GEORGE CHURCH: Is there a section? No, there are no sections for a few days. The sections, both the weekly sections and the extra sections, will be on the website. That's your best place to look. If you have questions after you've looked at the website, then you should contact your teaching fellow. But definitely look at the website. It will be updated daily, and your sections will be assigned probably by Thursday or Friday, OK.

OK, welcome back from the break. As promised, we're going to discuss the innards of computers that are going to be helping us with computational biology. We have a schematic diagram here, illustrating in part-- we'll have similar schematic diagrams for biological systems or biochemical systems. This is one for transistors.

We have these nonlinear transistor elements here with an input voltage and output voltage in a controlled voltage VDD, and a ground here, this little triangle in the lower left. These transistors are in this circuit here that is certainly a higher-level description. It allows you to put down the more detailed description of this voltage curve as a function of time.

So time is the horizontal axis here, ranging out to 200 nanoseconds, and this is all done in a program, a simulator called Spice. of this complementary metal oxide inverter. And this sort of simulator is one of the things that we will be talking about in biological systems. And it's very useful for designing these circuits, and you can see as the voltage, VDD, goes up in this straight blue line from 0 volts, which is off, to 5 volts, which is on.

You get this almost all or none, but certainly nonlinear response, where the output voltage on the vertical axis goes from 5 volts, on, to 0 volts, off. It's basically the opposite. When it's 0, it's 5. When 5, it's zero. But in between, you can see there's this gray zone. You want to stay away from this in digital circuitry. You want to stay fully saturated or basically 0 volts.

And then these inverters can be wired together in even higher-level diagram into what are called registers, which allow you to store multi-digit binary numbers. These can then be coupled together. So you can take two registers and add, digit by digit, the contents of those registers. That's called an adder. Adders and a variety of other higher-level electronic components can be put together and then addressed by software called a compiler.

Well, how did you get this software in there in the first place? We have to toggle the hardware until it gets into a state where you sort of manually get the transistors to be in the right set of on and off voltages, 5 and 0 volts. Once you get the first compiler, you can now work at a much higher level. A compiler is basically something-- all this code that I've been showing you so far, the Perl, the Fortran 77, the Mathematica, so forth, those are all things like, print x above 1.

That is like a compiler. It's a code which you can type in. It's almost English. Some of you may say it's not nearly close enough to English. But it's much closer than dealing with these little voltages, OK. And once you have one compiler, you can make another one. You can use that pseudo English to write a more complicated compiler that can deal with a even higher-level language, and then reduce it. Then it takes care of the bookkeeping, or reduces it down to telling the computer what voltages to put where and when.

As you go up the still higher ones, we have these high level application programs, which might have intense graphics or something like that, OK, so that you really get a world view that is much more in resonance with our primate visual and auditory senses.

Now, this idea of self-compiling and self-assembling is very interesting, very self-referential, as many of the things in this course will be. We have biological components which have these very interesting complementary surfaces that I talked about a couple of slides ago, where the two strands of DNA which are not covalently linked-- covalently means that you have these strong bonds along this one ribbon-- and connected by a series of stacking interactions, where plates, the sort of planar bases stack up.

And they form weak bonds from one strand to the other where the rules I mentioned before-- now slightly more realistic symbols here for the GC and AT base pairs rather than just the alphabetics. But this is still symbols. This is not really electron density, where you're using letters instead of the electron density of nitrogens, carbons, oxygens, hydrogens.

But these hydrogens make this hydrogen bond a weak bond, and the surfaces are complementary, so that if you try to pair a C with a T, you have the wrong spacing. You have steric clashes and so on. This process by which one sequence will not make the same sequence right away-- replication does not make an identical sequence. It actually makes a complementary surface.

And then one more cycle, and you get back to the original, just like that example, I gave before of the trinucleotide going to a hexanucleotide. And then that helps the second round. This is very analogous, except now these molecules, instead of being six nucleotides long, they can be hundreds of millions of nucleotides long.

