Groups:

Group 1:
Amelia Camino
Jake Magee
Amanda Muratore
Julissa Sarmiento

Group 2:
Finian Ashmead
Francis Burk
Mykola Chernyashevskyy
Mohamed Ismail
Ehteshamul Karim

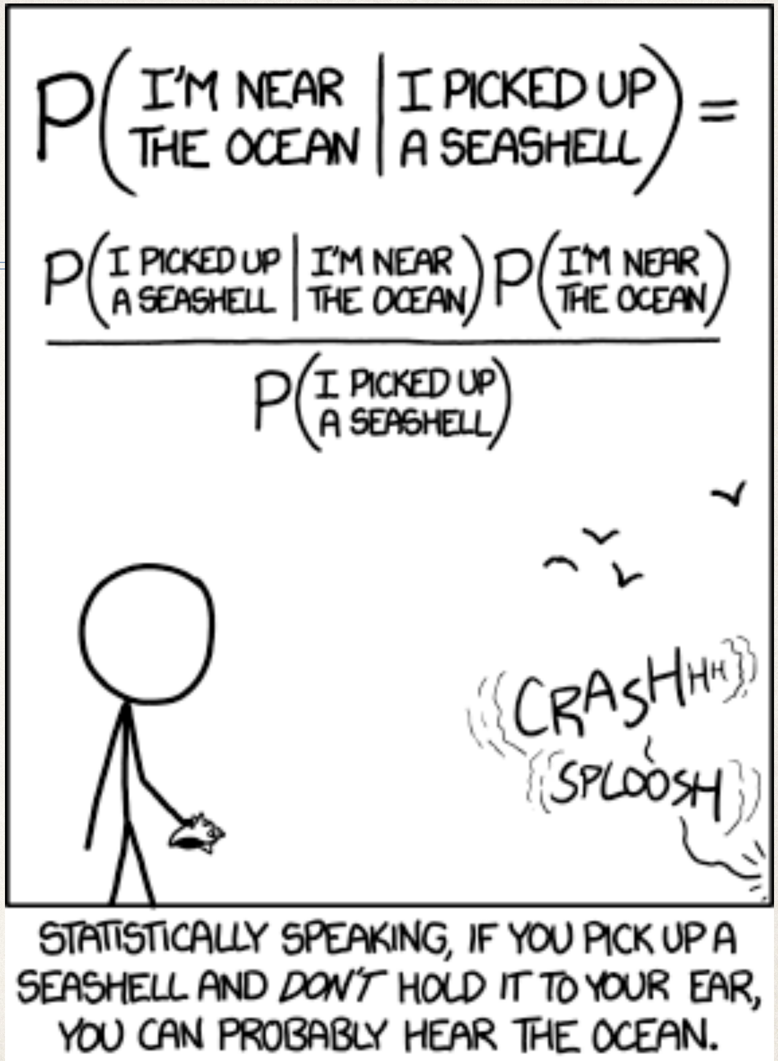
Group 3:
Cullen Abelson
Alia Dawood
Mira Salman
Yunchong Zhang

Group 4:
Nathalie Chicoine
Lauren Elicker
Yoki Salcedo
Marcos Tamargo-Arizmendi

Probability Distributions

Statistics and Data Science

Spring 2025



http://xkcd.com/1236/

Goals for today: you should be able to...

- Explain the Bayesian definition of probability
- Apply Bayes' theorem
- **Lecture 5 notebook:** Explain what the binomial distribution is and apply it
- **Reminder: Homework due Friday!**

Review: the Frequentist definition of "probability"

- In a well-controlled situation, we can define the *probability* of an event as the fraction of times it will occur if we infinitely repeat an experiment - i.e., its *frequency*.

$$P(x) = \lim_{n\to\infty} \frac{n_x}{n_t}$$

Another view of probability

- We've seen that in the frequentist view, probability is just the fraction of times an event happens out of an infinite number of trials.
- We can get very similar results with a very different definition: if we take probability to represent our ***state of knowledge*** that something will occur or that something is true; sometimes called its 'plausibility'
  - It is possible to define logic so that not just 0=false and 1=true, but values between 0 and 1 work too.
  - Alternatively, in some formulations, probability is taken to indicate our degree of belief.
- These ideas were formalized in the ~1950's, but are related to methods developed by Rev. Thomas Bayes in the 1700's; as a result, this is referred to as the ***Bayesian*** definition of probability.

Applying this concept

- Consider flipping a coin again.
- If we know it is fair, then we would expect that one-half of the time we will get heads, and one-half of the time we will get tails.
- So we would assign the same probability to the two events: 1/2 - and so would, presumably, any other person.

Odds

- Another way of expressing probability is as the *odds* on an event: the probability it occurs, divided by the probability it does not occur (i.e., $\frac{p}{1-p}$)
- So the odds of getting heads is 1 to 1 (or 1.0) .
- The odds of rolling a die and getting 2 would be 1 to 5 (or 0.2): it is 5 times more likely we roll something besides 2 than 2.
- This is the inverse of typical gambling odds; e.g. something that happens one-fourth of the time would "pay 3 to 1".

What role does belief have in science?

- Many people are uncomfortable with the Bayesian view of probability:
  - Our goal as scientists is to be impartial judges, right?
  - Different scientists might have different beliefs about what the probabilities might be - in the Bayesian view there may not be a single possible probability (vs. the frequentist view), but instead each of us has our own probability for event X!
  - ☑ However, we often think about data in Bayesian ways (e.g., what should I conclude about the true magnitude of an object if I observe that it is 20+/-0.1)?
  - ☑ In practice, using the Bayesian vs. frequentist definition can make little difference - e.g., the probability on a coin flip is still 1/2 in each view.

Many things work out the same in both Bayesian & Frequentist views

- R. T. Cox showed that if a definition of plausibility follows some simple assumptions - e.g. *p(A is false) = 1-p(A)*; if *p(A) > p(B)* and *p(B) > p(C)* then *p(A) > p(C)*; and if plausibility depends only on the information received, not order - it will lead to the "Kolmogorov" axioms of probability theory:
  - Any random event has probability p between 0 and 1
  - An event that is certain to occur has p=1; the total probability that some event occurs is 1
  - If A and B are mutually exclusive (i.e., both A and B cannot be true simultaneously), then p(A OR B)=p(A)+p(B)
- These axioms allow us to manipulate probabilities, define how they combine, etc.

Independence and conditional probabilities

Commonly treated as (**important**!) definitions:

- ➡If events A and B are ***independent*** - i.e., whether A is true has no relation to whether B is true (i.e., what we know about B doesn't affect what we expect for A) - then: $p(A \text{ AND } B)=p(A) \times p(B)$
- ➡ Many things aren't independent. For instance, if it rains today, it is more likely (than if you averaged all days) that it will rain tomorrow. The ***conditional probability*** that A will be true, given that B is, turns out to be: $p(A|B)=p(A \text{ AND } B)/p(B)$
- ➡Note that if we combine these, we find that $p(A|B)=p(A)$ if - and only if - A and B are independent.

Manipulating probabilities

- If events A and B are mutually exclusive, then *p(A OR B) = p(A) + p(B)* .
- What if they aren't? In general, *p(A OR B) = p(A) + p(B) - p(A AND B)*
- For instance, suppose it is 50% likely that when a particular couple have a child it will be a boy, and 50% likely that any child they have will have red hair.
- Since these are independent, we calculate *p(red-haired boy)* = ½ x ½ = ¼ , and *p(red-haired OR boy)* = ½ + ½ - ¼ = ¾ .
- We could have figured this out by counting equally probable cases:

boy, red hair	girl, red hair
boy, brown hair	girl, brown hair

Marginalization

- Let's suppose there are a finite number of possible results for event B, $\{B_i\}$. E.g.: if we flip a coin, B='the coin came up heads' can be true or false, which we could call B_0 and B_1.
- It's possible to show that:

$$p(A) = \Sigma_i p(A \mid B_i)\ p(B_i) = \Sigma_i p(A\ \ AND\ \ B_i)$$

- We call this ***marginalization***: we have probabilities for A & B, but can use those to just get out probabilities for A.
- This can be extremely useful: e.g., if we want the probability distribution of the cosmological parameter Ω_m , irrespective of the value of another parameter *h*, when we know $p(\Omega_m\ AND\ h) = p(\Omega_m, h)$, we can get it from the continuous version of this:
  - $p(A) = \int p(A \mid B)\ p(B)\ dB = \int p(A, B)\ dB$

Example

- Let A= 'we roll 5 on a die', B_1='the roll of that die is even', and B_2='the roll of that die is odd'.
- Does $p(A) = \Sigma\, p(A \mid B_i)\, p(B_i)$?

- We know p(A) = 1/6. If the result of a die is even, A is impossible, so $p(A \mid B_1) = 0$; while if the result is odd, A will occur one-third of the time (out of the equally likely possibilities 1, 3, 5), so $p(A \mid B_2) = 1/3$.
- It is equally likely that the roll is even or odd, so we'd get:

$p(A) = (0)(1/2) + (1/3)(1/2) = 1/6$,

- as expected!

An astronomical example

- Suppose 40% of early-type (elliptical or 'lenticular') galaxies have AGN (actively accreting supermassive black holes), and 10% of late-type (spiral/irregular) galaxies have AGN: i.e., $p(A \mid E) = 0.4$, $p(A \mid L) = 0.1$. Further assume that galaxies are an even mix of early-and late-type: $p(E) = p(L) = 0.5$.
- **What is the probability that a randomly-chosen galaxy harbors an AGN?**
- Try to reason intuitively first, and then apply the formula:

$$p(A) = \sum p(A \mid B_i)\, p(B_i)$$

If you need a calculator, use python! e.g.: `0*(1/2)+(1/3)*(1/2)` for the previous problem...

Summary: key rules of probability to remember

- If events A and B are ***independent*** then: $p(A \text{ AND } B) = p(A) \times p(B)$
- The ***conditional*** probability A is true, given that B is true:
  $p(A \mid B) = p(A \text{ AND } B)/p(B)$
- $p(A \mid B) = p(A)$ if - and only if - A and B are independent.
- In general, $p(A \text{ OR } B) = p(A) + p(B) - p(A \text{ AND } B)$
- ***Marginalization:*** $p(A) = \sum p(A \mid B_i)\, p(B_i)$

Bayes' Theorem

- An important result comes from setting:
  $$p(A \text{ AND } B) = p(B \text{ AND } A)$$
- so:
  $$p(A \mid B)\, p(B) = p(B \mid A)\, p(A)$$
- so:
  $$p(B \mid A) = p(A \mid B)\, p(B) \,/\, p(A)$$
- This last statement is known as Bayes' Theorem, after its discoverer (in the 1700's).
- Despite having Bayes in the name, it is equally valid in both Bayesian and frequentist views of probability.

Breaking it down

$$p(B|A) = p(A|B)\,p(B)\,/\,p(A)$$

- Let's let:

  B = the true value of some set of parameters (some or all of which we want to know), and

  A = the observed set of data

- Then we call:

  $p(B|A)$: the ***posterior probability*** : i.e., the probability we'd conclude for B after applying Bayes' theorem

  $p(A|B)$: the ***likelihood*** : i.e., how likely is it we'd get A in scenario B

  $p(B)$: the ***prior***: our guess at what the values of B might be, in the absence of experiment A

What about $p(A)$?

$$p(B|A) = p(A|B)\,p(B)\,/\,p(A)$$

- You may notice that the book doesn't really talk about $p(A)$ (which is sometimes called the "evidence").
- That's because it doesn't matter in many calculations - it's basically a normalization factor.
- We could construct it, though, using a definition we encountered before:

$$p(A) = \Sigma\, p(A|B_i)\,p(B_i)$$

i.e., marginalizing over all possible values of B_i.

- The evidence is sometimes used to compare the effectiveness of different models in describing data.

How does this relate to Bayesian probability?

- Let's take probability to refer to our level of belief.

- Then Bayes' theorem tells us how to **update** our beliefs based upon some set of observed data. The prior in fact represents our prior beliefs about the possible distribution of values for B.

$$p(B \mid A) = p(A \mid B)\, p(B) / p(A)$$

Bayesian vs. frequentist analyses

- Frequentist calculations often focus on how often we would get the observed result, given some presumed true situation (=hypothesis).
- A Bayesian calculation would focus on how probable we find different possible true situations to be, given the observed result.
- Notice that, if $p(B)$ and $p(A)$ are constant, we just have:
  - $p(B \mid A) \propto p(A \mid B)$
- In many cases, the inference will be same whether we work from a Bayesian or frequentist perspective!

So how do the two views of probability differ?

- Much of the difference is not in whether they accept Bayes' theorem - but in how seriously they take it.
- Bayes' theorem requires a prior - but assigning a prior generally is a subjective choice (there are some rules of thumb).
- In many cases, the choice of prior doesn't make much difference.
- There are problems that are only really solvable in the frequentist view, and others that only work out from Bayesian assumptions.
- It is often most obvious how to pose a problem in the Bayesian view, so we'll generally be following that.

Why has Bayesianism become more common recently?

- Although much of the framework was developed at about the same time as classical statistics, Bayesian methods fell by the wayside.
- This is mostly because it is computationally harder - we have to integrate over more complicated functions (thanks to the prior), which may not be Gaussian.
- These days, numerical integration can handle almost arbitrarily complex scenarios easily.

Back to our example case

- Again, let's suppose 40% of early-type galaxies have AGN, and 10% of late-type galaxies have AGN. Further assume that galaxies are an even mix of early-and late-type.
- You find a particular galaxy in an X-ray catalog, letting you know it's an AGN.
- Is it more likely to be an early-type galaxy or a late-type galaxy? If someone bets you $10 that it's a late-type galaxy (so you win $10 if it's early-type, and lose $10 if it's not), would you take the bet?

- **Discuss with your groups!**

$$p(B \mid A) = p(A \mid B)\, p(B) / p(A)$$

Bayesian (and frequentist) techniques can be applied to problems well outside of science

- Go to: https://web.archive.org/web/20201009065503/https://election.princeton.edu/2020/06/19/its-alive-2/
  - used the average of polling in each state, together with the nominal statistical uncertainties and an estimate of extra uncertainty, to predict a range of election outcomes
  - uses Monte Carlo simulations based on each poll's results to get predictions
  - basically a pure-frequentist implementation, plus a model of how uncertain polls are at some point in time at predicting results in November
- Actual result gave 306 electoral votes to Biden, https://www.270towin.com/2020_Election/interactive_map

We can think of expert opinion as being the equivalent of a strong prior:

- Go to: https://www.270towin.com/maps/cook-political-2020-electoral-ratings
  - experts look at each state race on its own
  - polls are only one ingredient used to make judgements; past experience is key
  - good track record; e.g. races listed as 'toss-ups' by Cook Report in the past have, on average, been won ~50% by Republicans, ~50% by Democrats
  - priors are one way of encoding expert knowledge: in the absence of other information, previous experience of similar years leads to judgements of how likely each candidate is to win

Hybrid techniques are also possible

- Go to: https://projects.fivethirtyeight.com/2020-election-forecast/
  - uses a model incorporating national and state polling, how similar different states are, biases of different pollsters compared to the average, etc., to predict a range of possible results (using Monte Carlo simulations of how far off each poll could be). Priors are based on economic conditions.
  - There's some amount of subjective choice in modeling. What properties of a state define 'similarity'? To what degree will this election be like past elections? The recipe the site uses has changed over time.
  - The models are tuned so that they give the right rates of success for past elections.
  - There are now a number of forecasters doing this sort of thing; see https://projects.economist.com/us-2020-forecast/president for an open-source equivalent.

Let's generalize from coins and dice to more complicated situations...

- We'll come back to applications of Bayes' theorem, but need more background to go further.
- Till now, we've mostly talked about probabilities with just a few possibilities, equally likely. However, we can instead consider arbitrary probabilities as a function of continuous variables: a ***probability density function*** or PDF (for functions of discrete variables, the term is ***probability mass function*** or PMF).
- Probability density functions just have to be nonnegative everywhere, and integrate to 1 (or have sum 1 in the discrete case). For a continuous PDF $f(x)$,

$$p(a < x < b) = \int_a^b f(x)\, dx$$

- Sometimes, the term ***probability distribution*** is used instead of PDF.

More on distributions

- It is sometimes helpful to look at the fraction of the integral of *f(x)* which is below some value. This is the ***cumulative distribution function*** or CDF, generally written *F(x)*:

$$F(x) = \int_{-\infty}^{x} f(y)\, dy$$

- Another useful thing to look at can be the expectation value of some function *g(x)*:

$$\mathsf{E}g = \int_{-\infty}^{\infty} g(x)\, f(x)\, dx$$

- The expectation value is the value of *g(x)*, weighted by the probability of each possible x.

Expectation values

- Ex.: suppose I'll pay you $5 if a coin comes up heads, and you pay me $5 if it comes up tails. What is the expectation value of your winnings after a coin toss?
- Some important expectation values are based on the moments of the distribution (remember basic mechanics...):
  - The ***mean*** is, simply, the first moment of f(x) - compare to the center of mass in mechanics. It provides a measure of the ***location*** or ***center*** of $f(x)$.

$$\mu = \mu_1 = \mathrm{E}x = \int_{-\infty}^{\infty} x\, f(x)\, dx$$

Variance and standard deviation (*of a probability distribution*)

- We often are interested in the width, not just central value, of a probability distribution (e.g. an error bar). We can measure this with a second moment (compare to the moment of inertia about the center of mass):

$$\sigma^2 = \mu_2 = \mathrm{E}(x-\mu)^2 = \int_{-\infty}^{\infty} (x-\mu)^2 \; f(x)\; dx$$

- In statistics, μ_2 is called the ***variance***, and is equal to the ***standard deviation*** squared. σ provides a measure of width in the same units as x.
- We can also construct statistics to describe other moments: the ***skewness*** = μ_3 / σ^3 = $\mathbf{E}(x\text{-}\mu)^3 / \sigma^3$ describes how asymmetric a distribution is, the ***kurtosis*** = μ^4 / σ^4 describes how flattened it is, etc. A Gaussian distribution has 0 skewness and kurtosis of 3.

Although $f(x)$ can take arbitrary form (if nonnegative and normalized to have integral 1), there are dozens of well-studied cases.

- Let's start, again, with dice rolls. What is the probability of getting M ones when we roll N dice? Open up today's jupyter notebook...
- E.g.: if we roll 100 dice, how many 1's will we observe, in total?

```
import numpy.random as random
import numpy as np
nsims=int(1E5)
prob=1/6.
is_one=(random.rand(????,100) < prob)
ndice=100
# plot a histogram of the total # of 1's from each sim
plt.hist(np.sum(is_one[:,0:ndice],????) )
```

- We'd like to try this with ndice=2,5,10,50,100, and do a few repeats, plotting each time. This can get to be a pain retyping things or copying and pasting -- so I've made a function for you in a module file.

Using another module

- Last week, we wrote a function to give the result of 2 coin flips.
- This time, I made a module for you that rolls dice: **dice.py**
- Download **dice.py** from Courseweb to a directory in your `PYTHONPATH:` e.g., `~/python/`
- Open the file up in vscode to see what is in it...

Contents of dice.py

```
import numpy.random as random
import numpy as np
import matplotlib.pyplot as plt
```

Contents of dice.py

```
def rolldice(nsims):
# nsims is number of simulations to do
nsims =int(nsims)
    prob=1/6.
    is_one=(random.rand(nsims,100) < prob)
# generate nsims sets of 100 rolls
    ndice_array=[2,5,10,25,100]

    for i,ndice in enumerate(ndice_array):
        plt.figure(i) # create a new figure for each plot
        plt.hist( np.sum(is_one[:,0:ndice],axis=1),
range=(-0.5,ndice+0.5),bins=(ndice + 1))
        plt.title(str(ndice) + ' dice')
 # convert ndice to a string with str(), then use that to title the plot
```

Running the function

- **Import dice and run `dice.rolldice()`, e.g., with 50_000 simulations.**

What would we expect to see?

- Let p = the probability of getting a 1 (=1/6.)
- Each roll of the dice is independent of all others, so *p(A AND B)=p(A) p(B)* for each combination of die rolls
- e.g. probability of rolling a 1 on the first roll, the second, etc. N times must be p^N
- probability of a 1 on the first M rolls, and non-1 all the rest would be: $p^M(1-p)^{N-M}$, since each roll is independent
  - the same must be true for *any* specific ordering of the rolls that gives M ones total, as we'll have to have M factors of p and N factors of (1-p)

How many ways can we get *M* ones?

- If we have *N* things, there are *N*! (*N* factorial) different ways of ordering them.
- However, any case with a 1 coming up on dice 1, 3, and 5 only, say, is indistinguishable, and we shouldn't double-count them.
- There are *M*! ways to reorder the *M* cases of *p* that are indistinguishable, and then we can still reorder the *N-M* cases of (1-*p*) - so there are a total of $C(N,M)=N!/(M!(N-M)!)$
- So, summing up the probability of each case with M ones (since they are mutually exclusive), we find:

$$prob(M\ ones) = C(N,M)\ p^{M}\ (1-p)^{N-M}$$

The Binomial Distribution

- We didn't really use the fact that we're looking at dice anywhere in that derivation. In general, if there is a probability p of success, and we do N trials, then:

  $prob(M\ successes) = C(N,M)\ p^M\ (1-p)^{N-M}$

- This formula defines the binomial distribution. This is the distribution that controls coin tosses (like in the homework), or any other set of independent events of fixed probability.
- It has mean = $<x>$:

  $\mu = N\ p$

- and variance = $<x^2> - (<x>)^2$:

  $\sigma^2 = N\ p\ (1-p)$

Testing our distributions

- We expect that the number of ones we get will follow a binomial distribution with N=ndice. Let's test this by modifying the rolldice function in dice.py:

1) Does the data have mean $\mu = N\, p$?

- In python, we can check this with either np.mean or np.sum:

```
print(f'ndice: {ndice}')
print(f'np.mean: {np.mean( np.sum(is_one[:,
     0:ndice],axis =1) ) }')
print(f'np.sum: {np.sum( np.sum(is_one[:,0:ndice],axis=1) )*1./nsims:.4f}')
print(f'Expected mean: {ndice*prob:.4f} ')
```

- Note: some good links on f-string formatting: https://realpython.com/python-f-strings/ , https://fstring.help/ , http://cissandbox.bentley.edu/sandbox/wp-content/uploads/2022-02-10-Documentation-on-f-strings-Updated.pdf

Be sure to reload the module when you make changes!

Testing our distributions

2) Does the data have variance $\sigma^2 = N * p * (1\text{-}p)$?

- In Python, we can check this with `np.std()` or `np.var()`:

```
print(f'np.std**2:{np.std( np.sum(is_one[:,
    0:ndice],axis=1) )**2:.4f}')
print(f'np.var: {np.var( np.sum(is_one[:,
    0:ndice],axis=1) ) :.4f}')
print(f'Expected variance: {ndice*prob*(1-prob):.4f}')
#Then to make things look prettier print a blank line
# after each set of numbers:
print('')
```

Be sure to reload the module when you make changes!

Results

- Did that all check out? If so, let's make rolldice also plot the predicted distributions. First, we need to add an import:

```
from scipy.misc import factorial,comb
```

- Then, after:

```
plt.hist( np.sum(n_ones[:,0:ndice],axis=1),range=(-0.5,ndice+0.5),bins=(ndice + 1) )
```

