{ordr}: Toward Tidy Principles for Matrix-Decomposed Data

New Developments in Graphing Multivariate Data

Section on Statistical Graphics • Joint Statistical Meetings 2022

Jason Cory Brunson

Laboratory for Systems Medicine

Division of Pulmonary, Critcial Care, and Sleep Medicine • University of Florida

2022 Aug 8

Acknowledgments

Development

Emily Paul (UPenn)

Discussion

Joyce Robbins (Columbia)

Applications

Tom Agresta (UConn)
Ritchie Vaughan (UVA)
Martinna Bertolini (UFRJ)
Carol Mathews (UF)

Land

Potano tribe $\subset$ Timucua people

Resources

UCONN
HEALTH

UF
UNIVERSITY of
FLORIDA

Slideshow

remark.js (Ole Petter Bang) + **xaringan** (Yihui Xie)
Catlab.jl (Evan Patterson &al)
Flaticon.com

Prerequisites

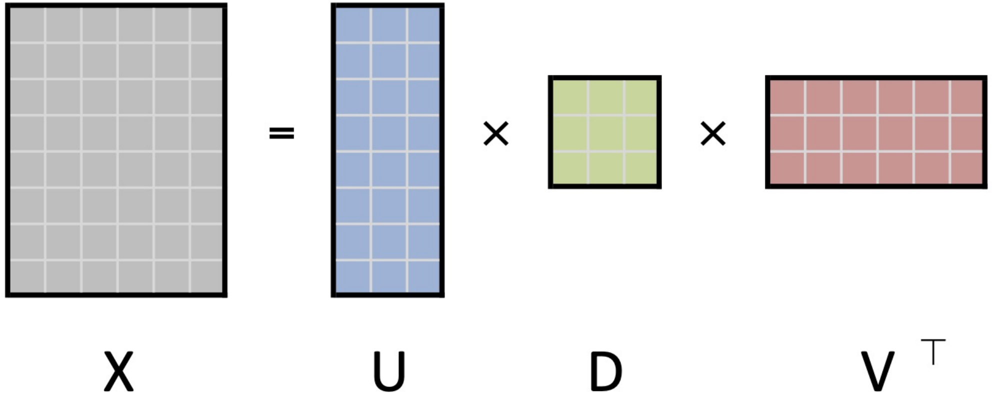
Singular value decomposition

1. Data matrix $X_{n \times p} \in \mathbb{R}^{n \times p}$
2. SVD into **inertia** and rotation matrices of left and right **singular vectors**:

$$X = U_{n \times k} \, D_{k \times k} \, {V_{p \times k}}^{\top}$$

3. Low-dimensional approximation via first r **ordinates**:

$$X \approx U_r \, D_r \, {V_r}^{\top}$$



Biplots

Principal coordinates $F = UD$ or $G = VD$ approximate the geometry of the rows or columns of X:

- Distances $\lVert f_{j\,\bullet} - f_{i\,\bullet} \rVert$ approximate distances $\sqrt{(x_{j\,1} - x_{i\,1})^2 + \cdots + (x_{j\,p} - x_{i\,p})^2}$
- Inner products $\frac{1}{n}\, g_{i\,\bullet} \cdot g_{j\,\bullet}$ approximate covariances $\operatorname{cov}(x_{\bullet\,i}, x_{\bullet\,j})$
    - Cosines $\cos(\arg(g_{i\,\bullet}, g_{j\,\bullet}))$ approximate correlations $\operatorname{cor}(x_{\bullet\,i}, x_{\bullet\,j})$

Standard coordinates U or V pair with principal coordinates to recover the values of X:

- Inner products $f_{i\,\bullet} \cdot v_{j\,\bullet}$ and $u_{i\,\bullet} \cdot g_{j\,\bullet}$ approximate entries x_{ij}

Biplots superimpose row and column coordinates with shares (a, b) of inertia:

$$\begin{bmatrix} U\ D^a \\ V\ D^b \end{bmatrix} \in \mathbb{R}^{(n+p)\times r}$$

Biplots

Principal coordinates $F = UD$ or $G = VD$ approximate the geometry of the rows or columns of X:

- Distances $\lVert f_{j\,\bullet} - f_{i\,\bullet} \rVert$ approximate distances $\sqrt{(x_{j\,1} - x_{i\,1})^2 + \cdots + (x_{j\,p} - x_{i\,p})^2}$
- Inner products $\frac{1}{n}\, g_{i\,\bullet} \cdot g_{j\,\bullet}$ approximate covariances $\operatorname{cov}(x_{\bullet\,i}, x_{\bullet\,j})$
  - Cosines $\cos(\arg(g_{i\,\bullet}, g_{j\,\bullet}))$ approximate correlations $\operatorname{cor}(x_{\bullet\,i}, x_{\bullet\,j})$

Standard coordinates U or V pair with principal coordinates to recover the values of X:

- Inner products $f_{i\,\bullet} \cdot v_{j\,\bullet}$ and $u_{i\,\bullet} \cdot g_{j\,\bullet}$ approximate entries x_{ij}

Biplots superimpose row and column coordinates with shares (a, b) of inertia:

$$\begin{bmatrix} U\ D^a \\ V\ D^b \end{bmatrix} \in \mathbb{R}^{(n+p)\times r}$$

Pick (at most) **two out of three**:

1. Approximate row geometry
2. Approximate column geometry
3. Approximate measured values

Pitch: Principal components analysis

PCA of Anderson's iris data

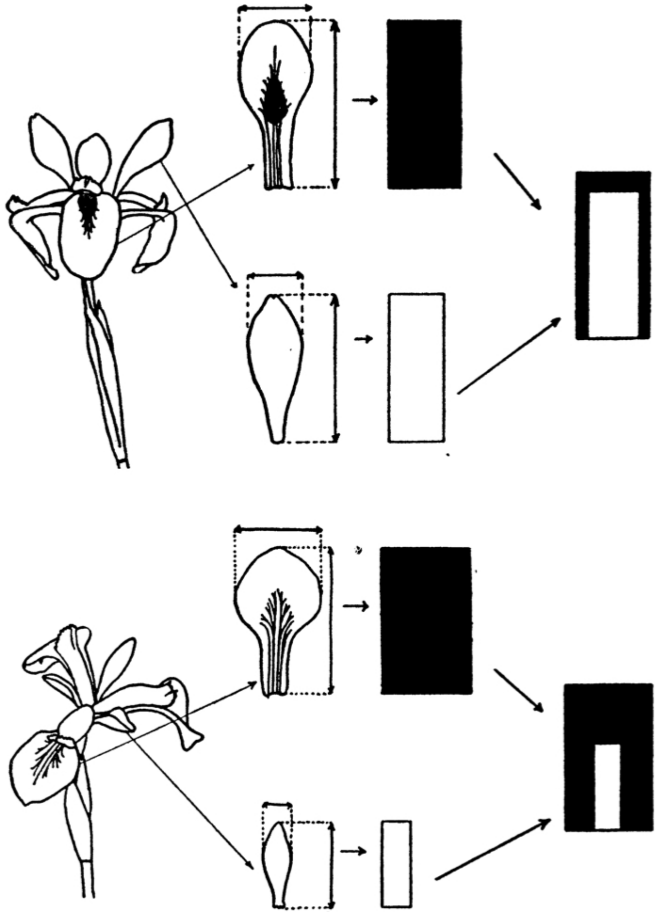


Fig. 8. Diagram illustrating how petal length and width, and sepal length and width are combined to form an ideograph. Above, *Iris virginica*; below, *I. versicolor*.

```
head(iris)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
## 5          5.0         3.6          1.4         0.2  setosa
## 6          5.4         3.9          1.7         0.4  setosa
```

```
summary(iris)
```

```
##   Sepal.Length    Sepal.Width     Petal.Length    Petal.Width   
##  Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100  
##  1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300  
##  Median :5.800   Median :3.000   Median :4.350   Median :1.300  
##  Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199  
##  3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800  
##  Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500  
##        Species  
##  setosa    :50  
##  versicolor:50  
##  virginica :50  
##                 
##                 
##                 
```

PCA of Anderson's iris data

```
(pca <- prcomp(iris[, 1:4]))
```

```
## Standard deviations (1, .., p=4):
## [1] 2.0562689 0.4926162 0.2796596 0.1543862
##
## Rotation (n x k) = (4 x 4):
##                      PC1         PC2         PC3        PC4
## Sepal.Length  0.36138659 -0.65658877  0.58202985  0.3154872
## Sepal.Width  -0.08452251 -0.73016143 -0.59791083 -0.3197231
## Petal.Length  0.85667061  0.17337266 -0.07623608 -0.4798390
## Petal.Width   0.35828920  0.07548102 -0.54583143  0.7536574
```

```
(pca_ord <- ordinate(iris, prcomp, cols = 1:4))
```

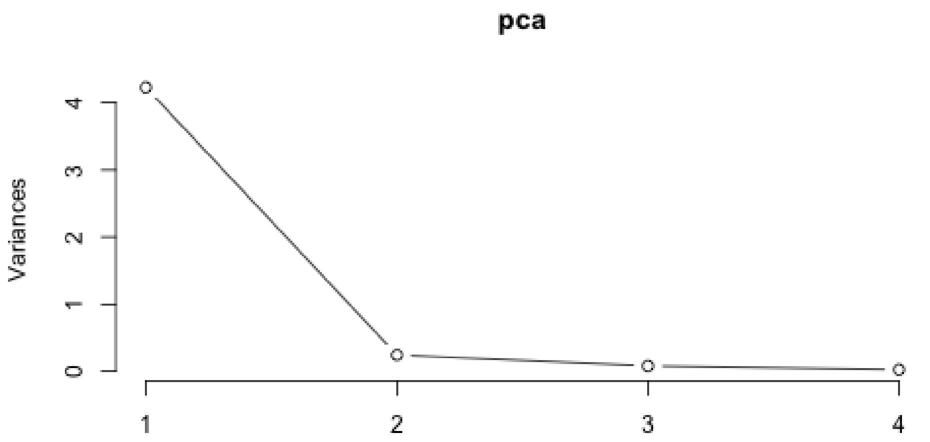
```
## # A tbl_ord of class 'prcomp': (150 x 4) x (4 x 4)'
## # 4 coordinates: PC1, PC2, ..., PC4
## #
## # Rows (principal): [ 150 x 4 | 1 ]
##     PC1    PC2     PC3 ... |   Species
##                            |   <fct>
## 1 -2.68 -0.319  0.0279     | 1 setosa
## 2 -2.71  0.177  0.210  ... | 2 setosa
## 3 -2.89  0.145 -0.0179     | 3 setosa
## 4 -2.75  0.318 -0.0316     | 4 setosa
## 5 -2.73 -0.327 -0.0901     | 5 setosa
## # … with 145 more rows
## # ℹ Use `print(n = ...)` to see more rows
## #
## # Columns (standard): [ 4 x 4 | 2 ]
##       PC1     PC2     PC3 ... |   .name        .center
##                               |   <chr>          <dbl>
## 1  0.361   -0.657   0.582     | 1 Sepal.Length    5.84
## 2 -0.0845 -0.730   -0.598 ... | 2 Sepal.Width     3.06
## 3  0.857   0.173   -0.0762    | 3 Petal.Length    3.76
## 4  0.358   0.0755 -0.546      | 4 Petal.Width     1.20
```

PCA of Anderson's iris data

```r
summary(pca)
```

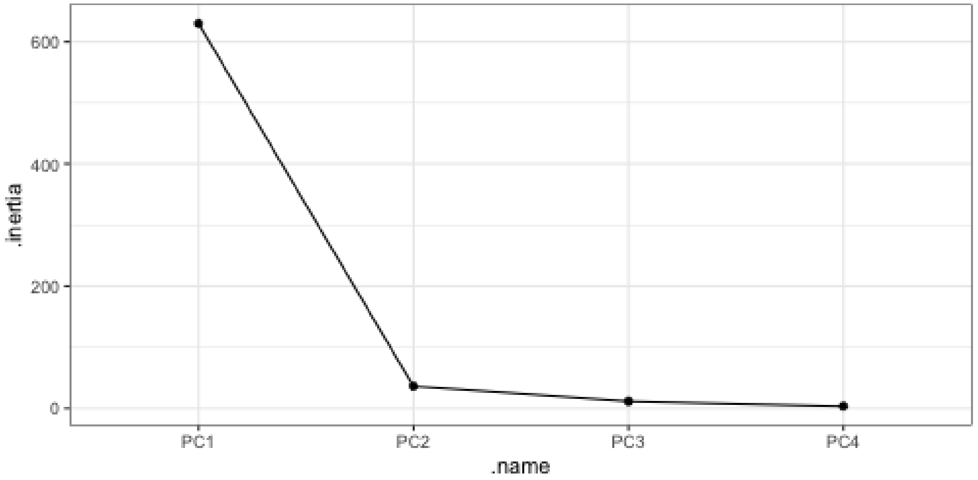
```
## Importance of components:
##                           PC1     PC2    PC3     PC4
## Standard deviation     2.0563 0.49262 0.2797 0.15439
## Proportion of Variance 0.9246 0.05307 0.0171 0.00521
## Cumulative Proportion  0.9246 0.97769 0.9948 1.00000
```

```r
screeplot(pca, type = "lines")
```



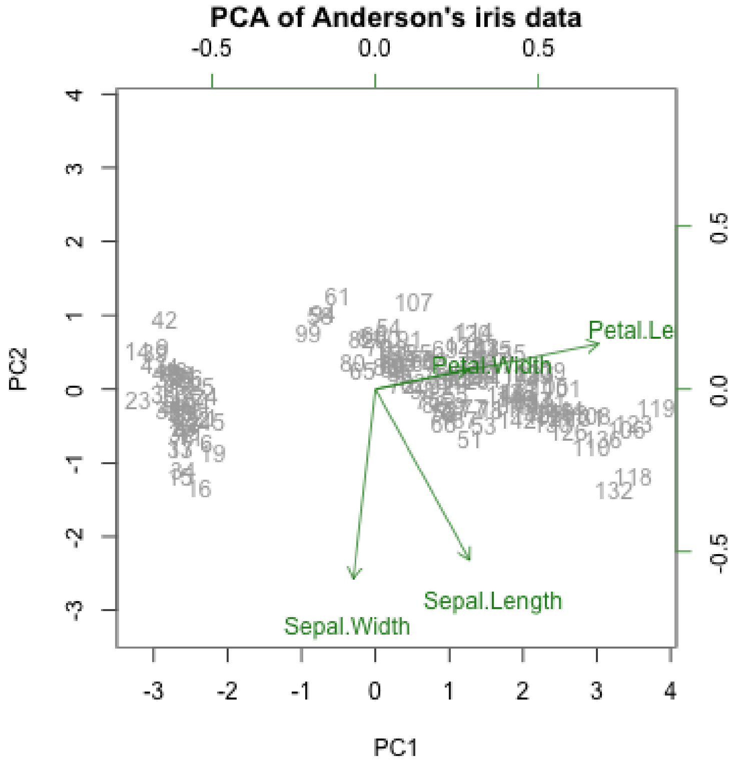
```r
tidy(pca_ord) %T>% print() %>%
  ggplot(aes(x = .name, y = .inertia, group = 1)) +
  geom_line() + geom_point()
```

```
## # A tibble: 4 × 4
##   .name .sdev .inertia .prop_var
##   <fct> <dbl>    <dbl>     <dbl>
## 1 PC1   2.06    630.     0.925
## 2 PC2   0.493    36.2    0.0531
## 3 PC3   0.280    11.7    0.0171
## 4 PC4   0.154     3.55   0.00521
```

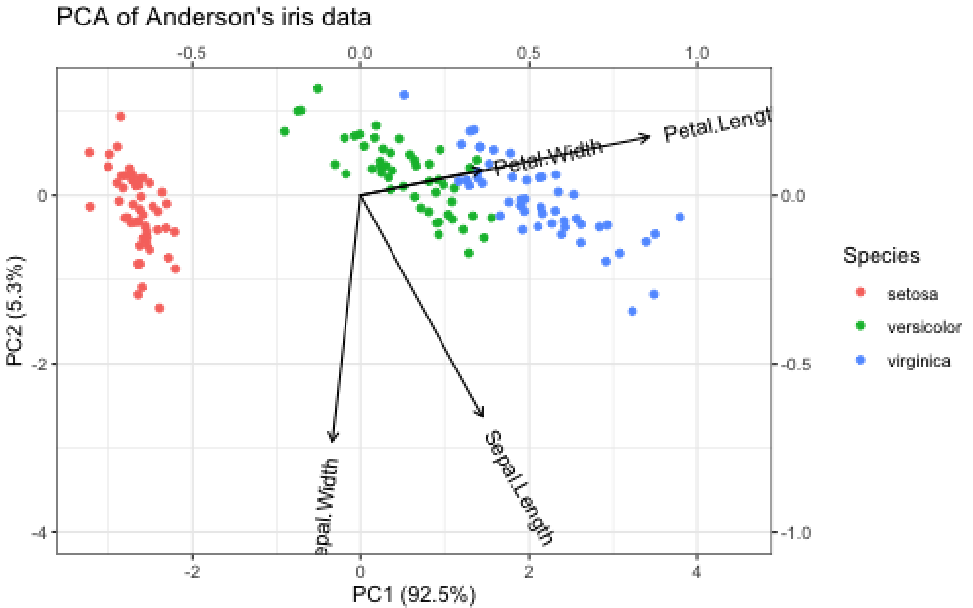


PCA of Anderson's iris data

```
biplot(pca, scale = 0, col = c("darkgrey", "forestgreen")
       main = "PCA of Anderson's iris data")
```



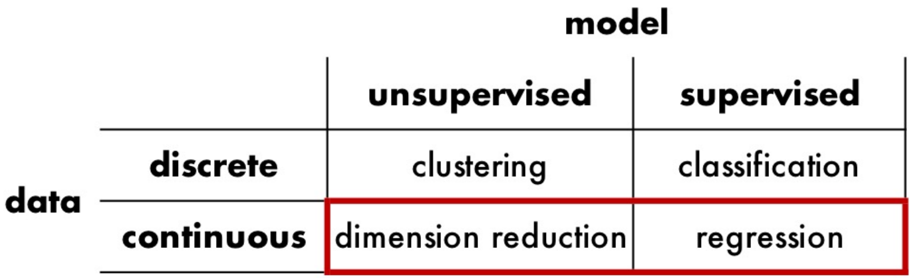
```
pca_ord %>% confer_inertia("rows") %>%
  ggbiplot(aes(label = .name),
           sec.axes = "cols", scale.factor = 4) +
  geom_rows_point(aes(color = Species)) +
  geom_cols_vector() + geom_cols_text_radiate() +
  expand_limits(x = c(NA, 4.5), y = c(-4, NA)) +
  ggtitle("PCA of Anderson's iris data")
```



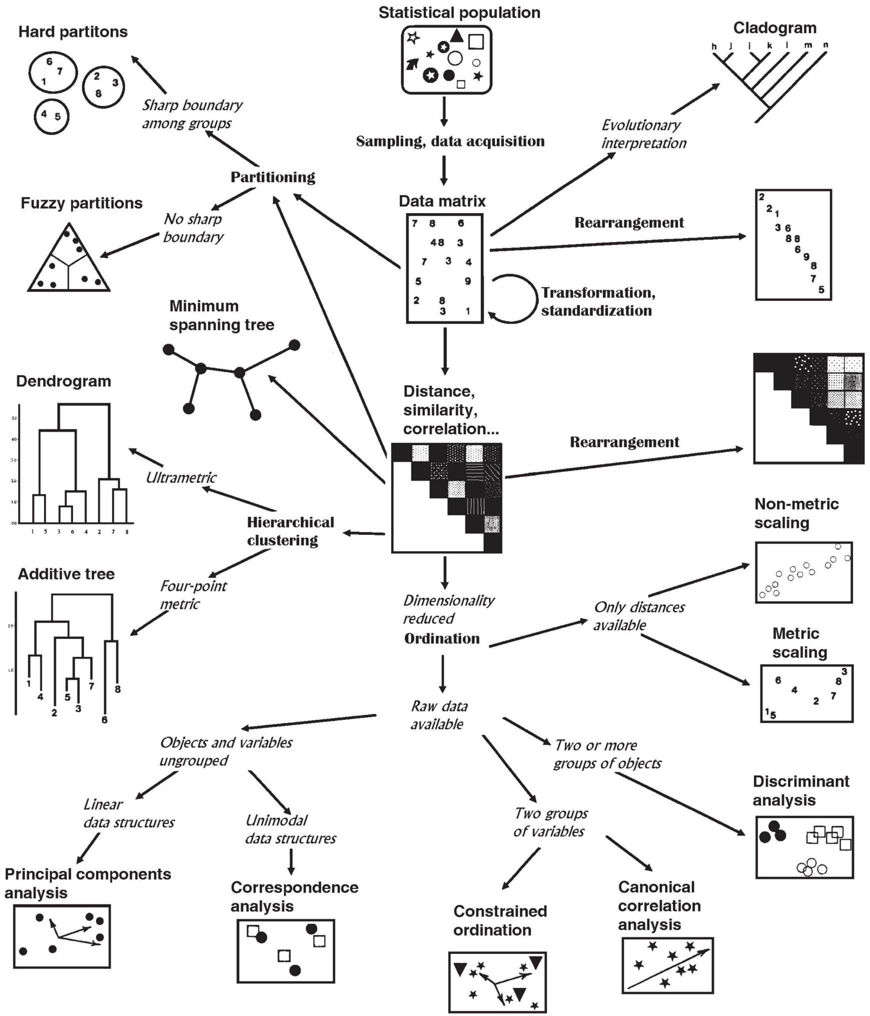
Motivation

Ordination

> "[A]ny technique that extracts artificial variables in order to reduce the dimensionality of the data is referred to as **ordination**."[1]

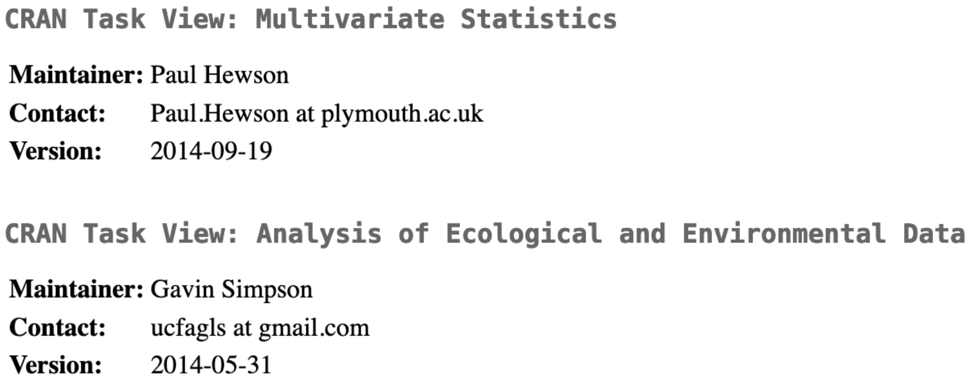


		model	
		unsupervised	**supervised**
data	**discrete**	clustering	classification
	continuous	dimension reduction	regression



Statement of need

R is replete with ordination methods!



CRAN Task View: Multivariate Statistics

Maintainer: Paul Hewson
Contact: Paul.Hewson at plymouth.ac.uk
Version: 2014-09-19

CRAN Task View: Analysis of Ecological and Environmental Data

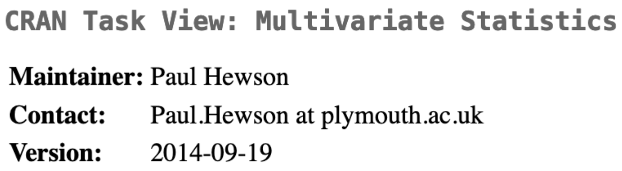
Maintainer: Gavin Simpson
Contact: ucfagls at gmail.com
Version: 2014-05-31

… but they are

- **specialized**: unweildy or unintuitive inspection
- **heterogeneous**: diverse, dissimilar conventions
- **standalone**: not designed to be interoperable

Statement of need

R is replete with ordination methods!



CRAN Task View: Multivariate Statistics

Maintainer: Paul Hewson
Contact: Paul.Hewson at plymouth.ac.uk
Version: 2014-09-19

CRAN Task View: Analysis of Ecological and Environmental Data

Maintainer: Gavin Simpson
Contact: ucfagls at gmail.com
Version: 2014-05-31

... but they are

- **specialized**: unweildy or unintuitive inspection
- **heterogeneous**: diverse, dissimilar conventions
- **standalone**: not designed to be interoperable

The tidyverse strives to be

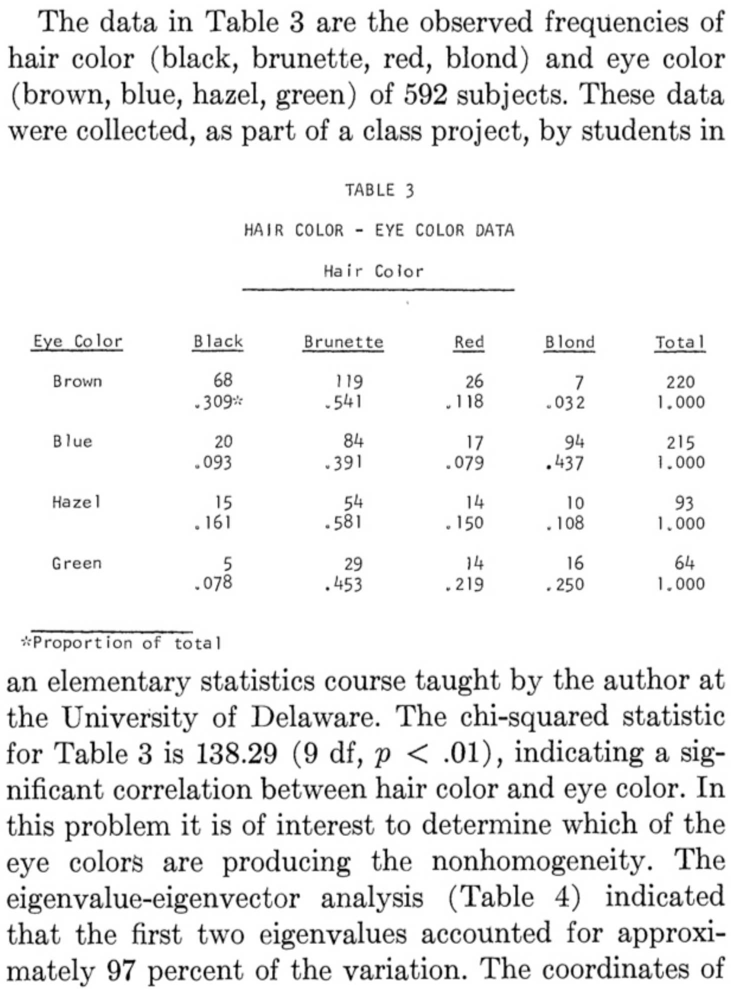
- **human-centered**: supports data analysis conducted by humans
- **consistent**: ensures that learning transfers between packages
- **composable**: enables modular data exploration and study design
- **inclusive**: is developed and informed by a broad community of users

Statement of purpose

*The {**ordr**} package leverages a **unifying framework** for matrix decomposition models and the **philosophy**, **conventions**, and **popularity** of the tidyverse in order to more **naturally** and **efficiently** integrate ordination analyses into **common** data science workflows.*

Pitch: Correspondence analysis

CA of Snee's hair and eye color data



The data in Table 3 are the observed frequencies of hair color (black, brunette, red, blond) and eye color (brown, blue, hazel, green) of 592 subjects. These data were collected, as part of a class project, by students in

TABLE 3

HAIR COLOR - EYE COLOR DATA

	Hair Color				
Eye Color	Black	Brunette	Red	Blond	Total
Brown	68 .309*	119 .541	26 .118	7 .032	220 1.000
Blue	20 .093	84 .391	17 .079	94 .437	215 1.000
Hazel	15 .161	54 .581	14 .150	10 .108	93 1.000
Green	5 .078	29 .453	14 .219	16 .250	64 1.000

---

*Proportion of total

an elementary statistics course taught by the author at the University of Delaware. The chi-squared statistic for Table 3 is 138.29 (9 df, $p < .01$), indicating a significant correlation between hair color and eye color. In this problem it is of interest to determine which of the eye colors are producing the nonhomogeneity. The eigenvalue-eigenvector analysis (Table 4) indicated that the first two eigenvalues accounted for approximately 97 percent of the variation. The coordinates of

```
(haireye <- as.data.frame(rowSums(HairEyeColor, dims = 2L)))
```

```
##       Brown Blue Hazel Green
## Black    68   20    15     5
## Brown   119   84    54    29
## Red      26   17    14    14
## Blond     7   94    10    16
```

```
summary(HairEyeColor)
```

```
## Number of cases in table: 592 
## Number of factors: 3 
## Test for independence of all factors:
##     Chisq = 164.92, df = 24, p-value = 5.321e-23
##     Chi-squared approximation may be incorrect
```

CA of Snee's hair and eye color data

```r
(ca <- MASS::corresp(haireye, nf = 2L))
```

```
## First canonical correlation(s): 0.4569165 0.1490859 
## 
##  Row scores:
##              [,1]       [,2]
## Black -1.1042772  1.4409170
## Brown -0.3244635 -0.2191109
## Red   -0.2834725 -2.1440145
## Blond  1.8282287  0.4667063
## 
##  Column scores:
##              [,1]       [,2]
## Brown -1.0771283  0.5924202
## Blue   1.1980612  0.5564193
## Hazel -0.4652862 -1.1227826
## Green  0.3540108 -2.2741218
```

```r
(ca_ord <- ordinate(haireye,
                    model = MASS::corresp, nf = 2L))
```

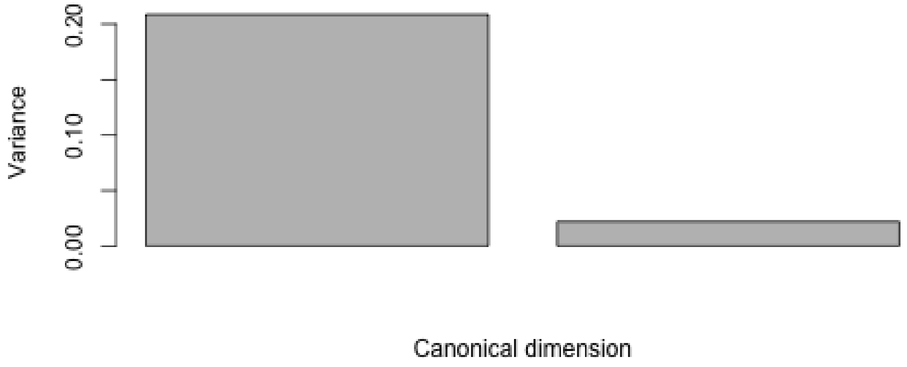
```
## # A tbl_ord of class 'correspondence': (4 x 2) x (4 x 2)'
## # 2 coordinates: Can1 and Can2
## # 
## # Rows (standard): [ 4 x 2 | 1 ]
##     Can1   Can2 |   .name
##                 |   <chr>
## 1 -1.10   1.44  | 1 Black
## 2 -0.324 -0.219 | 2 Brown
## 3 -0.283 -2.14  | 3 Red  
## 4  1.83   0.467 | 4 Blond
## # 
## # Columns (standard): [ 4 x 2 | 1 ]
##     Can1   Can2 |   .name
##                 |   <chr>
## 1 -1.08   0.592 | 1 Brown
## 2  1.20   0.556 | 2 Blue 
## 3 -0.465 -1.12  | 3 Hazel
## 4  0.354 -2.27  | 4 Green
```

CA of Snee's hair and eye color data

```
summary(ca)
```

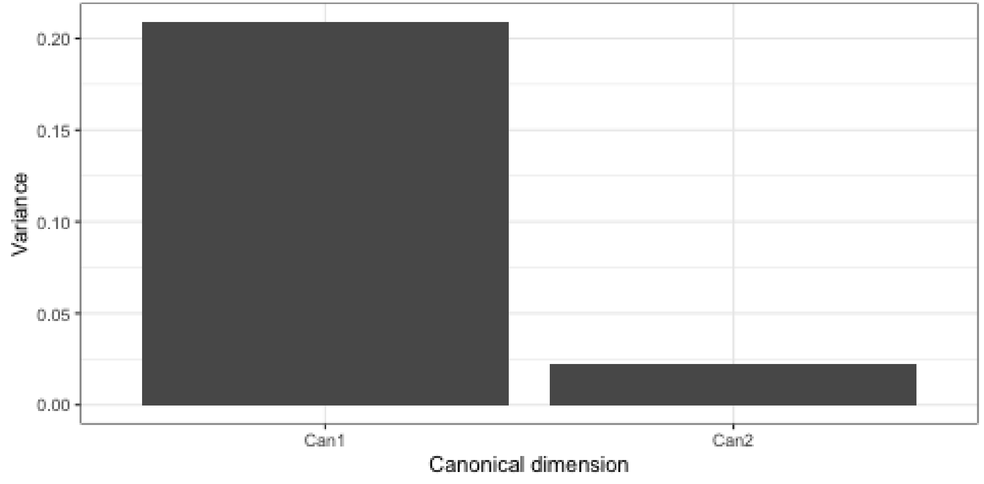
```
##         Length Class  Mode   
## cor      2     -none- numeric
## rscore   8     -none- numeric
## cscore   8     -none- numeric
## Freq    16     -none- numeric
```

```
barplot(ca$cor^2,
        xlab = "Canonical dimension", ylab = "Variance")
```



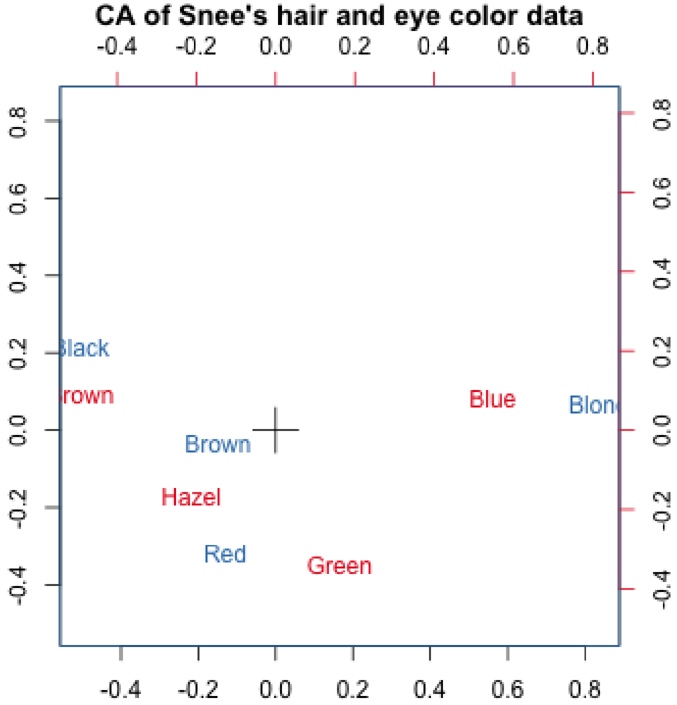
```
tidy(ca_ord) %T>% print() %>%
  ggplot(aes(x = .name, y = .cor^2)) +
  geom_col() +
  labs(x = "Canonical dimension", y = "Variance")
```

```
## # A tibble: 2 × 4
##   .name  .cor .inertia .prop_var
##   <fct> <dbl>    <dbl>     <dbl>
## 1 Can1  0.457   0.209     0.904 
## 2 Can2  0.149   0.0222    0.0962
```

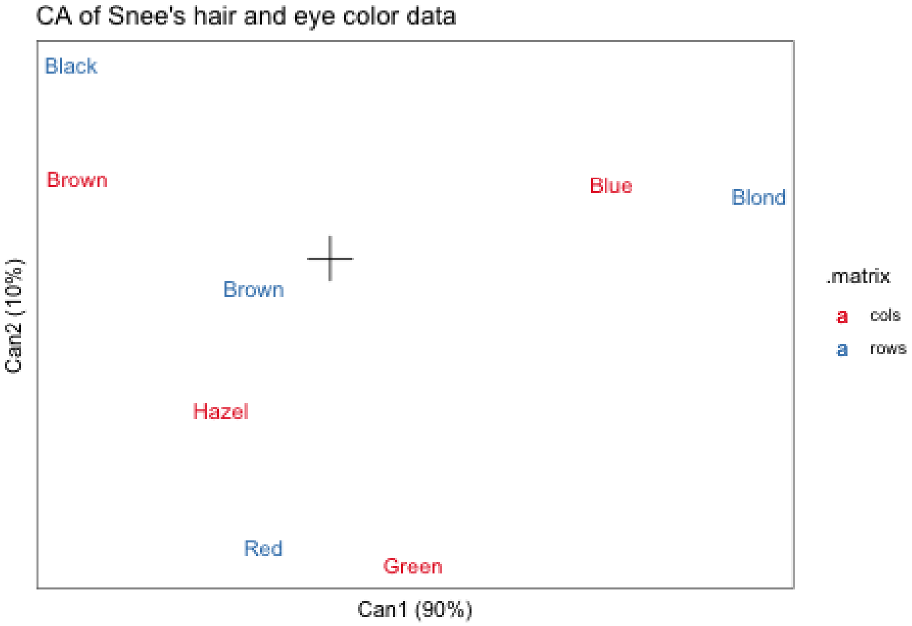


CA of Snee's hair and eye color data

```
biplot(ca, type = "symmetric",
       col = c("#377eb8", "#e41a1c"),
       main = "CA of Snee's hair and eye color data")
```



```
ca_ord %>% confer_inertia("symmetric") %>%
  ggbiplot(aes(label = .name, color = .matrix)) +
  scale_color_brewer(type = "qual", palette = "Set1") +
  theme_biplot() + geom_origin() +
  geom_rows_text() + geom_cols_text() +
  ggtitle("CA of Snee's hair and eye color data")
```



Design

Principles

{ordr} strives for a coherent grammar[2] that is

accessible

- easy to learn and straightforward to advance
- couples easily with other data science tools

opinionated

- foregrounds the unity of supported techniques
- rewards good practice, relieves choice paralysis

extensible

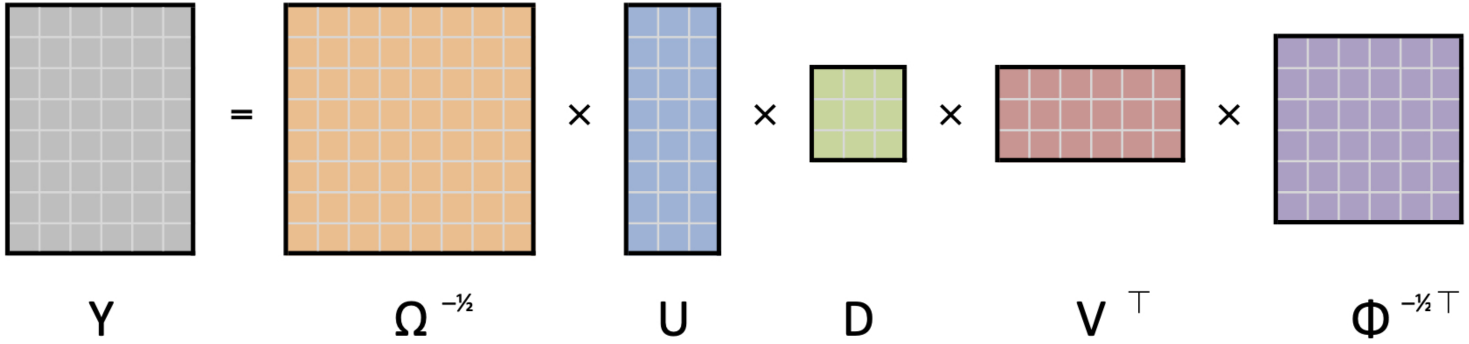
- relies on a small set of core generics
- borrows conventions from widely-used tools

Inspiration

A general approach to ordination techniques:[3]

1. **Preprocess data** (centering; double-centering): $X \rightsquigarrow Y$
2. **Generalized SVD** (weights; sphering):

$$Y = (\Omega^{-1/2}\, U)\, D\, (\Phi^{-1/2}\, V)^\top$$



3. **Biplot**, with $a + b = 1$ (row-principal; column-principal; symmetric):

$$F = \Omega^{-1/2}\, U\, D^a, \quad G = \Phi^{-1/2}\, V\, D^b$$

Core generics

Engine

For each supported model class, **recover**:

- left & right matrix factors (singular vectors)

$$U,\ \ V$$

- inertia and its distribution unto the factors

$$D = \mathrm{diag}(d_1, \ldots, d_k),\ \ (a, b)$$

- supplementary elements

$$U\ D = X\ V,\ \ X'\ V,\ \ \text{etc.}$$

- metadata about model components

Core generics

Engine

For each supported model class, **recover**:

- left & right matrix factors (singular vectors)

$$U,\ \ V$$

- inertia and its distribution unto the factors

$$D = \mathrm{diag}(d_1, \ldots, d_k),\ \ (a, b)$$

- supplementary elements

$$U\ D = X\ V,\ \ X'\ V,\ \ \text{etc.}$$

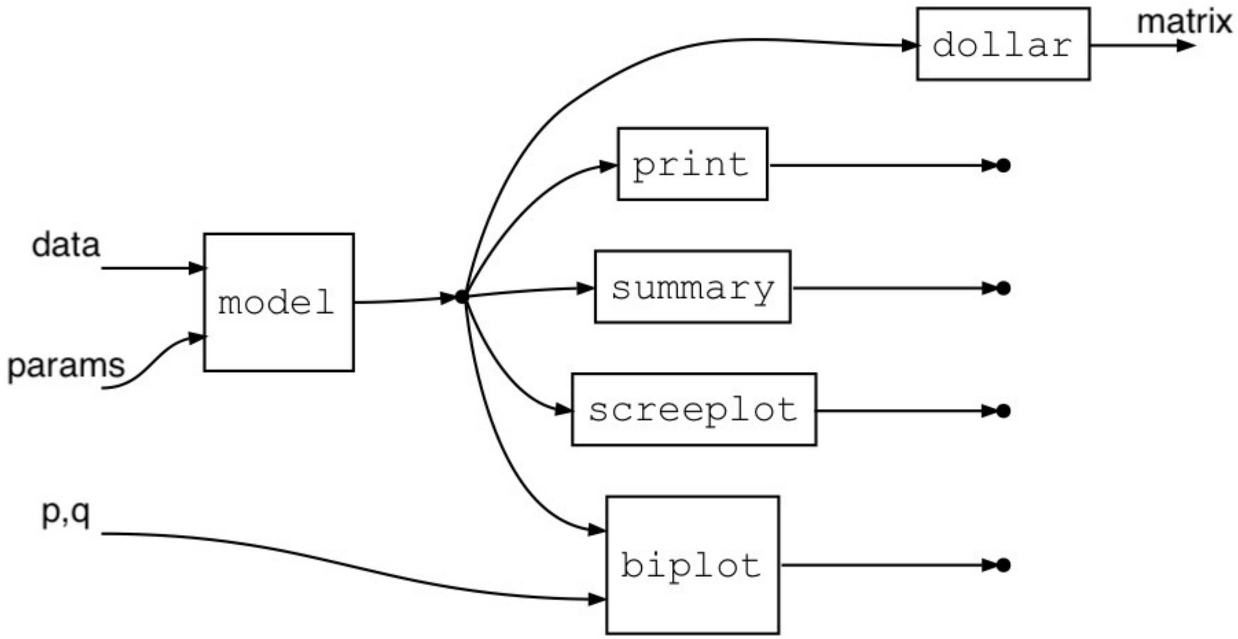
- metadata about model components

Dashboard

Via the wrapper class `'tbl_ord'`, **enable**:[4]

- clear & consistent formatting via {**tibble**}
- re-distribution of inertia
- negations of axes
- annotation of components via {**dplyr**}
- summaries for further analysis via {**broom**}
- grammatical construction of biplots via {**ggplot2**}
- workflows via {**magrittr**} (or `|>` if one must)

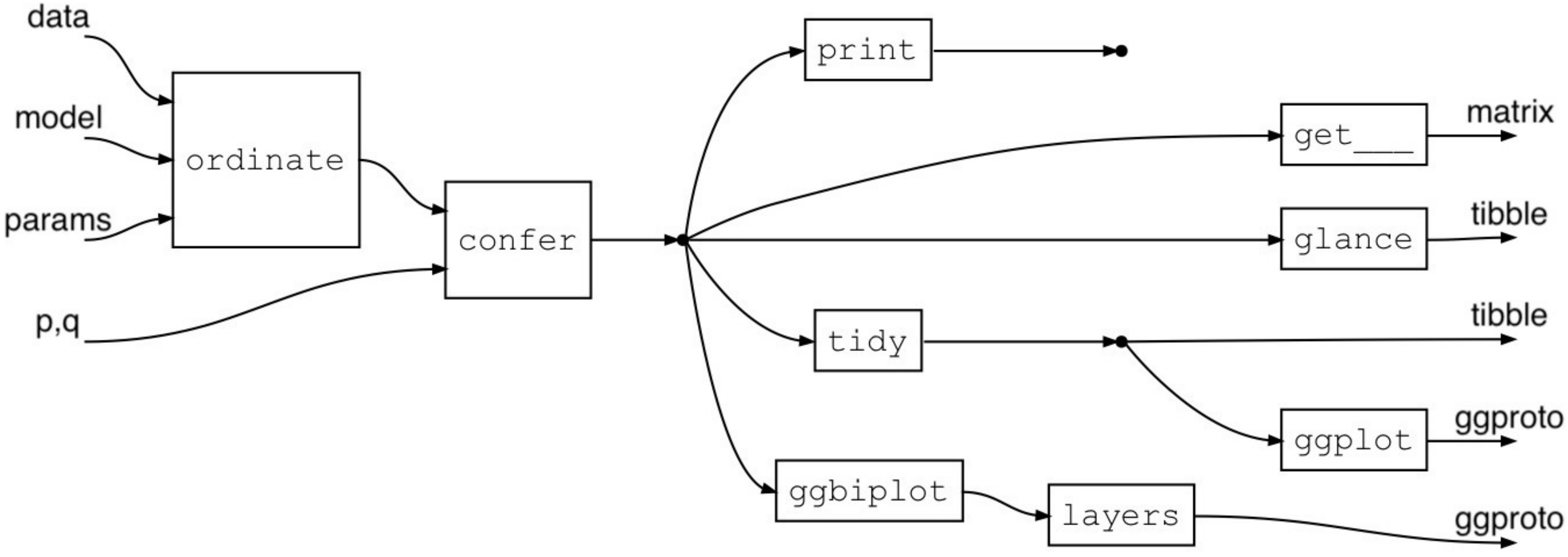
Base R workflow



Limitations

- idiosyncratic fit function and output class
- retrievable model components are nonstandard
- model summaries often have no class or methods
- (bi)plots offer limited customization and annotation

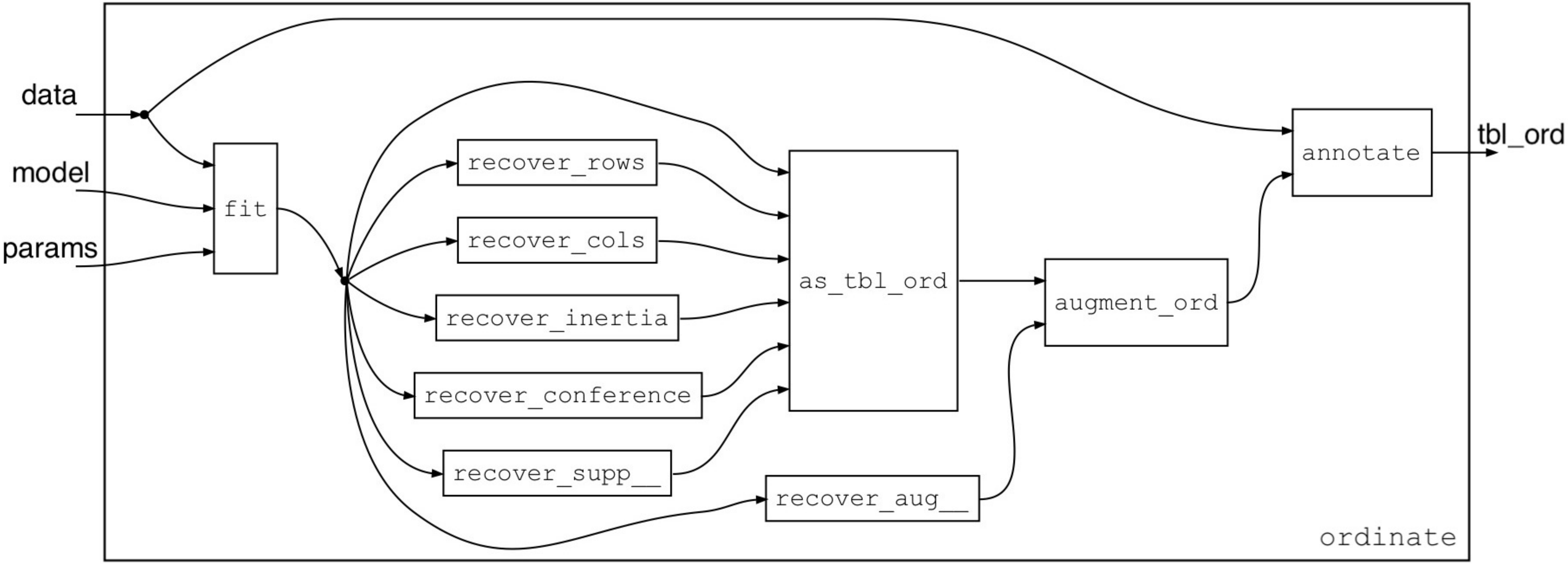
{ordr} workflow: Dashboard



Differences

- unified fit function and modifiable wrapper class
- retrieval methods for standard components
- model summaries are returned as data frames
- biplots can be layered

{ordr} workflow: Engine



Wrapper `'tbl_ord'`

- connects underlying models to unified functions by way of recovery methods

Decorator `augment_ord()`

- annotates recovered model components with recovered metadata

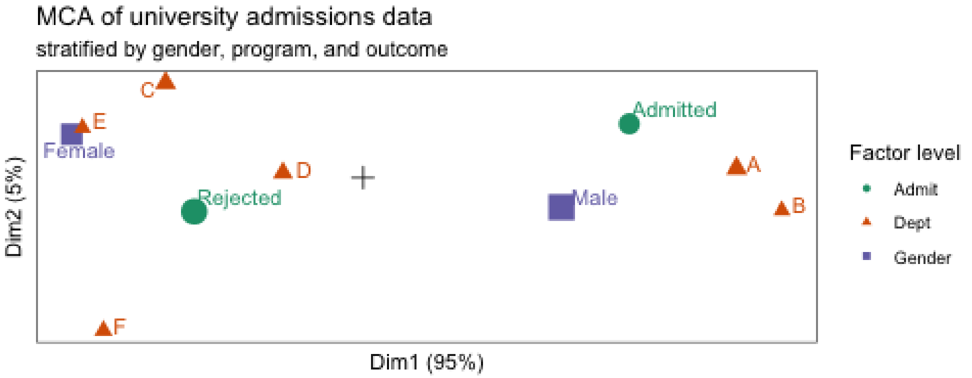
Gallery

Gallery: Multiple correspondence analysis of university admissions data

```
ca::mjca(UCBAdmissions) %>%
  as_tbl_ord() %>% augment_ord() %>%
  print() -> admissions_mca
```

```
## # A tbl_ord of class 'mjca': (4526 x 2) x (10 x 2)'
## # 2 coordinates: Dim1 and Dim2
## # 
## # Rows (standard): [ 4526 x 2 | 4 ]
##    Dim1  Dim2 |   .name    .mass   .dist
##               |   <chr>    <dbl>   <dbl>
## 1  3.33  2.68 | 1 1     0.000221 0.00672
## 2  3.33  2.68 | 2 2     0.000221 0.00672
## 3  3.33  2.68 | 3 3     0.000221 0.00672
## 4  3.33  2.68 | 4 4     0.000221 0.00672
## 5  3.33  2.68 | 5 5     0.000221 0.00672
## # … with 4,521 more rows, and
## #   1 more variable:
## #   .inertia <dbl>
## # ℹ Use `print(n = ...)` to see more rows, and `colnames()` to
## # 
## # Columns (standard): [ 10 x 2 | 6 ]
##      Dim1    Dim2 |    .name .factor .level  .mass
##                   |    <chr> <chr>   <chr>   <dbl>
##  1  1.08   0.975  |  1 Admi… Admit   Admit… 0.129
##  2 -0.681 -0.617  |  2 Admi… Admit   Rejec… 0.204
##  3 -1.18   0.786  |  3 Gend… Gender  Female 0.135
##  4  0.802 -0.536  |  4 Gend… Gender  Male   0.198
```

```
admissions_mca %>%
  confer_inertia("colprincipal") %>%
  ggbiplot() + theme_biplot() + geom_origin() +
  geom_cols_point(aes(color = .factor, shape = .factor,
                      size = .mass)) +
  geom_cols_text_repel(aes(label = .level, color = .facto
                       show.legend = FALSE) +
  scale_color_brewer(palette = "Dark2") +
  scale_size_area(guide = "none") +
  labs(color = "Factor level", shape = "Factor level") +
  ggtitle("MCA of university admissions data",
          "stratified by gender, program, and outcome")
```

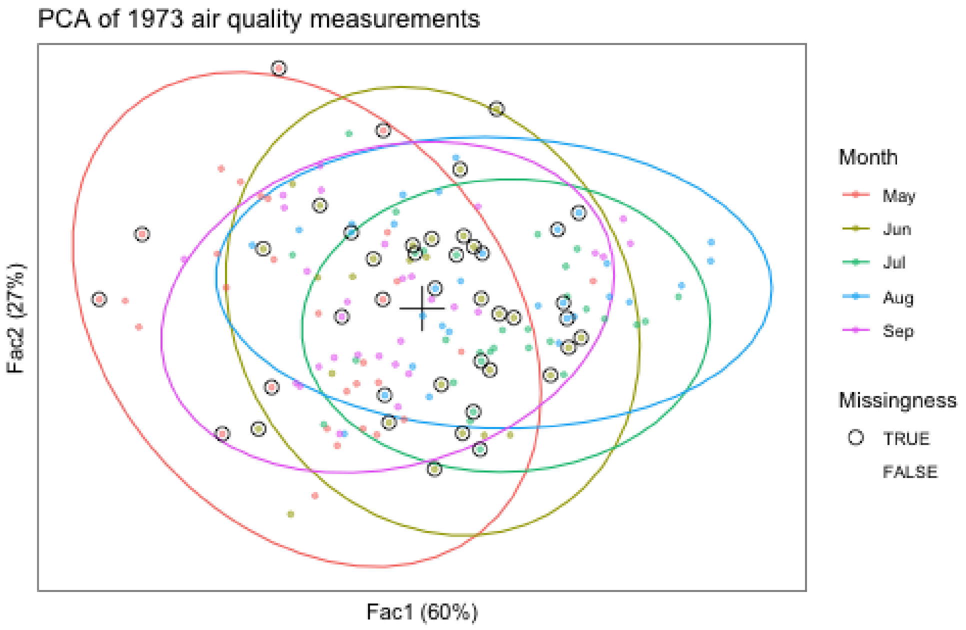


Gallery: NIPALS of (incomplete) air quality data

```
ordinate(airquality, seq(4L), ade4::nipals, nf = 3L) %>%
  mutate_rows(
    Month = factor(month.abb[Month], levels = month.abb),
    Missingness = apply(is.na(airquality[, 1:4]), 1L, any
  ) %>%
  print() -> airqual_nipals
```

```
## # A tbl_ord of class 'nipals': (153 x 3) x (4 x 3)'
## # 3 coordinates: Fac1, Fac2, Fac3
## # 
## # Rows (principal): [ 153 x 3 | 4 ]
##     Fac1    Fac2   Fac3 |   .name Month   Day Missingn…¹
##                         |   <chr> <fct> <int> <lgl>
## 1 -0.305  0.334  -1.25  | 1 1     May       1 FALSE
## 2 -0.426  0.930  -0.501 | 2 2     May       2 FALSE
## 3 -1.27  -0.0592  0.279 | 3 3     May       3 FALSE
## 4 -1.16  -1.46   -1.44  | 4 4     May       4 FALSE
## 5 -3.41   0.906  -0.280 | 5 5     May       5 TRUE
## # … with 148 more rows, and
## #   abbreviated variable name
## #   ¹Missingness
## # ℹ Use `print(n = ...)` to see more rows
## # 
## # Columns (standard): [ 4 x 3 | 1 ]
##     Fac1    Fac2   Fac3 |   .name
##                         |   <chr>
## 1  0.582  0.0175  0.104 | 1 Ozone
```

```
airqual_nipals %>%
  ggbiplot() + theme_biplot() + geom_origin() +
  stat_rows_ellipse(aes(color = Month)) +
  geom_rows_point(aes(color = Month), size = 1, alpha = .
  geom_rows_point(aes(shape = Missingness), size = 3) +
  scale_shape_manual(values = c(`TRUE` = 1L, `FALSE` = NA
  ggtitle("PCA of 1973 air quality measurements")
```

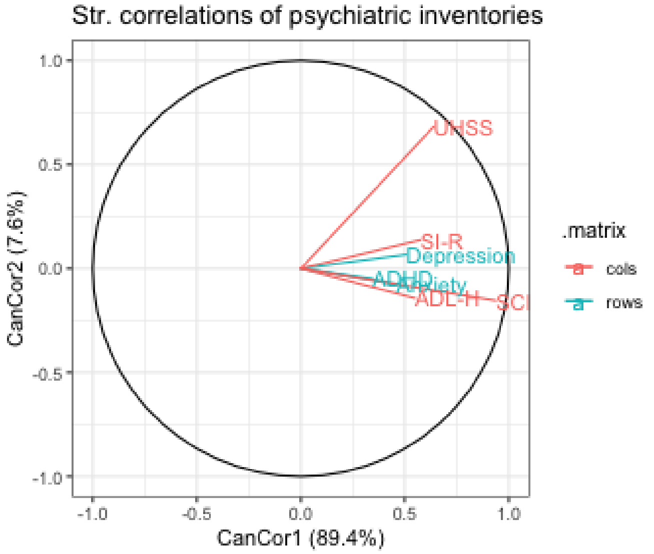


Gallery: Canonical correlation analysis of psychiatric inventories

```
cca_ord <- cancor_ord(pred, resp, scores = TRUE)
(cca_ord <- augment_ord(as_tbl_ord(cca_ord)))
```

```
## # A tbl_ord of class 'cancor_ord': (322 x 3) x (324 x 3)'
## # 3 coordinates: CanCor1, CanCor2, CanCor3
## # 
## # Rows (standard): [ 322 x 3 | 3 ]
##    CanCor1    CanCor2    CanCor3 |    .name      .center .element
##                                  |   <chr>        <dbl> <chr>   
## 1  0.00146 -0.00494  -0.00306  | 1 Anxiety       18.9 active  
## 2  0.00264  0.00507  -0.000169 | 2 Depression    20.7 active  
## 3  0.00143 -0.000979  0.00371   | 3 ADHD          38.0 active  
## 4 -0.0307   0.00214   0.0151    | 4 <NA>          NA   score   
## 5  0.0263   0.148    -0.0476    | 5 <NA>          NA   score   
## # … with 317 more rows
## # ℹ Use `print(n = ...)` to see more rows
## # 
## # Columns (standard): [ 324 x 3 | 3 ]
##     CanCor1    CanCor2  CanCor3 |    .name .center .element
##                                  |   <chr>   <dbl> <chr>   
## 1 -0.000433 -0.00240  -0.00560 | 1 SI-R     65.4  active  
## 2  0.0208   -0.0303   -0.0176  | 2 ADL-H     2.13 active  
## 3  0.00154  -0.000812  0.00120 | 3 SCI     100.   active  
## 4  0.00247   0.0136    0.00318 | 4 UHSS     30.0  active  
## 5 -0.0510   -0.169    -0.0889  | 5 1        NA    score   
## # … with 319 more rows
## # ℹ Use `print(n = ...)` to see more rows
```

```
cca_ord %>% confer_inertia("rows") %>%
  ggbiplot(aes(label = .name, color = .matrix)) +
  geom_unit_circle() +
  geom_rows_vector(elements = "structure", arrow = NULL)
  geom_cols_vector(elements = "structure", arrow = NULL)
  geom_rows_text(elements = "structure", hjust = "outward
  geom_cols_text(elements = "structure", hjust = "outward
  expand_limits(x = c(-1, 1), y = c(-1, 1)) +
  ggtitle("Str. correlations of psychiatric inventories")
```

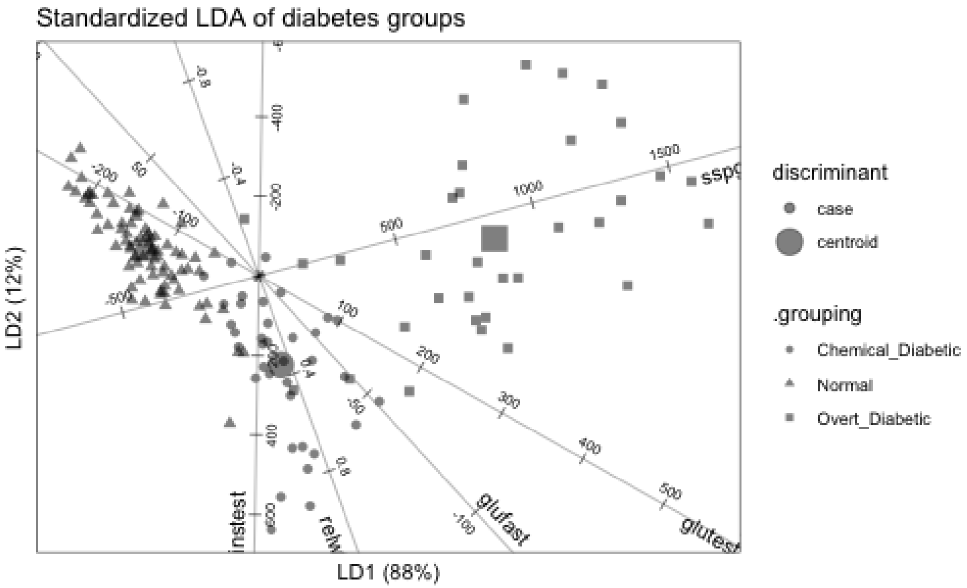


Gallery: Linear discriminant analysis of diabetes data

```
MASS::lda(group ~ ., heplots::Diabetes) %>%
  as_tbl_ord() %>% augment_ord() %>%
  mutate_rows(discriminant = ifelse(
    .element == "active", "centroid", "case"
  )) %>%
  print() -> diabetes_lda
```

```
## # A tbl_ord of class 'lda': (148 x 2) x (5 x 2)'
## # 2 coordinates: LD1 and LD2
## # 
## # Rows (principal): [ 148 x 2 | 6 ]
##      LD1    LD2 |   .name .prior .counts .grou…¹
##                 |   <chr>  <dbl>   <int> <chr>
## 1 -1.75   0.400 | 1 Norm…  0.524      76 Normal
## 2  0.340 -1.38  | 2 Chem…  0.248      36 Chemic…
## 3  3.66   0.580 | 3 Over…  0.228      33 Overt_…
## 4 -1.72   0.663 | 4 1     NA          NA Normal
## 5 -2.85   1.30  | 5 2     NA          NA Normal
## # … with 143 more rows, 2
## #   more variables:
## #   .element <chr>,
## #   discriminant <chr>, and
## #   abbreviated variable name
## #   ¹.grouping
## # ℹ Use `print(n = ...)` to see more rows, and `colnames()` to
## # 
## # Columns (standard): [ 5 x 2 | 2 ]
```

```
diabetes_lda %>% confer_inertia(1) %>%
  ggbiplot() + theme_biplot() +
  geom_rows_point(aes(shape = .grouping, size = discrimin
                  alpha = .5) +
  geom_cols_axis(aes(label = .name),
                 color = "#888888", num = 8L,
                 text_size = 2.5, label_dodge = .02) +
  ggtitle("Standardized LDA of diabetes groups")
```



Wrap

Needs

Engagement

- bug & need reports
- use cases
- code contributions

Documentation

- how to use {ordr} with **new model classes**
- how to use **new plot layers** with `ggbiplot()`

Features

- quality measures
- predictive biplots[5]

Welcome

Install **ordr**, and optionally methods for additional S3 classes:

```
remotes::install_github("corybrunson/ordr")
remotes::install_github("corybrunson/ordr.extra")
```

Thank you!

[1] Podani J • 2000 • *Introduction to the Exploration of Multivariate Biological Data* • http://ramet.elte.hu/~podani/books.html

[2] Wilkinson L • 2005 • *The Grammar of Graphics* • https://link.springer.com/book/10.1007/0-387-28695-0

[3] Greenacre M • 1984 • *Theory and applications of correspondence analysis* • http://www.carme-n.org/

[4] Pedersen TL • 2022 • {tidygraph}: A Tidy API for Graph Manipulation • https://CRAN.R-project.org/package=tidygraph

[5] Gower JC, Lubbe SG, le Roux NJ • 2011 • *Understanding Biplots* • https://www.wiley.com/go/biplots