Distributions and Critical Values: Sections 10 - 14

How to Check for Normality Using a Normal Probability Plot

How to Get Normal Critical Values for Common Significance Levels

How to Generate a t-Distribution Graph and Obtain t-Critical Values

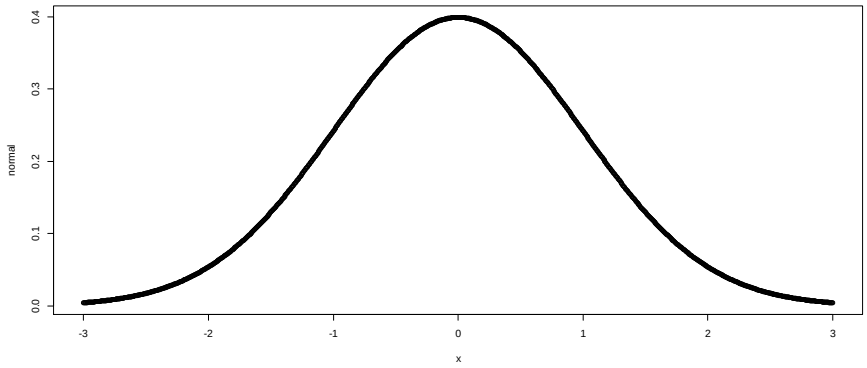
How to Generate Chi-Square and F Distributions and Obtain Their Critical Values

How to Generate and Graph a Binomial Distribution

Section 10: How to Check for Normality Using a Normal Probability Plot

(Uses data file: MO Life Expectancy.txt)

There are several commonly used distributions in elementary statistics – usually the binomial, the normal, the Student's-t, the chi-squared and the F-distributions. By far, the most used distribution is the normal. This follows the traditional "bell-curve" as shown below.



Most of the procedures that are in an elementary statistics course require that the sample come from a population that is approximately normal. R is very good for helping you visualize your data. You can visually assess your data set for normality in two ways: (1) create a histogram and see if it looks reasonably normal, or (2) create a "normal Q-Q plot." This plots the quantiles of a normal distribution on the horizontal axis, and plots the quantiles of your data on the vertical axis. If the sample comes from a normal population, the points should fall approximately along a straight line.

The following example uses life-expectancy data for the counties of Missouri in the period 2004-2012. (Source: http://health.mo.gov/data/lifeexpectancy/).

First, read the data into R. attach it, and display it if you choose.

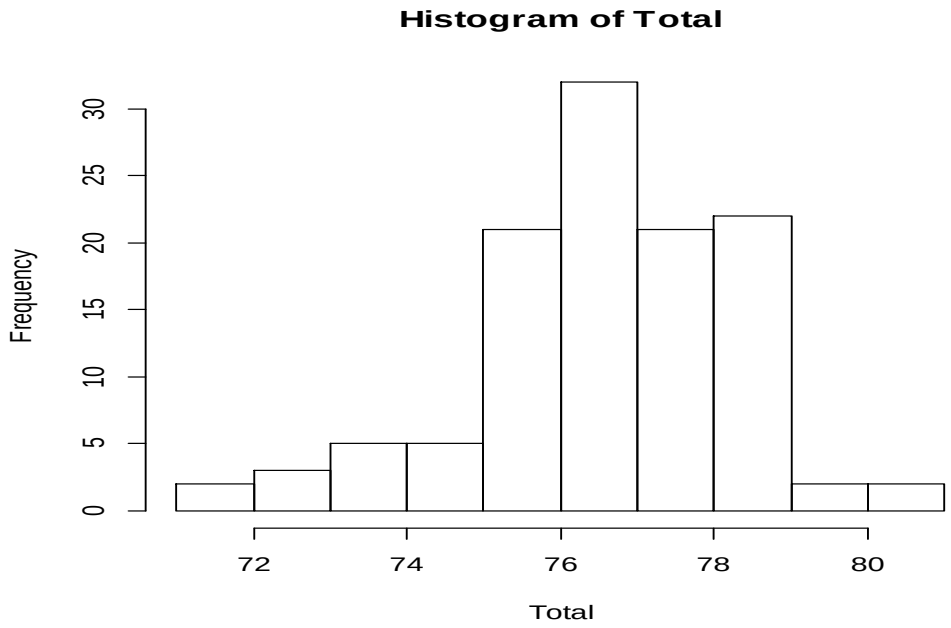
```
> LifeExp = read.table ("E:/Data Files/MO Life Expectancy.txt", header = TRUE)
> attach (LifeExp)
> LifeExp
```

The beginning and the end of data set is shown below. The example will use the column called "Total."

