


## Lab Report #1: Moments & Cumulants

Revised: September 6, 2014

*Due at the start of class. You may speak to others, but whatever you hand in should be your own work. Use Matlab where possible and attach your code to your answer.*

**Solution:** Brief answers follow, but see also the attached Matlab program. Download this document as a pdf, open it, and click on the pushpin: 

1. *Moments of the standard normal.* This should be review, but will get you started with moments and generating functions.

Suppose  $x$  is a standard normal random variable, meaning normal with mean  $\mu = 0$  and standard deviation  $\sigma = 1$  (equivalently variance  $\sigma^2 = 1$ ).

- (a) What is  $x$ 's probability density function?
- (b) What is  $x$ 's moment generating function (mgf)? (Don't derive it, just write it down.)
- (c) What is  $E(e^x)$ ? (Ditto.)
- (d) Differentiate the mgf to find the first two moments. How are they related to the mean and variance?
- (e) Find the third and fourth moments the same way. What are they?

**Solution:**

- (a) The word "standard" here means  $\mu = 0$  and  $\sigma = 1$ , so we have  $p(x) = (2\pi)^{-1/2} \exp(-x^2/2)$ .
- (b)  $h(s) = \exp(s^2/2)$ .
- (c) The mgf is  $h(s) = E(e^{sx}) = e^{s^2/2}$ , so  $E(e^x)$  is  $h(1) = e^{1/2}$ .
- (d) The first two moments are zero and one. They're both raw and central moments, because the mean is zero.
- (e) The third and fourth moments are zero and three. Why three? It's a curious property of the normal distribution.

2. *Sample moments of the standard normal.* It's often helpful to experiment with artificial test problems, just to remind ourselves how the code works. Here we compute sample moments of artificial data generated in Matlab and verify that calculations of various moments do what we think they do.

This generates the data we'll use:

```

format compact          % single-spacing of output
nobs = 1000;            % number of observations
rng('default');         % sets "seed" so you can replicate the output
x = randn(nobs,1);      % generates a vector of standard normals

```

These commands generate “pseudo-random” numbers from a standard normal distribution and put them in the vector **x**. (Standard normal means normal with mean equal to zero and variance equal to one.) As always, you can find out what Matlab commands do by typing **help command** at the prompt; for example, **help rng** or **help randn**.

- (a) Our first check is to see if the sample moments correspond, at least approximately, to our knowledge of normal random variables. Run the commands:

```

xbar = mean(x)
moments = mean([(x-xbar).^2 (x-xbar).^3 (x-xbar).^4])

```

What do you get? How do your calculations compare to the analogous moments of the standard normal distribution?

- (b) Our second check is on the Matlab commands **std(x)**, **skewness(x)**, and **kurtosis(x)**. How do they compare to the calculations you did above?

#### **Solution:**

- (a) The moments are close to what the distribution implies: mean one, variance (and standard deviation) one, skewness zero, and kurtosis three. I get -0.01, 0.985, -0.002, and 3.36, respectively. The small differences reflect sampling variability. You can see this by increasing the sample size, which typically makes the sample moments closer to the “population” moments.
- (b) The **skewness** and **kurtosis** commands give exactly the same answers as our own calculations, which assures us that they do what we want them to do. The standard deviation is a little different: the Matlab function divides the sum of squared deviations by the number of observations **minus one**, rather than just the number of observations. Our calculation takes the square root of the mean of the squared differences of **x** from **xbar**. Here I get 0.9928 when I use **std** and 0.9923 when I compute it as the square root of the mean of second sample moment.

3. *Cumulants of geometric random variables.* We say a random variable has a *geometric distribution* if it takes on the values  $x = 0, 1, 2, \dots$  with probabilities  $p(x) = (1 - \omega)\omega^x$  for some parameter  $\omega$  satisfying  $0 < \omega < 1/e$ .

Your mission: Compute the moments and cumulants of  $x$ . Use Matlab where possible.

- (a) Verify that this is a legitimate probability distribution.
- (b) Derive the moment generating function. (Hint: Apply the definition.) What is the cumulant generating function?
- (c) Use Matlab and the cgf to find the first four cumulants, labelled  $\kappa_1$  through  $\kappa_4$ . What are the mean and variance?

