PROBABILITY AND SIMULATION

ETHAN LEVIEN

Contents

1. Statistical models

A central concept to this course is that of a **model**. Before doing any mathematics, let's reflect on the notion of a model in the abstract.

Exercise 1. *What does it mean to model something?*

Broadly speaking, models are simplified representations of things in the world. In science, we use **mathematical models** to learn new things about the world and make predictions. You might be familiar with Newton's equation:

$$F = ma \tag{1}$$

which related the force (F) acceleration (a) and mass (m) of a particle. This a mathematical model of the motion of a (non-relativistic) particle in a force field. While it isn't always true, it holds for such a wide range of applications that we call it a *fundamental* law of nature, and it turns out to be extremely powerful. We can combine Newton's equation with other fundamental lows build models of more complicated systems involving many particles. For example, to build a model of planetary motion, we can combine Newton's equation with his other law of gravity, which states that the gravitation force between two objects of masses m_1 and m_2 a distance r from each other is

$$F = G\frac{m_1 m_2}{r^2} \tag{2}$$

where G is a constant. We refer to these types of models – that is, those which are built upon fundamental laws of nature – as **mechanistic** models.

It can be difficult to define fundamental laws and mechanistic models precisely, as one can debate what exactly is fundamental. In statistics, however, we are often interested in problems where nothing remotely close to a fundamental laws exist. Instead, they are based directly on observations (data) or our intuitions. In economics, for example, models are built in numerous assumptions about the human nature, many of which may seem obvious or intuitive. Yet our intuition about economics is itself shaped by the particular economic system we experience, and in reality these assumptions regularly break down as economic conditions change. We will call these **phenomenological models**. Phenomenological models can still be quite useful provided we are aware of their limitations.

Often our models cannot make exact predictions about the value of a variable (this is true in both mechanistic and phenomenological models). Instead, they only tell us the probability that a variable has a certain value. For example, if we were to construct a model of the (y) of all the pine trees in New Hampshire as a function of their age (x), we might begin by searching for a function f such that

$$y = f(x) \tag{3}$$

However, if we take a **sample** of the population, meaning we go out and measure the height of some trees for which we know the age, we will quickly find that trees of the same age can have different heights. This variation will be as result of

Date: April 2022.

many variables which are not in our model, e.g. the surrounding, the specific subspecies of pine, genetic variation within a subspecies to name just a few. We could construct a more sophisticated model which includes, say, the DNA sequence of the tree (g):

$$y = f(x, g). \tag{4}$$

This model would presumably have less variation. That is, if we collect a sample of trees for which we know the height and DNA sequence, we would find that the variation between trees with the same DNA and height is less than the variation between trees with only the same height. However, variation will remain unless we include all the variables which effect tree height. This is of course impossible, as there are million, perhaps billions of such variables and their effects are far beyond our understanding of biology. Moreover, it is not so useful to include these variables, since we can actually measure most of them and many of them will have small effects. One approach to dealing with these "hidden" sources of variation is to define a **statistical model**, where y is not the same for each tree of the same height, but is instead a **random variable**. On type of a statistical model is a regression model

$$y = f(x) + \epsilon \tag{5}$$

where ϵ is a random variable (usually one that is zero, on average).

Exercise 2. *When we write down a model, statistical or otherwise, we make assumptions. What are some of the assumptions we have made by writing down Equation* (5)*? Hint: think about other ways in which you might compute a random term, ϵ, with a function like f.*

2. Random variables and distributions

We can define a random variable more generally as a variable we can not predict exactly prior to an observation of the variable, no matter how much information we have (e.g. the roll of a dice). When we observe a random variable, it will take on a value from a set of possible outcomes ($1, 2, \ldots, 6$ for the dice). We can describe a characterize a random variable using a **probability distribution**, which maps a set of possible outcomes to to real numbers between 0 and 1. Usually the outcomes are numbers, even if we use a number to represent a non-numerical quantity.

We can describe a random variable by a probability distribution, which tells us the chance we observe each possible outcome. For example, suppose we ask a random student in the college whether they identify as male or not and let Y denote their answer. What is a model for Y? A random variable Y taking on the possible outcomes in $\{0, 1\}$ is said to be a Bernoulli random variable. In this case we can write the probability for Y to take on these different outcomes as

$$\mathbb{P}(Y) = \begin{cases} q & Y = 0 \\ 1 - q & Y = 1 \end{cases} \tag{6}$$

We call $\mathbb{P}(Y)$ the probability distribution of Y. It is very important that the sum of $\mathbb{P}(Y)$ over all possible outcomes is 1 – this is simply saying that we are certain one of the outcomes will happen. The number q is called a **parameter** – it changes properties of the distribution while preserving the defining features of that distribution (that the variable has two outcomes).