	County	Total	Male	Female
1	Adair	77.7	75.4	79.9
2	Andrew	77.8	75.1	80.3
3	Atchison	78.3	75.5	81.2
4	Audrain	77.2	73.3	80.7
:				
114	Worth	78.7	75.3	81.8
115	Wright	75.5	72.7	78.2

Make R create the histogram of the Total variable.

```
> hist (Total)
```

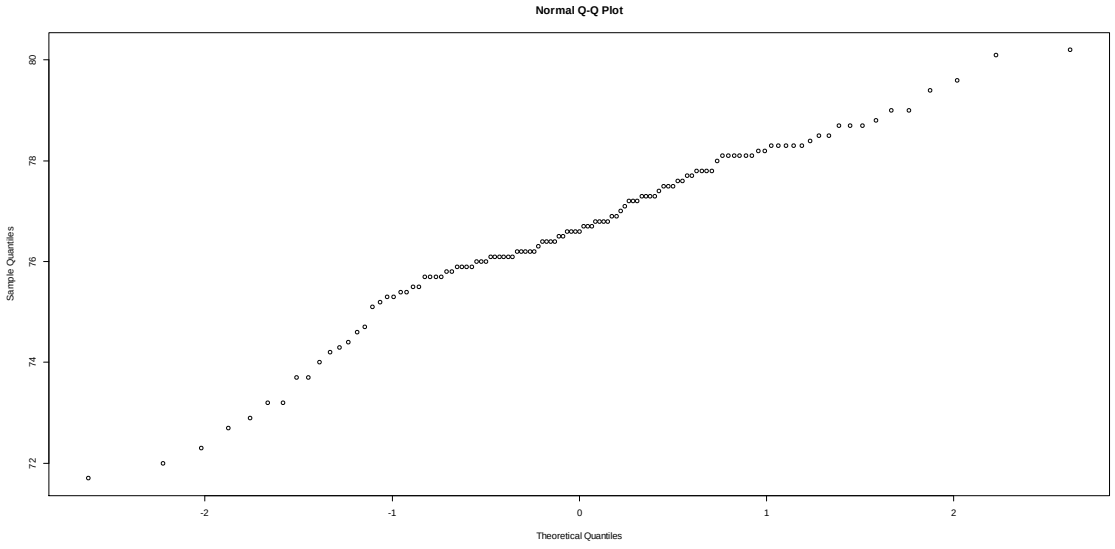


The histogram appears to be slightly skewed left, but not too badly.

Now make R create the normal probability plot of the variable Total. This requires two commands. The first command will show just the data points in the graph below. The second command will make R superimpose the line that corresponds to a perfect normal distribution on top of the plot.

```
> qqnorm (Total)
```

