




tidypaleo: Visualizing Paleoenvironmental Archives Using ggplot2

Dewey W. Dunnington 
Dalhousie University

Nell Libera 
Queen's University

Joshua Kurek 
Mt. Allison University

Ian S. Spooner 
Acadia University

Graham A. Gagnon 
Dalhousie University

Abstract

This paper presents the **tidypaleo** package for R, which enables high-quality reproducible visualizations of time-stratigraphic multivariate data that is common to several disciplines of the natural sciences. Rather than introduce new plotting functions, the **tidypaleo** package defines several orthogonal components of the **ggplot2** package that, when combined, enable most types of stratigraphic diagrams to be created. We do so by conceptualizing multi-parameter data as a series of measurements (rows) with attributes (columns), enabling the use of the **ggplot2** facet mechanism to display multi-parameter data. The orthogonal components include (1) scales that represent relative abundance and concentration values, (2) geometries that are commonly used in paleoenvironmental diagrams created elsewhere, (3) facets that correctly assign scales and sizes to panels representing multiple parameters, and (4) theme elements that enable **tidypaleo** to create elegant graphics. Collectively, this approach demonstrates the efficacy of a minimal **ggplot2** wrapper to create domain-specific plots.

Keywords: plotting software, stratigraphic diagrams, R, **ggplot2**.

1. Introduction

Paleoenvironmental archives are critical to our understanding of the past, present, and future (Smol 2009). Analysis of rocks, soils, sediments, and fossils from throughout Earth's history is a powerful approach to reconstruct environmental conditions in the absence of direct human measurements (Smol 2010). For example, tree rings, ice cores, and lake sediments can be used to reconstruct environmental conditions over hundreds to thousands of years (Cohen

2003). Ocean sediments, loess deposits, and ancient soils can be used to reconstruct environmental conditions over hundreds to millions of years (Williams *et al.* 2018). Collectively, paleoenvironmental archives are our link to the past and inform our predictions about the future environment.

Effective visualization of these complex, multivariate data matters. For example, researchers have used paleoenvironmental archives from previous periods of rapid warming to understand our current warming planet (Vincent and Cwynar 2016). Aquatic scientists have used lake sediment archives to recognize that lakes acidified as a result of sulfur emissions from coal burning (Charles *et al.* 1990). Environmental scientists have used records from lake and ocean sediments to confirm the source of potentially toxic pollutants (Dunnington *et al.* 2020). These are data that directly inform policy, and lead to legislation (Smol 2009); high-quality reproducible figures are critical if these data are to be understood and trusted.

tidypaleo is a package for R statistical software (R Core Team 2021) to create effective visualizations for paleoenvironmental data. These data have several characteristics that distinguish them from traditional time series data (Dunnington and Spooner 2018):

- They are generally multi-proxy and multi-archive, in that evidence from multiple parameters (e.g., the relative abundance of dozens of taxa and/or geochemical measures) and multiple archives (e.g., two lake sediment cores from different locations) must be interpreted collectively.
- The connection between position in the archive (e.g., depth in the archive) and calendar age (e.g., AD 1900±15) is important and both must be communicated alongside parameter measurements.
- These archives tend to be oriented vertically (e.g., an ice core), and tend to be plotted vertically with parameter measurements on the x -axis and depth or time on the y -axis.

Because of the unique nature of paleoenvironmental data, constructing diagrams that are elegant, technically correct, and reproducible is challenging and often time-consuming. Whereas previous software packages for creating stratigraphic diagrams (e.g., **C2**, **Tillia*Graph**, **rioja**, and **analogue**) use a graphical user interface or base R plotting approach (Grimm 2016; Juggins 2011, 2020; Simpson 2007), in the **tidypaleo** package, we use the defaults and flexible interface of the **ggplot2** package (Wickham 2016) as a base on which effective paleoenvironmental diagrams can be built. Package **tidypaleo** (Dunnington 2022) is available from the Comprehensive R Archive Network (CRAN) at <https://CRAN.R-project.org/package=tidypaleo>.

2. Example data

In this paper, we use the `kellys_lake_geochem` and `kellys_lake_cladocera` data set, which contain geochemical measurements and microfossil zooplankton (Cladocera) counts from Kellys Lake, Nova Scotia, Canada, to demonstrate the features of the **tidypaleo** package (Figure 1–2).

```
R> library("tidypaleo")
R> data("kellys_lake_geochem", package = "tidypaleo")
R> data("kellys_lake_cladocera", package = "tidypaleo")
```

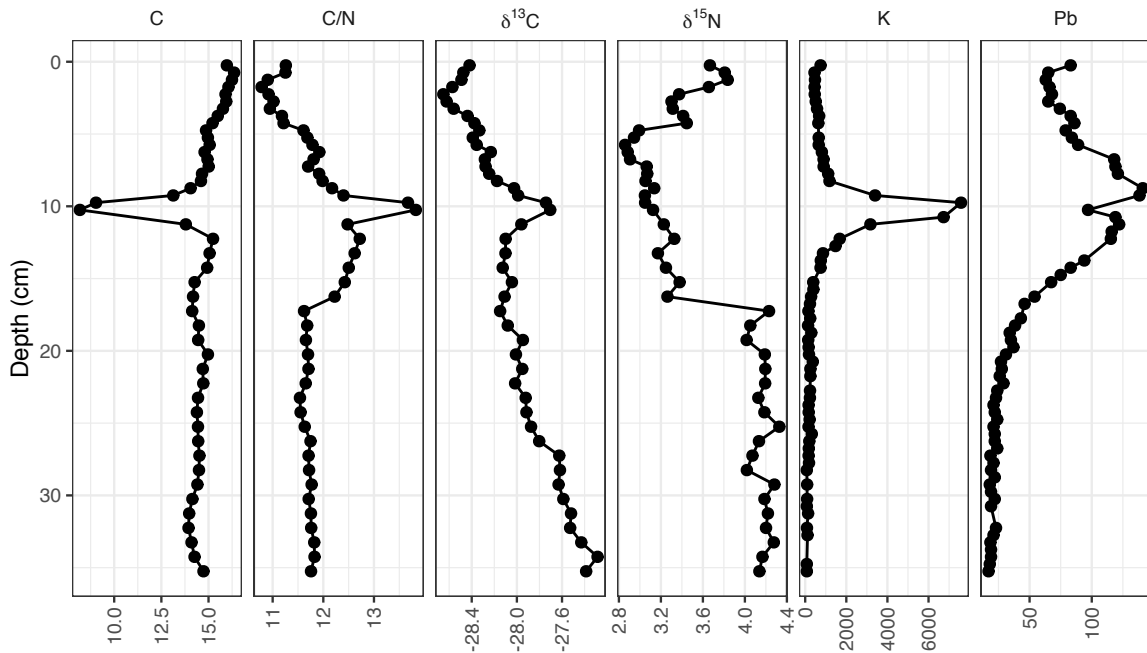


Figure 1: Geochemical measurements from Kellys Lake, Nova Scotia, Canada.

