Multivariate questions

Many variables, we don't want to treat one as the 'response'

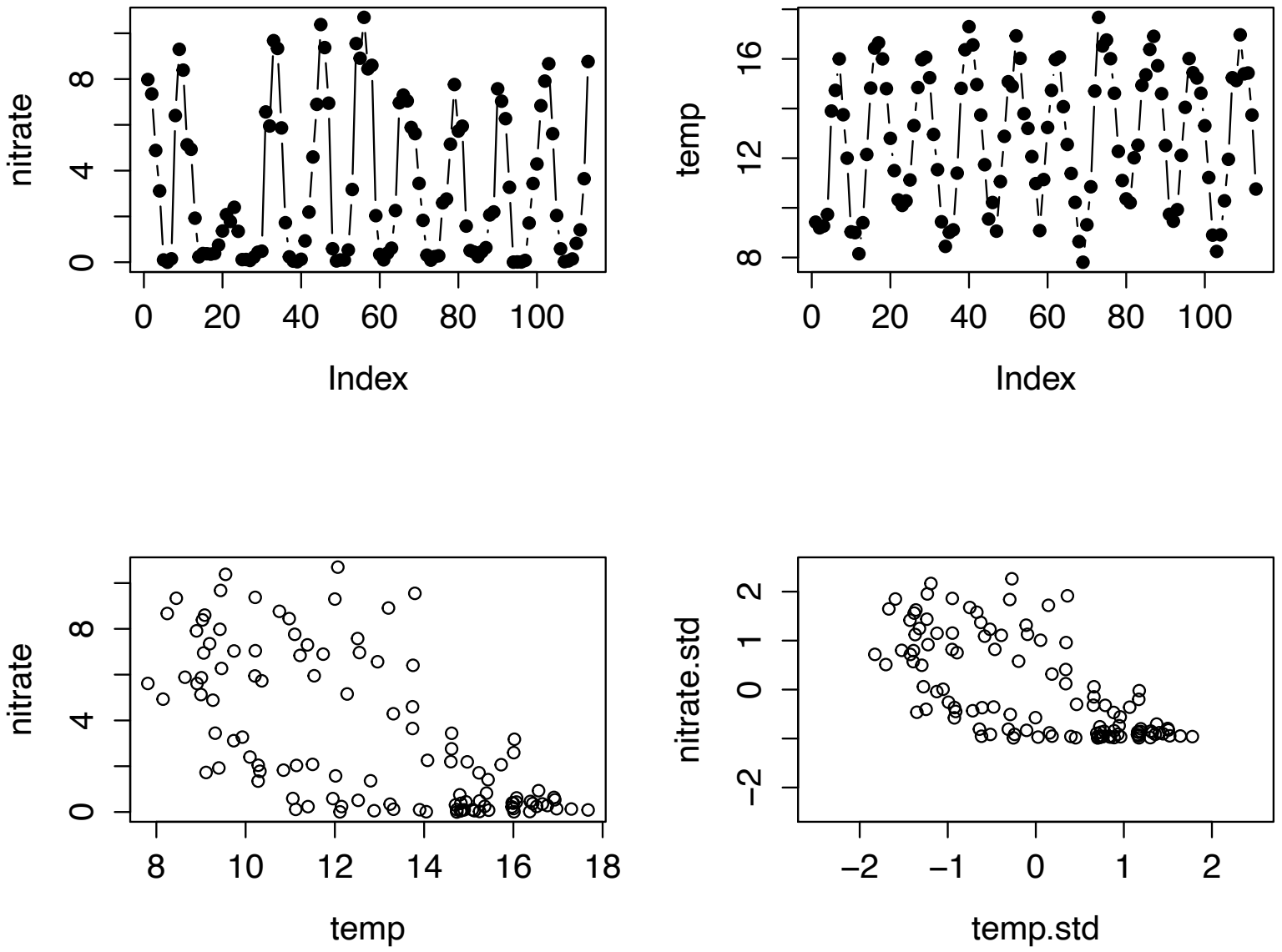
Many environmental predictors, many species' abundances, many traits measured on individuals

- What are the main patterns of covariation?
- Are there dominant axes of variation? Can we visualize multivariate patterns using just 1-3 dimensions?
- Do samples (e.g. communities) tend to form natural groups?
- How well is multivariate variation explained by predictors?

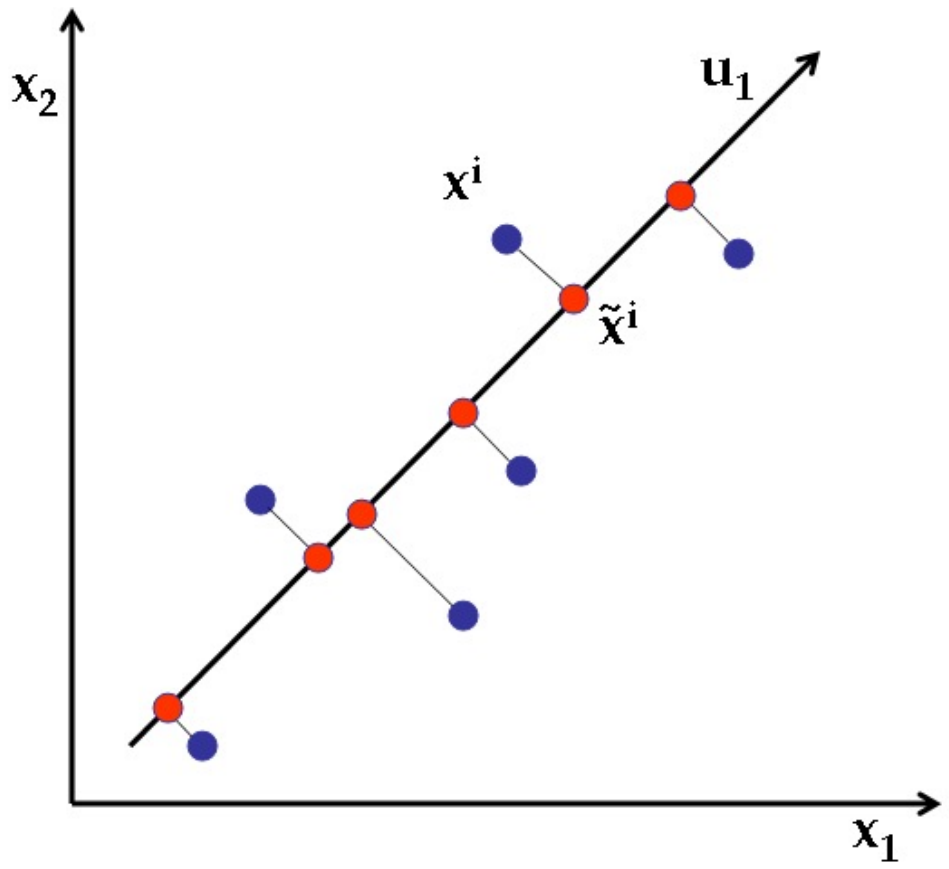
Principal components analysis

Useful for many things (but not all)

Take many variables, represent variation among them using a small number of orthogonal axes



What is the dominant axis of variation among nitrate and temp?

