Discrete equations often arise as discrete variants of continuous equations. The reason for this is that many phenomena in nature are governed by continuous equations, such as ODEs and PDEs, while numerical analysis deals with numbers only at a finite precision. The process that takes a continuous equation and makes it discrete is called discretization. It involves a choice from a range of scheme called numerical methods. We will discuss several of these below

1. Different ways to discretize the first derivative

There are several different ways to discretize the first derivative. The most common of which is the Forward Euler method which states the following:

$$\frac{dx}{dt} = \lim_{h \to 0} \frac{x(t+h) - x(t)}{h}$$

This limit is a continuous operator that can only be approximated on a computer. If we choose extremely small h, i.e $h \approx 0$ or $h << 1$, then $\frac{dx}{dt} \approx \frac{x(t+h)-x(t)}{h}$

This formula can be used to discretize a continuous equation, or simply make a continuous equation discrete by swapping the differentiation with a difference equation. Note that the Forward Euler is said to be an explicit discretization, since the method predicts the value of x(t+h) given the value at x(t). Conversely if we want to predict the value of x(t-h) given x(t) we get the Backward Euler equation which looks like: $\frac{dx}{dt} \approx \frac{x(t)-x(t-h)}{h}$

This method is implicit since it goes back in time rather than forward in time.

Note the approximations in the equations above. Whenever you replace the derivative by a difference equation, you get a discretization error, which makes them no longer equal but still approximately equal. This error is O(h), which can be read as big O of h, or order h. In this case the error is linear, as it is a function of h. Therefore the smaller we pick our h, the smaller the linear error will become. However the smaller that we pick our h, the more time steps are requires to get from our starting time to ending time. This means that the runtime goes up, as the error goes down.

2. The logistic equation

In the discrete case the recursion form is:
$N(t+1) = N(t)(r(1 - \frac{N(t)}{K}) + 1)$
and it's difference equation form is:
$\Delta N = N(t+1) - N(t) = rN(t)(1 - \frac{N}{K})$
and it's continuous form is:
$\frac{dN}{dt} = rN(1 - \frac{N}{K})$
One obtain the latter (differential) equation from the former (difference) equation by discretizing. Using forward Euler, we get
$\frac{N(t+h)-N(t)}{h} = rN(1 - \frac{N}{K})$
which reduces to:
$N(t+h) = hN(t)(r(1 - \frac{N(t)}{K}) + \frac{1}{h})$
This is a discrete recursion equation for x with steps of h. For h = 1, the familiar discrete logistic equation is obtained.

If you wish to obtain the value of N at time $t = T = nh$ from an initial condition $N(0)$, then you timestep forward in time by computing N(h) from N(0), and then N(2h) from N(h) and so forth till you reach N((n 1)h) and finally N(nh).

3. Other methods

A more accurate method is the Runge-Kutta method (RK4):
$x(t+h) = x(t) + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$
Where:
$k_1(t) = f(x(t))$
$k_2(t) = f(x(t) + \frac{hk_1}{2})$
$k_3(t) = f(x(t) + \frac{hk_2}{2})$
$k_4(t) = f(x(t) + hk_3)$
It is easy to code and has a huge advantage in the discretization error. It is 4th order accurate meaning the error is h^4, so if $h = 0.01$ the error is 0.00000001.