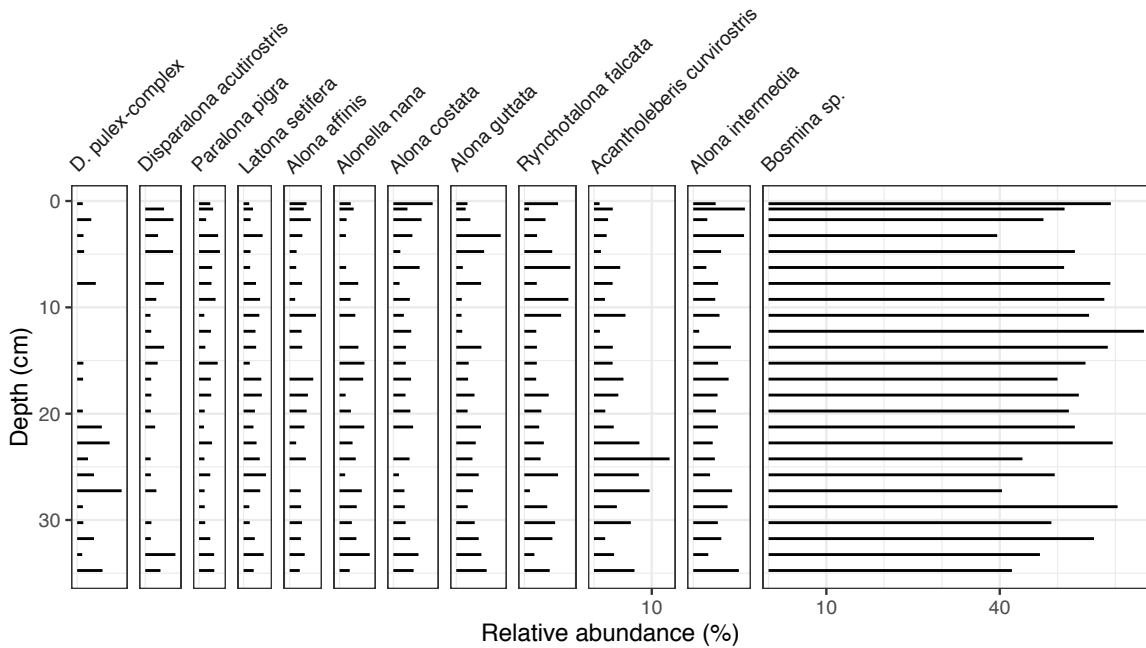


Figure 2: Microfossil zooplankton (Cladocera) relative abundances from Kellys Lake, Nova Scotia, Canada.

```
R> kellys_geochem_plot <- ggplot(kellys_lake_geochem,
+   aes(x = value, y = depth)) +
+   geom_lineh() +
+   geom_point() +
```

```

+   scale_y_reverse() +
+   facet_geochem_gridh(vars(param)) +
+   labs(y = "Depth (cm)", x = NULL)
R> kellys_geochem_plot
R> kellys_abund_plot <- ggplot(kellys_lake_cladocera,
+   aes(x = rel_abund, y = depth)) +
+   geom_col_segsh() +
+   scale_y_reverse() +
+   facet_abundanceh(vars(taxon)) +
+   labs(y = "Depth (cm)")
R> kellys_abund_plot

```

3. Design and implementation

Like **ggplot2**, the **tidypaleo** package provides a number of reusable components that can be combined in flexible and powerful ways to communicate a wide variety of data types. Here, we use the structure laid out by [Wilkinson \(2005\)](#) when describing the *Grammar of Graphics*.

3.1. Data

Essential to **tidypaleo** is a tabular data structure composed of one row per measurement. Columns contain information about each measurement, including common dimensions (for example depth, age, core identifier, and parameter measured), and values specific to each measurement (e.g., measured values, units, and errors). This form of data provides an opportunity for measurement-level details to be retained that is not possible with a more traditional “tidy” format ([Wickham 2014](#); [Dunnington and Spooner 2018](#)), where columns are mix of common dimensions and which parameter was measured, rows represent individual sediment samples, and cells can only represent a single value. Data with measurements as rows and attributes as columns (e.g., core identifier and parameter name) allows a natural use of the **ggplot2** grouping and facet mechanisms to plot multiple parameters and locations. Importantly, this also supports communication of error, since value error is another column in the data structure. The `kellys_lake_geochem` data set used in [Figure 1](#) is shown below.

```
R> kellys_lake_geochem
```

```

# A tibble: 305 x 9
  location param depth age_ad value error error_type n_detect    n
  <chr>      <chr> <dbl> <dbl> <dbl> <dbl> <chr>      <int> <int>
1 KLY17-2   C      0.25  2017.  16.0   NA sd         1      1
2 KLY17-2   C      0.75  2016.  16.4   NA sd         1      1
3 KLY17-2   C      1.25  2015.  16.2   NA sd         1      1
4 KLY17-2   C      1.75  2014.  16.1   NA sd         1      1
5 KLY17-2   C      2.25  2013.  15.9   NA sd         1      1
6 KLY17-2   C      2.75  2011.  15.9   NA sd         1      1
7 KLY17-2   C      3.25  2009.  15.8   NA sd         1      1
8 KLY17-2   C      3.75  2007.  15.5   NA sd         1      1

```

```

 9 KLY17-2 C      4.25 2005. 15.2  NA sd      1      1
10 KLY17-2 C      4.75 2003. 14.9  NA sd      1      1
# ... with 295 more rows

```

Most data are not provided to or entered by paleoenvironmental researchers in this format. A more common storage/data entry format is a spreadsheet with one row per sample and one column per geochemical parameter and/or taxon. For geochemical data, paired columns are required to store error and/or detection limits for each measurement (Dunnington and Spooner 2018). To read and convert a spreadsheet to the form required by **tidypaleo**, one can use the **readxl** and **tidyr** packages (Wickham and Bryan 2019; Wickham 2021). In particular, the `pivot_longer()` function from the **tidyr** package converts the table from a form where each row represents a sample to one where each row represents a measurement. Because the table contains error information in addition to measurement values for each parameter, two pivot operations are needed with a join to ensure that error information is available in the final table.

```

R> kellys_lake_geochem_wide <- readxl::read_excel("kellys_lake_geochem.xlsx")
R> kellys_lake_geochem_error <- kellys_lake_geochem_wide %>%
+   select(location, depth, ends_with("_sd")) %>%
+   pivot_longer(-c(location, depth), names_to = "param",
+     values_to = "error") %>%
+   mutate(param = str_remove(param, "_sd"))
R> kellys_lake_geochem_long <- kellys_lake_geochem_wide %>%
+   select(-ends_with("sd")) %>%
+   pivot_longer(-c(location, depth, age_ad), names_to = "param",
+     values_to = "value") %>%
+   filter(!is.na(value)) %>%
+   left_join(kellys_lake_geochem_error,
+     by = c("location", "depth", "param"))

```

Another format used primarily to enter bioindicator data is a spreadsheet with one row per taxon and one column per sample. Converting this form of data to one where each row represents a measurement requires an extra step to separate the location, start depth, and end depth, several of which may be encoded in the column names.

```

R> kellys_lake_cladocera_wide <- readxl::read_excel(
+   "kellys_lake_cladocera.xlsx")
R> kellys_lake_cladocera_long <- kellys_lake_cladocera_wide %>%
+   pivot_longer(-taxon, names_to = "sample_id", values_to = "count") %>%
+   separate(sample_id, into = c("location", "depth"), sep = " ") %>%
+   separate(depth, into = c("depth_start", "depth_end"),
+     sep = "-", convert = TRUE) %>%
+   mutate(depth_mid = (depth_start + depth_end) / 2) %>%
+   select(location, starts_with("depth"), taxon, count)

