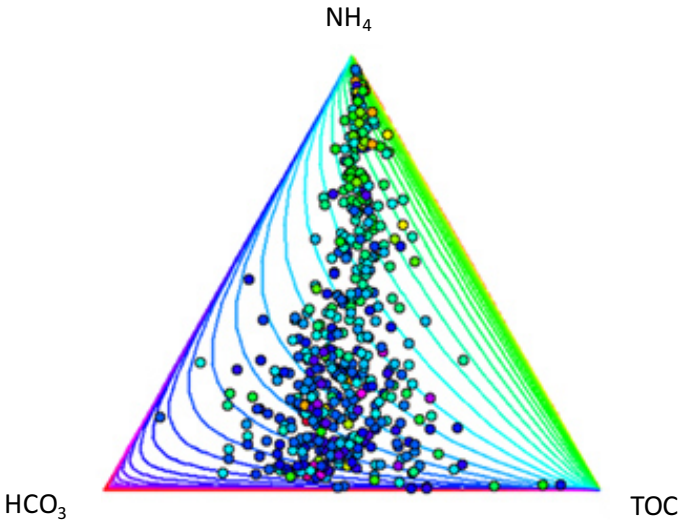
Compositional Data Analysis with R



Javier Palarea-Albaladejo, javier.palarea@bioss.ac.uk
Senior Statistical Scientist at Biomathematics & Statistics Scotland (BioSS)
(http://www.bioss.ac.uk)

Some CoDa background

- Compositional data (CoDa) are multivariate data defined on a *simplex*:

$$\mathcal{S}^D = \{\mathbf{x} = [x_1, x_2, \ldots, x_D] : x_1 > 0, x_2 > 0, \ldots, x_D > 0; \sum_{i=1}^{D} x_i = \kappa\}$$

- Represent parts of a whole, relative weights, commonly expressed in %, ppm, mg/kg, wt.%, hours/day, …

- E.g: chemical/nutritional compositions, activity patterns, relative abundances, metabolomic profiles, mRNA data, multiparty electoral data, investment portfolios, …

- They have a *relative and symmetric scale*

Daily minutes spent in sleep, sedentary behaviour, light and vigorous physical activity within 24 hours (1440 min.)

Sleep	Sedentary	Light	Vigorous	SUM
540	468	417	16	1441
420	684	327	9	1440
480	690	223	47	1440
360	661	397	22	1440
420	435	545	41	1441
240	342	835	23	1440

→

In Percentage:

Sleep	Sedentary	Light	Vigorous	SUM
37.47	32.48	28.94	1.11	100
29.17	47.50	22.71	0.63	100
33.33	47.92	15.49	3.26	100
25.00	45.90	27.57	1.53	100
29.15	30.19	37.82	2.85	100
16.67	23.75	57.99	1.60	100

Some CoDa background

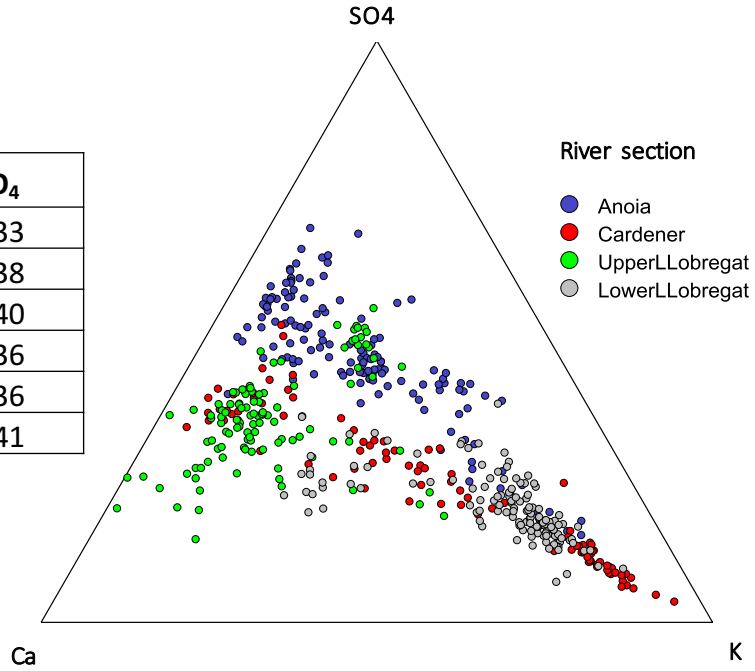
- Basic principles:
  - *Scale invariance*
  - *Subcompositional coherence*
  - *Permutation invariance*

Hydrochemical subcomposition in mass proportions to molar proportions using their molar weight

K	Ca	SO_4
0.24	0.25	0.51
0.20	0.23	0.57
0.20	0.22	0.58
0.21	0.24	0.54
0.20	0.27	0.53
0.18	0.23	0.59

→

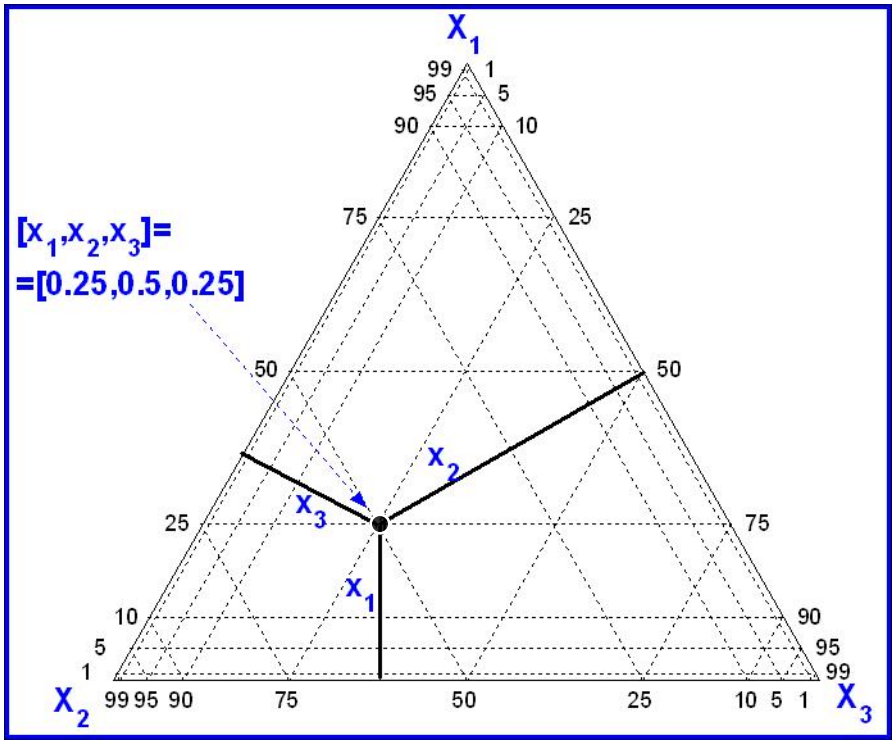
K	Ca	SO_4
0.42	0.25	0.33
0.37	0.24	0.38
0.36	0.24	0.40
0.39	0.25	0.36
0.36	0.28	0.36
0.34	0.25	0.41



Key reference: Aitchison, J. (1986) The statistical analysis of compositional data, Chapman & Hall. (Reprinted in 2003 by The Blackburn Press)

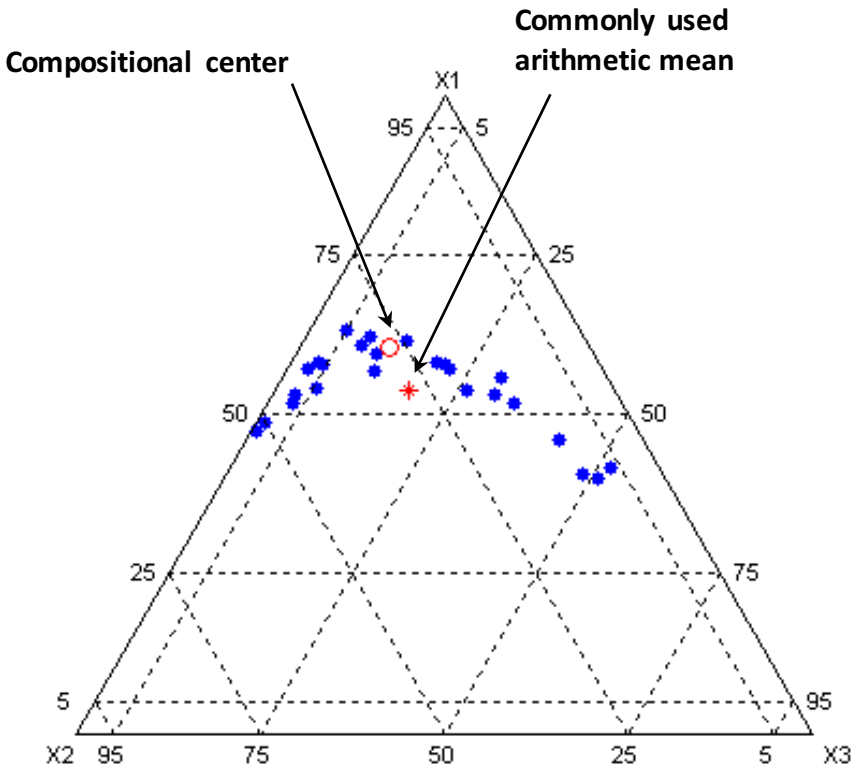
Compositions and standard statistics

The "scatterplot" for compositions:
Ternary plots



Standard measures of position, variability, similarity/distance and so on are not coherent with compositions

Center of the data distribution:



Spurious covariance and correlation

$$Cov(x_i, x_1) + \ldots + Cov(x_i, x_D) = 0, \quad i = 1, 2, \ldots, D$$

Subcompositional incoherence of correlations

Correlation matrices from 25 samples of a full 5-part composition [A, B, C, D, E] and two subcompositions [C, D, E] and [A, B, E]

[A, B, C, D, E]

	A	**B**	C	D	**E**
A	1	0.51	-0.82	-0.05	-0.17
B	0.51	1	-0.90	-0.50	-0.57
C	-0.82	-0.90	1	0.33	0.42
D	-0.05	-0.50	0.33	1	-0.21
E	-0.17	-0.57	0.42	-0.21	1

[A, B, E]

	A	**B**	E
A	1	-0.94	0.61
B	-0.94	1	-0.85
E	0.61	-0.85	1

[C, D, E]

	C	D	**E**
C	1	-0.87	-0.74
D	-0.87	1	0.32
E	-0.74	0.32	1

The `compositions` R package

`compositions: Compositional Data Analysis`

The package provides functions for the consistent analysis of compositional data (e.g. portions of substances) and positive numbers (e.g. concentrations) in the way proposed by Aitchison and Pawlowsky-Glahn.

Version:	1.40-1
Depends:	R (≥ 2.2.0), tensorA, robustbase, energy, bayesm
Suggests:	rgl
Published:	2014-06-07
Author:	K. Gerald van den Boogaart, Raimon Tolosana, Matevz Bren
Maintainer:	K. Gerald van den Boogaart <support at boogaart.de>

Table 1
Summary table of definition of the geometries available

Significant size?	Yes	Yes	No	No
Natural scale by	Difference	Quotient	Difference	Quotient
Class	`rplus`	`aplus`	`rcomp`	`acomp`
Inner sum	Sum $z_i = x_i + y_i$	Perturbation $z_i = x_i \cdot y_i$	Sum $z_i = x_i + y_i$ (*)	Closed perturbation $z_i = \mathrm{clo}(x_i \cdot y_i)$
Product by a scalar	Product $z_i = \lambda \cdot x_i$	Power $z_i = x_i^\lambda$	Product $z_i = \lambda \cdot x_i$ (*)	Closed power $z_i = \mathrm{clo}(x_i^\lambda)$
Scalar product	$\sum_{i=1}^{D} x_i \cdot y_i$	$\sum_{i=1}^{D} \ln x_i \cdot \ln y_i$	$\sum_{i=1}^{D} x_i \cdot y_i - \frac{1}{D}$	$\sum_{i=1}^{D} \ln \frac{x_i}{g(\mathbf{x})} \cdot \ln \frac{y_i}{g(\mathbf{y})}$
Neutral element	$\mathbf{0}$ (origin)	$\mathbf{1}$	–	$\mathrm{clo}(\mathbf{1}) = \frac{1}{D}\mathbf{1}$ (barycenter)
Form of linear combinations	Weighted sums	(Unbalanced) Mass action equations	End-member mixtures	Balanced mass action equations
Related references	–	Pawlowsky-Glahn (2003)	Rehder and Zier (2002)	Aitchison (1986, 2002)

Notation: $\mathbf{x} = (x_1, x_2, \ldots, x_D)$ is a vector of observations, $g(\mathbf{x}) = \sqrt[D]{x_1 \cdot x_2 \cdots x_D}$ is the geometric mean of its parts, $\mathbf{0}$ and $\mathbf{1}$ are, respectively, vectors of D zeroes and D ones, and $\mathrm{clo}(\cdot)$ is the closure operation (Eq. (1)). (*) These operations return an `rmult` object.

