Lecture 25. PCA; Ordination I.

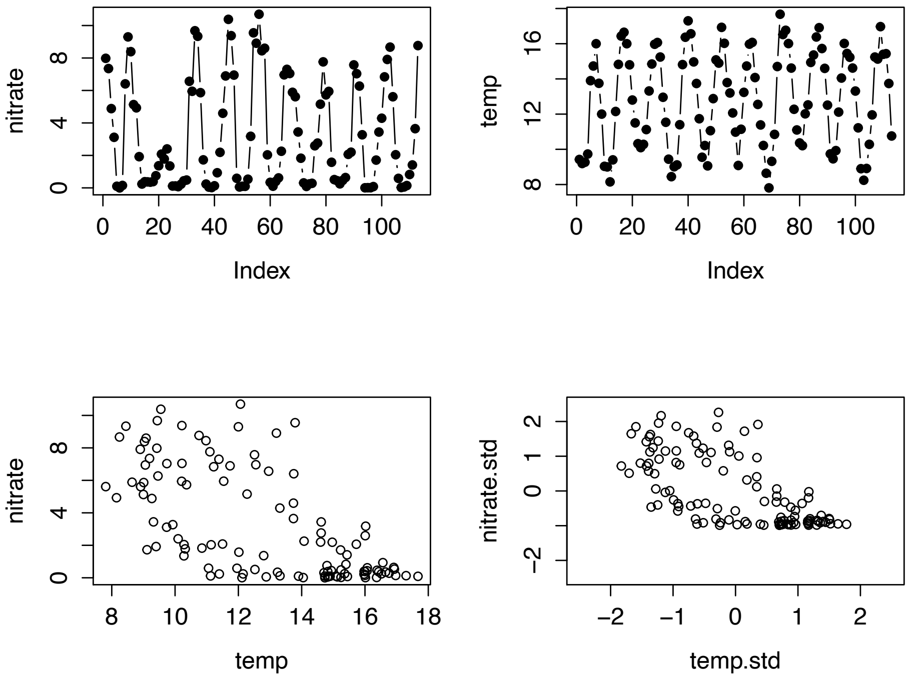
The next few lectures will be about visualizing and analyzing complex multivariate datasets. Of course we've been dealing with complex multivariate datasets throughout the course: whenever we make a model that has more than one predictor, we have to start thinking about patterns of correlation among the predictors, in addition to the relationship between the predictors and the response variable. But fundamentally these analyses had a univariate focus, because there was some single response variable we were modeling.

Now we're going to focus on methods for more truly multivariate problems. We'll try to answer the following kinds of questions, which are (mostly) hard to answer with the univariate methods we've already learned:

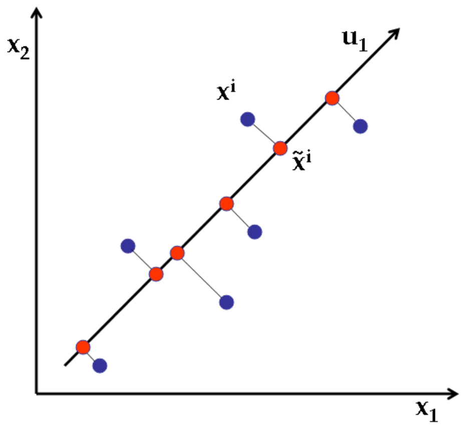
- I have a number of variables that might be related to each other (a set of environmental variables; species abundances at many sites). What are the dominant patterns of covariation, when we consider all the variables together?
- Are there dominant axes, representing (unobserved) drivers that structure the data? Can multivariate data be represented in terms of a smaller number of dimensions?
- Do samples (communities, environmental conditions, a species' traits) tend to form natural groups?
- Are there patterns in multivariate data not revealed by univariate analysis?
- How well is variation in a multivariate response explained by one or more predictors?

Principal components analysis

We'll start with principal components analysis (PCA). This is a simple and widely used technique, and understanding it well will also help us understand more complex methods that are applied when PCA is less helpful. PCA can be used for various purposes, which we will discuss, but a common use is to understand the important patterns of covariation among a number of variables. This is accomplished by finding a small number of orthogonal axes that explain as much variation in the data as possible. This is easier to explain with graphs. I'll use some data from the L4 time series in the English Channel. To start I'll just plot nitrate and temperature over time, and a scatterplot:



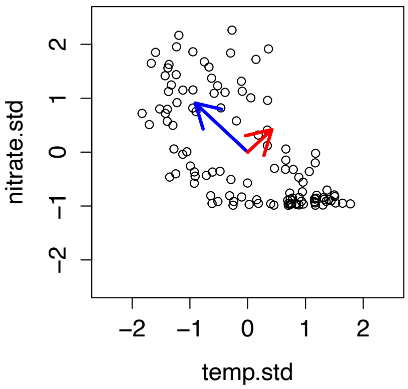
On the top, each point is the data from one month. Both variables fluctuate fairly regularly due to seasonal variation (more on that later). The plot on the bottom left is the raw data. The plot on the bottom right has both variables centered and divided by their standard deviations (this is how I will actually analyze the data, more on that below). These variables are negatively correlated. With a principal components analysis, we can ask “what is the dominant axis of variation among these two variables?”. PCA answers this question by finding the the line through the data that explains the most variation in the data. For example, in this schematic:



the blue points are the raw data for x_1 and x_2. We can draw a new line u_1 through the data, and project each point onto that line (the red points). Now we’re treating

u_1 as a new axis, i.e. as a new coordinate system to map out the points. PCA finds the axis u_1 along which the points have the most variance. This is equivalent to finding the line that minimizes the (squared) distance between the points and the line. This axis is called the first *principal component*. The next step is to find a second axis, *orthogonal* (i.e. perpendicular) to the first, that explains as much of the *residual* variation as possible. This is the second principal component. But in this simple example we only have two dimensions, and so there is only one possible second principal component, which is the axis perpendicular to the first principal component. We find orthogonal axes because we can interpret them as 'independent' axes of variation, in the sense of not being correlated with each other.

If we do a PCA on the nitrate and temperature data, we'll get axes that look like this:



Where the blue line is the first PC, and the red line is the second PC. Because we only have two dimensions, the blue axis accounts for as much variation as possible, and the red axis accounts for the rest. I've drawn the red arrow shorter simply to indicate that it explains less variation in the data. Let's see how to do this in R. There are a number of different functions one could use, I'll use princomp:

```
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
```

First I made a data frame with only nitrate and temperature as the variables, then I used the princomp() function to perform a PCA. I set cor = TRUE. What does this mean? As I've discussed, PCA is finding axes that account for the maximal amount of variation. Because we are looking at multiple variables, the scale of the variables matters. E.g. if temperature ranged from 0 to 30, but nitrate ranged from 0.1 to 1, then there is much more variance in the temperature data, and it would dominate the results. Because of this issue of different scales for different variables, it is typically more useful to first standardize all variables so that they have a variance of 1 (this can be achieved by dividing each variable by its standard deviation). This makes it so that each variable has the same total variance, and therefore will be of equal importance in determining the PC axes. This is done automatically with cor = TRUE. summary() shows how much variation is explained by the two principal components. The first component explains 83% of the variation, i.e. nearly all the variation, while the second axis gets the remaining 17%. What are these axes? We can see with loadings():

```
loadings(pca)
```

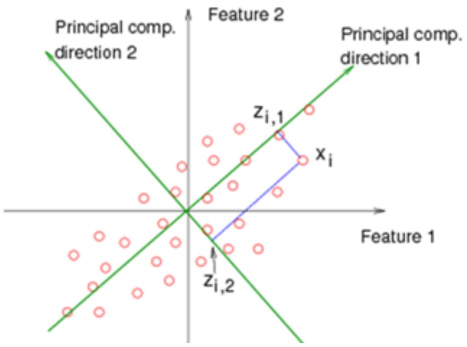
```
##
## Loadings:
##         Comp.1 Comp.2
## temp    -0.707 -0.707
## nitrate  0.707 -0.707
```

The numbers returned for the two components are vectors. This means the first component is in the direction between (0,0) and (-0.707, 0.707), and the second component is in the direction between (0,0) and (0.707, 0.707). These are the numbers I used to plot the vectors on the plot above. The other important thing we can extract is the coordinates of the data, using the PC axes as a coordinate system. These are referred to as scores:

```
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
…
```

This is abbreviated; there is a row for every observation in the dataset. This means for the first observation, the scores on the first axis is 1.899, and the score on the second axis is -0.1419. We can visualize how this works with this schematic:



Here the red points are the raw data, and the green lines are the principal components. The observation x_i has a score of $z_{i,1}$ on PC1, and a score of $z_{i,2}$ on PC2.

The math underlying PCA can also be explained in terms of *eigenanalysis*. This is a linear algebra method that we don't have time to fully explain, but it is helpful to be aware of the terminology, because it comes up a lot in multivariate analyses. We start with a variance-covariance matrix for our variables:

```
cov(enviro.small)

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

The variance of temperature is 7.49, the variance for nitrate is 10.8, and the covariance between then is -5.9. Because we're normalizing these variables to have equal variance (and therefore equal weight in the PCA), we're actually going to analyze the correlation matrix:

```
cor(enviro.small)

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

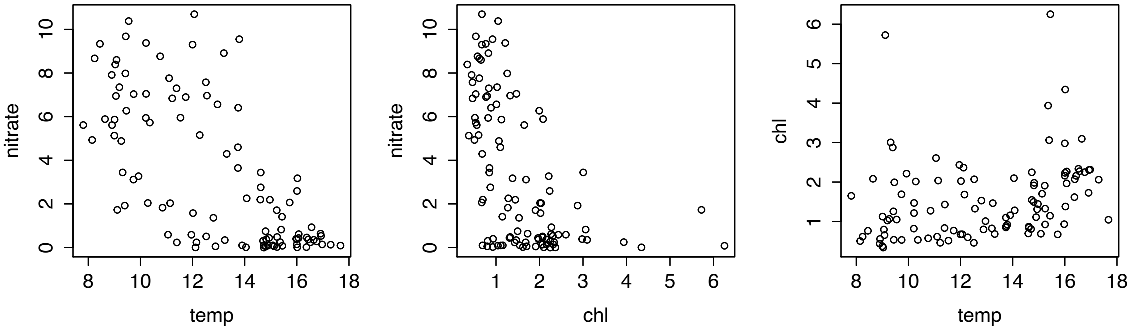
The correlation of temperature with itself is 1; the correlation between temperature and nitrate is -0.65. This matrix can be decomposed into *eigenvectors* and corresponding *eigenvalues*:

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

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

Explaining what eigenvectors and eigenvalues are in the abstract is a bit challenging, but you can think of them as like atomic elements of a square matrix. When you're analyzing a covariance matrix, the eigenvectors are the the principal components, and the eigenvalues tell you the relative amount of variation accounted for by those components. So you can see that the vectors above are the same as what we got for loadings(pca). And to get the %variance explained, we can do 1.65/(1.65 + 0.349) = 0.83. So that's the variance explained by PC1. When I did summary(pca), the row 'Standard deviation' is the square root of the eigenvalues.

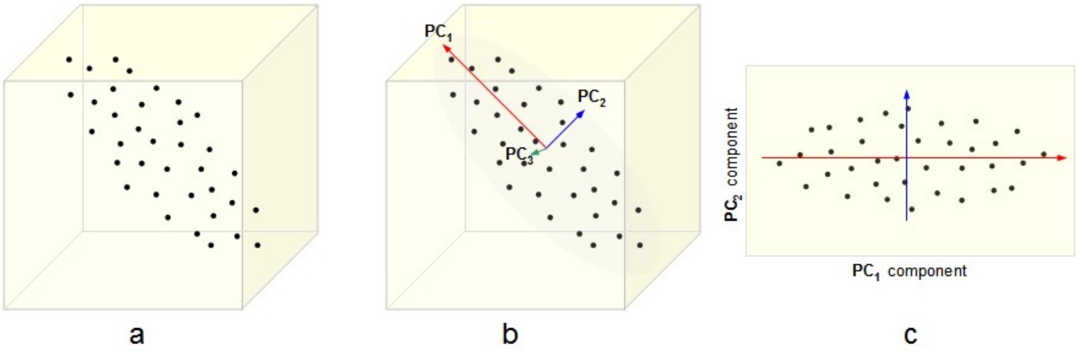
PCA in 3 dimensions

Now we've covered the basics of PCA results: variation explained by the different axes, the vectors, the scores. Before I explain more about using PCA, let's consider 3 variables, because 2 variables isn't really what PCA is useful for.



Now we're looking at nitrate, temperature, and chlorophyll-a. We already saw that nitrate and temperature are negatively correlated. It also looks like nitrate and chlorophyll are negatively correlated, though less strongly. Also temperature and chlorophyll are somewhat positively correlated. Looking at pairwise correlations is important, but we may want to think about how all three variables are related simultaneously, and that's what PCA is useful for.

In 3 dimensions the protocol is the same, but a bit harder to visualize. Again the first PC will be the axis through the data that accounts for the most possible variation; the second PC will be the axis orthogonal to the first that account for the most leftover variation; and the third PC will be the axis orthogonal to the first two. We can visualize it with this schematic:



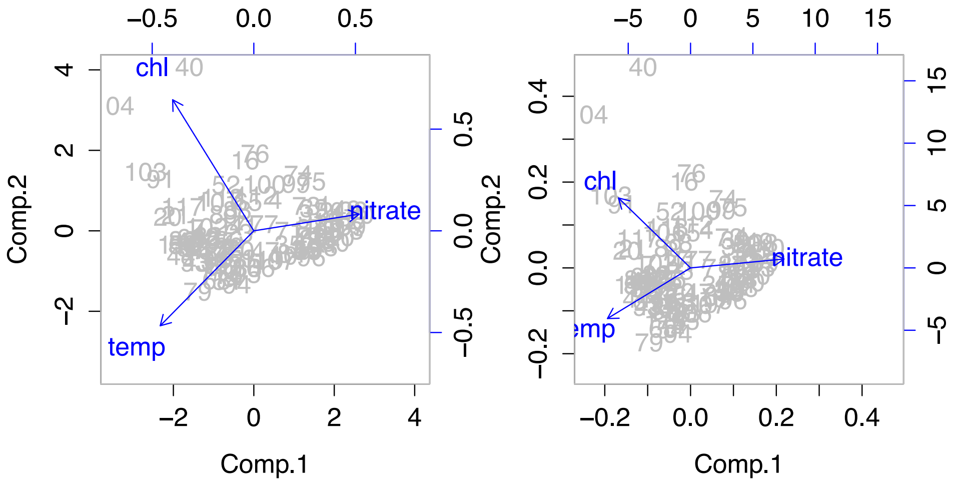
Here we have a cloud of points, roughly ellipsoid in shape, in three dimensions. There is fairly strong correlation among all three variables, which is captured by PC1. There is moderate additional variation orthogonal to this captured by PC2, while there is little variation left for PC3. The number of principal components will always equal the number of variables in the dataset; you can think of PCA as a rotation from the original orthogonal axes to the orthogonal PC axes. The plot on the right shows how the first two PC axes can be used to represent the data, by plotting them using their scores on PC1 and PC2. Geometrically, the plot on the right shows the raw data projected onto the plane formed by the first two PCs.