The Bernoulli distribution is our default model for any variable that can take two possible outcomes. In order to say that we are modeling Y with a Bernoulli distribution symbolically, we write

$$Y \sim \text{Bernoulli}(q). \tag{7}$$

We might also say "Y *follows* as Bernoulli distribution" or "Y is a Bernoulli random variable". More generally, we say that a variable in a model follows a given distribution by writing

$$\text{Variable} \sim \text{Distribution}(\text{parameters}). \tag{8}$$

Turning back to the model, we notice there is a missing piece: the value of q. Without this, we can't make any predictions at all. The process is determining q based on data (e.g. a survey of Dartmouth students or conversations we have had) is known as **statistical inference**.

2.1. **Simulating random variables in python.** Often when we generate samples using a computer we call them **simulations**. In Python, we can simulate random variables using the `numpy` library:

```
> import numpy as np
> q = 0.5
> y = np.random.choice([0,1],p=[q,1-q])
```

We can generate multiple samples using a for loop

```
> n_samples = 100
> y = np.zeros(n_samples) # makes an empty list (i.e. array) of n_samples zeros.
> for k in np.range(n_samples):
>   y[k] = np.random.choice([0,1],p=[q,1-q])
```

or we can simply write

```
> y = np.random.choice([0,1],n_samples,p=[q,1-q])
```

An important tool for visualizing samples is a histogram

```
> plt.hist(samples,100,density=true)
```

The histogram shows us the frequency of different outcomes.

Example 1. *Write a function which simulates flipping a fair coin until we get 2 heads in a row.*

Solution:

```
> def flip_until_two():
>   num_heads = 0
>   total_flips = 0
>   while num_heads <2:
>     y = np.random.choice([0,1])
>     if y == 0:
>       num_heads = 0
>     else:
>       num_heads = num_heads + 1
>     total_flips = total_flips + 1
>   return total_flips
```

Often, we run many simulations of a model in order to say something about the distribution without performing any analytical calculations. We call these **Monte Carlo** simulations.

Exercise 3. *Use Monte Carlo simulations to estimate the average number of flips it takes to get 2 heads in a row from a fair coin. Plot a histogram and estimate the chance it takes more than 6 flips.*

Solution:

```
> n_monte = 1000
> flips = np.zeros(n_monte)
> for k in range(n_monte):
>   flips[k] = flip_until_two()
> plt.hist(flips)
```

2.2. **Means and variances.** There are ways in which we summarize attributes of random variables. If we have many samples $Y_1, Y_2, \ldots, Y_n$ of a random variable (e.g. answers to a survey question), the **sample mean** is defined as

$$\bar{y} = \frac{1}{n}\sum_i y_i \tag{9}$$

Often it is useful to quantify the deviations from the mean. Suppose y_i can take on outcomes $y_1, \ldots, y_m$. If n is large, then the fraction of samples for which $Y_1 = y_1$ will be $\mathbb{P}(Y_1 = y_1)$.

$$\bar{y} \approx \frac{1}{n}\sum_{\text{outcomes}} y_i n_i = \sum y_i \mathbb{P}(Y_i = y_i) \tag{10}$$

The expression on the right is the definition of expected value, often denoted $\mathbb{E}[Y]$.

For this, we have the **sample standard deviation**

$$\sigma = \sqrt{\frac{1}{n-1}\sum_i (y_i - \bar{y})^2}. \tag{11}$$

We will see why this makes sense as a measure of how spread our a distribution is later on when we talk about inference. For large n, this converges to the square root of the variance

$$\mathrm{Var}(Y) = \sum (y_i - \mathbb{E}[Y])^2 \mathbb{P}(Y_i = y_i). \tag{12}$$

Importantly, we can always calculate the mean and standard deviation of a sample, regardless of the distribution it has been drawn from. However, we need to be careful, as the results may not be so meaningful.