Source: van den Boogaart & Tolosana-Delgado (2008). ''compositions'': a unified R package to analyze compositional data, Computers & Geosciences 343, 320-338.
See also "Analyzing compositional data with R" by the same authors, Springer, 2013.

The compositions R package

compositions: Compositional Data Analysis

The package provides functions for the consistent analysis of compositional data (e.g. portions of substances) and positive numbers (e.g. concentrations) in the way proposed by Aitchison and Pawlowsky-Glahn.

Version:	1.40-1
Depends:	R (≥ 2.2.0), tensorA, robustbase, energy, bayesm
Suggests:	rgl
Published:	2014-06-07
Author:	K. Gerald van den Boogaart, Raimon Tolosana, Matevz Bren
Maintainer:	K. Gerald van den Boogaart <support at boogaart.de>

Table 1
Summary table of definition of the geometries available

Significant size?	Yes	Yes	No	No
Natural scale by	Difference	Quotient	Difference	Quotient
Class	`rplus`	`aplus`	`rcomp`	`acomp`
Inner sum	Sum $z_i = x_i + y_i$	Perturbation $z_i = x_i \cdot y_i$	Sum $z_i = x_i + y_i$ (*)	Closed perturbation $z_i = \mathrm{clo}(x_i \cdot y_i)$
Product by a scalar	Product $z_i = \lambda \cdot x_i$	Power $z_i = x_i^\lambda$	Product $z_i = \lambda \cdot x_i$ (*)	Closed power $z_i = \mathrm{clo}(x_i^\lambda)$
Scalar product	$\sum_{i=1}^D x_i \cdot y_i$	$\sum_{i=1}^D \ln x_i \cdot \ln y_i$	$\sum_{i=1}^D x_i \cdot y_i - \frac{1}{D}$	$\sum_{i=1}^D \ln \frac{x_i}{g(\mathbf{x})} \cdot \ln \frac{y_i}{g(\mathbf{y})}$
Neutral element	**0** (origin)	**1**	–	$\mathrm{clo}(\mathbf{1}) = \frac{1}{D}\mathbf{1}$ (barycenter)
Form of linear combinations	Weighted sums	(Unbalanced) Mass action equations	End-member mixtures	Balanced mass action equations
Related references	–	Pawlowsky-Glahn (2003)	Rehder and Zier (2002)	Aitchison (1986, 2002)

Notation: $\mathbf{x} = (x_1, x_2, \ldots, x_D)$ is a vector of observations, $g(\mathbf{x}) = \sqrt[D]{x_1 \cdot x_2 \cdots x_D}$ is the geometric mean of its parts, **0** and **1** are, respectively, vectors of D zeroes and D ones, and clo(·) is the closure operation (Eq. (1)). (*) These operations return an `rmult` object.

Source: van den Boogaart & Tolosana-Delgado (2008). ''compositions'': a unified R package to analyze compositional data, Computers & Geosciences 343, 320-338. See also "Analyzing compositional data with R" by the same authors, Springer, 2013.

Basic operations with the acomp class

- Closure (adding up to a constant)

$$\mathbf{x} = \mathcal{C}[x_1, \ldots, x_D] = \left[\frac{x_1 \cdot \kappa}{\sum_{i=1}^{D} x_i}, \ldots, \frac{x_D \cdot \kappa}{\sum_{i=1}^{D} x_i}\right]$$

- Perturbation (translation, change)

$$\mathbf{x} \oplus \mathbf{y} = \mathcal{C}[x_1 y_1, \ldots, x_D y_D]$$
$$\mathbf{x} \ominus \mathbf{y} = \mathcal{C}[x_1 / y_1, \ldots, x_D / y_D]$$

- Powering (like product by scalar)

$$a \odot \mathbf{x} = \mathcal{C}[x_1^a, \ldots, x_D^a]$$

```
# Time amongst 3 activities x = [A, B, C] = [20, 60, 20]%

x <- acomp(c(20,60,20)) # last year
y <- acomp(c(40,30,30)) # current year

clo(x) # close to 1

## [1] 0.2 0.6 0.2

y-x # difference between years (perturbation)

## [1] 0.500 0.125 0.375
## attr(,"class")
## [1] acomp

# Change in A is 4 times change in B
# Change in B is 1/3 change in C

# Bacterial growth: 3-species composition
x <- acomp(c(50,30,20)) # relative abundances today
# They are multiplied by 2, 3 and 5 respectively per day
# Distribution tomorrow?
g <- acomp(c(2,3,5))
x+g

## [1] 0.3448 0.3103 0.3448
## attr(,"class")
## [1] acomp

# In 5 days?
x+(5*g)

## [1] 0.02241 0.10212 0.87547
```

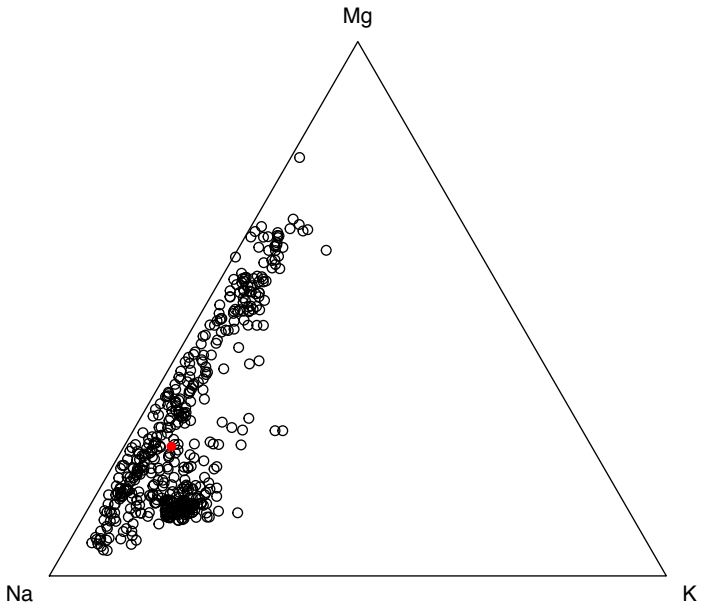
Compositional summary statistics

- Centre (compositional mean)

$$\mathbf{g} = \mathcal{C}[g_1, \ldots, g_D], \quad \text{with} \quad g_j = (\prod_{i=1}^{n} x_{ij})^{1/n}, \; j = 1, \ldots, D$$

- Variation matrix (compositional dispersion)

$$\mathbf{T} = [\tau_{ij}] = \begin{bmatrix} \operatorname{var}\left(\log \frac{X_1}{X_1}\right) & \operatorname{var}\left(\log \frac{X_1}{X_2}\right) & \cdots & \operatorname{var}\left(\log \frac{X_1}{X_D}\right) \\ \operatorname{var}\left(\log \frac{X_2}{X_1}\right) & \operatorname{var}\left(\log \frac{X_2}{X_2}\right) & \cdots & \operatorname{var}\left(\log \frac{X_2}{X_D}\right) \\ \vdots & \vdots & \ddots & \vdots \\ \operatorname{var}\left(\log \frac{X_D}{X_1}\right) & \operatorname{var}\left(\log \frac{X_D}{X_2}\right) & \cdots & \operatorname{var}\left(\log \frac{X_D}{X_D}\right) \end{bmatrix}$$




```
# Hydrochemical 14-part composition

data(Hydrochem)
# select 4-part subcomposition
sub <- acomp(Hydrochem[,c(7:10)])
mean(sub) # compositional centre

##      Na       K      Mg      Ca
## 0.35850 0.04035 0.12749 0.47367
## attr(,"class")
## [1] acomp

variation(sub) # variation matrix

##         Na      K     Mg     Ca
## Na 0.0000 0.4539 0.7443 1.1118
## K  0.4539 0.0000 1.1885 1.6973
## Mg 0.7443 1.1885 0.0000 0.2196
## Ca 1.1118 1.6973 0.2196 0.0000

# Ternary plot of first 3 parts

sub2 <- acomp(sub[,1:3])
plot(sub2)
plot(mean(sub2),col="red",pch=16,add=T)
# Centered data set
sub2.centred <- sub2 - mean(sub2)
plot(sub2.centred)
plot(mean(sub2.centred),col="red",pch=16,add=T)
mean(sub2.centred)

##     Na      K     Mg
## 0.3333 0.3333 0.3333
```

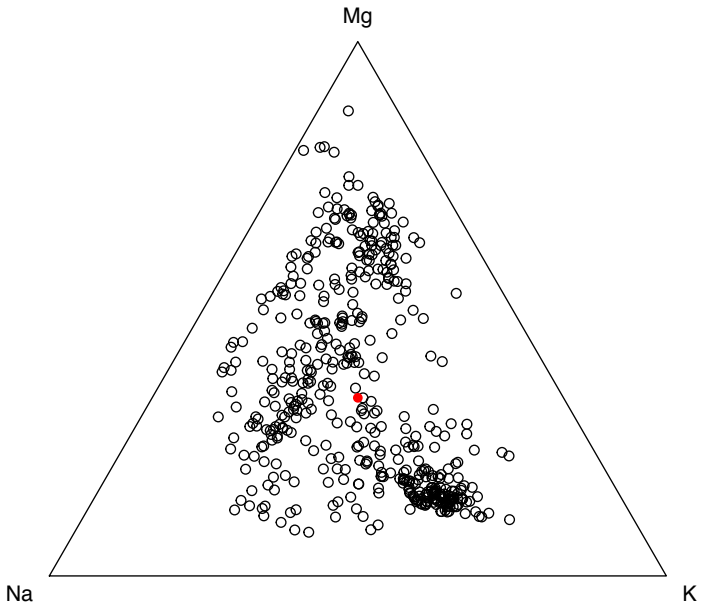
Compositional summary statistics

- Centre (compositional mean)

$$\mathbf{g} = \mathcal{C}[g_1, \ldots, g_D], \quad \text{with} \quad g_j = (\prod_{i=1}^{n} x_{ij})^{1/n}, \; j = 1, \ldots, D$$

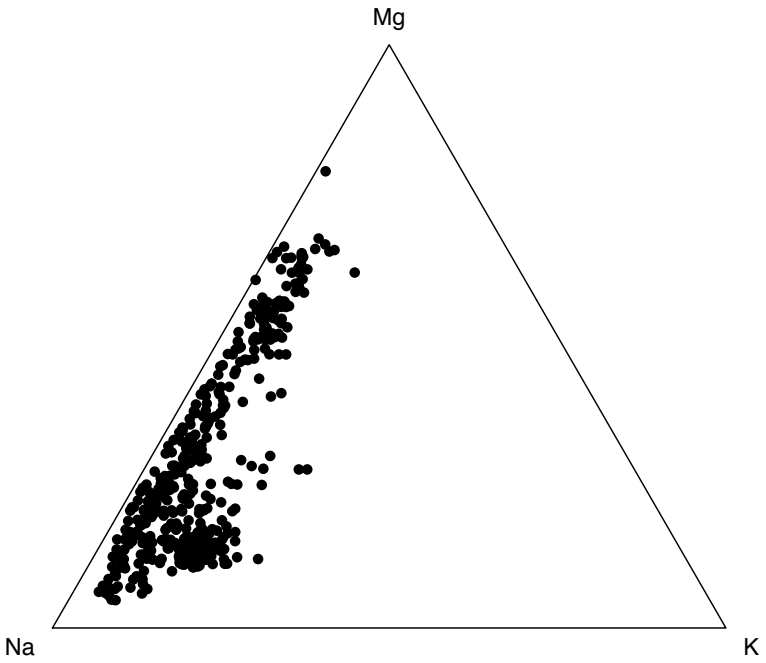
- Variation matrix (compositional dispersion)

$$\mathbf{T} = [\tau_{ij}] = \begin{bmatrix} \operatorname{var}\left(\log \frac{X_1}{X_1}\right) & \operatorname{var}\left(\log \frac{X_1}{X_2}\right) & \cdots & \operatorname{var}\left(\log \frac{X_1}{X_D}\right) \\ \operatorname{var}\left(\log \frac{X_2}{X_1}\right) & \operatorname{var}\left(\log \frac{X_2}{X_2}\right) & \cdots & \operatorname{var}\left(\log \frac{X_2}{X_D}\right) \\ \vdots & \vdots & \ddots & \vdots \\ \operatorname{var}\left(\log \frac{X_D}{X_1}\right) & \operatorname{var}\left(\log \frac{X_D}{X_2}\right) & \cdots & \operatorname{var}\left(\log \frac{X_D}{X_D}\right) \end{bmatrix}$$



