Penguins Go Parallel: a grammar of graphics framework for generalized parallel coordinate plots

Author 1[1*] Author 2[2] Author 3[3]

[1] University 1; [2] University 2; [3] University 3

September 22, 2022

Abstract

Parallel coordinate plots (PCP) are a useful tool in exploratory data analysis of high-dimensional numerical data. The use of PCPs is limited when working with categorical variables or a mix of categorical and continuous variables. In this paper, we propose generalized parallel coordinate plots (GPCP) to extend the ability of PCPs from just numeric variables to dealing seamlessly with a mix of categorical and numeric variables in a single plot. In this process we find that existing solutions for categorical values only, such as hammock plots or parsets become edge cases in the new framework. By focusing on individual observations rather than a marginal frequency we gain additional flexibility. The resulting approach is implemented in the R package ggpcp.

Contents

*Corresponding author: xxx

A **Example usage** **21**

Broad Comments:

- Replace fleas with penguins
- Make colors consistent - gentoo are always ___ color - but with a different (but consistent) color scheme in the clustering example.
- Leave out missing data until it's explicitly discussed in the example with labeled ? points

1 Introduction

Few approaches in data visualization exist that are truly high-dimensional. Most visualizations are projections of data into two or three dimensions enhanced by facetting or additional mappings to plot aesthetics, such as point size and color. Parallel coordinate plots are one of the exceptions: in parallel coordinate plots we can actually visualize an arbitrary many number of variables to get a visual summary of a high-dimensional data set. In a parallel coordinate plot, each variable takes the role of a vertical (or parallel) axis; giving the visualization its name. Multivariate observations are then plotted by connecting their respective values on each axis across all axes using polylines (cf. Figure 1). For just two variables this switch from orthogonal axes to parallel axes is equivalent to a switch from the familiar Euclidean geometry to the Projective Space. In the projective space, points take the role of lines, while lines are replaced by points, i.e. points falling on a line in the Euclidean space correspond to lines crossing in a single point in the Projective Space. This duality provides a good basis for interpreting geometric features observed in a parallel coordinate plot [Inselberg, 1985].

The origins of parallel coordinate plots date back to the 19th century and are, depending on the source, either attributed to d'Ocagne [1885] or Gannett [1880]. Modern era parallel coordinate plots go back to Inselberg [1985] and Wegman [1990]. Parallel coordinate plots are used in an exploratory setting as a way to get a high-level overview of the marginal distributions involved, to identify outliers in the data, and to find potential clusters of points. In the absence of those, Parallel Coordinate Plots are often criticized for the amount of clutter they produce, resembling a game of mikado (also known as pickup-sticks – if you are not familiar with the game, imagine spilling a box of spaghetti) rather than organized data. This clutter is sometimes mitigated by the use of α-blending [Miller and Wegman, 1991], density estimation [Heinrich and Weiskopf, 2009], or edge-bundling parallel coordinate plots [McDonnell and Mueller, 2008]. For a detailed overview of these and other techniques see Heinrich and Weiskopf [2013].

While parallel coordinate plots have some shortcomings, the greatest challenge is in using categorical variables alongside quantitative variables. In current solutions, levels of categorical variables are transformed to numbers and variables are then used as if they were numeric. This introduces ties into the data, and the resulting parallel coordinate plot becomes uninformative, as it only shows lines from each level of one variable to all levels of the next variable. Some versions of parallel coordinate plots have been specifically developed to deal with categorical data: parallel set plots [Kosara et al., 2006], Hammock plots [Schonlau, 2003], and common angle plots [Hofmann and Vendettuoli, 2013], but these solutions do not accommodate quantitative variables. Instead, they are intended for use with tabularized data and show bands of observations from one categorical variable to the next. Hammock plots and common angle plots mitigate effects of the sine-illusion [Day and Stecher, 1991, VanderPlas and Hofmann, 2015] on parallel sets plots. An attempt to combine categorical and numeric variables in a parallel coordinate plot is introduced in the categorical parallel coordinate plots of Pilhöfer and Unwin [2013] by treating factor variables as numeric. Similar to parallel sets, this approach is also based on marginal frequencies for the categorical variables. Categorical parallel coordinate plots are the closest of these variations to our solution, but the `extracat` package has not been updated recently and is no longer on CRAN.

The remainder of the paper is organized as follows: section 2 introduces the `ggpcp` syntax and explains the improvements in `ggpcp` over other parallel coordinate plot software packages. section 3 describes the data processing for parallel coordinate plots and how this wrangling is separated from the plot rendering in `ggpcp`. section 4 discusses the rendering of parallel coordinate plots and factors such as plotting order

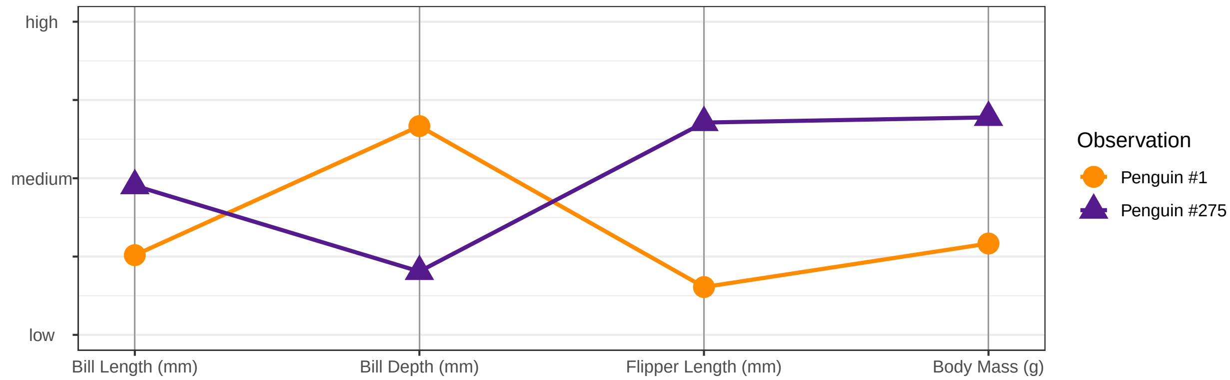


Figure 1: *Sketch of a parallel coordinate plot of two observations in four dimensions. Each dimension is shown as a vertical axis, observations are connected by polylines from one axis to the next.Two penguins from the Palmer Penguin data set (see section 5) were sampled for this example.*

and tie-breaking which are important for the design of PCPs. **??** provides three examples which highlight the use of generalized PCPs in exploratory settings.

2 Motivation and Package Usage

XXX Gestalt motivation of what makes this package unique - ability to follow a single data point through the parallel coordinate plot.XXX

```
pcp <- penguins %>%                         # data management:
  pcp_select(species, 3:6, sex) %>%         #   variable selection, see section 3.1
  pcp_scale(method="uniminmax") %>%         #   setting scaling method, see section 3.2
  pcp_arrange() %>%                         #   arranging categorical data
  ggplot(aes_pcp()) +                       # plotting the chart:
    geom_pcp_axes() +                       #   vertical lines for axes
    geom_pcp(aes(colour = species))         #   parallel coordinate line segments
```

XXX Make the code into a subfigure and put alongside figure 1. Switch to penguins data, select 2 penguins from the set. XXX

By leveraging the full ggplot2 philosophy instead of using highly specific wrapper functions, `ggpcp` allows users to focus on the data, rather than the names of various parameters used for customization. `ggpcp` adopts tidy conventions for data wrangling, separating the necessary data manipulation to generate a parallel coordinate plot from the visual rendering. Scaling, ordering of cases, and the arrangement of the parallel axes are completed using `pcp_select`, `pcp_arrange`, and `pcp_scale`,respectively; the resulting data frame is then passed directly into the familiar `ggplot()` call. During the plotting state, the only modification from default `ggplot2` syntax is the use of `aes_pcp()` in place of `aes()`; this is necessary to handle the multiple axes in a parallel coordinate plot while maintaining the ability to map all other variables of the original data frame to aesthetics such as linetype and color. The user has complete control over layers such as PCP lines (`geom_pcp`), labels (`geom_pcp_labels`), and boxes around categorical variables (`geom_pcp_boxes`), but there are additional advantages to the use of `ggplot2`. Users can also augment their parallel coordinate plots with additional information, such as boxplots or violin plots, with standard ggplot2 syntax.

One of the strengths of `ggplot2` is its handling of small multiple plots with `facet_grid` and `facet_wrap`; these functions are fully supported in `ggpcp`.

In addition, basing ggpcp on ggplot2 expands the functionality available to users without much additional code, thanks to other packages such as plotly [Sievert, 2020] and gridsvg [Murrell and Potter, 2020] which leverage ggplot2 to create interactive graphics for the web.

3 Data management

One of the ideas behind this re-implementation of parallel coordinate plots is to expose parallel coordinate plots at a functional level. Rather than using a single function with parameters controlling every aspect, we separate the data management from the visual rendering. In particular, we separate out the data management into three parts:

1. Variable selection and reshaping data,
2. Scaling of axes, both at the individual level and in the relationship of the axes to each other, and
3. Treatment of ties in categorical axes.

XXX Cite subfigure above with code XXX

The modularization of the data wrangling process has the additional advantage to lay out the necessary elements in successive steps. Some of these steps are even optional - e.g. scaling variables might not be necessary, if all variables are already on the same scale (i.e. method 'raw' in GGally); similarly, using pcp_arrange to break ties is only necessary if there are any factor variables, and if we actively want to spread these observations out. In addition, by exposing these elements of the pcp data wrangling process, we allow users to create additional functions for handling these tasks.

The treatment of ties is an aspect not generally addressed in the original parallel coordinate plots of Inselberg [1985] and Wegman [1990]. We have found a need to deal with ties, because ties are visually the main obstacle of allowing the viewer to follow an observation from axis to axis through the high-dimensional space. If we can track a single observation through the high-dimensional space, we have the ability to look beyond two-variable comparisons (sequential axes). This allows users to more easily summarize main trends and identify observations which do not follow those trends. When ties are not handled appropriately and users cannot follow individual observations, higher-dimensional insights are next to impossible.

