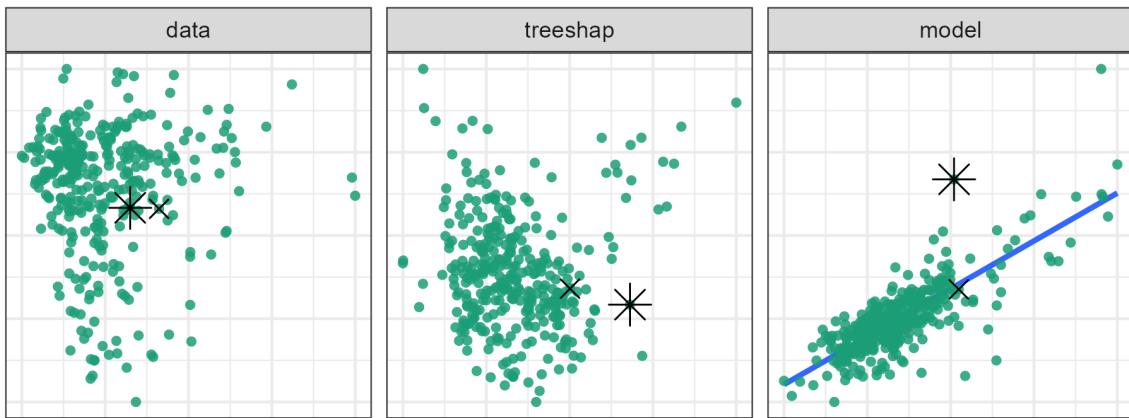
5.3.4 Ames housing 2018, sales price regression

Ames 2018, housing data was subset to North Ames (the neighborhood with the most house sales). The remaining are 338 house sales. A random forest model has regressed this price with the features Lot Area (**LtA**), Overall Quality (**Qlt**), Year the house was Built (**YrB**), Living Area (**LvA**), number of Bathrooms (**Bth**), number of Bedrooms (**Bdr**), total number of Rooms (**Rms**), Year the Garage was Built (**GYB**), and Garage Area (**GrA**). Using interaction from the global view, we select a house with an extreme negative residual and an accurate instance close to it in the data.

Figure 5.9 selects the house sale 74, a sizable under prediction with a large Lot Area contribution. The CI has a similar predicted price though the prediction was accurate and gives almost no attribution to lot size. The attribution projection is places instances with high living areas to the right. We control the contribution of this feature. As the contribution of lot area decreases, the predictive power decreases for the PI, while the CI remains stationary. This large of importance in the Living Area is relatively uncommon. Boosting tree models may be more resilient to such an under-prediction as up-weighting this residual would force its inclusion in the final model.

a

Global view



b Radial tour, select frames

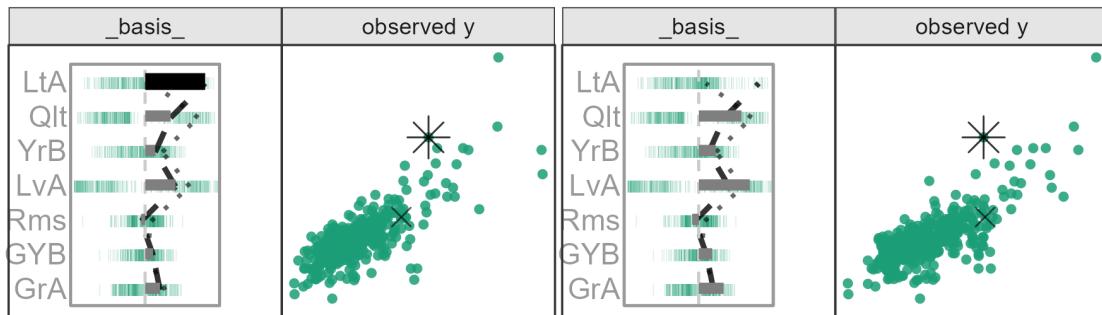


Figure 5.9: Exploring an instance with a large residual as the PI from fitting sales price [USD] to other variables in the Ames housing 2018. The sale price of the PI was under-predicted. The local explanation indicates a sizable attribution to Lot Area (LtA). The CI has a similar predicted sales price and smaller residual and has minimal attribution to Lot Area. In the attribution projection, the PI has a higher sales price than the CI. Reducing the contribution of lot area brings these two prices in line. This suggests if the model did not consider Lot Area, then the two houses would be quite similar. That is, the large residual is due to a lack of factoring in the lot area for the prediction of PI's sales price. (A video showing the animation is at vimeo.com/666431134.)