So this introduction to minimal life, all these life have in common self-assembly, catalysis, replication, mutation and selection. The monomers, meaning the simple molecules taken from the environment, the environment being defined by some boundary-- this could be a fairly flexible boundary. It could be a membranous boundary. It could just be some kind of aggregate.

The small, simple monomers, these combine to make complicated polymers. And then this replication process continues. When we go to more complicated systems, now this is a central dogma. The DNA, the long-term stable storage, is accessed by making ephemeral RNA, which then encodes proteins.

And in some systems, RNA can replicate. RNA can be reverse-transcribed to DNA. Unfortunately, the AIDS virus is one of them that does this. DNA can replicate itself, and certain pathological cases, or actually certain cases, I should say, of proteins, can in an autocatalytic cycle recruit other proteins to their particular state. These are called prions, and they're based in Mad Cow disease and things like that.

But again, you've got a boundary for the replication, and simple structures come in and complicated structures are generated. When we talk about the polymers, we want to quantitate their amounts in their interactions. They initiate, elongate, terminate. They fold. They get modified in various ways. So it's not just simple linear polymers.

Their position in space is important, and they are either degraded or diluted during the replication process. This gets even a little more complicated when we talk about functional genomics. Here, we measure the growth rate. We measure the concentration of RNA and proteins. Sometimes, their localization, it's important to measure that too. That's called gene expression.

And the interactions are important to be part of this measurement process. These measures and models that I'm talking about are how we get to defining enough about living systems that we can model. So here is another model. This is a Rorschach test as you might take. You go into a psychologist's office maybe long ago, and they'd say, look at this inkblot. What does it remind you of? You know, bad things your father did, so forth. Here, the Rorschach test is, what does this curve remind you of. Give me some hints.

STUDENT: Exponential curve.

GEORGE CHURCH: Exponential curve, great. And how do you get-- and this is like the stock crisis before a dotcom crash. How do you get this? How do you find it in biology for the biologists, or how do you find it mathematically for the rest? Yeah.

STUDENT: It kind of reminds me of human population growth.

GEORGE CHURCH: Human population growth. Yeah, that's a good biological example. Maybe, and Malthus and others have said, this can't go on forever. But this is not a fact. It's a pretty solid speculation. So how do you get this? Well, for those of you who prefer to see or expect to see stocks go down as well as up, we've got an exponential decay curve in magenta. It sort of is a reflection of the exponential growth curve.

These are just e to the kt, or e to the minus-- or where k is positive. And this is the world's simplest differentiable equation. y, the y-axis, is a function of time, the horizontal axis. And the small changes in y with small changes in t, time, t, dydt-- that's sort of the slope of the y as it goes up, the blue exponential curve-- is related to how much y. The more humans are, the faster the human race replicates, OK. And it just keeps getting more and more. And that's why it has this exponential curve. This is much steeper than quadratic.

And its origins are way back here. It's similar to exponential decay, as you might get with radioactive substances. It follows a reverse process. And if you integrate this, you get a very simple integral. It's what we've been talking about. It's e to the kt. So it's an exponential function of time, y, say the human population or your stocks, where e is this number that we highlighted before, about 2.7.

If you're interested in half life, which sometimes people are, like radioactive decay or half-life of replication of bacteria in a solution, it's a very simple formula that gets you from the rate constant k. This is like a biochemical rate constant to a half-life. This is growth and decay.

So what limits this? Why doesn't it just keep going up? What we've been looking at is the lower left-hand corner of this graph in slide 29, where it goes exponentially up from close to 0, not quite. And eventually, it will plateau, or worse yet, it might come down. And what causes this plateau is exhaustion of resources. And if you get enough accumulation of waste products, or enough exhaustion of resources. You can plummet.

