



# Functional Diversity through space and time: Methods for calculating FD

By Lindsey and Vanessa

# Outline

## 1. Functional Diversity:

- a) Functional richness – Fric
- b) Functional evenness - FEve
- c) Functional divergence - FDiv
- d) Functional dispersion – Fdis
- e) Functional Group Richness – FGR

## Authors:

Mason *et. al* 2005

Laberte and Legandgre 2010

## 2. Null modeling

Kraft, 2008

## 3. Beta diversity

- a) Functional Sorensons –Fsor
- b) Nearest Neighbor distance – Dnn
- c) Pairwise dissimilarity - DPw

Swenson, 210

# Functional Diversity

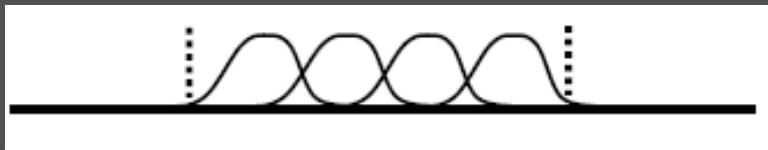
The distribution of the species and abundance of a community in niche space, including:

- a) the amount of niche space filled by species in the community (functional richness)
- b) the evenness of abundance distribution in filled niche space (functional evenness)
- c) the degree to which abundance distribution in niche space maximizes divergence in functional characters within the community (functional divergence).

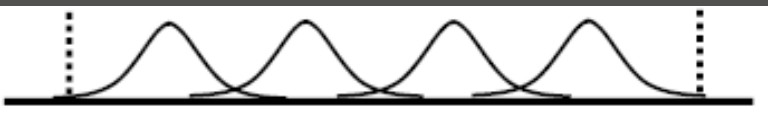
# Functional Diversity

Low

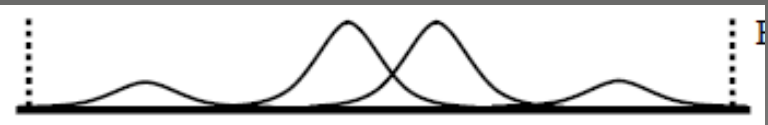
High



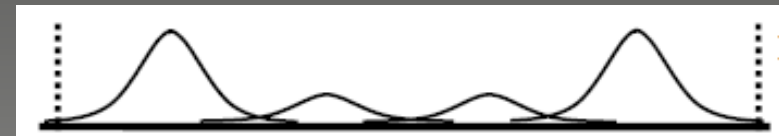
Functional richness



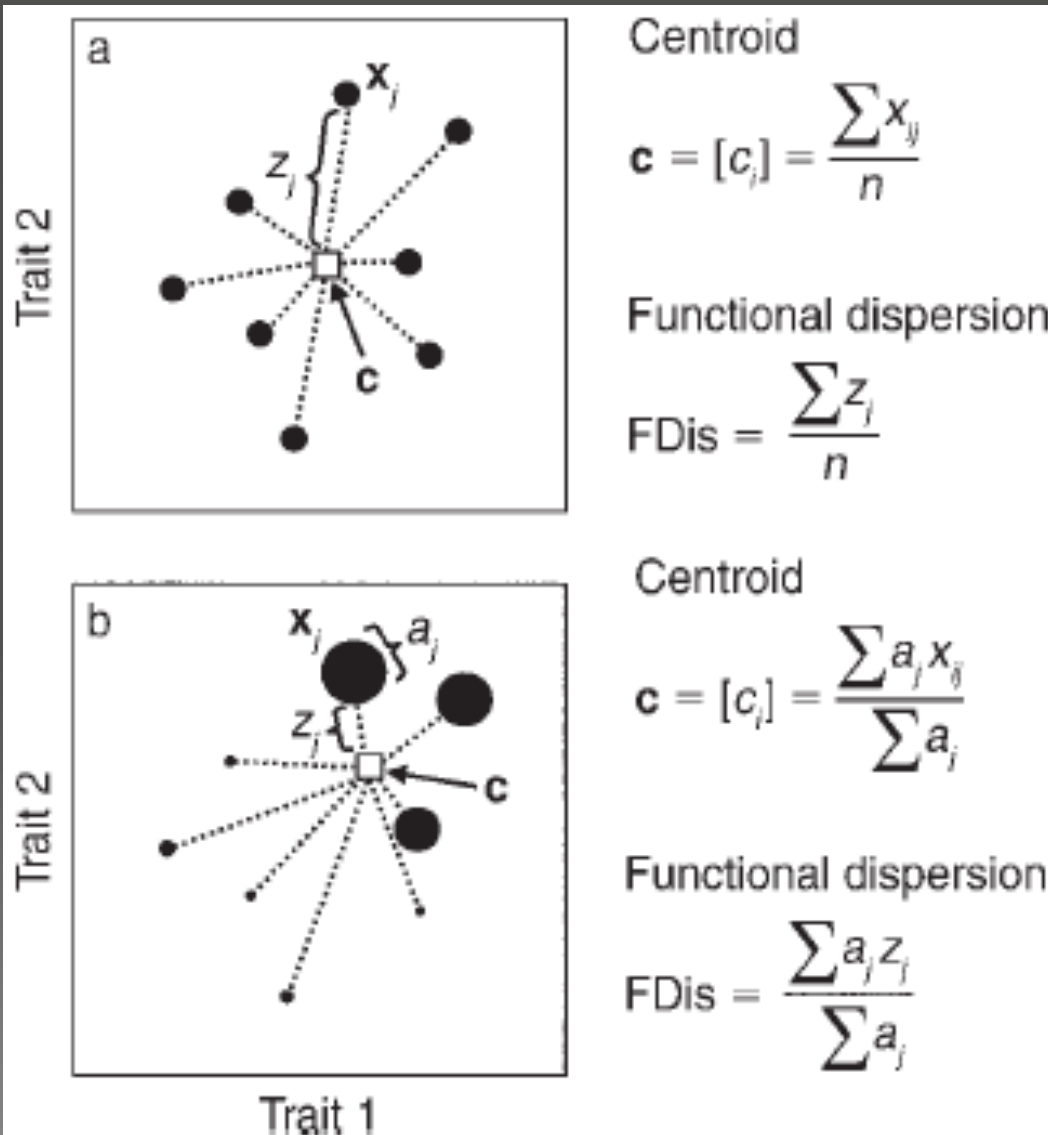
Functional evenness



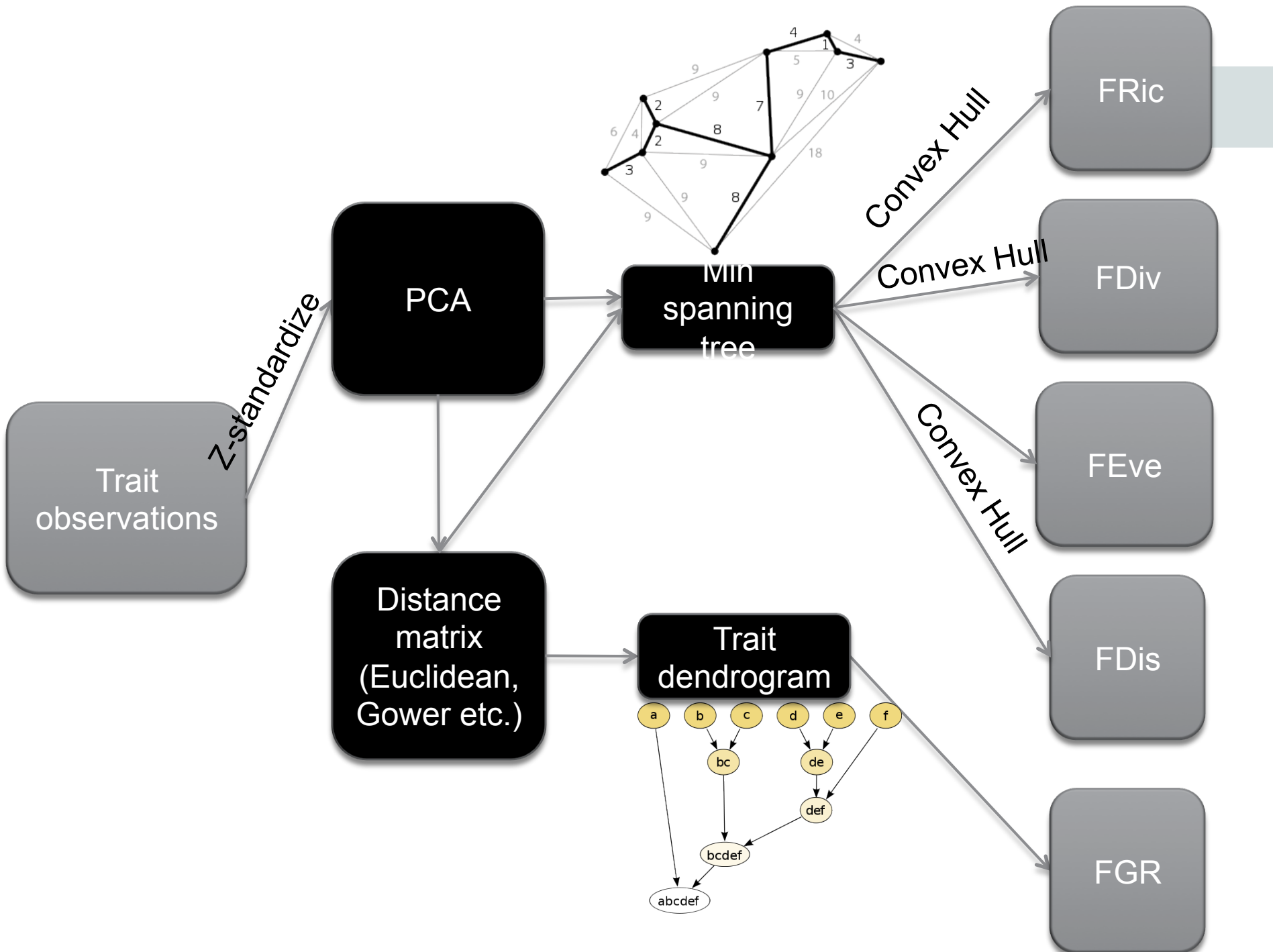
Functional divergence



# Functional Dispersion



Is the multivariate analog of the weighted mean absolute deviation (MAD), or Rao's Q



# dummy\$trait

trait data frame of 8 functional traits on 8 species

	num1	num2	fac1	fac2	ord1	ord2	bin1	bin2
sp1	9	4.5	A	X	3	2	0	1
sp2	8.1	6	A	Z	NA	1	0	1
sp3	NA	2.3	C	Y	5	3	1	1
sp4	3.2	5.4	B	Z	1	7	0	0
sp5	5.8	1.2	C	X	2	6	NA	0
sp6	3.4	8.5	C	Y	2	1	1	1
sp7	7.5	2.1	B	X	3	2	1	0
sp8	4.3	6.5	NA	Z	1	3	0	1

# dummy\$abun

matrix of abundances of the 8 species from 10 communities

	sp1	sp2	sp3	sp4	sp5	sp6	sp7	sp8
com1	1	1	0	0	4	2	0	0
com2	0	0	0	2	1	0	0	5
com3	2	0	0	0	0	1	0	3
com4	1	0	7	0	0	0	0	0
com5	0	0	2	3	3	0	0	0
com6	0	3	0	0	5	6	1	6
com7	3	5	0	3	0	0	0	0
com8	0	0	0	0	6	2	1	2
com9	4	1	1	3	0	0	2	0



# Dummy.dist <- gowdis(dummy\$trait)

	sp1	sp2	sp3	sp4	sp5	sp6	sp7	sp8
sp1	0	0.21818836	0.52400522	0.67374429	0.52911128	0.61001614	0.44842348	0.40728344
sp2	0.21818836	0	0.66780822	0.56100277	0.81456988	0.59325865	0.68633736	0.20394426
sp3	0.52400522	0.66780822	0	0.82257012	0.48622527	0.27847358	0.48486628	0.59589041
sp4	0.67374429	0.56100277	0.82257012	0	0.48432643	0.70739254	0.5575126	0.23909621
sp5	0.52911128	0.81456988	0.48622527	0.48432643	0	0.60673235	0.30234159	0.55855246
sp6	0.61001614	0.59325865	0.27847358	0.70739254	0.60673235	0	0.61878444	0.44702072
sp7	0.44842348	0.68633736	0.48486628	0.5575126	0.30234159	0.61878444	0	0.70301865
sp8	0.40728344	0.20394426	0.59589041	0.23909621	0.55855246	0.44702072	0.70301865	0

- gowdis computes the Gower (1971) similarity coefficient exactly as described by Podani (1999), then converts it to a dissimilarity coefficient by using  $D = 1 - S$ . It integrates variable weights as described by Legendre and Legendre (1998).

```
dummy.fdisp<-fdisp(dummy.dist,dummy.abun)
```

comm	eig
1	0.51601447
2	0.41108607
3	0.25021121
4	0.07199211
5	9.71E-17
6	-0.0067201
7	-0.074215
8	-0.0913455