3.1 Variable selection and Order of the variables

One of the biggest strengths of the Grammar of Graphics is its mapping between data variables and visual aesthetics. In standard plots any mapping is a function between one aesthetic and one data variable. In a parallel coordinate plot, this one-to-one mapping between data and plot aesthetics is seemingly turned into a one-to-many mapping between arbitrarily many data variables to the x axis. By transforming the wide form of the data set into a long form [Wickham, 2007, Wickham et al., 2021, Wickham, 2014, 2021], we get to a form of the dataset in which we achieve a one-to-one mapping to a now discrete x axis consisting of the (names of the) original data variables.

From the user's perspective this data reshaping has purely the form of a data selection, while the data wrangling is going on behind the scenes in `pcp_select`.

`pcp_select(data, ...)` allows a selection of variables to be included in the parallel coordinate plot. Variables can be specified by an any combination of the following methods:

- position, e.g. `1:4, 7, 5, 4,`
- name, e.g. `class, age, sex, aede1:aede3` or
- using pattern selectors, e.g. `starts_with("aede")`, see `?tidyselect::select_helpers`

Variables can be selected multiple times and will then be included in the data and the resulting plot multiple times. Note that the order in which variables are selected determines the order in which the

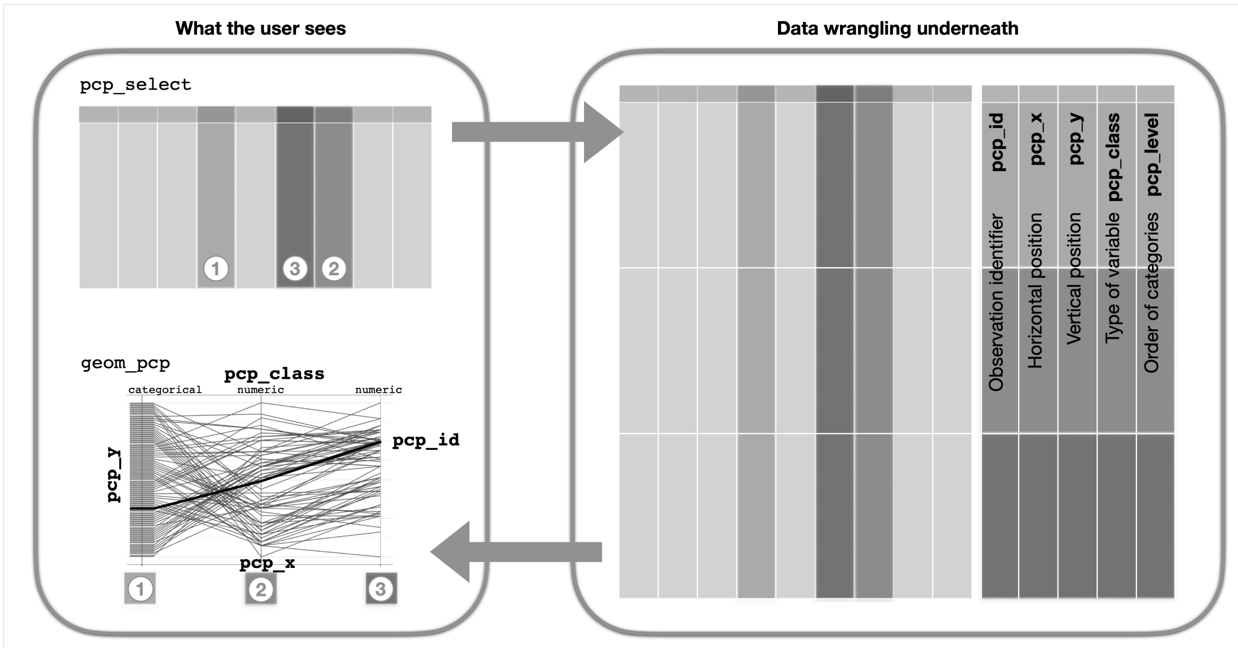


Figure 2: *The user selects a set of three variables (top left). On the right, an overview of the data wrangling step before a parallel coordinate plot can be drawn (bottom left). Note that the order in which variables are selected is reflected in the order in which variables are included in the parallel coordinate plot.*

corresponding axis is drawn in the parallel coordinate plots. `pcp_select` transforms the selected variables to long form and embellishes the data set with a number of additional variables. All of the newly created and added variables start with the prefix `pcp_`:

- `pcp_id`: integer variable identifying each observation in the original dataset. This variable is used as the grouping variable to identify which values should be connected by a line segment in the parallel coordinate plot.
- `pcp_x`: discrete variable consisting of the names of the selected variables in the order that they were selected - this is the order in which the variables will be included in the plot.
- `pcp_y`: numeric variable containing the values of all of the selected variables. In case a selected variable is not numeric, it is converted to a factor variable and the (numeric) factor levels are saved in `pcp_y`.
- `pcp_class`: character variable containing the class information of a selected variable.
- `pcp_level`: character variable containing the factor levels of selected data variables. In case of numeric variables, the data values are stored (in textual form). The ordering of factor variables will be discussed below but is implemented using this added variable.

As a consequence of these design decisions, users have several ways to perform different tasks within the flow of generating data for a parallel coordinate plot. For instance, users can reorder variables using `pcp_select` or after variable selection using the `pcp_x` variable.

XXX univariate transformations, such as reversing an axis for a variable can be done using a mutate statement before the variable selection. example? changing order of levels on a categorical variable or a log transform or square root transformation work the same.

```
flea %>%
  mutate(
    species_inv = factor(species, levels = rev(levels(.$species))),
    head_log = log(head)) %>%
```

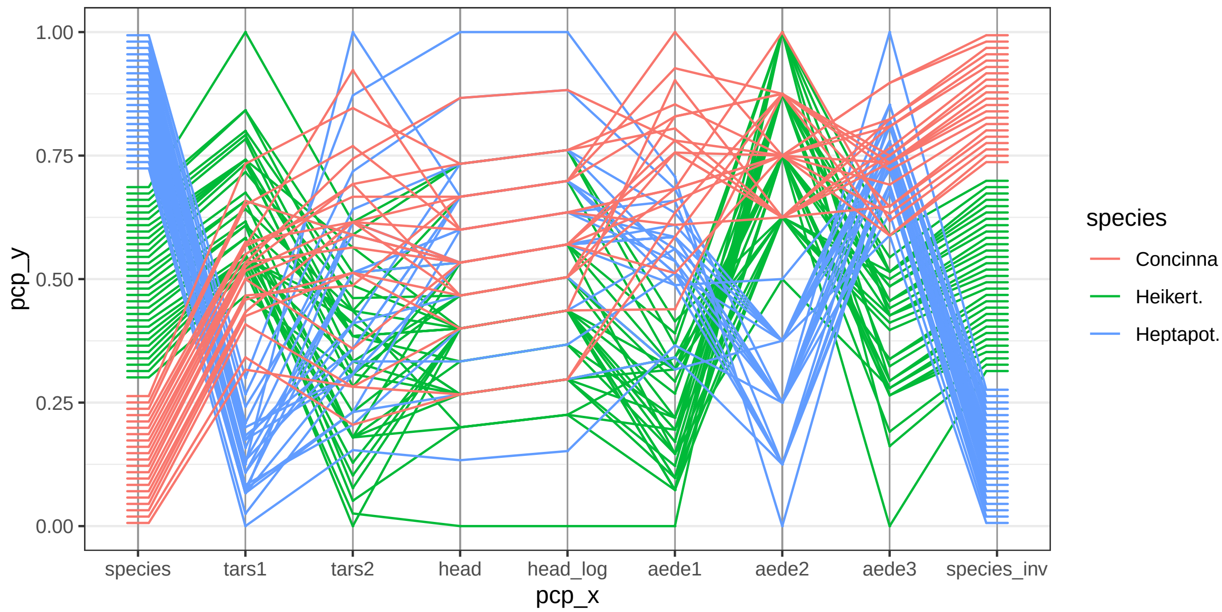


Figure 3: *Mutations before the `pcp_select` stage can be used to reorder factors or transform numerical variables.*

```
pcp_select(species, 2:4, head_log, 5:7, species_inv) %>%
pcp_scale(method="uniminmax") %>%
pcp_arrange() %>%
ggplot(aes_pcp()) +
  geom_pcp_axes() +
  geom_pcp(aes(colour = species))
```

Ordering of levels in factor variables (XXX not sure where to put this, yet): What we are doing with the ordering of levels in factor variables, is to stick with the basic interpretation of factor variables as a type of variable that has both labels and an ordering of those labels. Whenever we assign a numeric value to the ordering, we refer to the associated score, which is an integer value from one to the number of categories, if not specified explicitly otherwise. This means in particular, that the first level of a categorical variable is mapped to the lowest value along the y axis rather than the 'top' value as e.g. when mapped to 'fill' in a barchart. This might lead to a visual inconsistency between the orderings in the levels of a categorical variable and an accompanying color legend. In those situations we suggest to reverse the order in the legend by using the command `guides(color = guide_legend(reverse=TRUE))` as shown in the example in subsection 5.2.

3.2 Scaling

`pcp_scale(data, method)` scales the values on each axis and determines the relative relationship of the axes to each other.