5.4 Discussion

New tools to assist with the interpretability of black-box models is increasingly important. This chapter provides a technique that builds on local interpretations to explore the feature importance local to an instance. The local interpretations form an attribution projection from which feature contributions are varied using a radial tour. Several diagnostic plots are provided to assist with understanding the sensitivity of the prediction to particular features. A global view shows the data space, explanation space, model fit. The user can interactively select instances to compare, contrast, and study further.

The approach has been illustrated on four data examples using random forest models and using the tree SHAP local explanation. In the penguins example, we showed how the misclassification of a penguin arose due to it having an unusually small flipper size compared to others of its species. This was verified by making a follow-up plot of the data, including this variable. In the chocolates example, we showed how a dark chocolate was misclassified primarily due to its value on Fiber, and a milk chocolate was misclassified as dark due to its lowish Sodium value. For the FIFA example, we show how low Messi's salary would go if it depended on defensive skills. In the Ames housing data, an inaccurate prediction for a house was likely due to the Lot Area not being effectively used.

The approach is manually intensive and thus only feasible for investigating a few instances. The approach is to select instances that the model has not done well and compare it with an instance where it did fit well. The radial tour launches from the attribution projection to enable exploration of the sensitivity of the prediction to any feature. It can be helpful to make additional plots of the features and responses in order to cross-check interpretations made from the cheem viewer. This methodology provides an additional tool in the box for studying model fitting.

An implementation is provided in the open-source **R** package **cheem**, available on [CRAN](#). Example data sets are provided, and you can upload your data after model fitting and computing the local explanations. In theory, this approach would work with any black-box model but the implementation currently only calculates tree SHAP for tree-based models supported by **treeshap** (tree based models from **randomForest**, **ranger**, **gbm**,

xgboost, **lightgbm**, or **catboost**). Tree SHAP was selected because of its computational efficiency. The SHAP and oscillation explanations could be added with the use of the `DALEX::explain()` and would be an excellent direction to extend the work (Biecek, 2018; Biecek and Burzykowski, 2021).

Chapter 6

Conclusion

Cleanup

Most data are multivariate. This thesis make important contributions to the interactive visualization of data where all variables are quantitative. The methods are build on dynamic animation of linear projections, called tours. The thesis developed the radial tour, illustrating that it improved understand variable importance in higher dimensional data, and can be used to better understand black box models (XAI).

6.1 Contributions

The contributions of this thesis can be split into scientific knowledge and software.

6.1.1 Scientific knowledge

link back RQ/contributions in introduction can add more detail

Chapter 3 develops radial tour methodology... «< I clarify the use of Rodrigues' rotation formula to solve the rotation matrix with the definition of two manipulation angles. This sets up a scaffolding to extend the manual tour to three dimensions with another rotation angle to span the manipulation space. This work also supports the use of manual tours by illustrating use cases on high-energy physics data.

Then I define a task and accuracy measure for a variable attribution of the separation of two clusters. A crowdsourced user study is conducted comparing the performance from PCA, the grand tour, and the radial tour. There is considerable evidence for a sizable improvement in accuracy with the use of the radial tour, which participants also subjectively prefer.

Lastly, I purpose the **cheem** analysis to explore the variable support of local explanations from black-box models. From a nonlinear model, the local explanations of all observations are calculated. Then compare data space, attribution space, and model information side-by-side as an ensemble graphic. From this, identify a primary observation to explore the explanation in detail. The normalized attribution this point becomes the starting basis for a radial tour. Then vary the contributions of variables, testing their contribution's sensitivity to the predictive discernment identified in the explanation.

