5. Discrete Distributions

Chris Piech and Mehran Sahami

Oct 2017

There are some classic random variable abstractions that show up in many problems. In this chapter you will learn about several of the most significant discrete distributions. When solving problems, if you are able to recognize that a random variable fits one of these formats, then you can use their precalculated Probability Mass Function (PMF), expectations, and variance.

Random variables of this sort are called "parametric" random variables. If you can argue that a random variable falls under one of the studied parametric types, you simply need to provide parameters. A good analogy is a Class in programming. Creating a parametric random variable is very similar to calling a constructor with input parameters.

Bernoulli Random Variable

A Bernoulli random variable is the most simple random variable. It can take on two values, a 1 or a 0. It takes on a 1 if an experiment with probability p resulted in success and a 0 otherwise. Some example uses include a coin flip, random binary digit, whether a disk drive crashed or whether someone likes a Netflix movie.

Let X be a Bernoulli Random Variable $X \sim Ber(p)$.

Probability Mass Function
$P(X = 1) = p$
$P(X = 0) = (1 - p)$

Expectation
$E[X] = p$

Variance
$Var(X) = p(1 - p)$

Some folks call a Bernoulli random variable an Indicator random variable. A random variable I is called an indicator variable for an event A if $I = 1$ when A occurs and $I = 0$ if A does not occur. $P(I = 1) = P(A)$ and $E[I] = P(A)$. Indicator random variables **are** Bernoulli random variables.

Binomial Random Variable

A Binomial random variable is random variable that represents the number of successes in n successive independent trials of a Bernoulli experiment. Some example uses include # of heads in n coin flips, # dist drives crashed in 1000 computer cluster.

Let X be a Binomial Random Variable. $X \sim Bin(n,p)$ where p is the probability of success in a given trial.

$$P(X=k) = \binom{n}{k} p^k (1-p)^{n-k}$$
$$E[X] = np$$
$$Var(X) = np(1\text{–}p)$$

Example 2

Let X = number of heads after a coin is flipped three times. $X \sim Bin(3,0.5)$. What is the probability of different outcomes?

$$P(X=0) = \binom{3}{0} p^0 (1-p)^3 = \frac{1}{8}$$
$$P(X=1) = \binom{3}{1} p^1 (1-p)^2 = \frac{3}{8}$$
$$P(X=2) = \binom{3}{2} p^2 (1-p)^1 = \frac{3}{8}$$
$$P(X=3) = \binom{3}{3} p^3 (1-p)^0 = \frac{1}{8}$$

Example 3

When sending messages over a network there is a chance that the bits will become corrupt. A Hamming Code allows for a 4 bit code to be encoded as 7 bits, and maintains the property that if 0 or 1 bit(s) are corrupted then the message can be perfectly reconstructed. You are working on the Voyager space mission and the probability of any bit being lost in space is 0.1. How does reliability change when using a Hamming code?

Image we use error correcting codes. Let $X \sim Bin(7,0.1)$

$$P(X=0) = \binom{7}{0} (0.1)^0 (0.9)^7 \approx 0.468$$
$$P(X=1) = \binom{7}{1} (0.1)^1 (0.9)^6 = 0.372$$
$$P(X=0) + P(X=1) = 0.850$$

What if we didn't use error correcting codes? Let $X \sim Bin(4,0.1)$

$$P(X=0) = \binom{4}{0} (0.1)^0 (0.9)^4 \approx 0.656$$

Using Hamming Codes improves reliability by 30%

Binomial in the Limit

Recall example of sending bit string over network. In our last class we used a binomial random variable to represent the number of bits corrupted out of four with a high corruption probability (each bit had independent probability of corruption $p = 0.1$). That example was relevant to sending data to space craft, but for earthly applications like HTML data, voice or video, bit streams are much longer (length $\approx 10^4$) and the probability of corruption of a particular bit is very small ($p \approx 10^{-6}$). Extreme n and p values arise in many cases: # visitors to a website, #server crashes in a giant data center.

Unfortunately X $\sim$ Bin(10^4, 10^{-6}) is unwieldy to compute. However when values get that extreme we can make approximations that are accurate and make computation feasible. Recall the Binomial distribution. First define $\lambda = np$. We can rewrite the Binomial PMF as follows:

$$
\begin{aligned}
P(X=i) &= \frac{n!}{i!(n-1)!}\left(\frac{\lambda}{n}\right)^i\left(1-\frac{\lambda}{n}\right)^{n-i} \\
&= \frac{n(n-1)\dots(n-i-1)}{n^i}\frac{\lambda^i}{i!}\frac{(1-\lambda/n)^n}{(1-\lambda/n)^i}
\end{aligned}
$$

This equation can be made simpler by observing how some of these equations evaluate when n is sufficiently large and p is sufficiently small. The following equations hold:

$$
\begin{aligned}
&\frac{n(n-1)\dots(n-i-1)}{n^i} \approx 1 \\
&(1-\lambda/n)^n \approx e^{-\lambda} \\
&(1-\lambda/n)^i \approx 1
\end{aligned}
$$

This reduces our original equation to:

$$
P(X=i) = \frac{\lambda^i}{i!}e^{-\lambda}
$$

This simplification turns out to be so useful, that in extreme values of n and p we call the approximated Binomial its own random variable type: the Poisson Random Variable.

Poisson Random Variable

A Poisson random variable approximates Binomial where n is large, p is small, and $\lambda = np$ is "moderate". Interestingly, to calculate the things we care about (PMF, expectation, variance) we no longer need to know n and p. We only need to provide λ which we call the rate.

There are different interpretations of "moderate". The accepted ranges are $n > 20$ and $p < 0.05$ or $n > 100$ and $p < 0.1$.

Here are the key formulas you need to know for Poisson. If $Y \sim Poi(\lambda)$:

$$
\begin{aligned}
&P(Y=i) = \frac{\lambda^i}{i!}e^{-\lambda} \\
&E[Y] = \lambda \\
&Var(Y) = \lambda
\end{aligned}
$$

Example

Let's say you want to send a bit string of length $n = 10^4$ where each bit is independently corrupted with $p = 10^{-6}$. What is the probability that the message will arrive uncorrupted? You can solve this using a Poisson with $\lambda = np = 10^4 10^{-6} = 0.01$. Let $X \sim Poi(0.01)$ be the number of corrupted bits. Using the PMF for Poisson:

$$
\begin{aligned}
P(X=0) &= \frac{\lambda^i}{i!}e^{-\lambda} \\
&= \frac{0.01^0}{0!}e^{-0.01} \\
&\sim 0.9900498
\end{aligned}
$$

We could have also modelled X as a binomial such that $X \sim Bin(10^4, 10^{-6})$. That would have been computationally harder to compute but would have resulted in the same number (up to the millionth decimal).

Geometric Random Variable

X is Geometric Random Variable: $X \sim Geo(p)$ if X is number of independent trials until first success and p is probability of success on each trial. Here are the key formulas you need to know. If $X \sim Geo(p)$:

$$P(X = n) = (1-p)^{n-1}p$$
$$E[X] = 1/p$$
$$Var(X) = (1-p)/p^2$$

Negative Binomial Random Variable

X is Negative Binomial: $X \sim NegBin(r, p)$ if X is number of independent trials until r successes and p is probability of success on each trial. Here are the key formulas you need to know. If $X \sim NegBin(p)$:

$$P(X = n) = \binom{n-1}{r-1} p^r (1-p)^{n-r} \text{ where } r \leq n$$
$$E[X] = r/p$$
$$Var(X) = r(1-p)/p^2$$

Zipf Random Variable

X is Zipf: $X \sim Zipf(s)$ if X is the rank index of a chosen word (where s is a parameter of the language).

$$P(X = k) = \frac{1}{k^s \cdot H}$$

Where H is a normalizing constant (and turns out to be equal to the Nth harmonic number where N is the size of the language.