`method` is a character string specifying the method to be used when transforming the values of each variable into a common y axis. By default, the method `uniminmax` is chosen, which univariately scales each variable into a range of [0,1] with a minimum at 0 and the maximum at 1. `globalminmax` maps the values across all axes into a an interval of [0,1]. This method should only be used if the values across all variables are comparable. The method `robust` normalizes values univariately by mapping the median value to 0.5 and a robust 95% confidence interval (based on the median absolute deviation) to an interval

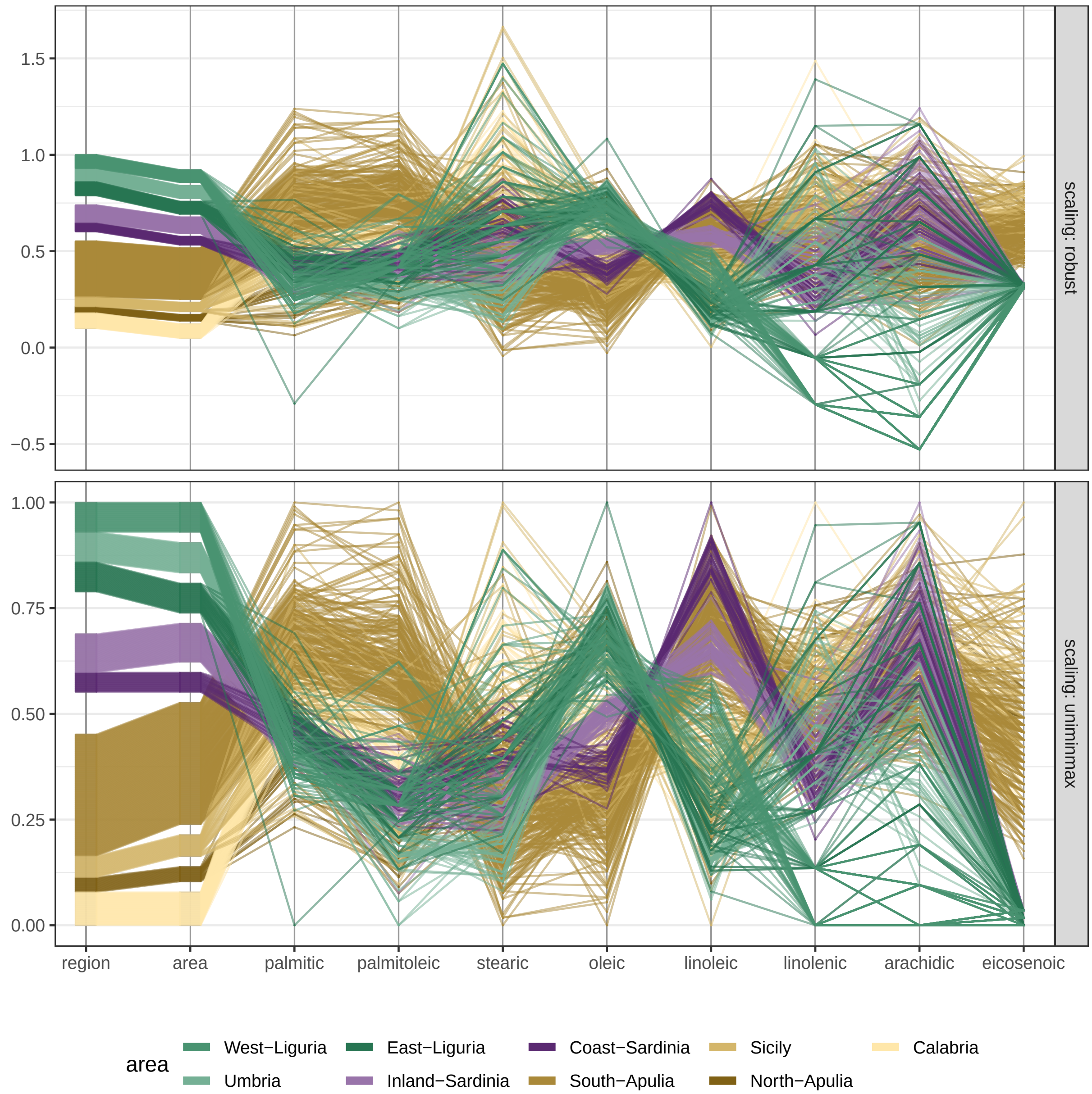


Figure 4: *Two scaling methods showing fatty acid compositions of olive oils from different regions in Italy, areas within each region are colored using similar hues within region (green for Northern Italy, purple for Sardinia, and tans for Southern Italy).robust scale: median to 0 check units XXX The two scaling methods roughly allow the same conclusions.*

of 0 to 1.

Figure 4 shows two of the scaling methods at the example of the olive oil data [Cook and Swayne, 2007, Wickham et al., 2011, Forina et al., 1983]: measurements of fatty acids in 572 olive oils from three different regions in Italy are visualized as parallel coordinate plots. Similar to the findings in Cook and Swayne [2007], we see that eicosenoic acid is only found in increased quantities in olive oils from Southern Italy. Quantities of oleic and linoleic acids allow a separation between olive oils from Sardinia and Northern Italy. Both scaling methods enable us to find these conclusions. While uniminmax scaling uses the space alloted to the chart most efficiently, the robust normalization method emphasizes the heavy tails and skewness of some of the measurements, such as the percentages of stearic and arachidic fatty acids.

4 Visual Rendering

4.1 Breaking ties on categorical axes

XXX How to deal with ties has a huge visual impact. In order to get to that we need to do some more data wrangling. This subsection is somewhere between data management and visualization.

XXX it might be good to separate the section into 'how' (data) and 'why' (visual aspects).

`pcp_arrange(data, method, space)` provides a rescaling of values on categorical axes to break ties. `method` is a parameter specifying which variables to use to break ties. The two implemented methods are `"from-left"` and `"from-right"`, meaning that ties are broken using a hierarchical ordering using variables' values from the left or the right, respectively. The parameter `space` specifies the amount of the 'y' axis to use for space between levels of categorical variables. By default, 5% of the axis is used for spacing.

XXX hierarchical ordering seems to minimize line crossings - but I don't have a proof. If we have a reference on ordering metrics this might go well here

Figure 5 shows several approaches of dealing with categorical variables in parallel coordinate plots. The left-most panel shows two categorical variables and the typical net of lines that forms between them in an original parallel coordinate plot. The other three panels show three different approaches of breaking the ties resulting from the categorical variables, with our favored solution shown on the right: all observations are spaced out evenly. This results in a natural visualization of the marginal frequencies along each axis (additionally enhanced by the lightly greyed boxes grouping observations in the same category). The ordering of the observations within the level is such that a minimal number of line crossings occurs between the axes. This method of dealing with categorical variables is the one we propose in the generalized parallel coordinate plot. While it is aesthetically pleasing, it also allows us in the spirit of the original parallel coordinate plots to follow an individual observation from left to right through the plot even for categorical variables. The other two solutions in the middle panels of Figure 5 show two intermediary solutions of breaking ties in categorical variables: jittering and equi-spaced (unordered) values.

4.2 Plotting Order of line segments

XXX We need to include some code somewhere to also show the 'how'.

XXX Order of levels and order of variables are important. Drawing pcps depends on what your goal might be.

One of the primary advantages of the generalized approach to dealing with categorical variables is the ability to follow a single observation throughout the plot. As the number of observations increase, this becomes less feasible because of overplotting of line segments, particularly for slightly larger data sets. As more observations and line segments are drawn, more lines cross each other, increasing the effort required to follow a polyline from one side of the plot to the other. XXX the order in which crossing lines are drawn is only visible when the lines have different colors. As a countermeasure we carefully control the order in which line segments are plotted.

Two methods to control this order are implemented and are regulated by the parameter `overplot`:

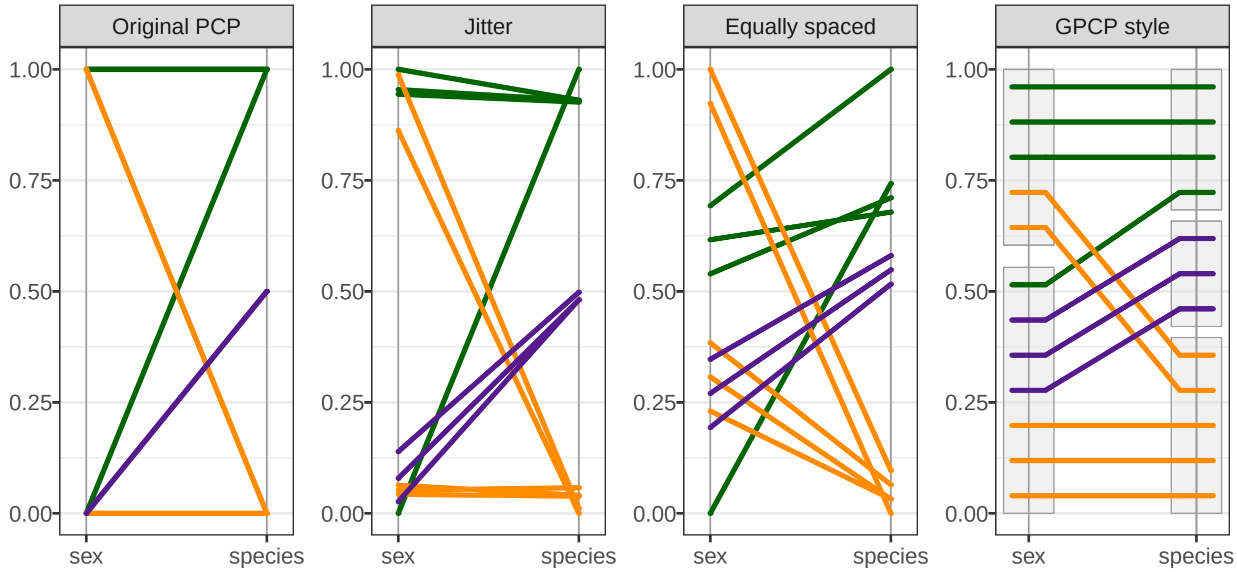


Figure 5: *At the example of 12 randomly sampled penguins from the Palmer penguin data, we show four different approaches of dealing with categorical variables: the panel on the left shows the typical net of lines resulting from categorical variables in regular parallel coordinate plots. In the other three panels ties in categorical levels are broken using different approaches: from left to right, jittering, equi-spaced line segments and ordered equi-spaced line segments are shown.*

1. `"small-on-top"` groups of lines (defined by the same color) between categorical variables are are ordered by size and drawn from largest to smallest group.