If you just zoom in on this little part here, one way of analyzing it, very hopefully known to some of you, is that you take the logarithm of y and plot it versus t. So t is linear axis, and the vertical axis is logarithmic. Now you get a straight line, at least for the beginning here. And eventually it will plateau the same as this one does. And that's a way of telling that you have a simple exponential.

If you have e to the t power, or 2 to the t power, or anything to the t power, simple. Those are all simple exponentials, and they'll give a line when you take the logarithm. Now, what does Mathematica do to help us here? You set up this equation. Instead of saying dydt, you could say, y prime of t. That's just shorthand. It's very commonly used in calculus.

y prime of t the first derivative of t, of y with respect to t, is directly proportional to y. That is to say, your slope of the human race expanding is directly proportional to y, the number of humans, OK. And then you're going to start at time equals 0. We've got one human. Well, that's probably not enough. Well, OK. Maybe a bacteria. OK.

You have initial conditions, OK. And then so you just say solve it. You tell the computer, so everything to the right of this equation is something you can type in to Mathematica. Do a differentiable equation solve of this string that I typed in here, which tells you the initial conditions and the formula. And boom, out comes the out. You didn't type this in. Mathematica came up with this, e to the t power.

That's pretty cool. And even though it's the world's simplest differentiable equation, it solved it. Try to do that in your other favorite programming languages, Excel, or Fortran, or Perl, or Python or whatever, C. This is really powerful. Now, this is analytic or symbolic or formal. These are various terms you would say for this trick. And as the equations get more and more complicated, this becomes more and more amazing, almost intelligent.

Eventually, they get complicated enough that neither humans nor Mathematica can solve them. And so what you do then is use a numerical approximation where you take little steps and you solve it by numeric approximation. But you set it up the same way. You tell it that the derivative y with respect to t is proportional to y. Or in this case, proportionality counts as one.

Same initial conditions, one starting bacterium. But now you tell it what interval you want to do this. You don't want it to have to do these little steps all over, all negative deposit of infinity for time. You just want to say, I'm just interested in time from 0 to 3 minutes or hours or whatever years, OK. And then you evaluate it, and you can plot this, which appeals to the primate visual system, these plots. And you'll see lots of plots in this course.

But now here, y, a function of t, is this exponential curve. And if we plotted log of y as a function of t with this numerical solver, it would be a straight line. Now, I give you some where it isn't a straight line. These are all logarithmic on the y-axis, and they're all linear on the horizontal axis. And they're all time on the horizontal axis, linear time.

And they're more than simple exponentials. They're going up faster. Rather than going out slower, which we think human population, bacterial population and so forth, they'll go up linear on a log plot and then flatten out, these things are just going faster and faster. What are these things? Well, even though your dotcoms didn't work out, if you had had a stock portfolio in Western European commerce in the year 1000, you'd be in really good shape now in the year 2000.

This is not only going exponential, it's going steeper than exponential. And we all hope that this will keep going forever, that the gross domestic product of people in Western Europe and the world will just keep going up. And this is due to technology. And technology keeps reinventing itself. And hopefully, it can keep doing that.

Here's another example . This is more drilling down to specific technologies that have been on a superexponential or hyperexponential-- I don't know quite what the right term here is-- for a long time. These are greater than linear, steeper than linear. These are close to quadratic. And so it's an exponent of a quadratic.

And these are for transmission rate in pink, of data from the Morse code in the 1830s to optical fibers here in the present. And then the blue are digital processing from the first census in the 1890s to modern computers. And this is in instructions per second per $1,000.

Now, this unit, the little piece of this, these integrated circuits that Moore's law refers to from just the tiny end of this from 1965 onward, refers to integrated circuits. And these will run out of gas pretty soon, everybody tells us. But this curve may not, because it goes beyond it. It predates integrated circuits and it will post date them. And who knows where this leads.

STUDENT: Question.

GEORGE CHURCH: Yeah.

STUDENT: What's the r squared?