6.1.2 Software

The **R** package **spinifex** facilitates the creation of manual tours, which allow an analyst control over the contributions of a variable. It handles data transformations and the identification of various starting bases. It creates a framework for the layered composition of tours interoperably with the previous **tourr** package. This composition will feel at home to **ggplot2** users. After composition, tours can be animated and exported as interactive HTML widgets or fixed animations as gif, or mp4 files. Vignettes and interactive application help users rapidly understand the concepts of facilitated work. The impact of **spinifex** can be seen in two ways. My contributions to **spinifex** and **tourr** won the ACEMS Impact and Engagement Award, 2018. Furthermore, the package is available on [CRAN](#) with vignettes and version notes on its [pkgdown](#) site. It has been downloaded over 14,400 times from CRAN between 09 April 2019 and 28 November 2021.

The second software contribution is the **cheem** package. It facilitates the tree SHAP local explanations from tree-based models. The global view visualizes data space, explanation space, and model information as an ensemble graphic with linked brushing. After observations of interest have been identified in the full context, their local explanation is explored with the radial tour. Doing so allows the support of the explanation to be tested.

An interactive application uses these visuals to facilitate the cheem analysis. Several preprocess datasets are included and allows analysts to upload their data after processing. This package was recently uploaded to [CRAN](#) and has a corresponding [pkgdown site](#).

6.2 Limitations and future work

Break apart limitations and future work scope:
mostly scatterplot display, modest dimensionality

Below a call and response format is used to discuss the limitations and possible future work to address them following chapter order.

The **spinifex** package is built around one and two-dimensional manual tours. The layered composition of tours is interoperable with $d \in [1, 2]$ tours produced with **tourr**. However, non-manual tours can easily increase the embedded dimensionality. As such, $d > 2$ tours are under-supported in the layered composition. Additional functions could be added to facilitate geometric display to parallel coordinate plots, Chernoff faces, and pixel-based displays. These displays are potentially best used with data with relatively few observations and many variables.

In contrast, extending the output dimension of the manual tour would require an additional manipulation angle. Which would be rotated in a 4D manipulation space. Chapter 3 included the scaffolding for this task. Namely, this would require applying another angle of rotation with Rodrigues' rotation formula on the current 3D rotation matrix. Another display dimension may benefit the detection and understanding of the higher dimensional structure though the input of three angles may prove less natural and harder to capture.

The manual tour only controls the contribution to one variable at a time. This can become cumbersome and time-consuming as dimensionality increases. In addition to the manual tour controlling the contribution of a single variable, it may be insightful to change the contributions of several variables at once (manipulating a linear combination). Alternatively, a sort of dimension reduction tour, appending several manual tours that sequentially zero

the contributions of variables contributing less than some threshold, may prove to expedite analysis, especially for approaching a feature's intrinsic dimensionality.

Setting aside the manual tour, there are several extensions to the layered composition of tours in **spinifex**, such as extending the type of geometric display functions available, such as adding a text table of the basis, convex hulls, alpha hulls, and a ‘high-density region’ display where the bulk of the data is shown as density contour, while the outermost observations are displayed as points (Hyndman, 1996).

The radial tour’s current three segments (increase-decrease-increase of magnitude) takes a bit to unpack. It may be more approachable to directly relate the position of the slider to the magnitude of the manipulation variable. The display order of the manual tour could be changed to more of an *ink tank* display, where the first frame would contain zero variable contribution, and the last would have a complete contribution. The starting frame would be the original contribution that would also be exaggerated or annotated to reference against.

It may be interesting to experience tours as 3D scatterplots in extended reality with stereoscopically true head tracking may be fruitful. Nelson, Cook, and Cruz-Neira (1998) explore 2D tour is in virtual reality. Other works view 3D scatterplot tours on 2D displays (Yang, 1999, 2000). It would be interesting to see modern implementations using WebGL, Mozilla A-frame, or Unity. One concern would be keeping hardware and software as generalized as possible.