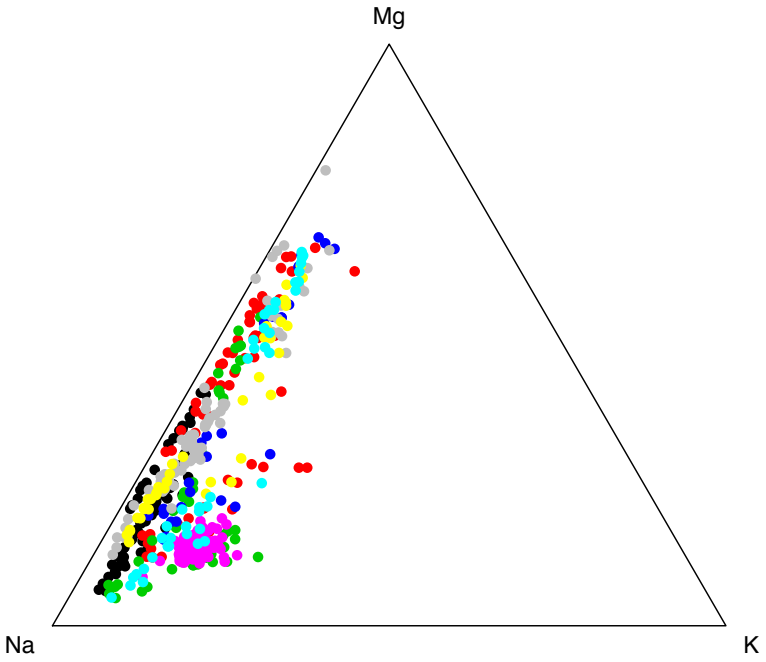

```
# Hydrochemical 14-part composition

data(Hydrochem)
# select 4-part subcomposition
sub <- acomp(Hydrochem[,c(7:10)])
mean(sub) # compositional centre

##      Na       K      Mg      Ca
## 0.35850 0.04035 0.12749 0.47367
## attr(,"class")
## [1] acomp

variation(sub) # variation matrix

##         Na      K     Mg     Ca
## Na 0.0000 0.4539 0.7443 1.1118
## K  0.4539 0.0000 1.1885 1.6973
## Mg 0.7443 1.1885 0.0000 0.2196
## Ca 1.1118 1.6973 0.2196 0.0000

# Ternary plot of first 3 parts

sub2 <- acomp(sub[,1:3])
plot(sub2)
plot(mean(sub2),col="red",pch=16,add=T)
# Centered data set
sub2.centred <- sub2 - mean(sub2)
plot(sub2.centred)
plot(mean(sub2.centred),col="red",pch=16,add=T)
mean(sub2.centred)

##     Na      K     Mg
## 0.3333 0.3333 0.3333
```

Some plotting options



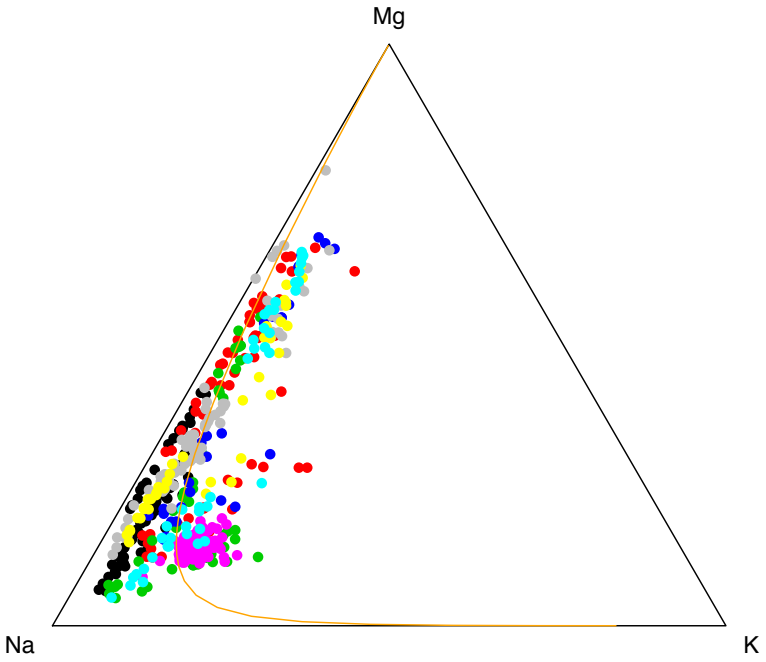

```
# Ternary plot of first 3 parts

colours <- as.numeric(Hydrochem$Location)
sub2 <- acomp(sub[,1:3])
plot(sub2,pch=16)
plot(sub2,col=colours,pch=16) # Colour by location
# Add first principal component line
plot(sub2,col=colours,pch=16,pca=TRUE,col.pca="orange")
isoPortionLines(lty=2) # Add reference grid
```

Some plotting options



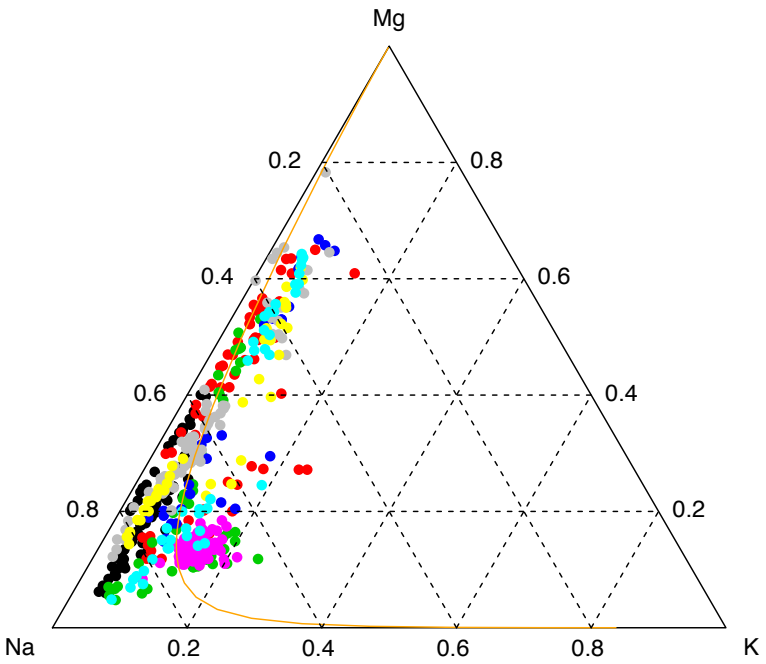

```
# Ternary plot of first 3 parts

colours <- as.numeric(Hydrochem$Location)
sub2 <- acomp(sub[,1:3])
plot(sub2,pch=16)
plot(sub2,col=colours,pch=16) # Colour by location
# Add first principal component line
plot(sub2,col=colours,pch=16,pca=TRUE,col.pca="orange")
isoPortionLines(lty=2) # Add reference grid
```

Some plotting options



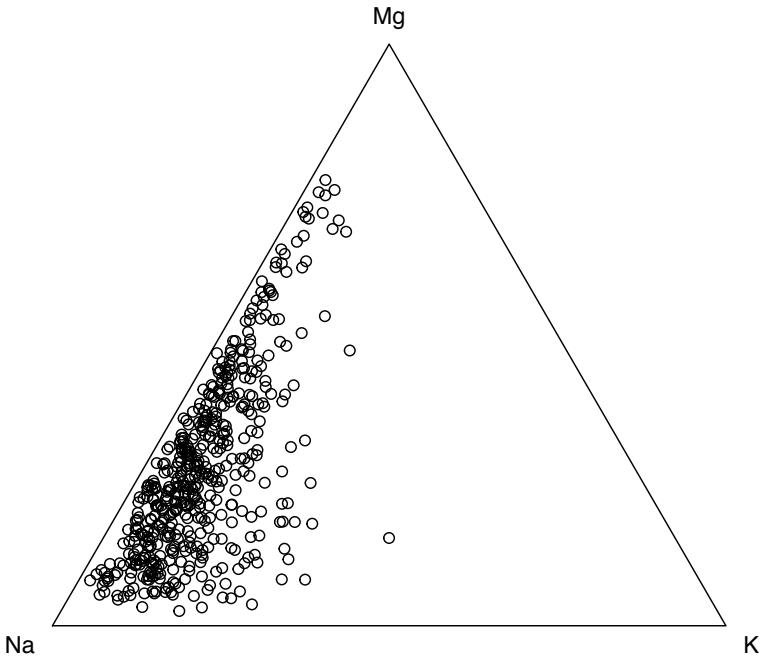

```
# Ternary plot of first 3 parts

colours <- as.numeric(Hydrochem$Location)
sub2 <- acomp(sub[,1:3])
plot(sub2,pch=16)
plot(sub2,col=colours,pch=16) # Colour by location
# Add first principal component line
plot(sub2,col=colours,pch=16,pca=TRUE,col.pca="orange")
isoPortionLines(lty=2) # Add reference grid
```

Some plotting options



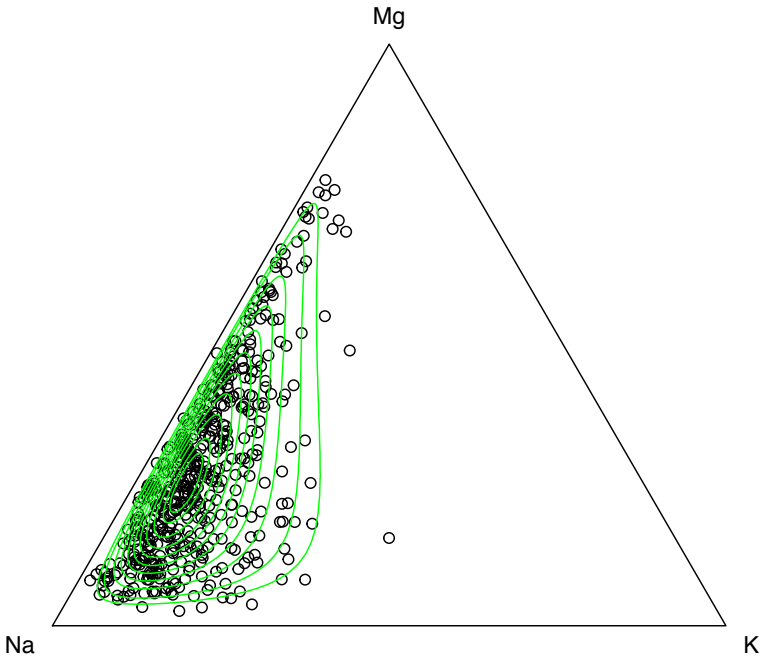

```
# Ternary plot of first 3 parts

colours <- as.numeric(Hydrochem$Location)
sub2 <- acomp(sub[,1:3])
plot(sub2,pch=16)
plot(sub2,col=colours,pch=16) # Colour by location
# Add first principal component line
plot(sub2,col=colours,pch=16,pca=TRUE,col.pca="orange")
isoPortionLines(lty=2) # Add reference grid
```