Let's analyze the three variables:

```
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
```

The first component explains 67%, the second 24%, and the third 9%. That means we capture a lot of what's going on with a single axis of variation, and nearly everything with two axes. Along the first component, nitrate increases while temperature and chlorophyll decrease. Along the second component, temperature and chlorophyll are negative correlated. These relationships are easier to visualize using a *biplot*.

```
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)
```



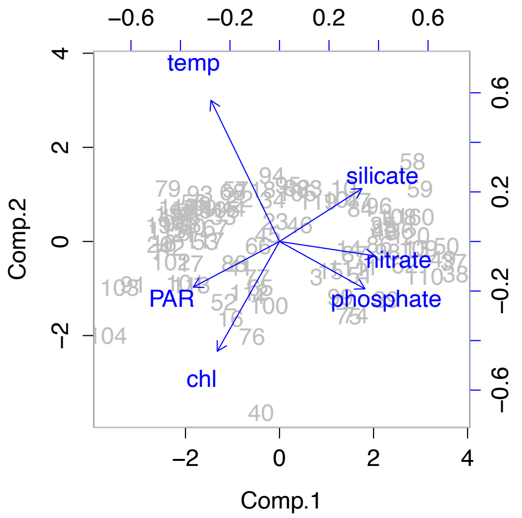
The idea of the biplot is to summarize PCA results by plotting two things at once, using two PC axes: 1) the scores of the observations, 2) the relationship between the original variables and the PC axes. On this plot, the x-axis is the first principal components, and the y-axis is the second principal component. So now we're visualizing the data on the plane formed by these two axes. The reason I've plotted two versions is because the axes can be scaled in different ways, which has some effect on how the plot is interpreted. The plot on the left (scale = 0) is called a *correlation biplot*, and it is scaled so that the angle between the vectors approximates the correlation between those vectors, and the angle between a vector and an axis approximates the correlation between that vector and that PC component. So, for example, nitrate is highly correlated with PC1 (the x-axis), but hardly correlated with PC2, and is also negatively correlated with temperature and chlorophyll. In contrast temperature is correlated with both PC axes. At this point we could start interpreting *why* these variables show this pattern of multivariate covariation (it has to do with seasonality and stratification), but we're going to look at this in more detail shortly.

The biplot on the right (scale = 1) is called a *distance biplot*, and is scaled so that the distance between the observation scores (the gray numbers, which label the points by row numbers) approximates the euclidean distance between the observations in the raw data. As you can see, the plots overall look similar, and give you similar information, but if you're more focused on the *variables* you should plot the correlation biplot, while if you're more focused on the *observations* you should plot the distance biplot. They're slightly different because they are scaled to optimize different things.

Let's add some more variables into the mix:

```
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
##
```

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

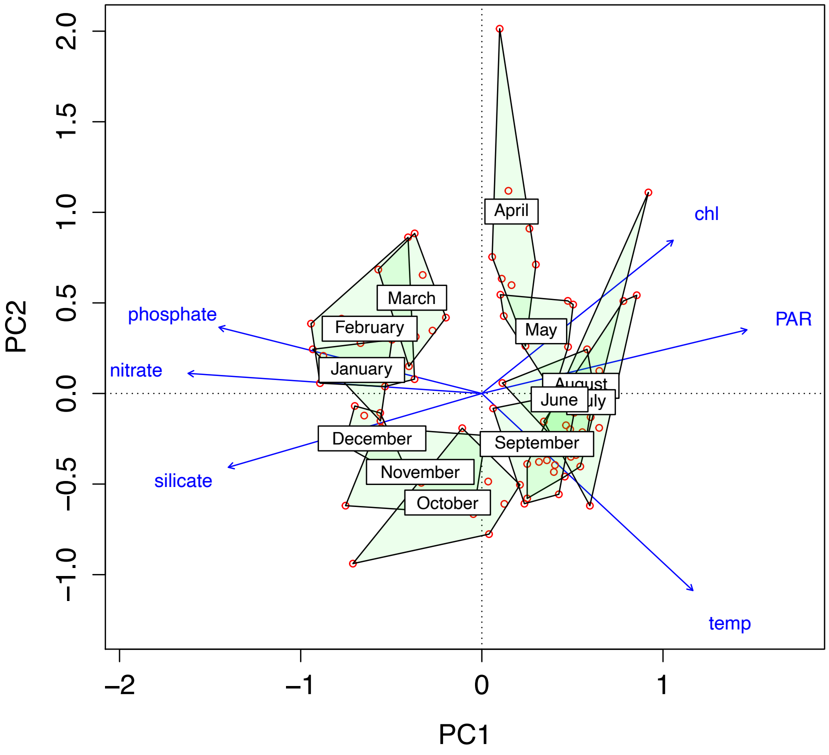


Now we have six variables, and six principal components. The first axis dominates, explaining 65% of the variation. None of the remaining axes seem of similar importance; the second explains 13%, and the third 11%. Sometimes the question comes up, "How many axes should I use / look at / think about?". This definitely depends on your goals, because PCA has many uses. For thinking about environmental variation, I often find that only the first 2 or 3 axes are easily interpretable, in terms of understanding what they represent. But in some other contexts, such as morphological variation along many trait axes, the first PC may be the most trivial (e.g. representing the effect of body size on all traits), and the smaller PCs will reveal more interesting patterns of trait covariation.