	V1	V2
sp1	-0.0744394	0.10529491
sp2	-0.3701972	0.17301902
sp3	0.30011889	0.26033177
sp4	-0.217381	-0.3757648
sp5	0.27449437	-0.257528
sp6	0.10644436	0.25870469
sp7	0.263649	-0.1668387
sp8	-0.282689	0.00278108

- Dummy.fdisp\$eig
- Dummy.fdisp\$vectors

```
Dummy.dbFD <- dbFD(dummy$trait, dummy$abun)
```

	nbspp	sing.sp	FRic	qual.FRic	FEve	FDiv	FDis	RaoQ
com1	4	4	0.17420135	0.61083523	0.84323343	0.84292209	0.34816867	0.1283544
com2	3	3	0.10209717	0.61083523	0.46286348	0.85932501	0.16705601	0.04063622
com3	3	3	0.00215764	0.61083523	0.86596574	0.63030311	0.23758085	0.06497224
com4	2	2	NA	0.61083523	NA	NA	0.11462614	0.03003235
com5	3	3	0.1431512	0.61083523	0.90812085	0.76313459	0.32113662	0.11858388
com6	5	5	0.2310837	0.61083523	0.86517337	0.89666873	0.33023297	0.11738479
com7	3	3	0.07368337	0.61083523	0.76819911	0.83560952	0.2532751	0.07868047
com8	4	4	0.17422061	0.61083523	0.760665	0.86811628	0.28779307	0.09308505
com9	5	5	0.3374462	0.61083523	0.79946382	0.70151183	0.34216865	0.12431265
com10	4	4	0.22890412	0.61083523	0.49446007	0.97125544	0.3503927	0.12490783

## What do all of these errors mean?

Species x species distance matrix was not Euclidean. 'sqrt' correction was applied.

FEVe: Could not be calculated for communities with <3 functionally singular species.

FRic: To respect  $s > t$ , FRic could not be calculated for communities with <3 functionally singular species.

FRic: Dimensionality reduction was required. The last 5 PCoA axes (out of 7 in total) were removed.

FRic: Quality of the reduced-space representation (based on corrected distance matrix) = 0.6108352

FDiv: Could not be calculated for communities with <3 functionally singular species.

```
Dummy.dbFD <- dbFD(dummy$trait, dummy$abun)
```

	nbspp	sing.sp	FRic	qual.FRic	FEve	FDiv	FDis	RaoQ
com1	4	4	0.17420135	0.61083523	0.84323343	0.84292209	0.34816867	0.1283544
com2	3	3	0.10209717	0.61083523	0.46286348	0.85932501	0.16705601	0.04063622
com3	3	3	0.00215764	0.61083523	0.86596574	0.63030311	0.23758085	0.06497224
com4	2	2	NA	0.61083523	NA	NA	0.11462614	0.03003235
com5	3	3	0.1431512	0.61083523	0.90812085	0.76313459	0.32113662	0.11858388
com6	5	5	0.2310837	0.61083523	0.86517337	0.89666873	0.33023297	0.11738479
com7	3	3	0.07368337	0.61083523	0.76819911	0.83560952	0.2532751	0.07868047
com8	4	4	0.17422061	0.61083523	0.760665	0.86811628	0.28779307	0.09308505
com9	5	5	0.3374462	0.61083523	0.79946382	0.70151183	0.34216865	0.12431265
com10	4	4	0.22890412	0.61083523	0.49446007	0.97125544	0.3503927	0.12490783

“Species x species distance matrix was not Euclidean. 'sqrt' correction was applied.”

This is because “dbFD” does a PCoA and then has to account for native eigenvalues. The package comes with four possible correction methods and the default here is to take the square root of the distances. This doesn’t always work so you always choose another correction method.

```
Dummy.dbFD <- dbFD(dummy$trait, dummy$abun)
```

FRic: To respect  $s > t$ , FRic could not be calculated for communities with <3 functionally singular species.

FRic: Dimensionality reduction was required. The last 5 PCoA axes (out of 7 in total) were removed.

FRic: Quality of the reduced-space representation (based on corrected distance matrix) = 0.6108352

FDiv: Could not be calculated for communities with <3 functionally singular species

Fric  
FDiv  
Feve

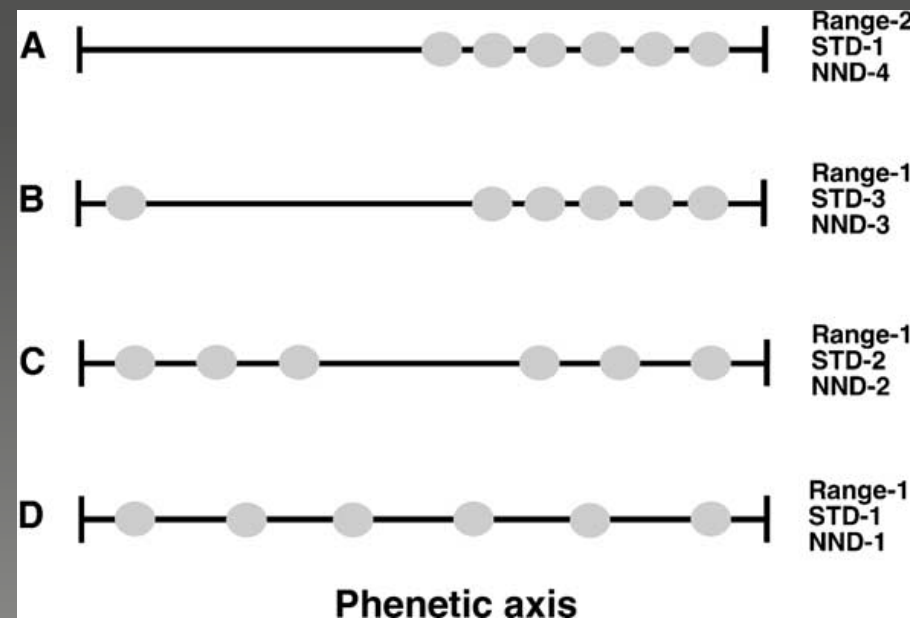
} requires convex hull measurement

For these two measures you need more species than traits. If you don't have this you can subset of the PCoA axes as "traits." Since this is a loss of information you can use "**qual.Fric**" as an  $R^2$  type ratio. In this example we have about 0.61