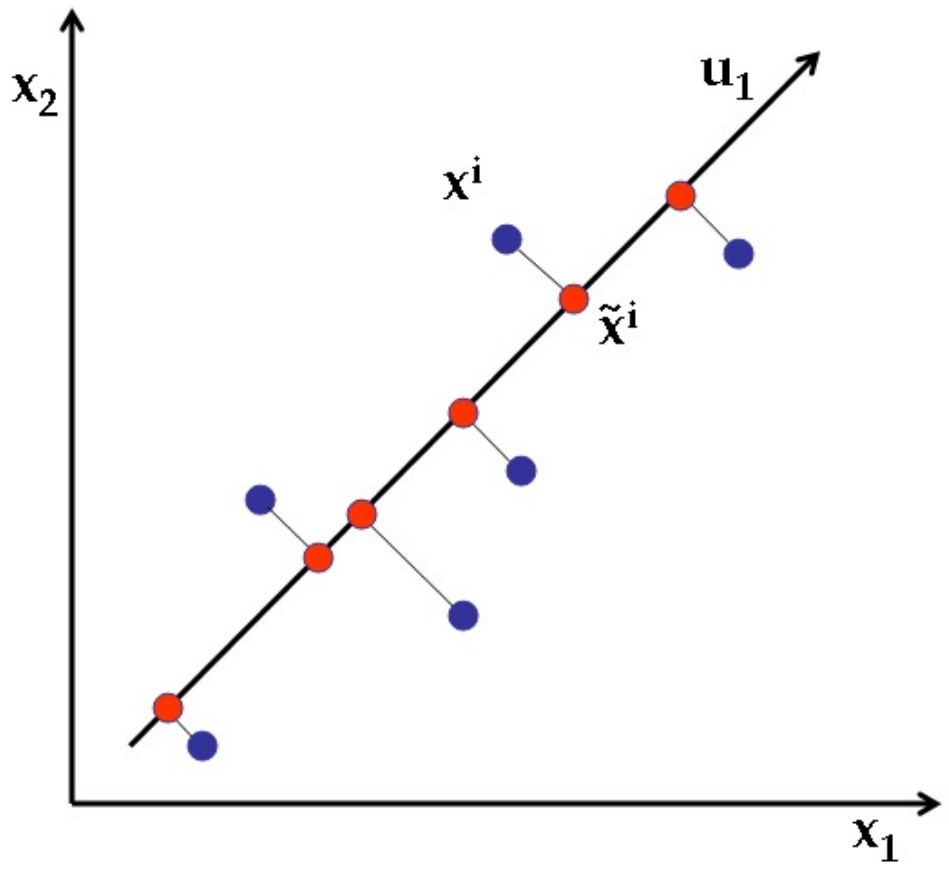


u_1 = first principal component

Maximizes variance along the axis

Minimizes squared distance perpendicular to the axis

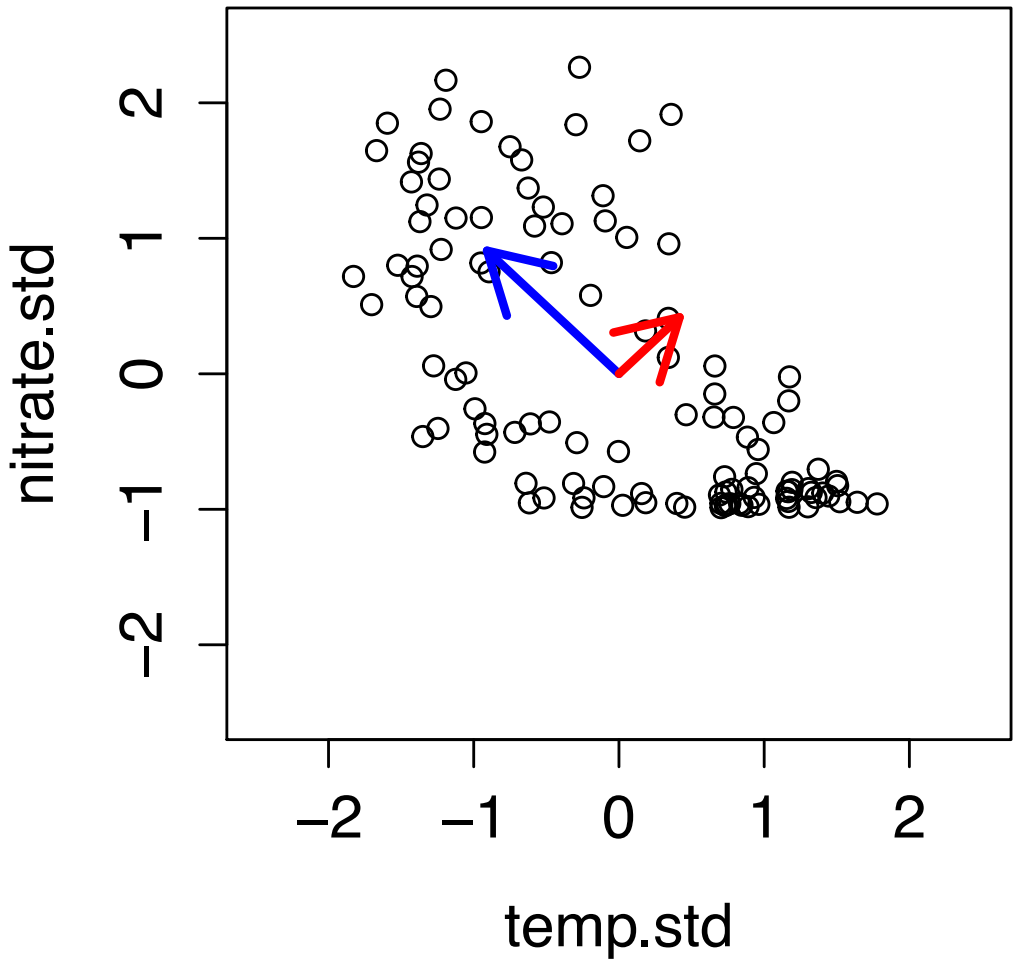
How is this different from a linear regression?



Step 2: find an orthogonal axis that accounts for the most remaining variation

But there are only two dimensions here

We're looking for orthogonal axes because then they'll be 'independent' in our interpretation


nitrate.std
2
1
0
−1
−2
−2
−1
0
1
2
temp.std

```r
enviro.small = subset(enviro, select = c("temp", "nitrate"))
enviro.small = na.omit(enviro.small)

pca = princomp(enviro.small, cor = TRUE)
summary(pca)

## Importance of components:
##                           Comp.1 Comp.2
## Standard deviation     1.2851 0.5905
## Proportion of Variance 0.8257 0.1743
## Cumulative Proportion  0.8257 1.0000
```

Often we want to use “normalized” variables

Same variance = same importance in determining the axes of variation

First PC: 83%

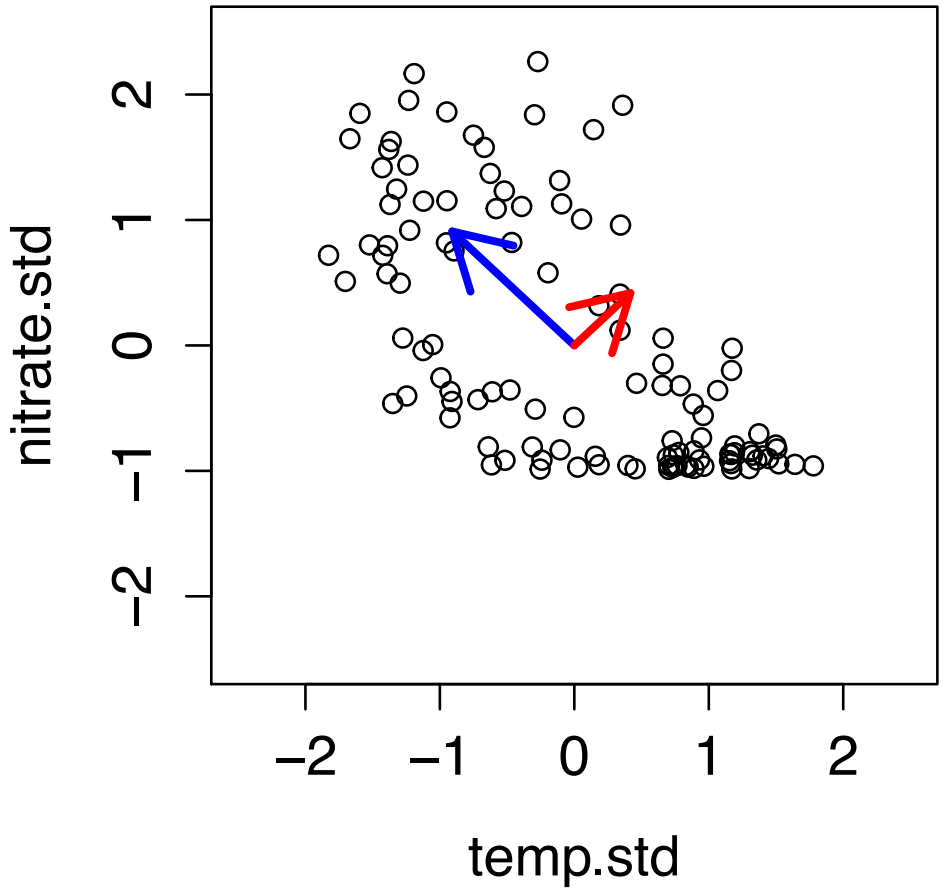
83% of the variation in these two variables can be explained by a single axis of variation

```
loadings(pca)
```

```
##
## Loadings:
##          Comp.1 Comp.2
## temp     -0.707 -0.707
## nitrate   0.707 -0.707
```