2. `"none"` the order of the observations is left un-touched by the data wrangling process. This option allows the user to specify a certain order before plotting.

additional counter-measures: use α-blending
the option of `"none"` is very flexible.

5 Examples

5.1 Palmers Penguins

Several aspects in Parallel Coordinate Plots depend on orderings: order of variables along the x axis, order of levels in a categorical variable, and order in which lines are drawn. Different orders emphasize different aspects of the data. Changing orders should therefore (a) have good defaults, and (b) be easily changeable.

XXX Drawing pcps depends on what your goal might be. It reminds me of rotating plots and all the optimisation criteria that were suggested for a variety of purposes. They did not necessarily work well, but they suggested some ways to proceed and generally improved things a bit. What criteria might be similarly helpful for pcps?

Figure 6 shows a generalized parallel coordinate plot of the Palmer penguins data [Horst et al., 2020]. The data consists of body measurements, such as weight, flipper length, bill length, and depth, of three species of penguins. What can be seen is that Adelie penguins generally have smaller bill lengths than the other two species, while Gentoo penguins can be distinguished from the other two species by their relatively larger flipper lengths.

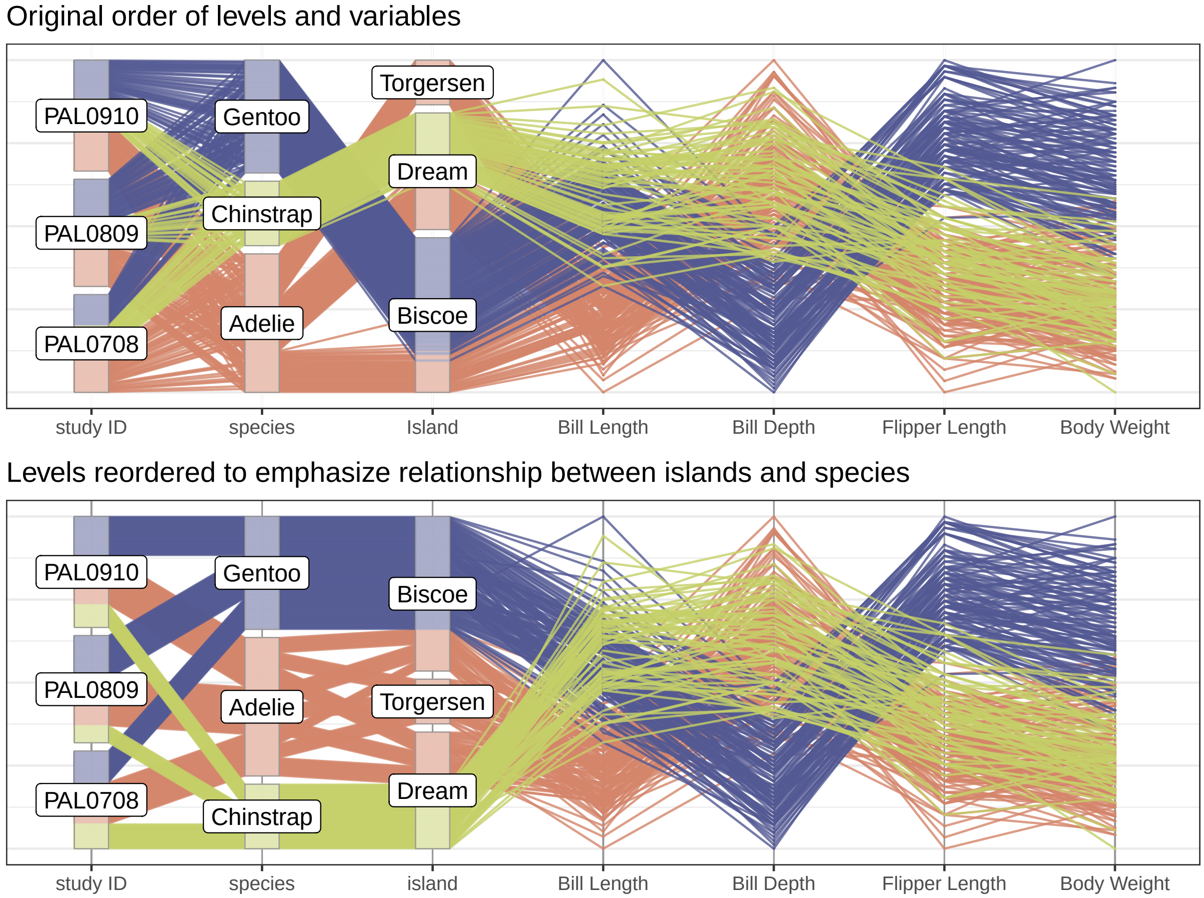


Figure 6: *Both of the levels of the island and the species variable re-ordered to reflect that two of the species are each only found on one island.*

5.1.1 Distinguishing species

Figure 6 shows the effect of re-ordering the levels of both the 'species' and the 'island' variable in the generalized parallel coordinate plots. This re-ordering of factor levels has the effect of emphasizing the fact that Gentoo penguins and Chinstrap penguins are each found on only one island, while Adelie penguins are found on all three islands.

In addition to the ordering of factor levels, it is also important to carefully order the variables on the x-axis, as shown in Figure 7, where the variables have been re-ordered to allow the viewer to see what body measurements distinguish the species. Note that besides the re-ordering, the axes for bill depth is also reversed. XXX explain why this helps Again, this reversal helps with distinguishing between species: Gentoo penguins have the lowest bill depth, while generally having the longest flippers. Reversing the axis for bill depths, aligns the smallest bill depths with the longest flippers, thereby moving Gentoo penguins closer together as a group. The plot shows that the Gentoo penguins are bigger, that Gentoo and Chinstrap are both only found on single islands, and, finally, that Adelie and Chinstrap are distinguished by the lengths of their bills.

Note that we are using facetting by penguins' sex in Figure 7. While all of the results are the same for the two sexes, any variability of body measures due to sex is removed from the plot by facetting. This makes the results stand out more.

Interestingly, some potential outliers that were not visible previously, now become visible. Note for example the two Gentoo males with particularly short flippers.

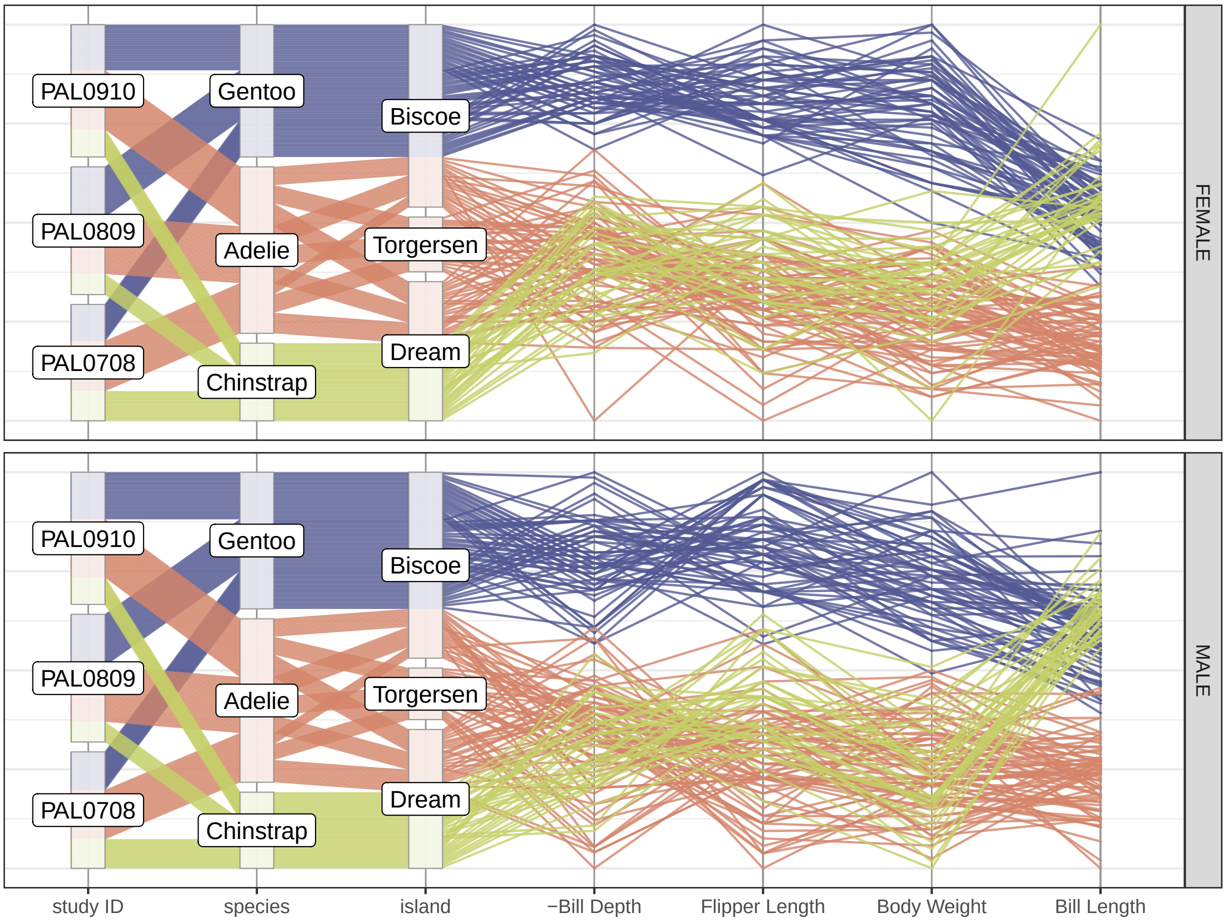


Figure 7: *Changing the order of the variables along the x-axis emphasizes the differences in body measurements between the species.*

XXX Possible other ordering approaches would be to use some kind of association measure or a matrix reordering algorithm for two categorical variables, medians by category or R2 from a linear model for continuous and categorical, correlations for two continuous variables.

