Visualising Data

Jason Matthiopoulos 2011

Overview: Discovering patterns in raw data is one of the most necessary and satisfying parts of the job of a scientist. In this practical, we learn how to plot different types of variables informatively, how to improve the presentation of these graphics and how to calculate basic descriptive summaries from data sets.

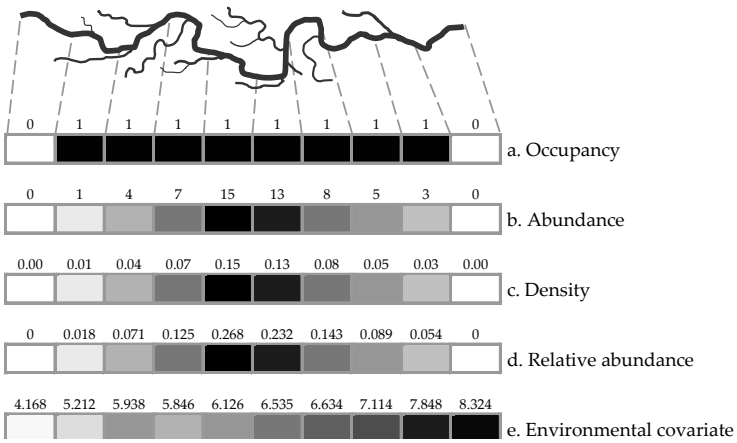
1. the Study System

The Amazon river dolphin (*Inia geoffrensis*), also known as boto, is the largest of the river dolphins - males can be as long as 2.5 m and weigh up to 207 kg. It lives along the entire length of the Amazon and its main tributaries. Their range spans 6 South American countries. Their mobility is limited by impassable falls, rapids and lengths of shallow water. The boto are active day and night, foraging for fish mostly in the early morning and late afternoon. They are generalist piscivores, whose diet contains over 43 species of fish from 19 families. They are mostly solitary animals, and the rare sightings of social groups don't exceed four individuals.

The Amazon river dolphin

2. ecological theory

There is near-correspondence between the different types of ecological measurement and the different types of numbers. For example, consider the measurements typically used for the distribution of populations in space. To make things simpler, think of a linear study site, such as a stretch of river. The site has a length of 1km and has been subdivided into 10 segments. We are interested in characterising the spatial distribution of a particular species along the length of the river. The easiest description is in terms of occupancy, the presence or absence of the species. Although occupancy can be thought of as a nominal variable, it is often converted into an ordinal variable by attributing the value 0 to absence and the value 1 to presence. If, in addition, there are data on the number of occurrences in each segment, then distribution can be described by a series of counts which take their values from the set of non-negative integers, $0,\mathbb{N}$. These count data can be readily converted to densities by dividing each count by the size of the segment – in this case 100m. Standardised density (or relative abundance) can be obtained by dividing the count in each segment by the total number of observations. This conveys the proportion of the total count occurring in any given segment. Both density and relative abundance are rational numbers. Finally, we may want to compare the distribution of the species with that of some environmental covariate that could be used as a proxy at other unsurveyed sites (A covariate is a quantity that is closely related to the measure of interest). These measurements may take their values from the set of real numbers.



Measurements belonging to different sets of numbers naturally occur in ecology. In quantifying the spatial distribution of a species, we may use nominal data (occupancy), non-negative integers (counts of abundance), rational values (density, or relative abundance, and real numbers (environmental covariates).

3. Mathematical theory

Consider a random variable X. A particular **realisation** of that variable in a sample of measurements is written x. The number of occurrences of the value x in the sample is called its **absolute frequency** (we may write it as $f(x)$). If the sample has size n, then the relative frequency of the value x, is defined as

$$F(x) = \frac{f(x)}{n} \tag{1}$$

Plotting a histogram of the relative or absolute frequencies of all possible values x, produces the **frequency distribution** of the sample. There are several **measures of position** for a frequency distribution. The **average** is defined in three equivalent ways

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n} = \frac{\sum_{i=1}^{m} x_i f(x_i)}{\sum_{i=1}^{m} f(x_i)} = \sum_{i=1}^{m} x_i F(x_i) \tag{2}$$

Where, m is the number of different values that can be assumed by the random variable X. The **mode** is the value of X that occurs with the highest frequency. The **median** is the middle observation in am ordered data set.

There are several measures of spread for a frequency distribution. The **range** comprises the minimum and maximum value observed in the data. The **inter-percentile range** is the interval of values between the top and bottom $a\%$ of observations. For example, the **interquartile range** is the interval of values in the middle 50% of observations. The standard deviation is calculated as a standardized measure of dispersion

$$S.D.(x) = \sqrt{\frac{\sum_{i=1}^{n} (x_i - \bar{x})^2}{n-1}} \tag{3}$$

■ *Plotting histograms*: If you have a list of raw data you can view their frequency distribution using the command `hist()`. However, when you have already constructed a list of relative or absolute frequencies then you can achieve the same result by using the command `barplot()`. The command `hist()` generates a plot and a data object. To see the contents of the object, assign the output of the command to a name e.g. `a<-hist(mydata)`. Typing `a$mids`, will give you the midpoints of the binning categories used for the histogram. Typing `a$counts`, will give you the absolute frequency of observations falling in each histogram bin.

■ *Getting absolute frequencies for nominal variables:* In R, nominal variables are called factors. The command `table()` will calculate the absolute frequencies for a data set that contains values of a factor.

■ *Pie charts:* Start with a vector of non-negative values (say, `a<-c(10, 20, 30,15)`) and a vector of labels (say, `lab<-c("Class 1", "Class 2", "Class 3", "Class 4")`. If you pass to the command `pie()`, the output from the command `table()`, the pie chart is labeled automatically.

■ *Scatterplots*: A scatterplot requires two lists (one for the x and one for the y-axis) of equal length. Simply typing `plot(xlist,ylist)` will generate the plot.

■ *Adding lines to plots*: The command `abline(c,m)` will create a straight line with slope `m` and intercept `c`.

■ *General plotting options:* There are many different options for plotting. You can learn about all of them by typing ?par. The command `par()` can be used to set the options for all graphs generated during an R session. Alternatively, you may specify potting options within each individual plotting command. Here is a small selection of the most useful options.
- `main="my plot"` will add a title to the top of your plot.
- `xlab="my x label"`, `ylab="my y lablel"` will add labels to your axes
- `xlim=c(0,10), ylim=c(-10,10)`, will define a plotting range for each of the two axes
- `type` specifies what type of plot should be drawn. Three possible types are `"p"` for **p**oints, `"l"` for **l**ines, `"b"` for **b**oth.
- `lty` specifies the line type. Line types can either be specified as an integer (`0=blank, 1=solid` (default), `2=dashed, 3=dotted, 4=dotdash, 5=longdash, 6=twodash`) or as one of the character strings `"blank", "solid", "dashed", "dotted", "dotdash", "longdash"`, or `"twodash"`, where `"blank"` uses 'invisible lines' (i.e., does not draw them)

■ *Creating composite plots:* If you want to create a plot containing several individual plots you can do this by using the options `mfcol` or `mfrow` within the general graph specification command `par()`. Here is an example:

```
n<-seq(0,2,0.1) # Creates a sequence from 0 to 2 at steps of 0.1
par(mfrow=c(3,2)) # Splits graphics output into a table of 3 rows and 2 columns
# The next six commands generate six different graphs
plot(n,sqrt(n), xlim=c(0,2), ylim=c(0,10))
plot(n,n, xlim=c(0,2), ylim=c(0,10))
plot(n,n^2, xlim=c(0,2), ylim=c(0,10))
plot(n,n^3, xlim=c(0,2), ylim=c(0,10))
plot(n,n^4, xlim=c(0,2), ylim=c(0,10))
plot(n,n^5, xlim=c(0,2), ylim=c(0,10))
par(mfrow=c(1,1)) # This line resets the graphics device to a 1x1 table
```

■ Summary statistics: Summary statistics for any particular variable, or even for an entire data frame, can be obtained via the command `summary()`.

5. Practical tasks

The data set for today's practical can be found in the Excel file "Boto data" (downloadable from webCT or the memory sticks floating around in the practical venue). This is a record of observations from 50 segments of a river tributary, each 200m long. The columns provide segment-specific information on the maximum group size observed, the total number of group detections, the slope, depth and substrate of the river bed, the average water speed and the total number of visits made by the observers.

■1■ Read carefully the material in sections 1-4 above.
■2■ Read through the tasks in this section.
■3■ Read through the assessment components in the next section before you carry out any tasks
■4■ Import the data into R as a data frame (say, `botodat`).
■5■ Create histograms for the variables depth, slope and speed. Make sure that each is appropriately labelled.
■6■ Calculate the absolute frequency of different substrate types in the sample.
■7■ Create a pie chart of the relative prevalence of different substrate types in the sample
■8■ Observation effort varies between river segments. Calculate a list (say, `detS`, of detections per visit.
■9■ Append this as a new column to your data frame.
■10■ Create a scatterplot of standardised detections versus slope. Annotate this plot.
■11■ Create two more scatterplots relating standardised detections to speed and depth.
■12■ Pick one of the three plots (preferably the one which shows the strongest association between its two variables). By trial-and-error, add to this plot a best-fit line. Observe your method for carrying out this task.
■13■ Calculate the following quantities: i) The range of speed, ii) the inter-quartile range of depth, iii) the average number of visits and iv) the mode of the distribution of depth.
■14■ From these data, is it possible to estimate the relative abundance of groups of dolphins along the river? Is it possible to estimate the relative abundance of individual dolphins?

6. Assessment

Write a report, no longer than 3 sides of A4, that contains the following:

■1■ Histograms from task 5.
■2■ Substrate pie chart from task 7.
■3■ Scatterplots and best-fit line from tasks 10 & 11.
■4■ Answer to question in task 13
■5■ Answer to question in task 14
■6■ Fully commented Tinn-R code