Besides changing the support of the experimental factors to the user study, it would be more interesting to try different tasks or compare other visualization techniques. The discussed user study crowdsourced participants with little exposure to linear projections. It would be interesting to compare the results from more familiar and experienced participants.

- Pedagogical side for newcomers.

The outlined cheem analysis can be generally applied models and local explanations. The package **cheem** currently calculates tree SHAP for tree-based models supported by **treeshap**. This could be generalized more broadly to other models and local explanations. Those

facilitated by `DALEX::explain()` seems to be an excellent direction to extend (Biecek, 2018; Biecek and Burzykowski, 2021). However, processing runtime would quickly become a formidable obstacle. Alternatively, other statistics may better show the structure identified by explanations.

Extending explanations and a user study

In **cheem**, we had focused primarily on continuous numeric predictors. Perhaps this analysis could be extended to image, text, or time series analysis. The presence of covariates in these cases may prove essential to having meaningful variable importance to test variable sensitivity of the structure of the explanation.

Chapter 7

Appendix

.1 Glossary

Table 1: *Glossary of*

Term	Notation	Alias
observation	$\$n\$$	instance, item, row
variable	$\$p\$$	feature, column (
data	$\$X_{n \times p}\$$	
basis	$\$A_{p \times d}\$$	
projection	$\$Y_{n \times d} = X_{n \times p} \cdot A_{p \times d}\$$	embedding
tour		
Intrinsic data dimensionality	$\$iid\$$	
Nonlinear model		black-box model

.2 Animated tour links

Table 2: *Animated tour links in this thesis*

Reference	Description	Link
Figure \@ref(fig:ch1fig3)	radial tour, penguins	vimeo.com/676723431
Figure \@ref(fig:ch2fig4)	grand tour, penguins	vimeo.com/676723441

Figure \@ref(fig:caselenguins)	cheem, penguins	vimeo.com/666431172
Figure \@ref(fig:casechocolates)	cheem, chocolates	vimeo.com/666431143
Figure \@ref(fig:casechocolatesinverse)	cheem, chocolates (inverse)	vimeo.com/666431148
Figure \@ref(fig:casefifa)	cheem, fifa	vimeo.com/666431163
Figure \@ref(fig:caseames)	cheem, ames housing 2018	vimeo.com/666431134

.3 Supplementary material

Table 3: *Supplementary material*

Context	Description	Link
spinifex	CRAN Link	CRAN.R-project.org/package=spinifex
spinifex	vignette, Getting started with spinifex	nspyrison.github.io/spinifex/articles/getting-started-with-spinifex.html
spinifex	vignette, Ggproto api	nspyrison.github.io/spinifex/articles/ggproto-api.html
cheem	CRAN Link	CRAN.R-project.org/package=cheem
cheem	vignette, Getting started with cheem	nspyrison.github.io/cheem/articles/getting-started-with-cheem.html
thesis	repository	github.com/nspyrison/thesis_ns
thesis	thesis, pdf format	github.com/nspyrison/thesis_ns/blob/master/docs/thesis_ns.pdf
thesis	thesis, html format	https://nspyrison.github.io/thesis_ns/

Bibliography

- Adadi, A and M Berrada (2018). Peeking inside the black-box: a survey on explainable artificial intelligence (XAI). *IEEE access* **6**, 52138–52160.
- Anscombe, FJ (1973). Graphs in Statistical Analysis. *The American Statistician* **27**(1), 17–21.
- 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.
- Arrieta, AB, N Díaz-Rodríguez, J Del Ser, A Bennetot, S Tabik, A Barbado, S García, S Gil-López, D Molina, and R Benjamins (2020). Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion* **58**, 82–115.
- Asimov, D (1985). The Grand Tour: a Tool for Viewing Multidimensional Data. *SIAM journal on scientific and statistical computing* **6**(1), 128–143.
- Barrow, M, J Chong, and N Spyrisson (2020). Changes to Victorian Intraday Electricity Demand Following COVID-19 Restrictions. en. In: *Melbourne Datathon 2020, Insights category*, pp.4. https://github.com/nspyrisson/melb_datathon2020/blob/master/_paper/paper.pdf (visited on 02/09/2022).
- Batch, A, A Cunningham, M Cordeil, N Elmquist, T Dwyer, BH Thomas, and K Marriott (2019). There is no spoon: Evaluating performance, space use, and presence with expert domain users in immersive analytics. *IEEE transactions on visualization and computer graphics* **26**(1), 536–546.
- Becker, RA and WS Cleveland (1987). Brushing Scatterplots. *Technometrics* **29**(2), 127–142.

