Lecture 2. Probability distributions you should know

> "The most important questions of life are, for the most part, really only problems of probability." Pierre Simon, Marquis de Laplace

We typically treat variation we can't explain as "noise". The deterministic part of a statistical model has the interesting stuff (the predictors we care about), and the stochastic part is just a way of expressing our ignorance about extra variability. Classically we treat this variation as normally distributed, or we transform the data to make it look as normal as possible, and then put it to the side. If the data can't be transformed to normality, often a "non-parametric" test is used that has less assumptions.

This point of view has some problems. From a practical point of view, non-parametric tests have low power, because they make few assumptions about what the data look like, and they should be avoided in the modern era. Likewise, transforming the data can work well in some situations but often does not, again resulting in low power or biased results. More fundamentally, the shape of the residual variation will reflect the biological processes that are generating the data. So it behooves us to think about why the variation looks the way it does, and furthermore by modeling it appropriately our hypothesis tests will be more sensitive.

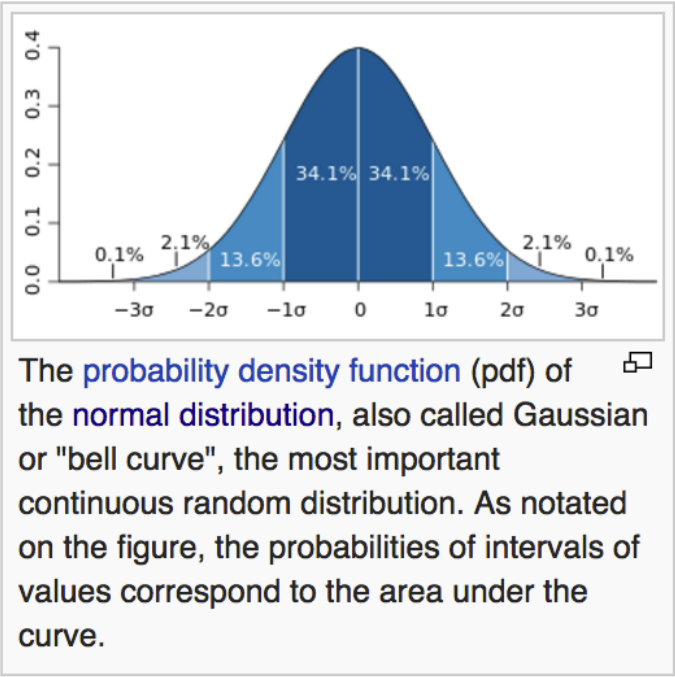
It's useful to think about residual variation as comprising two components: *measurement error*, and *process noise*. Measurement error is due to inaccuracies in the technology or methods we use to measure things, and this will also just add noise and make it harder to pull out the interesting signal. Process noise is essentially a statement of our ignorance: if we knew all the important processes, and measured them, and modeled them appropriately, then there would be no process noise and we would be able to predict how the world works very well. In reality there are many processes we don't measure or don't understand, and that constitutes most of the unexplained variation, especially in ecological datasets.

To think about the different ways to model biological variability, we'll first review some basic probability theory:

- Probability distributions can be *continuous* or *discrete*. Discrete distributions only take on integer values (0,1,2,…), while continuous distributions can take on any real value (over their defined range).

- Continuous distributions are defined by their *probability density function*

(pdf):

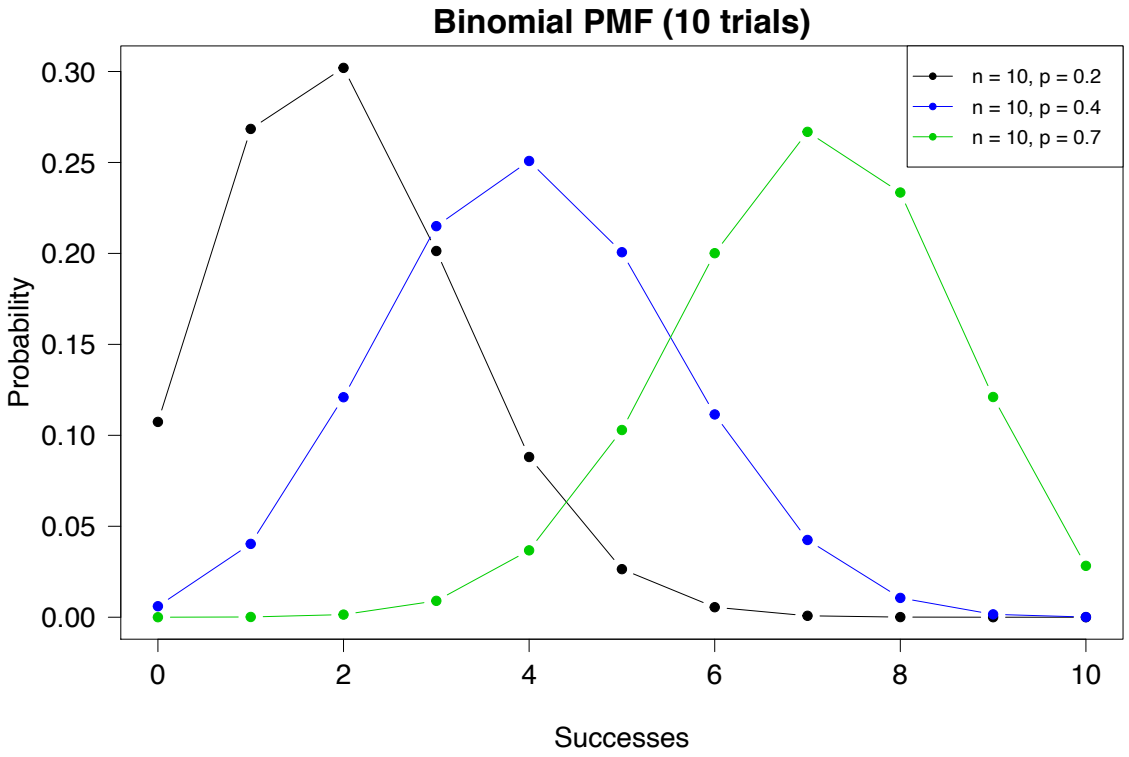


The probability density function (pdf) of the normal distribution, also called Gaussian or "bell curve", the most important continuous random distribution. As notated on the figure, the probabilities of intervals of values correspond to the area under the curve.