GEORGE CHURCH: Oh, sorry. We'll get to that at the end of this lecture, but it's a correlation coefficient, which is, it's to what extent is there a fit between one curve and another. How well does the calculated curve fit the observed data collected? And so these are around 99, which is very good correlation. And it's better than the linear plot, but of course, you have more adjustable parameters.

OK. Another sign of hopefulness is data are coming in faster. So our life is getting better. Our computers are getting faster, and data is coming in from the Genome Project. And this little inflection point, where it was log linear for a while and then a new log linear-- so overall, it's superexponential. And this is for the number of base pairs we can get per dollar, starting with transfer RNAs in the late '60s and ending with who knows how many human genomes that we will have by the year 2010.

Now, where does this all this exponential growth go? Some people think we will be creating computers that are smarter than we are soon. What would this require? Here's a nice back of the envelope example of systems analysis where biology meets computers. Let's analyze our retina. All of our retinas are processing right now, hopefully.

And Hans Moravec simulated a retina for video imaging where he did edge and motion detection, and it required about a billion instructions per second to match the 10 times per second which you're updating the retina. The brain is about 100,000 times larger than the retina, and if this scales linearly, which is speculative, then you need a computer that has about 100 million MIPs or about 10 to the 14 instructions per second of compute power, and a similar number of bytes.

Now, back in 1998, that was still quite a ways away. But here in 2002, the best supercomputer-- and this site keeps tract of the top 500 supercomputers. And trust me, your computer is not on that list. But anyway, the top one is within a factor of 10 of this compute power. Now probably, Earth's scientists that own this thing will not bother to try to see if it can do ordinary human things like watching soap operas. But we're in that range. We need to be cognizant of the possibility.

Here's another model. I've tried to put it in the same units we've been talking about, this exponential growth. Again, we have the rate constant k. We have the y, the human or bacterial population growing exponentially. And here now, we try to model the case where yes, as you have a great population size, it means greater growth until it gets close to the maximum carrying capacity, the 100%, the 1, the maximum it can go. And then it will plateau. So you have a plateauing near 1.

And this is called the logistic map. It is the basis of that complexity calculation we talked about earlier. And here, the population size is a function of rate constant, and it has both grows when y is small. As y scales up, it goes up exponentially. And then finally, as it approaches a maximum of 1, it plateaus.

However, if you get greedy, and you increase your growth rate beyond, say-- here, it's very, very small, very ungreedy, just 1.01. That's like a 1% interest in your bank account, OK. But still, you'd grow exponentially given enough time. However, you get greedy and say, I want a 300% return on my investment. Well, then you start getting these little cycles of like stock market going up and down, OK.

And if you get really greedy where you need a 400% improvement each cycle, you get chaos. And then you can eventually drop down very close to 0 and crash, and the population can go extinct because it used up as resources or made non-optimal use and maybe toxic side products.

OK, graphs. We have directed acyclic graphs. Just as an example, graphs are made up of nodes. You can think of these here as, the nodes are people or organisms. And you start with one bacterium here on the far left-hand side of slide 35. And you have a direction. You can only go forward in time.

So the node is the bacterium individual, and the lines connecting them are edges in the graph terminology. And they're directed. And they can't go in cycles because you can't have a daughter giving rise to a mother, OK. So this all makes intuitive sense. But you can use these kind of graphs for a whole variety of interesting things. You have not just the pedigree we talked about, but phylogeny in general, ancient connections between organisms. The biopolymer backbone, you have a simple linear backbone or a branch backbone. This can be represented. It doesn't covalently cross back on itself.

If you want to know what's near one another as this polymer folds up, like that transfer I showed you in the first slides, those contacts are indicated. Now you start getting cycles, because A can contact B, B, C, D, back to A again. You get cycles in a three-dimensional structure. You get cycles in a regulatory network. You can have, in order to maintain homeostasis in your body, A can regulate B. B can regulate C, and back to A again. But that's all directed.