BIBLIOGRAPHY

- Bertini, E, A Tatu, and D Keim (2011). Quality metrics in high-dimensional data visualization: An overview and systematization. *IEEE Transactions on Visualization and Computer Graphics* **17**(12), 2203–2212.
- Besse, P, C Castets-Renard, A Garivier, and JM Loubes (2019). Can Everyday AI be Ethical? Machine Learning Algorithm Fairness. *Machine Learning Algorithm Fairness (May 20, 2018)*. *Statistiques et Société* **6**(3).
- Biecek, P (2018). DALEX: explainers for complex predictive models in R. *The Journal of Machine Learning Research* **19**(1), 3245–3249.
- Biecek, P and T Burzykowski (2021). *Explanatory Model Analysis: Explore, Explain, and Examine Predictive Models*. en. CRC Press.
- Box, GE (1976). Science and statistics. *Journal of the American Statistical Association* **71**(356), 791–799.
- Breiman, L (2001). Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical science* **16**(3), 199–231.
- Brown, TA (2015). *Confirmatory factor analysis for applied research*. Guilford publications.
- 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>.
- Card, SK, TP Moran, and A Newell (1983). *The psychology of human-computer interaction*. Crc Press.
- Cattell, RB (1966). The scree test for the number of factors. *Multivariate behavioral research* **1**(2), 245–276.
- Chambers, J, W Cleveland, B Kleiner, and P Tukey (1983). Graphical Methods for Data Analysis. eng.
- Chang, W, J Cheng, J Allaire, C Sievert, B Schloerke, Y Xie, J Allen, J McPherson, A Dipert, and B Borges (2021). *shiny: Web Application Framework for R*. <https://CRAN.R-project.org/package=shiny>.
- 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.

BIBLIOGRAPHY

- Cook, D and A Buja (1997). Manual Controls for High-Dimensional Data Projections. *Journal of Computational and Graphical Statistics* **6**(4), 464–480.
- Cook, D, A Buja, J Cabrera, and C Hurley (1995). Grand Tour and Projection Pursuit. *Journal of Computational and Graphical Statistics* **4**(3), 155.
- Cook, D, A Buja, EK Lee, and H Wickham (2008). “Grand Tours, Projection Pursuit Guided Tours, and Manual Controls”. en. In: *Handbook of Data Visualization*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp.295–314. http://link.springer.com/10.1007/978-3-540-33037-0_13.
- Cook, D, U Laa, and G Valencia (2018). Dynamical projections for the visualization of PDFSense data. *Eur. Phys. J. C* **78**(9), 742.
- Dastin, J (2018). Amazon scraps secret AI recruiting tool that showed bias against women. en. *Reuters*.
- Díaz, M, I Johnson, A Lazar, AM Piper, and D Gergle (2018). Addressing age-related bias in sentiment analysis. In: *Proceedings of the 2018 chi conference on human factors in computing systems*, pp.1–14.
- Dimara, E and C Perin (2019). What is interaction for data visualization? *IEEE transactions on visualization and computer graphics* **26**(1), 119–129.
- Duffy, C (2019). Apple co-founder Steve Wozniak says Apple Card discriminated against his wife. *CNN*.
- Espadoto, M, RM Martins, A Kerren, NST Hirata, and AC Telea (2021). Toward a Quantitative Survey of Dimension Reduction Techniques. *IEEE Transactions on Visualization and Computer Graphics* **27**(3), 2153–2173.
- Evans, R and M Boreland (2017). Multivariate Data Analytics in PV Manufacturing—Four Case Studies Using Manufacturing Datasets. *IEEE Journal of Photovoltaics* **8**(1), 38–47.
- Feurer, M, A Klein, K Eggensperger, J Springenberg, M Blum, and F Hutter (2015). Efficient and robust automated machine learning. *Advances in neural information processing systems* **28**.
- Gabriel, KR (1971). The biplot graphic display of matrices with application to principal component analysis. *Biometrika* **58**(3), 453–467.