source: Wikipedia

This figure shows the pdf for the normal distribution. It's important to note that *the points along a pdf curve are not probabilities*, although they are closely related to probabilities. Because a continuous variable can take on a continuous range of values, the probability of any unique value is infinitesimally small (hopefully this is bringing back memories of calculus). Probabilities are defined by *integration*, and pdfs are defined so that the integral under the whole curve equals 1. This makes sense, because when you draw a random number the probability of getting some number is 1 by definition. In the figure above, the area between **μ** and **1*σ** is labeled 34.1%, because the integral of this curve from **μ** to **1*σ** is 0.341, which means that the probability of a random draw from this distribution coming from the shaded area is 0.341.

- Discrete distributions are defined instead by a *probability mass function* (pmf):



This shows the probability mass function for the binomial distribution with 10 trials, for three different probabilites of success (we will discuss this distribution shortly). In general, the values of a probability mass function are in fact probabilities that the distribution will have that value, in contrast to continuous distributions.

For both continuous and discrete distributions, the shape of the distribution will depend both on the equation for the probability function, as well as the parameters that define the distribution. For our purposes we won't need to pay as much attention to the formula, but we will need to think about how many parameters define a distribution, and what they mean. For the normal distribution, the equation for the pdf is

$$\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

where x is the random variable, μ is the mean of the distribution, and σ is the standard deviation. When thinking about the kinds of shape a distribution can have, it's good to keep an eye on how the *mean*, *variance*, and *skew* of the distribution change as the parameters change. For the normal distribution this is very easy: the distribution is defined in terms of its mean and variance, and these quantities can vary independently. In addition, the normal distribution is always symmetric, and so it has no skew.

In R, each distribution gets a set of four functions to help define/use it. For the normal distribution, these are:

- dnorm() – the density function: tells you the height of the pdf for some value *X*, with the mean and variance of the distribution specified
- pnorm() – the 'distribution function': this tells you the probability that a random variate will be less than *X*. In other words, this function integrates over the pdf for you.
- rnorm() – take a random draw from a distribution with a specified mean and variance
- qnorm() – the inverse of pnorm(). tells you what *X* should be to get probability equal to some *p* you specify.

A brief digression on random number generators: the random numbers generated by rnorm() or any analogous function are not truly *random*. To really get a random number, you'd have to rely on some random physical process, like atmospheric noise or the decay of radioactive elements. R and other software uses *pseudo-random number generators*, which are complex deterministic functions that produce a series of numbers that pass many tests of apparent randomness. This means that when you write some code that uses random draws, it is possible to reproduce the exact same results at a later date using the set.seed() function. E.g. if you start your R script with set.seed(1000), then do rnorm(1), you will always get the same answer.

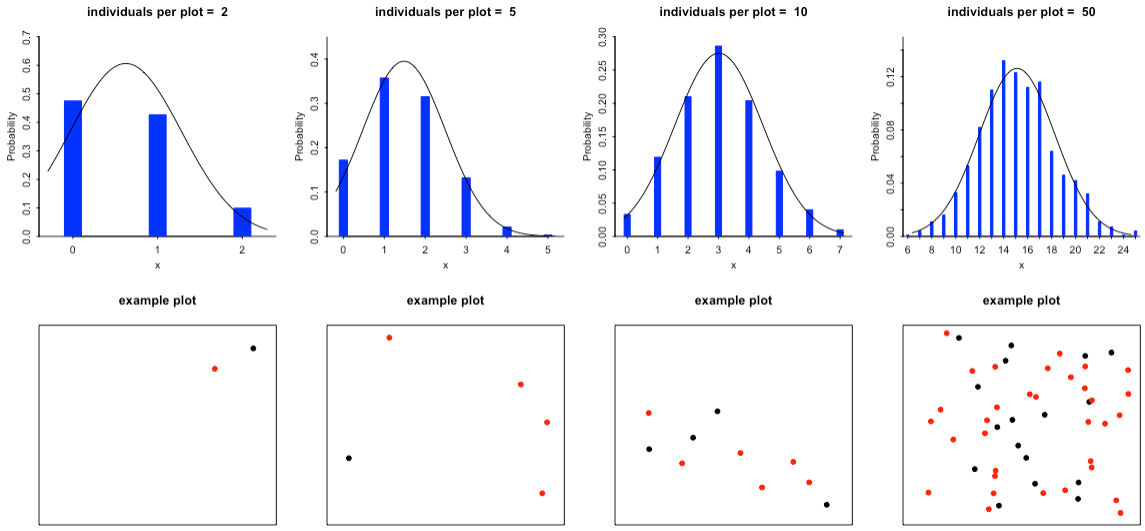
Random processes and the distributions they generate

Now we'll think about the biological processes that can generate different kinds of probability distribution. We'll start with the normal distribution, because it's enlightening to understand why the normal distribution is so common, as well as the conditions under which variability is not normally distributed.