```

A final step that warrants example is the conversion from counts to relative abundance when preparing bioindicator data for visualization and analysis in **tidypaleo**. When each

row represents a measurement, this can be accomplished using `dplyr::group_by()` and `dplyr::mutate()`.

```
R> kellys_lake_cladocera_long %>%
+   group_by(location, depth_mid) %>%
+   mutate(relative_abundance = count / sum(count) * 100) %>%
+   ungroup()

# A tibble: 875 x 7
  location depth_start depth_end depth_mid taxon      count relative_abundan~
  <chr>      <dbl>      <dbl>      <dbl> <chr>      <dbl>      <dbl>
1 KLY17-2      0          0.5        0.25 Bosmina~    61         59.2
2 KLY17-2      0.5         1          0.75 Bosmina~    63         51.2
3 KLY17-2      1.5         2          1.75 Bosmina~    39         47.6
4 KLY17-2      3           3.5        3.25 Bosmina~    36         39.6
5 KLY17-2      4.5         5          4.75 Bosmina~    44         53.0
6 KLY17-2      6           6.5        6.25 Bosmina~    45         51.1
7 KLY17-2      7.5         8          7.75 Bosmina~    55         59.1
8 KLY17-2      9           9.5        9.25 Bosmina~    61         58.1
9 KLY17-2     10.5        11         10.8 Bosmina~    61         55.5
10 KLY17-2     12          12.5       12.2 Bosmina~    63         64.9
# ... with 865 more rows
```

3.2. Scales

Paleoenvironmental data in a one row per measurement structure has several common data types that should be scaled differently when passed to **ggplot2**. Discrete variables such as location, parameter, and sample groupings, such as zones, are well-represented by the existing discrete scales in **ggplot2**. Continuous variables require special consideration, including archive position (typically depth), age, relative abundance, and concentration values.

Position in the archive (e.g., depth) and its age (e.g., year common era (CE) or years before present) are values that are related by a monotonic transformation. Whereas position in the archive is known to a high degree of precision, age values are typically estimated, and communicating uncertainty around this value is essential. In the **tidypaleo** package, the relationship between archive position, age, and age uncertainty is represented by an `age_depth_model()`. An `age_depth_model()` is constructed using previously estimated age and depth values, and provides various options for interpolating and extrapolating. Default interpolation and extrapolation provides a reasonable approximation for visualization, although ideally these values should be provided at a high enough resolution so that interpolation and extrapolation is minimal. Age-depth models can be passed to `scale_(x|y)_depth_age()` and `scale_(x|y)_age_depth()`, which use **ggplot2**'s `sec_axis()` framework to communicate both age and depth (Figure 3). These scales also enforce the convention that time should be visualized from bottom to top when on the *y*-axis, and from left to right when on the *x*-axis.

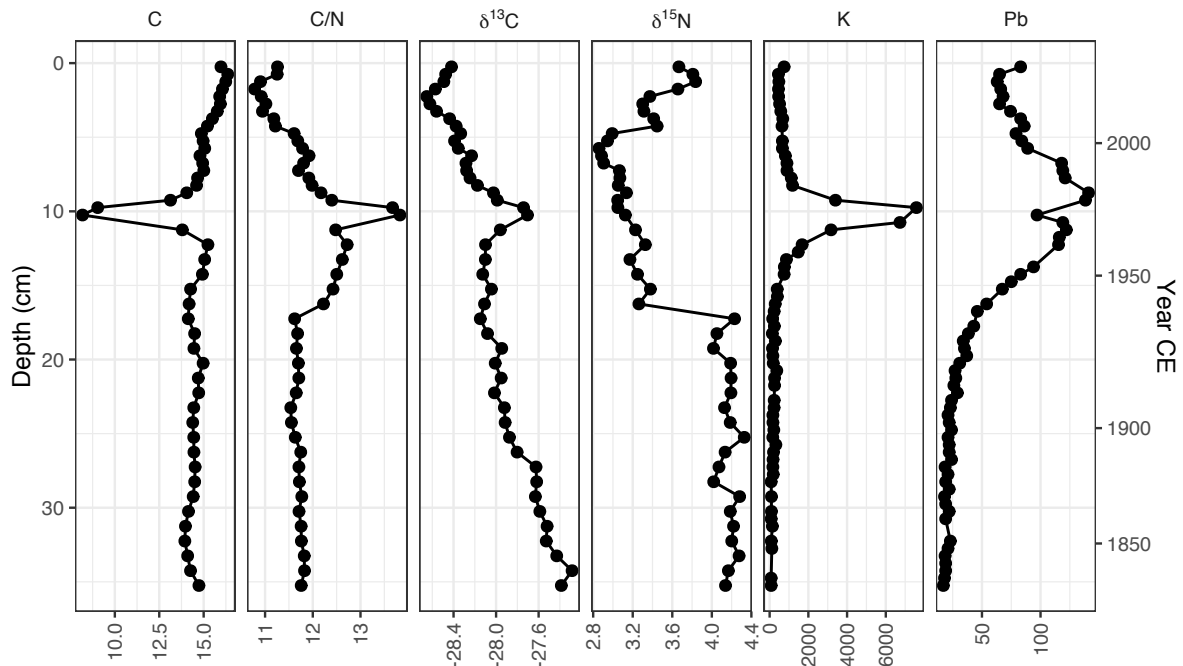


Figure 3: Age-depth transformation and scales applied to a stratigraphic plot of geochemical measurements.

```
R> data("kellys_lake_ages", package = "tidypaleo")
R> kellys_adm <- age_depth_model(depth = kellys_lake_ages$depth,
+   age = kellys_lake_ages$age_ad)
R> kellys_geochem_plot +
+   scale_y_depth_age(kellys_adm, age_name = "Year CE")
```

Relative abundance of microfossils is a common type of value in paleoenvironmental diagrams. Relative abundance values are always zero or positive, and breaks should always occur at the same intervals on all panels. Because negative values are impossible, zero should always be the minimum limit. Expansion below zero can be misleading and tends to produce unnecessary space. Expansion above the maximum value should be a fixed amount (rather than the default 5%) to keep spacing between panels uniform. These defaults are encapsulated in the `scale_(x|y)_abundance()` scales, which wrap `ggplot2::scale_(x|y)_continuous()` to produce the optimal labels, limits, and expansion for relative abundance values.

Concentration values are also common in paleoenvironmental diagrams. Concentration values are theoretically always positive, although in practice values below detection or quantification limits are frequently (if incorrectly) encoded as zeroes. Concentration values are usually well-represented by the default continuous scale, which scales to the minimum and maximum of the data. Occasionally it is useful to convey the relative change in concentration between parameters, in which setting the bottom limit to zero (`limits = c(0, NA)`) is appropriate. Similarly, log-scales are appropriate for some parameters (e.g., ^{210}Pb activities). Because the scale modifications required for concentration values are minimal, the **tidypaleo** package does not provide specific scales for concentration values.

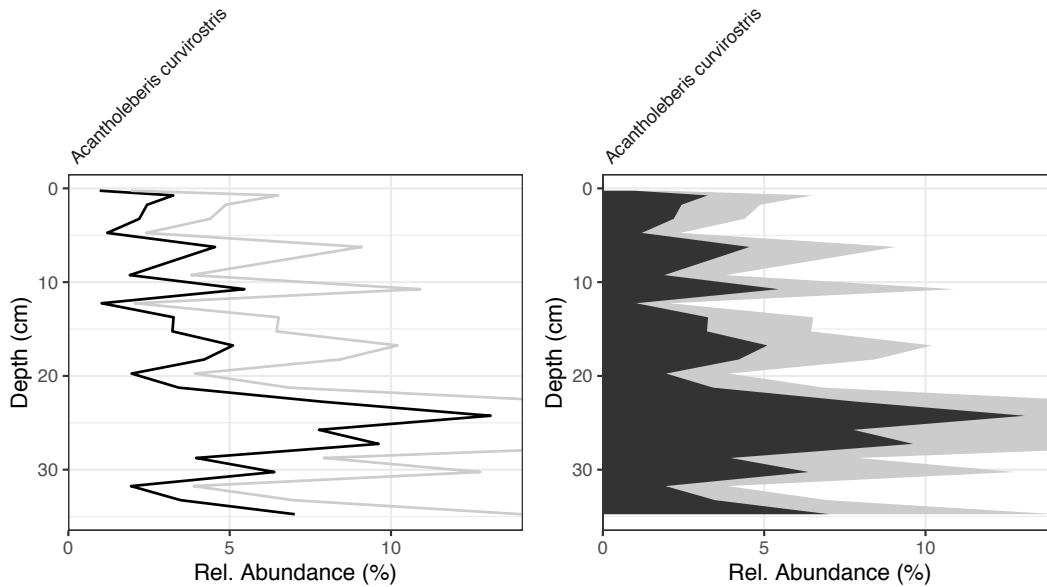


Figure 4: Exaggerated line and area geometries for highlighting relative change of parameters with a large change in magnitude.