BIBLIOGRAPHY

- Gijsbers, P, F Pfisterer, JN van Rijn, B Bischl, and J Vanschoren (2021). Meta-learning for symbolic hyperparameter defaults. In: *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pp.151–152.
- Gorman, KB, TD Williams, and WR Fraser (2014). Ecological sexual dimorphism and environmental variability within a community of Antarctic penguins (genus Pygoscelis). *PloS one* **9**(3), e90081.
- Gosiewska, A and P Biecek (2019). IBreakDown: Uncertainty of model explanations for non-additive predictive models. *arXiv preprint arXiv:1903.11420*.
- 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.
- Greenwell, B (2020). *fastshap: Fast Approximate Shapley Values*. <https://CRAN.R-project.org/package=fastshap>.
- Grinstein, G, M Trutschl, and U Cvek (2002). High-Dimensional Visualizations. en, 14.
- Hevner, AR, ST March, J Park, and S Ram (2004). Design science in information systems research. *MIS quarterly*, 75–105.
- Horst, AM, AP Hill, and KB Gorman (2020). *palmerpenguins: Palmer Archipelago (Antarctica) penguin data*. <https://allisonhorst.github.io/palmerpenguins/>.
- Huber, W, VJ Carey, R Gentleman, S Anders, M Carlson, BS Carvalho, HC Bravo, S Davis, L Gatto, and T Girke (2015). Orchestrating high-throughput genomic analysis with Bioconductor. *Nature methods* **12**(2), 115–121.
- Hutter, F, L Kotthoff, and J Vanschoren (2019). *Automated machine learning: methods, systems, challenges*. Springer Nature.
- Hyndman, RJ (1996). Computing and graphing highest density regions. *The American Statistician* **50**(2), 120–126.
- Karwowski, W (2006). *International Encyclopedia of Ergonomics and Human Factors*, -3 Volume Set. CRC Press.
- Keim, DA (2000). Designing pixel-oriented visualization techniques: Theory and applications. *IEEE Transactions on visualization and computer graphics* **6**(1), 59–78.
- Kleinberg, J and E Tardos (2006). *Algorithm design*. Pearson Education India.