Normal distribution: central limit theorem. The reason the normal distribution is so common is explained by the *central limit theorem*. The central limit theorem is easy to understand in nontechnical terms:

1) Start with some random process
2) Create many independent random events from this process
3) Add up the results of these random events
4) If you add up a large enough number of events, the sum will be normally distributed. This also means the average of a large number of random events will be normally distributed, because the average is just the sum divided by the number of events.

Let's look at an example:



Imagine that we plant some trees in a number of plots, and we come back later to see how many survived. For each plot we record the total number of survivors (or the average number of survivors per plot would give the same result). I've shown four scenarios, with different numbers of trees planted per plot: 2, 5, 10, and 50. On the top row I've plotted for 1000 hypothetical plots what the distribution of # survivors looks like. I've also plotted the curve for a normal distribution with the same mean and variance as the simulated data. On the bottom row is just an illustration of what a hypothetical plot looks like, with 30% survival on average. The central limit theorem says that the sum of a large number of random variables will be normally distributed, and indeed as #trees per plot increases the distribution of #survivors becomes closer to a normal distribution. Code for this simulation is below:

```
#make a vector for four scenarios
individuals.per.plot = c(2, 5, 10, 50)
#define the number of times to repeat the experiment
nplots = 1000
#make an empty vector to store the results
survivors = vector()
#define the probability of survival
survival.probability = 0.3
#set up the graphics to hold 8 plots
par(mfcol = c(2,4))

for (i in 1:4) {
  for (j in 1:nplots) {
    #use the binomial distribution to draw random survivors, and sum the numbe
r of survivors
    survivors[j] = sum(rbinom(individuals.per.plot[i], 1, 0.3))
  }
  #make a histogram of the number of survivors per plot
  discrete.histogram(survivors, col = 'skyblue', main = paste("individuals per
```

```
 plot = ", individuals.per.plot[i]), freq = FALSE)
  #plot a normal curve with the same mean and variance as the simulated data
  curve(dnorm(x, mean = mean(survivors), sd = sd(survivors)), add = TRUE)
  #throw some points down on a plane to illustrate trees in a plot
  plot(runif(individuals.per.plot[i]), runif(individuals.per.plot[i]), xlab =
'', ylab = '',  xaxt = 'n', yaxt = 'n', pch = 19, col = c('red', 'black')[rbin
om(individuals.per.plot[i],1,0.3)+1], main = 'example plot', xlim = c(0,1), yl
im = c(0,1))
}
```

The central limit theorem show why the normal distribution is so common: if a variable is generated by adding or averaging a large number of components, then that variable will tend to be normally distributed. In the simulated example, we had a large number of random survival events added at the plot level to yield a normal distribution of total survivors. Another classic example comes from quantitative genetics. Some traits are determined by a single gene, such as sickle-cell anemica. But many traits are influenced by many different genes, such as human height. If a trait is influenced by a large number of genes, then the distribution of that trait in the population will be normally distributed. This is illustrated by this human histogram:

Figure 10.15 Continuous variation in body height, one of the traits that help characterize the human population.

(**a**) A bar graph can depict continuous variation in a population. The proportion of individuals in each category is plotted against the range of measured phenotypes. The curved line above this particular set of bars is an idealized example of the kind of bell-shaped curve that emerges for populations showing continuous variation in a trait.

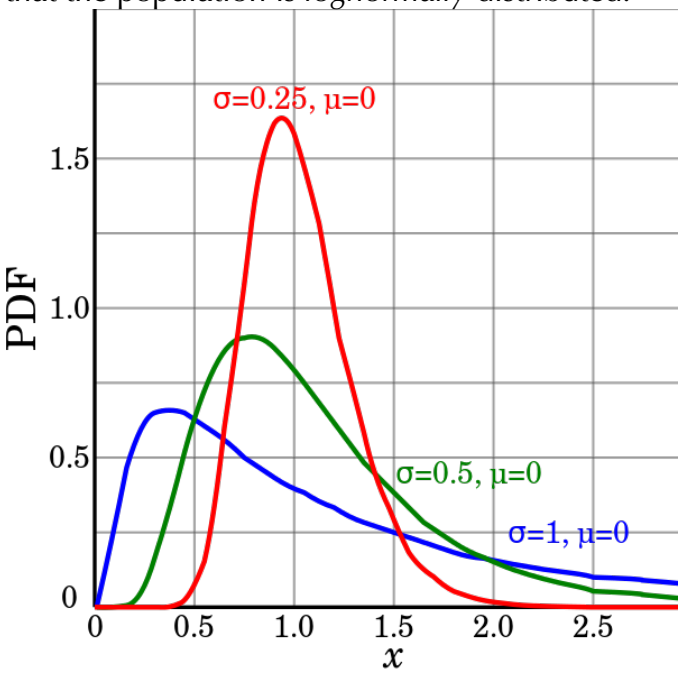
(**b**,**c**) Jon Reiskind and Greg Pryor wanted to show the frequency distribution for height among biology students at the University of Florida. They divided students into two groups: male and female. For each group, they divided the range of possible heights, measured the students, and assigned each to the appropriate category.

c Two examples of continuous variation: Biology students (males, *left*; females, *right*) organized by height.