In the current example, we can start to think about what the first two axes mean. The first axis is roughly an axis that separates high nutrient conditions (nitrate, phosphate, silicate all increase to the right) from high light / temperature / chlorophyll (in this case PAR is photosynthetically active radiation at the surface). So this is clearly an axis that reflects variation in stratification, which has a strong seasonal cycle in this temperate coastal ecosystem. During the summer (high light), greater irradiance and less wind leads to stratification, which leads to enhanced phytoplankton growth (higher chl), and nutrient depletion. It makes sense that this axis of seasonal stratification dominates the signal of multivariate environmental variation. Can we understand the second PC, which explains far less variation but still might be telling us something interesting/important? Because there is a seasonal signal to this data, I decided to plot the PC scores in terms of the month of the year when they were taken. Rather than extract the scores and color code them or something, we can use a nice function from the 'vegan' package:

```r
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')
```

I refit the pca using the rda() function from the vegan package, so that I could take advantage of the vegan plotting functions (now we say scale = TRUE instead of cor = TRUE). The function ordihull() uses a grouping factor (here month.use, which is just a factor for the month of each sample) to draw a polygon showing the *convex hull* for each level of the factor. The convex hull is just the smallest polygon that contains all the points for the group.



So now we have a biplot again (note that the orientation of the axes has been flipped, because I used a different PCA function, and the orientation is arbitrary). In addition the points from each month are labeled with a convex hull (there are 10 points per month, because I'm using a 10 year time series). The hulls partially overlap, some months more than others, but we can see a nice pattern of the yearly cycle, going from the less stratified winter conditions on the left to the more stratified summer conditions on the right. In addition, the second PC axis seems to differentiate spring vs. fall. Both spring and fall tend to be intermediate in terms of nutrients, temperature, light, and chlorophyll; but in addition, the spring tends to have more chl and lower temperature than the fall. This makes sense because the spring bloom tends to be larger than the fall bloom in this system, and water temperature lags behind the seasonal irradiance. This plot also reveals why the data has a circular shape, with a gap in the middle.

Another PCA example: trait covariation

Looking at patterns of environmental covariation is a classic use of PCA in biology. Another classic use is looking at patterns of trait covariation across individuals. I won't go through a detailed example, but here is a quick one from a study by Chase et al. (2002, PNAS, "Genetic basis for systems of skeletal quantitative traits…"). The authors were looking for genetic markers of variation in body shape of portuguese water dogs. Using 300 water dogs, they measured a large number of skeletal distances using X-rays:

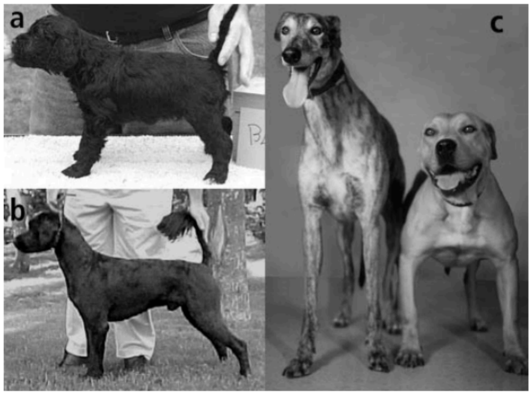


Fig. 1. Comparison of a young (*a*) with an adult (*b*) PWD. (*c*) Comparison of a Greyhound (*Left*) with a Pit Bull (*Right*). The adult PWD was shorn to display body shape.

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

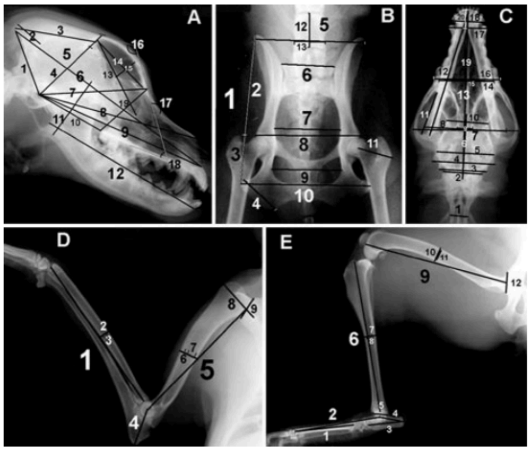


Fig. 2. Five x-ray views of a PWD. (*A*) Profile of skull. (*B*) Pelvis. (*C*) Ventral–dorsal view of skull. (*D*) Fore limb. (*E*) Hind limb. Trait measurements are numbered in each view. Trait numbers are referenced in Table 2.