3.3. Geometries

Two common visual representations of relative abundance values are difficult to recreate using existing **ggplot2** geometries. First, vertical or horizontal segments drawn from the x - or y -axis to the relative abundance value are common and recommended for some types of data (Juggins and Telford 2012). The **tidypaleo** package provides `geom_col_segs()` to create this type of visual representation. This geometry is implemented as a subclass of `'ggplot2::GeomSegment'`, and is parameterized identically to `geom_col()`, `geom_area()`, `geom_point()`, and `geom_line()`.

Second, “exaggerations” are common to communicate low-magnitude variability when one or more large relative abundance values obscure this trend. From a Grammar of Graphics perspective, these “exaggerations” are statistics, as they modify the original data in such a way that existing graphical representations can be used to draw them (Wilkinson 2005). In **tidypaleo** we implement exaggerations as subclasses of the existing `'ggplot2::Geom'` classes, because implementing them as subclasses of `'ggplot2::Stat'` results in the scales expanding to include all exaggerated values. The **tidypaleo** package provides `geom_point_exaggerate()`, `geom_line_exaggerate()`, and `geom_area_exaggerate()` to create this type of graphical representation (Figure 4).

```
R> kellys_demo_base <- kellys_lake_cladocera %>%
+   filter(taxon == "Acantholeberis curvirostris") %>%
+   ggplot(aes(x = rel_abund, y = depth)) +
+   scale_y_reverse() +
+   facet_abundanceh(vars(taxon)) +
+   scale_x_abundance(breaks = waiver(),
+     expand = expansion(add = c(0, 1))) +
+   labs(x = "Rel. Abundance (%)", y = "Depth (cm)")
```

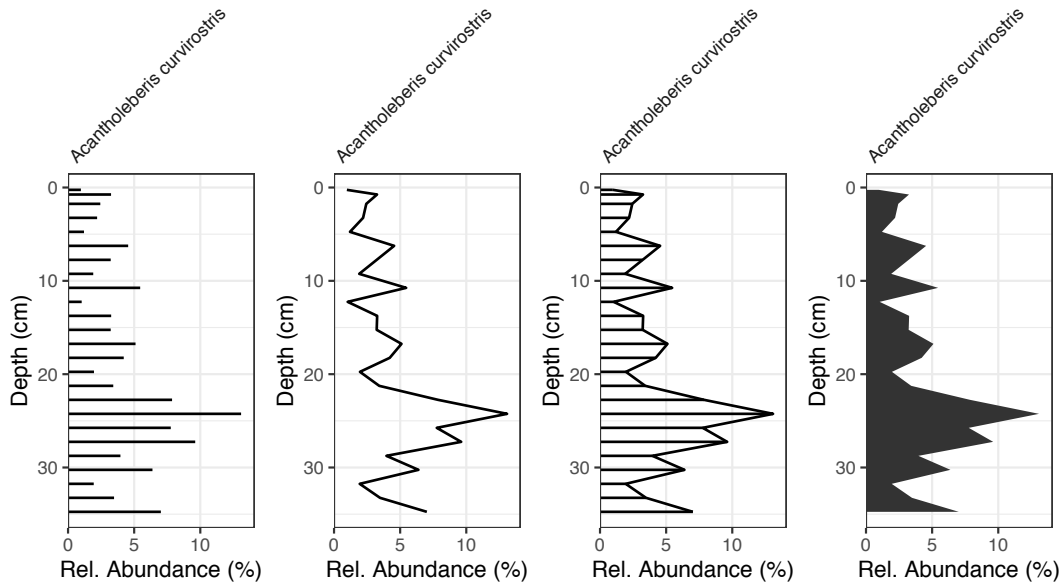



Figure 5: Column segment, line, area, and combinations commonly used to represent relative abundance values on stratigraphic diagrams.

```
R> patchwork::wrap_plots(
+   kellys_demo_base +
+     geom_lineh() +
+     geom_lineh_exaggerate(exaggerate_x = 2, col = "grey80"),
+   kellys_demo_base +
+     geom_areah_exaggerate(exaggerate_x = 2, fill = "grey80") +
+     geom_areah(),
+   nrow = 1)
```

In recent **ggplot2** versions ($\geq 3.3.0$), horizontal geometries such as `geom_errorbar()`, `geom_col()`, `geom_line()`, `geom_smooth()`, `geom_ribbon()`, and `geom_area()` can be oriented vertically using `orientation = "y"` (in previous versions of **ggplot2**, vertically-oriented diagrams were more challenging to create). The **tidypaleo** package provides `geom_errorbarh()`, `geom_colh()`, `geom_col_segsh()`, `geom_lineh()`, `geom_smoothh()`, `geom_ribbonh()`, and `geom_areah()` to specifically handle vertically-oriented plots, some of which are re-exported from the **ggplot2** and **ggstance** packages (Wickham 2016; Henry, Wickham, and Chang 2020). While these wrappers are minimal, they reflect that vertically-oriented plots are the most common type of paleoenvironmental diagram and make it easier to switch the orientation of a diagram if required (Figure 5).

```
R> patchwork::wrap_plots(
+   kellys_demo_base + geom_col_segsh(),
+   kellys_demo_base + geom_lineh(),
+   kellys_demo_base + geom_col_segsh() + geom_lineh(),
+   kellys_demo_base + geom_areah(),
+   nrow = 1)
```

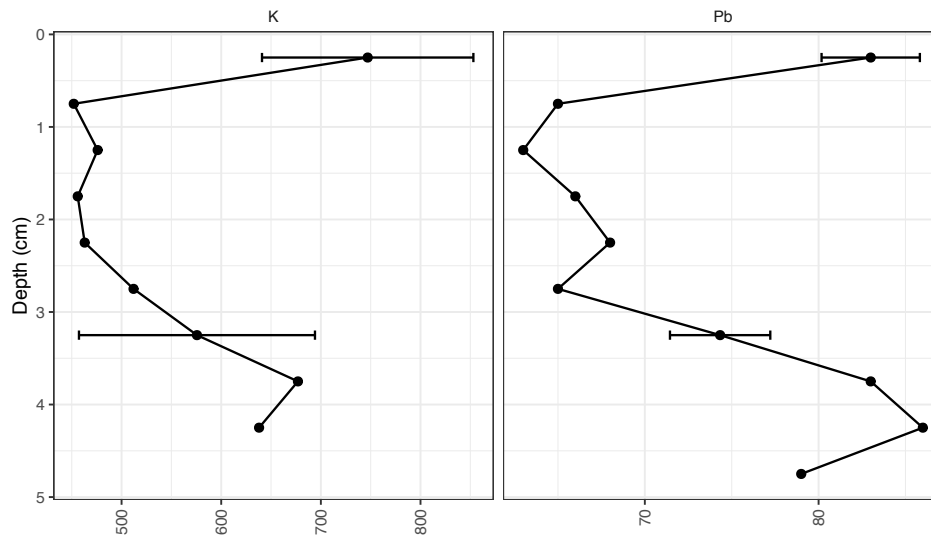


Figure 6: Communicating error in a stratigraphic diagram using error bars.