Lognormal distribution: multiply many things

Similar reasoning shows why log-transforming will very often yield normally distributed error. Many processes in biology (and elsewhere) involve *multiplying*

many small effects, rather than adding many small effects. For example, if a population is growing exponentially, and the environmental conditions are variable, then $N(t) = N(0)* \lambda_0 * \lambda_1 * \lambda_2 *\ldots \lambda_{t-1}$, where N is the population size at time (t), and λ_{t-1} is the finite rate of increase from *t-1* to *t*. Environmental variation causes the lambdas to vary randomly over time, and so the size of the population results from the multiplication of many independent random events. The result is that the population is *lognormally* distributed:



source: Wikipedia

This plot shows three different density functions for the lognormal distribution. This distribution is called lognormal because if *X* is lognormally distributed, then log(*X*) is normally distributed. We can see this from the rules for logarithms:

$\log(\lambda_0 * \lambda_1 * \lambda_2 *\ldots) = \log(\lambda_0) + \log(\lambda_1) + \log(\lambda_2) + \ldots$

In other words, if something like population abundance results from multiplying together many events, then log(population abundance) is the addition of many events, and so log(population abundance) will be normally distributed. This means that analyzing this data on a log scale will work very well.

An interesting application of the lognormal distribution that gets into ecological theory is the *species abundance distribution*. A classic question in ecology is why are some organisms abundant and others rare. To quantify patterns in abundance and rarity, samples of a community can be plotted like this:

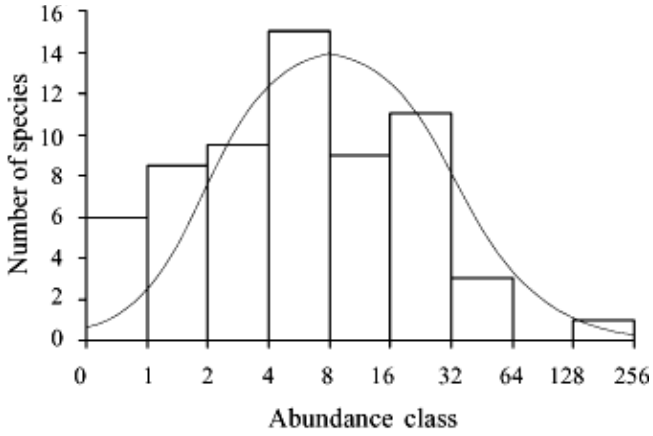


Figure 1. Abundance distribution of woody species sampled in a cerrado fragment in São Carlos, southeastern Brazil, with expected normal curve.

source: Oliveira & Batalha 2005

Here the abundances of all the woody species in a tropical savannah are plotted, with the abundance binned on a log scale. The distribution is approximately normal, which means that the distribution of abundances in the community is lognormal. Many different kinds of curve have been fitted to this kind of data for many kind of community, and ecologists still argue about how to interpet these patterns and what kinds of ecological processes cause these patterns. A classic argument by Robert May is the same one I gave above for population growth. If the abundance of each species is bounced around randomly by randomly varying exponential growth, then the distribution of abundance across species will be lognormally distributed. This explanation may not be very ecologically satisfying, because it ignores all the details of the processes that affect populations, but the fact that the simple model predicts the data well is thought-provoking.

Non-normal distributions: modeling smallish numbers

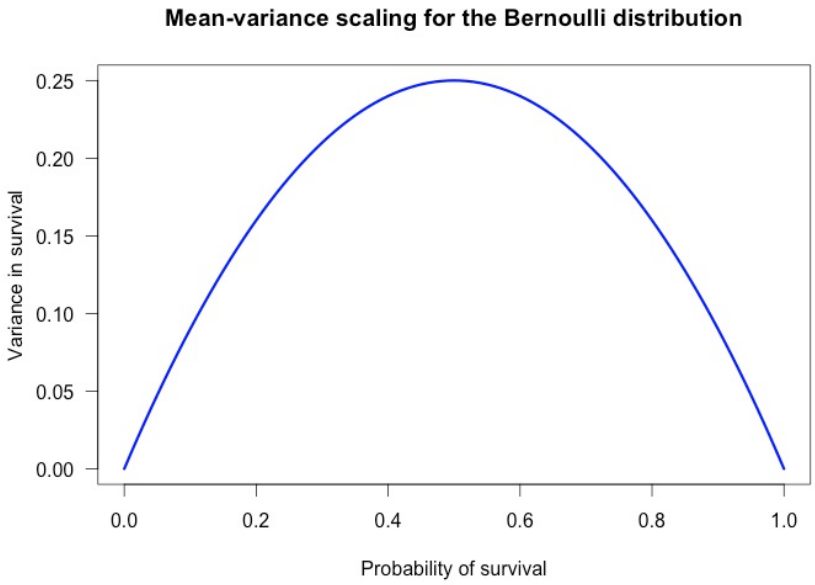
The normal and lognormal distribution results from combining many small effects into one random variable. Conversely, these distributions don't work very well when we are measuring a process involving a smaller number of events. This is the case for two other distributions very important in biological applications, the binomial and Poisson distributions.

Binomial: events with two outcomes

To understand the binomial distribution, let's start with a special case, the Bernoulli distribution. The Bernoulli distribution is very simple, it's what you get when a variable can be either 0 or 1, e.g. the survival of an organism where you code death as 0 and life as 1. The random events are often called "trials". The distribution is parameterized with one parameter, p, which is the probability that