# Functional Richness - FRic

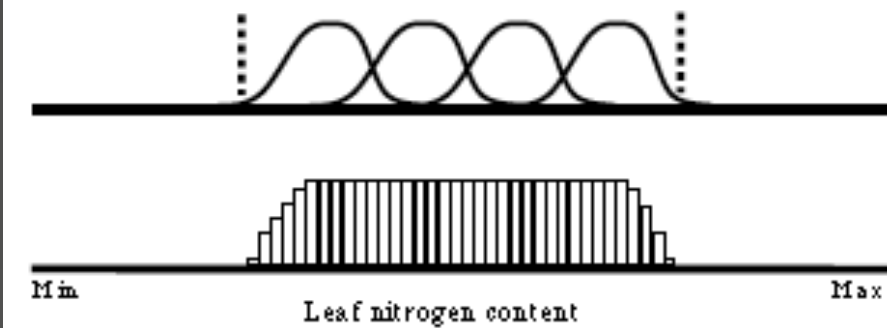
	nbspp	sing.sp	FRic	equal.FRic	FEve	FDiv	FDis	RaoQ
com1	4		0.17420135	0.61083523	0.84323343	0.84292209	0.34816867	0.1283544
com2	3		0.10209717	0.61083523	0.46286348	0.85932501	0.16705601	0.04063622
com3	3		0.00215764	0.61083523	0.86596574	0.63030311	0.23758085	0.06497224
com4	2		NA	0.61083523	NA	NA	0.11462614	0.03003235
com5	3		0.1431512	0.61083523	0.90812085	0.76313459	0.32113662	0.11858388
com6	5		0.2310837	0.61083523	0.86517337	0.89666873	0.33023297	0.11738479
com7	3		0.07368337	0.61083523	0.76819911	0.83560952	0.2532751	0.07868047
com8	4		0.17422061	0.61083523	0.760665	0.86811628	0.28779307	0.09308505
com9	5		0.3374462	0.61083523	0.79946382	0.70151183	0.34216865	0.12431265
com10	4		0.22890412	0.61083523	0.49446007	0.97125544	0.3503927	0.12490783

FRic is the convex hull volume.  
It can be abundance weighted.  
If it's one trait it's the "range"



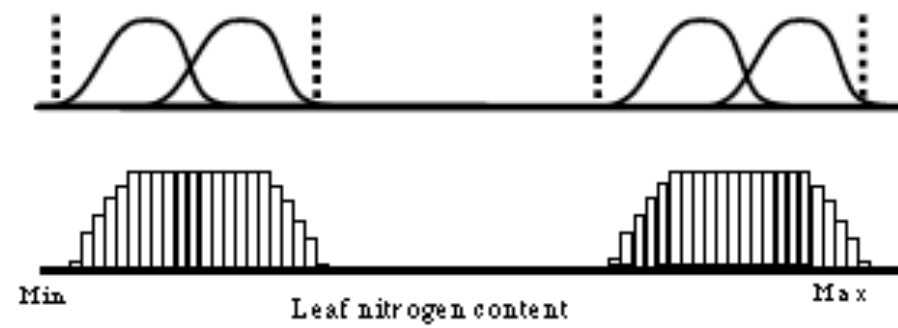
# Functional Richness - FRic

(A1) Lower functional richness



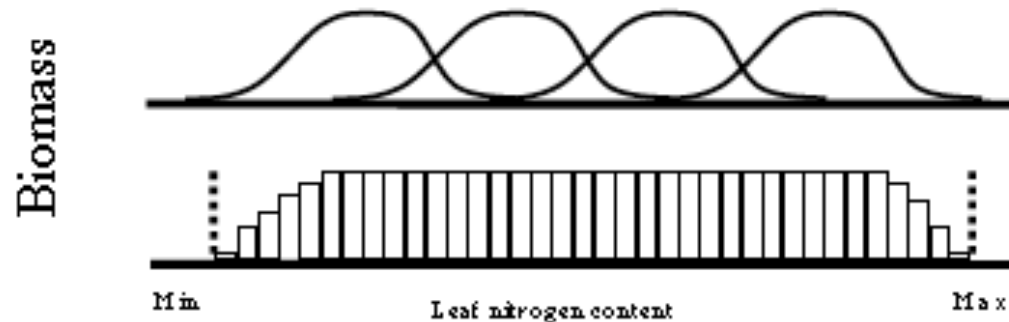
$$FR_{ci} = 0.500$$
$$E_{var} = 0.773$$

(A2) Lower functional richness



$$FR_{ci} = 0.500$$
$$E_{var} = 0.644$$

(B) High functional evenness and functional richness

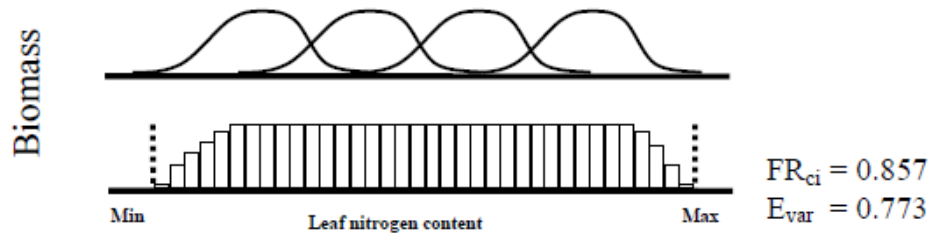


$$FR_{ci} = 0.857$$
$$E_{var} = 0.773$$

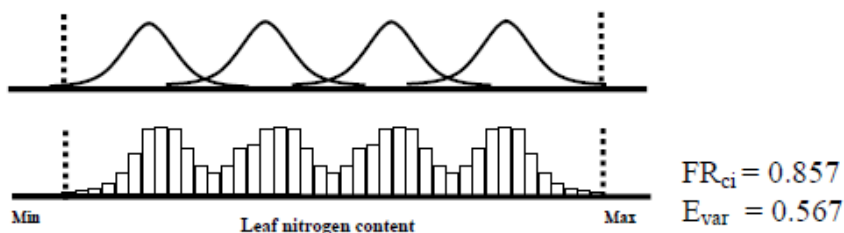
# Functional Evenness - FEve

	nbspp	sing.sp	FRic	qual.FRic	FEve	Div	FDis	RaoQ
com1	4	4	0.17420135	0.6108352	0.84323343	0.84292209	0.34816867	0.1283544
com2	3	3	0.10209717	0.6108352	0.46286348	0.85932501	0.16705601	0.04063622
com3	3	3	0.00215764	0.6108352	0.86596574	0.63030311	0.23758085	0.06497224
com4	2	2	NA	0.6108352	NA	NA	0.11462614	0.03003235
com5	3	3	0.1431512	0.6108352	0.90812085	0.76313459	0.32113662	0.11858388
com6	5	5	0.2310837	0.6108352	0.86517337	0.89666873	0.33023297	0.11738479
com7	3	3	0.07368337	0.6108352	0.76819911	0.83560952	0.2532751	0.07868047
com8	4	4	0.17422061	0.6108352	0.760665	0.86811628	0.28779307	0.09308505
					0.79946382	0.70151183	0.34216865	0.12431265
					0.49446007	0.97125544	0.3503927	0.12490783