There are system models that we and others will study. They have in common-- this is slide 37 on the left-hand side-- the system models. And they've been chosen mainly because in the pre-genomic era, it was very hard to get data sets. Certain systems were just technically easier to get large data sets, genetic or biochemical.

These include E. Coli, going toward food and away from toxins. Red blood cell is a nice metabolic system because it doesn't have any polymer synthesis. Makes it simpler. Cell division cycle is really key for understanding pathogen replication, cancer. Circadian rhythm, a huge number, many if not possibly all organisms, have some circadian rhythms that keep their biochemistry optimal, and keep us hopefully awake, right now, anyway, until it's time.

OK. Plasmid DNA replication is an example of single-molecule precision. And we'll talk about the DNA single molecules in just a moment. So that's where we're aiming right now, is from graphs and pedigrees down to the single molecules that allow replication to work. This replication is achieved by interconnected machines they are somewhat modular. "Modular" is also a computer term where you try to put code that works together into something that's defined spatially and functionally.

So you have these little modules that replicate the DNA, make RNA from it. A different module does the protein synthesis. There is some interconnection between these. This will be discussed. these kind of complicated machines that biologists love to simplify in diagrams will be described in more detail next week.

But the idea is this idea of modules versus extensively coupled networks. This is how we get the replication. The way we analyze the replication is somewhere here in the middle, where I've had a scale here that goes from high resolution, very accurate descriptions of physical processes sort of in the nanometer femtosecond range on the far right-hand side of slide 40, to things that are very long timescale, very large scale sort of kilometers, years that happen in population dynamics, sometimes global population dynamics.

You should understand that all of these models we'll be talking about are approximations. As we go down the scale, it gets more and more computable to compute more and more complicated things, but at the cost of greater and greater approximations.

Even the molecular mechanics that we use in conjunction with crystallographic diffraction data is amazing computational chemistry, but it's a great approximation of quantum mechanics, which in turn is an approximation of quantum electrodynamics, which itself is an approximation. And all of these things are very hard to compute on any even reasonable atomic system multi-body problem.

The big approximation for molecular mechanics is that you have spherical atoms, so you don't have the distortion of the dipole that occurs in very useful bonds, such as the hydrogen bonds and almost all non-bond interactions. That's poorly approximated, but it's the best we have that can be computed right now with most computers and even modestly large molecules.

Then now, as we go down, we can think of it as higher and higher-level abstraction. Just like high-level programming languages, we're now programming chemical systems and thinking about them. Now instead of dealing with single atoms in molecular mechanics that produced the tRNA structure that I showed you, we now deal with that whole tRNA as a single molecule. But it's still a great depth of precision, because each molecule has its own life, and you track each one on the computer. And that's stochastic simulation.

The next higher level beyond that is now, we don't deal with single molecules. We deal with populations of molecules, or we deal with a concentration as a function of time. The ordinary differentiable equations that we've already been talking about, like that exponential growth curve, is one way of dealing with concentration of bacteria as a function of time. That's appropriate.

There are other cases where we want to do this optimization. We want to study how close to optimal a system is. One way to study that is with these economic functions, these linear programming, to look at the fluxes. Now, we're no longer talking about concentration and time, because we're interested in the rates through which chemicals are flowing through a biological system, where any particular chemical concentration is at a steady-state level. And that means you have things going in and things going out, but stuff in the middle is staying the same.

That's a very useful approximation. It's used time and again. Even though these are dynamic systems, you can find them. In a pseudo-steady state, you can apply these very powerful computing tools. And then you can do computations that would be very hard to do in these more precise and complete methods.

And we'll go through these in much more detail later on you'll find very interesting connections between the larger-scale things we're talking about, where we talk about stochastics of whole organisms in big populations, and the stochastic of single molecules.