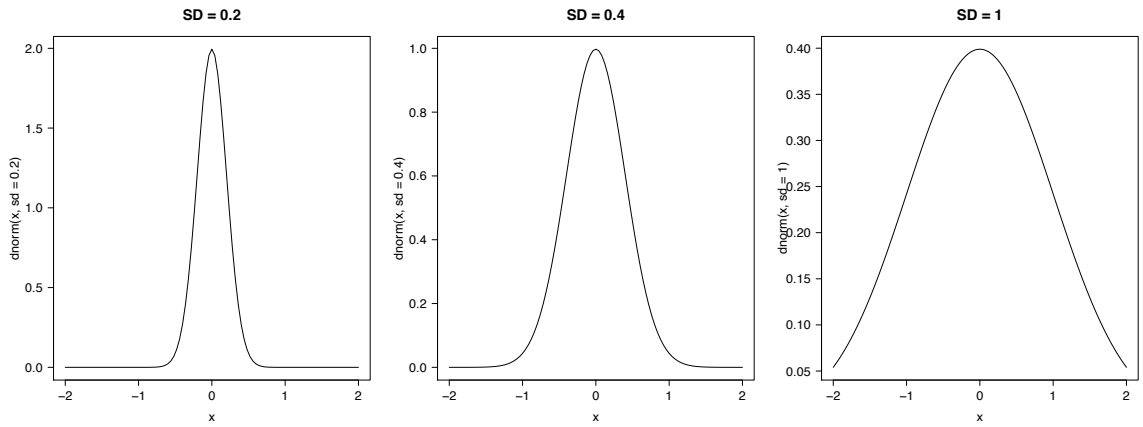
X=1 (i.e. the probability of living in our example).

A digression on mean-variance scaling. One thing we can learn from this simple distribution is that some distributions have a specific *mean-variance scaling*. What does this mean? If we were to look at a large number of trials, across a range of mean survival probabilities, then the variability in the outcome of those trials looks like this:



The variability in survival goes up and then down as the probability of survival increases; mathematically the variance is *p*(1-p)*. This makes sense, because if the probability of survival is very low (or very high), then nearly everyone will be dead (or alive). You'll see the greatest variability in outcomes when there is an equal probability of living or dying.

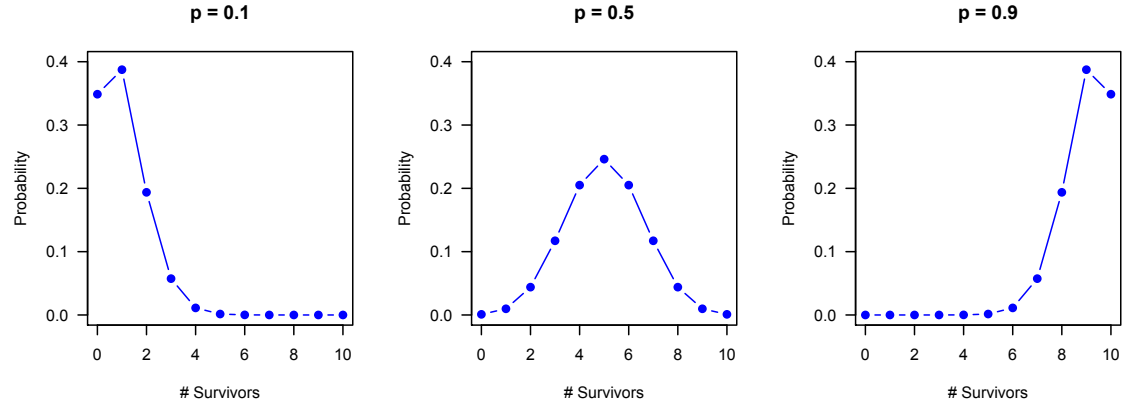
Mean-variance scaling is an important difference between the normal distribution (and lognormal) and the other distributions we'll be learning in this course. The normal distribution is defined by two parameters, the mean and the variance, which means the variability in the data can be big or small independent of the mean:



In contrast, when we start to model data with other distributions, we'll have to pay attention to whether the relationship between the mean and variance is correctly modeled. This is important because modeling the variance correctly has a large effect on whether you get valid results.

Now back to the binomial distribution. The binomial is just the sum of a certain number of Bernoulli trials. For example, you follow 10 individuals and see how many of them survive. The binomial is parameterize with two parameters, n = number of trials, and p = probability of 'success'. The mean of the distribution is $n*p$, which should be intuitive because it's just the expected number of successes. The variance of the distribution is a generalization of the variance for the Bernoulli: $n*p*(1-p)$.

Because the binomial distribution is modeling the trials X out of some total n, it can take on a variety of shapes. When the probability of success is low, it will be right-skewed, while if the probability of success is high it will be left-skewed, and for intermediate probabilities it is more symmetric:

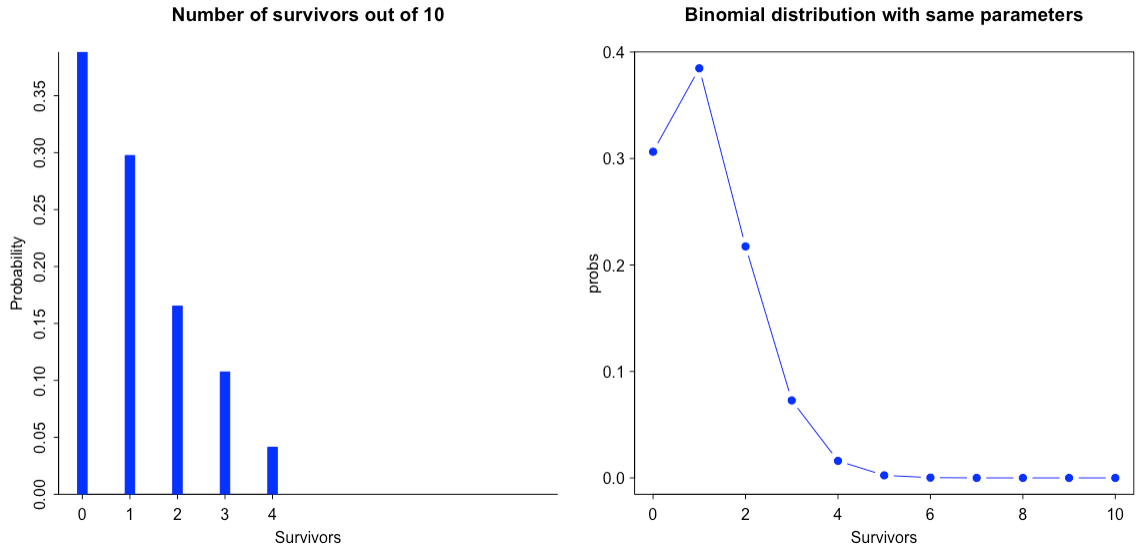


As mentioned above for the Bernoulli (which is a binomial distribution with the

number of trials = 1), this leads to a hump-shaped relationship between *p* (probability of success) and the variance of the distribution.