Loadings are **vectors** that tell us the **direction** of the principal components

- Note that the vector can also go in the opposite direction and it's still the same principal component



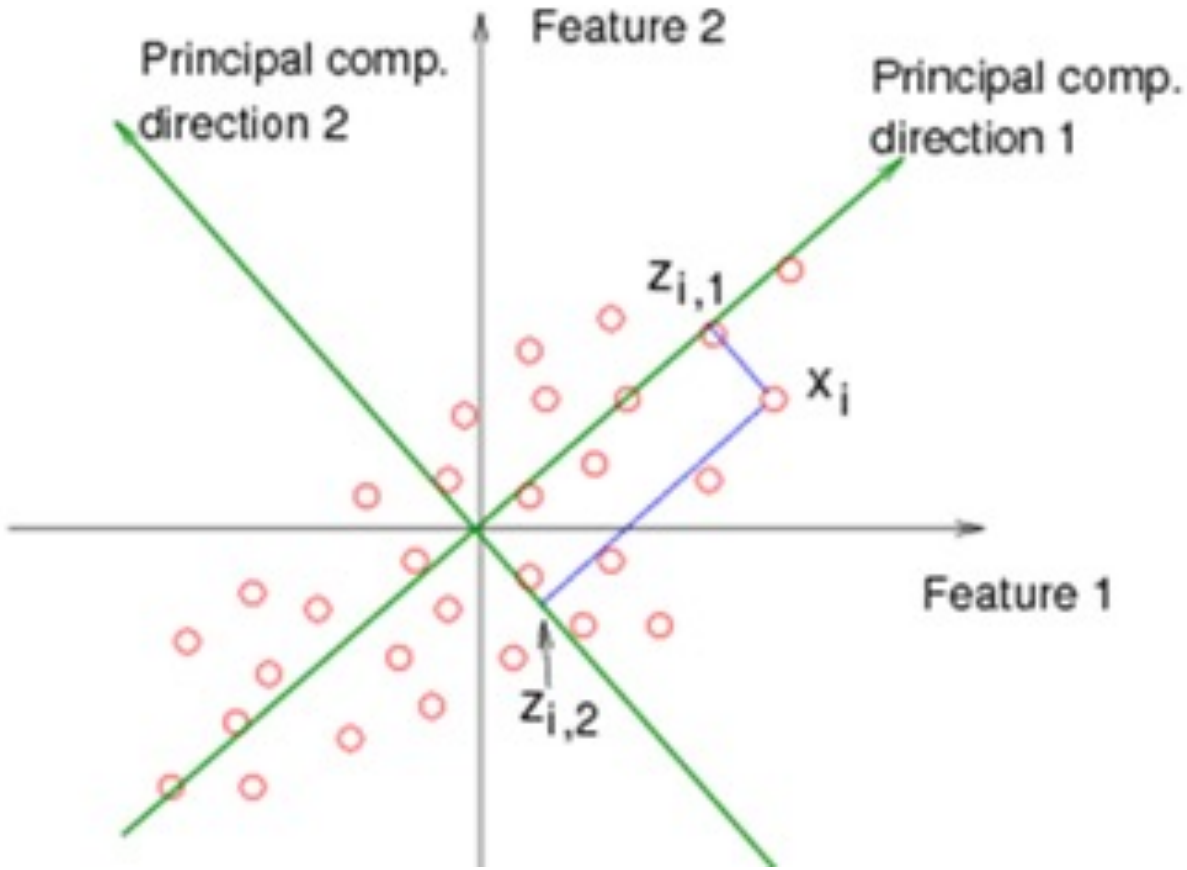
```
pca$scores
```

```
##       Comp.1     Comp.2
## 1    1.89986 -0.1419024
## 2    1.82370  0.0538916
## 3    1.27227  0.5672473
## 4    0.76945  0.8277660
## 6   -0.96298  0.3968138
## 7   -1.20091  0.2020583
## 9   -1.49737 -0.1591524
## 10   0.43716 -0.9254459
```

...

The PCs are a new **coordinate system**

The **scores** tell us where the data are located in these coordinates



The PCs are a new **coordinate system**

A **rotation** of the original

The **scores** tell us where the data are located in these coordinates

Eigenanalysis: some linear algebra that comes up a lot in multivariate stats

For PCA, we can do the math using the variance-covariance matrix for the data

```
cov(enviro.small)
```

```
##             temp nitrate
## temp     7.485  -5.864
## nitrate -5.864  10.829
```

But for normalized variables, actually the correlation matrix

```
cor(enviro.small)
```

```
##              temp nitrate
## temp     1.0000 -0.6514
## nitrate -0.6514  1.0000
```

Eigenanalysis: some linear algebra that comes up a lot in multivariate stats

We can decompose the correlation matrix into **eigenvectors** and **eigenvalues**

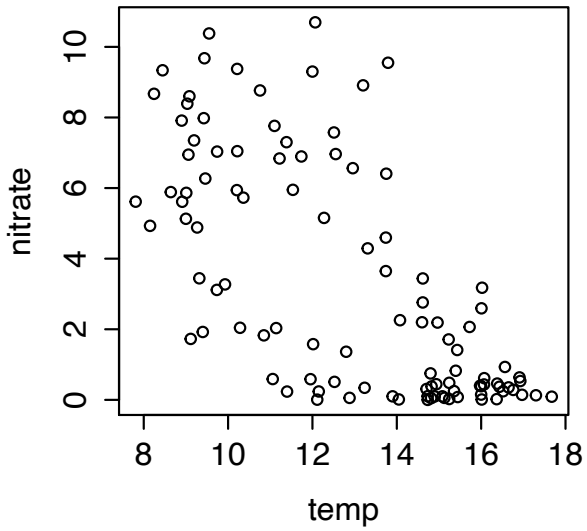
```
eigen(cor(enviro.small))

## $values
## [1] 1.6514 0.3486
##
## $vectors
##         [,1]    [,2]
## [1,] -0.7071 -0.7071
## [2,]  0.7071 -0.7071
```

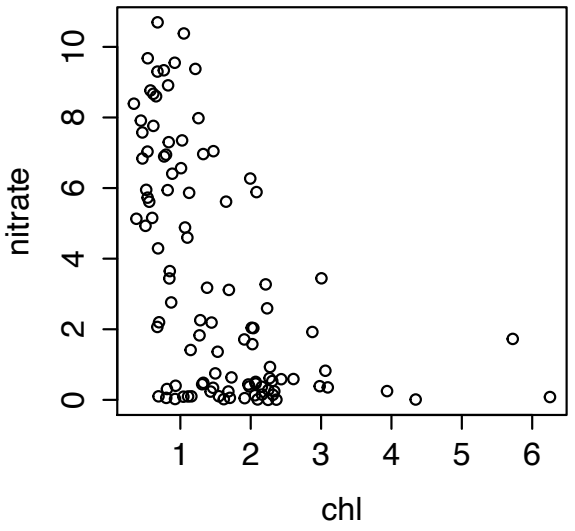
Can think of these as atomic elements from which you can reconstruct a square matrix