BIBLIOGRAPHY

- Kodiyan, AA (2019). An overview of ethical issues in using AI systems in hiring with a case study of Amazon's AI based hiring tool. *Researchgate Preprint*.
- Kominsarczyk, K, P Kozminski, S Maksymiuk, and P Biecek (2021). *treeshap*. <https://github.com/ModelOriented/treeshap>.
- Laa, U, D Cook, and G Valencia (2020). A slice tour for finding hollowness in high-dimensional data. *Journal of Computational and Graphical Statistics* **29**(3), 681–687.
- Larson, J, S Mattu, L Kirchner, and J Angwin (2016). How We Analyzed the COMPAS Recidivism Algorithm. en. *ProPublica*.
- Lee, S, D Cook, N da Silva, U Laa, N Spyris, E Wang, and HS Zhang (2021). The state-of-the-art on tours for dynamic visualization of high-dimensional data. en. *WIREs Computational Statistics*, e1573.
- Lee, S, U Laa, and D Cook (2020). Casting Multiple Shadows: High-Dimensional Interactive Data Visualisation with Tours and Embeddings. *arXiv preprint arXiv:2012.06077*.
- Leone, S (2020). *FIFA 20 complete player dataset*. en. <https://www.kaggle.com/stefanoleone992/fifa-20-complete-player-dataset>.
- Liaw, A and M Wiener (2002). Classification and regression by randomForest. *R news* **2**(3), 18–22.
- Liu, S, D Maljovec, B Wang, PT Bremer, and V Pascucci (2017). Visualizing High-Dimensional Data: Advances in the Past Decade. *IEEE Transactions on Visualization and Computer Graphics* **23**(3), 1249–1268.
- Lubischew, AA (1962). On the use of discriminant functions in taxonomy. *Biometrics*, 455–477.
- Lundberg, SM, GG Erion, and SI Lee (2018). Consistent individualized feature attribution for tree ensembles. *arXiv preprint arXiv:1802.03888*.
- Lundberg, SM and SI Lee (2017). A unified approach to interpreting model predictions. In: *Proceedings of the 31st international conference on neural information processing systems*, pp.4768–4777.
- Maaten, L van der, E Postma, and J Van den Herik (2009). Dimensionality Reduction: a Comparative Review. *J Mach Learn Res* **10**(66-71), 13.
- Marriott, K, F Schreiber, T Dwyer, K Klein, NH Riche, T Itoh, W Stuerzlinger, and BH Thomas (2018). *Immersive Analytics*. en. 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>.
- Molnar, C (2020). *Interpretable machine learning*. Lulu. com. christophm.github.io/interpretable-ml-book/.
- Nelson, L, D Cook, and C Cruz-Neira (1998). XGobi vs the C2: Results of an Experiment Comparing Data Visualization in a 3-D Immersive Virtual Reality Environment with a 2-D Workstation Display. en. *Computational Statistics* **14**(1), 39–52.
- Nonato, LG and M Aupetit (2018). Multidimensional projection for visual analytics: Linking techniques with distortions, tasks, and layout enrichment. *IEEE Transactions on Visualization and Computer Graphics* **25**(8), 2650–2673.
- O’Neil, C (2016). *Weapons of math destruction: How big data increases inequality and threatens democracy*. Crown.
- Ocagne, Md (1885). *Coordonne’ses paralle’les et axiales. Me’thode de transformation ge’ome’trique et proce’dé nouveau de calcul graphique de’duits de la considé’ration des coordonne’ses paralle’les*. French. Paris: Gauthier-Villars.
- Palan, S and C Schitter (2018). Prolific. A Subject Pool for Online Experiments. *Journal of Behavioral and Experimental Finance* **17**, 22–27.
- Panagiotelis, A (2020). *Manifold Learning*. Monash University.
- Pearson, K (1901). LIII. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* **2**(11), 559–572.
- Pedersen, TL and D Robinson (2020). *ggridge: A Grammar of Animated Graphics*. <https://CRAN.R-project.org/package=ggridge>.
- Pfisterer, F, JN van Rijn, P Probst, AC Müller, and B Bischl (2021). Learning multiple defaults for machine learning algorithms. In: *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pp.241–242.

- Probst, P, AL Boulesteix, and B Bischl (2019). Tunability: importance of hyperparameters of machine learning algorithms. *The Journal of Machine Learning Research* **20**(1), 1934–1965.
- Probst, P, MN Wright, and AL Boulesteix (2019). Hyperparameters and tuning strategies for random forest. *Wiley Interdisciplinary Reviews: data mining and knowledge discovery* **9**(3), e1301.
- Ribeiro, MT, S Singh, and C Guestrin (2016). "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '16. New York, NY, USA: Association for Computing Machinery, pp.1135–1144. <https://doi.org/10.1145/2939672.2939778>.
- Roberts, JC (2007). State of the art: Coordinated & multiple views in exploratory visualization. In: *Fifth international conference on coordinated and multiple views in exploratory visualization (CMV 2007)*. IEEE, pp.61–71.
- Rodrigues, O (1840). Des lois géométriques qui régissent les déplacements d'un système solide dans l'espace: et de la variation des coordonnées provenant de ces déplacements considérés indépendamment des causes qui peuvent les produire. *Journal de Mathématiques Pures et Appliquées* **5**, 380–440.
- Scrucca, L, M Fop, TB Murphy, and AE Raftery (2016). mclust 5: Clustering, Classification and Density Estimation Using Gaussian Finite Mixture Models. *The R journal* **8**(1), 289–317.
- 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.
- Shapley, LS (1953). *A value for n-person games*. Princeton University Press.
- Shmueli, G (2010). To explain or to predict? *Statistical science* **25**(3), 289–310.
- Shrikumar, A, P Greenside, and A Kundaje (2017). Learning important features through propagating activation differences. In: *International Conference on Machine Learning*. PMLR, pp.3145–3153.