5.1.2 Determining sex

XXX intro to ... now we are shifting focus to draw generalized pcps to determine sex.

The lines in Figure 8 are colored by sex of penguins. What can be seen is that within each species, the males tend to be larger in size and heavier than the females. For several of the penguins, sex could not be determined because either the sexing primer did not amplify or no blood sample was obtained [Gorman et al., 2014]. These penguins are represented by dark lines. Based on these penguins' body measurements within the context of the other penguins, we can make some suggestions regarding their sex. **??** shows another version of a generalized parallel coordinate plot: ribbons of inter-quantile ranges XXX define inter-quantile range are plotted across body measurements. XXX inter-quantile range: find quantiles on each axis and connect. In data of neighboring axes, this creates a bounding box: for independent variables we can say that the probability that a line is within an inter-quantile range of p is p^2, while the probability to be within one margins (regardless of what happens on the other side) is $2 \times p - p^2$. For dependent variables these probabilities change XXX We use the inter-quantile ranges to reduce the noise introduced by individual lines. This helps us evaluate and assess lines for individual penguins within the context of

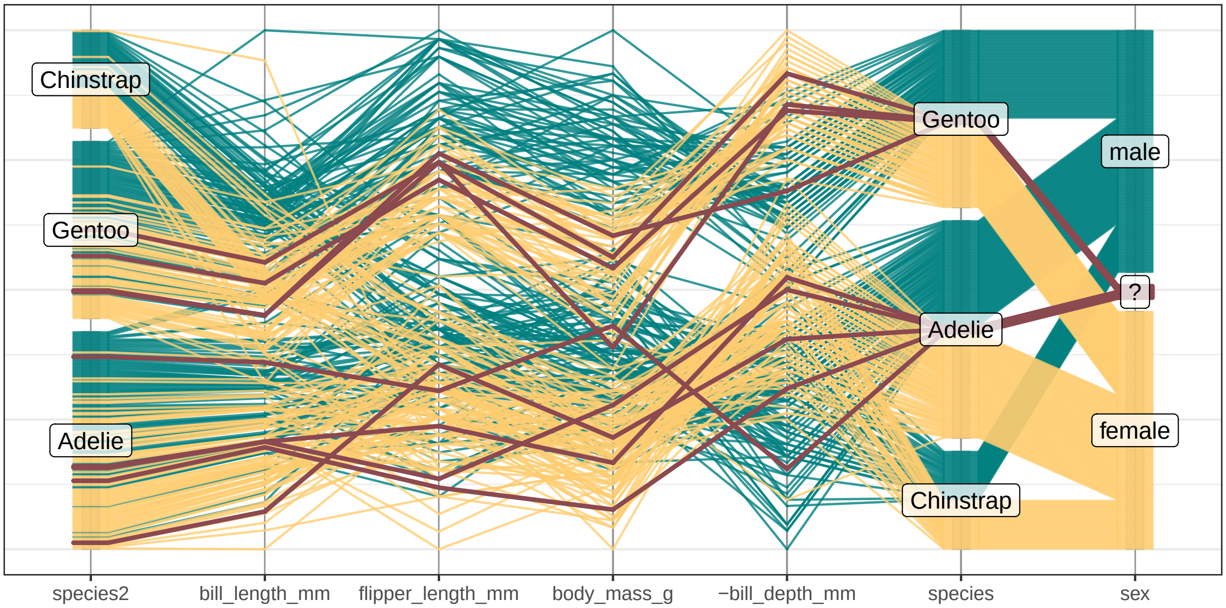


Figure 8: *Generalized Parallel Coordinate Plot of the Palmer penguins data with sex of penguin mapped to color. Dark lines represent penguins for which sex could not be determined. We see that researchers were able to sex all of the Chinstrap penguins. Note that species is included twice (with different order of the levels). XXX reason for that here or in the text?*

the marginal distributions (in this case their sex and species). XXX In **??** we facet by species and sex. The two ribbons shown represent the interquartile range (middle 50% of the data) and the range between the 2.5% and the 97.5% quantiles (inner 95%). These ribbons provide a basic summary of the each variable and its two dimensional density with each of its adjacent axes. XXX that needs some work - marginal densities don't work well with higher dimensions Body measurements of the unsexed animals are represented as line segments on top. facetted version of the previous generalized parallel coordinate plots. While we facet both by species and sex, note that the axes are re-scaled within each species to make use of the full range in y. However, we use the same scale between the two sexes of each species. This different treatment of facetting variables is achieved by the use of the grouping variable. The listing in **??** shows the code for prepping the data shown in **??**. By grouping on species but not on sex (see line 9), data is being rescaled within species but the same scaling is used across males and females. Measurements for unsexed animals are shown as line segments on top of the interquantile-ribbons of both sexes. Viewers are encouraged to draw a conclusion about an animal's sex based on their values within the (2d density) context of their species and hypothetical sex. Statistically, this comparison relates to a likelihood ratio test: the viewer is asked to make an assessment of the likelihood to observe the measurements of an animal under each of the two competing hypotheses of sex.

```
1  penguins_pcp <- penguins %>%
2      filter(species != "Chinstrap") %>%                   # no unsexed animals in Chinstrap
3      mutate(
4          sex = ifelse(is.na(sex),"?", as.character(sex)), # make assignment more readable
5          sex = factor(sex, levels = c("female", "?", "male"))
6      ) %>%
7    filter(!is.na(body_mass_g)) %>%
8    pcp_select(6:5, 3:4) %>%
9    group_by(species) %>%                                  # re-scale by species
10   pcp_scale() %>%
```

```
11   pcp_arrange()
```

Chinstrap penguins are excluded (line 2) because all of their individuals in the data have a sex assigned. The general pattern of measurements of the Gentoo penguins suggests that three of the four individuals with missing sex information are female (the three with the lowest bill depth). The fourth animal has an exceptionally deep bill, however, all other measurements suggest that this animal, too, is female. For further evidence, we find from the original data that their nest partners are all sexed as male. While assuming that nest partners are male and female is not a perfect method, in particular, for penguins, which have been shown to live in same-sex partnerships, in all three of the studies considered for this data only nests with breeding successes have been considered. More details can be found in Gorman et al. [2014]. For Adelie penguins the situation is not quite as clear-cut, but based on body mass and bill length measurements the three lightest penguins might be female, while the heaviest one could be male. The fifth penguin exhibits measurements that are neither typically male not typically female. Trying to confirm these putative assignments is a bit more tricky, because four of the five unsexed penguins are nest partners. The lightest unsexed penguin is the partner of a sexed male penguin. The putative assignments do not contradict the hypothesis that each nest is host to a male and a female penguin. Using this assumption, the last unsexed penguin would be resolved as the male of the nest.

5.2 Getting a second, third, ... and seventh opinion

Figure 12 shows data from Agresti [2002] published as part of the `poLCA` package [Linzer and Lewis, 2011]. Seven pathologists were asked to assess the same 118 slides for the presence or absence of carcinoma in the uterine cervix. Binary responses for each slide were recorded (yes/no). Pathologists all agreed on about 25% of slides, which they considered to be carcinoma free, and a further 12.5% of slides, which were considered to show carcinoma by all pathologists. For the remaining 62.5% of slides there was some disagreement. However, we see that this disagreement is not random. When pathologists are ordered (by moving the corresponding axes) left to right from fewest number of overall carcinoma diagnoses to highest number, we see that generally for a slide more pathologists make a carcinoma diagnosis from left to right.

note: in this example we do not need to scale the variables. Aside from the actual scale the values are ordered in the same way.

5.3 Clustering with PCPs

Let's revisit the penguins data for an example of working with clusters in the framework of generalized parallel coordinate plots. We use k-means clustering on all numeric body measurements and investigate which observations are generally captured in each of the clusters.

Because k-means clustering assigns cluster labels arbitrarily based on random cluster centers, in order to maintain a persistent ordering over different values of k we reorder the clusters by the value of `body_mass_g`. This allows us to compare between e.g. k and $k+1$ clusters.

Figure 13 shows the numeric measurements along with the assigned clusters, with categorical variables sex and species on the right. Each line is colored by the assigned cluster, allowing us to determine how the categorical variables relate to the quantitative variables and the resulting clusters.

When $k = 2$, Figure 13a shows that the largest difference in the observed data is between Gentoo penguins and the other two species. When $k = 3$, in Figure 13b, the additional cluster separates the Adelie and Chinstrap penguins into two groups with a few misclassifications; this additional cluster is based on the length of the bill (which we can follow due to the clear connection between data values in the generalized PCP). Adding a fourth cluster, as in Figure 13c splits Adelie penguins into males and females, though again there are again some penguins who are misclassified. The addition of a fifth cluster