Communicating error is important for geochemical data, age-depth models, and quantities calculated from bioindicator data such as the results of inference models. Communicating error in stratigraphic diagrams is infrequently discussed and we suspect that the complexity of adding error bars to multi-panel diagrams has contributed to their infrequent use. Geometries that communicate error (e.g., `geom_errorbar()` and `geom_errorbarh()`) are already included in **ggplot2**; however, the data structure used by **tidypaleo** is well-suited to using them with minimal effort (Figure 6), particularly for those already familiar with adding error bars to diagrams created in **ggplot2**.

```
R> kellys_lake_geochem %>%
+   filter(depth <= 5, param %in% c("K", "Pb")) %>%
+   ggplot(aes(value, depth)) +
+   geom_lineh() +
+   geom_point() +
+   geom_errorbarh(aes(xmin = value - error, xmax = value + error),
+   height = 0.1) +
+   facet_geochem_wrap(vars(param)) +
+   scale_y_reverse() +
+   labs(y = "Depth (cm)", x = NULL)
```

3.4. Facets

In the Grammar of Graphics, facets are defined as displaying subsets of a data set in panels of the same graphic (Wilkinson 2005). Using a data structure with one row per measurement, plotting multiple parameters on same plot using facets is a natural way to represent these data. In **ggplot2**, facets also coordinate the scales and labels for each panel. As noted in Section 3.2, different parameter types can have different scaling requirements. Furthermore, proper labeling of geochemical parameters and species names can be challenging. Facets are a useful way to solve these challenges for diagrams with a common data type. The **tidypaleo**

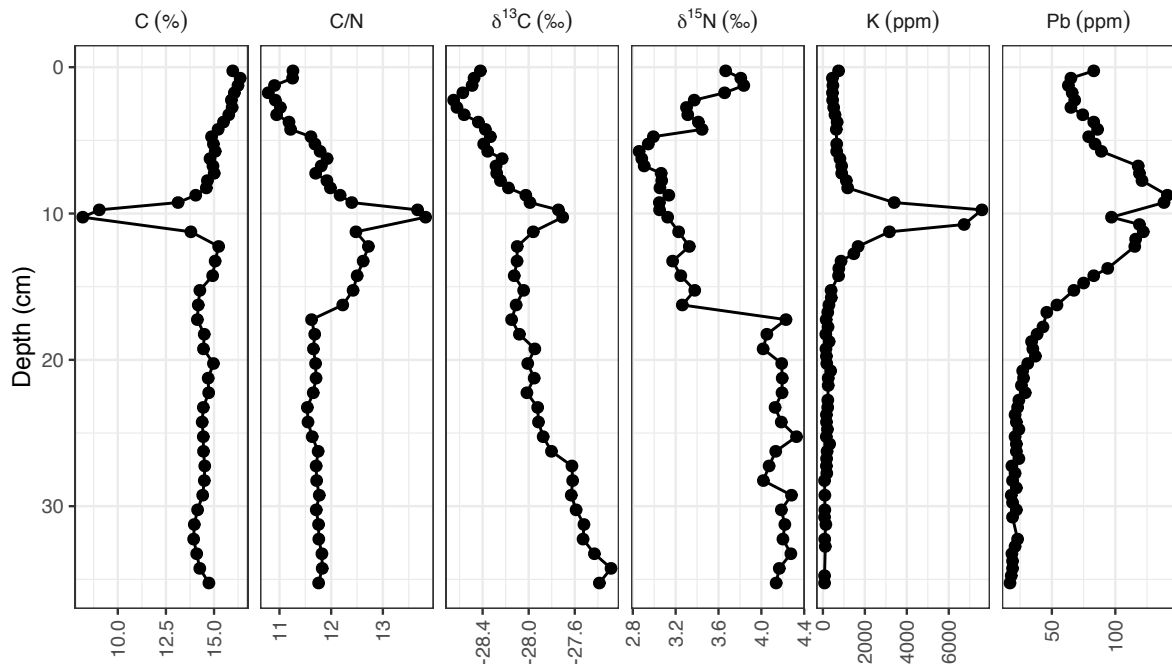


Figure 7: Adding units to multi-parameter plots using the geochemistry labeller.

package provides facet types for two common cases: relative abundance data and geochemical data.

Facets for relative abundance data (`facet_abundance()` and `facet_abundanceh()`) wrap `ggplot2::facet_grid()`, applying best-practice rules for the relationship between abundance scales. When plotted, relative abundance values on each panel should have the same weight (i.e., 5% on one panel should take up the same amount of space as 5% on another panel; `space = "free_x"` or `space = "free_y"`). Facet labels must have species names italicized for publication in most journals, but modifiers (e.g., strain III) must not be italicized. This constraint can be accommodated by setting the default labeller to a function that understands the form of most species input (`label_species()`). Finally, facet labels are typically too long to fit horizontally above each panel and must be rotated to be legible. While rotating a facet label is possible using **ggplot2** theme modifications, eliminating the horizontal clip is not. Control over the clip parameter of the strip text is likely in a future **ggplot2** version, however packaging both as part of the facet is a way make this implementation detail transparent to the user.

Panels representing different geochemical parameters do not require `space = "free_x"` or `space = "free_y"`, and thus can wrap either `facet_wrap()` or `facet_grid()`. The **tidypaleo** package provides `facet_geochem_wrap()`, `facet_geochem_grid()`, `facet_geochem_wrap_h()`, and `facet_geochem_grid_h()` for geochemical measurements (Figure 1). Scales should be independent between panels representing different parameters, as fixed scales could hide meaningful trends in parameters with a smaller absolute magnitude (for this reason, it is also generally not appropriate to display two geochemical parameters on the same panel). Units of measurement may be different between panels, and thus need to be included in the facet label (Figure 7). Furthermore, many common parameter names contain non-ASCII char-

acters that require R `plotmath` to display on all graphics devices. Both of these labeling constraints are practically difficult to achieve, so we include `label_geochem()` as the default labeller to convert common ASCII representations of parameter names to parseable `plotmath` and provide an interface to specify measurement units. Finally, including many parameters on a vertically-oriented plot inevitably results in overlapping x -axis labels. Because of this, we also rotate x -axis labels by 90 degrees by default in `facet_geochem_wrap()` and `facet_geochem_gridh()`.

```
R> kellys_geochem_plot +
+   facet_geochem_gridh(vars(param), units = c("C" = "%", "C/N" = NA,
+     "d13C" = "%", "d15N" = "%", "Pb" = "ppm", "K" = "ppm"))
```

While these facets could be implemented as subclasses of ‘`ggplot2::FacetGrid`’ and of ‘`ggplot2::FacetWrap`’, both relative abundance and concentration facets are instead implemented as wrappers around the `ggplot2::facet_grid()` and `ggplot2::facet_wrap()` functions that also add the appropriate scale and theme elements. These labellers, scales, and theme elements are also available as separate components should the user wish to use a different base facet type (e.g., from an extension package).