These distances were put into a PCA to uncover the dominant patterns of mophological variation. PC1 reflected variation in overall size: all skeletal distances had positive loadings on that axis. The other axes reflected variation in shape; for example, PC3 showed negative correlations limb length / skull length and the width/height of the cranium. This is similar to how greyhounds have elongate bodies with small/narrow heads, in comparison to short-limbed, big-headed pit bulls. The authors then used the PC scores to find genetic markers for the trait variation captured by the PC axes. So, roughly, they looked at the score for PC3 for all 330 dogs, and saw whether high/low values of that score were correlated with the presence of certain DNA sequences. This example shows how PCA can be used both to discover patterns of covariation, and also used for subsequent analyses (using PC scores). We can find similar applications when using environmental data. E.g. PC1 in the English Channel example could be used as a composite index of stratification, and the scores along this axis could be used as predictors when modeling the abundance of organisms.

What PCA can and can't do

Now that we have a sense for how PCA works and what it's good for, we should think about limitations and caveats. It's important to note that *PCA is not a model*, at least not in the sense I've been using 'model' in this course. We are not assuming that the data is drawn from some random distribution, with parameters that might be affected by some predictors. So in that sense, PCA does not have assumptions in the same sense that a GLM has assumptions. Likewise, there is no 'response' variable, no likelihood, and no p-values. Really PCA is just a kind of complex data transformation. Sometimes people want to see if the loadings of variables on PC axes are 'significant', or something like that, and they use some kind of bootstrapping scheme to assess this. But because PCA isn't actually a model, you essentially have to create your own model/assumptions of what the data are supposed to look like before you can try to test a null hypothesis with PCA results; we'll see examples of this sort of approach in a few lectures.

Although PCA isn't a model, it still has properties that make it good for some kinds of data/questions and bad for others. By definition, principal components are looking for linear relationships in the data. So if variables in your dataset have nonlinear relationships, these will not be reflected in the analysis. A common approach is to transform the data to make things look more linear before doing PCA; in the example above, nitrate vs. chlorophyll looks kind of nonlinear in the scatterplot, so log-transforming chlorophyll might be a good idea (but I checked this and it didn't change the results). Likewise, PCA is all about decomposing the variance in the data, and this makes it sensitive to outliers, which can have a large effect on how much variation an axis explains. These should be evident in biplots, which allows you to remove outliers as a sensitivity analysis.

Ordination, why PCA is not optimal.

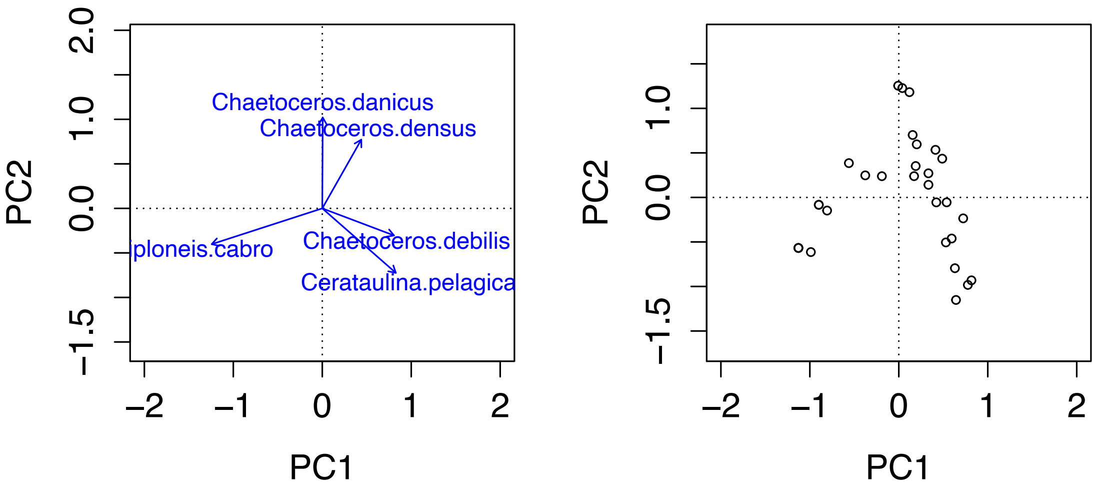
When we made a biplot with the English Channel example, we saw how the PCA revealed 1) the patterns of covariation among the variables, and 2) the location of the samples in PC space. This PC space was a 2D representation of environmental space, i.e. the 6D space defined by the environmental variables. By plotting the month of the sample, we also saw how PC space reflected seasonal variation. By mapping out the location of samples in this space, we can visualize their relation to each other, in terms of these environmental/seasonal axes.