Some plotting options



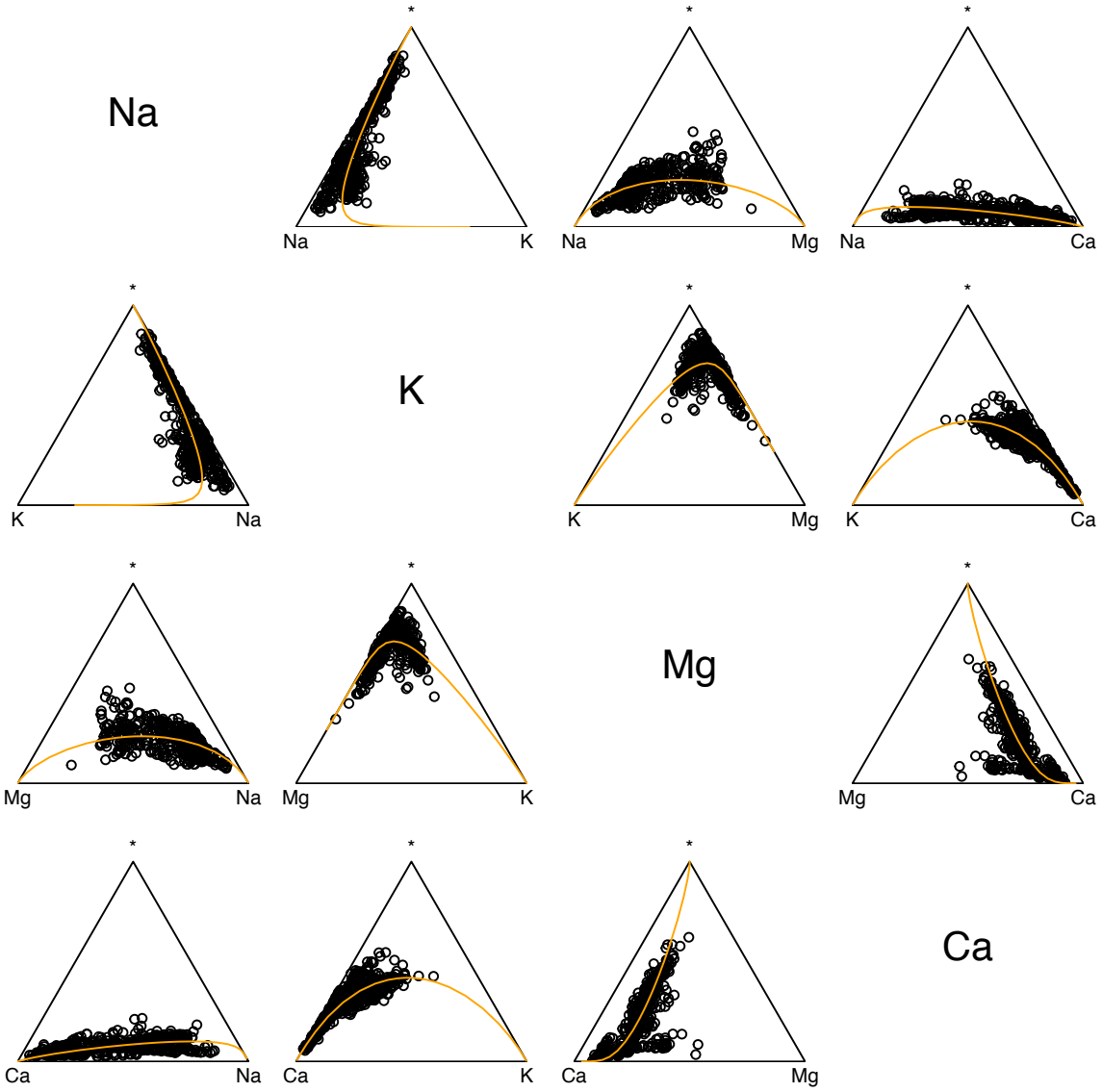

```r
# Generate 500 random values from
# normal distribution on the simplex
xr = rnorm.acomp(n=500,mean(sub2),var(sub2))
plot(xr)
# Add isodensity ellipses
for (p in c(0.5,1:9,9.5)/10){
  r = sqrt(qchisq(p=p,df=2))
  ellipses(mean(sub2),var(sub2),r,col="green")
}
```

Some plotting options




```
# Generate 500 random values from
# normal distribution on the simplex
xr = rnorm.acomp(n=500,mean(sub2),var(sub2))
plot(xr)
# Add isodensity ellipses
for (p in c(0.5,1:9,9.5)/10){
  r = sqrt(qchisq(p=p,df=2))
  ellipses(mean(sub2),var(sub2),r,col="green")
}
```

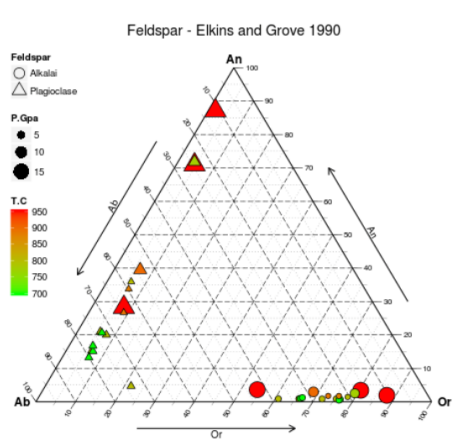
Some plotting options

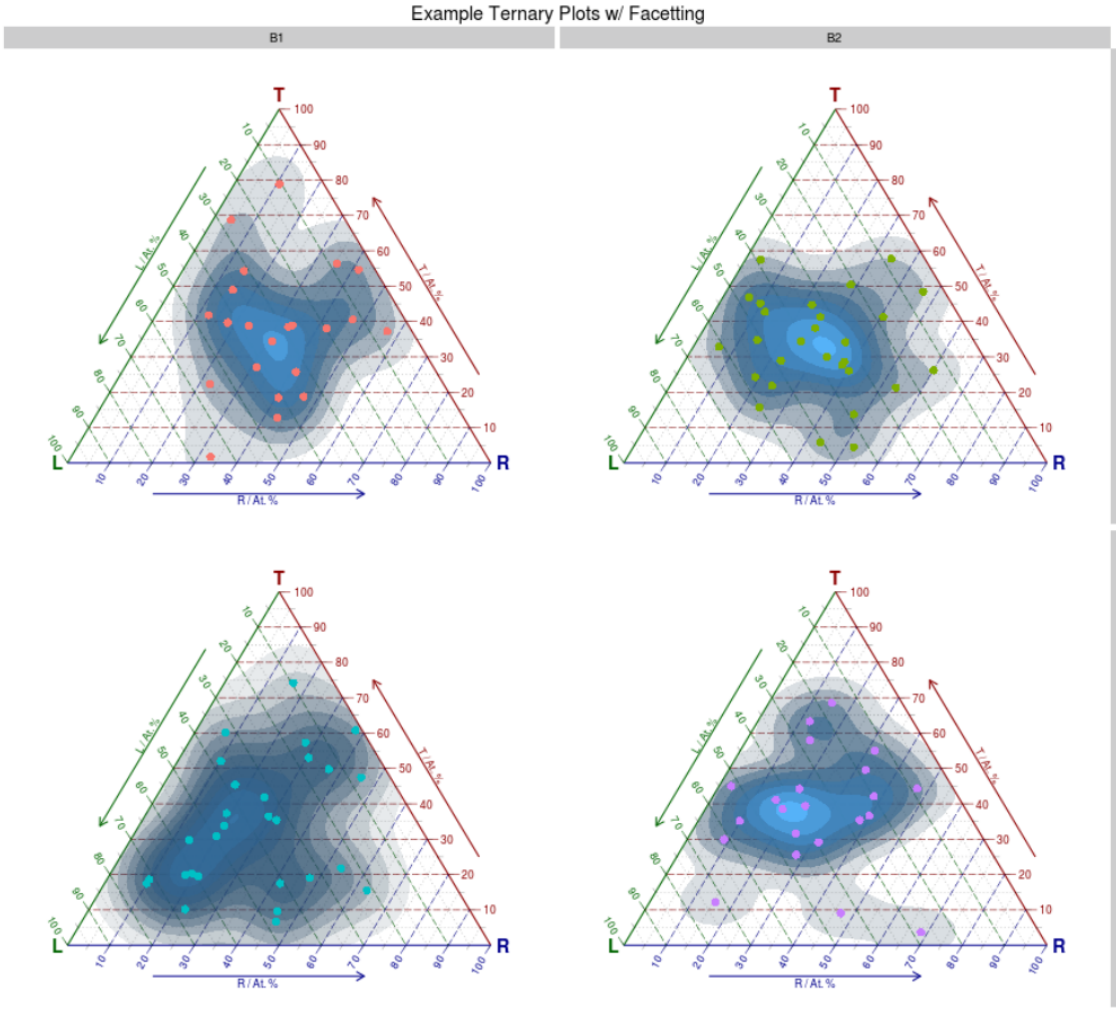


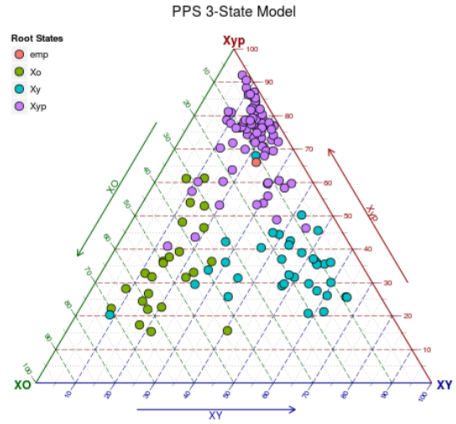

```r
# Ternary plot matrix

plot(sub,pca=TRUE,col.pca="orange")
```

And for ggplot2 lovers …

ggtern package (http://www.ggtern.com): extension to ggplot2 for the plotting of ternary diagrams by Nicholas Hamilton