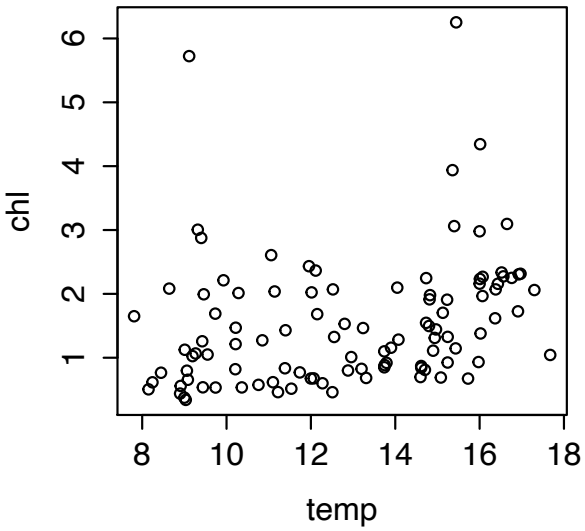
Eigenvectors are the principal components

Eigenvalues are the proportion of variance accounted for by each eigenvector

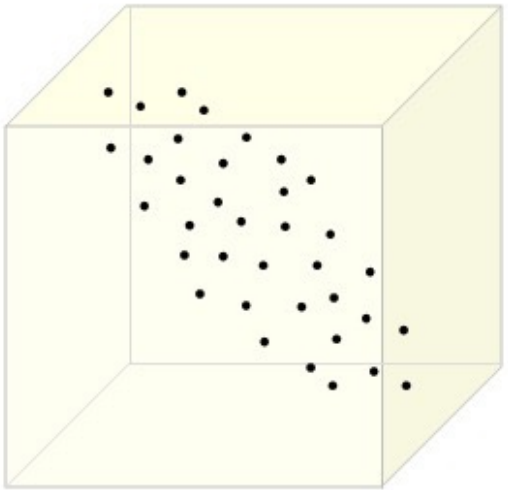
1.65/(1.65 + 0.349) = 0.83

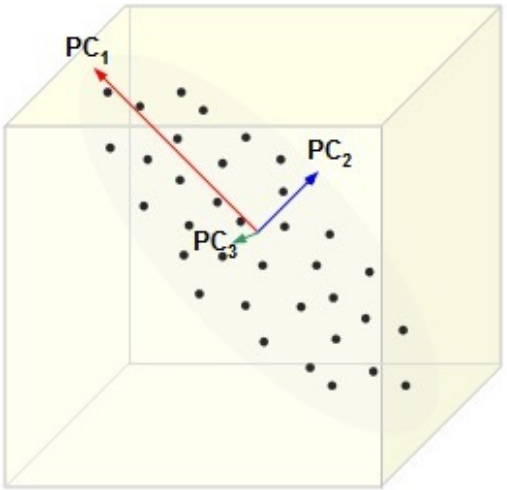


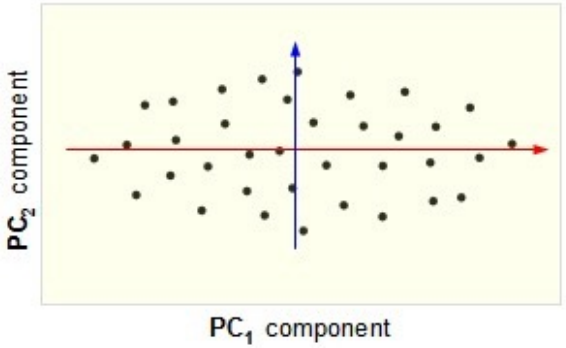




Now PCA becomes more useful: how do these three variables *jointly* vary?



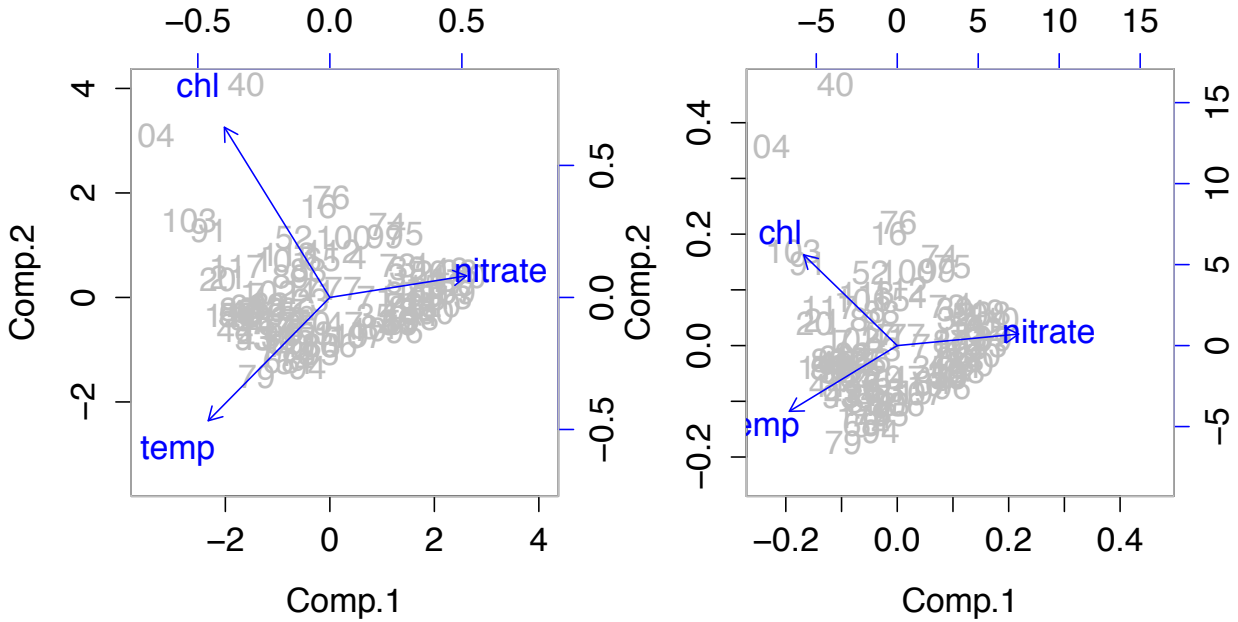




```r
enviro.small = subset(enviro, select = c("nitrate", "temp", "chl"))
pca = princomp(enviro.small, cor = TRUE)

summary(pca)

## Importance of components:
##                          Comp.1 Comp.2  Comp.3
## Standard deviation     1.4187 0.8527 0.51010
## Proportion of Variance 0.6709 0.2423 0.08674
## Cumulative Proportion  0.6709 0.9133 1.00000

loadings(pca)
## Loadings:
##         Comp.1 Comp.2 Comp.3
## nitrate  0.648  0.102 -0.755
## temp    -0.576 -0.584 -0.573
## chl     -0.499  0.806 -0.319
```

- First axis pretty important – explains 67% of the variation in three variables
- nitrate goes up, temp and chl go down

```
par(mfrow = c(1,2))
biplot(pca, pch = 21, col = c('grey', 'blue'), scale = 1)
biplot(pca, pch = 21, col = c('grey', 'blue'), scale = 0)
```