Now we're going to turn our attention to the challenge of analyzing data on the composition of ecological communities. Although this is a somewhat specialized topic, it is relevant to many students in this course, and many of the techniques can be applied to other situations as well. One of the main challenges is mapping out variation in community composition, similar to how we used PC axes to map out variation in environmental conditions. In this case the data will be a matrix where each row is a particular site or timepoint that was sampled, and each column is the abundance of a particular species. For example, I have a matrix of phytoplankton abundances that correspond to the same timepoints from the English Channel when the environmental data were measured.

```
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
```

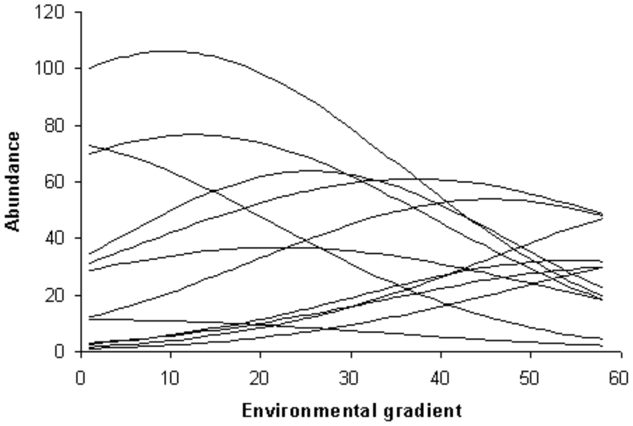
This shows the first eight columns (species) from the first sample. These are densities (cells per mL). The number of species identified in this time series is large (~100). The question I want to ask is: “What are the dominant patterns of variation in community composition?”. This is usually called *ordination*, because we want to order the community samples along some axes of dominant variation. Also, we often want to know if composition changes when particular conditions change, or if experimental treatments change, but for now we’re not thinking about the environmental conditions, just the species densities, and we want to know how the compositition of the community tends to vary over time. More commonly, researchers have samples from many spatial locations, but the question is the same.

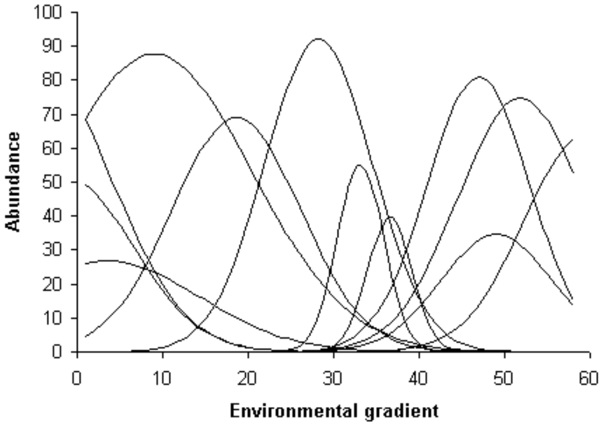
Can we use PCA for this task? It is usually not the best option, but let’s see what happens. If we used PCA, it will find axes of variation along which multiple species tend to increase or decrease in a correlated way.



Here I’ve used only 5 most common species in the dataset, just to keep things simple and legible. I’ve split up the biplot for clarity; the loadings of the species on the left, and the scores of the samples on the right. PC1 explains 34% of variation in the abundance data, and PC2 explains 25%. The species loadings are a bit hard to read, but they say, e.g., that *Chaetoceros danicus* tends to increase along PC2, but does not vary systematically along PC1; in contrast, *Diploneis cabro* tends to decrease along PC1, but does not vary along PC2. So you can see how we get information about which species tend to covary, and which ones don’t. In addition, we get scores for each sample, and we can map those onto PC space. This is the ordination part, where we can map out how community composition tends to vary across sample. And we could code those samples by month, or see which species are dominant at different ends of the plot. We could see if the PC axes tend to correlate with enviromental variables, or if we had an experiment, we could see if different experimental treatments are different in PC space. We also get information on the dimensionality of the variation in community composition; in this case there isn't one axis that explains a ton of variation, so composition might vary in a lot of different ways, perhaps reflecting different combinations of environmental conditions.

The problem with this approach is that PCA is a linear technique, but the relationships between species and environmental conditions, and between each other, are often not linear. This is particularly true if there is a lot of beta diversity, i.e. if there is a lot of turnover in species composition from sample to sample. The typical way to visualize this issue is by thinking about how species vary along an environmental gradient:





These are just imaginary curves that reflect how community data often looks. Each line represents the abundance of a species, along some environmental gradient. This could be soil moisture for plants across a region, or it could be irradiance for phytoplankton over time in the English Channel. The plot on the top has low species turnover; all species can often be found at all time, but their relative abundance changes. This situation might be OK for PCA, because we could capture most of the pattern via positive and negative correlations between species along this gradient. The plot on the bottom has high species turnover. Species present at one end of the gradient are absent at the other end, and the response of a species is strongly unimodal.

A linear method like PCA isn't great for the bottom scenario, because species don't simply increase or decrease relative to each other, and so linear relationships between species' abundances will not allow us to capture how the composition of the community varies. We can actually see this in the PC scores for the english channel example above. The scores have a U-shaped pattern. This is called the 'arch effect' or 'horseshoe effect', and is commonly seen in a variety of ordination methods. The presumed cause can be seen from the high-turnover plot above. If we compare a species that is only present at low values, and a species that is only present at high values, their abundance should not be highly correlated across samples. But there is a whole region in the middle of the gradient where both species are absent, and that tends to increase their correlation, because they both have zeros in those samples. So in our example U-shaped plot, the samples at opposite tips of the horseshoe may be at opposite ends of the environmental gradient, but they get bent towards each other on PC2 because they are similar in that they do not contain the intermediate species.