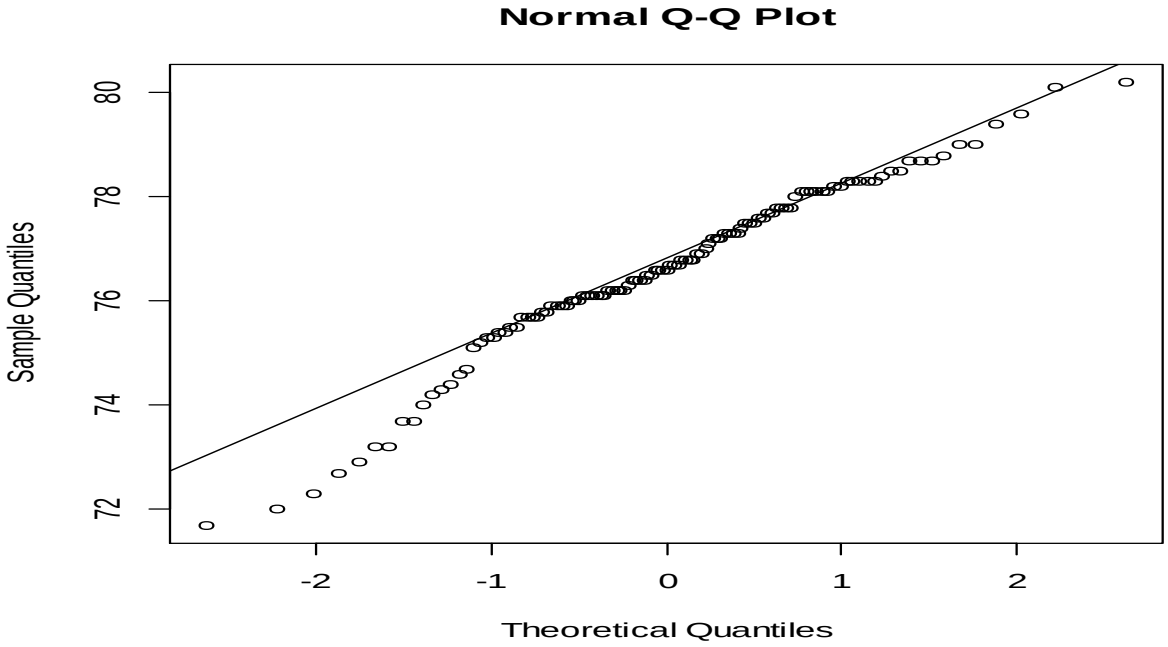
After this first command, the graph looks like the following.



Now the second command will superimpose the line on this graph.

```
> qqline (Total)
```

The resulting graph, showing both the data and the line, is a normal QQ-plot.



At first glance, the line looks similar to the one that you probably think of as the graph of "y=x." However, if you look at the axis labels, the plot shows sample quantiles vs. theoretically normal quantiles. Therefore, normally distributed data should fit snugly around this line.

Observe how well your data does so. This plot looks reasonably normal, although the life expectancies on the lower end fall a bit below where they should in a normal distribution. This could mean that some counties have something influencing them that causes them to differ from normal (e.g., very poor access to health care, proximity to an environmental problem such as lead or radioactive materials, substandard nutrition, etc.). You cannot determine a reason from the data available in this data set.

Section 11: How to Get Normal Critical Values for Common Significance Levels
(Uses no data files)

There are certain common levels of significance used in hypothesis tests, commonly denoted by the Greek letter alpha (α). The most common is α = 0.05; other common values are 0.01 and 0.10. Corresponding to any specific value for α, there is a critical value. When you are working with a normal distribution, a critical value is a z-score that defines the boundary between occurrences that "rare" and those that are "not rare." (Critical values may also be obtained when you are using other distributions; see sections 12 and 13 for other cases.)

To obtain a critical value for a standard normal distribution, you use: " qnorm (area to the left, 0, 1)." The "0" and"1" are the mean and standard deviation for the standard normal distribution, respectively.

To obtain two-tailed standard normal critical values, assign the value for α, then proceed as shown. This example uses α = 0.05. Since you want two-tailed critical values, α is divided by two.

For the lower critical value, you want the area to the left to be α/2. So you type the following.

```
> Alpha = 0.05
> LowerCV = qnorm (Alpha/2, 0, 1)
> LowerCV
```

Then R displays the lower critical value.

```
-1.959964
```

Now get the upper critical value. For the upper critical value, the area to the right should be α/2, so the area to the left is 1 – α/2.

```
> UpperCV = qnorm (1-Alpha/2, 0, 1)
> UpperCV
```

Then R displays the upper critical value.

```
1.959964
```

Try a new level of significance, say α = .01. You only need to change the line that assigns the value of α.

```
> Alpha = .01
> LowerCV = qnorm (Alpha/2, 0, 1)
> LowerCV
```

The lower critical value is displayed.

```
-2.575829
```

Now get the upper critical value.

```
> UpperCV = qnorm (1-Alpha/2, 0, 1)
> UpperCV
```

R displays the following upper critical value.

```
2.575829
```

What about one-tailed critical values?