In python, functions for implementing the mean and standard deviation are follows:

```
> np.mean(y)
> np.std(y)
```

Example 2. *Verify that the formula for the mean and standard deviations with Monte Carlo simulations*

$$\mathbb{E}[Y] = q, \quad \sqrt{\mathrm{Var}(Y)} = \sqrt{q(1-q)} \tag{13}$$

Solution:
The following code will estimate the mean and standard deviations for various values of q:

```
> def mean_std_bern(q):
>   sample = np.random.choice([0,1],500,p=[1-q,q])
>   return np.mean(sample),np.std(sample)
>
> # generates one hundred evenly spaced numbers between 0 and 1
> q_range = np.linspace(0,1,100)
> means = np.zeros(len(q_range))
> stds = np.zeros(len(q_range))
> for j in range(len(q_range)):
>   means[j],stds[j] = mean_std_bern(q_range[j])
```

Here is a plot of the mean compared to the prediction

```
> fig,ax = plt.subplots(figsize=(5,2))
> ax.plot(q_range,means)
> ax.plot(q_range,q_range,"k--")
```

and the standard deviations

```
> fig,ax = plt.subplots(figsize=(5,2))
> ax.plot(q_range,stds)
> ax.plot(q_range,np.sqrt((1-q_range)*q_range),"k--")
```

3. Joint probabilities, Independence and correlation

We introduce, very briefly, the concepts of independence and conditioning. Just as we have considered single random variables, we can consider multiple random variables within the same model. Suppose we have two Bernoulli random variables Y_1 and Y_2 which model whether a person has mutations at two different genomes. In this case, we need a model of both variables together:

$$\mathbb{P}(Y_1, Y_2) = \begin{cases} q_{00} & \text{if } Y_1 = 0 \text{ and } Y_2 = 0 \\ q_{01} & \text{if } Y_1 = 0 \text{ and } Y_2 = 1 \\ q_{10} & \text{if } Y_1 = 1 \text{ and } Y_2 = 0 \\ q_{11} & \text{if } Y_1 = 1 \text{ and } Y_2 = 1 \end{cases} \tag{14}$$

Example 3. *Write python code to generate a sample of Y_1 and Y_2 using*

$$q_{00} = 1/8, \quad q_{01} = 1/8, \quad q_{10} = 1/4, \quad q_{11} = 1/2 \tag{15}$$

Solution:
There are 4 outcomes, so we can identify each with a number as follows:

$$(0,0) \to 1, \quad (0,1) \to 2, \quad (1,0) \to 3, \quad (1,1) \to 4 \tag{16}$$

We then use the random choice function

```
> np.random.choice([1,2,3,4],p=[1/8,1/8,1/4,1/2])
```

The probability distribution $\mathbb{P}(Y_1, Y_2)$ tells us the probabilities for observing *both* variables together, e.g. observing a person with both mutations. It does not directly tell us the probabilities of observing e.g. someone with only one mutation. This can be obtained via marginalization; that is, summing over the other variable:

$$\mathbb{P}(Y_1) = \sum_y \mathbb{P}(Y_1, y) = \mathbb{P}(Y_1, 0) + \mathbb{P}(Y_1, 1) \tag{17}$$

where in the general the sum is taken over all possible outcomes for the second variable. $\mathbb{P}(Y_1)$ is defined similarly. Sometimes, we are interested in the value of a random variable given, or **conditioned on** another random variable. The probability of that $Y_1 = 1$ if we know $Y_2 = 0$, denoted $\mathbb{P}(Y_1 = 1|Y_2 = 0)$ would be the chance that gene 1 has a mutation in a person if we know there is no mutation at gene 2.

How do we calculate this? Bayes theorem tell us that for two random variables X and Y

$$\mathbb{P}(Y|X) = \frac{P(Y, X)}{P(X)} \tag{18}$$

Two variables are independent if $\mathbb{P}(Y|X) = P(X)$.

Example 4. *What is $\mathbb{P}(Y_1 = 1|Y_2 = 0)$ in terms of the $q_{Y_1Y_2}$ variables?*

Solution:
In this case, we have

$$\frac{P(Y_1 = 2, X_2 = 0)}{P(Y_2 = 0)} = \frac{q_{10}}{q_{10} + q_{01}} \tag{19}$$

Example 5. *Use Monte Carlo simulations to confirm Equation* (19) *for the values given in exercise ??.*

Solution:
This is achieved by the following code:

```
> y = np.random.choice([1,2,3,4],1000,p=[1/8,1/8,1/4,1/2])
> y_sub = y[y ==1 or y==3]
> len(y_sub[y==1])/len(y_sub)
```

Exercise 4. *Let Y be the value of a 6 sided die. What is $\mathbb{P}(Y = 3|Y > 2)$? Confirm you answer with Monte Carlo simulations.*

Solution:
This is achieved by the following code:

```
> y = np.random.choice([1,2,3,4,5,6],1000,p=[1/6,1/6,1/6,1/6,1/6,1/6])
> y_sub = y[y >2]
> len(y_sub[y==3])/len(y_sub)
```