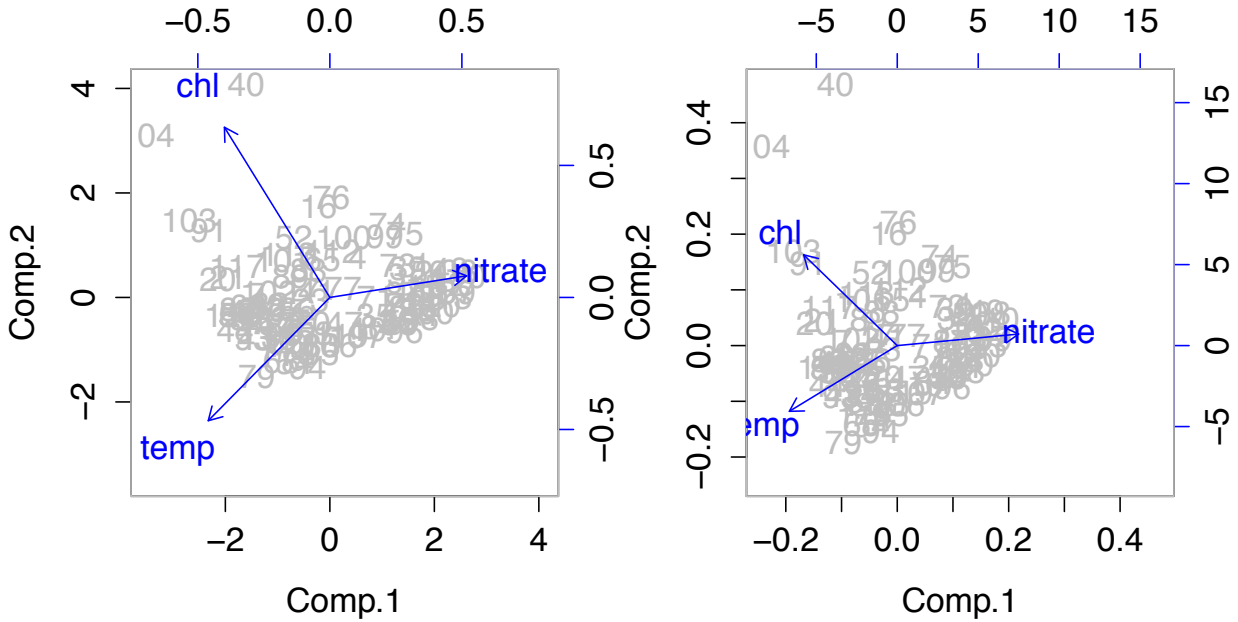
Biplot: visualizing two things at once

1) How the variables correlate with the PCs
2) How the observations are distributed in PC space

Scale = 1. **Correlation biplot**. Optimized for the vectors. Angle approximates correlation.

Scale = 0. **Distance biplot**. Optimized for the observation. Distance approximates euclidean distance.

```r
par(mfrow = c(1,2))
biplot(pca, pch = 21, col = c('grey', 'blue'), scale = 1)
biplot(pca, pch = 21, col = c('grey', 'blue'), scale = 0)
```



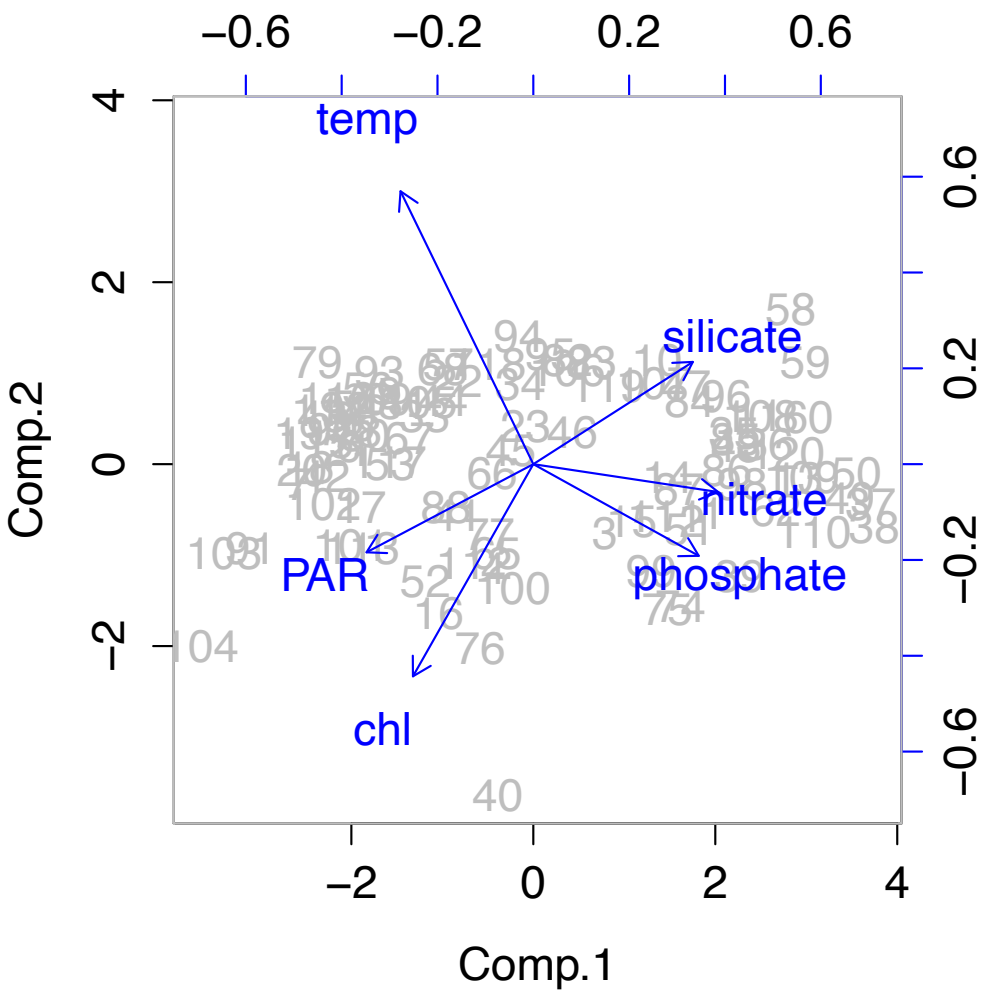
Scale = 1. **Correlation biplot**. Optimized for the vectors. Angle approximates correlation.

Scale = 0. **Distance biplot**. Optimized for the observation. Distance approximates euclidean distance.

Interpretation?

```
enviro.small = subset(enviro, select = c("nitrate", "temp", "chl", "phosphate"
 "silicate", "PAR"))
enviro.small = na.omit(enviro.small)

pca = princomp(enviro.small, cor = TRUE)

summary(pca)

## Importance of components:
##                          Comp.1 Comp.2 Comp.3  Comp.4  Comp.5  Comp.6
## Standard deviation     1.9677 0.8939 0.8105 0.57201 0.50283 0.30377
## Proportion of Variance 0.6453 0.1332 0.1095 0.05453 0.04214 0.01538
## Cumulative Proportion  0.6453 0.7785 0.8879 0.94248 0.98462 1.00000

loadings(pca)

##
## Loadings:
##           Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6
## nitrate    0.482                0.233 -0.275  0.795
## temp      -0.346  0.712 -0.380 -0.167  0.217  0.392
## chl       -0.314 -0.554 -0.739  0.203
## phosphate  0.432 -0.239 -0.154 -0.673  0.522
## silicate   0.416  0.267 -0.500 -0.205 -0.569 -0.375
## PAR       -0.435 -0.230  0.188 -0.618 -0.530  0.245
##
```