Suppose you want just an upper critical value with all of α in the upper tail. As before, first specify the value you want to use for α. Then you use the command above for "UpperCV" except that you do not divide α by 2.

Finally, suppose you want just a lower critical value with all of α in the lower tail. Use the command above for "LowerCV" but do not divide α by 2.

NOTE: If you are working with a non-standard normal distribution, it will have either a different mean, a different standard deviation, or both. In that case, put its mean in place of the "0" and its standard deviation in place of the "1." The commands are otherwise the same.

Section 12: How to Generate a t-Distribution Graph and Obtain t-Critical Values
(Uses no data files)

Another distribution that you will need frequently is the Student's-t distribution. This looks almost like a normal distribution but is more variable. It requires that you enter a parameter called the degrees of freedom. It is traditionally abbreviated as "df." This value is dependent on the sample size, but can vary depending upon exactly what you are testing.

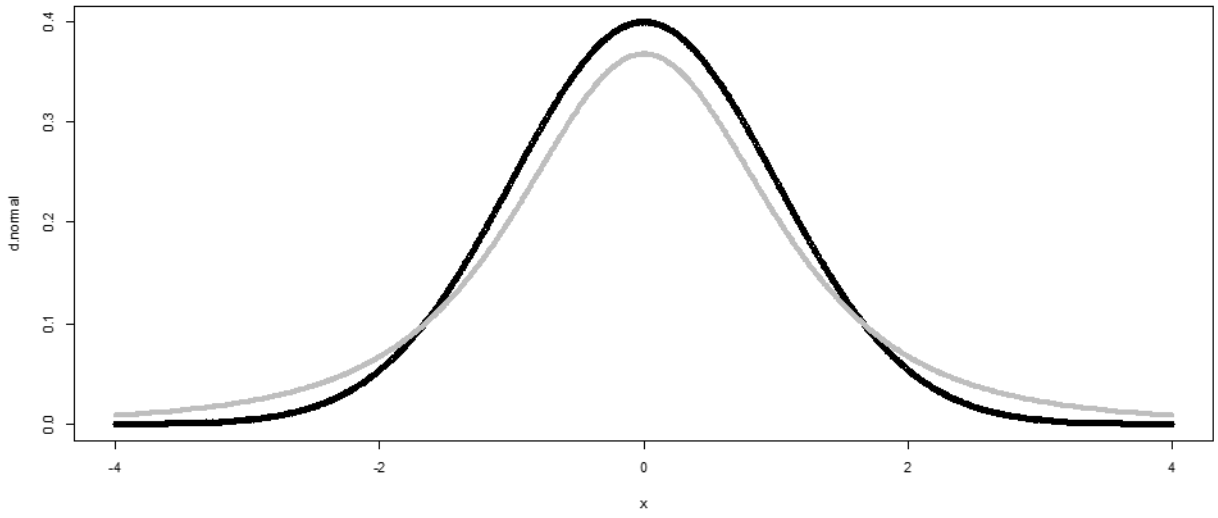
For the example below, the degrees of freedom parameter (named degreefree) is set equal to three. The random variable x is along the horizontal axis, which is scaled from -4 to 4 in increments of .005.

The normal distribution shows as the heavy black line; the t-distribution with df=3 is the thin gray line. You access the normal distribution with "dnorm (x)" and the t-distribution with "dt (x, degreefree)." These are assigned to the variables names d.normal and d.tdist, respectively, which can then be used in the plots.