4. More random variables

4.1. Binomial: Distribution.

A situation that often arrises is that we take many, say n, independent samples from a Bernoulli distribution. Now let Y be the number of 1s. Then Y follows **binomial distribution**:

$$Y \sim \text{Binomial}(n, q) \tag{20}$$

The binomial distribution has two parameters, k and p, and the probability distribution is

$$\mathbb{P}(Y) = \binom{n}{Y} q^Y (1 - q)^{n-Y} \tag{21}$$

The binomial distribution has a mean qn and variance $nq(1 - q)$. We can draw samples from a binomial distribution

```
> y = np.random.binomial(n,p,n_samples)
```

Exercise 5. *Run simulations of the Binomial distribution and plot the average and standard deviation of the average as a function of n.*

Solution:
The following code will estimate the mean and standard deviations for various values of q:

```
> def mean_std_bern(n,q):
>   sample = np.random.binomial(n,q,100)
>   return np.mean(sample),np.std(sample)
>
> # generates one hundred evenly spaced numbers between 0 and 1
> q_range = np.linspace(0,1,100)
> means = np.zeros(len(q_range))
> stds = np.zeros(len(q_range))
> for j in range(len(q_range)):
>   means[j],stds[j] = mean_std_bern(q_range[j])
```

Here is a plot of the mean compared to the prediction

```
> fig,ax = plt.subplots(figsize=(5,2))
> ax.plot(q_range,means)
> ax.plot(q_range,q_range,"k--")
```

and the standard deviations

```
> fig,ax = plt.subplots(figsize=(5,2))
> ax.plot(q_range,stds)
> ax.plot(q_range,np.sqrt((1-q_range)*q_range),"k--")
```

4.2. **Normal distribution and the central limit theorem.** In the previous example, we say that if we take the average of many Bernoulli random variables, we get a histogram that looks a lot like a "bell curve" with a standard deviation that scales as $1/\sqrt{n}$.

It turns out this is true when we add up *any* random variables which are sufficiently independent (we will make this precise soon). Since the "bell curve" arrises in the limit where we sum or average many random variables. This motivates us to define the Normal distribution. Unlike previous random variables we've seen, a normal random variable

$$Y \sim \text{Normal}(\mu, \sigma) \tag{22}$$

can take on any number, positive or negative, decimal or integer. We can generate Normal random variables in python with

```
> np.random.normal(0,1)
```

We want to describe this random variable in terms of a probability distribution, but the probability for Y to equal any given value of Y is zero. To see this, note that the probabilities must sum to one In this case, we can describe the probability distribution

$$\mathbb{P}(y_1 < Y < y_2) = \int_{y_1}^{y_2} f(y)dy \tag{23}$$

for a function f called the probability density. It is given by

$$f(y) = \frac{1}{\sqrt{2\pi\sigma}} \, ^{-\frac{-(y-\mu)^2}{2\sigma}}. \tag{24}$$

[SHOW FIGURE OF BELL CURVE]

Example 6. *Confirm the probabilities for the Normal distribution using Monte Carlo simulation*

Going forward, you will need to know the following properties of Normally distributed random variables. Let

$$Y_1 \sim \text{Normal}(0,1), \quad Y_2 \sim \text{Normal}(0,1) \tag{25}$$

Then

$$aY_1 + b \sim \text{Normal}(b, a) \tag{26}$$

$$aY_1 + bY_2 \sim \text{Normal}(0, \sqrt{a^2 + b^2}) \tag{27}$$

Example 7. *Show that if*

$$Y_1 \sim \text{Normal}(\mu_1, \sigma_1), \quad Y_2 \sim \text{Normal}(\mu_2, \sigma_2) \tag{28}$$

then

$$aY_1 + bY_2 + c \sim \text{Normal}(\mu_1 + \mu_2 + c, \sqrt{(a\sigma_1)^2 + (b\sigma_2)^2}) \tag{29}$$

We now return to the connection to sums of random variables and the Binomial distribution. The **central limit theorem** tells us that for a set of random variables $x_1, \ldots, x_n$ with each have $\mathbb{E}[x_i] = \mu_x$ and $\text{var}(x_i) = \sigma_x^2$,

$$Y = \sum x_i \tag{30}$$

is approximately Normal with mean μ and variance $\sigma/\sqrt{n}$. In other words Normal random variables emerge when we add up many small sources of randomness.

Exercise 6. *Read the wikipedia page for the following random variables. Look up the following random variables on wikpedia and for each one, determine (1) whether the random variable is discrete or continuous and (2) what the parameters are. Then generate Monte Carlo simulations to confirm the given formula's for means and variances.*

- *Binomial*
- *Chi-Squared*

Solution: