



MONASH University

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

Nicholas Spyris

B.Sc. Statistics, Iowa State University

A thesis submitted for the degree of Doctor of Philosophy at

Monash University in January 2022

Department of Human-Centred Computing

January 2022

Contents

Copyright notice	v
Abstract	vii
Declaration	1
Publications as part of this thesis	1
Publications and papers during candidature not part of this thesis	1
Acknowledgments	5
Preface	7
1 Introduction	9
1.1 Research questions	13
1.2 Methodology	14
1.3 Contributions	15
1.4 Thesis structure	16
2 Background	19
2.1 Scatterplot matrices	20
2.2 Parallel coordinate plots	21
2.3 Dimension reduction	22
2.4 Tours, animated linear projections	24
3 spinifex: an R package for creating user-controlled animated linear projections	27
3.1 Introduction	28
3.2 Algorithm	28
3.3 Oblique cursor movement	34
3.4 Package structure	35
3.5 Use cases	37
3.6 Discussion	43
3.7 Acknowledgments	44
4 A Study on the Benefit of a User-Controlled Radial Tour for Variable Importance for Structure in High-Dimensional Data	45
4.1 Introduction	46

CONTENTS

4.2 Background	48
4.3 User study	52
4.4 Results	61
4.5 Conclusion	66
4.6 Accompanying tool: radial tour application	66
4.7 Acknowledgments	67
4.8 References	67
4.9 Appendix	67
5 Exploring Local Explanations of Non-linear Models Using Animated Linear Projections	75
5.1 Introduction	76
5.2 Local explanation statistics	78
5.3 Tours and the radial tour	81
5.4 The cheem viewer	81
5.5 Case studies	88
5.6 Discussion	99
5.7 Acknowledgments	99
5.8 References	99
6 Conclusion	101
6.1 Contributions	101
6.2 Summary statement	102
6.3 Future work	103
Bibliography	105

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 crucial to exploratory data analysis, checking model assumptions, and validating model performance. Yet visualization quickly becomes difficult as the dimensionality of the data or features increases. Traditionally, static, low-dimensional linear embeddings are used to mitigate this complexity. Such embedded space is a lossy approximation of the full space. However, viewing many embeddings with interactive supporting views improves the information conveyed. Data visualization *tours* are a class of dynamic linear projections that animate many linear projections over small changes to the projection basis.

Tours are categorized by the path of their bases. *Manual tours* allow for User-Controlled Steering (UCS) of bases path, where the contributions of individual variables can be controlled. I create a free, open-source **R** package, **spinifex**, that facilitates creating such tours. These manipulations of the variables can be precomputed or made in real-time. Display composition and exportable formats are interoperable with the existing package **tourr** and can be rendered with recent graphics interfaces **plotly** and **ganimate**. The former offers interactive use with HTML widgets, while the latter can render to static formats such as .gif or .mp4 files. An interactive application is included to illustrate the use of the manual tour.

Theoretically, the UCS of the manual tour should enable an analyst to better explore the variable attribution to the structure identified in an embedding. I find strong evidence to support that radial tour leads to a significant and large increase in accuracy as compared with Principal Component Analysis (PCA) and an alternative tour variant, the grand tour. I conduct a within-participant, crowd-sourced user study comparing these visualization

methods. Each of the $n = 108$ participants performs two trials with each visual for 648 total trials. The task is a supervised classification problem asking participants which variables attribute to the separation of 2 clusters.

Modern modeling techniques are sometimes referred to as black-box models due to the uninterpretable nature of model terms which are often many and non-linear. 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 at one point in the data. I build an **cheem**, an **R** package to streamline model and local explanation preprocessing, and two novel visuals. First the global view pits the data- and explanation-spaces side-by-side with a residual plot. Interactive features such as linked bushing and tooltips aid in identifying observations to compare. The local explanation from the selected observation becomes a basis used in creating the radial tour. This allows an analyst to evaluate the explanation and explore the sensitivity of the explanation to changes in the variable contribution.

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.

Publications as part of this thesis

Three paper are included in this thesis, one of which has been accepted for publication in *The R Journal* (Spyrison and Cook, 2020). This paper clarifies some of the theoretical work for user control offered by the manual tour and induces an open-source **R** package facilitating manual tours. The user study described in Chapter 4 will shortly be submitted to the *IEEE Transactions on Visualization and Computer Graphics*. This study finds large evidence for a sizable increase in a variable attribution task from the user control of the manual tour over principal component analysis and the grand tour. Then we extend the interpretability of complex non-linear models. I purpose exploring the support of local explanation of a model with the manual tour. This research covered in Chapter 5 will shortly be submitted to *The Journal of Machine Learning research*.

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:

CONTENTS

- *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.
- “*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.

I have renumbered and updated sections of submitted or published papers to generate a consistent presentation within the thesis. Specifically, the code used and primary function table have been redone to highlight the new ggproto API. This approach is more flexible, allowing for a layered composition of tours that will feel familiar to **ggplot2** users.

Student name: Nicholas Spyrison

Student signature:



Date: 2021-12-12

CONTENTS

The undersigned hereby certify that the above declaration correctly reflects the nature and extent of the student's co-authors' contribution to this work. In instances where I am not the responsible author I have consulted with the responsible author to agree on the respective contributions of the authors.

Main Supervisor name: Kimbal Marriott

Main Supervisor 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 pragmatic 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.

A special thanks to Professor Przemyslaw Biecek for giving his input in the formulation stages of my project and his easy-to-understand explanations of complex modeling aspects. Thank you for responding to cold emails and being available for collaboration. Thank you to Julie Holden for helping proofread and tighten up my writing.

I thank my fellow Ph.D students, lab members, and Pomodoro partners, especially on the occasional 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. I am looking forward to seeing you all in person shortly.

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. An online and .pdf versions of the thesis is available at nspyrison.github.io/thesis_ns/ and github.com/nspyrison/thesis_ns/blob/master/docs/thesis_ns.pdf, respectively.

I recognize that terminology is often overburdened by ambiguous use, with several changing meanings coming from different fields. My education background comes from Statistics, and I will default to terms from statistics and geometry.

Chapter 1

Introduction

Exploratory Data Analysis (EDA) is the process of the initial summarization and visualization of a dataset. This is a critical first step of checking for realistic values, finding improper data formats, and revealing insights (Tukey, 1977). Early and frequent data visualization is key to the data analyst's workflow (Wickham and Grolemund, 2017).

As modern datasets grow in complexity, the use of multivariate data becomes more ubiquitous. As the number of variables in the data increases, so to does the difficulty in conveying the information they contain. The number of observations in the data matters though overly dense points can be alleviated by decreasing opacity of points or changing the

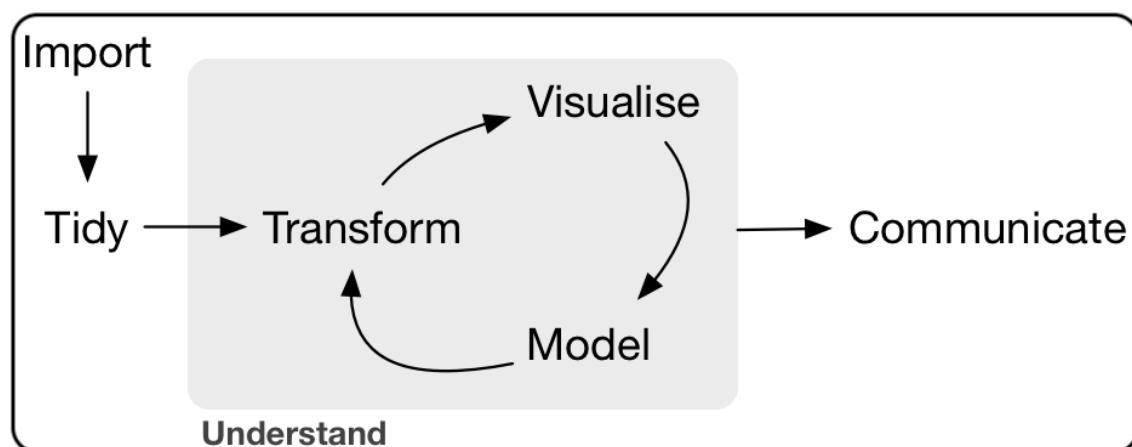


Figure 1.1: Data analysis workflow (Wickham and Grolemund, 2017). This work focuses primarily on multivariate data visualization, and also facilitates transforming data and interpretation of models.

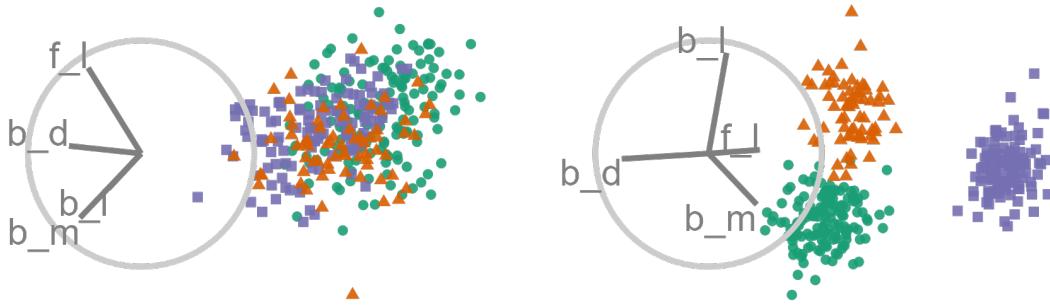


Figure 1.2: Palmer Penguin data (Gorman, Williams, and Fraser, 2014; Horst, Hill, and Gorman, 2020), three species of penguins measured across four physical variables. These are two projections of 4D variable space. The basis orientation, is illustrated with a biplot. Separation of the species clusters is the structure of interest. Some orientations of the data do not show identify cluster separation while others will.

geometric display to an aggregated heatmap or 2D density contours. This thesis hings on the visualization of multivariate data, central to this iterated workflow. The titanic dataset is a common starting dataset for machine learning and contains nine variables. Sports and house sale datasets are also common and contain dozens of variables. It is important to extend the use of data visualization even with the added complexity of many variables.

Linear projections use a linear combinations of variables to define a new component. The components can contain signal from multiple variables at once, make a good step at extending the visuals as dimensionality increases. These component linear combinations can be thought of as a reorientation of the origin basis. One example of linear projections is that of Principal Component Analysis (PCA, Pearson, 1901) which creates a component space ordered by descending amount of variation. It uses eigenvalue decomposition to identify the basis reorientation. These components are typically viewed as discrete orthogonal pairs. Where the data space is commonly approximated in fewer components. The basis of linear projections is common illustrated with a biplot (Gabriel, 1971). The biplot expresses the magnitude and angle each variable contributes to the resulting display dimensions.

A data visualization *tour* (Cook et al., 2008; Lee et al., 2021) is a class of linear projections that animate small changes to the projection basis. Intuitively, they show a projection of the data object down to lower-dimensional space in the same way that a 3D object

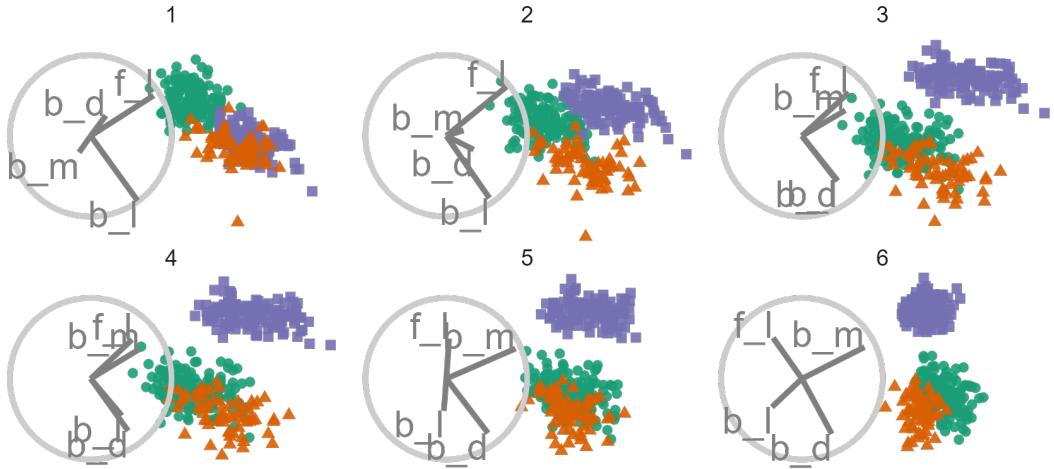


Figure 1.3: *Frames of a grand tour of Palmer Penguin data. Tours animates linear projections over small changes in the basis. In a grand tour target frames are selected randomly.*

casts a 2D shadow. A key feature of the tour is the permanence and tractability of the points through many frames. In the shadow analogy, an object such as a barstool, may cast a circular shadow that could come from any number of shapes. However, if the stool was rotated, its legs would show in the shadow quickly giving an intuitive interpretation of the object. Similarly, looking at a data object over the rotation of its variables yields information about its structure. The permanence of observations between frames aided interaction such as linked brushing should plausibly convey more information than static orthogonal views. Chapter 2 walks through some of the previous alternatives, and why we ultimately continue with this class of animated linear dimension reduction.

Component spaces and the grand tour optimize for specific objectives and selected randomly. None of the previous methods allow an analyst to control the contributions of variables to the basis. This is a crucial component for *human-in-the-loop* analysis (Karwowski, 2006). If the analyst wants to explore what would happen to the structure if one variable were removed or the contribution of another was increased, they had no way to do so. Cook and Buja (1997) introduces the *manual tour* offering means for user control over a basis. By selecting a variable and initializing an additional manipulation dimension to an embedding

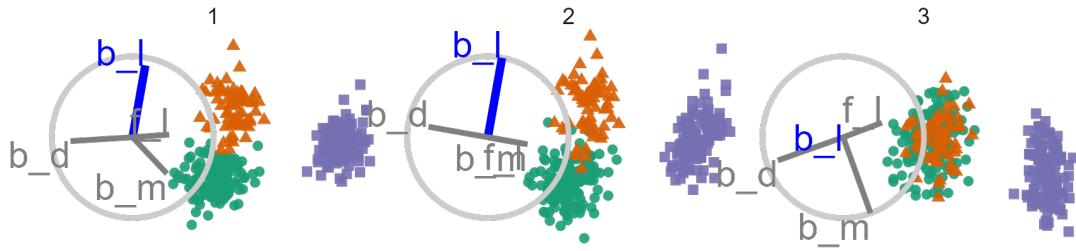


Figure 1.4: Frames of a radial, manual tour of Palmer Penguin data manipulating the contribution of bill length (b_l). When bill length has a large contribution to the frame the orange and green species clusters are separated. When its contribution is small the clusters overlap. Because of this we say that bill length is sensitive to the separation of these two variables.

the contribution of the variable can be controlled. The manual tour can control the angle and magnitude that the variable contributes though the latter is generally more meaningful angular manipulations are effectively rotating the projection which changes the relative position but the not fundamentally what information is shown. Because of this distinction this work mostly focuses on the a specific manual tour, the *radial tour* where the angle of the manipulation variable is fixed and the magnitude is varies the radius in the original direction.

There are many things features that an analyst may be looking for or come across in the course of an analysis. This thesis will regularly be looking for linear components that identify spindles or separation of clusters. Both of these distinguish some subset of observations from the rest. More generally the type of data and hypothesis will likely influence the exact feature of interest. Which variables contribute to the structure of a feature is important to know and communicate to others. The manual tour can vary the contribution of a variable. If removing the contribution of a variable results in the feature no longer being resolved we say that the variable is sensitive to the structure of the feature identified.

1.1 Research questions

The over-arching question of interest can be stated as:

Can the radial tour with user interaction help analysts understand linear projections, and explore the sensitivity of structure in the projection to the variables contributing to a projection?

Understanding variable sensitivity to the structure in a projection frame is a crucial factor analysis after identifying a feature of interest. If an identified feature is the object of an analysis, then the variable attribution is the location vantage point for that feature. The user interaction afforded by the radial tour should allow for a more precise exploration of this structure by testing variable sensitivity to that structure.

We have the theoretical work for the manual tour. However, we lack a publicly available implementation, implementation notes, and have no evaluation of its performance over alternatives.

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

Neither component spaces nor the grand tour provide a means for changing a desired variable's contributions. To facilitate variable interaction, we need to have a means of manipulating the basis. This is crucial to explore the sensitivity of the variable contributions to the structure in a frame.

PCA is a common, but static method to approximate multivariate spaces. The grand tour introduced the idea of viewing spaces animated over small changes to the basis. The manual tour gives an analyst the means to steer the basis to test the sensitivity of variables. However, it remains unclear which methods lead to the most perceptible way to visualize multivariate spaces.

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

How can we be sure that having a means of control leads to a meaningfully better analysis?

Comparing alternative methods should lend insight into which methods perform well under specific tasks.

Complex non-linear models are also being applied with more frequency to predict or classify with many predictors. While these models lead to increased accuracy over linear models, they have a loss of the interpretability of their variables. One aspect of eXplainable Artificial Intelligence (XAI) tries to preserve the interpretability of such models through the use of local explanations. These explanations are essentially linear variable importance in the vicinity of one observation in a model. That is, which variables help the model explain the difference between the intercept and this observation's prediction. The user control from the radial tour may allow an analyst better understand or even explore 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?

The tension from the trade-off between accuracy and interpretability of black-box models is rising. There is a clear need to be able to explain black-box models. After extrapolating local explanations of all observations, we want to scrutinize the explanation of one observation by changing the variable-contribution with the radial tour.

1.2 Methodology

The research corresponding with RQ 1 entails *algorithm/software design* adapting the algorithm from Cook and Buja (1997). This allows for interactive control of 2D projections and serves as a foundation for the remaining work.

To address RQ 2, a controlled *experimental study* has explored the efficacy of interactive radial tours as compared with two benchmark methods: PCA and the grand tour (Asimov, 1985).

The research for RQ #3 involves *design science*. A local explanation is an approximation of the linear variable importance in the vicinity of one observation. Our design must accommodate the selection of observations to explore and the then performing radial tours on the

This SHAP value will also serve as the 1D basis for the radial tour. While using SHAP as a projection basis is novel, it is not particularly insightful, we use the radial tour to change variable contributions. We also provide linkage information, the within-class distributions of the SHAP components as parallel coordinate marks on the basis. We also offer a global view and related statistics to map to color to aid in exploring the sensitivity of the SHAP-space relative to the sensitivity of the original data space.

1.3 Contributions

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

1.3.1 Software

- **spinifex**, an **R** package offering a consistent framework for performing radial tours and the rendering of any tours to various formats:
 - Perform radial tours to explore the variable sensitivity to structure
 - Transform of numeric variable in the data
 - Extract various bases exposing features of the data
 - Interoperable with tours generated with **tour** (Wickham et al., 2011)
 - Layered interface for producing tours that mirror the layered approach of **ggplot2** (Wickham, 2016)
- **cheem**, an **R** package applying the radial tour to local explanations of black-box models, including:
 - Preprocessing; creation of a random forest model, the extraction of all observation's tree SHAP local explanation, and extraction of other statistics for display
 - Visualization of approximations of the data- and attribution-space side-by-side with linked brushing, hover tooltips, and tabular display of selected points

1.3.2 Knowledge

- A user study comparing the efficacy of the radial tour against the alternatives of PCA the grand tour includes:
 - Creation of supervised classification task to assess the variable attribution to the separation of clusters performed under various levels of 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 while the task is performed fastest with the grand tour
 - Attribution of the error by performing a mixed model regression helps to explain the source of the error term into the variability of participants skill aptitude for this task and variability of the difficultly of a simulation by random chance
 - Introduces an interactive application to preprocess data and explore. Users can choose from six supplied datasets or upload their own
- Cheem analysis, given a non-linear model and local explanations of all observations I purpose the exploring the support of local explanation by varying the contributions with the radial, manual tour.
 - Using a global view with approximation of variable space, attribution space, and model information side-by-side with linked brushing identify a primary observation of interest and optional comparison observation
 - Normalize the variable attribution from the selected observation and use this as a projection basis
 - Explore and evaluate the local explanation; by use of the radial tour we can explore the sensitivity to the structure identified checking the support the explanation holds realistic or completely unreliable

1.4 Thesis structure

The remainder of the thesis is organized as follows: 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. There is ample evidence that using the radial tour increases the accuracy of responses for the supervised variable-attribution task describing the separation between two variables. Chapter 5 extends the use of manual tours to improve the interpretability of non-linear models, where manual tours explore variable sensitivity to the structure identified in local explanations of the model. Lastly, Chapter 6 concludes with some takeaways and a discussion of possible extensions.