Note: in the last line of the code, setting "lwd=5" controls the thickness of the gray curve. You can play with changing this number if you want the actual curve to be plotted with a thicker or thinner line.

```
> # The first three lines plot the standard normal curve (in black)
> x = seq(-4, 4, .005)
> d.normal = dnorm (x)
> plot (x, d.normal)
> # The next three lines add the t-distribution curve (in gray)
> degreefree = 3
> d.tdist = dt (x, degreefree)
> lines (x, d.tdist, col="gray", lwd=5)
```

Here is the graph that results.



The fact that the t-distribution is more variable than the normal can be seen in the graph. The tails of the t-distribution (shown in gray) are "fatter" and the hump in the center is lower than it is in the normal (shown in black). If you do more plots with the df-values getting larger, the t-distribution graph approaches the same shape as the normal.

There are certain common levels of significance used in hypothesis tests, commonly denoted by the Greek letter alpha (α). The most common is α = 0.05; other common values are 0.01 and 0.10. Corresponding to any specific value for α, there is a critical value. When you are using the t-distribution, a critical value is a t-score that defines the boundary between occurrences that "rare" and those that are "not rare."

If you want to obtain critical values of the t-distribution, you specify alpha (α) as you did for the normal, and also a value for the degrees of freedom. The general syntax is: "qt (area to left of critical value, df)." Therefore, if you want α in the left tail, then you use α for the area to the left. If you want α in the right tail, you use 1 – α as the area to the left.

For example, suppose you want a critical value with all of α = .05 in the upper tail and df = 3. Since α is the area to the right of the critical value, then 1 – α = .95 is the area to its left. So the command is:

```
> UCV = qt (.95, 3)
> UCV
```

Then R returns the upper critical value.

```
2.353363.
```

The critical values returned by R match those in the tables supplied with most textbooks (but carry more decimal places). The advantage of using R to obtain critical values is that the tables only cover a few values of α and degrees of freedom, but R can handle others that are not included in the tables.

Section 13: How to Generate Chi-Square and F Distributions and Obtain Their Critical Values

(Uses no data files)

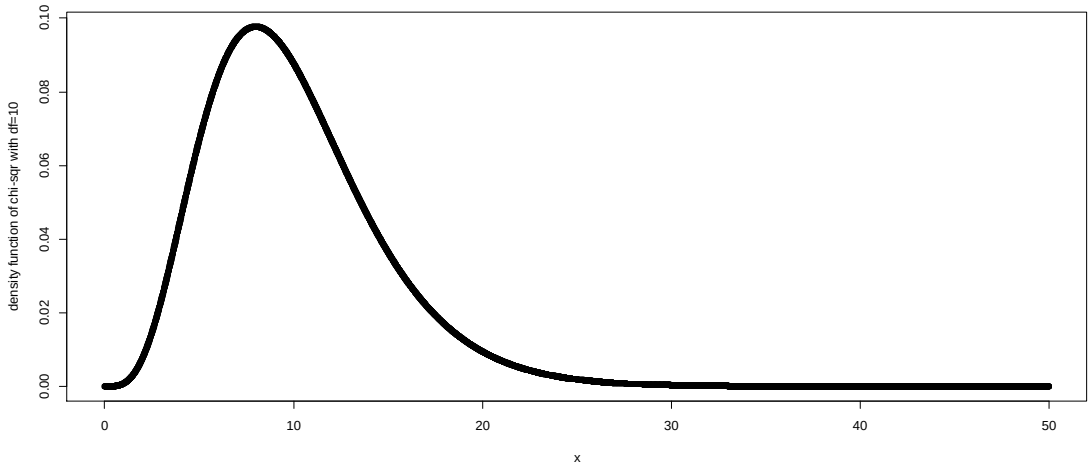
Two distributions that you will probably need are the chi-square distribution and the F-distribution. The chi-square distribution is used in some non-parametric tests and also in testing hypotheses about a variance. The F-distribution is used in ANOVA and in testing hypotheses that compare two variances.

Both require you to supply values for degrees of freedom. For the F-distribution, you will actually need to supply two df values – one goes with the numerator of a fraction involved in the testing process, and the other goes with the denominator. For now, the df-values will be randomly selected for purposes of illustration.