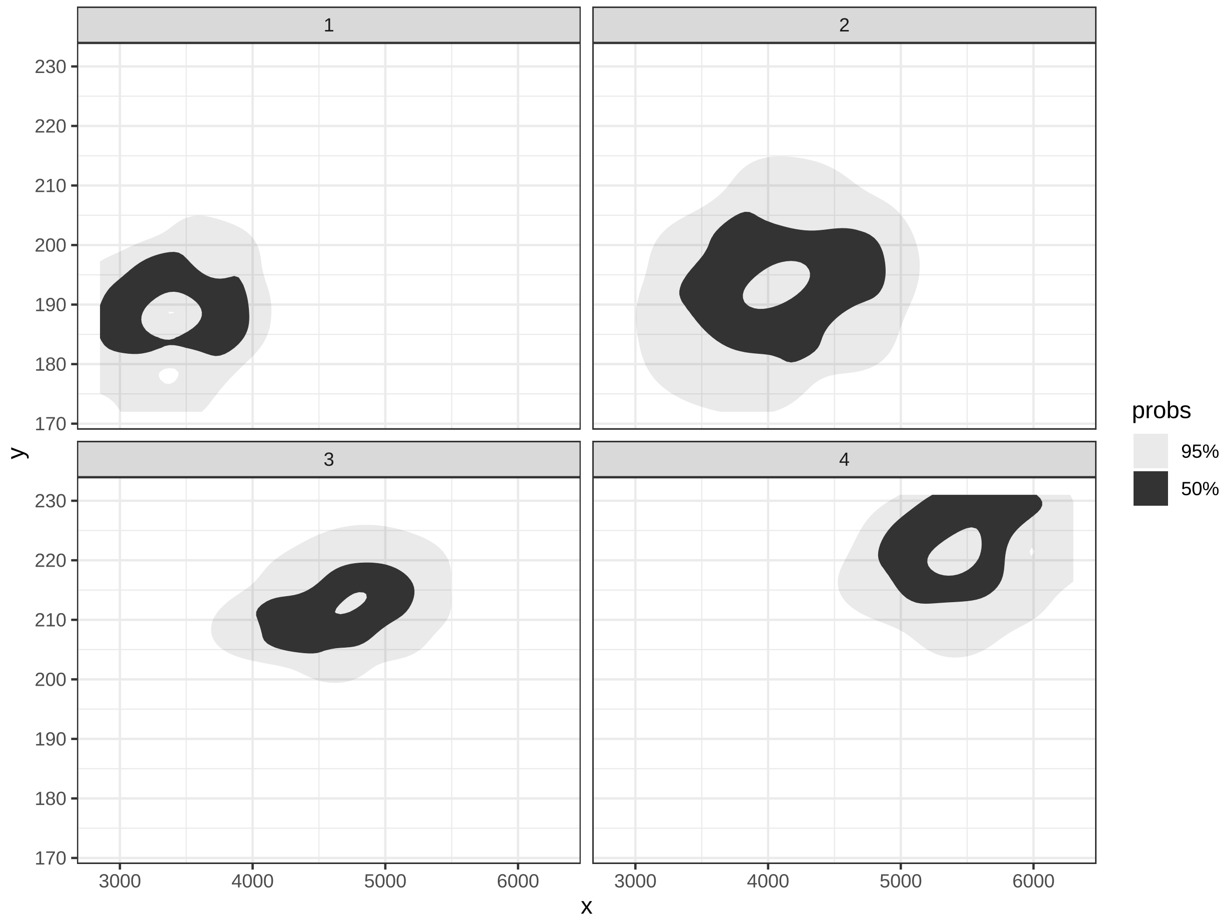


Figure 9: *Closer investigation of non-sexed Adelie and Gentoo penguins. The group_by call before pcp_scale is responsible for scaling by species while the same scale is kept across sex within species. Penguins without assigned sex (based on blood markers) are drawn on top of both sexes. The labels to the left of the ribbons are our best guess at a penguin's sex based on body measurements of other penguins of the same species. The markers on the right indicate four unsexed penguins that are nest partners.*

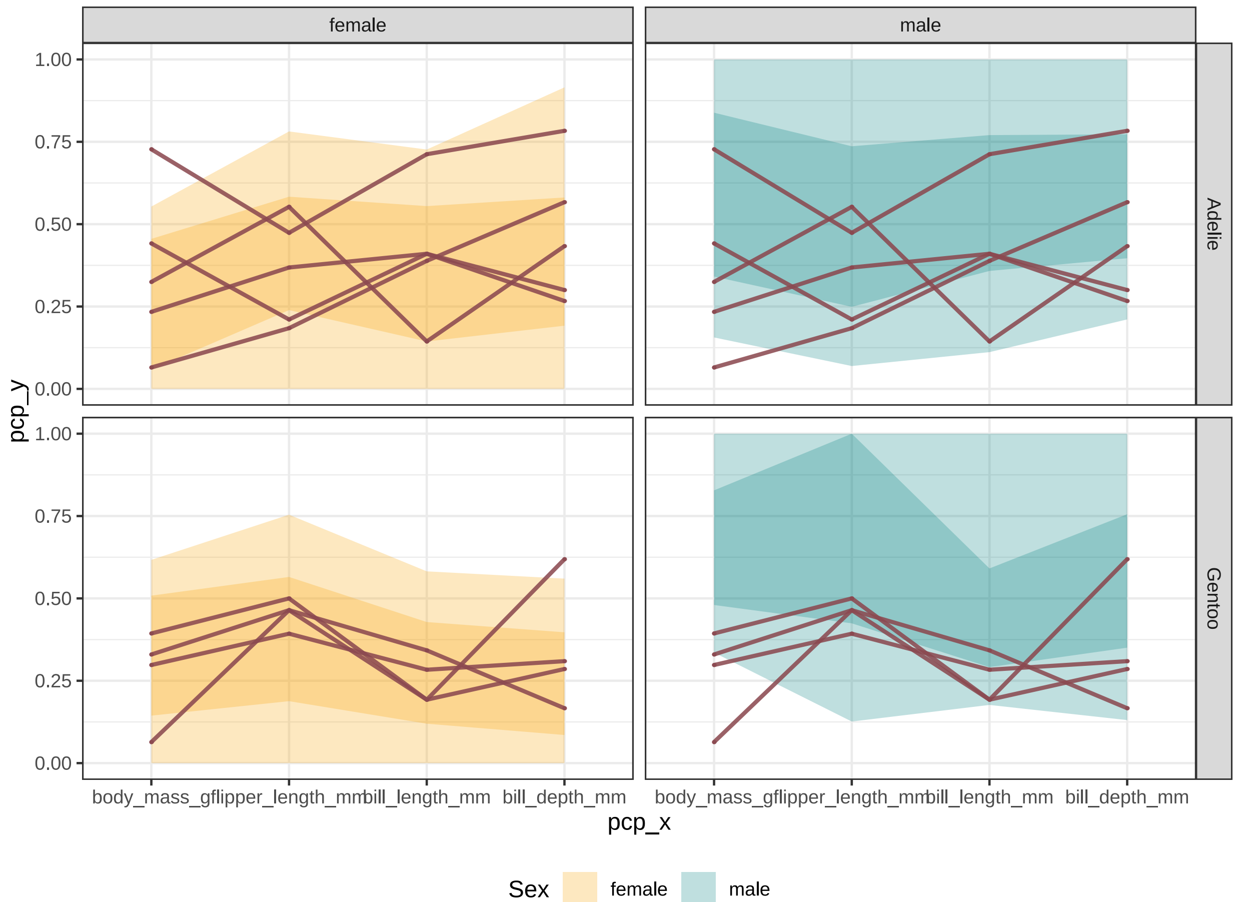


Figure 10: *Closer investigation of non-sexed Adelie and Gentoo penguins. The group_by call before pcp_scale is responsible for scaling by species while the same scale is kept across sex within species. Penguins without assigned sex (based on blood markers) are drawn on top of both sexes. The labels to the left of the ribbons are our best guess at a penguin's sex based on body measurements of other penguins of the same species. The markers on the right indicate four unsexed penguins that are nest partners.*

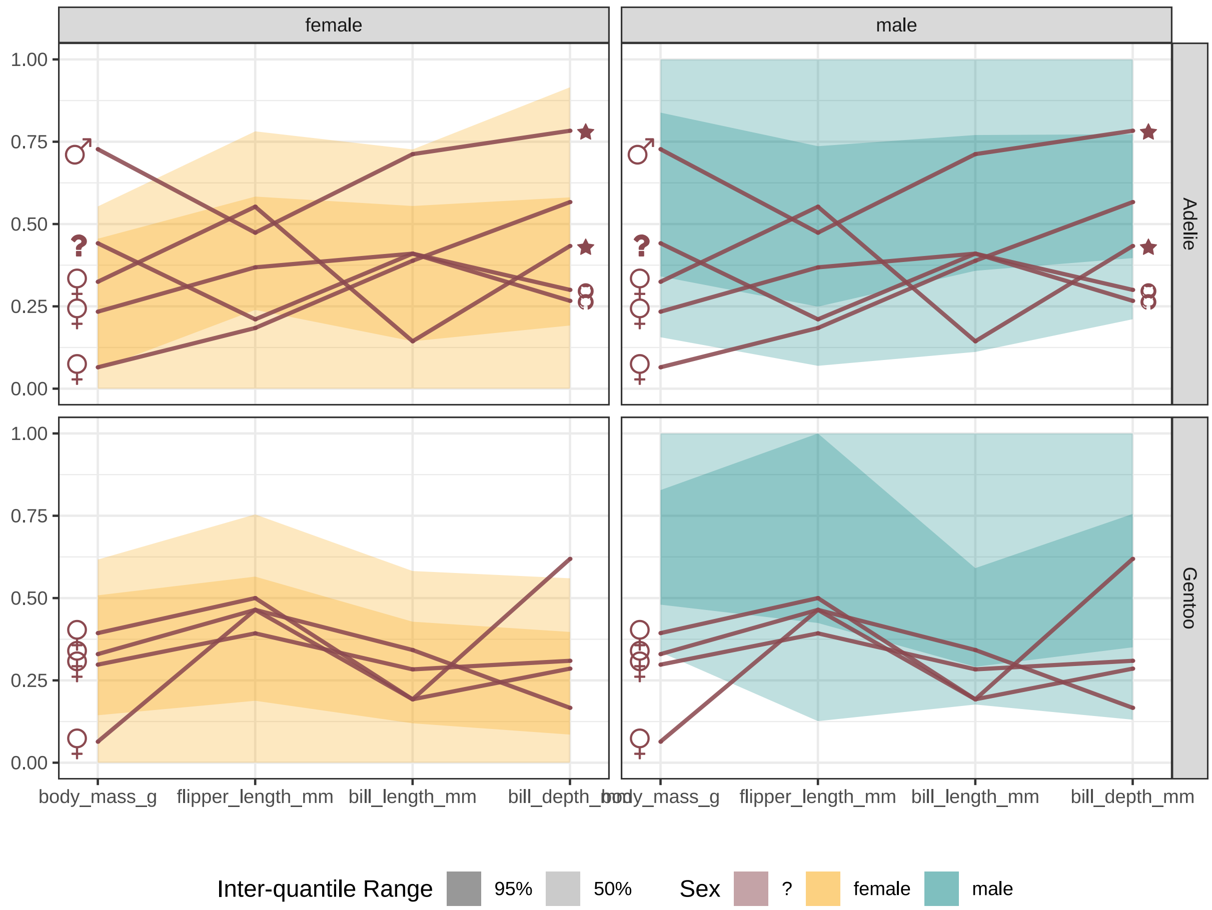


Figure 11: *Closer investigation of non-sexed Adelie and Gentoo penguins. The group_by call before pcp_scale is responsible for scaling by species while the same scale is kept across sex within species. Penguins without assigned sex (based on blood markers) are drawn on top of both sexes. The labels to the left of the ribbons are our best guess at a penguin's sex based on body measurements of other penguins of the same species. The markers on the right indicate four unsexed penguins that are nest partners.*