Here is an example of real survival data from an experiment where the Silverleaf whitefly, and agriculture pest, was subjected to heat shock. There were 10 individuals per replicate, and the number surviving was recorded for different temperature treatments and source populations.

```
#load the data
survival = read.csv("heat_shock_survival.csv")
#set up graphics
par(mfrow = c(1,2))
#take a subset from just one treatment and population
sub = subset(survival, treatment == "40-45\xa1C" & population == "PAV")
#plot the distribution of #survivors
with(sub, discrete.histogram(Survival, main = 'Number of survivors out of 10',
 col = 'skyblue', xlab = 'Survivors', freq = FALSE, xlim = c(0,10))))
#plot the equivalent binomial distribution with the same mean survivorship
survivors = 0:10
probs = dbinom(survivors, size = 10, mean(sub$Survival/10))
plot(probs ~ survivors, main = "Binomial distribution with same parameters", c
ol = 'blue', type = 'b', pch = 19, xlab = 'Survivors', las = 1)
```

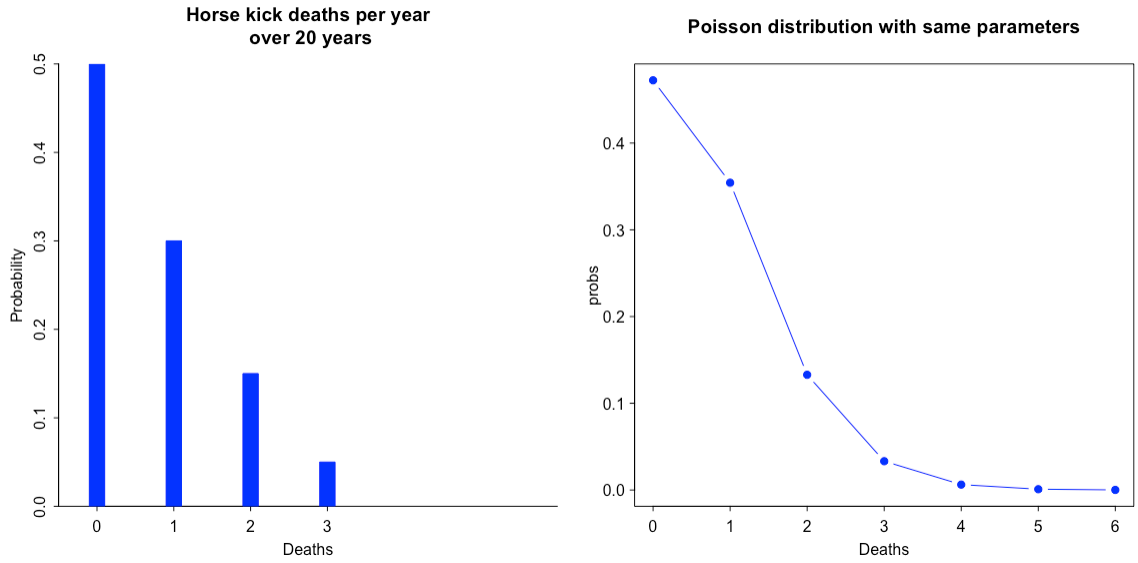


The empirical distribution on the left looks pretty similar to the predicted binomial distribution on the right. Here I've plotted the x-axis as total number of survivors, but we could also plot it as the proportion that survived (then the column that corresponds to 2 surivors would just be labeled as the proportion 0.2 instead). Because the binomial distribution is defined in terms of n=#trials, *this distribution is most commonly used to model proportion data when you have a fixed number of 'trials'*. Survival data is a common use, but anything that fits this kind of process is appropriate. For example, if you are measuring a trait that has two states (e.g. a flower that can be white or blue), then you can model the proportion of flowers

that are blue with a binomial distribution. Or the number of individuals infected with a disease out of the total population.