The facets included in the **tidypaleo** package do not solve the problem of paleoenvironmental diagrams that display more than one data type; however, the **patchwork** package provides syntax to align multiple **ggplot2** plots and assign common elements among them (Pedersen 2020). Creating diagrams in this way imposes the constraint that plots with the same data type must be grouped. This constraint makes it easier to describe any differences in plot scales (e.g., for panels where a log scale is appropriate), and we do not think a more complex implementation of a facet would result in better diagrams or syntax.

3.5. Theme elements

This package includes `theme_paleo()`, which is a minimally-modified version of the function `ggplot2::theme_bw()` that removes the grey background of panel labels. This reflects the look and feel of these diagrams created by other software. Elements of the abundance and geochemical facets are also available as theme modifiers, including `rotated_axis_labels()` and `rotated_facet_labels()`.

3.6. Statistical helpers

The definition of data used by **tidypaleo** is different than the type of data that is needed to compute common multivariate summaries such as ordinations and/or stratigraphically-constrained cluster analysis (Juggins 2020; Grimm 1987). To facilitate visualizing the results of these analyses, **tidypaleo** provides a framework for transforming one-row-per-measurement data into a data frame suitable for input to most multivariate summaries (generally one row per sample). The **tidypaleo** package exposes the most common analyses (principal components analysis (PCA) and **CONISS**) and transformation options with the `nested_*()` functions. The example below computes a PCA for each `location` in the data set using the `nested_data()` and `nested_prcomp()` functions.

```
R> keji_lakes_prcomp <- keji_lakes_plottable %>%
+   group_by(location) %>%
```

```
+ nested_data(qualifiers = depth, key = taxon, value = rel_abund,
+   trans = sqrt) %>%
+ nested_prcomp()
```

The purpose of the `nested_data()` function is to compute a data frame with one row per sample and one column per predictor, which is usually a geochemical parameter or taxon. The options provided by `nested_data()` include (1) the transformation to apply to each column, (2) the filter to apply to the data, and (3) the selection criteria to apply to the data. Option (1) can be used to apply scaling appropriate for a specific statistical analysis (e.g., `scale()` for PCA); options (2) and (3) can be used to handle non-finite values resulting from parameter measurements on different samples. The `keji_lakes_plottable` data set does not contain any non-finite values and are on a common scale (relative abundance), but applying a square-root transformation makes euclidean distance (preserved by PCA) a reasonable measure of dispersion between samples (Legendre and Birks 2012).

A single data frame can be obtained from the result of a `nested_*()` function by calling `unnested_data()` on one or more list-columns whose values contain nested data frames with the same number of rows. For example, the object returned by `nested_prcomp()` has columns `qualifiers` and `scores`, both of which are aligned row wise (each contain one row per sample). The `unnested_data()` function can combine these columns into a single data frame whose value is more useful.

```
R> keji_lakes_prcomp %>%
+   select(location, qualifiers, data, scores)

# A tibble: 2 x 4
  location      qualifiers      data      scores
  <chr>         <list>      <list>    <list>
1 Beaverskin Lake <tibble [17 x 2]> <tibble [17 x 6]> <tibble [17 x 6]>
2 Peskawa Lake   <tibble [20 x 2]> <tibble [20 x 5]> <tibble [20 x 5]>

R> keji_lakes_prcomp %>%
+   select(location, model, variance, loadings)

# A tibble: 2 x 4
  location      model      variance      loadings
  <chr>         <list>    <list>      <list>
1 Beaverskin Lake <prcomp> <tibble [6 x 6]> <tibble [6 x 7]>
2 Peskawa Lake   <prcomp> <tibble [5 x 6]> <tibble [5 x 6]>

R> keji_lakes_prcomp %>%
+   unnested_data(qualifiers, scores)

# A tibble: 37 x 9
  location      depth row_number  PC1    PC2    PC3    PC4    PC5
  <chr>         <dbl>    <int>  <dbl> <dbl> <dbl> <dbl> <dbl>
1 Beaverskin Lake 0.125      1 -3.06  1.64 -0.775 0.0215 0.497
```

```

2 Beaverskin Lake 0.375      2 -2.36    1.51      0.0229  0.615  -0.188
3 Beaverskin Lake 0.825      3 -2.67   -0.131    0.704  -0.467  0.330
4 Beaverskin Lake 2.12       4 -3.03   -0.165    0.440  -0.134  -0.0538
5 Beaverskin Lake 3.12       5 -2.08   -0.370   -0.333  0.0801  -0.393
6 Beaverskin Lake 4.12       6 -1.29    0.00973  -0.498  -0.614  -0.544
7 Beaverskin Lake 5.38       7 -0.578  -0.966    0.190  0.455   0.218
8 Beaverskin Lake 6.38       8  0.0574  -0.832    0.476  0.363  -0.124
9 Beaverskin Lake 7.62       9  0.343  -0.766    0.295  -0.0171  0.254
10 Beaverskin Lake 9.12      10  0.819  -0.828   -0.142  0.322  -0.319
# ... with 27 more rows, and 1 more variable: PC6 <dbl>

```

Whereas experienced users may prefer to use their existing knowledge of data frame manipulation to fit one or more ordinations and/or cluster analyses to their data, new users may prefer the `nested_data()`, `nested_prcomp()`, and `nested_chlust()` functions. These functions are designed to produce reasonable results for common analyses in most situations; however, as analyses become more specialized and users become more familiar with data frame manipulation, we expect that users will transition to more generic approaches.

4. Example

A motivating example for the development of this package was the ability to create stratigraphic diagrams of multiple archives with statistical summaries. For relative abundance data, it is common to plot the results of an ordination and a cluster analysis alongside the raw data. For this example, we will use diatom count data from lakes in Kejimikujik National Park, Nova Scotia, Canada sourced from the Neotoma paleoecological database ([Goring *et al.* 2015](#); [Ginn, Cumming, and Smol 2007](#)).

The stratigraphic diagram for these lakes is an example where exaggerated geometries are useful: while common scales for the same taxa between lakes correctly communicates the difference in magnitude, relative trends are difficult to assess for some taxa without the exaggerated geometry (`geom_areah_exaggerate()`). Relative abundance scales with comparable size across panels are added via `facet_abundanceh()`, which also correctly labels taxa with a rotated facet label.

```

R> keji_plot <- ggplot(keji_lakes_plottable, aes(x = rel_abund, y = depth)) +
+   geom_areah_exaggerate(exaggerate_x = 5, alpha = 0.2) +
+   geom_areah() +
+   scale_y_reverse() +
+   facet_abundanceh(vars(taxon), grouping = vars(location),
+     scales = "free") +
+   labs(x = "Relative abundance (%)", y = "Depth (cm)")

```

Using `nested_data()` and `nested_chclust_coniss()`, it is possible to calculate a cluster analysis for both lakes. Here we use a square-root transformation such that the euclidean distance measure used by default in `nested_chclust_coniss()` is useful ([Legendre and Birks 2012](#)).

One measure of the quality of a stratigraphically-constrained cluster analysis is a broken-stick analysis to obtain the number of statistically plausible groups ([Bennett 1996](#)).

These results can be obtained by unnesting the `broken_stick` column in the result of `nested_coniss_chclust()`. For both cores, the cluster analysis identified 3 groups whose dispersion was more than would be expected from a cluster analysis of a random shuffle of the samples.

```
R> keji_coniss %>%
+   unnested_data(broken_stick) %>%
+   group_by(location) %>%
+   slice(1:4)

# A tibble: 8 x 4
# Groups:   location [2]
  location          n_groups dispersion broken_stick_dispersion
  <chr>              <int>      <dbl>                <dbl>
1 Beaverskin Lake     2         9.27                  5.29
2 Beaverskin Lake     3         3.24                  3.72
3 Beaverskin Lake     4         1.53                  2.94
4 Beaverskin Lake     5         2.17                  2.42
5 Peskawa Lake        2        14.4                  6.26
6 Peskawa Lake        3         4.59                  4.50
7 Peskawa Lake        4         3.27                  3.61
8 Peskawa Lake        5         2.38                  3.03
```

The dendrogram associated with these cluster analyses can be added to a plot using function `layer_dendrogram()`. Because the scale of the dendrogram is not in relative abundance, a separate plot is needed

```
R> dendro_plot <- ggplot() +
+   layer_dendrogram(keji_coniss, aes(y = depth), label = "CONISS") +
+   scale_y_reverse() +
+   facet_grid(vars(location), vars(label), scales = "free_y") +
+   labs(y = NULL, x = "Dispersion")
```

These plots can be combined using `patchwork::wrap_plots()`, resulting in the finished stratigraphic plot (Figure 8). In this case we have chosen to align the relative abundance scales to communicate trend between cores at the expense of trend within each core being less clear for some taxa. The `patchwork::wrap_plots()` function can also be used to stack stratigraphic diagrams without aligning the relative abundance scales between cores.

```
R> wrap_plots(
+   keji_plot +
+     theme(strip.background = element_blank(),
+           strip.text.y = element_blank()),
+   dendro_plot +
+     theme(axis.text.y.left = element_blank(),
+           axis.ticks.y.left = element_blank()) +
+   labs(y = NULL),
+   nrow = 1,
+   widths = c(6, 1))
```

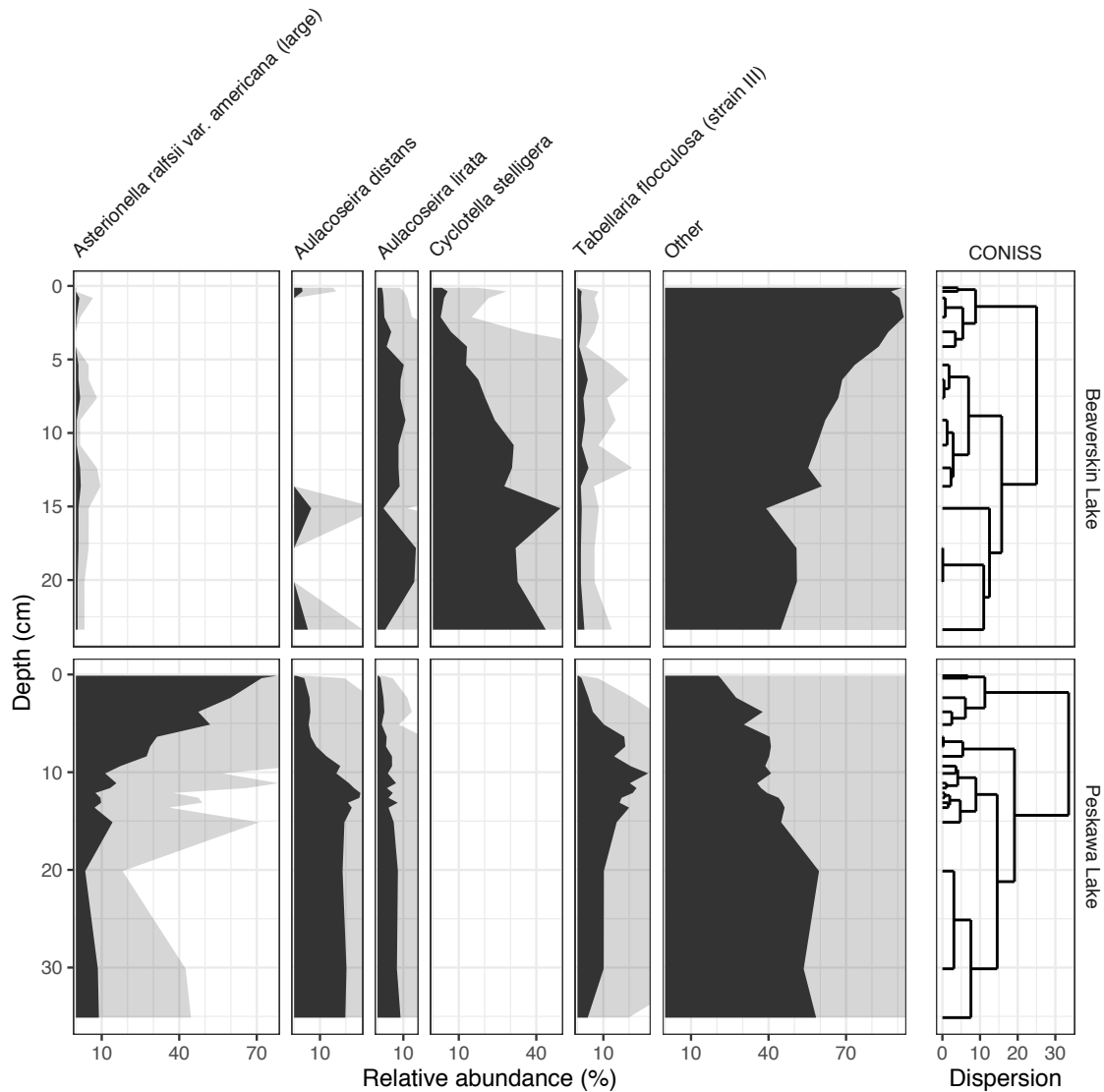


Figure 8: Relative abundances and cluster analysis of microfossil diatom relative abundance from two lakes in Kejimikujik National Park, Nova Scotia, Canada.

5. Discussion

Several programs are available that are capable of producing paleoenvironmental diagrams of the type produced by this package. In R, the **analogue** and **rioja** packages have functions that specifically produce stratigraphic diagrams (Simpson 2007; Juggins 2020). Functions in both these packages require data with parameters as columns and use dozens of arguments to a single function to control the various options for display. This makes the functions highly useful for specific types of data, but limit the ease with which multiple plots can be aligned and the ease with which error can be communicated.

Several desktop applications are also commonly used to create paleoenvironmental diagrams, including **Tilla*Graph** (Grimm 2016) and **C2** (Juggins 2011). **Tilla*Graph** creates graphics

that are highly customizable, and the program includes the ability to visualize core lithology in addition to plots of multiple parameters. **C2** also creates graphics that are highly customizable, and the program includes the ability to use two y-axes next to each other to clearly communicate possible changes to the sedimentation rate. Both programs share the disadvantages of many desktop applications, notably, these applications are only available for Windows, and the need for point-and-click edits introduces non-reproducibility if the figure data must be updated.

The **tidypaleo** package has attempted to build on the best from each of these software packages: functions from **analogue** and **rioja** produce excellent reproducible representations of relative abundance data and include the ability to visualize hierarchical clusters as zones or dendrograms as part of the visualization. **Tilla*Graph** and **C2** produce visually appealing diagrams and have interfaces that are well-suited to a non-technical audience. While **tidypaleo** does require coding to make a publishable diagram, there are a large number of resources from which users can draw to learn **ggplot2** (Wickham and Grolemund 2017; Wickham 2016; Healy 2018). We think this also benefits users, who can re-use **ggplot2** concepts from **tidypaleo** to create high-quality diagrams in other disciplines.

6. Conclusions

The **tidypaleo** package is an extension of **ggplot2** for R statistical software that provides a number of reusable components for visualizing paleoenvironmental data (Wickham 2016; R Core Team 2021). These components are based on a data structure in the form of one row per measurement, which allows existing concepts in **ggplot2** and the Grammar of Graphics (Wilkinson 2005) to be used to create high-quality paleoenvironmental diagrams. It is our hope that this software will increase the reproducibility of figures included in paleoenvironmental publications, and demonstrate best practices for creating a minimal discipline-specific wrapper around the **ggplot2** package.

Acknowledgments

We would like to acknowledge the core developer team of the **ggplot2** package, including Hadley Wickham, Thomas Lin Pedersen, Claus Wilke, Kara Woo, and Hiroaki Yutani, for their tireless work maintaining the package and implementing the features that we build upon in **tidypaleo**. This work was funded by the Natural Sciences and Engineering Research Council of Canada (NSERC) postgraduate scholarship program (D.W. Dunnington), the NSERC Halifax Water Industrial Research Chair program (G.A. Gagnon, NSERC Grant No. IRCPJ: 349838-16), and the CBRM Water Utility (J. Kurek, I. Spooner). We would like to thank Anthony Mazzocca (CBRM Water Utility) for sharing his knowledge of Kellys Lake and for his assistance in the field.

References

- Bennett KD (1996). “Determination of the Number of Zones in a Biostratigraphical Sequence.” *New Phytologist*, **132**(1), 155–170. doi:10.1111/j.1469-8137.1996.tb04521.x.

- Charles DF, Binford MW, Furlong ET, Hites RA, Mitchell MJ, Norton SA, Oldfield F, Paterson MJ, Smol JP, Uutala AJ (1990). “Paleoecological Investigation of Recent Lake Acidification in the Adirondack Mountains, NY.” *Journal of Paleolimnology*, **3**(3), 195–241. doi:10.1007/bf00219459.
- Cohen AS (2003). *Paleolimnology : The History and Evolution of Lake Systems*. Oxford University Press, Oxford.
- Dunnington DW (2022). *tidypaleo: Tidy Tools for Paleoenvironmental Archives*. R package version 0.1.2, URL <https://CRAN.R-project.org/package=tidypaleo>.
- Dunnington DW, Roberts S, Norton SA, Spooner IS, Kurek J, Kirk JL, Muir DCG, White CE, Gagnon GA (2020). “The Distribution and Transport of Lead over Two Centuries as Recorded by Lake Sediments from Northeastern North America.” *Science of the Total Environment*, **737**, 140212. doi:10.1016/j.scitotenv.2020.140212.
- Dunnington DW, Spooner IS (2018). “Using a Linked Table-Based Structure to Encode Self-Describing Multiparameter Spatiotemporal Data.” *FACETS*, **3**(1), 326–337. doi:10.1139/facets-2017-0026.
- Ginn BK, Cumming BF, Smol JP (2007). “Long-Term Lake Acidification Trends in High- and Low-Sulphate Deposition Regions from Nova Scotia, Canada.” *Hydrobiologia*, **586**(1), 261–275. doi:10.1007/s10750-007-0644-3.
- Goring S, Dawson A, Simpson G, Ram K, Graham R, Grimm E, Williams J (2015). “Neotoma: A Programmatic Interface to the Neotoma Paleoecological Database.” *Open Quaternary*, **1**(1), Art. 2. doi:10.5334/oq.ab.
- Grimm E (2016). *Tilia*Graph*. Version 2.6.1, URL <https://www.tiliait.com/>.
- Grimm EC (1987). “CONISS: A Fortran 77 Program for Stratigraphically Constrained Cluster Analysis by the Method of Incremental Sum of Squares.” *Computers & Geosciences*, **13**(1), 13–35. doi:10.1016/0098-3004(87)90022-7.
- Healy K (2018). *Data Visualization: A Practical Introduction*. Princeton University Press, Princeton. doi:10.23943/princeton/9780691181141.003.0003.
- Henry L, Wickham H, Chang W (2020). *ggstance: Horizontal ggplot2 Components*. R package version 0.3.5, URL <https://CRAN.R-project.org/package=ggstance>.
- Juggins S (2011). *C2*. Version 1.7.7, URL <https://www.staff.ncl.ac.uk/stephen.juggins/software/C2Home.htm>.
- Juggins S (2020). *rioja: Analysis of Quaternary Science Data*. R package version 0.9-26, URL <https://CRAN.R-project.org/package=rioja>.
- Juggins S, Telford RJ (2012). “Exploratory Data Analysis and Data Display.” In HJB Birks, AF Lotter, S Juggins, JP Smol (eds.), *Tracking Environmental Change Using Lake Sediments*, volume 5, pp. 123–141. Springer-Verlag, Dordrecht. doi:10.1007/978-94-007-2745-8_5.

- Legendre P, Birks HJB (2012). “From Classical to Canonical Ordination.” In HJB Birks, AF Lotter, S Juggins, JP Smol (eds.), *Tracking Environmental Change Using Lake Sediments*, volume 5, pp. 201–248. Springer-Verlag, Dordrecht. doi:10.1007/978-94-007-2745-8_8.
- Pedersen TL (2020). **patchwork**: *The Composer of Plots*. R package version 1.1.1, URL <https://github.com/thomasp85/patchwork>.
- R Core Team (2021). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna. URL <https://www.R-project.org/>.
- Simpson GL (2007). “Analogue Methods in Palaeoecology: Using the **analogue** Package.” *Journal of Statistical Software*, **22**(2), 1–29. doi:10.18637/jss.v022.i02.
- Smol JP (2009). *Pollution of Lakes and Rivers: A Paleoenvironmental Perspective*. John Wiley & Sons, New York.
- Smol JP (2010). “The Power of the Past: Using Sediments to Track the Effects of Multiple Stressors on Lake Ecosystems.” *Freshwater Biology*, **55**, 43–59. doi:10.1111/j.1365-2427.2009.02373.x.
- Vincent JH, Cwynar LC (2016). “A Temperature Reversal within the Rapid Younger Dryas-Holocene Warming in the North Atlantic?” *Quaternary Science Reviews*, **153**, 199–207. doi:10.1016/j.quascirev.2016.10.005.
- Wickham H (2014). “Tidy Data.” *Journal of Statistical Software*, **59**(10). doi:10.18637/jss.v059.i10.
- Wickham H (2016). **ggplot2**: *Elegant Graphics for Data Analysis*. Springer-Verlag, New York.
- Wickham H (2021). **tidyr**: *Tidy Messy Data*. R package version 1.1.4, URL <https://CRAN.R-project.org/package=tidyr>.
- Wickham H, Bryan J (2019). **readxl**: *Read Excel Files*. R package version 1.3.1, URL <https://CRAN.R-project.org/package=readxl>.
- Wickham H, Grolemund G (2017). *R for Data Science: Import, Tidy, Transform, Visualize, and Model Data*. O’Reilly Media, Sebastopol.
- Wilkinson L (2005). *The Grammar of Graphics*. Statistics and Computing. Springer-Verlag, New York.
- Williams JW, Grimm EC, Blois JL, Charles DF, Davis EB, Goring SJ, Graham RW, Smith AJ, Anderson M, Arroyo-Cabrales J, Ashworth AC, Betancourt JL, Bills BW, Booth RK, Buckland PI, Curry BB, Giesecke T, Jackson ST, Latorre C, Nichols J, Purdum T, Roth RE, Stryker M, Takahara H (2018). “The Neotoma Paleocology Database, a Multiproxy, International, Community-Curated Data Resource.” *Quaternary Research*, **89**(1), 156–177. doi:10.1017/qua.2017.105.

Affiliation:

Dewey W. Dunnington
Dalhousie University
1360 Barrington St. Halifax, NS, Canada
E-mail: dewey.dunnington@dal.ca
URL: <https://fishandwhistle.net/>