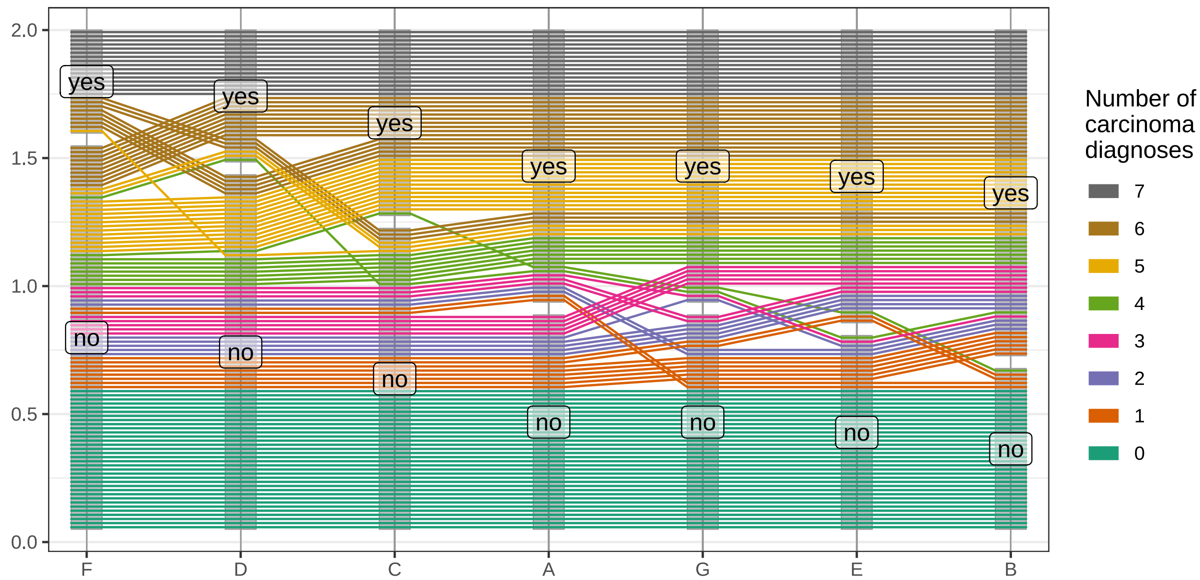


Figure 12: *Pathologists' diagnoses of absence (no) or presence (yes) of carcinoma in the uterine cervix based on 118 slides. Each slide is shown by a polyline.*

in Figure 13d splits Chinstrap penguins into male and female. Once we add a sixth cluster in Figure 13e, we finally split the Gentoo penguins by sex as well, though again this clustering is not perfect.

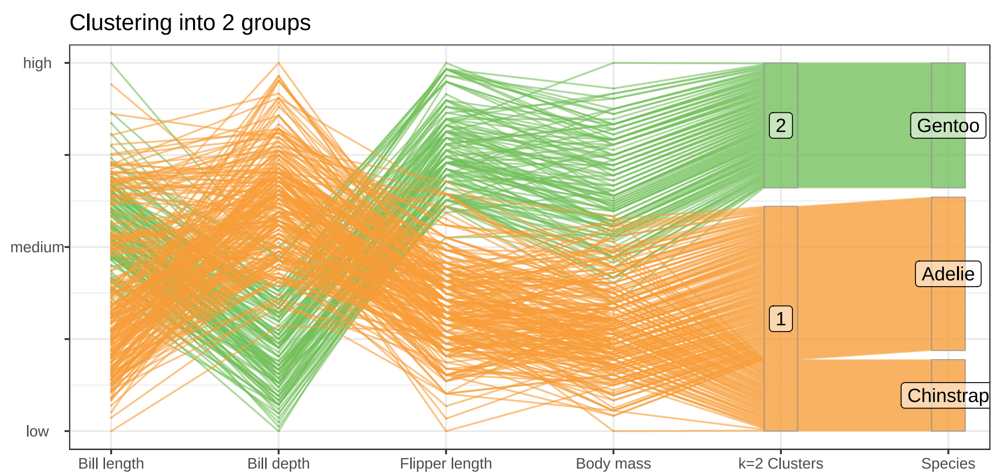
What is clear from this exercise is that Adelie and Chinstrap penguins are much more similar to each other than they are to Gentoo penguins, but that there is still noticable sexual dimorphism within each species. We also see from the figure that some of separation into sexes is lost from one clustering to the next. This is typical for non-hierarchical clustering algorithms. Rather than refining a previous cluster, a switch from x classes to $x + 1$ classes starts the clustering process anew. If the signal in the data to separate into k classes is not strong or ambiguous, we will see this reflected in the results; observations might be quite arbitrarily put together into groups, or a group of observations might be split into multiple clusters. In the Hartigan-Wong Hartigan and Wong [1979] algorithm used here for the clustering, points are assigned to random clusters in the initialization. In order to assess the effect of this non-deterministic start on the results, we are often advised to investigate the cluster stability by repeating the clustering multiple times for the same number of classes k (if $k > 1$). Figure 14 shows a comparison of the results from multiple runs of the k-means algorithm for $k = 6$. The lines in this figure are colored by species and sex. We see that the splits by species are relatively stable - there are only a few cases across all results in which individuals end up in clusters with individuals from another species, and if they do, it is the same individuals across different results. Splits by sex show more variability: Chinstrap penguins rarely split into male/female clusters, while Gentoo shows a relatively stable separation into males and females. The Adelie population has subsets of individuals that are separated into males only, females only, and a third, more variable subset of a combination of the two.

XXX with Susan: using generalized parallel coordinate plots and item sorting allows us to assess the stability of clustering results.

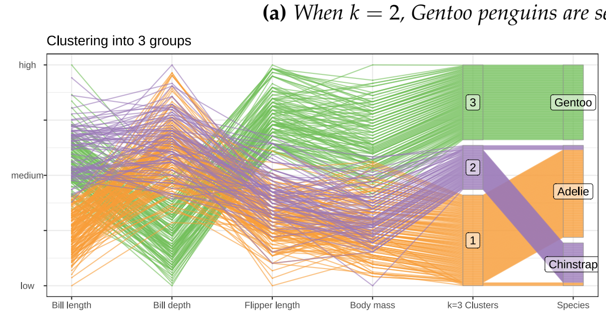
6 Discussion

References

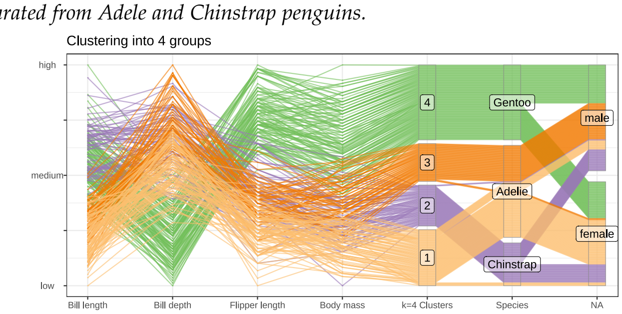
A. Agresti. *Categorical Data Analysis*. John Wiley & Sons, Hoboken, 2 edition, 2002.



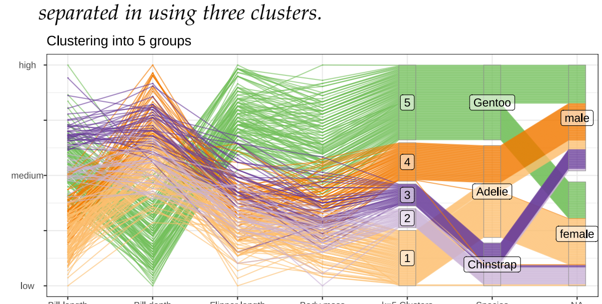
(a) *When $k = 2$, Gentoo penguins are separated from Adele and Chinstrap penguins.*



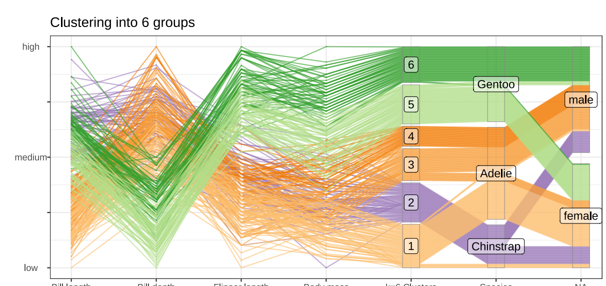
(b) *The third cluster splits the former second cluster into two based on the length of the bill. The three species are almost perfectly separated in using three clusters.*



(c) *The fourth cluster splits Adelie penguins (mostly) into males and females of the species.*



(d) *The fifth cluster splits the group of Chinstrap penguins (mostly) into females and males. Only a handful of individual penguins are grouped with the wrong species or the wrong sex.*



(e) *With the introduction of a sixth cluster, all previous clusters change: Gentoo penguins are split into males and females. The sex separation for Chinstrap penguins gets lost, but Adelie males get split into two separate clusters.*

Figure 13: *An overview of the use of parallel coordinate plots to examine which variables contribute to clustering and to identify individuals who are misclassified.*