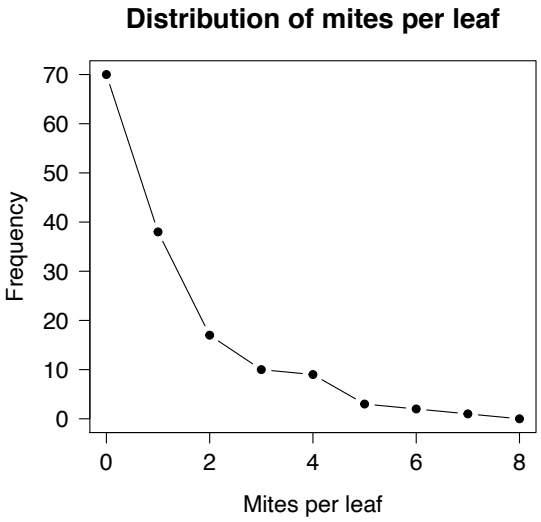
Poisson distribution: count data

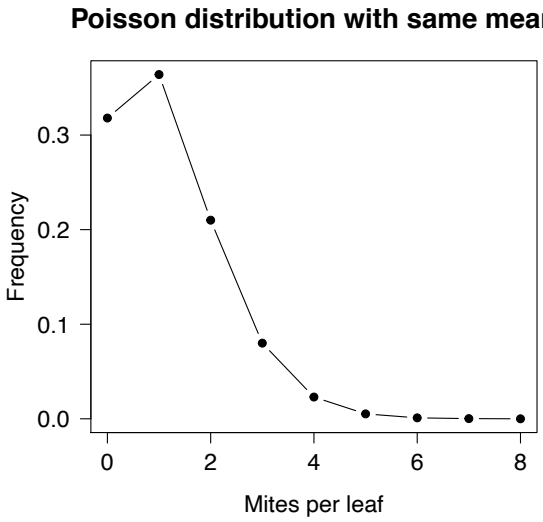
The other important discrete distribution for biological data is the Poisson. What kind of processes lead to a Poisson distribution? Suppose that there is some event that has many chances to occur, but actually happens rarely. The Poisson distribution tells you how many times the event occurs, if it has a mean probability of λ. This idea is illustrated by some interesting data from a book called *The Law of Small Numbers* by Bortkiewicz. Among other things, he looked at data from the Prussian military, where they had records for how many deaths per year were due to a soldier getting kicked by a horse. So this is something that happens rarely (thankfully), but has many chances to occur (these were cavalry units). Here are some of the data:



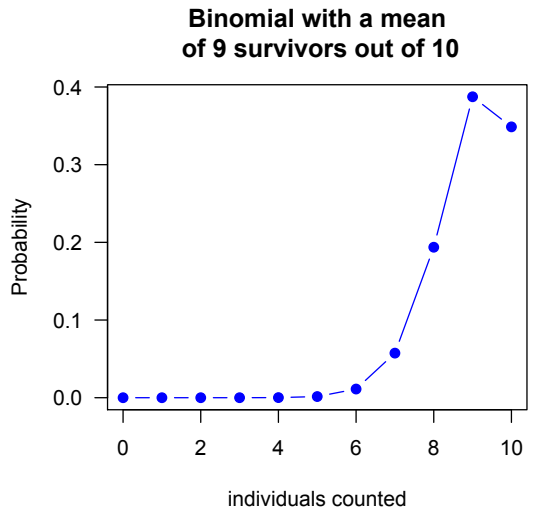
As Bortkiewicz showed, the data follow a Poisson distribution pretty well.

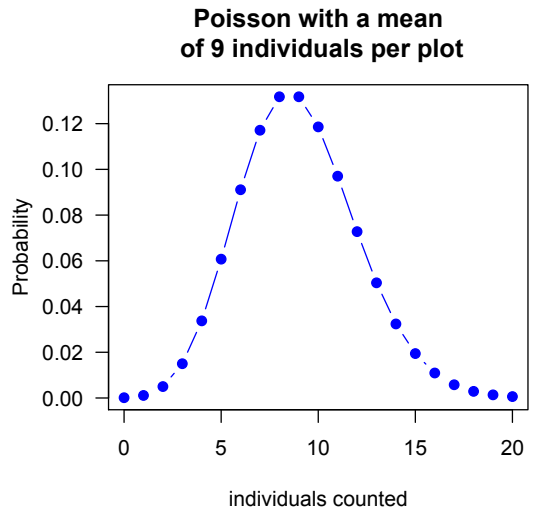
It turns out this kind of process is widely applicable in nature, especially when counting small to medium numbers of something. For example, if we count how many fish occur in a quadrat, there are many locations a fish could occur, but we only see a fish in a small subset of those locations. *For our purposes, count data on organisms will be the most common use of the Poisson distribution*. Here's an example from counts of red mites on apple leaves:



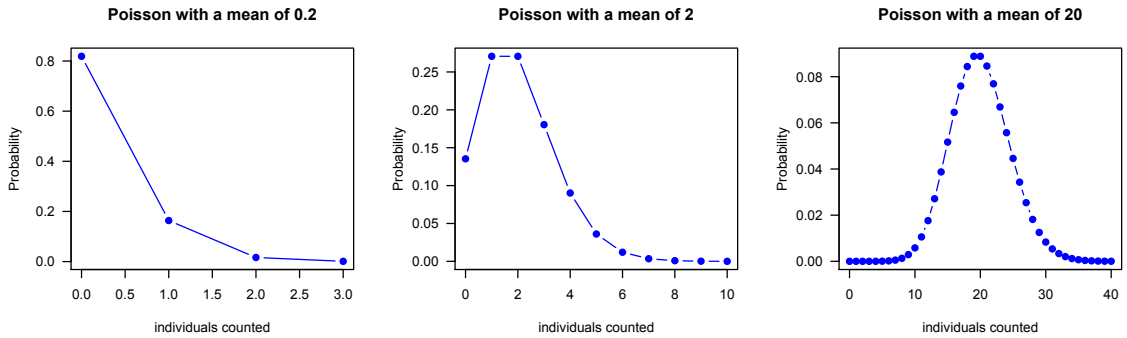


You may have noticed that the binomial distribution and the Poisson have some conceptual similarities, and the example plots I've shown actually look pretty similar. The important difference is that the Poisson distribution has no upper limit (e.g. there is no *a priori* limit on how many mites per leaf), and in fact the Poisson distribution can be derived from the binomial by assuming that there are an infinite number of trials that each have an infinitesimally small probability of success. But in practical terms, use the binomial when there are a fixed number of "trials", and use the Poisson when you are counting things that have no upper limit. Here is an example of how the shapes of the distributions look different when the probability of success is high for the binomial:





Another thing you can see in this plot is that the Poisson distribution looks almost like a normal distribution. In fact, as the mean of the Poisson distribution becomes large, it becomes approximately equal to a normal distribution.