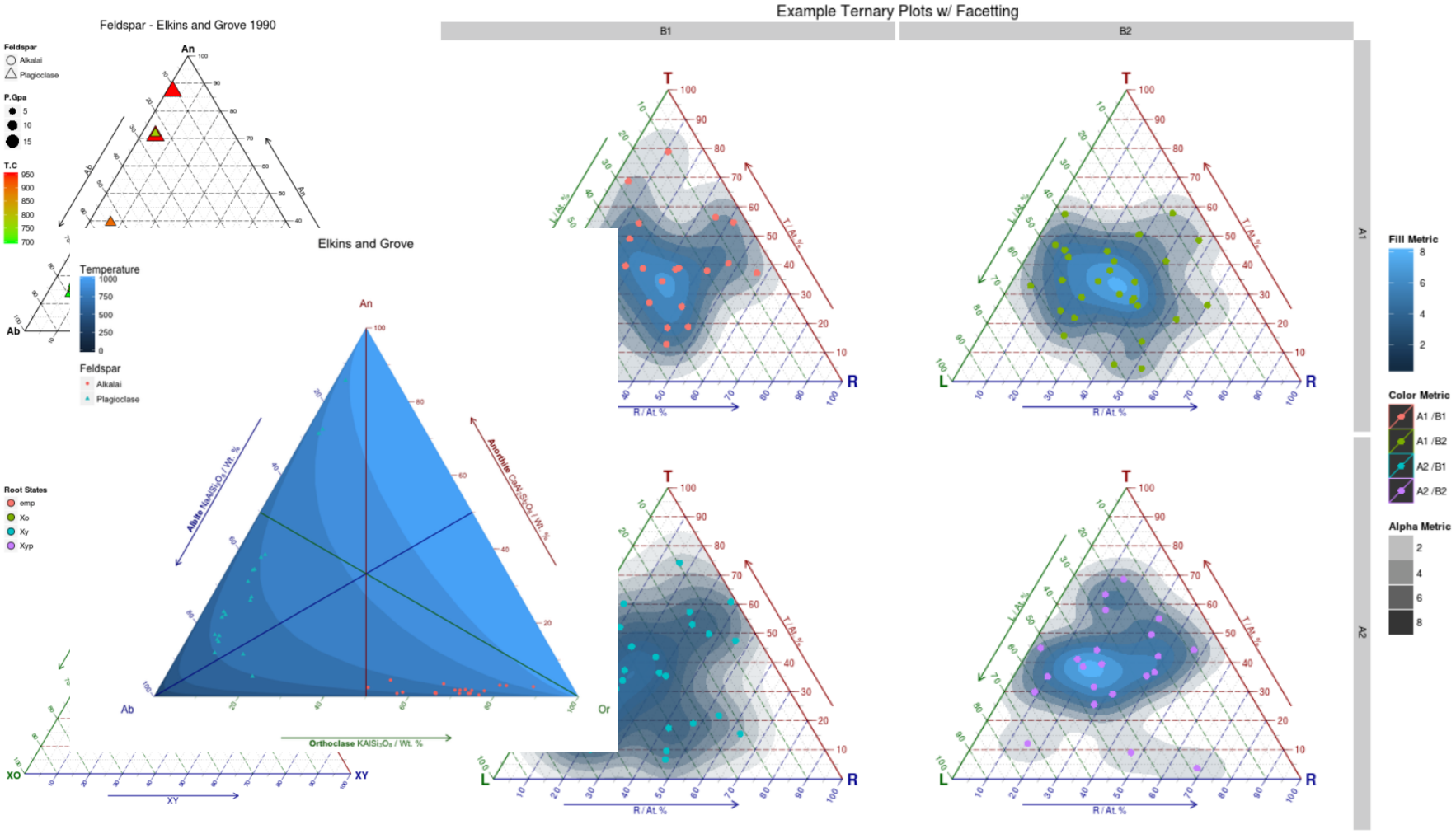






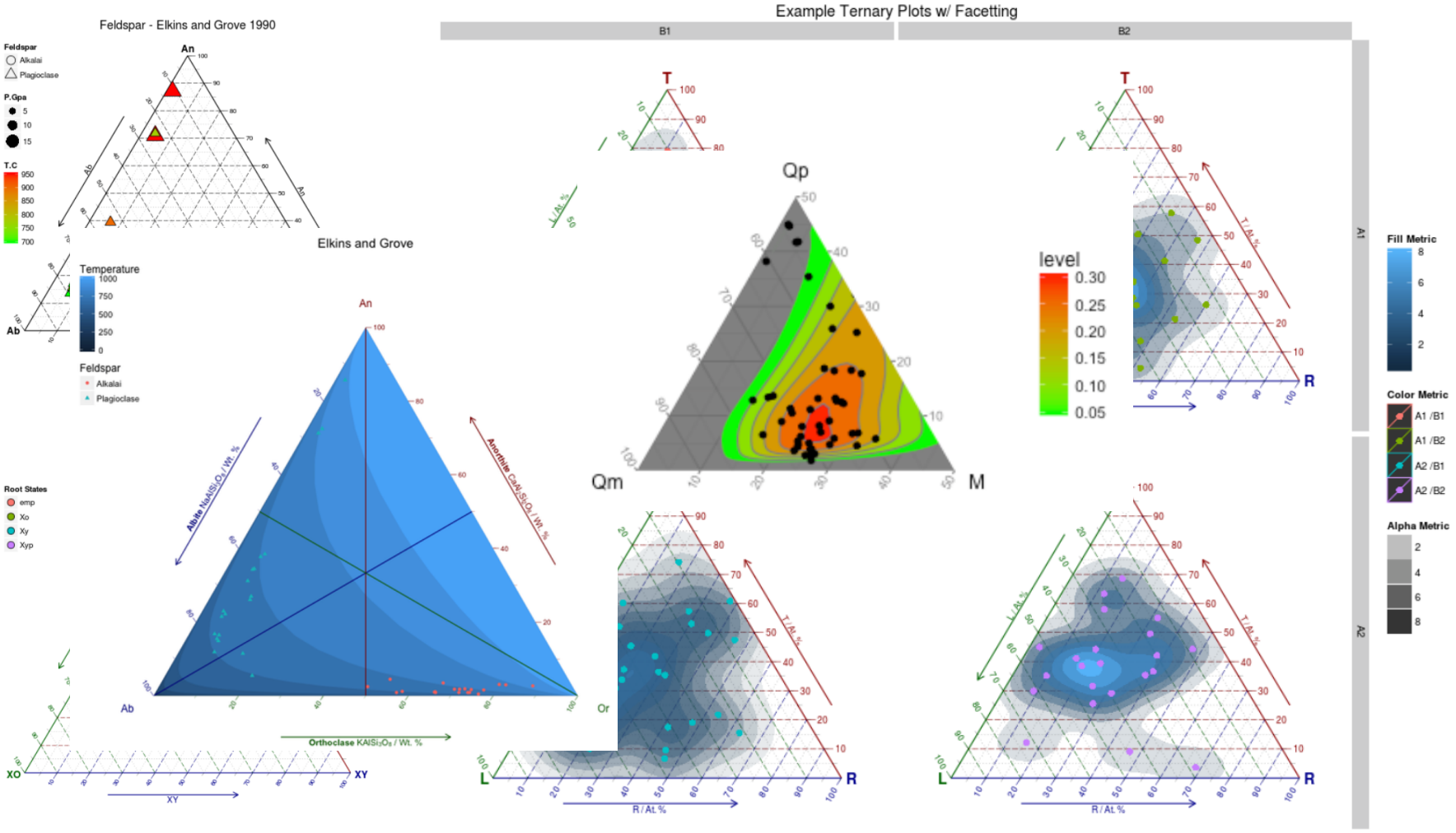
And for ggplot2 lovers …

ggtern package (http://www.ggtern.com): extension to ggplot2 for the plotting of ternary diagrams by Nicholas Hamilton



And for ggplot2 lovers …

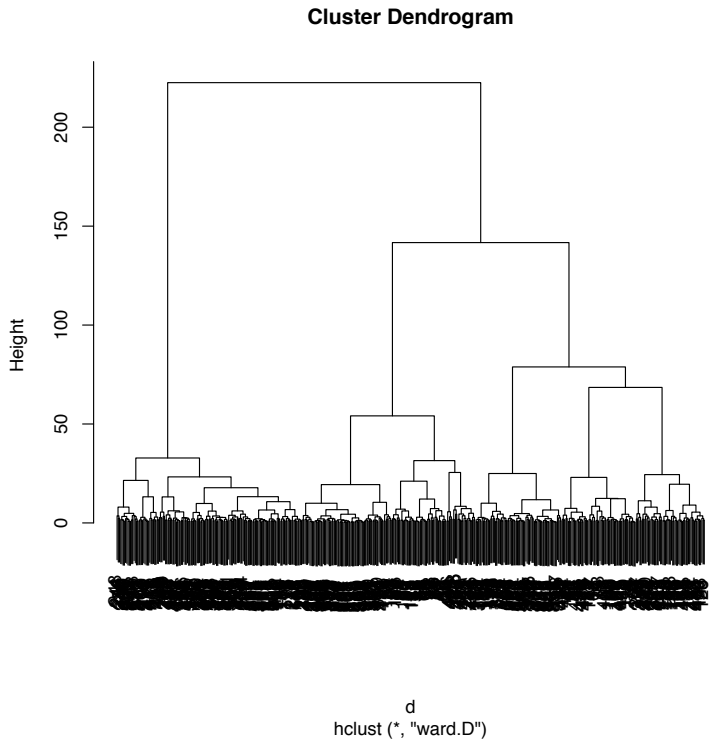
ggtern package (http://www.ggtern.com): extension to ggplot2 for the plotting of ternary diagrams by Nicholas Hamilton



Example: clustering analysis

- Using a coherent distance for compositions (Aitchison distance)

$$\mathrm{d_c}(\mathbf{x},\mathbf{y}) = \left(\frac{1}{D} \sum_{i=1}^{D-1} \sum_{j=i+1}^{D} \left(\log \frac{x_i}{x_j} - \log \frac{y_i}{y_j} \right)^2 \right)^{1/2}$$



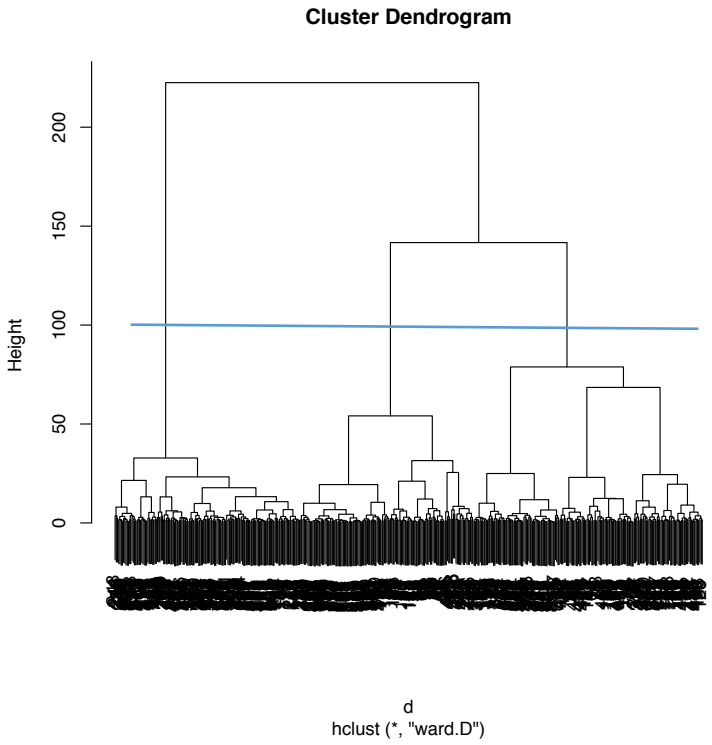

```r
# Clustering analysis
comp <- acomp(Hydrochem[,6:19])
d <- dist(comp) # Aitchison distance
h <- hclust(d,method = "ward.D") # Clustering by Ward's method
plot(h)
groups <- cutree(h,k=3) # 3-group solution
plot(acomp(comp,c("NH4","Na","Ca")),col = groups,center = T)
```

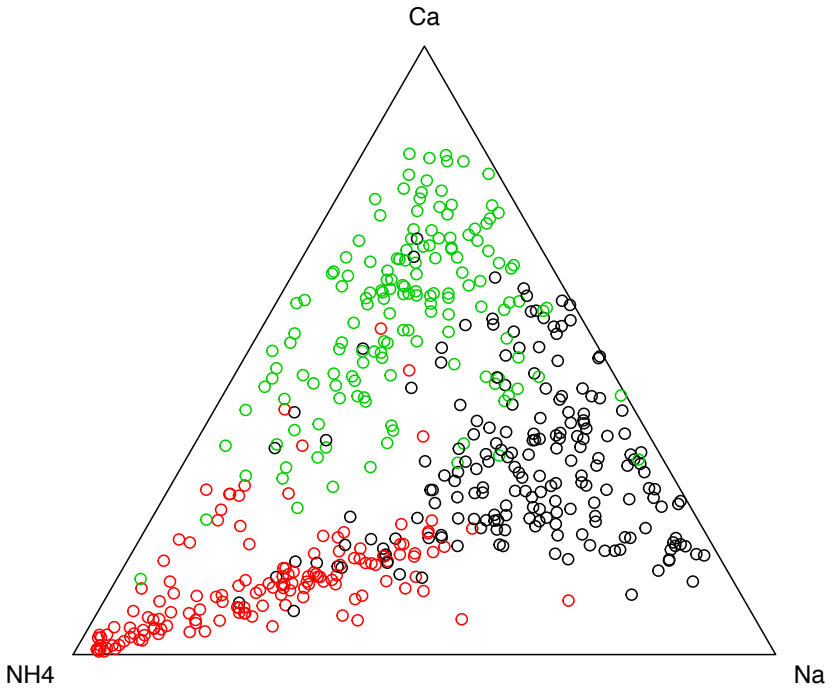
Example: clustering analysis

- Using a coherent distance for compositions (Aitchison distance)

$$\mathrm{d_c}(\mathbf{x},\mathbf{y}) = \left(\frac{1}{D} \sum_{i=1}^{D-1} \sum_{j=i+1}^{D} \left(\log \frac{x_i}{x_j} - \log \frac{y_i}{y_j} \right)^2 \right)^{1/2}$$

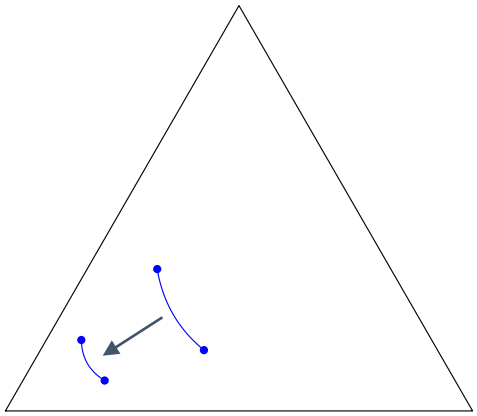
```
# Clustering analysis
comp <- acomp(Hydrochem[,6:19])
d <- dist(comp) # Aitchison distance
h <- hclust(d,method = "ward.D") # Clustering by Ward's method
plot(h)
groups <- cutree(h,k=3) # 3-group solution
plot(acomp(comp,c("NH4","Na","Ca")),col = groups,center = T)
```