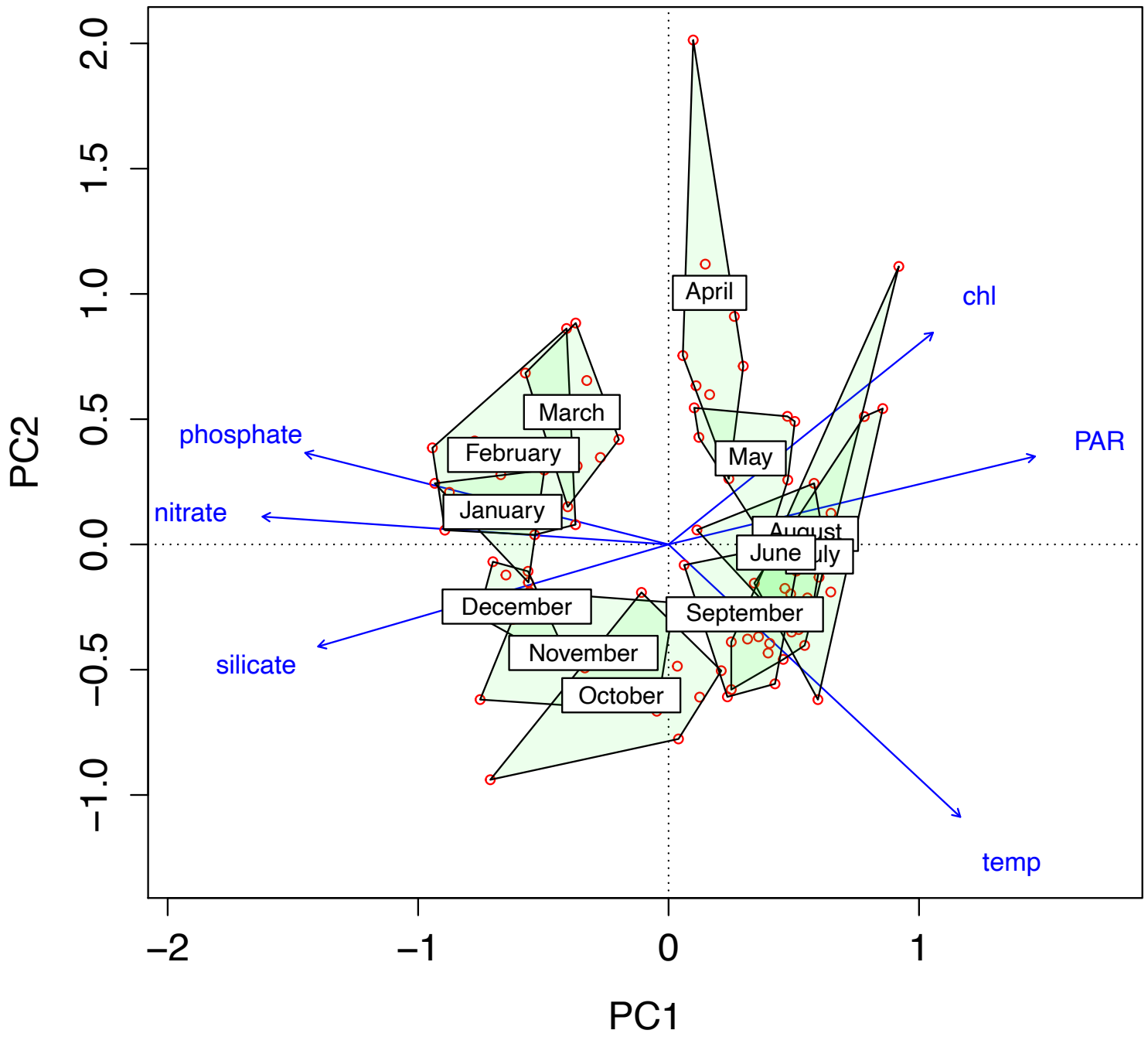
- Six variables, PC1 still explaining about two-thirds
- 2 and 3 smaller, but still could be interesting
- How many axes? Depends on your goal, ability to interpret the loadings



- PC1 looks like a general index of stratification
- What about PC2? Try plotting by month.

```
pca = rda(enviro.small, scale = TRUE)
biplot(pca, col = c('red', 'blue'), type = c("text", "points"), cex = 2)
ordihull(pca, month.use, label = TRUE, col = 'green', border = 'black', alpha
= 20, cex = 0.6, draw = 'polygon')
```

- Refit using a PCA function in vegan - rda
- Ordihull: draws a convex hull around points, based on a grouping variable


PC2
PC1
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−2
−1
0
1
April
March
February
January
May
June
December
September
November
October
chl
PAR
temp
phosphate
nitrate
silicate

Another PCA example: trait covariation across individuals

- Chase et al. (2002, PNAS, "Genetic basis for systems of skeletal quantitative traits…")
- Use 330 portuguese water dogs to look for genetic markers of morphological variation

Fig. 1. Comparison of a young (*a*) with an adult (*b*) PWD. (*c*) Comparison of a Greyhound (*Left*) with a Pit Bull (*Right*). The adult PWD was shorn to display body shape.

PC4 are the residuals from the regression of PC1, PC2, and PC3

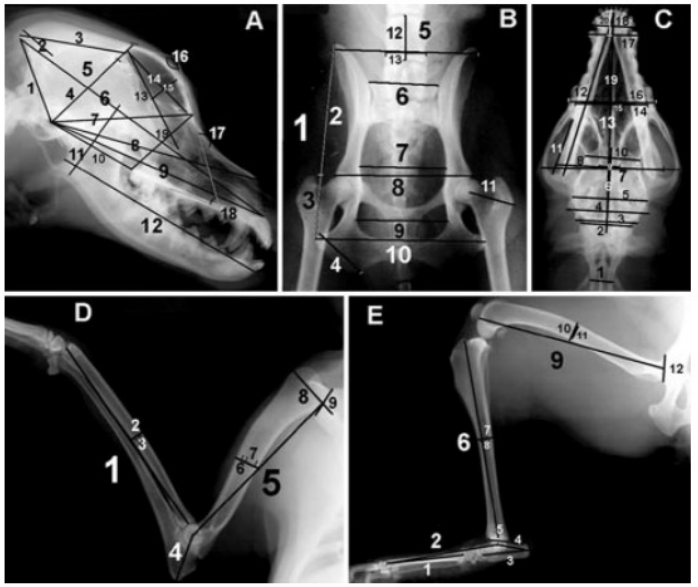


Fig. 2. Five x-ray views of a PWD. (*A*) Profile of skull. (*B*) Pelvis. (*C*) Ventral–dorsal view of skull. (*D*) Fore limb. (*E*) Hind limb. Trait measurements are numbered in each view. Trait numbers are referenced in Table 2.

- PC1 = size; PC3 = long limbs, long skull vs. wide/tall cranium
- Use PC scores for the 330 dogs to correlate with genetic markers
- Can use PC scores as data, e.g. 'stratification index' predictor

What PCA is and is not

PCA is not a model – more like a complex data transformation

No probability distribution, no likelihood, no hypothesis tests

No ‘assumptions’

Still, works better or worse in different contexts:

- Quantifies linear correlations
- Sensitive to outliers, because based on squared distances
- Transformation can help with both of these

Ordination

PCA for English Channel:

1) How the variables are correlated
2) Mapping out the samples in PC space
- Where those samples fall on environmental axes, how that relates to season, etc.

How about the same kind of mapping for **community composition**?

```
species.data[1,1:8]

##   Ceratium.fusus Ceratium.lineatum Nitzschia.closterium
## 1              0           0.02002                  0.2
##   Nitzschia.delicatissima Nitzschia.panduriformis Chaetoceros.danicus
## 1                   1.111                       0                   0
##   Chaetoceros.decipiens Roperia.tesselata
## 1                     0           0.03966
```

Ordination

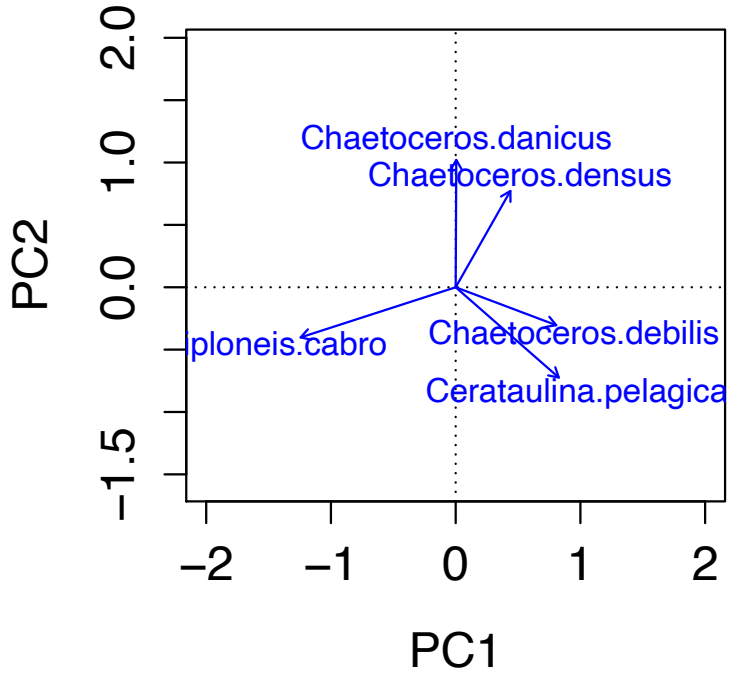
What are the major axes of variation in composition? Dimensionality?