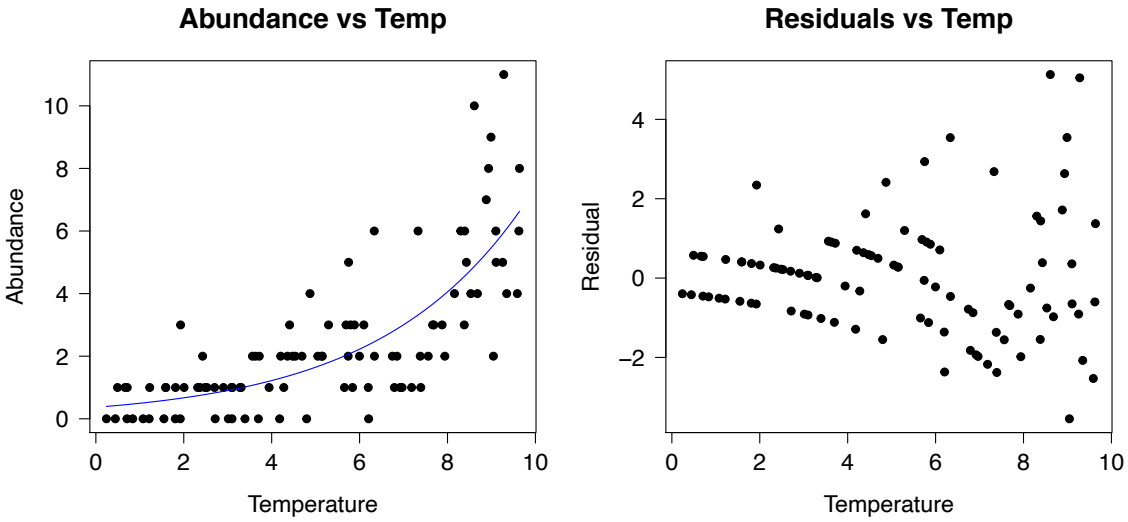


When the mean of the Poisson is small, the distribution is highly skewed, because it is bounded below by zero. As the mean increase, the distribution moves away from zero and becomes more symmetric. This means that count data where the counts are relatively large can be modeled with a normal distribution, which is often more convenient.

Too much variability

If life were easy, we could just think about whether our data are most appropriate for a normal, binomial, or Poisson distribution, and then move on to the next step in the analysis. Sadly things are often more complicated, and the reason is *too much variability*. We will discuss this extensively when we get to GLMs, but for now we'll introduce the mean-variance scaling for the Poisson distribution and show one way to work with excess variation.

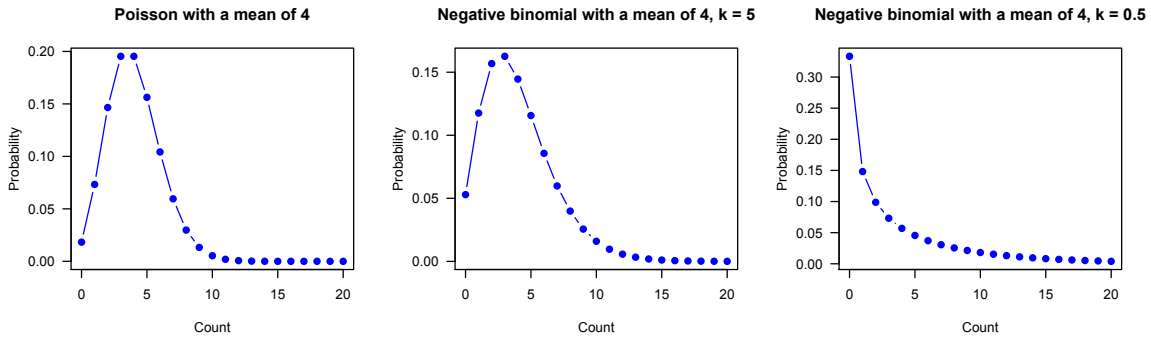
The Poisson distribution is defined by a single parameter, lambda. This is the mean of the distribution, but it is also the variance, i.e. *for the Poisson distribution the variance is equal to the mean*. It is intuitive that the variance and mean would be correlated, because the Poisson distribution is bounded below by zero, but has no upper limit. So when the mean is small the values are clustered around zero, but as the mean increases there is more room for the distribution to spread out. Here's a simulated example for a species whose abundance increases with temperature:



I've assumed that abundance is Poisson-distributed, with a mean that increases exponentially with temperature. On the right is the residuals from the fitted relationship, and they show a distinct funnel-shaped pattern of increasing variance as temperature increases. If we were assuming the data were normally distributed this would be a problem, but if we model the data with a Poisson distribution this is what we expect to see. We'll discuss this in depth later.

Problems arise because biological data often has other sources of variation that we aren't accounting for. For example, if we are measuring abundance in a number of locations, those locations probably differ in ways other than temperature that affect abundance. The result is that the residual variation will have more variance than the Poisson distribution predicts, which means the p-values from our analysis will be wrong (among other things). Having too much variation is called *overdispersion*.

There are several ways to deal with overdispersion, and for now we'll look at one way, which is to use a negative binomial distribution instead of a Poisson distribution. The negative binomial has two parameters (the Poisson only has one), and can be parameterized in multiple way. We will parameterize it in terms of μ, the mean, and an overdispersion parameter k. As k gets large, the distribution converges on the Poisson, while for smaller k the distribution has larger variance than the Poisson:



Note that the negative binomial examples have both more small values and more large values than the Poisson. Because the negative binomial looks like the Poisson but with more variance, it is commonly used to model overdispersed count data in ecology.