(B) High functional evenness and functional richness



(C) Lower functional evenness



FEve - the evenness of abundance distribution in filled niche space

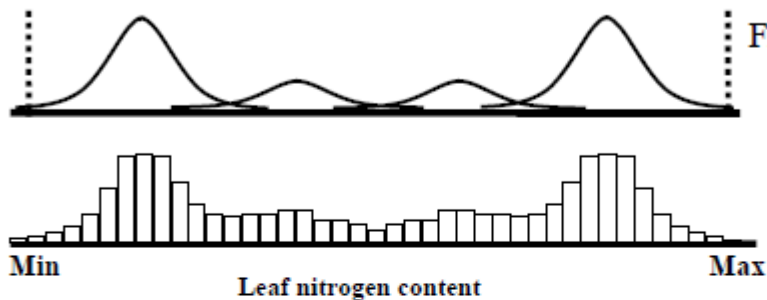


# Functional Divergence - FDiv

	nbspp	sing.sp	FRic	qual.FRic	FEve	FDiv	Dis	RaoQ
com1	4	4	0.17420135	0.61083523	0.8432334	0.84292209	0.34816867	0.1283544
com2	3	3	0.10209717	0.61083523	0.4628634	0.85932501	0.16705601	0.04063622
com3	3	3	0.00215764	0.61083523	0.8659657	0.63030311	0.23758085	0.06497224
com4	2	2	NA	0.61083523	NA	NA	0.11462614	0.03003235
com5	3	3	0.1431512	0.61083523	0.9081208	0.76313459	0.32113662	0.11858388
com6	5	5	0.2310837	0.61083523	0.8651733	0.89666873	0.33023297	0.11738479
					0.7681991	0.83560952	0.2532751	0.07868047
					0.76066	0.86811628	0.28779307	0.09308505
					0.7994638	0.70151183	0.34216865	0.12431265
					0.4944600	0.97125544	0.3503927	0.12490783

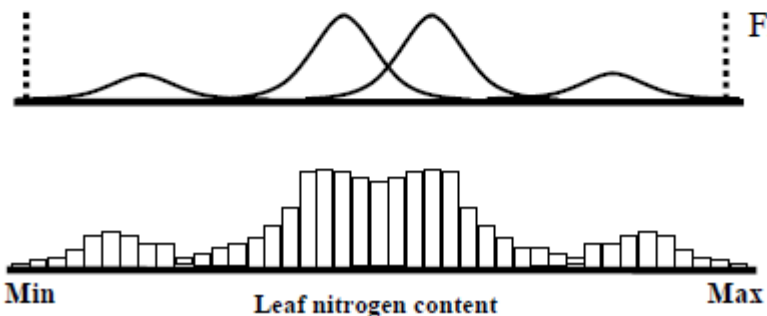
(A) High functional divergence

$FD_{var} = 0.830$



(B) Low functional divergence

$FD_{var} = 0.597$



Fdiv - the degree to which abundance distribution in niche space maximizes divergence in functional characters within the community

# Functional Dispersion - FDis

	nbspp	sing.sp	FRic	qual.FRic	FEve	FDiv	FDis	RaoQ
com1	4	4	0.17420135	0.61083523	0.84323343	0.8429220	0.34816867	0.1283544
com2	3	3	0.10209717	0.61083523	0.46286348	0.8593250	0.16705601	0.04063622
com3	3	3	0.00215764	0.61083523	0.86596574	0.6303031	0.23758085	0.06497224
com4	2	2	NA	0.61083523	NA	NA	0.11462614	0.03003235
com5	3	3	0.1431512	0.61083523	0.90812085	0.7631345	0.32113662	0.11858388
com6	5	5	0.2310837	0.61083523	0.86517337	0.8966687	0.33023297	0.11738479
com7	3	3	0.07368337	0.61083523	0.76819911	0.8356095	0.2532751	0.07868047
com8	4	4	0.17422061	0.61083523	0.760665	0.8681162	0.28779307	0.09308505
com9	5	5	0.3374462	0.61083523	0.79946382	0.7015118	0.34216865	0.12431265
com10	4	4	0.22890412	0.61083523	0.49446007	0.9712554	0.3503927	0.12490783

- Average distance to the (abundance weighted) centroid
- FDis is unaffected by species richness
- Can be computed from any distance or dissimilarity measure
- it can handle any number and type of traits
- not strongly influenced by outliers
- Similar to Rao's Q
- Can be used as a beta diversity measure

# CWM

CWM.num1	CWM.num2	CWM.fac1	CWM.fac2	CWM.ord1	CWM.ord2	CWM.bin1	CWM.bin2
5.8875	4.0375	C	X	2	6	0	1
4.2125	5.5625	B	Z	1	3	0	1
5.71666667	6.16666667	A	Z	1	3	0	1
9	2.575	C	Y	5	3	1	1
4.5	3.05	C	X	1	6	0	0
5.0952381	5.52857143	C	Z	2	1	0	1
7.00909091	5.42727273	A	Z	3	1	0	1
5.24545455	3.57272727	C	X	2	6	1	0
6.87	4.24545455	A	X	3	2	0	1
7.42727273	3.78333333	B	X	3	2	1	0

- Community weighted mean trait values.