Although we can see the arch effect in the plot above, that's not the only reason PCA might not be great. In general the nonlinear relationship between gradients and abundances means that PCA is likely to miss important patterns, unless the gradient is 'short' in terms of species turnover.

Dissimilarities

Now we'll think about alternatives to PCA for ordination. PCA picks axes by finding directions in multivariate space that account for as much variation as possible. When dealing with community data, the samples are points in high-dimensional 'species space', and the variation between two samples is the euclidean distance between the vectors of species abundances. Let's consider that for a second:

```
species.data.use[2,]

##    Cerataulina.pelagica Chaetoceros.danicus Chaetoceros.debilis
## 32                 0.06                0.12                0.28
##    Chaetoceros.densus Diploneis.cabro
## 32                  0           0.005

species.data.use[3,]

##   Cerataulina.pelagica Chaetoceros.danicus Chaetoceros.debilis
## 7                 0.32                0.08                0.06
##   Chaetoceros.densus Diploneis.cabro
## 7                  0               0
```

These are the abundances of the 5 species in two different samples. If we think of these as points in 5-dimensional space, then the euclidean distance between the two samples is:

```
euclidean.distance = sqrt(sum((species.data.use[2,] - species.data.use[3,])^2))
euclidean.distance

## [1] 0.343
```

So we can this of the euclidean distance as one metric for defining how dissimilar these two communities are, in terms of species composition. We can then think of PCA on community data like this: *PC1 is the axis along which the samples have the greatest total dissimilarity*. And PC2 is the orthogonal axis along which the samples have the greatest residual dissimilarity, etc. So this is what we're trying to achieve with ordination, is finding the axes along which the communities show the most variation. But we've seen how defining axes in terms of euclidean distance can be problematic (nonlinear responses to environmental gradients; the arch effect). The popular (better) ordination methods improve on PCA by defining dissimilarity between two community samples with some other number, and using that for the analysis.

So we need to imagine that we have two community samples, and we want to define how similar they are to each other. Then we can take that similarity, S, and use it to get a dissimilarity, D = 1 – S, and use that dissimilarity as a 'distance' for ordination. There are a mind-numbing number of ways to define community similarity. You can see check out ?vegdist in the vegan package to get a sense for the possibilities. Rather than try to parse out the pros and cons of all these metrics, I'll explain a couple that are commonly used, and left the rest up to you.

It's easier to think about similarity using presence-absence data, i.e. each species is just a 0 or a 1 in each sample. The Jaccard index (or Jaccard similarity coefficient) is defined like this:

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

Where *J* is the number of species present in both samples; *A* is the number of species present in sample 1 but not sample 2, and *B* is the number of species present in sample 2 but not sample 1. We can then define dissimilarity as D = 1-S. This number ranges from 0 to 1; if all species are present in both samples, it equals one, and if no species are in common between samples, it equasl zero. *Note that this coefficient ignores species that are absent from both samples*. The question of what to do with species that are absent from both samples is called the 'double zero problem'. The thought process is: if a species is present at two sites (or two times), we know that those sites both have conditions that allow the species to (temporarily) persist. But if a species is absent from two sites, does that mean those sites are similar? Not necessarily, because the species may be absent from the two sites for two different reasons (e.g. one is too hot and the other is too cold). So the recommendation is that mutual presence is a better indicator of similarity than mutual absense. Therefore an index like Jaccard's only uses presences.

Jaccard's index is for binary (presence-absence) data. A similar one that uses quantitative abundance data is called the Bray or Bray-Curtis index. This one is easier to write in terms of the dissimilarity:

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

Here the dissimilarity between samples *j* and *k* is determined by the absolute difference in abundance of species *i* between the two samples, summed over all species, and divided by the total abundance across all species and both samples.

The Bray dissimilarity is the default one in vegan, and is generally thought to be a good choice. One important choice for abundance data (as opposed to presence-absence) is transformation. An index like the Bray (and others) is based on how much abundance differs between sites. This means that a difference between 50 and 100 (e.g. an abundant species) will matter more than a difference between 5 and 10 (e.g. a rare species). Typically when looking at community composition, we get more sensitive results if each species gets equal 'weight'. At the same time, it's a good idea to drop rare species because they just add noise to the analysis. So, after dropping an somewhat arbitrary number of species, it is common to transform the data to put the species on equal footing. Using log(x + min(x[x>0])) is an easy one. The default in vegan, i.e. if you give an ordination function raw abundance data, is to first square root transform, and then do a 'Wisconsin' standardization, which divides each species by its maximum abundance (across sites), and then standardizes sites so that they have the same total abundance.

You're starting to see that in order to do ordination, there's a lot of black magic. First we (double) transform the data, then we use a dissimilarity coefficient to summarize how different two samples are. One of the issues with these methods, which we will return to, is that were don't actually have a model for the raw data. In contrast to something like GLMs, where we were trying hard to model the raw data, in order to get the most sensitive and not-spurious answers. Actually modeling the raw data is hard with a problem as complex as ordination of communities, but later we will consider some possibilities.

In the following lecture, we'll look at ways to do ordination using the dissimilarity indices that are more appropriate for communities.