First consider the chi-square distribution; here is a typical graph. You need three commands to produce it. The first command scales the random variable x, on the horizontal axis, from 0 to 50 in increments of .005. The second command accesses the chi-square distribution using "dchisq (x, df)" -- for this example, using df = 10. In the example, the result is assigned to the variable chidist, which is then used in the third command for the vertical axis in the plot.

```
> x = seq (0, 50, .005)
> chidist = dchisq (x, 10)
> plot (x, chidist, ylab = "density function of chi-sqr with df=10")
```

The resulting graph is:



There are certain common levels of significance used in hypothesis tests, commonly denoted by the Greek letter alpha (α). The most common is α = 0.05; other common values are 0.01 and 0.10. Corresponding to any specific value for α, there is a critical value. When you are using a chi-square distribution, a critical value is a chi-square score that defines the boundary between occurrences that "rare" and those that are "not rare."

You can obtain critical values of a chi-square distribution using "qchisq (area to the left, df)." For example, if you want the one-tailed, lower critical value that cuts off α=.025 in the lower tail of the distribution in the graph above, the command would be as show below.

```
> LCV = qchisq (.025, 10)
> LCV
```

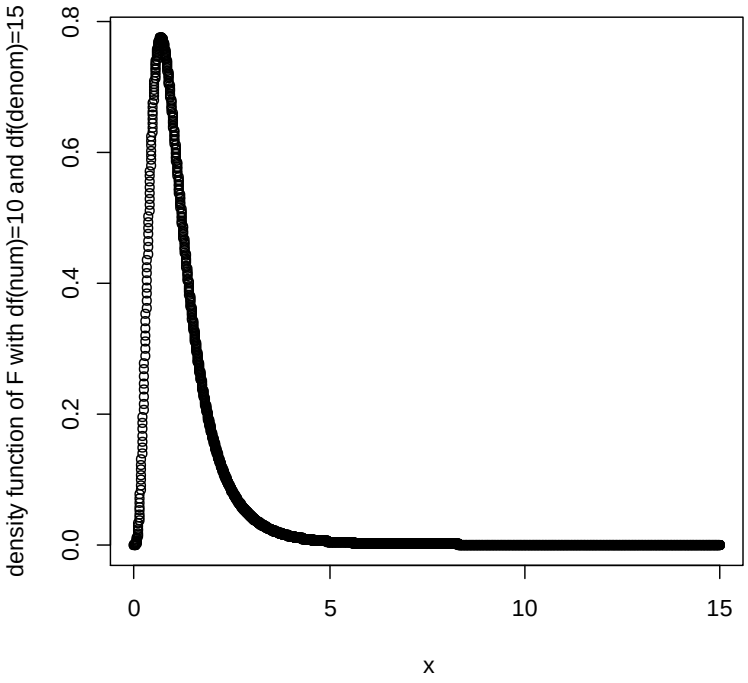
The critical value that results is:

```
 3.246973
```

The F distribution can be handled similarly. For example, the following commands plot an F-distribution with degrees of freedom 10 and 12. The first number, 10 in this example, is the number of degrees of freedom for the numerator. The second number, 12 in this example, is the number of degrees of freedom for the denominator. The random variable x is on the horizontal axis, scaled from 0 to 15 in increments of .005. The F distribution is accessed using "df (x, 10, 12)." The results are assigned to the variable Fdist, which is then used in the plot.

```
>  x = seq (0, 15, .005)
>  Fdist = df (x, 10, 12)
> plot (x, Fdist, ylab = "density function of F with df(num)=10 and df(denom)=15")
```

The resulting graph is:



Critical values may be obtained with a similar process to that used before. For example, to get two-tailed critical values of the distribution in the graph with $\alpha = .05$, split α in half to account for the two tails. This means you would put an area of .025 in each tail. Consequently, the upper critical value is obtained by using 1-.025=.975 as the area to its left. So you do the following.

```
>UCV = qf (.975, 10, 12)
>UCV
```