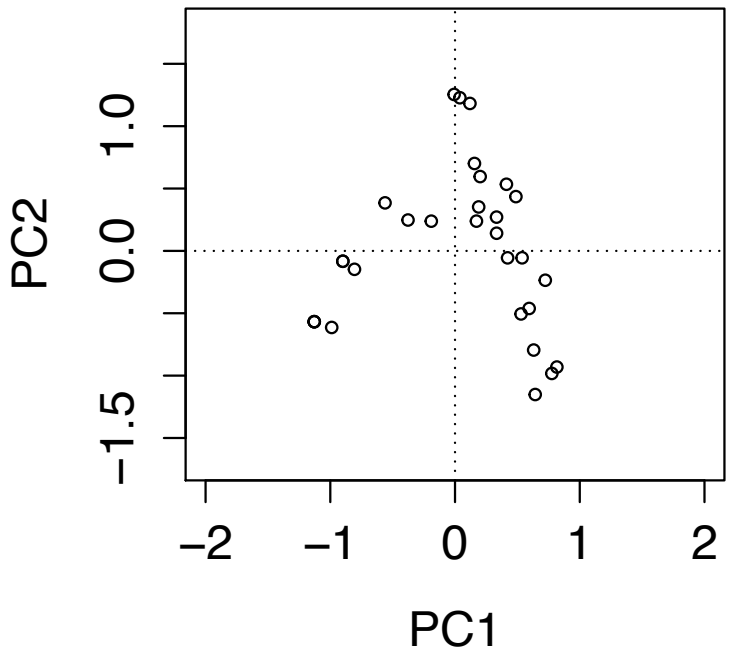
This is **ordination**: ordering community samples along gradients of composition

Can also use this to visualize if composition correlates with the environment, or with experimental treatments, etc.

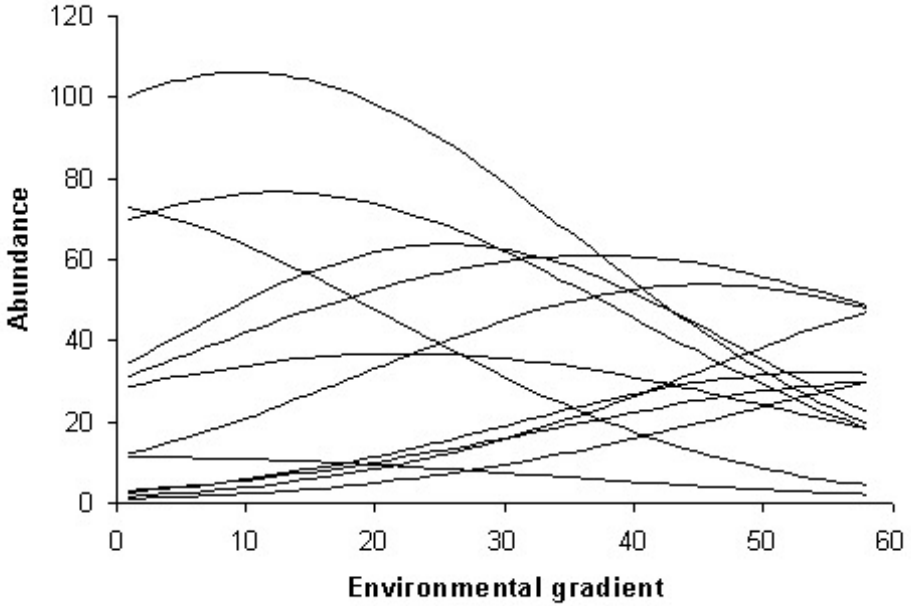
```
species.data[1,1:8]
```

```
##   Ceratium.fusus Ceratium.lineatum Nitzschia.closterium
## 1              0           0.02002                  0.2
##   Nitzschia.delicatissima Nitzschia.panduriformis Chaetoceros.danicus
## 1                   1.111                       0                   0
##   Chaetoceros.decipiens Roperia.tesselata
## 1                     0           0.03966
```



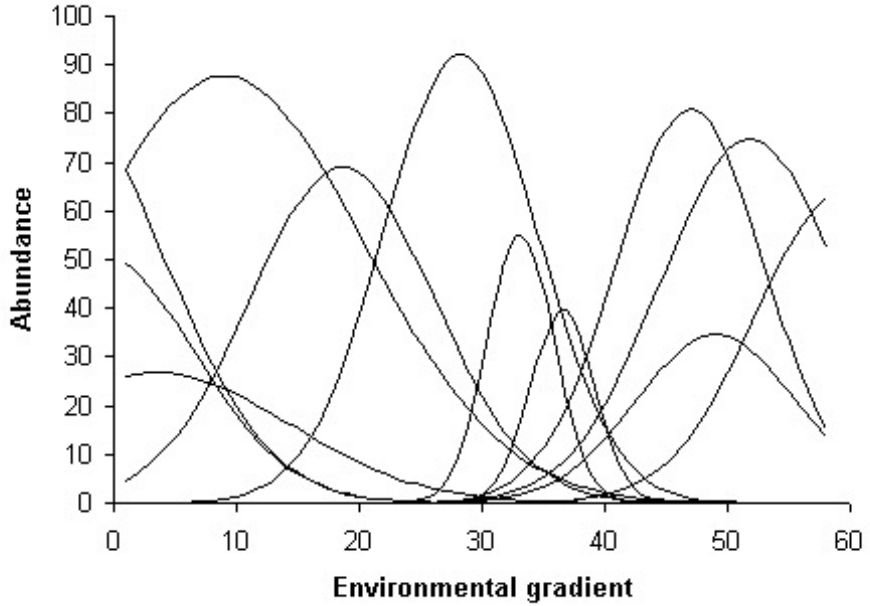


- PC1 34%, PC2 25%
- We can see how species tend to correlate along major axes
- We can map/ordinate samples in ‘species space’, for further analysis
- Problem: PCA is a linear method