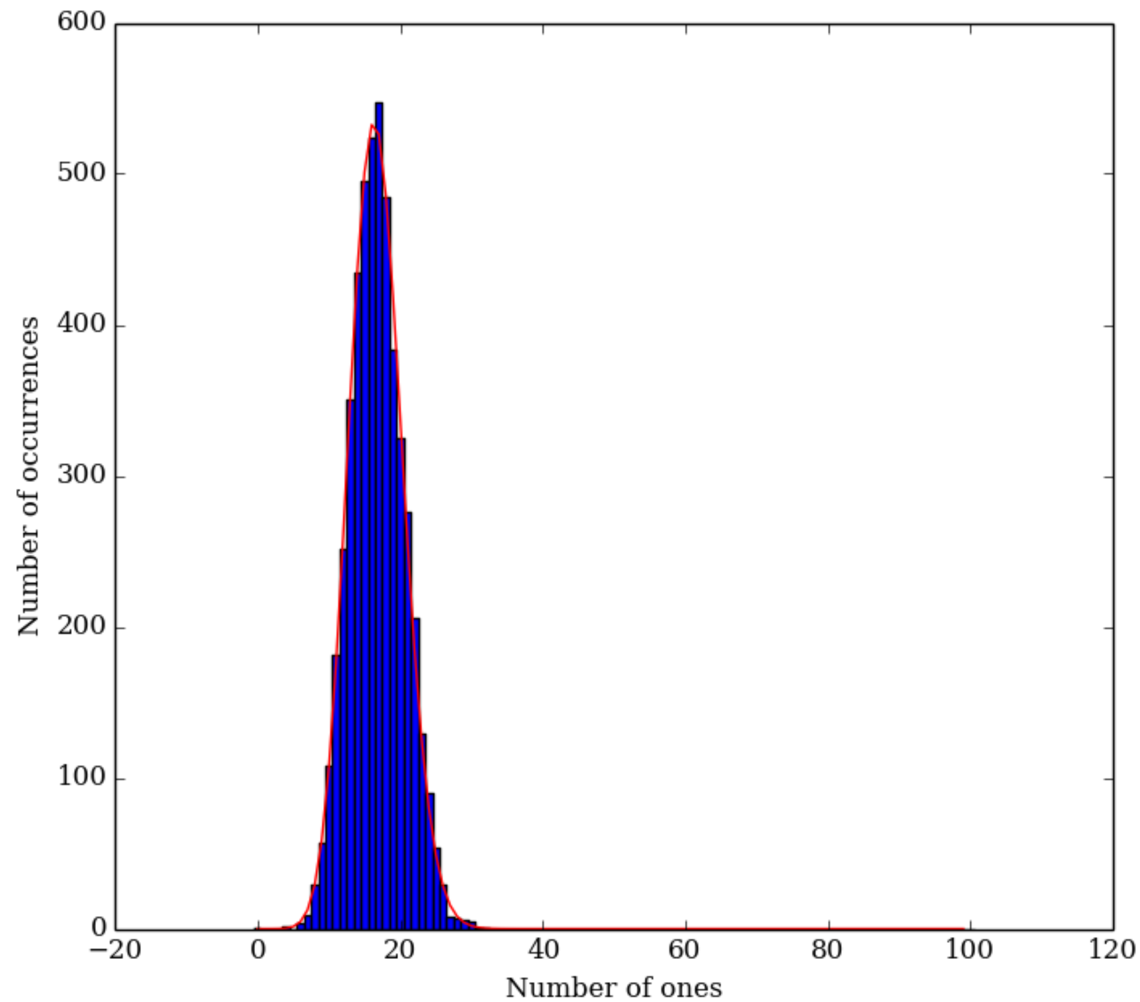
- add:

```
x=np.arange(ndice)
plt.plot(x,nsims*factorial(ndice)/factorial(x)/   \
 factorial(ndice-x)*prob**x*(1-prob)**(ndice-x),'r-')
```

- or:

```
plt.plot(x,nsims*comb(ndice,x)*prob**x*(1-prob)**(ndice-x),'ro')
```

It works!



600
500
400
300
200
100
0
Number of occurrences
−20
0
20
40
60
80
100
120
Number of ones

We could have used a scipy function instead

- `scipy` has a class (i.e., type of object), `scipy.stats.binom`, that allows you to calculate pretty much anything you'd want about a binomial distribution.
- It offers many subsidiary functions; `scipy.stats.binom.pmf(x,n,prob)` provides the probability of getting `x` occurrences out of `n` trials if the probability of an occurrence is `prob` (note that `x` can be an array!)
- alternatively, you can set up an object that is a member of the binomial class and inherits all of its functions ("methods"), but set up to assume n trials and probability prob, with e.g.

```
a = stats.binom(n,prob)
```

and then get the PMF (the discrete equivalent of a PDF) via

```
a.pmf(x)
```

We could have also used a scipy function

- `import scipy.stats` and look at the help information for `binom`. **Now modify your function** to add a curve to your plots showing the expected number of ones out of the simulation for each value of x using `stats.binom`; this should be `nsims` times the probability for one simulation...

- Note: you can actually look at the code for binom! Just do:

```
??scipy.stats.binom
```