# FGR

```
withFGR <- dbFD(dummy$trait, dummy$abun, calc.FGR = T)
withFGR$FGR
```

	FGR
com1	3
com2	3
com3	2
com4	2
com5	3
com6	3
com7	2
com8	3
com9	4
com10	3

spfgr	FGR
sp1	1
sp2	1
sp3	2
sp4	3
sp5	4
sp6	2
sp7	4
sp8	1

- The number of functional groups per community based on the dendrogram.
- The default cluster type is “ward,” but there are many clustering types

```
Dummy.functcomp <- functcomp(dummy$trait,
dummy$abun)
```

	num1	num2		fac2	ord1	ord2	bin1	bin2
com1	5.8875	4.0375		X	2	6	0	0
com2	4.2125	5.5625		Z	1	3	0	1
com3	5.71666667	6.16666667		Z	1	3	0	1
com4	9	2.575		Y	5	3	1	1
com5	4.5	3.05		X	2	7	0	0
com6	5.0952381	5.52857143		Z	2	1	0	1
com7	7.00909091	5.42727273		Z	3	1	0	1
com8	5.24545455	3.57272727		X	2	6	1	0
com9	6.87	4.24545455		X	3	2	0	1
com10	7.42727273	3.72727273		X	3	2	1	0

- functcomp returns the functional composition of a set of communities, as measured by the community level weighted means of trait values
- For a continuous trait CWM is the mean trait value of all species present in the community, weighted by their relative abundances.

```
Dummy.functcomp <- functcomp(dummy$trait,  
dummy$abun)
```

	num1	num2
com1	5.8875	4.166666667
com2	4.2125	5.166666667
com3	5.716666667	6.166666667
com4	9	2
com5	4.5	1
com6	5.0952381	5.528571429
com7	7.00909091	5.427272727
com8	5.24545455	3.572727273
com9	6.87	4.245454545
com10	7.42727273	3.783333333

fac1	fac2	ord1	ord2	bin1	bin2
C	X	2	6	0	0
B	Z	1	3	0	1
A	Z	1	3	0	1
C	Y	5	3	1	1
C	X	2	7	0	0
C	Z	2	1	0	1
A	Z	3	1	0	1
C	X	2	6	1	0
A	X	3	2	0	1
B	X	3	2	1	0

- For ordinal, nominal and binary traits, either the dominant class is returned, or the abundance of each individual class is returned (when CWM.type is "all" ).

# Null Models: Using Kraft's R-code as an example.

- Test if species in a community are a random subset of potential colonists from a larger species pool with respect to a single continuous trait.
- Test trait mean, range, variance, kurtosis, and three metrics of even spacing (SDNN, SDNNr, SDNDr).
- Compares the rank of the observed community to the null distribution (can be used to calculate a p-value), it also calculates a standard effect size.
- Random draws can be weighted by a species abundance or occurrence frequency.

# Run all the functions from the null model script first

- Now lets make and add the necessary data from the FD package.
- First load the species data which includes:
  - Species list
  - Two cwm traits for each species
  - Abundance
- `fd_sp<-read.csv("fd_sp.csv",header=T)`
- `sp<-c("sp1", "sp2", "sp3", "sp4", "sp5", "sp6", "sp7", "sp8")`
- `trait<-(fd_sp$sp.num2)`
- `abundance<-floor(fd_sp$abund)`



```
pool<-data.frame(sp, trait)
```

must contain all the species in all communities

sp	trait
sp1	4.5
sp2	6
sp3	2.3
sp4	5.4
sp5	1.2
sp6	8.5
sp7	2.1
sp8	6.5

```
sp_list_community1<-c("sp1", "sp2","sp5","sp6")
test_trait_data(sp_list_community1, pool, log=TRUE,
reps=999, abweight=TRUE, abdata=abundance,
verbose=TRUE)->summary_verbose
summary_verbose
```

test.richness	reps	mean_rank	range_rank	SDNN_rank	SDNNr_rank	SDNDr_rank
4	999	542	850	825	NA	NA
var_rank	obs_mean	null_mean_mean	null_mean_sd	obs_range	null_mean_range	
850	0.609991	0.5818826	0.135866	0.8502377	0.5617867	
null_range_sd	obs_SDNN	null_mean_SDNN	null_SDNN_sd	obs_kurt	null_mean_kurt	
0.2082579	0.2205077	0.1340082	0.08878143	2.363065	NaN	
null_kurt_sd	obs_var	null_mean_var	null_var_sd	obs_SDNNr	null_mean_SDNNr	
NA	0.1379797	0.07823636	0.04787506	0.2593483	NaN	
null_SDNNr_sd	obs_SDNDr	null_mean_SDNDr	null_SDNDr_sd	mean_ES	range_ES	var_ES
NA	0.3486316	NA	NA	0.2068834	1.385066	1.247901
SDNN_ES	SDNNr_ES	SDNDr_ES	kurt_ES			
0.9742975	NA	NA	NA			