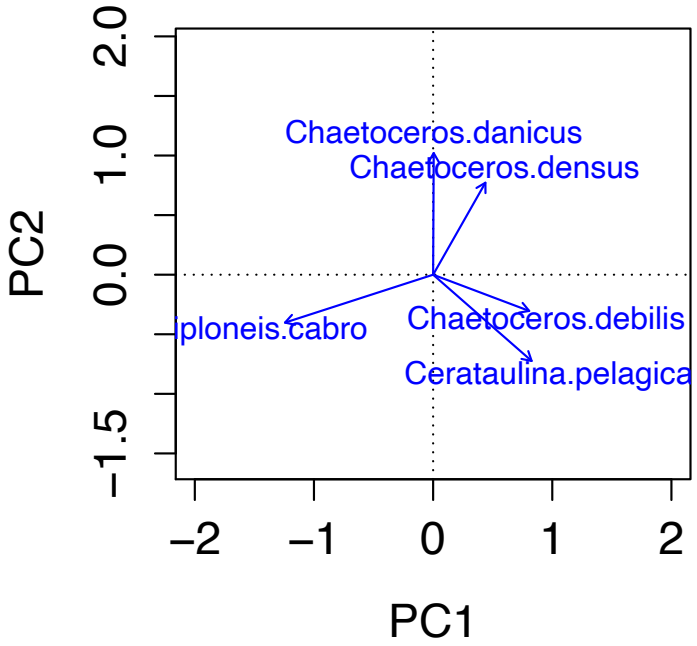
Low turnover / beta diversity

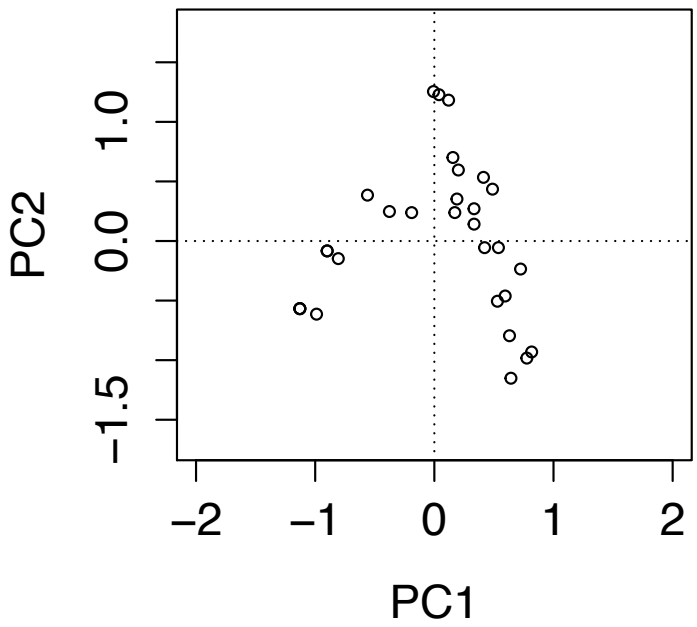
PCA might work OK

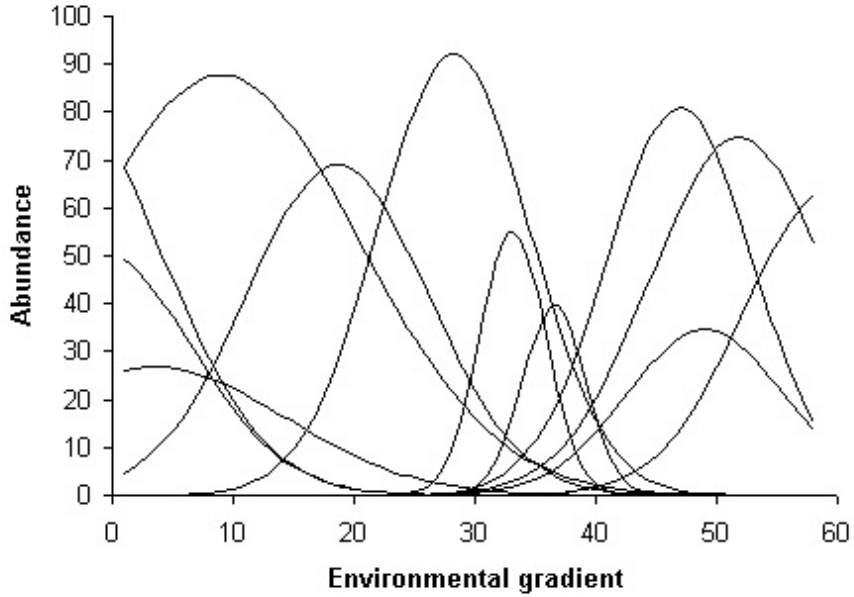


High turnover / beta diversity

PCA will do a poor job of reconstructing this



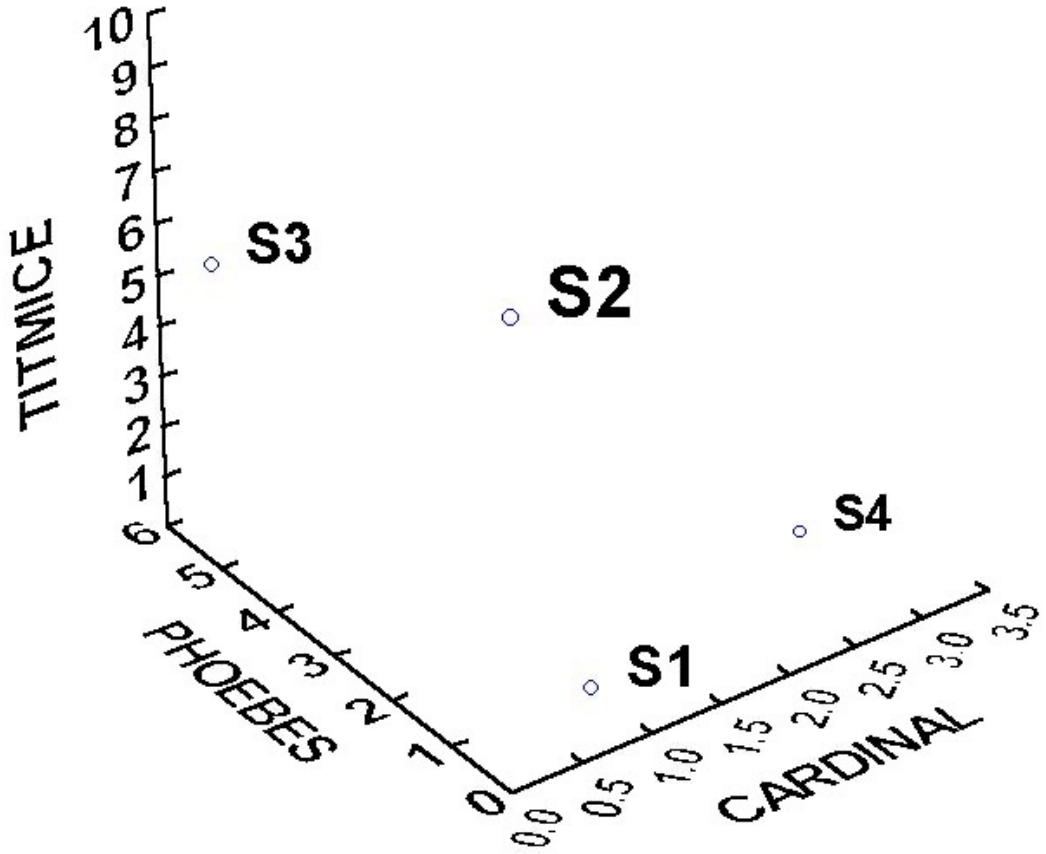




The ‘arch’ or ‘horseshoe’ effect

Samples at opposite ends of a gradient look somewhat similar, because intermediate species are absent

Ordination with PCA: **dissimilarity** between samples is **euclidean distance** in species space



PC1 is the axis along which total dissimilarity beween samples is greatest

Improve upon PCA by defining dissimilarity in a better way, and use methods that can handle it

- Note issue of double zeros, total abundance not accounted for

Community similarity / dissimilarity metrics

There are **many**

Jaccard index, for presence-absence (binary) data

$$S = \frac{J}{A + B + J}$$

J is the number of species present at both sites (or at both times)

A is the number of species present only at site A; ***B*** is the number of species only at site B

S = similarity, **D** = 1 – S = dissimilarity

Note this ignores species that are **absent from both sites**

The **'double zero'** problem: a species could be absent from two sites for two different reasons

Most community similarity metrics treat **presences as more informative than absences**

Community similarity / dissimilarity metrics

There are **many**

Bray-Curtis is a similar index, for abundance data

$$D_{jk} = \frac{\sum_i |X_{ij} - X_{ik}|}{\sum_i (X_{ij} + X_{ik})}$$

For abundance, you usually need to **transform** as well

To make species of equal importance, regardless of absolute abundance

E.g. the examples in vegan will often:
1) square root
2) Wisconsin double standardization: divide each species by its max; make each sample have the same total. This makes each samples and each species equally important to the analysis.

Also, dropping rare species entirely is good, they just add noise

Clearly there are a lot of judgment calls, because we aren't modeling the raw data