Chapter 2

Background

This section starts with setting the scope for the type of data we are focusing on, gives a brief motivation then works through previous multivariate visualizations and their scalability. By the end, the focus narrows to linear projections, and in particular, the class of animated linear projections known as the tour.

For our purposes I will be focusing on the case where data $X_{n \times p}$ contains n observations of p variables, is complete with no missing values, variables are numeric (ideally not ordinal levels), and $n > p$ typically many more observations than variables. While I write as though always operating on the original variable space, these methods could similarly be applied to component spaces or feature decompositions of data not fitting this format. In the case of the linear projection let, $Y_{n \times d} = X_{n \times p} \cdot A_{p \times d}$ be the embedding of the data mapped by the basis A , where $d < p$. When p is large, say over 10 or 20 variables, the viewing space is quite large. In these cases, a PCA initialization step is commonly used where the variables are approximated as fewer principal components, and this reduced space can be viewed, albeit with the disadvantage of having another linear mapping back to the original space. Wickham, Cook, and Hofmann (2015) also view model spaces and features while urging to prefer visualizing data-space directly.

Visualization is much more robust than numerical summarization alone (Anscombe, 1973; Matejka and Fitzmaurice, 2017). In these studies, data sets have the same summary statistics yet contain obvious visual trends and shapes that could go completely unheeded

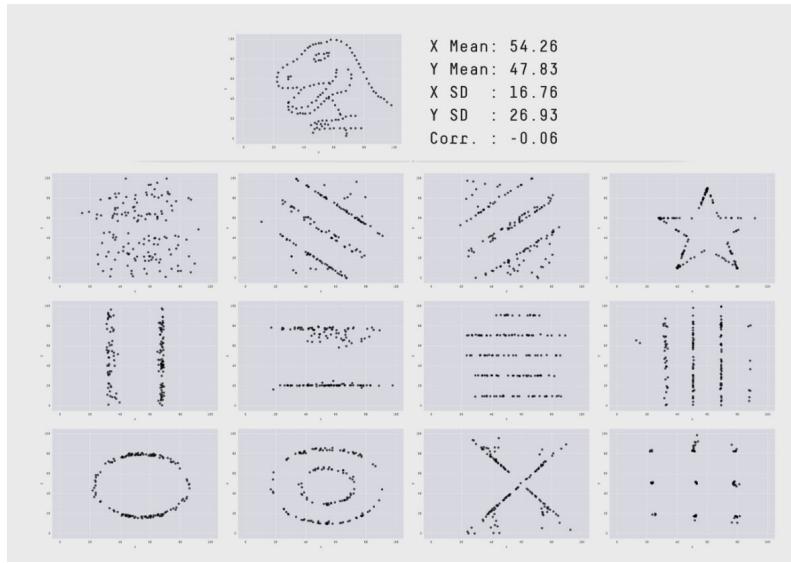


Figure 2.1: (*ref:ch2fig2-cap*)

if plotting is foregone. Data visualization is fundamental to EDA and quickly evaluating data supports ensuring that models are suitable.

2.1 Scatterplot matrices

The work in Grinstein, Trutschl, and Cvek (2002) gives a good taxonomy of high-dimensional visualization. We will follow a few examples conclude with tour methods. Broadly speaking, we are concerned with the question “How can an analyst visualize arbitrary p – dimensions?”. To illustrate some of the options for data I use the penguins data (Gorman, Williams, and Fraser, 2014; Horst, Hill, and Gorman, 2020). It contains 333 observations of four physical measurements across three species of penguins observed near Palm Station, Antarctica.

Viewing as many univariate histograms or density curves is one method. Similarly, one could look at all variable pairing as scatter plots. This forms the crux of the scatterplot matrices, also known as SPLOM (Chambers et al., 1983). In a scatterplot matrix, variables are displayed across the columns and rows. The diagonal elements show the univariate densities while off-diagonal positions show scatterplot pairs. This is useful for getting a handle on the support of the variables but is not going to scale well with dimension and is not a suitable audience-ready display. It is very busy and doesn’t draw attention to any

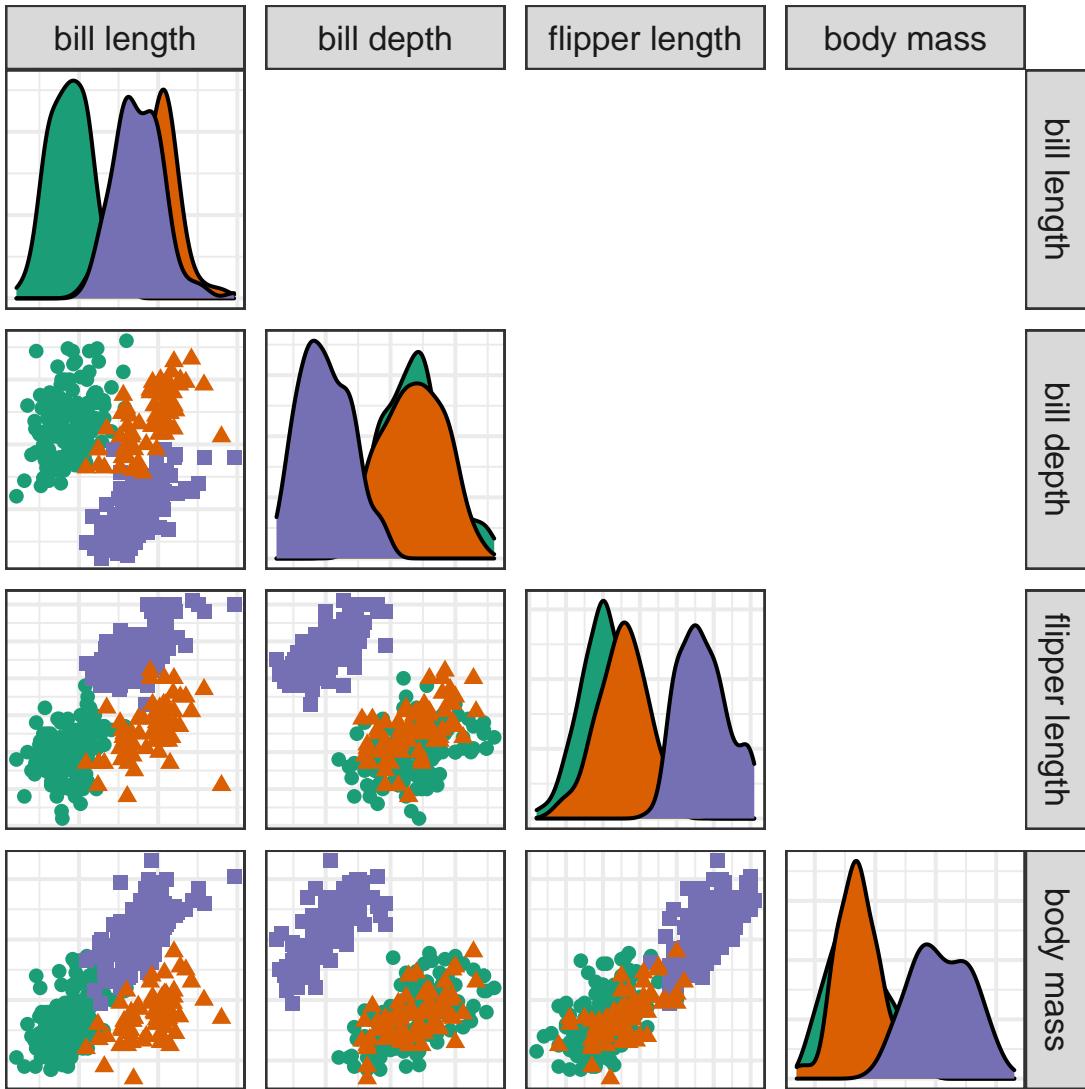


Figure 2.2: Scatterplot matrix of penguins data. This is a good initial step, but will not scale well at p increases.

one spot. Munzner (2014) reminds us to abstract all of the cognitive work out of the visual, allowing the audience to focus on seeing the evidence supporting the claim.

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 with the height mapping to the quantile or z-value for each variable. This scales much better with dimensions but poorly with observations. It also suffers from an asymmetry

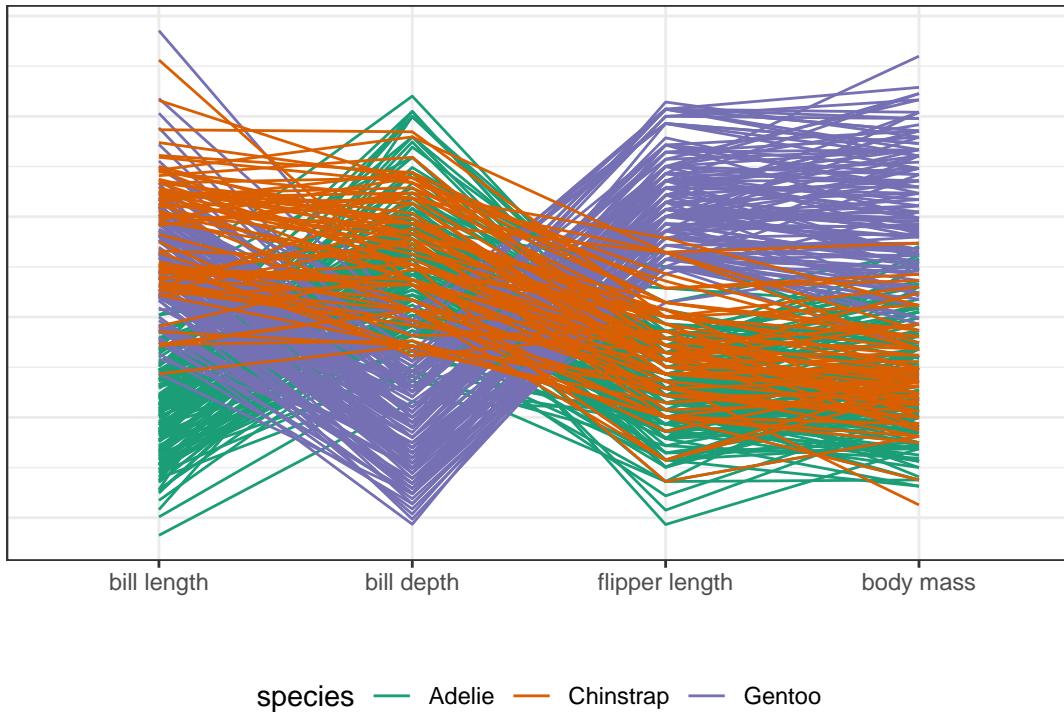


Figure 2.3: Parallel coordinate plots of penguins data. This does not scale well with observations, suffers from asymmetry with the variable ordering, and horizontal position is used for variables rather than observation levels.

with the variable order. That is, changing the order of the variable 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 like to reserve it for the values of the observations. The same issues persist across other observation-based displays such as radial variants, pixel-oriented visuals, and Chernoff faces (Keim, 2000; Chernoff, 1973). These visuals are better suited for the $n < p$ case with more variables than observations.

2.3 Dimension reduction

Ultimately, we will need to turn to dimension reduction to create a compelling visual allowing audiences to focus on features with contributions from multiple variables. Dimension reduction is separated into two categories, linear and non-linear. The linear case spans all affine mathematical transformations, essentially any mapping where parallel lines

stay parallel. Non-linear transformations are the complement of the linear case, think transformations containing exponents or interacting terms. Examples in low dimensions are relatable. For instance, shadows are examples of linear projections where a 3-dimensional object casts a 2D projection, the shadow. Our vision at any one instance of time, or a picture, is also a 2D projection. An example of a non-linear transformation is that of 2D representation of the globe. There are many different ways (and features to optimize) to distort the surface to display as a map. The most common may be rectangular displays where the area is distorted more and more with the distance away from the equator. Other distortions are created when the surface is unwrapped as a long ellipse. Yet others create non-continuous gaps in oceans to minimize the distortion of countries.

Non-linear techniques often have hyperparameters that affect how the spaces are distorted to fit into fewer dimensions. To quote Anastasios Panagiotelis, “All non-linear projections are wrong, but some are useful”, a play on George Box’s quote about models. Non-linear techniques distort the space in unclear ways, and what is more, they can introduce features not in the data depending on the selection of hyperparameters. The presence of structure in a non-linear model is necessary but not sufficient to conclude the existence of a structure in the data.

Unfortunately, there is no free lunch here. An increase in the original data dimensions will lead to a $p - d$ -dimensional viewing space in the linear case and an increasingly perturbed and distorted space in non-linear techniques. 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 important aspect of dimension reduction that does not have to end in visual, but is also a common 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 variables, while the theory would suggest the intrinsic dimensionality is five. The data likely picks up on other aspects and may be better summarized in with eight or ten dimensions. If this were the case, reducing the data to this space would be necessary to gate the exponentially increasing view time.

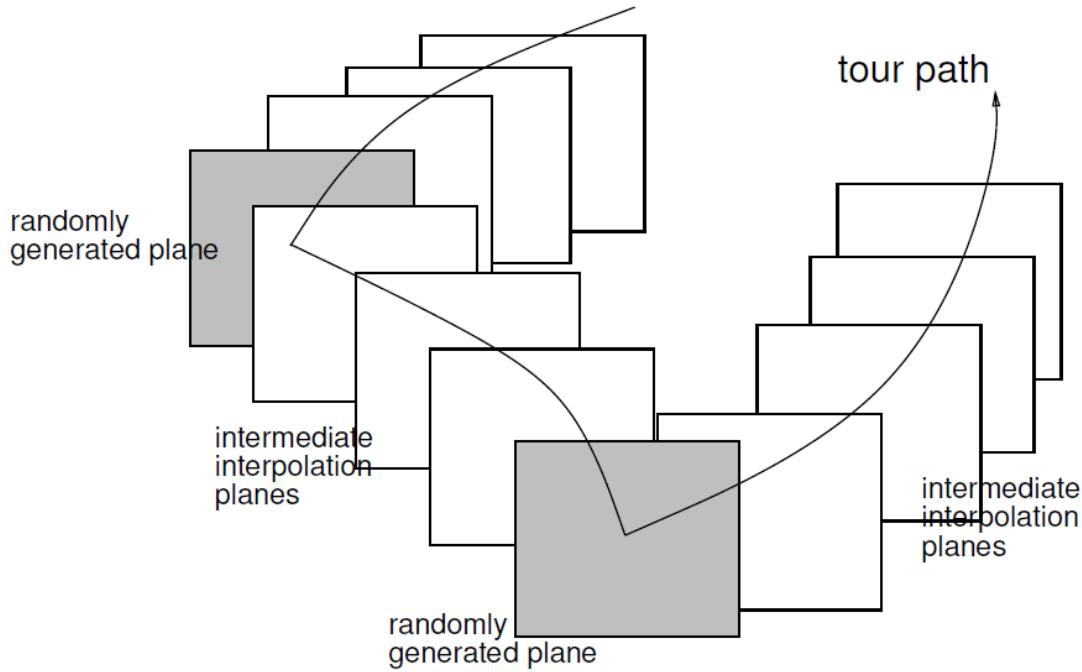


Figure 2.4: Illustration of the grand tour selecting random target frames (grey) connected via geodesically interpolated frame (white). Figure from Buja et al. (2005).

2.4 Tours, animated linear projections

A single linear projection is a resulting space from the data multiplied by the basis. The basis is an orthonormal matrix (the orientation of the unit origin) mapping the data space to a lower dimension. A data visualization tour animates many such projections through small changes in the basis. In the bar stool shadow analogy, structural information about a hidden object was 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 rotation of the data). Originally in a *grand tour* (Asimov, 1985) several target frames are randomly selected and then interpolated along their geodesic path.

There are various types of tours which are classified by the generation of their basis paths. A *guided* tour uses simulated annealing to move progressively closer to an objective function in the embedded space (Hurley and Buja, 1990). A more comprehensive discussion and review of tours can be found in the works of Cook et al. (2008) and Lee et al. (2021).

Tours are used for a couple of salient features: maintains transparency back to the original variable space, and the persistence data points from frame to frame convey more information

than looking at discrete jumps to other bases. Because of these features, tours are a good method to extend the visualization of data space as dimensionality increases.

The work below covers manual tours (Cook and Buja, 1997) in Chapter 3. Radial tours are one type of manual tour where the contribution of one variable is extended radially to a full contribution, removed completely, then restored to its original contribution. Chapter 4 compares the efficacy of the radial tour as compared with PCA and the grand tour in a user study. Lastly, Chapter 5 extends the use of the radial tour to evaluate the local explanation of black-box models.

Chapter 3

spinifex: an R package for creating user-controlled 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**.

3.1 Introduction

The paper is organized as follows. Section 3.2 describes the algorithm used to perform a radial manual tour implemented in the package **spinifex**. Section 3.4 discussed the functions, their usage composing tours with **ggplot2** (Wickham, 2016) and their animation by **plotly** (Sievert, 2020) or **ganimate** (Pedersen and Robinson, 2020). Package functionality and code usage following the order applied in the algorithm follows in section 3.4.2. Section 3.5 illustrates how this can be used for sensitivity analysis applied to multivariate data collected on high-energy physics experiments (Wang et al., 2018). Section 3.6 summarizes this paper and discusses potential future directions.

3.2 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 a fixed angle. 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.

3.2.1 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{M} is a $p \times 3$ matrix, defining the 3D subspace where data rotation occurs and is called the manip(ulation) space.
- \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.

3.2.2 Steps

Step 0) Setup

The flea data (Lubischew (1962)), available in the **tourr** package, 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 can be used as a visual guide of the magnitude and direction that each variable contributes

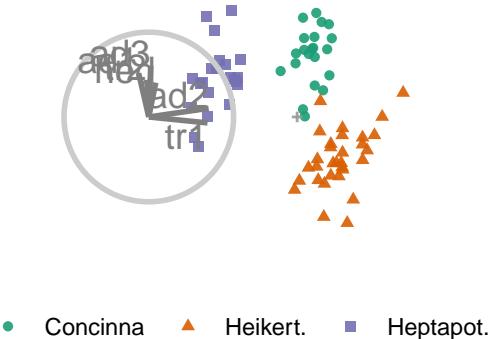


Figure 3.1: Biplot of the initial 2D projection: representation of the basis (left) and resulting data projection (right) of standardized flea data. The color and shape of data points are mapped to the species of flea beetle. The basis was produced by a projection pursuit guided tour with the holes index. 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.

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} .

$$e_k \leftarrow 1$$

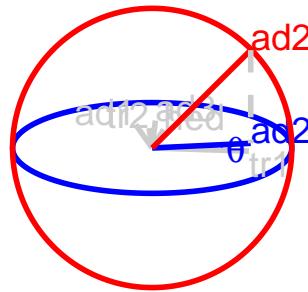


Figure 3.2: Illustration of a 3D manip space, the projection plane is shown as a blue circle extending into and out of the display. A manipulation direction is initialized, the red circle, orthogonal to the projection plane. This allows the selected variable, *aede2*, to change its contribution back 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.