BIBLIOGRAPHY

- Shrikumar, A, P Greenside, A Shcherbina, and A Kundaje (2016). Not just a black box: Learning important features through propagating activation differences. *arXiv preprint arXiv:1605.01713*.
- Sievert, C (2020). *Interactive Web-Based Data Visualization with R, plotly, and shiny*. Chapman and Hall/CRC. <https://plotly-r.com>.
- Simonyan, K, A Vedaldi, and A Zisserman (2014). Deep inside convolutional networks: Visualising image classification models and saliency maps. In: *In Workshop at International Conference on Learning Representations*. Citeseer.
- Snyder, JP (1987). *Map projections–A working manual*. Vol. 1395. US Government Printing Office.
- Spryison, N and D Cook (2020). spinifex: an R Package for Creating a Manual Tour of Low-dimensional Projections of Multivariate Data. en. *The R Journal* **12**(1), 243.
- Spryison, N, B Lee, and L Besançon (2021). "Is IEEE VIS *that* good?" On key factors in the initial assessment of manuscript and venue quality. *OSF preprints*.
- Strumbelj, E and I Kononenko (2010). An efficient explanation of individual classifications using game theory. *The Journal of Machine Learning Research* **11**, 1–18.
- 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.
- Tukey, JW (1977). *Exploratory data analysis*. Vol. 32. Pearson.
- Unwin, A and P Valero-Mora (2018). Ensemble Graphics. *Journal of Computational and Graphical Statistics* **27**(1), 157–165.
- Vanni, L, M Ducoffe, C Aguilar, F Precioso, and D Mayaffre (2018). Textual Deconvolution Saliency (TDS): a deep tool box for linguistic analysis. In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp.548–557.
- Venna, J and S Kaski (2006). Visualizing gene interaction graphs with local multidimensional scaling. In: *ESANN*. Vol. 6. Citeseer, pp.557–562.
- Wagner Filho, J, M Rey, C Freitas, and L Nedel (2018). Immersive Visualization of Abstract Information: An Evaluation on Dimensionally-Reduced Data Scatterplots. In:

BIBLIOGRAPHY

- Wang, BT, TJ Hobbs, S Doyle, J Gao, TJ Hou, PM Nadolsky, and FI Olness (2018). Mapping the sensitivity of hadronic experiments to nucleon structure. *Physical Review D* **98**(9), 094030.
- Wickham, H and G Grolemund (2017). *R for Data Science*. O'Reily Media. <https://r4ds.had.co.nz/index.html>.
- Wickham, H (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. <https://ggplot2.tidyverse.org>.
- 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.
- 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).
- wikipedia (2021). *List of map projections*. en. https://en.wikipedia.org/w/index.php?title=List_of_map_projections&oldid=1058962748 (visited on 02/10/2022).
- Winer, BJ (1962). Statistical principles in experimental design.
- Xie, Y (2016). *bookdown: Authoring Books and Technical Documents with R Markdown*. Boca Raton, Florida: Chapman and Hall/CRC. <https://github.com/rstudio/bookdown>.
- Yang, L (1999). 3D Grand Tour for Multidimensional Data and Clusters. en. In: *Advances in Intelligent Data Analysis*. Ed. by DJ Hand, JN Kok, and MR Berthold. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, pp.173–184.
- Yang, L (2000). Interactive exploration of very large relational datasets through 3D dynamic projections. en. In: *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '00*. Boston, Massachusetts, United States: ACM Press, pp.236–243. <http://portal.acm.org/citation.cfm?doid=347090.347134>.
- Yao, Q, M Wang, Y Chen, W Dai, YF Li, WW Tu, Q Yang, and Y Yu (2019). Taking Human out of Learning Applications: A Survey on Automated Machine Learning. en. *arXiv:1810.13306 [cs, stat]*. (Visited on 02/11/2022).