# Do this for each community that has a minimum of 4 species

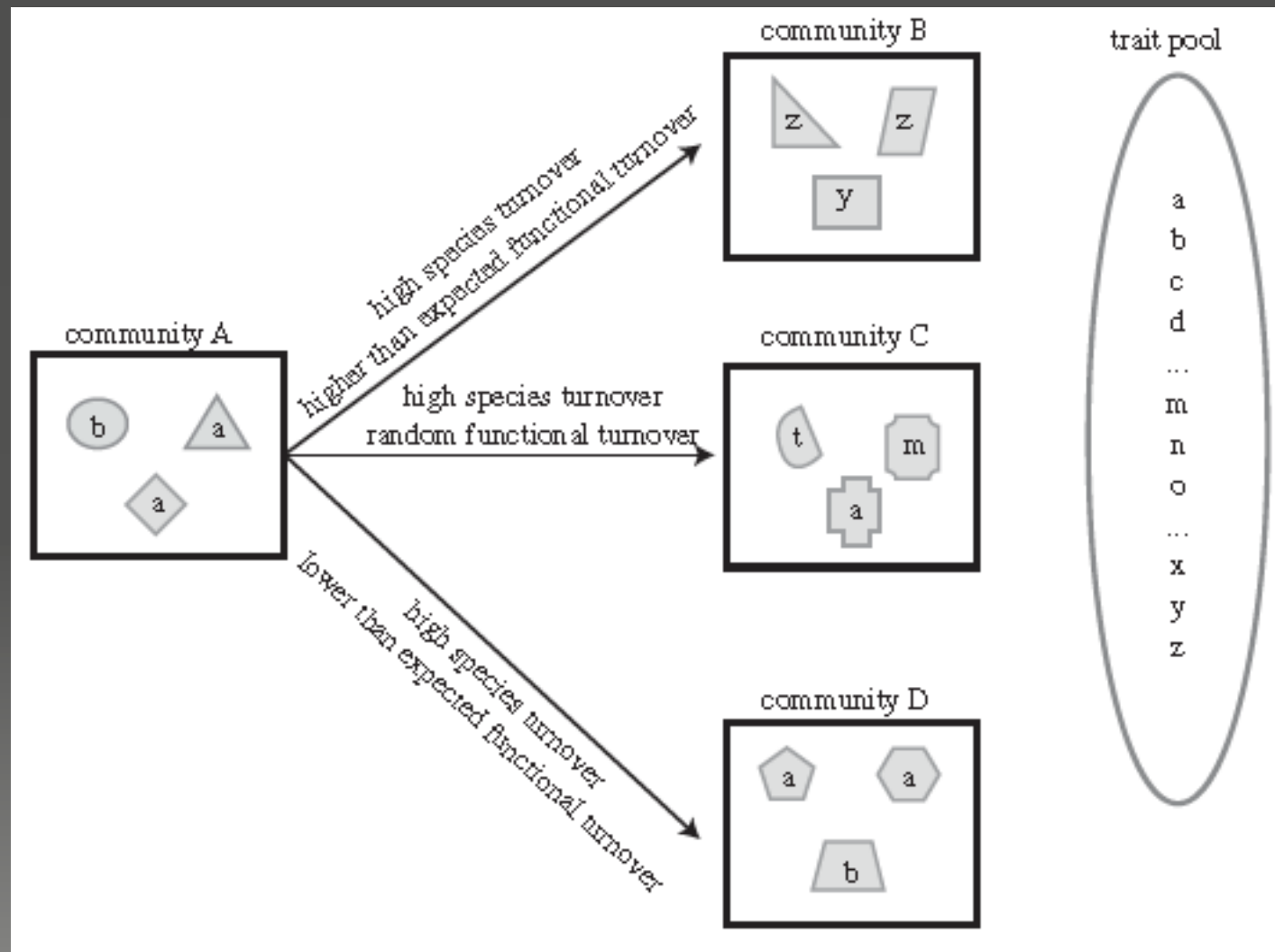
- `sp_list_community6<-c("sp2", "sp5","sp6","sp7","sp8")`  
`test_trait_data(sp_list_community6, pool, log=TRUE, reps=999, abweight=TRUE, abdata=abundance, verbose=TRUE)->summary_verbose`
- **summary\_verbose**
- `sp_list_community8<-c("sp5","sp6","sp7","sp8")`  
`test_trait_data(sp_list_community8, pool, log=TRUE, reps=999, abweight=TRUE, abdata=abundance, verbose=TRUE)->summary_verbose`
- **summary\_verbose**
- `sp_list_community9<-c("sp1","sp2","sp3","sp4","sp7")`  
`test_trait_data(sp_list_community9, pool, log=TRUE, reps=999, abweight=TRUE, abdata=abundance, verbose=TRUE)->summary_verbose`
- **summary\_verbose**
- `sp_list_community10<-c("sp2","sp3","sp7","sp8")`  
`test_trait_data(sp_list_communit10, pool, log=TRUE, reps=999, abweight=TRUE, abdata=abundance, verbose=TRUE)->summary_verbose`
- **summary\_verbose**

```
null<-read.csv("null_graph.csv",header=T)
```

species data including observed and null mean trait values for each community

community	observed_mean	null_mean
com1	0.609991	0.5800384
com6	0.5843768	0.5804775
com8	0.5359332	0.5812109
com9	0.5695409	0.5835467
com10	0.5687529	0.5760407

# Trait beta diversity



# Trait beta diversity

- Functional Sorenson's Shared function between two communities

$$F_{\text{sor}} = \frac{\text{BL}_{k_1 k_2}}{(\text{BL}_{k_1} + \text{BL}_{k_2})1/2}$$

where  $\text{BL}_{k_1 k_2}$  is the total dendrogram branch length common to all species in communities  $k_1$  and  $k_2$ , and  $\text{BL}_{k_1}$  and  $\text{BL}_{k_2}$  are the total dendrogram branch length common to the species within communities  $k_1$  and  $k_2$ , respectively.

- Nearest neighbor distance

More functionally similar communities will have lower nearest neighbor distances

$$D_{\text{nn}} = \frac{\sum_{i=1}^{n_{k_1}} f_i \min \delta_{ik_2} + \sum_{j=1}^{n_{k_2}} f_j \min \delta_{jk_1}}{2}$$

where  $n_{k_1}$  is the number of species in community  $k_1$ ,  $f_i$  the relative abundance of species  $i$  in community  $k_1$ , and  $\min \delta_{ik_2}$  the dendrogram branch lengths between species  $i$  in community  $k_1$  and its nearest functional neighbour in community  $k_2$  that is not the same species. Thus if two communities

- Pairwise dissimilarity All pairwise distances between species

$$D_{\text{pw}} = \frac{\sum_{i=1}^{n_{k_1}} f_i \overline{\delta_{ik_2}} + \sum_{j=1}^{n_{k_2}} f_j \overline{\delta_{jk_1}}}{2}$$

where  $\overline{\delta_{ik_2}}$  is the mean pairwise distance on the dendrogram between all species in community  $k_1$  and all species in community  $k_2$  excluding conspecific species. The  $D_{\text{pw}}$  is

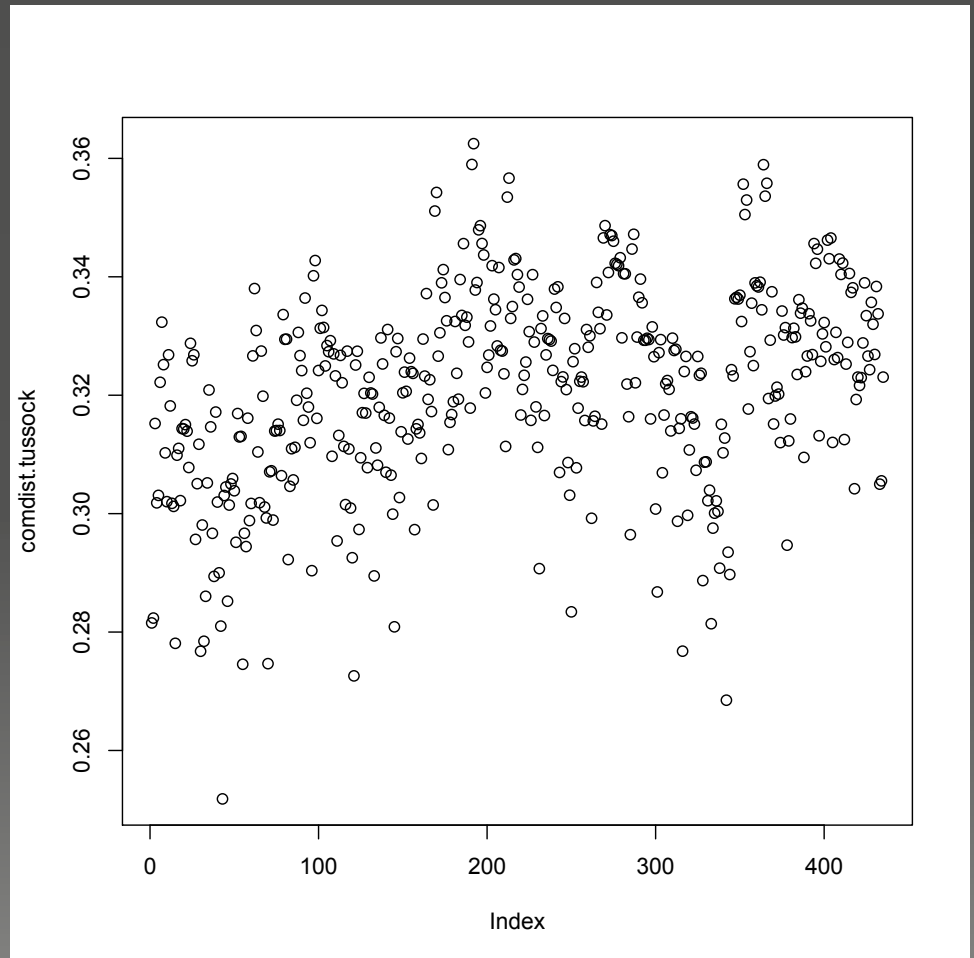
# Mean Pairwise distance

Calculates mean pairwise distance separating taxa in a community

```
dis.trait <- gowdis(tussock$trait)
dis.traitm<-as.matrix(dis.trait) #53 rows and 53 columns
abun.m<-as.matrix(tussock$abun) #30 rows and 53 columns
tussock.mpd<-mpd(abun.m, dis.traitm) #
ses.mpd.tussock.abundweighted<- ses.mpd (abun.m,dis.traitm,
    null.model="taxa.labels",abundance.weighted=TRUE) #standard
    effect size, not abundance weighted
```

# Nearest neighbor distance

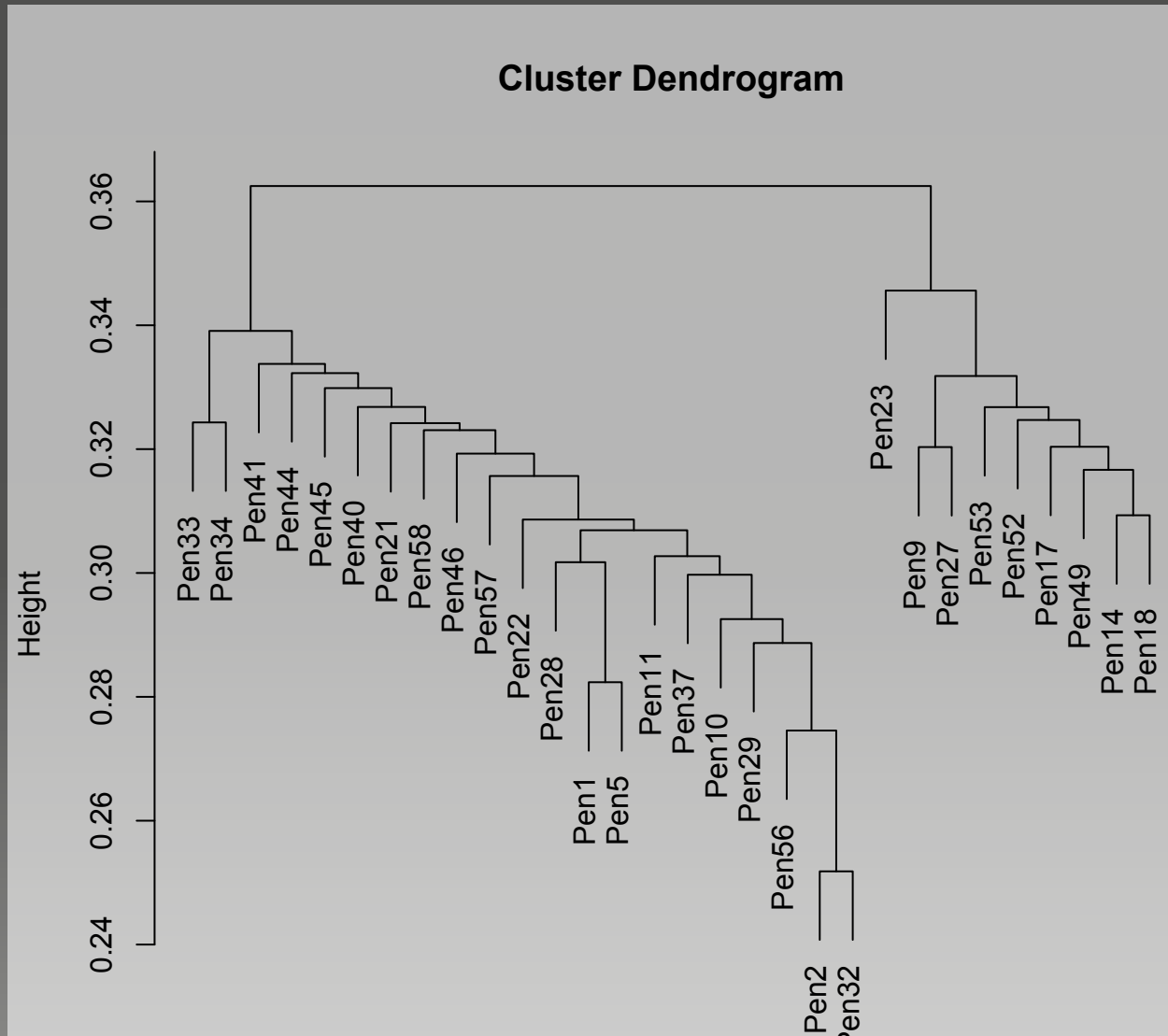
```
comdist.tussock <- comdist(abun.m,dis.traitm)  
plot(comdist.tussock)
```





# Trait dendrogram

```
comdist.clusters.tussock <- hclust(comdist.tussock)
plot(comdist.clusters.tussock)
```



# Phylosor



Shared function between two communities

```
clust.dist.trait<-hclust(dis.trait)
clust.dist.trait.asphylo<-as.phylo(clust.dist.trait)
phylsor.tussock<-phylosor(abun.m,clust.dist.trait.asphylo)
```

# Figures