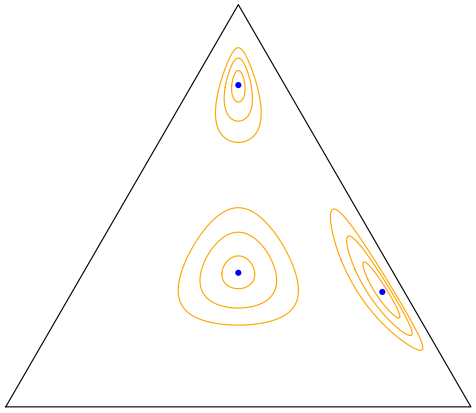


Modelling on log-ratio coordinates

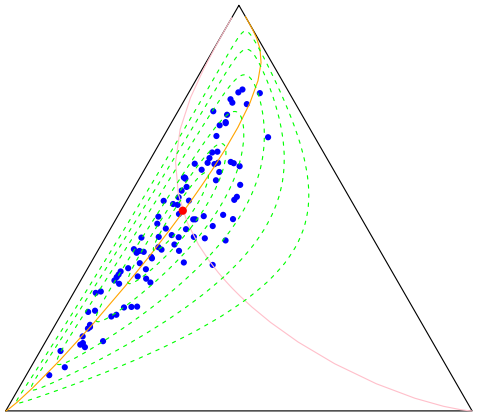
- The simplex is a (D-1)-dimensional *Euclidean vector space*



Translation (perturbation)



Circles



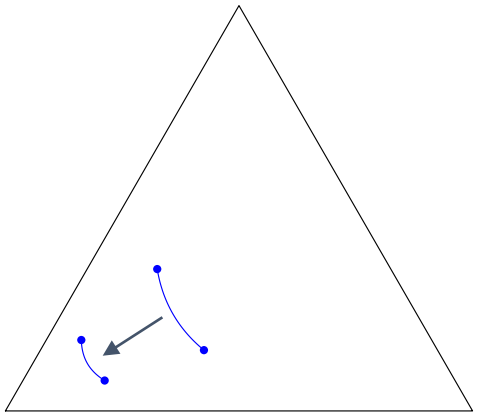
Normal distribution on S^3
+ principal components

Equivalence between the simplex and the real space R^{D-1}

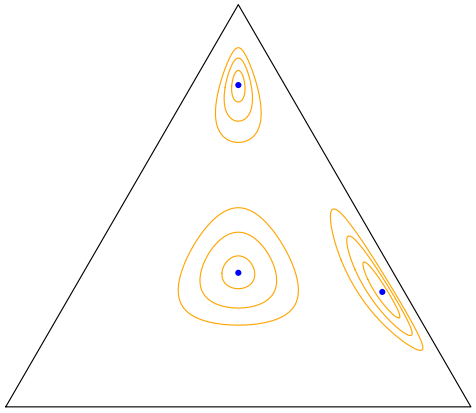
$$(0.2, 0.5, 0.3) = 0.2(1,0,0) + 0.5(0,1,0) + 0.3(0,0,1)$$
$$= \frac{1}{\sqrt{2}} \ln \frac{0.2}{0.5} \odot \mathbf{w}_1 \oplus \frac{1}{\sqrt{6}} \ln \frac{0.1}{0.09} \odot \mathbf{w}_2$$

Modelling on log-ratio coordinates

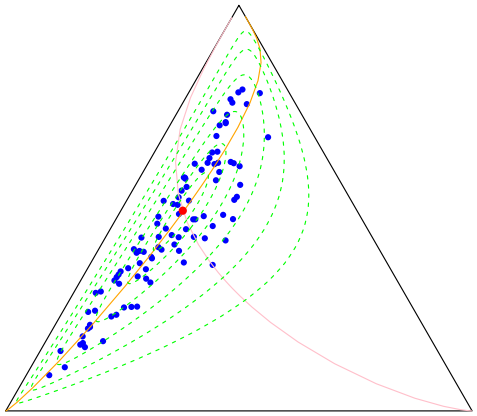
- The simplex is a (D-1)-dimensional *Euclidean vector space*



Translation (perturbation)



Circles



Normal distribution on S^3
+ principal components

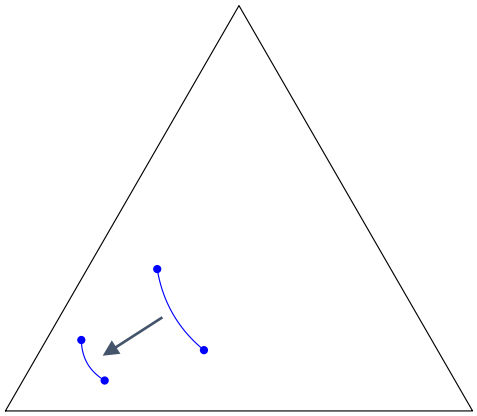
Equivalence between the simplex and the real space $\mathrm{R}^{\mathrm{D-1}}$

$$(0.2, 0.5, 0.3) = 0.2(1,0,0) + 0.5(0,1,0) + 0.3(0,0,1)$$
$$= \frac{1}{\sqrt{2}} \ln \frac{0.2}{0.5} \odot \mathbf{w}_1 \oplus \frac{1}{\sqrt{6}} \ln \frac{0.1}{0.09} \odot \mathbf{w}_2$$

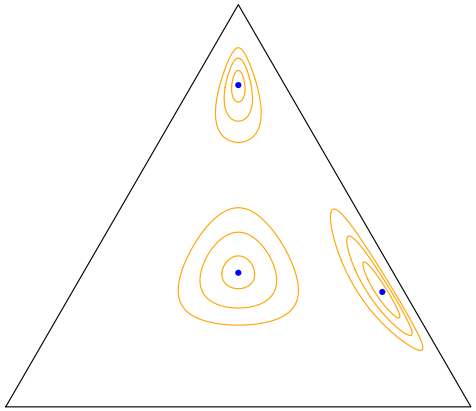
Orthonormal basis in S^3, **infinitely many!**

Modelling on log-ratio coordinates

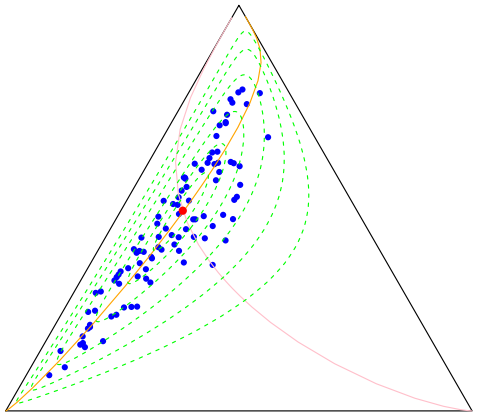
- The simplex is a (D-1)-dimensional *Euclidean vector space*



Translation (perturbation)



Circles



Normal distribution on S^3
+ principal components

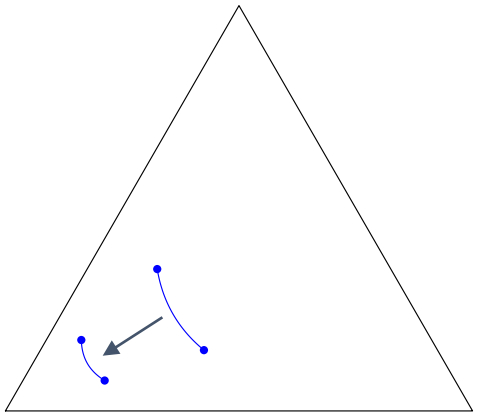
Equivalence between the simplex and the real space R^{D-1}

$$(0.2, 0.5, 0.3) = 0.2(1,0,0) + 0.5(0,1,0) + 0.3(0,0,1)$$
$$= \frac{1}{\sqrt{2}} \ln \frac{0.2}{0.5} \odot \mathbf{w}_1 \oplus \frac{1}{\sqrt{6}} \ln \frac{0.1}{0.09} \odot \mathbf{w}_2$$

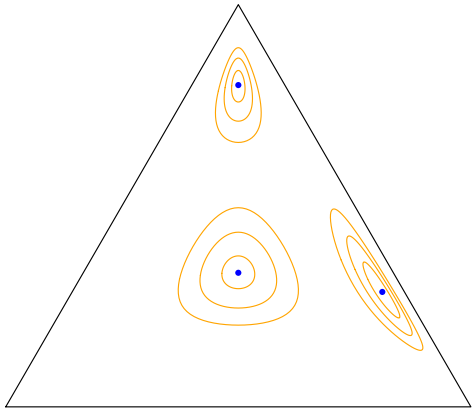
Analysis based on
isometric log-ratios
(ilr)

Modelling on log-ratio coordinates

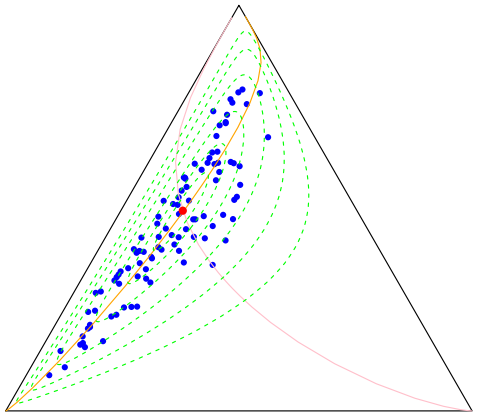
- The simplex is a (D-1)-dimensional *Euclidean vector space*



Translation (perturbation)



Circles



Normal distribution on S^3
+ principal components

Equivalence between the simplex and the real space R^{D-1}

$$(0.2, 0.5, 0.3) = 0.2(1,0,0) + 0.5(0,1,0) + 0.3(0,0,1)$$

$$= \frac{1}{\sqrt{2}} \ln \frac{0.2}{0.5} \odot \mathbf{w}_1 \oplus \frac{1}{\sqrt{6}} \ln \frac{0.1}{0.09} \odot \mathbf{w}_2$$

Analysis based on
isometric log-ratios (ilr)

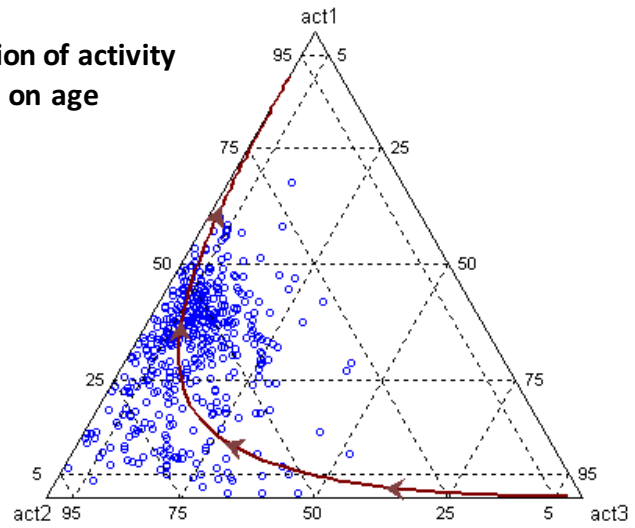
Statistical modeling mapped on the simplex through real (log-ratio) coordinates

Multivariate analysis based on log-ratios

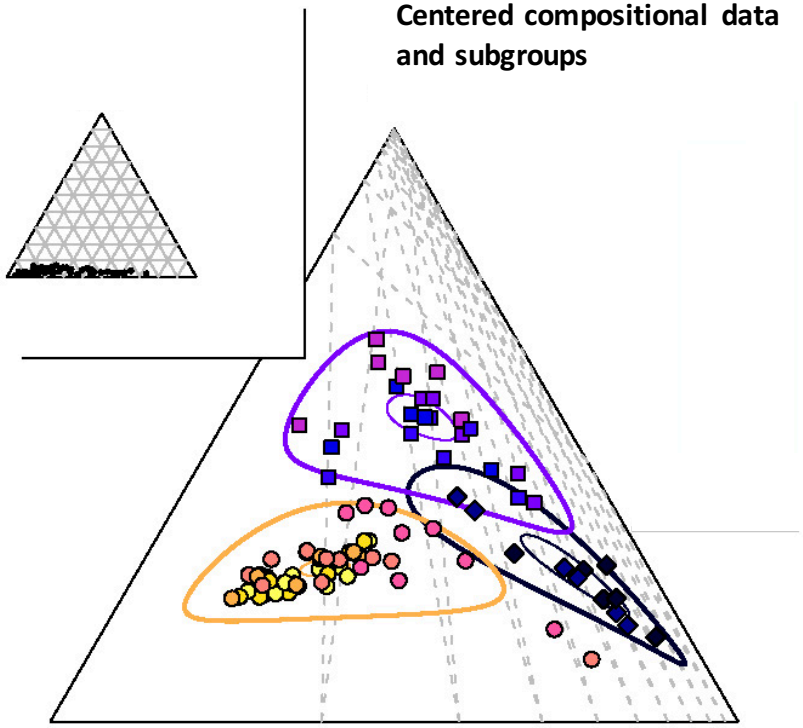
Ordinary multivariate data analysis techniques redefined in light of compositional principles:

- Principal component analysis
- Linear discriminant analysis
- Cluster analysis
- Canonical correlation analysis
- Multivariate regression/ANOVA
- …

Regression of activity patterns on age



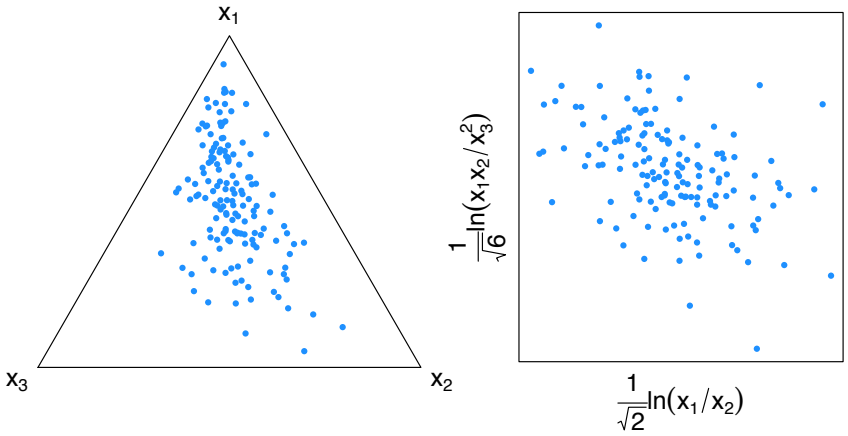
Centered compositional data and subgroups



Computing ilr coordinates

1. "Black-box" ilr transformation

$$y_i = ilr(\mathbf{x}) = \frac{1}{\sqrt{i(i+1)}} \ln \frac{\prod_{j=1}^{i} x_j}{x_{i+1}^{i}} \qquad i = 1, \ldots, D-1$$

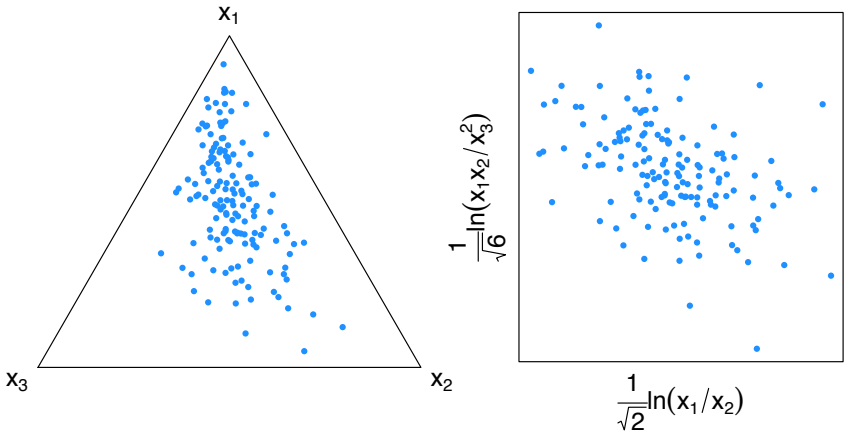



```
# ilr transformation
sub2[1:5,] # from 3D composition ...

##        Na       K     Mg
## 1 0.6169 0.02451 0.3586
## 2 0.6358 0.02288 0.3413
## 3 0.6440 0.02070 0.3353
## 4 0.6589 0.01881 0.3223
## 5 0.6207 0.03030 0.3490

sub2.ilr <- ilr(sub2)
sub2.ilr[1:5,] # ... to 2D real ilr coordinates

##      [,1]   [,2]
## 1 -2.281 0.8737
## 2 -2.351 0.8493
## 3 -2.431 0.8704
## 4 -2.514 0.8678
## 5 -2.135 0.7627
```

Computing ilr coordinates

1. "Black-box" ilr transformation

$$y_i = ilr(\mathbf{x}) = \frac{1}{\sqrt{i(i+1)}} \ln \frac{\prod_{j=1}^{i} x_j}{x_{i+1}^{i}} \qquad i = 1, \ldots, D-1$$




```
# Other lr transformations: alr and clr
sub2.alr <- alr(sub2) # additive lr transformation
sub2.alr[1,]

##      Na       K
##  0.5427 -2.6828

sub2.clr <- clr(sub2) # centred lr transformation
sub2.clr[1,]

##      Na       K      Mg
##  1.2561 -1.9694  0.7134
```

```
# ilr transformation
sub2[1:5,] # from 3D composition ...

##        Na       K      Mg
## 1 0.6169 0.02451 0.3586
## 2 0.6358 0.02288 0.3413
## 3 0.6440 0.02070 0.3353
## 4 0.6589 0.01881 0.3223
## 5 0.6207 0.03030 0.3490

sub2.ilr <- ilr(sub2)
sub2.ilr[1:5,] # ... to 2D real ilr coordinates

##      [,1]   [,2]
## 1 -2.281 0.8737
## 2 -2.351 0.8493
## 3 -2.431 0.8704
## 4 -2.514 0.8678
## 5 -2.135 0.7627
```

Computing ilr coordinates

2. Sequential binary partition (SBP) and balances (customised ilr representation)

$$[x_1, \ldots, x_D] \rightarrow [b_1, \ldots, b_{D-1}]$$

$$b_i = \sqrt{\frac{r_i s_i}{r_i + s_i}} \ln \frac{\left(\prod_{k=1}^{r_i} x_{ik}^{+}\right)^{1/r_i}}{\left(\prod_{k=1}^{s_i} x_{ik}^{-}\right)^{1/s_i}}, \; i = 1, \ldots, D-1$$

Example: 6-part volatile fatty acid (VFA) composition

SBP	b_1	b_2	b_3	b_4	b_5
Acetate			+	+	+
Propionate				−	+
Butyrate			−	+	+
Isobutyrate	+	−			−
Isovalerate	−	−			−
Valerate		+			−
r	1	1	1	2	3
s	1	2	1	1	3

Parts with "+" go into the numerator
Parts with "—" go into the denominator

Computing ilr coordinates

2. Sequential binary partition (SBP) and balances (customised ilr representation)

$$[x_1, \ldots, x_D] \rightarrow [b_1, \ldots, b_{D-1}]$$

$$b_i = \sqrt{\frac{r_i s_i}{r_i + s_i}} \ln \frac{\left(\prod_{k=1}^{r_i} x_{ik}^{+}\right)^{1/r_i}}{\left(\prod_{k=1}^{s_i} x_{ik}^{-}\right)^{1/s_i}}, \; i = 1, \ldots, D-1$$

Example: 6-part volatile fatty acid (VFA) composition

SBP	b_1	b_2	b_3	b_4	b_5
Acetate			+	+	+
Propionate				−	+
Butyrate			−	+	+
Isobutyrate	+	−			−
Isovalerate	−	−			−
Valerate		+			−
r	1	1	1	2	3
s	1	2	1	1	3

Parts with "+" go into the numerator
Parts with "—" go into the denominator

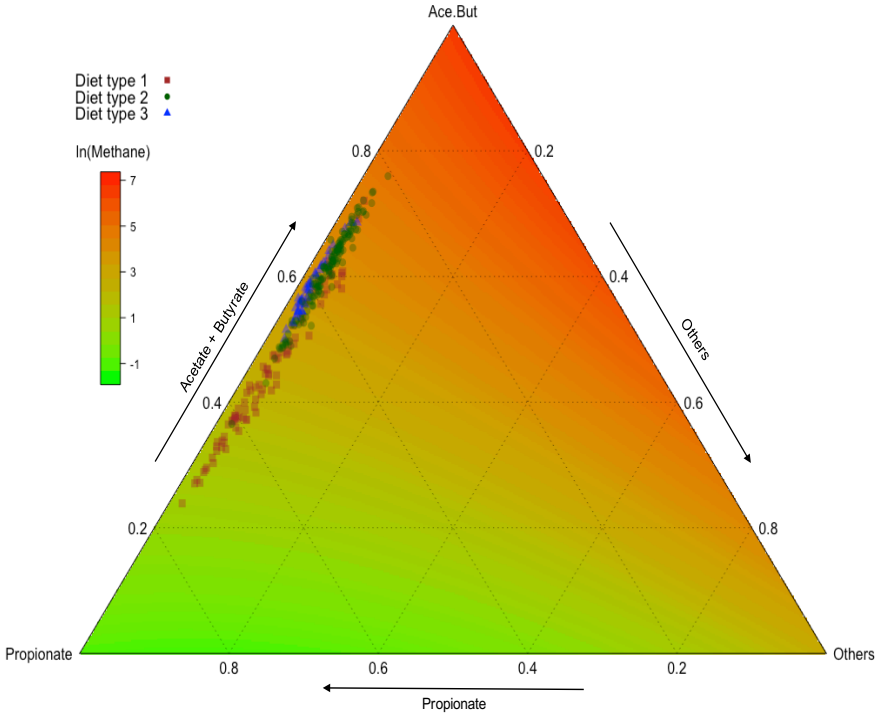
Define log-ratios between parts of practical interest

Computing ilr coordinates

2. Sequential binary partition (SBP) and balances (customised ilr representation)

$$[x_1, \ldots, x_D] \rightarrow [b_1, \ldots, b_{D-1}]$$

Using R …




```r
# Obtention of ilr balances
load("ExampleDataX.Rdata")
head(X) # VFA data set X
```

```
##      Acetate Propionate Butyrate Isobutyrate Isovalerate Valerate
## [1,]   558.0      310.0    96.00       12.000        7.000    17.00
## [2,]   477.5      417.4    64.06       11.011       15.015    15.02
## [3,]   542.5      326.7    93.91        9.990       12.987    13.99
## [4,]   562.6      329.3    76.08        9.009        8.008    15.02
## [5,]   551.0      329.0    78.00        9.000        6.000    27.00
## [6,]   537.5      292.3   113.11       14.014       28.028    15.02
```

```r
sm <- matrix(c(0,0,1,1,1, # create SBP matrix
               0,0,0,-1,1,
               0,0,-1,1,1,
               1,-1,0,0,-1,
              -1,-1,0,0,-1,
               0,1,0,0,-1),byrow=TRUE,ncol=5)
colnames(sm) <- c("b1","b2","b3","b4","b5")
rownames(sm) <- colnames(X)
B <- gsi.buildilrBase(sm) # built ilr basis according to sm
X.ilr <- ilr(X,B) # ilr coordinates
X.ilr[1:6,]
```