OK. We're talking about single molecules. This is our last topic today. And each of you do single-molecule manipulations on a regular basis, and your ancestors have been doing it for $10,000 years without a license, without a computer. And they've been doing a pretty good job of it. They've taken this little weedlike thing, teosinte, and turned it into this corn that would make the 4th of July quite proud. And dogs, who knows what their ancestors looked like. But right now, they span about three logarithms of mass.

And this was all done with the awesome power of single DNA molecule technology, crosses basically. And what happens in each of these cells in your body if all goes well is you start out with one chromosome of interest. And it divides, and then the cell divides.

And that chromosome-- we'll forget about all the other chromosomes in there for a moment-- that chromosome has a choice of when it divides, it can go one each into each of the daughter cells. Or both chromosomes can hang out together, since they're all tangled or something, and then one of the daughter cells doesn't get any copy of that chromosome. That obviously is not a good thing.

Even if two copies of the chromosome is OK, if you can tolerate that extra dosage, you certainly can't tolerate zero chromosomes. Well, what's the chances this will happen? Well this is really elementary probability. I'm going to ease you into it. It's that it's about 50/50 chance. These are all. This is the exhaustive list of the possibilities. And about 50% of them have the wrong dosage.

Well, what if we have a more realistic situation, human cells with 46 chromosomes? What are the odds that they'll all be right, that we'll get exactly one of each? We have 23 chromosomes from mom and 23 from pop. What's the odds that this is going to work out? Well, we're going to take a couple of slides to get to that answer.

But first, to motivate you, this is extremely important in a health care sense. It's the most common form of mutation, happens all the time. Unfortunately, at every chromosome, duplication or losses is a big change in the human state. And the mildest of all of the additions or subtractions of a chromosome-- here, you just have three copies of chromosome 21, everything else normal.

So 1.5 dosage of one of the smallest human chromosomes has an enormous impact. Most of you have seen someone with Down syndrome, which is severe mental retardation, and heart defects in various other organs.

STUDENT: [INAUDIBLE].

GEORGE CHURCH: Yeah, question.

STUDENT: This problem that you just described, in reality, though, it's not random, because there are mechanisms in the cell that would bias the tool in order not to segregate one cell.

GEORGE CHURCH: Right. This is a good point. This is what I'm setting us up for, exactly that conclusion. It cannot be random. Single molecules are subject to stochastics. And so to overcome that stochastic process that should be random, you have to have machines that involve multiple molecules, because only through multiple molecules can you get the statistics to overcome the single molecule.

And that's quite a trick. You can't just casually say, oh, there's some machine in there that takes care of this, OK. Just to expand this a little bit more, we know that certainly, the DNA is the case where the single molecule is always a problem. It has to be aided by molecular machines where you use energy. You expend energy in order to make sure the DNA molecules work.

We'll get back to that calculation of what the odds are at random, but I should say also, RNAs in many systems appear to be, on average-- remember, it's the population average to be close to 1 molecule per cell. They're produce in bursts, following stochastic bursts of RNA where you get transcription factors binding. And then that burst of RNA produces even bigger bursts of proteins. But on average, it comes out to be a very small number, because the proteins preserve. They persist through many cell divisions while the RNAs maybe can turn over more rapidly.

To get back to that question of how much variation is tolerable in biological systems, here's the very beginnings of your statistics. Some of you may have this hopefully already. There will be sections where you can cover this. But here are some of the really, really useful, easy statistics. What do we want to know about a distribution?

Making the fewest assumptions for now, we want to know its mean, its arithmetic average. What is the average number of chromosomes in a cell? If they're supposed to be 1, how close to the average are they? That's the variance. To get the arithmetic mean, you basically add up all the numbers and divide by the number that you counted. Add up the values which are the axes.

And so here you take a weighted sum. The sigma means the sum of the x values weighted by the frequency, f of x, the frequency that they occur in the sample that you take. Here, r for the mean is just 1. It's taking the first moment to get the mean. The analogous thing is, now you correct all the values that you're measuring, these variables, the number of chromosomes, say. And you subtract the mean.

So now the mean for this x minus mu is now effectively 0. The mean is 0, and you want to ask, how far from 0 do you deviate. For chromosomes, you want that deviation to be very small. You want the variance to be very small. And it's just the sum of the squares. We want to take the squares because if it deviates either more or less, it's still a tragedy, and you want to keep tract of that.

So these are two things. They don't make any assumption about what kind of distribution. The distribution can be anything. You can calculate the arithmetic mean and the standard deviation. Another useful concept that starts to make more assumptions in interpretation is, now you have two variables, say x and y. And you want to ask, do they covariate. Are they related to one another? When you do two different experiments, you want to ask whether they're giving similar results.

If they're two completely different kinds of experiments, you might want to know whether they're reinforcing each other. You want to know whether they're redundant. If you're observing two biological facts, you don't know whether they are related to one another. It's a discovery if they covary. That's what this means.

Covariance is, again, using this concept of expectation. The sum of the x's corrected, so that you subtract the means, their averages, and divide by their standard deviations, square root of the variance. So you basically end up with a mean of 0 and a variance of 1. And then whenever these normalize, when x goes up and y goes up, the product will be reflected in this sum.

So now this has an interesting property that when x and y are independent, unrelated, then the C or the Pearson correlation coefficient is 0. However, the reverse is not true. If C is 0, it does not imply that the x and y are independent. An example, a simple example, is the curve, the quadratic curve, y equals x squared.

Here, they are completely related to one another, but they give a correlation coefficient of 0. That's because this is a linear correlation coefficient. The model that you're testing is that they are linear related. They are either positively correlated, which means the extreme case of C will be equal to 1, or negatively correlated C is equal to minus 1.

You can plug this little formula here, handy-dandy practical form you can plug in to calculate probabilities. And this is an Excel formula. The probability that a correlation is far from 0, that's dependent upon the sample size where you sampled different x's and y's. You know, it could be head size and weight, or length and weight, and so forth.

If they're correlated, then this probability will be significant if your sample size is large enough. There are some very practical things. And now let's put these in the context of a particular class of distribution. Now, most of those did not require that we state what kind of distributions, but there's a big interesting set which are roughly bell-shaped curves.

And I've rigged this so that these three wildly different types of distribution happen to give similar curves. And you see in the next couple slides how I rigged it, but basically, this is a normal distribution, the Poisson distribution and the binomial. The binomial has a limited range. n goes from 0 to 40 in this case. It has a maximum n of 40.

The Poisson has a mean which in this case is 20. The normal distribution has a mean that's similar. The normal distribution can have any range, the standard deviation. And this is set here to be, the standard deviation to be, the square root of the mean of the Poisson distribution. That's how you can rig these to be similar to one another.

I think time is not going to permit me to go through all this in detail, but you will cover these in your statistics sections if you don't already know it. But suffice it to say that the binomial distribution is limit x. It Has to be an integer, and the integer is limited to going from 0 to n. This distinguishes it from a Poisson distribution, where it goes from 0 to infinity, and the normal curve, where it can go from negative infinity to positive infinity. Binomial and Poisson are discrete. They happen at integers while the Gaussians are continuous.

The way you calculate this is, the probability of an individual n happening has probability p, so say 0.01 in the previous slide. And then getting exactly x of those is p to the x power as a first approximation. But there are actually two cases here, hence the name binomial. If it were multiple cases, they'd be multinomial. But the two cases are basically p and 1 minus p. They have to sum to 1. All these probabilities have to sum to 1.

So now you have the probability that you have exactly x, and the leftovers go into 1 minus p. But then you also have to correct for the number of different ways that you could get this, the number of combinations, which is the total number of possibilities choosing x at a time. And then again, the leftover is n minus x. And this is n factorial over x factorial times n minus x factorial. This is the number of combinations.

And so now the binomial distribution is this, and the sum of all the terms here has to add to 1. That's one of the properties of probability distribution, is if you think of all the possibilities, they add up to 1. So the sum of the binomial distribution of all the x's is 1. Now, just to remind you that computers are fallible, here's what, when you take a fairly-- you know, x is equal to exactly 300 taken from a population of 700. A probability of 1 for a unit event.

The probability of getting exactly 350 from that kind of bell-shaped curve is very small, but not 0. And Mathematica gets it right, and Excels gueses it at 0. Good guess, but wrong. Poisson, you now can and must go out to positive infinity. And there, you often will make the approximation that for large n and small probability in the binomial we've been talking about-- n is the total number of objects you're looking at, you're choosing x from. And t is the unit probability of each of those.

The mean is now approximately n times p, and the binomial and Poisson are very similar. That's why they look similar in that plot. And here's some practical magic you can do with the Poisson. If you have a library of [INAUDIBLE] or combinatorial chemistry, or genomic clones and so forth, and x is a number of hits, and you want that to be greater than 0 so your thesis can proceed, you want the 0 hit term to be very small, e to the minus mu.

So you want the mean to be greater than 1 or 2 or maybe even-- if the mean is 10, the probability that for a given experiment you'll have a 0 hit is very small. And you can estimate this from the number and 1 2 and 3 hits you get. You can estimate the 0 hit, and you can estimate whether it actually fits a Poisson or not.

The final of this trio is the normal. Now you go from negative infinity to positive infinity. It's not just 0 to n or 0 to infinity. And it's continuous. That means everything takes a p. So now instead of summing up to 1, you integrate up to 1, because now the little delta x's are infinitely small. And so now, here's an exponential of a quadratic, just like the ones we were talking about earlier. So this is a negative quadratic. And that gives you a nice little bell-shaped curve. And the 2 pi sigma square root is a normalization, so it does actually integrate sum to 1.

Here, another approximation sometimes applied is when n times p times q is large, the normal is very similar to the binomial. People will abuse this and use one of these three distributions in the place of the other when it isn't appropriate. And we'll give some examples as we go.

So back to this calculation. If we apply the binomial that we had in those previous slides-- and I urge you to do this as an exercise. It's not on the problem set, but just do it. Just getting some for any 46 chromosomes, you get the right number of chromosomes, is about 8%. That's not too bad. But it's still fairly lethal, because you really want exactly the correct 46, which is 0.5 to the 46th power, which is infinitesimally unlikely to happen at random, which gets back to the point made by the audience here that this is not random.

But you can't fight the fact that single molecules are based in stochastics. So you have to have a lot of events adding up, energy being input to overcome that. We have selection that's optimizing this over long periods of time. We can use random numbers that underlie this for the simulations of these stochastic events, and also for permutation statistics, that when you have some data and you want to know whether it's significant or not, you can kind of do a Monte Carlo simulation of it.

Here's how you code it in a couple of different languages, Perl, Excel, Fortran 77, Mathematica. Even though you can't evaluate it by looking at these numbers on the screen, trust me. There are bad random number generators. There are random number generators which are not very random, OK.

Where they come from is this remainder operation, operated on very special numbers. You'll have to look these up in this reference to get a full feeling for it. But these are deterministic formulas that give you random numbers. It's not really the same as flipping a coin. The computer actually will give you-- It's the same random numbers over and over again unless you do something very special.

And typically, these give a uniform distribution between 0 and 1, or over some integer range. And then you can turn it into a normal distribution with this kind of trick here where you make a transformation. There's a difference between a uniform random distribution and a bell-shaped one. And you can generate both of them just with this slide alone.

So we full cycle back to these three different bell curves. They have very different properties that can be applied. Binomial only when you have a limited range, Poisson when you're positive infinity. Normal, negative, positive and continuous. Thank you for participating in this. These are the topics that we covered. See you back here in a week. Please hand in your questionnaires, and the sections should be assigned by Thursday or Friday.

STUDENT: Thank you.