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

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

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 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)):

$$\begin{aligned}\mathbf{R}_{[3, 3]} &= \mathbf{I}_3 + s_\phi \mathbf{U} + (1 - c_\phi) \mathbf{U}^2 \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 & 0 & c_\theta s_\phi \\ 0 & 0 & s_\theta s_\phi \\ -c_\theta s_\phi & -s_\theta s_\phi & 0 \end{bmatrix} + \begin{bmatrix} -c_\theta(1 - c_\phi) & s_\theta^2(1 - c_\phi) & 0 \\ -c_\theta s_\theta(1 - c_\phi) & -s_\theta^2(1 - c_\phi) & 0 \\ 0 & 0 & c_\phi - 1 \end{bmatrix} \\ &= \begin{bmatrix} c_\theta^2 c_\phi + s_\theta^2 & -c_\theta s_\theta(1 - c_\phi) & -c_\theta s_\phi \\ -c_\theta s_\theta(1 - c_\phi) & s_\theta^2 c_\phi + c_\theta^2 & -s_\theta s_\phi \\ c_\theta s_\phi & s_\theta s_\phi & c_\phi \end{bmatrix}\end{aligned}$$

where

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

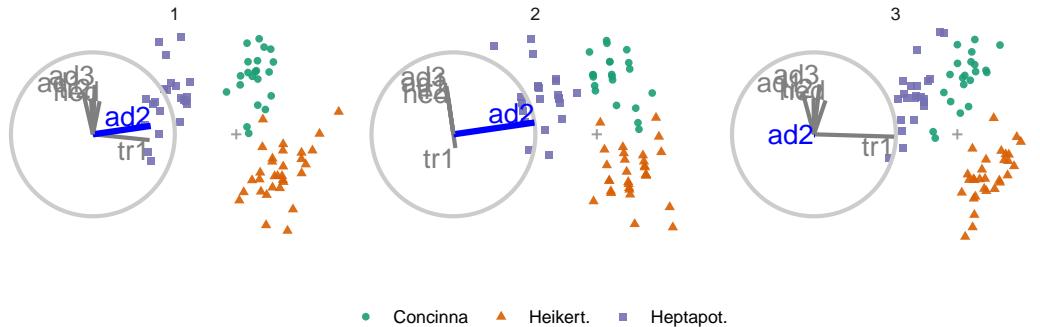


Figure 3.3: Select frames highlight the animation of a radial manual tour manipulating $aede2$: (1) original projection, (2) full contribution, (3) zero contribution, before returning to the original contribution.

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 e and its shadow in A .
2. Set an angle increment ($\Delta\phi$) 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**.
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 $\phi(0, \pi/2, \phi_1)$ 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.

3.3 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;
- \mathbf{M}_{pd+1} , the manipulation space
- 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:

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

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:

$$\mathbf{Y}_{npi} \leftarrow \mathbf{X}_{np} \cdot \mathbf{M}_{p \cdot 1:d}$$

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

end for

3.4 Package structure

This section describes the functions available in the package, their usage, and how to install and get up and running.

3.4.1 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.

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

3.4.2 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)

## The tour path of the bases

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(alpa = .8, size = 1.5)) +
```

```
proto_basis("left") +  
proto_origin()  
  
## Animating  
animate_plotly(ggt, fps = 10)
```

3.4.3 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")
```

3.5 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.

3.5.1 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.

3.5.2 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.

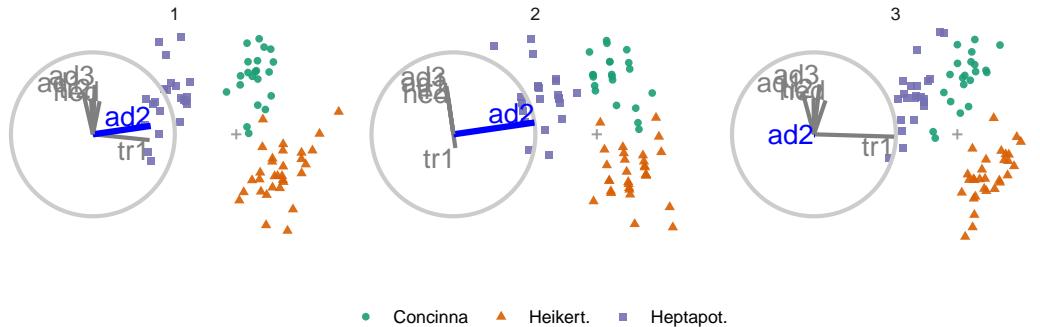


Figure 3.4: Select frames from a radial tour of PC4 within the jet cluster, with color indicating experiment type: ATLAS7new (green) and ATLAS7old (orange). When PC4 is removed from the projection (frame 10), there is little difference between the clusters, suggesting that PC4 is important for distinguishing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

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](#).

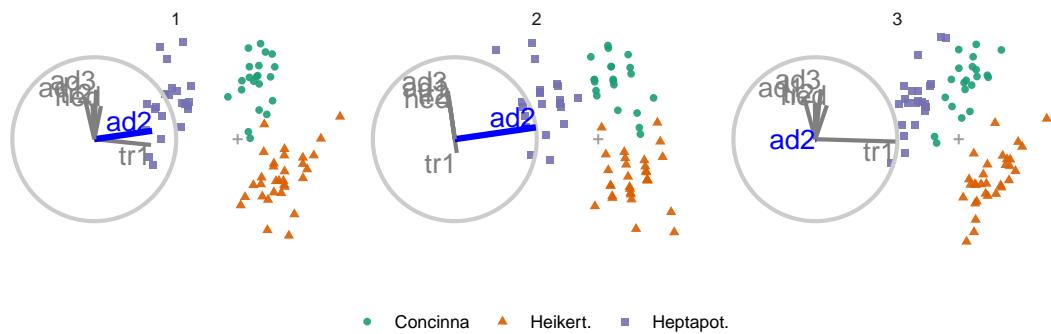


Figure 3.5: Frames from the radial tour manipulating PC_3 within the jet cluster, with color indicating experiment type: ATLAS7new (green) and ATLAS7old (orange). When the contribution from PC_3 is changed, there is little change in the separation of the clusters, suggesting that PC_3 is not important for distinguishing the experiments.

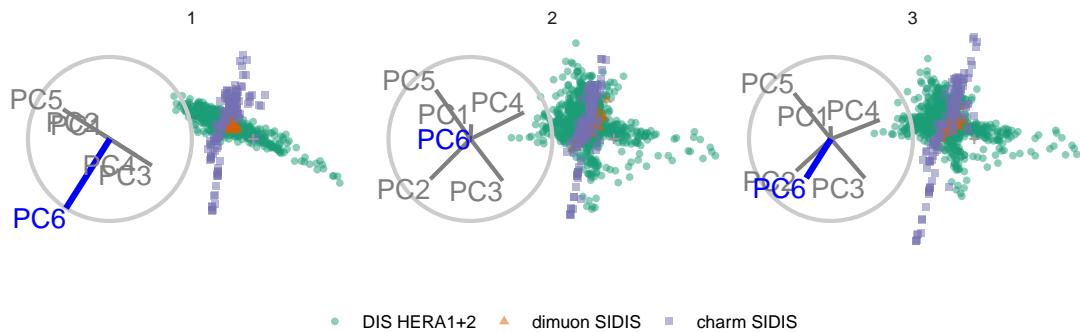


Figure 3.6: Select frames from a radial tour exploring the sensitivity that PC_6 has on the structure of the DIS cluster, with color indicating experiment type: DIS HERA1+2 (green), dimuon SIDIS (purple), and charm SIDIS (orange). DIS HERA1+2 is distributed in a cross-shaped plane, and charm SIDIS occupies 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 .

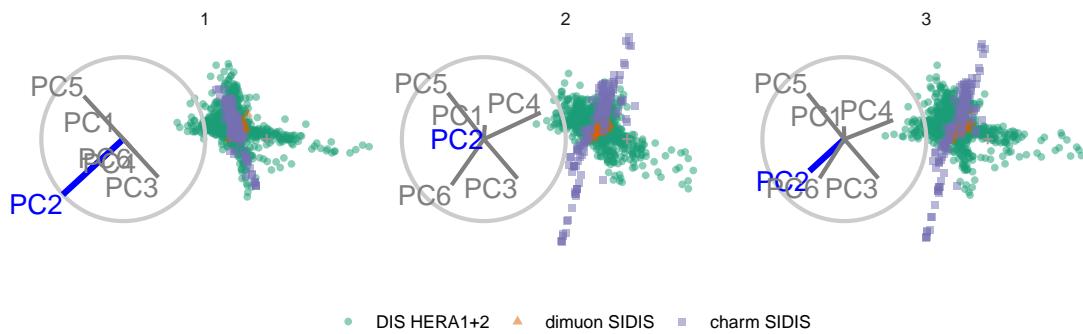


Figure 3.7: Frames from the radial tour exploring the sensitivity PC_2 to the structure of the DIS cluster, with color indicating experiment type: DIS HERA1+2 (green), dimuon SIDIS (purple), and charm SIDIS (orange). As the contribution of PC_2 is decreased, dimuon SIDIS becomes more distinguishable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

3.6 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.

Another future research topic could be to extend the algorithm for use on 3D projections. With the current popularity and availability of 3D virtual displays, this may benefit the detection and understanding of the higher dimensional structure or enable the examination of functions.

Having a graphical user interface would be useful for making it easier and more accessible to a general audience. This is possible to implement using **shiny** (Chang et al., 2020). The primary purposes of the interface would be to allow the user to interactively change the manip variable easily and the interpolation step for more or less detailed views.

3.7 Acknowledgments

This article was created in R, using **knitr** (Xie, 2020) and **rmarkdown** (Allaire et al., 2020), with code generating the examples inline. The source files for this article be found at github.com/nspyrison/spinifex_paper/. The animated gifs can also be viewed at this site. The source code for the **spinifex** package can be found at github.com/nspyrison/spinifex/.

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 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.

Principal component analysis is a long-standing go-to method for exploring multivariate data. Data visualization *tours* are a class of linear projections animated over small changes in the projection basis. The grand tour original selects target bases at random to animate between. Alternatively, the manual tour rotates the contribution of a selected variable novelly offering analysts a means to control the basis. This paper describes a within-participants user study evaluating the radial tour's efficacy compared with principal component analysis and the grand tour. 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.

4.1 Introduction

Despite decades of research multivariate data continues to provide fascinating challenges for visualization. It is important because data visualization is an key element of exploratory data analysis, (EDA, Tukey, 1977), assessing model assumptions and diagnosing the fit, and as a cross-check on numerical summarization (Anscombe, 1973; Matejka and Fitzmaurice, 2017; Yanai and Lercher, 2020). One of the challenges is determining whether a new technique yields better perception of information than current practices, for multivariate data.

Dimension reduction is commonly used with visualization to provide informative low-dimensional summaries of multivariate data. (Note that this is usually only appropriate when all variables are quantitative.) Principal component analysis (PCA, Pearson, 1901) is one of the first methods developed and it remains very popular. Visualization of PCA is typically in the form static scatterplots of a few of the leading components. This is usually accompanied by a representation of the original variables contribution to the principal components, using a biplot (Gabriel, 1971).

The class of dynamic linear projections known as *tours* (Asimov, 1985), shown as scatterplots or biplots, can be used to animate multiple principal components. Instead of a static view of two orthogonal components, interpolations between multiple components can be viewed. Asimov originally animated between randomly selected bases (which could be considered combinations of components) in the *grand tour*. The *manual tour* (Cook and Buja, 1997) uniquely allows for user-control over the basis changes. A selected variable (or component) can be rotated to the desired contribution. A radial tour (Spyrison and Cook, 2020) is a special version of the manual tour, available in the R package `spinifex`. The permanence of the data points from frame to frame and information held in intermediate interpolated frames or user-control of the basis could plausibly lead to more information perception than a static display. This is a hypothesis that could be assessed by a user study.

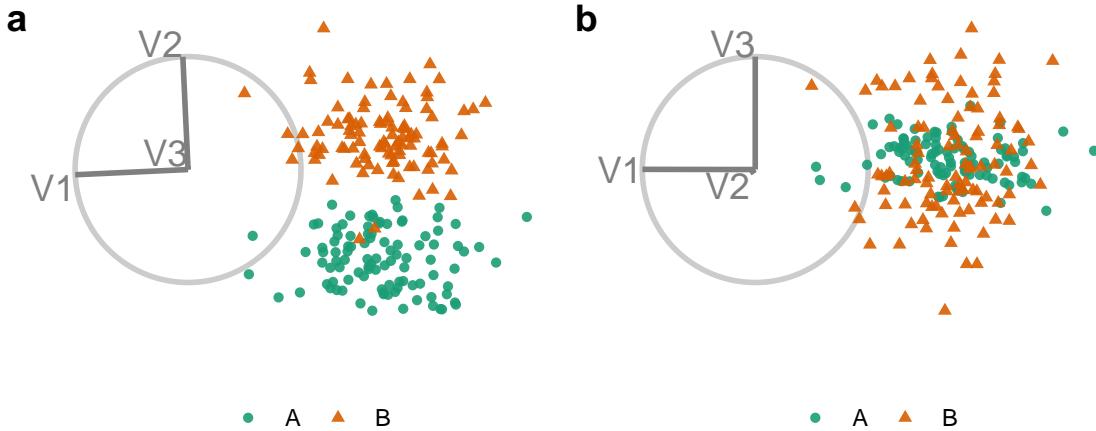


Figure 4.1: Illustration of cluster separation. Panel (a) shows clear separation in the V_2 , no separation in V_1 , V_3 has negligible contribution to the frame. The contributions of V_2 is then swapped with V_3 in panel (b). There is no separation between the cluster means in V_3 , and the separation contained in V_2 no longer influences the frame.

There have been several user studies comparing scatterplots of dimension-reduced spaces across 2D and 3D scatterplots on traditional 2D monitors (Gracia et al., 2016; Wagner Filho et al., 2018). There are also empirical statistics used to describe non-linear reduction and how well and faithfully they embed the data in a lower space (Bertini, Tatu, and Keim, 2011; Liu et al., 2017; Sedlmair, Munzner, and Tory, 2013; Maaten and Hinton, 2008). Nelson, Cook, and Cruz-Neira (1998) even compared 2D with 3D scatterplots for detecting clusters, and dependence, in multivariate data. However, there is an absence of studies comparing different visual techniques.

We are particularly interested in assessing the effectiveness of the new radial tour relative to common practice with PCA, and relative to a grand tour. The radial tour is particularly useful for understanding variable importance relative to structure visible in a scatterplot. If the contribution of a variable is reduced and the structure disappears, then that variable makes an important role in the presence of that structure in the data. For example, if in the scatterplot cluster can be seen, but them gap between them is reduced when a variable is removed it means that the clustering is due, at least partially, to that variable. Understanding which variables are important is a key aspect of interpretability of models.

Knowing which variables to use is also important for statistical modeling and their interpretations. Models are becoming increasingly complex involving many interacting, or otherwise non-linear terms cause an opaqueness to their interpretability. Exploratory Artificial Intelligence (XAI, Adadi and Berrada, 2018; Arrieta et al., 2020) is an emerging field that attempts to extend the interpretability of such black-box models by offering techniques to increase their interpretability. 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 paper is structured as follows. Section 4.2 discusses several visualization methods and orthogonal and observation-based visuals before arriving at the three linear dimension reduction techniques compared in the study. Section 4.3 describes the experimental factors, task, and accuracy measure used. The results of the study are discussed in Section 4.4. Conclusions and potential future directions are discussed in Section 4.5. The software used for the study is described in Section 4.6.

4.2 Background

Before discussing PCA, the grand tour, and the radial tour, this section covers orthogonal views and observation-based visuals of the full variable space. Consider data to be a complete matrix of n observations across p variables, $X_{n \times p}$.

4.2.1 Scatterplot matrix

One could consider looking at p histograms or univariate densities. Doing so will miss features in two or more dimensions. A scatterplot matrix (Chambers et al., 1983) is a $p \times p$ matrix with univariate densities on the diagonal and all combinations of pairs of variables in off-diagonal elements. Figure 4.2 shows a scatterplot matrix of the first four components of simulated data. Such displays do not scale well with dimension, quickly becoming dense. Scatterplot matrices also display information in two orthogonal dimensions; features in three dimensions will never be fully resolved.

4.2.2 Parallel coordinates plot

Another common way to display multivariate data is with a parallel coordinates plot (Ocagne, 1885), which shows observations by quantile or normalized values for each variable connected by lines to the quantile value in subsequent variables. Parallel coordinates plot and other observations-based visuals, such as pixel plots or Chernoff faces scale well with dimensions but poorly with observations. These are perhaps best used when there are more variables than observations.

Observations-based visuals have a couple of issues. They are asymmetric across variable ordering, leading to different conclusions or features being focused on due to variable order. Another notable observation-based visual is the graphical channel used to convey information. Munzner suggests that position is the visual channel that is most perceptible to humans (Munzner, 2014). In the case of parallel coordinates plots, the horizontal axes span variables rather than the values of one variable; the loss of a display dimension to be used by most discerning visual channel.

At some point, we will be forced to turn to dimension reduction to scale well. Non-linear transformations bend and distort spaces are not entirely accurate or faithful to the original variable space. In light of this, we preclude non-linear techniques and instead decide on PCA, the grand tour, and the radial tour.

4.2.3 Principal component analysis

PCA is a good baseline of comparison for linear projections because of its frequent and broad use across disciplines. Principal component analysis (Pearson, 1901) defines new components, linear combinations of the original variables, ordered by decreasing variation through the help of eigenvalue matrix decomposition. While the resulting dimensionality is the same size, the benefit comes from the ordered nature of the components. The data can be said to be approximated by the first several components. The exact number is subjectively selected given the variance contained by each component, typically guided using a scree plot (Cattell, 1966). Features with sizable signal regularly appear in the leading components that commonly approximate data. However, this is not always the

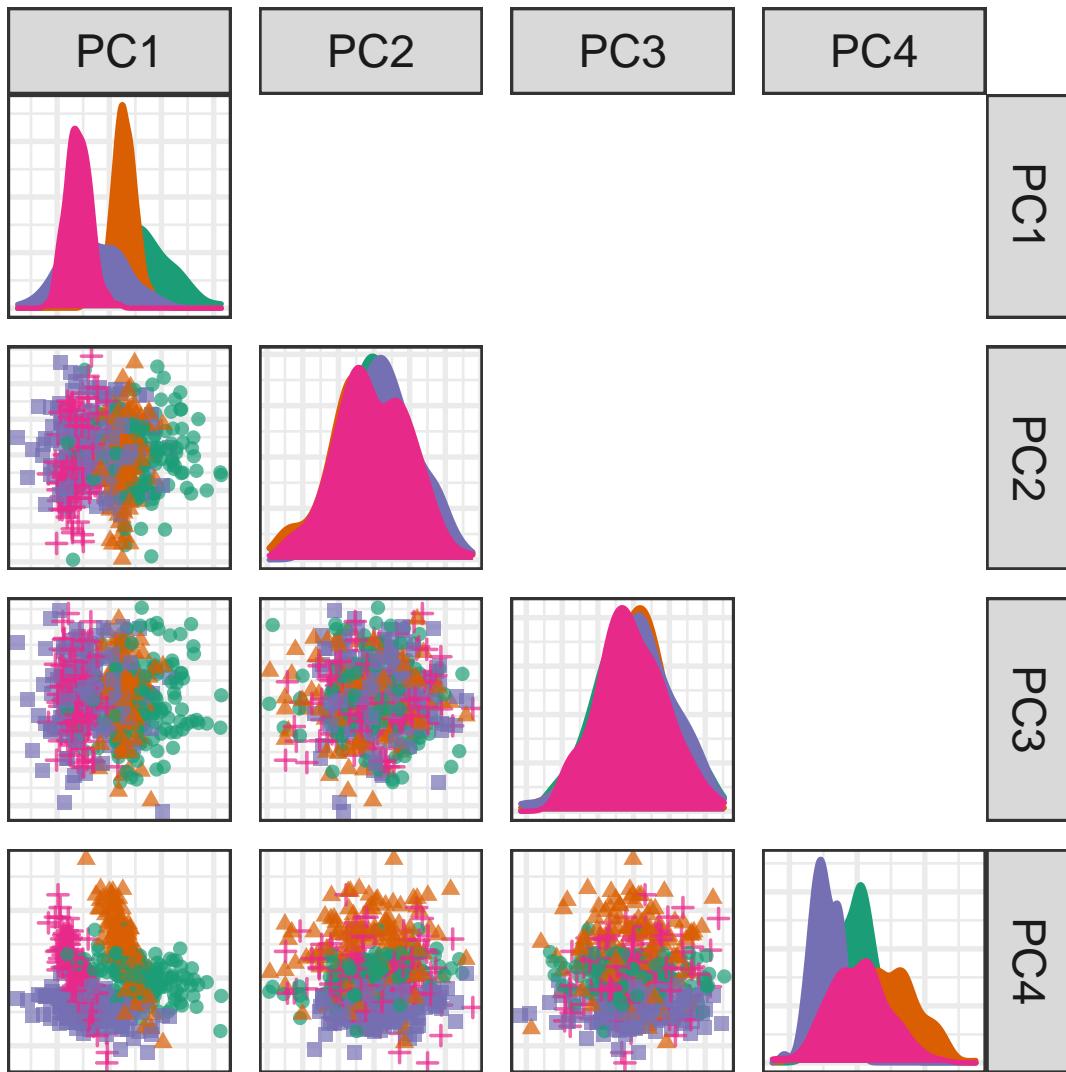


Figure 4.2: Scatterplot matrix of the first four principal components simulated data in six dimensions. An analyst would have to view both PC1 by PC2 and PC1 by PC4 to have a thorough take on which variables attribute to the separation between clusters.

case, and component spaces should be fully explored to look for signal in components that hold less variation.

4.2.4 Animated linear projections, tours

A data visualization *tour* animates many linear projections over small changes in the projection basis. One of the insightful features of the tour is the object permanence of the data points; one can track the relative changes of observations as the basis moves, as

opposed to discretely jumping to an orthogonal view with no intermediate information. Types of tours are distinguished by the selection of their basis paths (Lee et al., 2021; Cook et al., 2008). To contrast with the discrete orientations of PCA, we compare continuous linear projection changes with grand and radial tours.

Grand tours

Target bases are selected randomly in a grand tour (Asimov, 1985). These target bases are then geodesically interpolated for a smooth, continuous path. The grand tour is the first and most widely known tour. The random selection of target bases makes it a general unguided exploratory tool. The grand tour will make a good comparison that has continuity of data points similar to the radial tour but lacks the user control enjoyed by PCA and radial tours.

Manual and radial tours

Whether an analyst uses PCA or the grand tour, they cannot influence the basis. They cannot explore the structure identified or change the contribution of the variables. This user-control-steering is a key aspect of *manual* tours that should facilitate testing variable attribution.

The manual tour (Cook and Buja, 1997) defines its basis path by manipulating the basis contribution of a selected variable. A manipulation dimension is appended onto the projection plane, giving a full contribution to the selected variable. The target bases are then chosen to rotate this newly created manipulation space. This manipulation space similarly orthogonally restrained, the data is projected through its interpolated frames and rendered into an animation. When the contribution of one variable changes, the contributions of the other variables must also change, maintaining the orthonormality of the basis and space. A key feature of the manual tour is that it allows users to control the variable contributions to the basis. Such manipulations can be queued in advance or selected in real-time for human-in-the-loop analysis (Karwowski, 2006). Manual navigation is relatively time-consuming due to the vast volume of resulting view space and the abstract method of steering the projection basis. It is advisable first to identify a basis of particular

interest and then use the manual tour as a more directed, local exploration tool to explore the sensitivity of a variable's contribution to the feature of interest.

To simplify the task and keep its duration realistic, we consider a variant of the manual tour called a *radial* tour. In a radial tour, the selected variable can change its magnitude of contribution but not its angle; it must move along the direction of its original contribution radius. The radial tour benefits from both continuity of the data alongside grand tours and allows the user to steer via choosing the variable to rotate.

Manual tours have been recently made available in the **R** package **spinifex** (Spyrison and Cook, 2020), which facilitates manual tours (and radial variant). It also provides an interface for a layered composition of tours and exporting to .gif and .mp4 with **ganimate** (Pedersen and Robinson, 2020) or .html widget with **plotly** (Sievert, 2020). It is also compatible with tours made by **tourr** (Wickham et al., 2011). Now that we have a readily available means to produce various tours, we want to see how they fare against traditional discrete displays commonly used with PCA.

4.3 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.

4.3.1 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 both 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 4.9, 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 avoid 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:

$$H_0 : \text{task accuracy does not change across visualization method}$$
$$H_\alpha : \text{task accuracy does change across visualization method}$$

4.3.2 Experimental factors

In addition to visual factor, we simulate the data across three aspects: 1) 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, 2) the *shape* of the clusters, to reflect varying distributions of the data, and 3) the *dimension-ality* of the data. Below we describe the levels within each factor, while Figure 4.3 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 to this, 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

Levels of the experimental factors

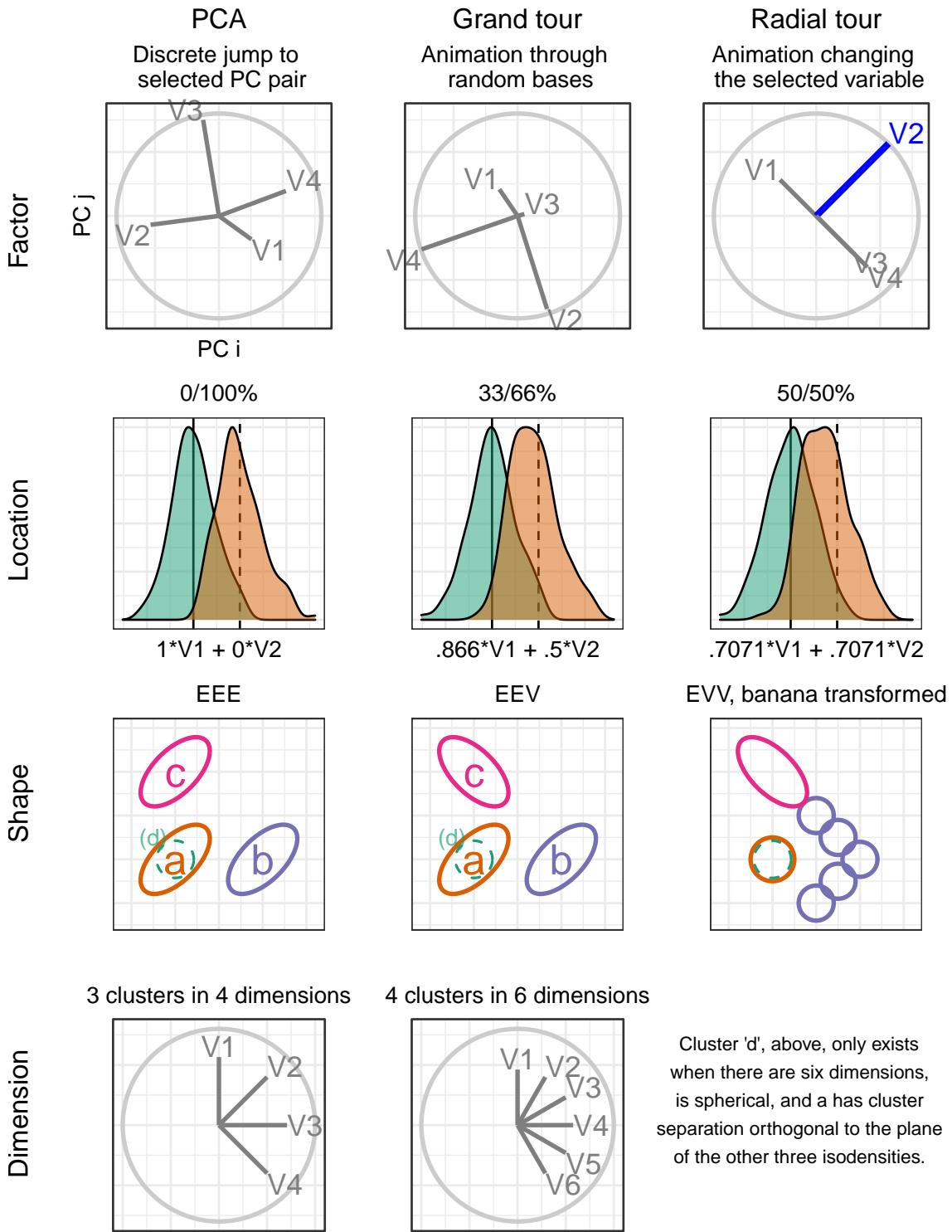


Figure 4.3: Illustration of the experimental factors, the parameter space of the independent variables, the support of our study.

of the clusters, the levels of which are either `_E_ual` 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.

4.3.3 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 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 uniformed 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 of 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. With these in mind, we define the following measure for evaluating the task.

Let a data $\mathbf{X}_{n \times p}$ 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 number of 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 amount of difference each variable would hold 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.

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

Where accuracy, A , is defined as:

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

Where $I(j)$ is the indicator function, the binary response for variable j .

4.3.4 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 4.5. 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.

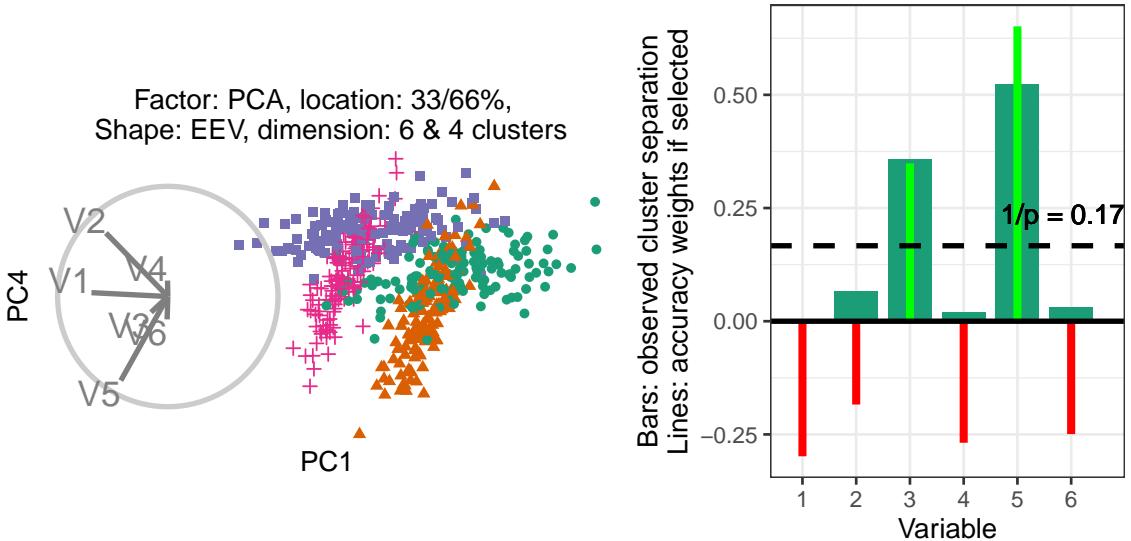


Figure 4.4: (L) PCA biplot of the components showing the most cluster separation with (R) illustration of the magnitude of cluster separation is for each variable (bars) and the weight of the variable accuracy if selected (red/green lines). The horizontal dashed line is $1/p$, the amount of separation each variable would have if evenly distributed. The weights equal the signed square of the difference between each variable value and the dashed line.

PCA inputs allowed users to select between the top four principal components for both the x- and y-axis regardless of the data dimensionality (four or six). Data was simulated to have cluster separation within the 2nd to 4th components, ideally not simple enough to be entirely observed in PC1 by PC2. But not burying 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. The display would change the start of radially increasing the contribution of the selected variable until it was full, zeroed, 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.

4.3.5 Data simulation

Each dimension is originally distributed as $\mathcal{N}(0, 1)$ | covariance Σ , a function of 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 vary. 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 4.3, 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, every pair of the top four principal components and saved as 12 plots. 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, while mitigating bias from fringe selection of target bases. 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 and no variable contributing in the opposite direction (to minimize the dependence between variable contributions), a radial tour is then produced for each variable and saved as a .gif.

4.3.6 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

Consider a new participant, the 63rd participant,

- 1) Set the factor order:

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

- 2) Set location order:

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

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

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

factor order
permutation

Period 1 Period 2 Period 3

location order
permutation

Period 1 Period 2 Period 3

Period Evaluation Factor¹ Location² Shape³ Dimensions⁴

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 EEV 4 (3cl)

2 3 PCA 50/50 EVV 4 (3cl)

2 4 PCA 50/50 EVV 6 (4cl)

3 Train 3 Radial 0/100 EVV 4 (3cl)

3 5 Radial 0/100 Ban 4 (3cl)

3 6 Radial 0/100 Ban 6 (4cl)

Figure 4.5: Illustration of how a hypothetical participant 63 is assigned experimental factors. Each of the 6 factor order permutations is exhausted before iterating to the next permutation of location order.

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 is evaluated on two tasks with the same factor and location level across the increasing difficulty of dimension and shape. The plot was removed after 60 seconds, though this limit was rarely reached by participants.

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 factors once. This randomization controls for potential learning effects the participant may receive. Figure 4.5 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.

4.3.7 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 factors. 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.

4.3.8 Data collection

Data were recorded by a **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 most minor 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.

4.4 Results

To recap, the primary response variable is task accuracy, as defined in section 4.3.3. 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 4.3.2. 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. The, we use likert plots and rank-sum tests to compare the subjective measures between the visual factors.

4.4.1 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 that can be attributed to the effect of the individual participant 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

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 penalizes 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	1000	1027	0.180	0.022	0.462
a+b+c+d	4	8	1026	1075	0.187	0.030	0.460
a*b+c+d	5	12	1036	1103	0.198	0.043	0.457
a*b*c+d	8	28	1069	1207	0.238	0.080	0.447
a*b*c*d	15	54	1125	1380	0.282	0.111	0.438

model with all interaction terms is included. The matrices for models with more than a two terms is rank deficient; there is not enough varying information in the data to explain all interacting terms.

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

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 $| l \in (4 \text{ variables \& 3 cluster}, 6 \text{ variables \& 4 clusters})$

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

We also want to visually explore the conditional variables in the model. Figure 4.6 explores violin plots of accuracy by factor while faceting on the location (vertical) and shape

Table 4.2: The task accuracy model coefficients for $\hat{Y} = \alpha \cdot \beta + \gamma + \delta$, with factor = pca, location = 0/100%, and shape = EEE held as baselines. Factor being radial is the fixed term with the strongest evidence supporting the hypothesis. When crossing the visual factor there is some evidence suggesting radial performs worse with 33/66% mixing. The model fit is based on the 648 evaluations by the 108 participants.

	Estimate	Std. Error	df	t value	Pr(> t)
(Intercept)	0.03	0.08	44.2	0.44	0.66
factor					
Factorpca	-0.15	0.09	622.4	-1.74	0.08
Factorradial	0.22	0.09	618.6	2.46	0.01 *
fixed effects					
Location33/66%	0.10	0.09	84.9	1.09	0.28
Location50/50%	0.05	0.09	83.0	0.58	0.56
ShapeEEV	0.04	0.06	11.5	0.79	0.44
Shapebanana	-0.03	0.06	11.5	-0.48	0.64
Dim6	-0.06	0.05	11.5	-1.39	0.19
interactions					
Factorpca:Location33/66%	0.06	0.13	587.3	0.49	0.63
Factorradial:Location33/66%	-0.28	0.13	577.6	-2.14	0.03 *
Factorpca:Location50/50%	0.09	0.13	589.6	0.68	0.50
Factorradial:Location50/50%	-0.10	0.13	588.0	-0.77	0.44

(horizontal). Use of the radial visual, on average, increases the accuracy, and especially so when the location of signal mixing is not 33/66%.

4.4.2 Subjective measures

The 84 evaluations of the post-study survey also collect four subjective measures for each factor. Figure 4.7 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 and found radial tours easier than grand tours. All factors have reportedly low familiarity, as expected from crowdsourced participants.

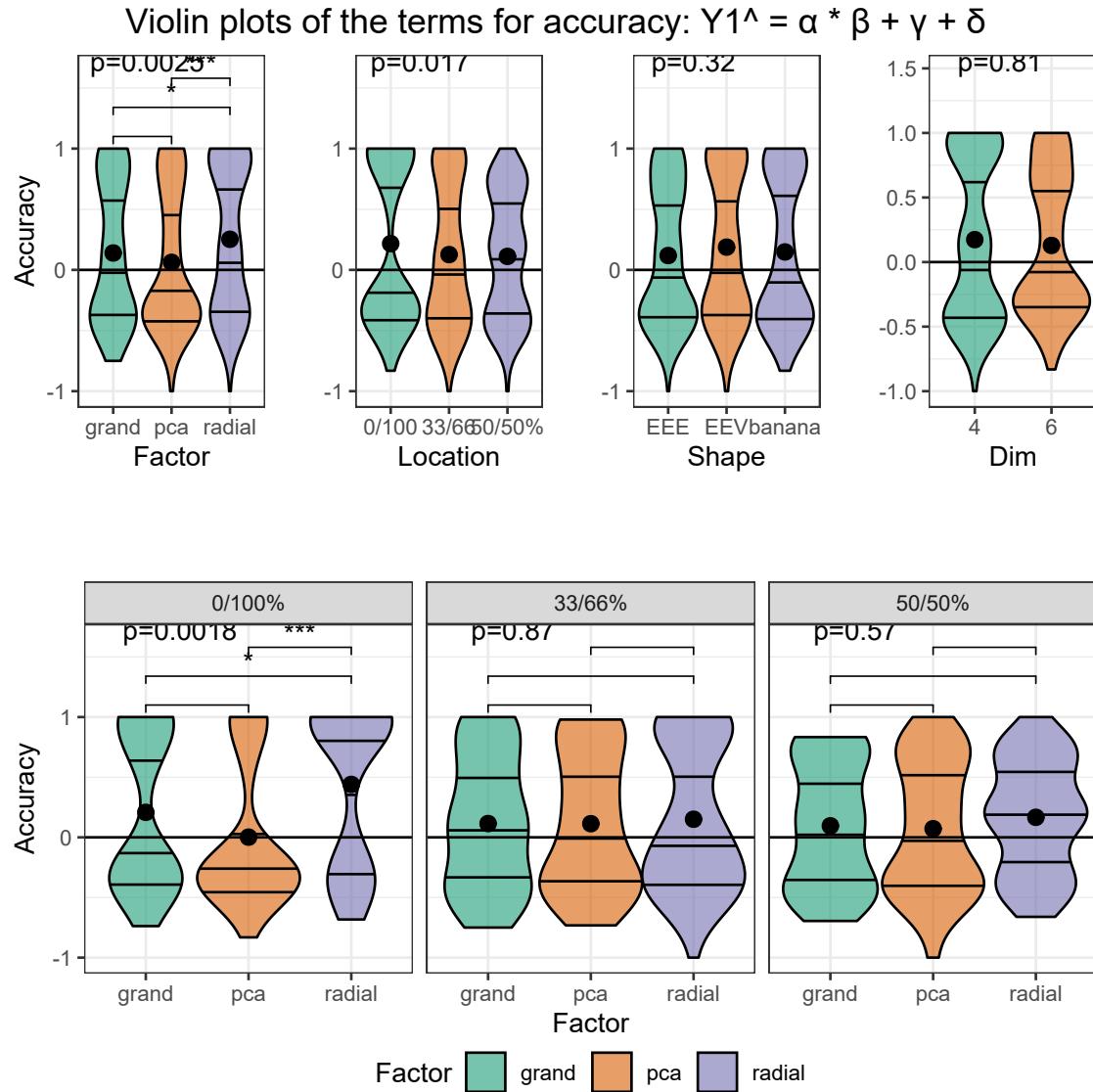


Figure 4.6: Violin plots of terms of the model $\hat{Y} = \alpha \cdot \beta + \gamma + \delta$. Overlaid with global significance from the Kruskal-Wallis test and pairwise significance from the Wilcoxon test, both are non-parametric, ranked-sum tests suitable for handling discrete data. Participants 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.