(d) Derive the standard measures of skewness and excess kurtosis:

$$\begin{aligned}\gamma_1 &= \kappa_3/(\kappa_2)^{3/2} \quad (\text{skewness}) \\ \gamma_2 &= \kappa_4/(\kappa_2)^2 \quad (\text{excess kurtosis})\end{aligned}$$

**Solution:**

(a) The probabilities are positive, so we're good there, but do they sum to one? We have

$$\sum_{x=0}^{\infty} (1-\omega)\omega^x = (1-\omega) \sum_{x=0}^{\infty} \omega^x = (1-\omega)/(1-\omega) = 1.$$

(b) The mgf is

$$\begin{aligned}h(s) &= E(e^{sx}) \\ &= (1-\omega) \sum_{x=0}^{\infty} \omega^x e^{sx} = (1-\omega) \sum_{x=0}^{\infty} (\omega e^s)^x = (1-\omega)/(1-\omega e^s).\end{aligned}$$

The cgf is the log of this:

$$k(s) = \log h(s) = \log(1-\omega) - \log(1-\omega e^s).$$

(c) The cumulants are

$$\begin{aligned}\kappa_1 &= \omega/(1-\omega) \\ \kappa_2 &= \omega/(1-\omega)^2 \\ \kappa_3 &= \omega(1+\omega)/(1-\omega)^3 \\ \kappa_4 &= \omega(1+4\omega+\omega^2)/(1-\omega)^4.\end{aligned}$$

The first two are the mean and variance.

(d) Skewness and kurtosis follow from the moments:

$$\begin{aligned}\gamma_1 &= \kappa_3/(\kappa_2)^{3/2} = (1+\omega)/\omega^{1/2} \\ \gamma_2 &= \kappa_4/(\kappa_2)^2 = 6 + (1-\omega)^2/\omega.\end{aligned}$$

This takes some work, Matlab doesn't give you anything this user-friendly. See also [Wikipedia](#) (the right column with  $p = 1 - \omega$ ).

4. *Sums of independent random variables.* Consider the sum of  $n$  random variables, say  $y = x_1 + x_2 + \cdots + x_n$  with the  $x_i$ 's "iid" (independent and identically distributed).

(a) Suppose the expansion of the cgf for one of the  $x_i$ 's is

$$k(s; x) = \kappa_1(x)s + \kappa_2(x)s^2/2 + \kappa_3(x)s^3/3! + \kappa_4(x)s^4/4! + \cdots.$$

What is the analogous expansion for  $y$ ? What does it tell you about the cumulants of  $y$ ?

- (b) Compute our measures of skewness and excess kurtosis for  $y$ . How do they vary with  $n$ ?

**Solution:**

- (a) The cgf of a sum is the sum of the individual cgfs, so here we have

$$\begin{aligned} k(s; y) &= nk(s; x) \\ &= n\kappa_1(x)s + n\kappa_2(x)s^2/2 + n\kappa_3(x)s^3/3! + n\kappa_4(x)s^4/4! + \cdots \\ &= \kappa_1(y)s + \kappa_2(y)s^2/2 + \kappa_3(y)s^3/3! + \kappa_4(y)s^4/4! + \cdots . \end{aligned}$$

The cumulants of  $y$  are  $n$  times the cumulants of  $x$ :  $\kappa_j(y) = n\kappa_j(x)$ .

- (b) Skewness and excess kurtosis are

$$\begin{aligned} \gamma_1(y) &= \kappa_3(y)/[\kappa_2(y)]^{3/2} = n\kappa_3(x)/[n\kappa_2(x)]^{3/2} = \gamma_1(x)/n^{1/2} \\ \gamma_2(y) &= \kappa_4(y)/[\kappa_2(y)]^2 = n\kappa_4(x)/[n\kappa_2(x)]^2 = \gamma_2(x)/n. \end{aligned}$$

We see both decline with  $n$ , so if  $n$  gets big enough they'll be arbitrarily close to zero. This is a close relative of the *central limit theorem*, in which the average of a sum of iid random variables gets closer and closer to normal as we increase the number of components  $n$ . Here we see skewness and excess kurtosis, which are zero in the normal case, go to zero, so roughly speaking the sum is getting more normal.