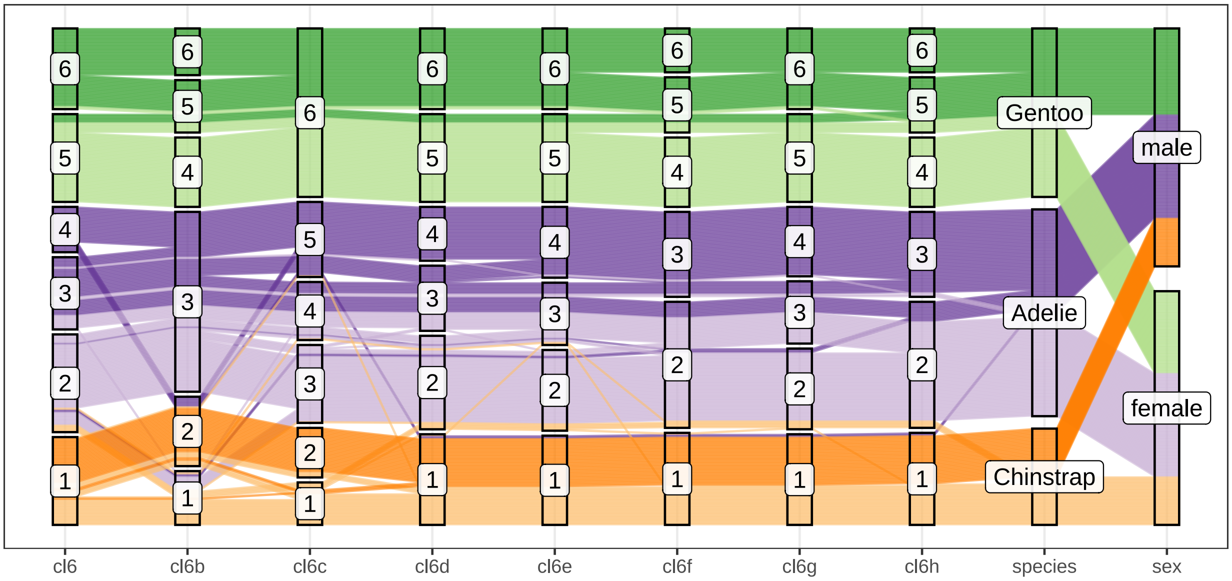


Figure 14: *Comparison of eight k-means runs for $k = 6$. Color of lines is given by species and sex. The differences between the clusters are introduced by the different random seeds in the inital cluster centers.*

D. Cook and D. F. Swayne. *Interactive and Dynamic Graphics for Data Analysis With R and GGobi*. Springer Publishing Company, Incorporated, 1st edition, 2007. ISBN 0387717617.

R. H. Day and E. J. Stecher. Sine of an illusion. *Perception*, 20:49–55, 1991.

M. d'Ocagne. Coordonnées parallèles et axiales : Méthode de transformation géométrique et procédé nouveau de calcul graphique déduits de la considération des coordonnées parallèles. *Gauthier-Villars*, page 112, 1885. URL `https://archive.org/details/coordonnesparal00ocaggoog/page/n10`.

M. Forina, C. Armanino, and S. Lanteri. Classification of olive oils from their fatty acid composition. *Food Research and Data Analysis*, pages 189–214, 01 1983.

H. Gannett. General summary showing the rank of states by ratios 1880, plate 71. In *Scribner's statistical atlas of the United States, showing by graphic methods their present condition and their political, social and industrial development*. Charles Scribner's Sons, New York, 1880.

K. B. Gorman, T. D. Williams, and W. R. Fraser. Ecological sexual dimorphism and environmental variability within a community of antarctic penguins (genus pygoscelis). *PLOS ONE*, 9(3):1–14, 03 2014. doi: 10.1371/journal.pone.0090081. URL `https://doi.org/10.1371/journal.pone.0090081`.

J. A. Hartigan and M. A. Wong. Algorithm as 136: A k-means clustering algorithm. *Applied Statistics*, 28 (1):100, 1979. doi: 10.2307/2346830. URL `http://dx.doi.org/10.2307/2346830`.

J. Heinrich and D. Weiskopf. Continuous Parallel Coordinates. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1531–1538, 2009. doi: 10.1109/TVCG.2009.131. URL `http://ieeexplore.ieee.org/document/5290770/`.

J. Heinrich and D. Weiskopf. State of the Art of Parallel Coordinates. In M. Sbert and L. Szirmay-Kalos, editors, *Eurographics 2013 - State of the Art Reports*. The Eurographics Association, 2013. doi: 10.2312/conf/EG2013/stars/095-116.

H. Hofmann and M. Vendettuoli. Common Angle Plots as Perception-True Visualizations of Categorical Associations. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2297–2305, Dec. 2013. doi: 10.1109/TVCG.2013.140. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6634157.

A. M. Horst, A. P. Hill, and K. B. Gorman. *palmerpenguins: Palmer Archipelago (Antarctica) penguin data*, 2020. URL https://allisonhorst.github.io/palmerpenguins/. R package version 0.1.0.

A. Inselberg. The plane with parallel coordinates. *The Visual Computer*, 1(2):69–91, Aug. 1985. doi: 10.1007/BF01898350. URL http://link.springer.com/10.1007/BF01898350.

R. Kosara, F. Bendix, and H. Hauser. Parallel Sets: interactive exploration and visual analysis of categorical data. *IEEE Transactions on Visualization and Computer Graphics*, 12(4):558–568, 2006. doi: 10.1109/TVCG.2006.76. URL http://ieeexplore.ieee.org/document/1634321/.

D. A. Linzer and J. B. Lewis. poLCA: An R package for polytomous variable latent class analysis. *Journal of Statistical Software*, 42(10):1–29, 2011. URL http://www.jstatsoft.org/v42/i10/.

K. T. McDonnell and K. Mueller. Illustrative Parallel Coordinates. *Computer Graphics Forum*, 27(3):1031–1038, May 2008. doi: 10.1111/j.1467-8659.2008.01239.x. URL http://doi.wiley.com/10.1111/j.1467-8659.2008.01239.x.

J. J. Miller and E. J. Wegman. *Computing and Graphics in Statistics*, chapter Construction of Line Densities for Parallel Coordinate Plots, pages 107–123. Springer-Verlag New York, Inc., New York, NY, USA, 1991. ISBN 0-387-97633-7. URL http://dl.acm.org/citation.cfm?id=140806.140816.

P. Murrell and S. Potter. *gridSVG: Export 'grid' Graphics as SVG*, 2020. URL https://CRAN.R-project.org/package=gridSVG. R package version 1.7-2.

A. Pilhöfer and A. Unwin. New Approaches in Visualization of Categorical Data: R Package extracat. *Journal of Statistical Software*, 53(7), 2013. doi: 10.18637/jss.v053.i07. URL http://www.jstatsoft.org/v53/i07/.

M. Schonlau. Visualizing Categorical Data Arising in the Health Sciences Using Hammock Plots. In *Proceedings of the Section on Statistical Graphics, American Statistical Association*, 2003.

C. Sievert. *Interactive Web-Based Data Visualization with R, plotly, and shiny*. Chapman and Hall/CRC, 2020. ISBN 9781138331457. URL https://plotly-r.com.

S. VanderPlas and H. Hofmann. Signs of the sine illusion–why we need to care. *Journal of Computational and Graphical Statistics*, 24(4):1170–1190, 2015. doi: 10.1080/10618600.2014.951547. URL https://doi.org/10.1080/10618600.2014.951547.

E. J. Wegman. Hyperdimensional data analysis using parallel coordinates. *Journal of the American Statistical Assoiation*, 85:664–675, 1990.

H. Wickham. Reshaping data with the reshape package. *Journal of Statistical Software*, 21(12), 2007. URL http://www.jstatsoft.org/v21/i12/paper.

H. Wickham. Tidy data. *Journal of Statistical Software, Articles*, 59(10):1–23, 2014. ISSN 1548-7660. doi: 10.18637/jss.v059.i10. URL https://www.jstatsoft.org/v059/i10.

H. Wickham. *tidyr: Tidy Messy Data*, 2021. URL https://CRAN.R-project.org/package=tidyr. R package version 1.1.3.

H. Wickham, D. Cook, H. Hofmann, and A. Buja. tourr: An R Package for Exploring Multivariate Data with Projections. *Journal of Statistical Software, Articles*, 40(2):1–18, 2011. ISSN 1548-7660. doi: 10.18637/jss.v040.i02. URL https://www.jstatsoft.org/v040/i02.

H. Wickham, R. François, L. Henry, and K. Müller. *dplyr: A Grammar of Data Manipulation*, 2021. URL `https://CRAN.R-project.org/package=dplyr`. R package version 1.0.7.

A Example usage

Commented code for Figure 12:

```
data(carcinoma, package = "poLCA")

# Prepping the Dataset
carcinoma$total <- rowSums(carcinoma) - 7
carcinoma <- carcinoma %>% mutate(
  across(A:G, .fns = as.factor)
)

carcinoma %>%

# Selecting and scaling variables
pcp_select(F, D, C, A, G, E, B, tot) %>%
pcp_scale(method="uniminmax") %>%
pcp_arrange() %>%

# Setting up ggplot for pcp and setting pcp display options
ggplot(aes_pcp()) +
geom_pcp_axes() +
geom_pcp_boxes(colour="black", alpha=0) +
geom_pcp(aes(colour = tot)) +
geom_pcp_labels(aes(label = pcp_level), fill="white", alpha = 1) +
```

```
# Choosing general ggplot display options
scale_colour_brewer("Number of\ncarcinoma\ndiagnoses", palette ="Dark2") +
theme_bw() +
guides(color = guide_legend(reverse=TRUE, override.aes = list(size = 5))) +
scale_x_discrete(expand = expansion(add=0.25)) +
xlab(NULL) + ylab(NULL) +
theme(axis.text.y=element_blank(), axis.ticks.y=element_blank())
```