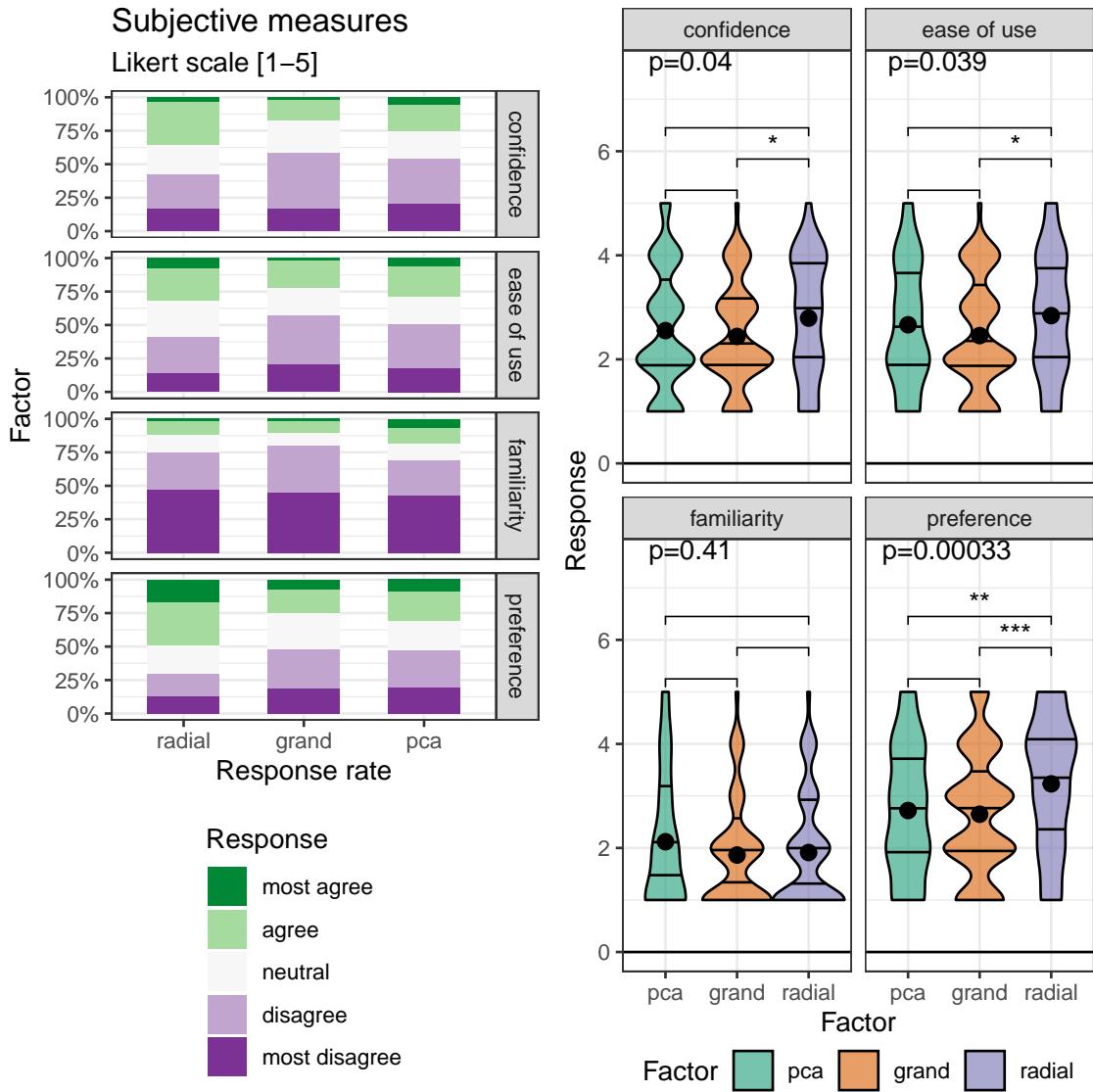


Figure 4.7: The subjective measures of the 84 responses of the post-study survey, five discrete Likert scale levels of agreement (L) Likert plots (stacked percent bar plots) with (R) violin plots of the same measures. Violin plots are overlaid with global significance from the Kruskal-Wallis test, and pairwise significance from the Wilcoxon test, both are non-parametric, ranked sum tests.

4.5 Conclusion

Data visualization is an integral part of EDA. Yet through exploration of data in high dimensions become 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 type of 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.

4.6 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., 2020) application that allows users to apply various preprocessing 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()
```

4.7 Acknowledgments

This research was supported by an Australian Government Research Training Program (RTP) scholarship. This article was created in **R** (R Core Team, 2020) and **rmarkdown** (Xie, Allaire, and Grolemund, 2018). Visuals were prepared with **spinifex**. All packages used are available from the Comprehensive R Archive Network **CRAN**. The source files for this article, application, data, and analysis can be found on **GitHub**. The source code for the **spinifex** package and accompanying shiny application can be found **here**. We thank Jieyang Chong for his help in proofreading this article.

4.8 References

00

4.9 Appendix

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.

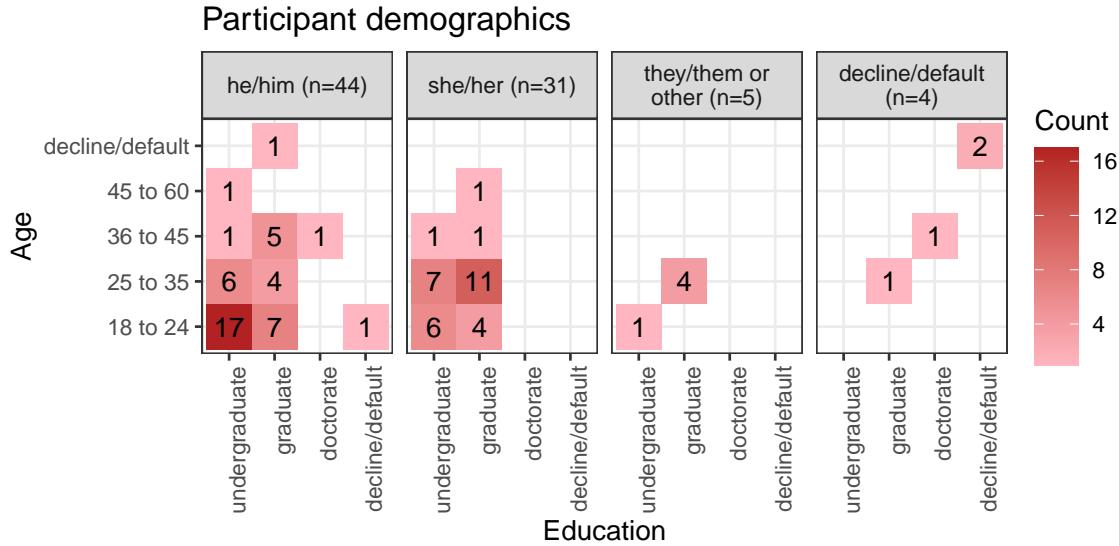


Figure 4.8: Heatmaps of survey participant demographics; counts of age group by completed education as faceted across preferred pronoun. Our sample tended to be between 18 and 35 years of age with an undergraduate or graduate degree.

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 is 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	1000	1027	0.180	0.022	0.462
a+b+c+d	4	8	1026	1075	0.187	0.030	0.460
a*b+c+d	5	12	1036	1103	0.198	0.043	0.457
a*b*c+d	8	28	1069	1207	0.238	0.080	0.447
a*b*c*d	15	54	1125	1380	0.282	0.111	0.438

4.9.1 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. 2) Select the model with the same effect terms, $\alpha \cdot \beta + \gamma + \delta$, and examine its coefficients.

Table 4.4: Model coefficients for log response time [seconds] $\widehat{Y}_2 = \alpha \cdot \beta + \gamma + \delta$, with factor = PCA, location = 0/100%, and shape = EEE 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)	2.48	0.14	42.8	17.41	0.00	***
Factor						
Factorpca	0.23	0.12	567.6	1.97	0.05	*
Factorradial	0.39	0.12	571.9	3.29	0.00	**
Fixed effects						
Location33/66%	0.29	0.14	42.0	2.06	0.05	*
Location50/50%	0.07	0.14	40.5	0.54	0.59	
ShapeEEV	-0.15	0.09	8.3	-1.61	0.14	
Shapebanana	-0.13	0.09	8.3	-1.42	0.19	
Dim6	0.14	0.08	8.3	1.90	0.09	
Interactions						
Factorpca:Location33/66%	-0.24	0.18	580.9	-1.34	0.18	
Factorradial:Location33/66%	-0.48	0.18	583.1	-2.63	0.01	**
Factorpca:Location50/50%	-0.12	0.18	578.6	-0.69	0.49	
Factorradial:Location50/50%	-0.08	0.18	580.9	-0.43	0.67	

Random effect ranges

Residual plots have no noticeable non-linear trends and contain striped patterns as an artifact from regressing on discrete variables. Figure 4.9 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 4.10 shows the Gelman simulations and marginal effects of the simulation and participants for Y_2 , the same model regressing on log response time.

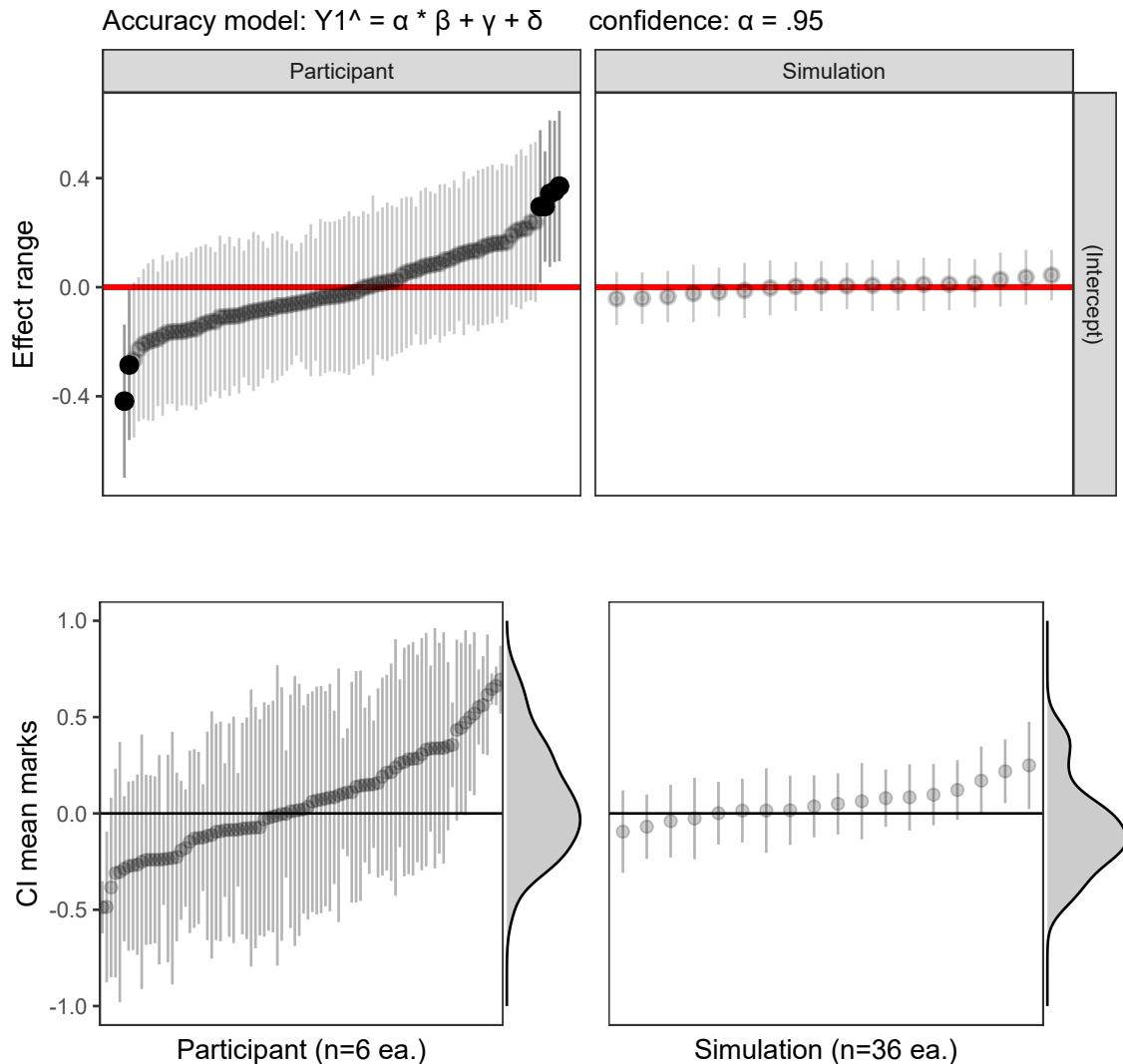


Figure 4.9: Accuracy model: (T) Estimated effect ranges of the random effect terms participant and data simulation of the accuracy model, $\widehat{Y}_1 = \alpha \cdot \beta + \gamma + \delta$. Confidence intervals are created with Gelman simulation on the effect posterior distributions. The effect size 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.

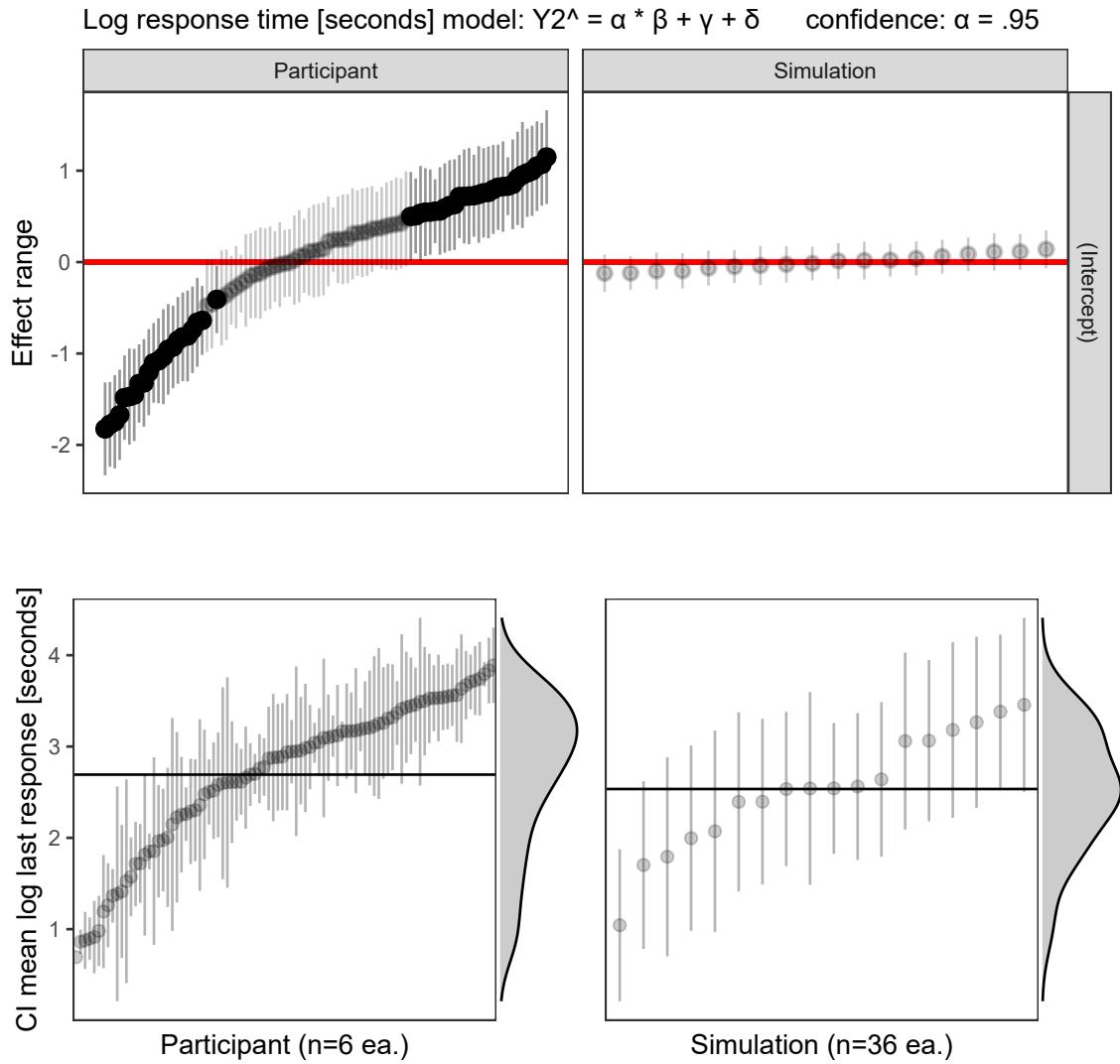
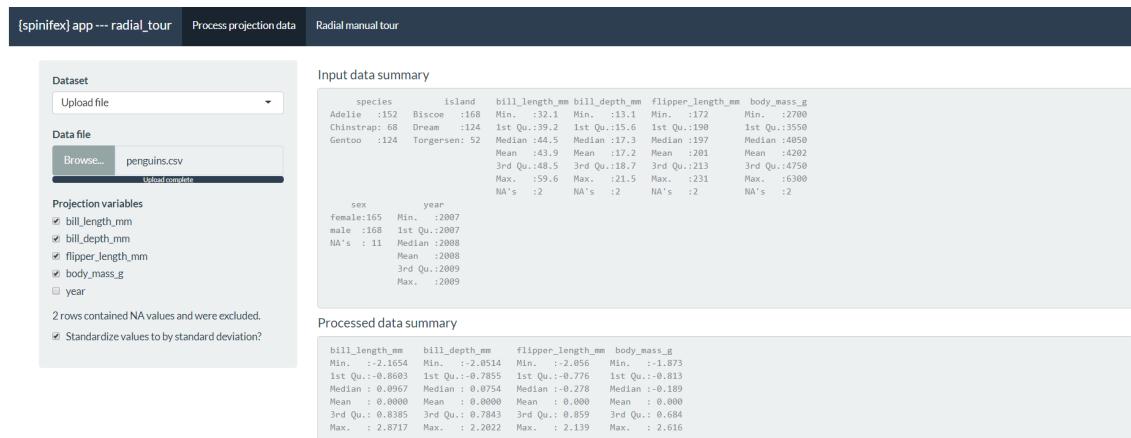


Figure 4.10: Log response time model: (T) The effect ranges of Gelman resimulation on posterior distributions for the time model, $\widehat{Y}_2 = \alpha * \beta + \gamma + \delta$. These show the magnitude and distributions of particular participants and simulations. Simulation has a relatively small 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.



radial_tour app, --- [spinifex] ver. 0.3.0 --- 2021-07-01

Figure 4.11: *Process data tab, interactively loads or select data, check which variables to project, and optionally scale columns by standard deviation.*

Radial tour application details

Below we describe the **shiny** app available in the **spinifex** package which runs locally. It streamlines creating and interacting with radial tours more interactive than the application used in the user study.

In the initial tab, 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 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.

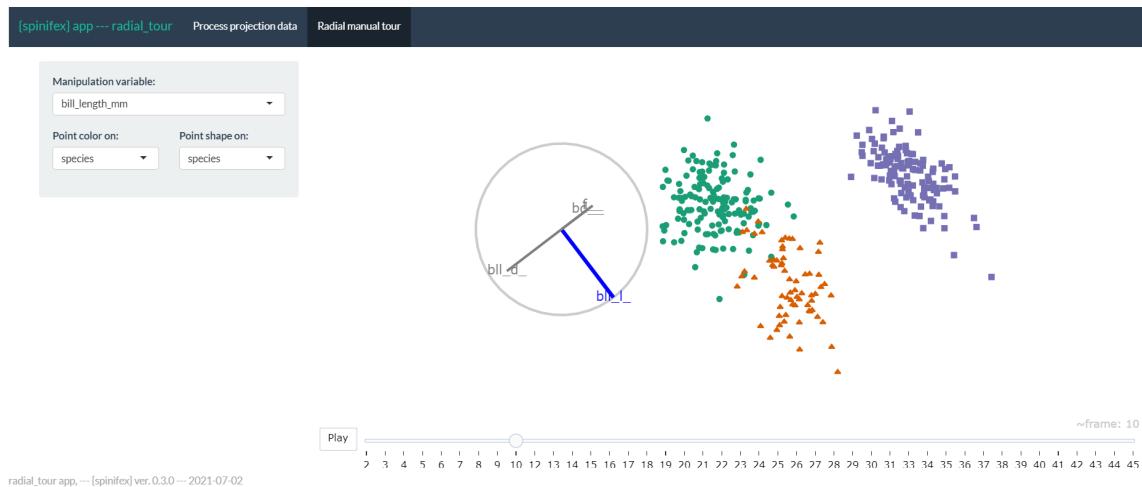


Figure 4.12: Radial tour tab, interactively create radial tours, changing the manipulation variable, and color or shape of the resulting manual tour. Here, the palmer penguins data is being explored, bill length was selected to manipulate as it is the only variable separating the green cluster from the 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.

Chapter 5

Exploring Local Explanations of Non-linear Models Using Animated Linear Projections

In the previous chapter I perform a within-participants user study. Now we can be more confident that the radial tour leads to better analysis of variable-level attribution to features identified in a projection. We want to increase the interpretability of complex models. Specifically, I suggest a using the radial tour to explore variable sensitivity to the structure identified in linear local explanations of non-linear models.

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) response features. The methods are implemented in the **R** package **cheem**, available on CRAN.

5.1 Introduction

There are different reasons and emphases to fit a model. Breiman (2001), reiterated by Shmueli (2010), taxonomize modeling based on its purpose; *explanatory* modeling is done for some inferential purpose, while *predictive* modeling focuses more on the predictions of out-of-sample instances. The intended use has important implications for model selection and development. In explanatory modeling, interpretability is vital for drawing inferential conclusions. While predictive modeling may opt for more accurate non-linear models. The use of black-box models is becoming increasingly common, but not without their share of controversy (O’Neil, 2016; Kodiyan, 2019). However, 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. Feature-level (variable-level) interpretability of models is essential in evaluating such biases. It is also generally important for many problems, where it is not enough to predict accurately. Still, one must be able to explain which predictors are most responsible for generating a response value.

Another concern is data drift, a shift in support or domain of the explanatory features (variable or predictors). Non-linear models are typically more sensitive and do not extrapolate well outside the training data domain. Better interpretability of the model means more transparency where models’ predictions may be plausible or completely unreliable.

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 feature importance at the location of each instance (observation), or the predictions at a specific point in the data domain. Because these are point-specific, it is challenging to visualize them to comprehensively understand a model.

In multivariate data visualization, a *tour* (Asimov, 1985; Buja and Asimov, 1986; Lee et al., 2021) is a sequence of linear projections of data onto a lower-dimensional space. Tours are viewed as an animation over minor changes to the projection basis. Structure in a projection can then be explored visually to see which features contribute to the formation of that structure. The intuition is similar to watching the shadow of a hidden 3D object change as the object is rotated; watching the shape of the shadow change conveys information about the structure and features of the object.

There are various types of tours distinguished by the generation of projection bases. In a *manual* tour (Cook and Buja, 1997; Spyrisson and Cook, 2020), the path is defined by changing the contribution of a selected feature. Applying tours to models has been done in a couple of contexts. Specifically for exploring various statistical model fits and classification boundaries (Wickham, Cook, and Hofmann, 2015), and using tree- and forest-based approaches as a projection pursuit index to generate a tour basis paths (Lee et al., 2013; da Silva, Cook, and Lee, 2021).

We use the radial manual tour to scrutinize a local explanation in our proposed approach. Additional interactivity allows the user to identify an instance of interest, then explore its local explanation by changing feature contribution with the radial tour. The methods are implemented in R package **cheem**. Example datasets are provided to illustrate usage for classification and regression tasks.

Using a radial tour can be considered similar to counterfactual, what-if analysis, such as *ceteris paribus* (Biecek, 2020). This phrase, Latin for “other things held constant” or “all else unchanged”, shows how an instance’s prediction would change from a marginal change in one explanatory feature given that other features are held constant. It ignores correlations of the features and imagines a case that was not observed. In contrast, our approach is a geometric explanation of the factual; it varies contributions of the features by rotating the basis, a reorientation of the data object. A constraint in our approach is that the basis must remain orthonormal. When the contribution of one feature decreases, the contributions of others necessarily increase such that there is a complete component in that direction. This also ensures that what is seen is strictly a low-dimensional projection from high-dimensions and is thus an interpretable visualization.

The remainder of this paper is organized as follows. The following Section, 5.2, covers the background of the local explanation and the traditional visuals produced. Section 5.3 explains the animations of continuous linear projections. Section 5.4 discusses the visual layout in the interactive interface, how they facilitate analysis, data preprocessing, and package infrastructure. Then Section 5.5 illustrates the application to supervised learning with categorical and quantitative response features. We conclude with Section 5.6 of the insights gained and directions that might be explored in the future.

5.2 Local explanation statistics

Consider a highly non-linear model. It can be hard to determine whether small changes in a feature’s value will make a class prediction change group or identify which features contribute to an extreme residual. Local explanations shed light on these situations by approximating linear feature importance in the vicinity of a single instance.

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, 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 instance-level explanations are used in various ways depending on the data. In image classification, where pixels correspond to predictors, saliency maps overlay or offset a heatmap indicating important pixels (Simonyan, Vedaldi, and Zisserman, 2014). For instance, pixels corresponding to snow may be highlighted when distinguishing if a picture contains a wolf or husky. 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 feature additive contributions from the model intercept to the instance’s prediction (Ribeiro, Singh, and Guestrin, 2016).

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 features already included. 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.

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 the use of SHAP. 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 wages [2020 Euros], from the skill measures. 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.

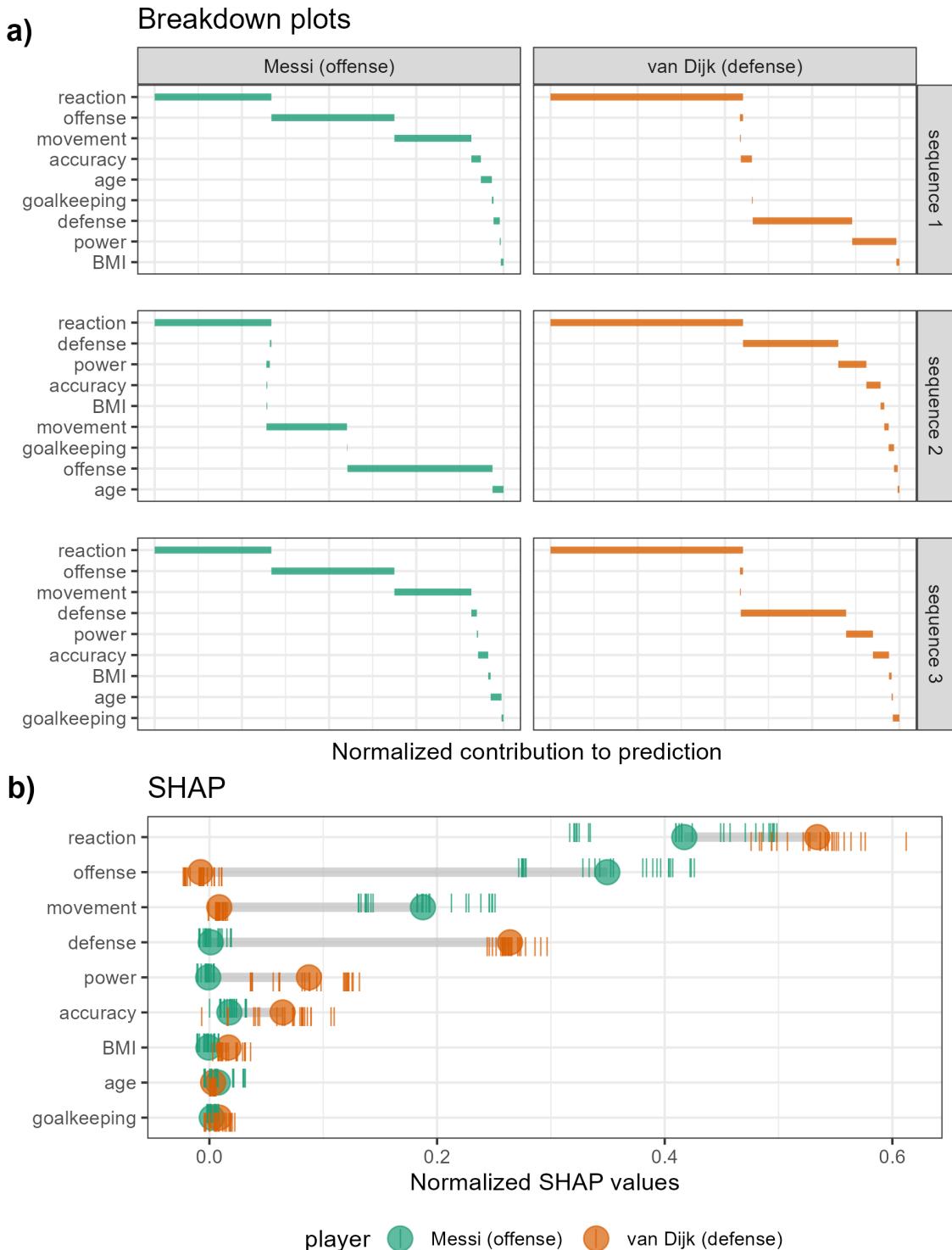


Figure 5.1: Illustration SHAP values for a random forest model for salaries of FIFA 2020 players based on nine predictors corresponding to different skills. A star offensive and defensive player are compared, L. Messi and V. van Dijk, respectively. Panel (a) shows breakdown plots of three sequences of 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 players. Offense and movement are important for Messi but not van Dijk, and conversely, defense and power are important for van Dijk but not Messi.

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.

5.3 Tours and the radial tour

A *tour* enables viewing of high-dimensional data by animating many linear projections with small incremental changes. It is achieved by following a path of linear projections (bases) of high-dimensional space. One of the features of the tour is the object permanence of the data points; one can track the relative change of instances in time and gain information about the relationships between points across multiple features. There are various types of tours that are distinguished by how the paths are generated (Lee et al., 2021; Cook et al., 2008).

The manual tour (Cook and Buja, 1997) defines its path by changing a selected feature's contribution to a basis to allow the feature to contribute more or less to the projection. The requirement constrains the contribution of all other features that a basis needs to be orthonormal (column correspond to vectors, with unit length, and orthogonal to each other). The manual tour is primarily used to assess the importance of a feature to structure visible in a projection. It also lends itself to pre-computation queued in advance or computed on-the-fly for human-in-the-loop analysis (Karwowski, 2006).

A version of the manual tour called a *radial tour* is implemented in Spyrisson and Cook (2020) and forms the basis of the new work. In a radial tour, the selected feature can change its magnitude of contribution but not its angle; it must move along the direction of its original contribution. The implementation allows for pre-computation and interactive re-calculation to focus on a different feature.

5.4 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. Figures 5.2 and 5.3 contain screenshots showing the cheem viewer for the two primary tasks: classification (categorical response) and regression (quantitative response).

5.4.1 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, which will have p values for each instance.

A visualization of these spaces is provided by the first two principal components of their respective spaces. In addition, a plot observed by predicted response is also provided (Fig. 5.2, b) A single 2D projection will not encompass all of the structure of higher-dimensional space, but it is generally a useful visual summary. 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 Fig. 5.2 a and c).

While the comparison of these spaces is interesting, the 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 in 2D spaces and long dashed and short dotted lines on 1D spaces.

5.4.2 Radial cheem tour

The global view facilitated the selection of instances. The attribution projection of the PI is the initial 1D basis in a radial tour. This approximation of the feature importance for the prediction of the instance best explains the difference between the data mean and an instance's prediction, not the local shape of the model surface.

Normalized attributions of the PI and CI are shown as dashed and dotted lines against the attribution distribution from all instances. These are depicted as a vertical parallel coordinate plot, where each line connects one instance’s feature attribution (Fig. 5.2, d). This is an important global context, the summarization of the attribution of all instances.

The current projection basis is depicted as the width of a bar, the feature contributions to the horizontal positions. The radial tour will vary one selected feature to a full contribution, zero contribution, and back to the attribution projection. Doing so tests a feature’s sensitivity to the structure identified by the local explanation. The default feature selected has the largest discrepancy between the attribution of the PI and CI. The data (scaled by standard deviation away from the mean) is projected through the current basis — the horizontal position of Fig. 5.2 e and Fig. 5.3 d.

The following sections elaborate on the difference in displays from applying this approach in classification and regression tasks. Now that we have introduced the global view and corresponding radial tour, let us discuss the differences between the classification and regression cases.

5.4.3 Classification task

Typically we select a misclassified instance compared to a correctly classified point nearby in data space. The model information in the global view depicts the model confusion matrix. The radial tour is 1D, with a density display, while the goal of the tour; exploring the sensitivity of each feature to structure identified by the local explanation, evaluating the support or robustness of the prediction.

5.4.4 Regression task

Structure resolved in the attribution space can be highlighted by coloring points on a statistic. For this purpose, we include residuals, log Mahalanobis distance of data space (a measure of outlyingness), and the correlation of the attribution projection with the observed response. In the radial tour, the horizontal positions are the same, the basis projection of the radial tour. The vertical position is fixed to the observed response feature and residuals in the middle and right panels. Correspondingly, the display changes from univariate

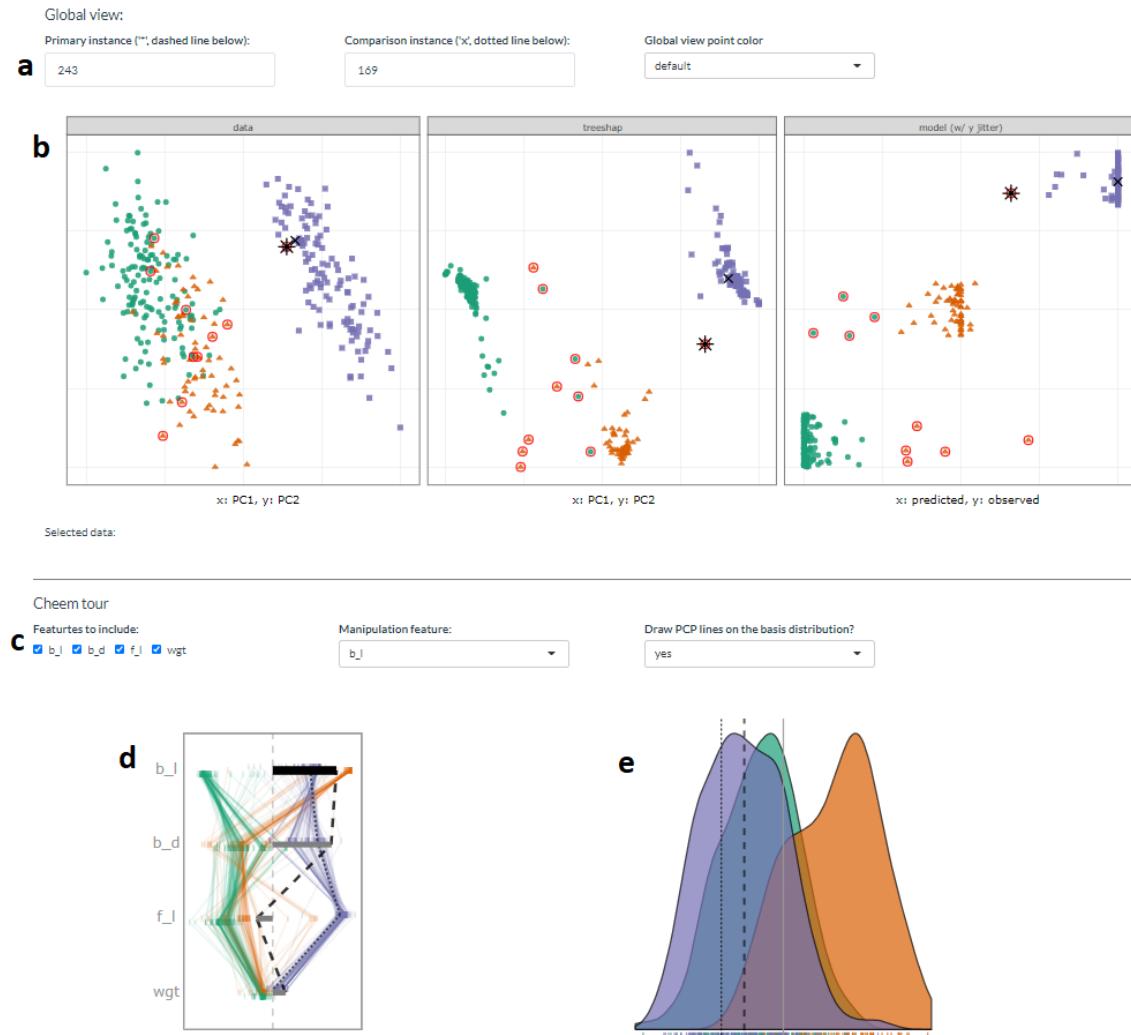


Figure 5.2: Overview of the cheem viewer for classification tasks. Global view inputs, (a), set the PI, CI, and color statistic. Global view, (b) PC1 by PC2 for data space, attribution space, and prediction by observed y (visual of the confusion matrix for classification). Points are colored by predicted class, and red circles indicate misclassified instances. Radial tour inputs (c) select features to include and which feature is changed in the tour. The visual depicting 1D basis (d) shows the distribution of the feature attribution, and bars show the current basis. The black bar is being varied. Panel (e) is the resulting projection of the data indicated as density in the classification case.

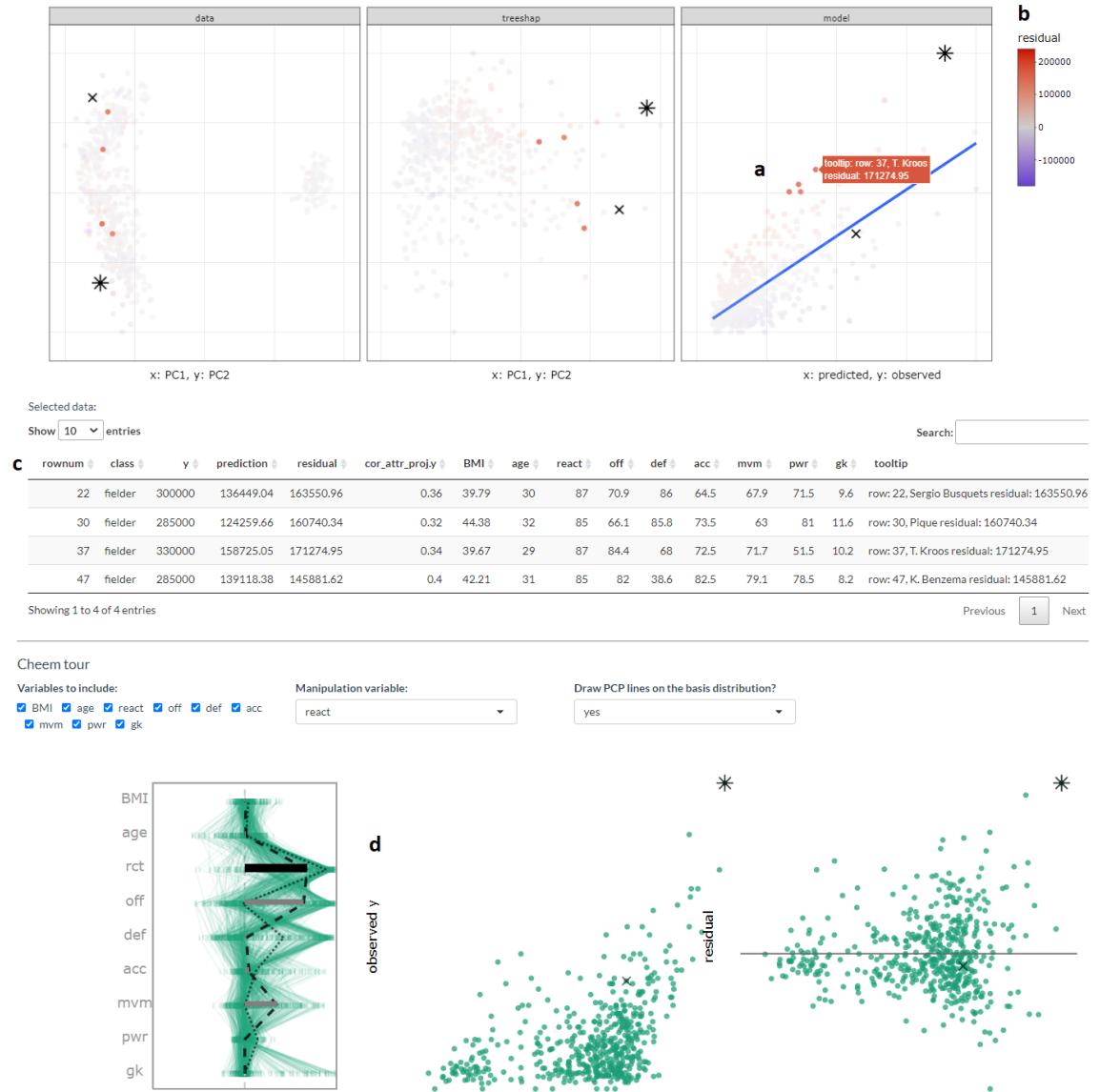


Figure 5.3: Overview of the cheem viewer for regression task highlighting the differences from the classification task and interactive features. Panel (a) shows linked brushing across the global view and the tooltip display when the cursor hovers over an instance. Coloring on a statistic (b) highlights structure organized in the attribution space. Interactive tabular display (d) populates when instances are selected. Regression projection (e) uses the same horizontal projection and fixes the vertical positions to the observed y and residuals, (left and right).

density to 2D scatterplot. The basis is still one component (horizontal) independent of the vertical position.

5.4.5 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.

5.4.6 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 of 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 over 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 calculated 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 with. 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.

5.4.7 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.

5.4.8 Installation and getting started

The following **R** code will help with getting up and running:

```
## Download the package
install.packages("cheem", dependencies = TRUE)

## Restart the R session, so the IDE has the correct directory structure
restartSession()

## Load cheem into session
library("cheem")

## Try the app
run_app()

# Processing your data

## Install treeshap from github to use as a local explainer
remotes::install_github('ModelOriented/treeshap')

## Follow the examples in cheem_ls()
?cheem_ls
```

5.5 Case studies

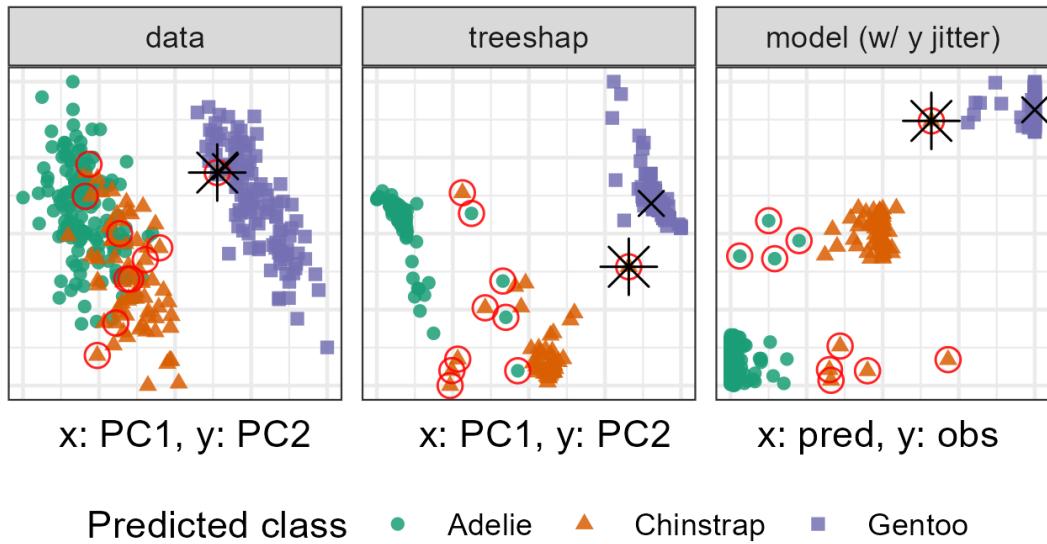
To illustrate the use of the cheem method, we apply it to modern datasets, two classification examples and then two of regression.

5.5.1 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_mm` (`b_l`), `bill_depth_mm` (`b_d`), `flipper_length_mm` (`f_l`), `body_mass_g` (`wgt`), 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

a Global view



b

Radial tour, select frames

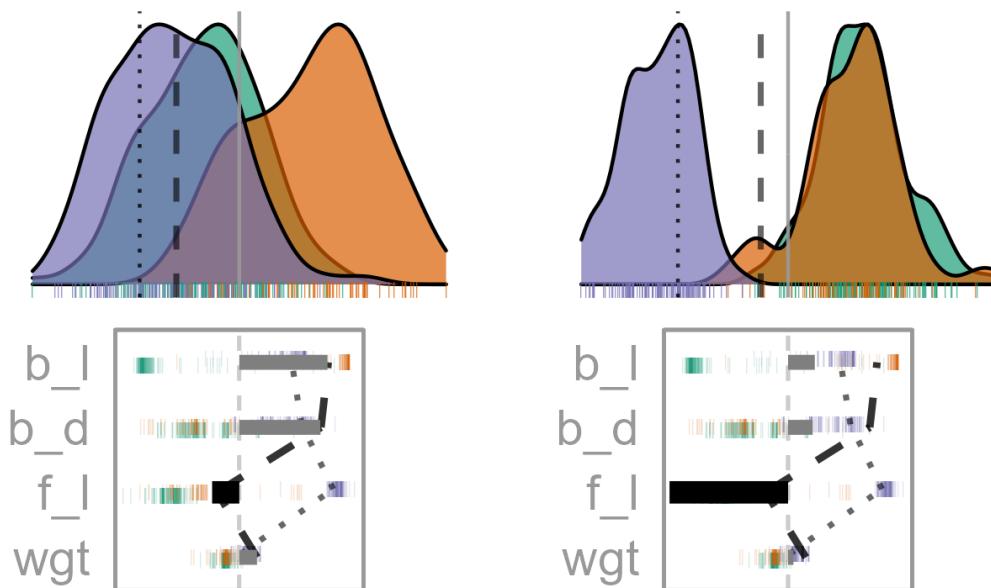


Figure 5.4: Examining the SHAP values for a random forest model classifying Palmer penguin species. The PI is an Chinstrap (orange) penguin that is misclassified as a Gentoo (purple), marked as an asterisk in (a), and the dashed vertical line in (b). The radial view shows varying the contribution of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

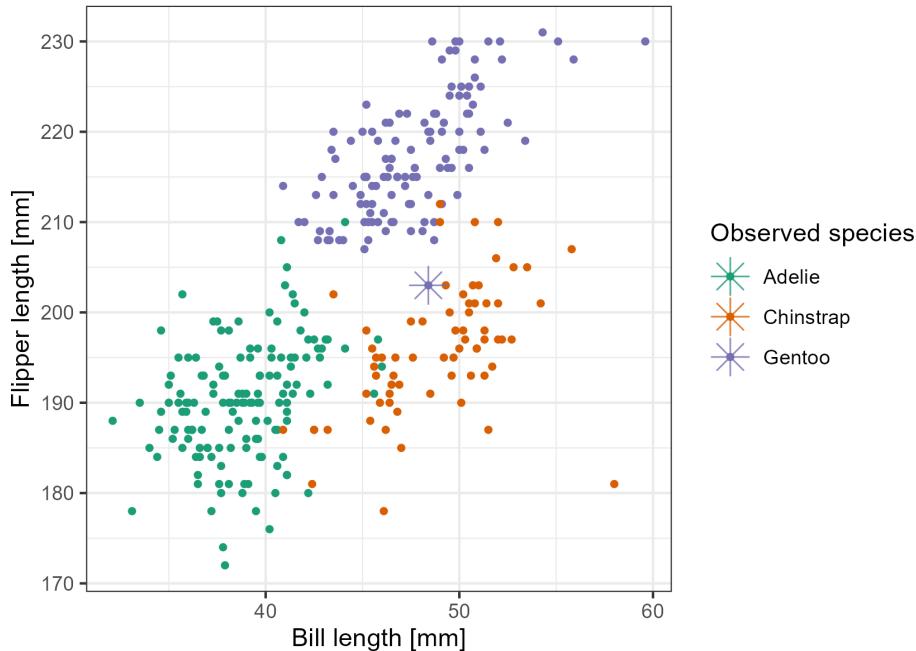


Figure 5.5: Checking what is learned from the cheem viewer. This is a plot of flipper length (f_l) and bill length (b_l), where an asterisk highlights the PI. A Gentoo (purple) misclassified as a Chinstrap (orange). The PI has an unusually small f_l length which is why it is confused with a Chinstrap.

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 as 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 b_1 and b_d with small contributions of f_1 and wgt . 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 (f_1). (In our exploration, this was the third feature explored. It is

typically useful to explore the features with larger contributions, here b_1 and b_d , but when doing this, nothing was revealed about how the PI differed from other Gentoos). On varying f_1 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_1 should be considered an important feature for explaining the (wrong) prediction.

Figure 5.5 confirms that flipper length (f_1) 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.

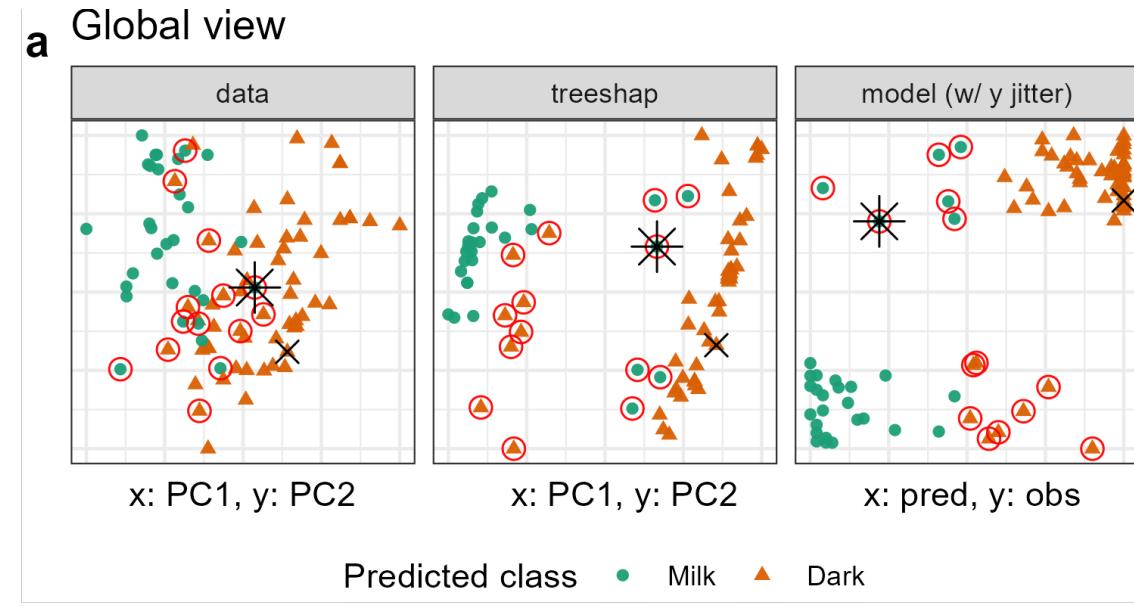
5.5.2 Chocolates, milk/dark chocolate classification

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 that is 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 SHAP PCA plots (a) show a big difference between the two chocolate types but with confusion for a handful of instances. The misclassifications are clearer in the observed vs predicted plot, and can be seen to be mistaken in both ways: milk to dark and dark to milk.

a Global view



b

Radial tour, select frames

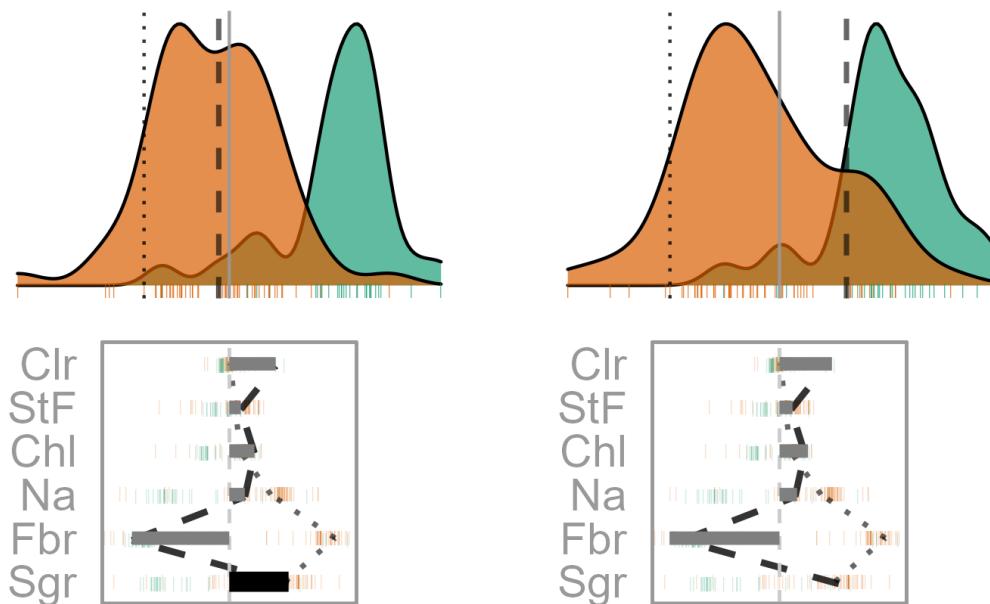


Figure 5.6: Chocolates data type classification (milk or dark). We select a chocolate labeled as dark though a random forest model predicts it to be milk chocolate from the values on the nutritional label. The attribution projection already looks more like a dark chocolate than milk. We remove four features with the lowest contribution for the selected instance and vary the contribution of Fiber. The misclassification seems improbable when sugar is near the max contribution. Animated tour can be found at vimeo.com/666431143.

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 the interpretation of 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 for milk chocolates is further to the right, indicating that they generally have low fiber and high sugar.

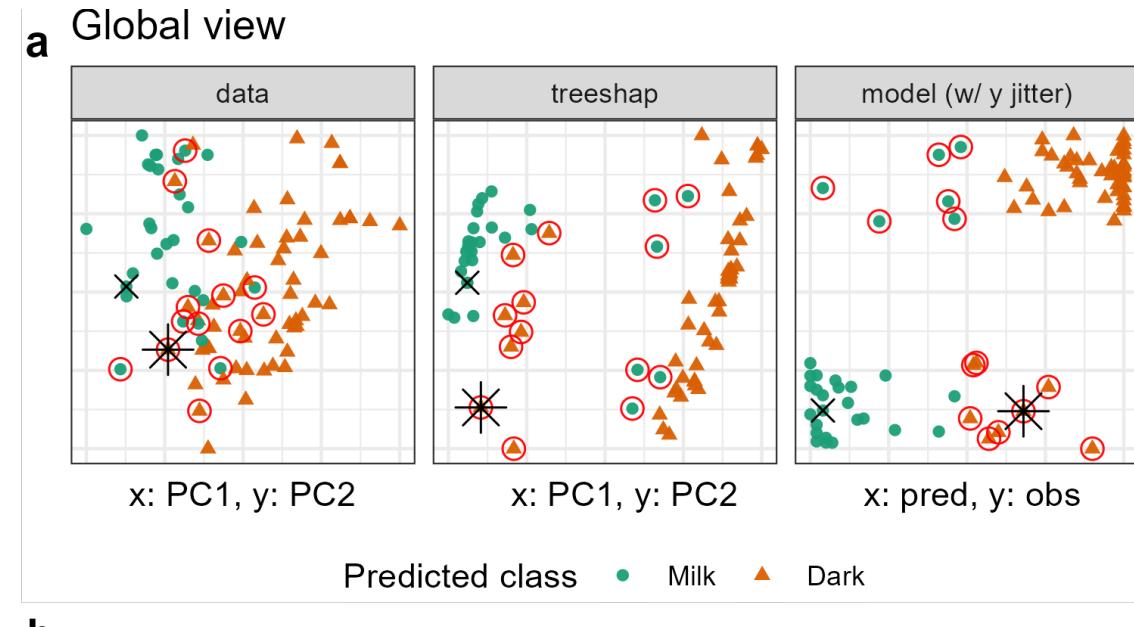
The instance of interest (dashed line) can be viewed against the comparison instance (dotted line). Now one needs to pay different attention to the parallel plot of the SHAP values, which are local to a particular instance, and the density plot, which is the same projection of all instances as specified 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, and completely 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; which features lead to a mild chocolate being misclassified as dark. 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. From the parallel coordinate lines, we identify discrepancies in Sodium and Fiber. We opt to vary Sodium, and find the attribution of the other milk chocolates not to support the prediction of a dark chocolate.

a Global view



b

Radial tour, select frames

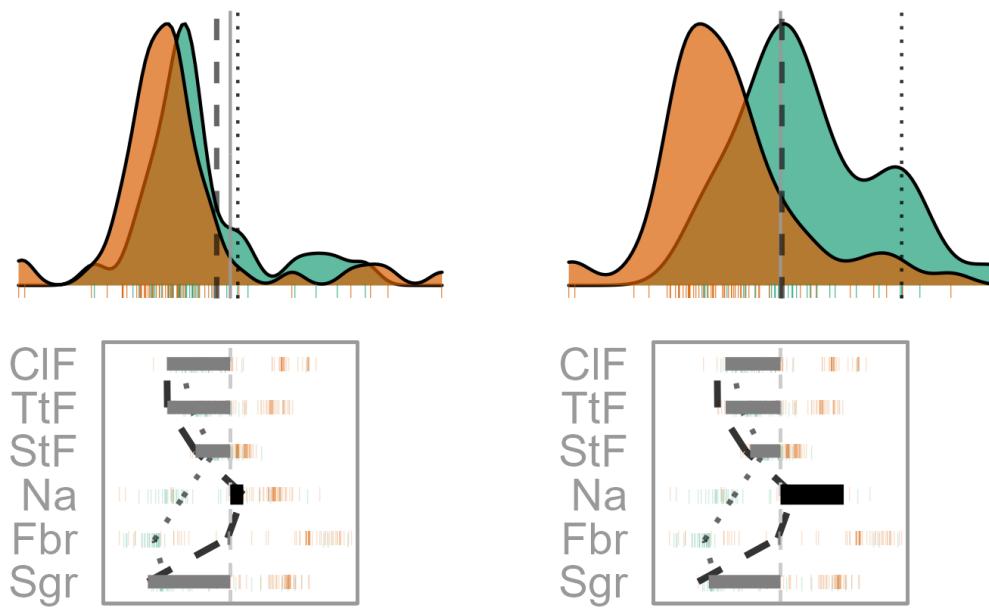


Figure 5.7: Chocolates data type classification (milk or dark). Looking at the inverse misclassification, we select a milk chocolate while the model predicts it to be dark chocolate. Sodium and Fiber (`Na` and `Fbr`) have the largest differences in attributed feature importance. We remove features with the lowest contributions and vary the contribution of Sodium. This misclassification is not supported when sodium has contribution close to the attribution aligned with the other milk chocolates. Animated tour can be found at vimeo.com/666431143.

5.5.3 FIFA, wage regression

The 2020 season FIFA data (Leone, 2020; Biecek, 2018) contains many skill measurements of soccer/football players and wage information. After aggregation of the skill measurements, we regress player wages [2020 euros] given just the skill aggregates. The model was fit from 5000 instances of the nine skill aggregates before being thinned to 500 players to mitigate occlusion and render time. 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.

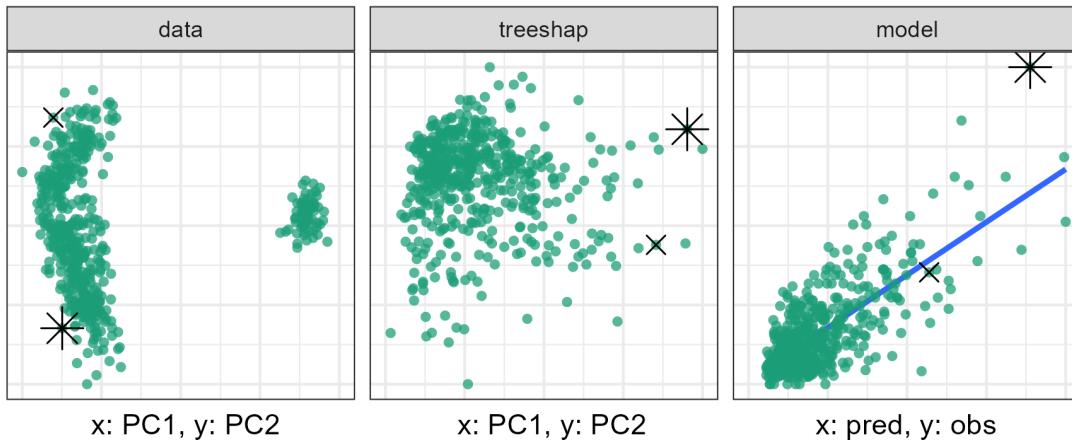
With figure 5.8, we will test the premise of the local explanation. While offensive and reaction skills (`off` and `rct`) are both crucial to explaining a star offensive player. As the contribution of defensive skills increases, Messi's is no longer separated from the group, and other defensive players are better predicted in this attribution case. In terms of what-if analysis, his predicted wages would be halved if Messi's tree SHAP attributions at these levels.

5.5.4 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 Build (`YrB`), Living Area (`LvA`), number of Bathrooms (`Bth`), number of Bedrooms (`Bdr`), total number of Rooms (`Rms`), Year the Garage was Build (`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 that has a large contribution from lot area. 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 an importance is living area is relatively uncommon.

a Global view



b Radial tour, select frames

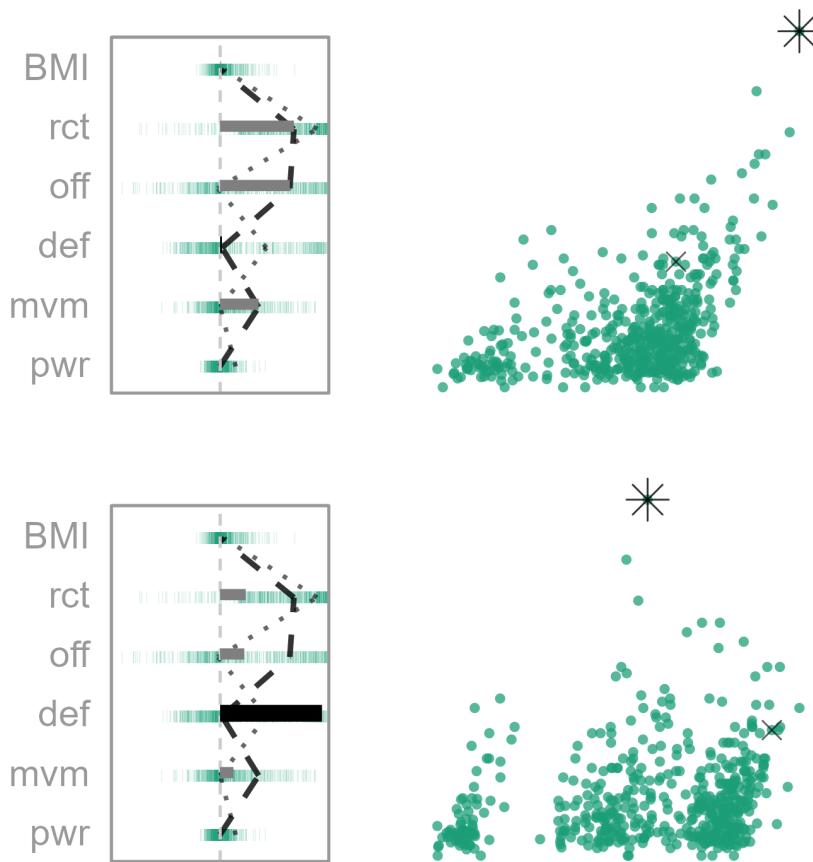
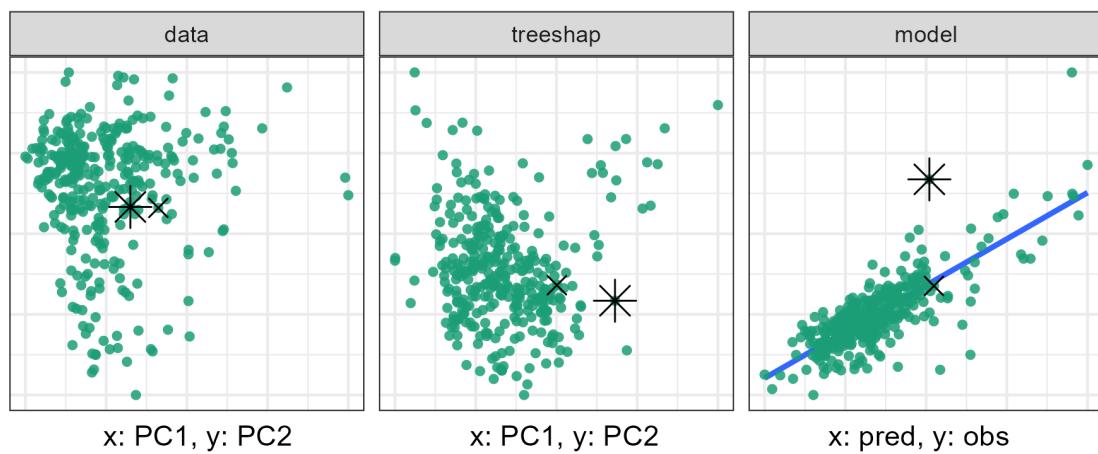


Figure 5.8: FIFA 2020 data, a random forest model, regresses wages [2020 Euros] from nine aggregated skill measurements. The PI is a star offensive player (L. Messi) compared with a top defensive player (V. van Dijk). We remove three features with low attribution from both players. The attribution projection 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. The animate radial tour can be found at vimeo.com/666431163.

a Global view



b Radial tour, select frames

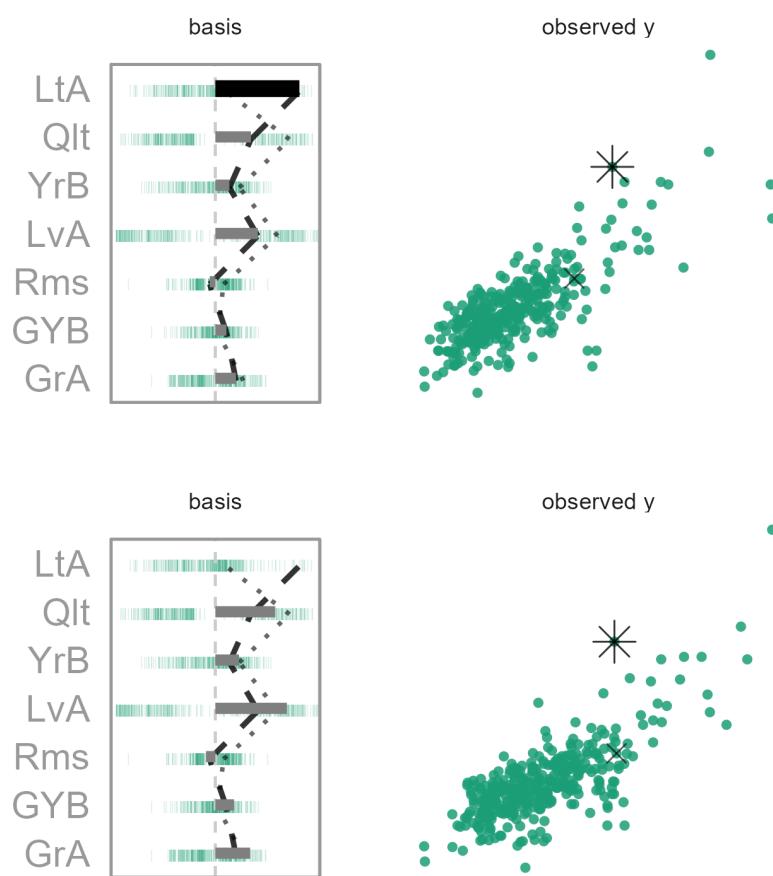


Figure 5.9: Ames housing 2018 regressing sales price [USD]. The PI sale price was under predicted and had sizable attribution to lot area (*LtA*). The CI was predicted sales price was similar and much more accurate with its observed sales price while it has very little attribution to lot area. Varying the contribution lot area the separation between these house sales crosses when there is a low contribution of *LtA*, which is important to explaining the PI and near invariant to the sales price of the CI. The corresponding animation is at vimeo.com/666431134.

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.

5.6 Discussion

The need to maintain the interpretability of black-box models is evident. One aspect uses local explanations of the model in the vicinity of an instance. Local explanations approximate the linear feature importance to the model. Our contribution is to assess explanations by examining the support by varying the contributions with a radial tour. First, a global view visualizes approximations of the data space, explanation space, model predictions side-by-side, using dynamic interaction to compare and contrast and identify instances of interest. The normalized linear importance from the explanation of the PI becomes the feature of interest to further explore with the radial tour. The tours explore the feature sensitivity to the structure identified in the explanation.

We have illustrated this method on random forest models using the tree SHAP local explanation, while it could be generally used with any compatible model-explanation pairing. We apply it to the classification and regression tasks. We have created an open-source **R** package **cheem**, available on [CRAN](#), to facilitate preprocessing and exploration with the described interactive application. Toy and real data are provided, or upload your data after preprocessing.

5.7 Acknowledgments

We would like to thank Professor Przemyslaw Biecek for his input early in the project and to the broader MI² lab group for the **DALEX** ecosystem of **R** and **Python** packages. This research was supported by Australian Government Research Training Program (RTP) scholarships. Thanks to Jieyang Chong for helping proofread this article.

The namesake, Cheem, refers to a fictional race of humanoid trees from Doctor Who lore. **DALEX** pulls on from that universe, and we initially apply tree SHAP explanations specific to tree-based models.

5.8 References

Chapter 6

Conclusion

We know that visualizing data is more robust than numerical summarization alone. It provides a means to get a feel for the data rapidly, check for erroneous values, and confirm model assumptions. But visualizing data spaces becomes increasingly complex as dimensionality increases. Static linear dimension reduction has been popular in extending the space viewed. Dynamic animations of linear projections, tours, further increase the perceptible information of linearly reduced spaces. Previously we have not had a means to test the sensitivity of individual variables' contribution to the structure in one embedding. Chapter 3 introduced the **spinifex** package, which facilitates user-controlled manual tours and layered composition and exporting animations of any tour, interoperably with **tourr**. The radial, manual tour does sizably improve the accuracy on variable-level supervised cluster separation task compared with PCA and the grand tour. Chapter 4 covers the within-participants user study that led to this conclusion. Finally, I extended use of the manual tour to increase the interpretability of non-linear models by using it to explore their local explanations in the package **cheem** is discussed in Chapter 5.

6.1 Contributions

This study makes two major contributions to extending knowledge in the domain of data visualization.

C1 The first contribution of this study is a software development package named **spinifex_**. This package facilitates the preprocess transformations with **scale_*** functions, identifies features with **basis_*** functions, and allows an analyst to control variable contributions with manual tours. It creates a framework for layered creation of tour visuals with **proto_*** functions that will feel at home to **ggplot2** users. A new key element of this package is the manual tours, and those created from **tourr** can be rendered and exported as interactive HTML widgets, .gif, or .mp4 files. Vignettes and interactive application help users rapidly understand the concepts and scope of the package. The impact of **spinifex** can be seen in two ways. My contributions to it and **tourr** won the ACEMS Impact and Engagement Award, 2018. Further, 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.

C2 The second contribution of the study is the **cheem** package. I suggest using manual tours to extend the interpretability of black-box models by exploring their local explanations. A workflow is developed to facilitate the creation of a random forest model, calculating tree SHAP local explanations for each observation. An interactive application illustrates the purposed visual and analysis from several datasets or upload your data after processing. Local explanations are evaluated by testing the variables' sensitivity to the structure identified in the explanation. **cheem** was recently uploaded to [CRAN](#) and has a corresponding [pkgdown](#) site.

6.2 Summary statement

Data visualization is crucial for EDA and conveys more information than numeric summarization alone. Yet as the dimensionality of data space increases, data visualization quickly becomes complex. At some point, dimension reduction will become necessary for visualizing multivariate spaces. Data visualization tours are a class of linear projections animated over small changes in the basis. The manual tour uniquely offers an analyst control over the contribution in the basis.

This thesis builds on the visualization of multivariate spaces and, particularly, user control affecting the variable contributions to linear bases. I have created a an open-source package

that facilitates the creation of manual tours and layered display compositions of any tour path interoperably with existing packages. I conduct a user study showing strong support that giving analysts control with the radial tour leads to a sizable increase in variable attribution to the separation of two clusters. Lastly, I introduce a new analysis to extend black-box models' interpretability. This analysis explores the observation-specific variable attribution of local explanation of the model. By varying the contributions of these normalized attributions, we can explore the variable-level support of the explanations to see under what conditions they make sense and where they lack evidence. I create another open-source package to preprocess, visualize, and interactively perform this analysis.

6.3 Future work

In addition to controlling the contribution of a single variable, it may be insightful to be able to change the contributions of several variables at once (manipulating a linear combination). A sort of dimension reduction tour that would append several manual tours together, zeroing the contributions of variables one at a time may prove interesting as the number of embedded dimensions approaches a features intrinsic dimensionality.

There are direct natural extensions to **spinifex**, such as extending the type of protos 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). An additional interpolation of the manual tour could correlate the frame number with the magnitude of a radial tour; the first frame would contain zero magnitude, and the last contains a full contribution, while the default starting frame would be the original contribution.

Experiencing tours as 3D scatterplots in extended reality with stereoscopically true head tracking may be fruitful. Nelson, Cook, and Cruz-Neira (1998) explore 2D tour 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.

Similarly, **cheem** analysis could be generalized to a broader range of models and local explanations. The models and local explanations facilitated by **DALEX**::**explain()** seems to be a good direction to extend (Biecek, 2018; Biecek and Burzykowski, 2021). Alternatively, other statistics may better show the structure identified by explanations that could be added.

Bibliography

- Adadi, A and M Berrada (2018). Peeking inside the black-box: a survey on explainable artificial intelligence (XAI). *IEEE access* **6**. Publisher: IEEE, 52138–52160.
- Allaire, JJ, Y Xie, J McPherson, J Luraschi, K Ushey, A Atkins, H Wickham, J Cheng, W Chang, and R Iannone (2020). *rmarkdown: Dynamic Documents for R*. <https://github.com/rstudio/rmarkdown>.
- Anscombe, FJ (1973). Graphs in Statistical Analysis. *The American Statistician* **27**(1), 17–21.
- 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**. Publisher: Elsevier, 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 Spyris (2020). Changes to Victorian Intraday Electricity Demand Following COVID-19 Restrictions. In: *Melbourne Datathon 2020, Insights category*, pp.4. <https://www.overleaf.com/project/5f614799515ac0000119daf7>.
- 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). Publisher: IEEE, 2203–2212.
- Biecek, P (2018). DALEX: explainers for complex predictive models in R. *The Journal of Machine Learning Research* **19**(1). Publisher: JMLR.org, 3245–3249.
- Biecek, P (2020). *ceterisParibus: Ceteris Paribus Profiles*. <https://CRAN.R-project.org/package=ceterisParibus>.

BIBLIOGRAPHY

- Biecek, P and T Burzykowski (2021). *Explanatory Model Analysis: Explore, Explain, and Examine Predictive Models*. en. Google-Books-ID: rq4SEAAAQBAJ. CRC Press.
- Breiman, L (2001). Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical science* **16**(3). Publisher: Institute of Mathematical Statistics, 199–231.
- Buja, A and D Asimov (1986). Grand Tour Methods: An Outline. In: *Proceedings of the Seventeenth Symposium on the Interface of Computer Sciences and Statistics on Computer Science and Statistics*. New York, NY, USA: Elsevier North-Holland, Inc., pp.63–67. <http://dl.acm.org/citation.cfm?id=26036.26046>.
- 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>.
- Cattell, RB (1966). The scree test for the number of factors. *Multivariate behavioral research* **1**(2). Publisher: Taylor & Francis, 245–276.
- Chambers, J, W Cleveland, B Kleiner, and P Tukey (1983). Graphical Methods for Data Analysis. eng.
- Chang, W, J Cheng, JJ Allaire, Y Xie, and J McPherson (2020). *shiny: Web Application Framework for R*. <https://CRAN.R-project.org/package=shiny>.
- 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*. R package version 1.7.0. <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.
- 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, 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.

BIBLIOGRAPHY

- Cook, D, U Laa, and G Valencia (2018). Dynamical projections for the visualization of PDFSense data. *Eur. Phys. J. C* **78**(9), 742.
- da Silva, N, D Cook, and EK Lee (2021). A Projection Pursuit Forest Algorithm for Supervised Classification. *Journal of Computational and Graphical Statistics*. Publisher: Taylor & Francis, 1–21.
- 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.
- Duffy, C (2019). Apple co-founder Steve Wozniak says Apple Card discriminated against his wife. *CNN*.
- Gabriel, KR (1971). The biplot graphic display of matrices with application to principal component analysis. *Biometrika* **58**(3), 453–467.
- 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). Publisher: Public Library of Science San Francisco, USA, 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.
- Horst, AM, AP Hill, and KB Gorman (2020). *palmerpenguins: Palmer Archipelago (Antarctica) penguin data*. <https://allisonhorst.github.io/palmerpenguins/>.
- Hurley, C and A Buja (1990). Analyzing High-Dimensional Data with Motion Graphics. *SIAM Journal on Scientific and Statistical Computing* **11**(6), 1193–1211.

BIBLIOGRAPHY

- 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.
- 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*. original-date: 2020-07-30T19:38:33Z. <https://github.com/ModelOriented/treeshap>.
- 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 Spyrlis, E Wang, and HS Zhang (2021). The state-of-the-art on tours for dynamic visualization of high-dimensional data. en. *WIREs Computational Statistics*. _eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/wics.1573>, e1573.
- Lee, YD, D Cook, Jw Park, and EK Lee (2013). PPtree: Projection pursuit classification tree. *Electronic Journal of Statistics* **7**. Publisher: Institute of Mathematical Statistics and Bernoulli Society, 1369–1386.
- Leone, S (2020). *FIFA 20 complete player dataset*. en. <https://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). Conference Name: IEEE Transactions on Visualization and Computer Graphics, 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*.

BIBLIOGRAPHY

- 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 and G Hinton (2008). Visualizing Data Using t-SNE. *Journal of machine learning research* **9**(Nov), 2579–2605.
- 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/.
- Munzner, T (2014). *Visualization analysis and design*. AK Peters/CRC Press.
- Nelson, L, D Cook, and C Cruz-Neira (1998). XGobi vs the C2: Results of an Experiment Comparing Data Visualization in a 3-D Immersive Virtual Reality Environment with a 2-D Workstation Display. en. *Computational Statistics* **14**(1), 39–52.
- O’Neil, C (2016). *Weapons of math destruction: How big data increases inequality and threatens democracy*. Crown.
- Ocagne, Md (1885). *Coordonne’s 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’es 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**. Publisher: Elsevier, 22–27.
- 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>.
- R Core Team (2020). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. <https://www.R-project.org/>.

- 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>.
- 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). Publisher: Institute of Mathematical Statistics, 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.
- 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.
- Spryripon, 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.

BIBLIOGRAPHY

- Spyrison, 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*. type: article.
- Strumbelj, E and I Kononenko (2010). An efficient explanation of individual classifications using game theory. *The Journal of Machine Learning Research* **11**. Publisher: JMLR.org, 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.
- Wagner Filho, J, M Rey, C Freitas, and L Nedel (2018). Immersive Visualization of Abstract Information: An Evaluation on Dimensionally-Reduced Data Scatterplots. In:
- Wang, BT, TJ Hobbs, S Doyle, J Gao, TJ Hou, PM Nadolsky, and FI Olness (2018). 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'Reilly 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).

BIBLIOGRAPHY

- Xie, Y (2016). *bookdown: Authoring Books and Technical Documents with R Markdown*. Boca Raton, Florida: Chapman and Hall/CRC. <https://github.com/rstudio/bookdown>.
- Xie, Y (2020). *knitr: A General-Purpose Package for Dynamic Report Generation in R*. <https://yihui.org/knitr/>.
- Xie, Y, JJ Allaire, and G Grolemund (2018). *R Markdown: The Definitive Guide*. Boca Raton, Florida: Chapman and Hall/CRC. <https://bookdown.org/yihui/rmarkdown>.
- Yanai, I and M Lercher (2020). A hypothesis is a liability. *Genome Biology* **21**(1), 231.
- 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>.