R returns the following upper critical value.

```
3.373553
```

Similarly, you enter the following two lines to get the lower critical value.

```
>LCV = qf (.025, 10, 12)
>LCV
```

R then produces the following lower critical value.

```
0.276171
```

Section 14: How to Generate and Graph a Binomial Distribution
(Uses no data files)

To generate a binomial distribution, you use the "dbinom ( )" command. Inside the parentheses, you will need three items: (1) the name of the random variable that you are using for the number of "successes," (2) the number of trials of the experiment, and (3) the probability of a success on any single trial. What R actually generates is the set of probabilities corresponding to the possible numbers of successes. The form of the command is:

```
> dbinom (variable name, size = no. of trials, prob = probability of a success on a single trial)
```

Here is an example.

Suppose twenty basketball players each take one turn at shooting a free throw. Based on previous attempts, the probability that an individual player will make the shot is 70%, or 0.7. Since there are twenty players, it is possible that none of them, all of them, or any number between, will make the shot. Assume that you want to obtain the probabilities corresponding to the numbers (0, 1, 2, …, 19, 20) of players successfully making the shot.

First create a random variable that stands for the number of successes. In this case, a success is a player making the shot. Have R assign the values 0 through 20 to this variable.

```
> y = 0:20
```

Next assign the probability for success (making the shot) to another variable.

```
> p = .7
```

Now you can use these variables to create the appropriate binomial distribution. The first command defines the specific binomial function to generate the distribution and assigns the probabilities to the variable Prob.y. The second command displays the results.

```
> Prob.y = dbinom (y,  size=20,  prob=p)
> Prob.y
```

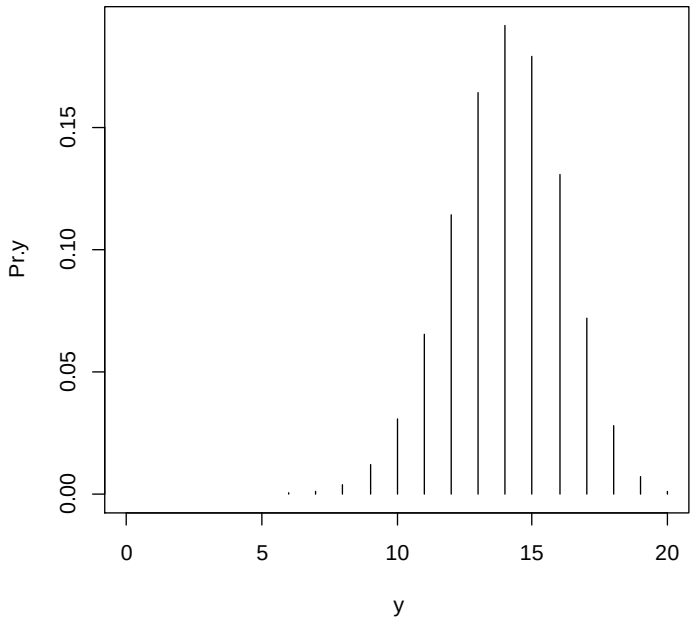
The output is the list of probabilities for 0 through 20 successes, in that order, as shown below. Note that these probabilities are in scientific notation, so the last one, for instance, means 7.97×10^{-4}.

```
3.486784e-11  1.627166e-09  3.606885e-08  5.049639e-07  5.007558e-06
3.738977e-05  2.181070e-04  1.017833e-03  3.859282e-03  1.200665e-02
3.081708e-02  6.536957e-02  1.143967e-01  1.642620e-01  1.916390e-01
1.788631e-01  1.304210e-01  7.160367e-02  2.784587e-02  6.839337e-03
7.979227e-04
```

Now you can graph the distribution if you wish. Use the "plot ( )" command. In parentheses, you need: (1) the name of the variable for the number of successes, (2) the name of the variable holding the probabilities you just generated, and (3) the subcommand: type="h").

```
> plot (y, Prob.y, type="h")
```

You get the following graph.



You can mentally, or with a pencil, draw a smooth curve over the upper ends of the bars to emphasize the shape of the distribution.