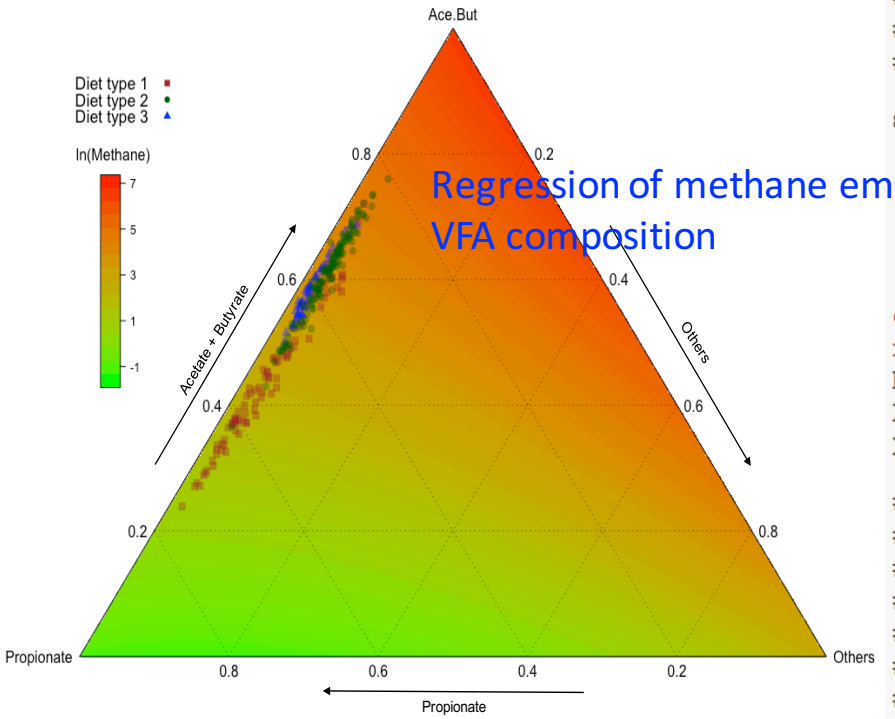
```
##            b1      b2    b3      b4    b5
## [1,]  0.38113  0.5044 1.245 -0.2386 3.822
## [2,] -0.21931  0.1266 1.420 -0.7103 3.489
## [3,] -0.18552  0.1676 1.240 -0.3019 3.725
## [4,]  0.08329  0.4652 1.415 -0.3796 3.868
## [5,]  0.28671  1.0625 1.382 -0.3771 3.748
## [6,] -0.49013 -0.2266 1.102 -0.1389 3.270
```

Computing ilr coordinates

2. Sequential binary partition (SBP) and balances (customised ilr representation)

$$[x_1, \dots, x_D] \rightarrow [b_1, \dots, b_{D-1}]$$

Using R …



Regression of methane emission on VFA composition

```
# Obtention of ilr balances
load("ExampleDataX.Rdata")
head(X) # VFA data set X

##      Acetate Propionate Butyrate Isobutyrate Isovalerate Valerate
## [1,]   558.0      310.0    96.00       12.000        7.000    17.00
## [2,]   477.5      417.4    64.06       11.011       15.015    15.02
## [3,]   542.5      326.7    93.91        9.990       12.987    13.99
## [4,]   562.6      329.3    76.08        9.009        8.008    15.02
## [5,]   551.0      329.0    78.00        9.000        6.000    27.00
## [6,]   537.5      292.3   113.11       14.014       28.028    15.02

sm <- matrix(c(0,0,1,1,1, # create SBP matrix
               0,0,0,-1,1,
               0,0,-1,1,1,
               1,-1,0,0,-1,
              -1,-1,0,0,-1,
               0,1,0,0,-1),byrow=TRUE,ncol=5)
colnames(sm) <- c("b1","b2","b3","b4","b5")
rownames(sm) <- colnames(X)
B <- gsi.buildilrBase(sm) # built ilr basis according to sm
X.ilr <- ilr(X,B) # ilr coordinates
X.ilr[1:6,]

##            b1      b2    b3      b4    b5
## [1,]  0.38113  0.5044 1.245 -0.2386 3.822
## [2,] -0.21931  0.1266 1.420 -0.7103 3.489
## [3,] -0.18552  0.1676 1.240 -0.3019 3.725
## [4,]  0.08329  0.4652 1.415 -0.3796 3.868
## [5,]  0.28671  1.0625 1.382 -0.3771 3.748
## [6,] -0.49013 -0.2266 1.102 -0.1389 3.270
```

Dealing with zeros and left-censoring (the `zCompositions` package)

`zCompositions: Imputation of Zeros and Nondetects in Compositional Data Sets`

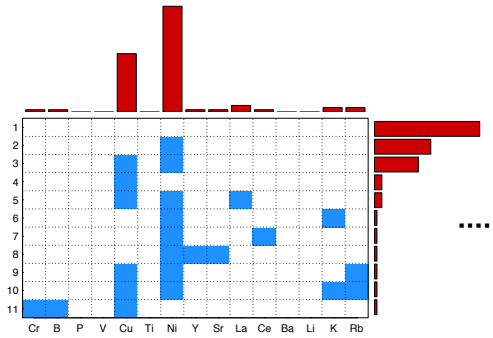
Implements principled methods to impute multivariate left-censored data and zeros in compositional data sets.

Version:	1.0.3-1
Depends:	R (≥ 2.14.0), MASS, NADA, truncnorm
Published:	2016-04-14
Author:	Javier Palarea-Albaladejo and Josep Antoni Martin-Fernandez
Maintainer:	Javier Palarea-Albaladejo <javier.palarea at bioss.ac.uk>

Palarea-Albaladejo, Martín-Fernández (2015), **zCompositions — R package for multivariate imputation of left-censored data under a compositional approach**, *Chemometrics and Intelligent Laboratory Systems* 143, 85–96.

- Typically rounded zeros or trace elements at concentrations below the detection limit (DL) of the analytical instrument (reported as "*<DL", less-thans, nondetects*)
- `zCompositions` offers exploration tools and parametric and non-parametric methods to deal with zeros and left-censoring within a compositional framework
- Ability to work with closed/non-closed continuous/discrete CoDa and varying DLs

Dealing with zeros and left-censoring (the `zCompositions` package)



[zPatterns]

Input data
(x)

Target value label
(label)

Thresholds
(dl)

Log-ratio EM
[lrEM]

rob
ini.cov
delta
tolerance
max.iter
rlm.maxit

Log-ratio DA
[lrDA]

ini.cov
delta
n.iters
m
store.mi

Multiplicative replacement
[multRepl]

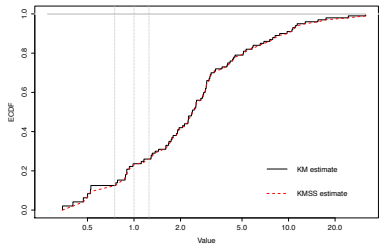
delta

Multiplicative lognormal
[multLN]

rob
random

Multiplicative KMSS
[multKM]

n.draws
n.knots



[splineKM]

Count multiplicative
[cmultRepl]

method
output
delta
threshold
correct
t
s

Scaling

Replaced data

Dealing with zeros and left-censoring (the `zCompositions` package)

```
data(LPdata) # non-closed data (ppm/micrograms per gram)

  Cr  B   P  V  Cu   Ti   Ni  Y  Sr La Ce  Ba  Li   K    Rb

 33.3 23 393 47 5.3 3715  9.1 24 38  9 180  48  76 16617  53
  8.9  8  96 14 0.0 1625  0.0 12 28  0  34  19  25  8509  35
 33.8 17 206 52 6.9 4135  0.0 11 19  8   0  85  67 23119  82
  8.3  7 322 15 0.0 1265  9.0  8 12  2  18  13  37  6514  37
  5.7  7 113 15 0.0 1235  0.0  5  9  0  63  21  41  6811  27
  7.1  4 225 14 3.6 1055  0.0  5 62  2  49  27  81     0  15

dl <- c(2, 1, 0, 0, 2, 0, 6, 1, 0.6, 1, 1, 0, 0, 632, 10) # single DLs
LPdata_lrEM <- lrEM(LPdata, label = 0, dl = dl)

No. iterations to converge:  32

  Cr  B   P  V  Cu   Ti   Ni  Y  Sr La   Ce   Ba  Li   K     Rb

 33.3 23 393 47 5.3 3715  9.1 24 38  9   180   48  76 16617   53
  8.9  8  96 14 0.9 1625  1.6 12 28  0.9  34   19  25  8509   35
 33.8 17 206 52 6.9 4135  3.3 11 19  8     0.9 85  67 23119   82
  8.3  7 322 15 1.1 1265  9.0  8 12  2    18   13  37  6514   37
  5.7  7 113 15 0.9 1235  1.7  5  9  0.9  63   21  41  6811   27
  7.1  4 225 14 3.6 1055  3.3  5 62  2    49   27  81   445.3 15
```

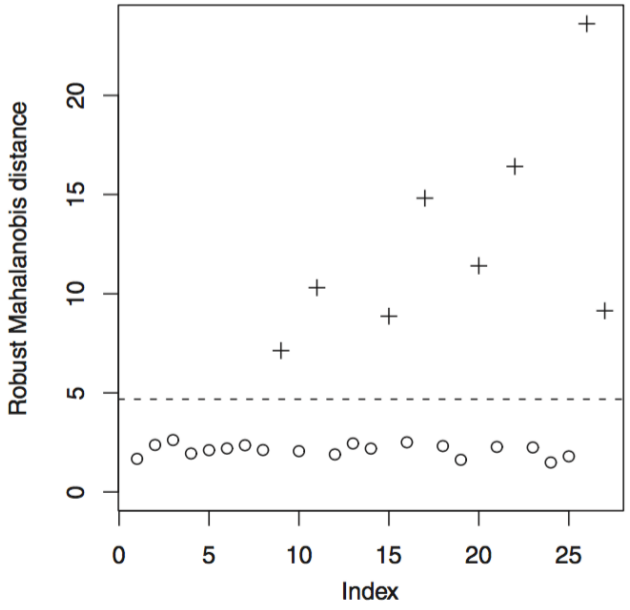
Robust approach to CoDa (the `robCompositions` package)

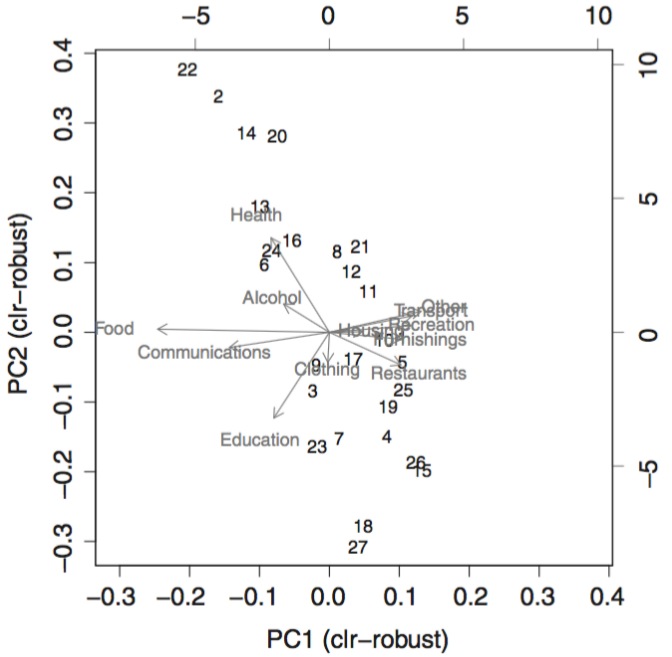
robCompositions: Robust Estimation for Compositional Data

Methods for analysis of compositional data including robust methods, imputation, methods to replace rounded zeros, (robust) outlier detection for compositional data, (robust) principal component analysis for compositional data, (robust) factor analysis for compositional data, (robust) discriminant analysis for compositional data (Fisher rule), robust regression with compositional predictors and (robust) Anderson-Darling normality tests for compositional data as well as popular log-ratio transformations (addLR, cenLR, isomLR, and their inverse transformations). In addition, visualisation and diagnostic tools are implemented as well as high and low-level plot functions for the ternary diagram.

Version:	2.0.0
Depends:	R (≥ 2.10), robustbase, ggplot2, data.table, pls
Imports:	e1071, cvTools, rrcov, GGally, MASS, sROC, VIM
Suggests:	knitr
Published:	2016-02-08
Author:	Matthias Templ, Karel Hron, Peter Filzmoser
Maintainer:	Matthias Templ <templ at tuwien.ac.at>

- Outlier detection methods and some “off the shelf” multivariate techniques using a fixed log-ratio transformation
- Emphasis on robust estimation procedures





So, in summary …

- `compositions`: provides the fundamentals for principled compositional data analysis
- `zCompositions`: methods to deal with zeros and left-censoring in CoDa
- `robCompositions`: outlier detection and robust methods for CoDa
- `ggtern`: ternary plots based on `ggplot2`
- Some other functions spread in other R packages.

In summary …

- `compositions`: provides the fundamentals for compositional data analysis
- `zCompositions`: principled methods to deal with zeros and left-censoring in CoDa
- `robCompositions`: outlier detection and robust methods for CoDa
- `ggtern`: ternary plots based on `ggplot2`
- Some other functions (mostly log-ratio transformations) spread in a few other R packages.