Normal Models

[Put your name here]

In this assignment we will learn how to work with normal models. We will also discuss z-scores and how to check that numerical data is "nearly normal" in its distribution.

Instructions

Presumably, you have already created a new project and downloaded this file into it. Please knit the document (probably to HTML as you're working) and work back and forth between this R Markdown file and the knit output to answer the following questions.

When you are finished with the assignment, knit to PDF, export the PDF file to your computer, and then submit to the corresponding assignment in Canvas. Be sure to submit before the deadline!

Sometimes you will be asked to add your own R code. That will appear in this document as a code chunk with a request for you to add your own code, like so:

```
## Add code here to [do some task]...
```

When you see that in a code chunk, you need to type some R code to complete a task.

Sometimes you will be asked to type up your thoughts. Instructions to do that will be labeled as follows. If you are currently reading this in the knit output, please look back at the R Markdown file to see the following text:

(When you knit the document, you can't see the text from the line above. That's because the crazy notation surrounding that text marks it as a "comment", and therefore it doesn't appear in the output.) In these areas of the assignment, please use contextually meaningful full sentences/paragraphs (unless otherwise indicated) and proper spelling, grammar, punctuation, etc. This is not R code (although it may include inline R code when necessary), but rather a free response section where you talk about your analysis and conclusions.

Getting started

Make sure you're in a project

If you're looking at this document, you should have already created a project and uploaded this R Markdown file to that project folder.

Save your file!

The first thing we **always** do is save our file. You'll probably want to save this under a new name. Go to the "File" menu and then "Save As". Once you've saved the file with the new name, from them on it's easier to just hit Ctrl-S (or Cmd-S on a Mac) to keep saving it periodically.

Remember that file names should not have any spaces in them. (In fact, you should avoid other kinds of special characters as well, like periods, commas, number signs, etc. Stick to letters and numerals and you should be just fine.) If you want a multiword file name, I recommend using underscores like this: `this_filename_has_spaces_in_it`.

Load Packages

We load the standard `mosaic` package.

```
library(mosaic)
```

The Central Limit Theorem

An important aspect of all the simulations that we've done so far is that their histograms all look like bell curves. Under some basic assumptions that we'll discuss in a later assignment, this will be typical of our simulated null distributions.

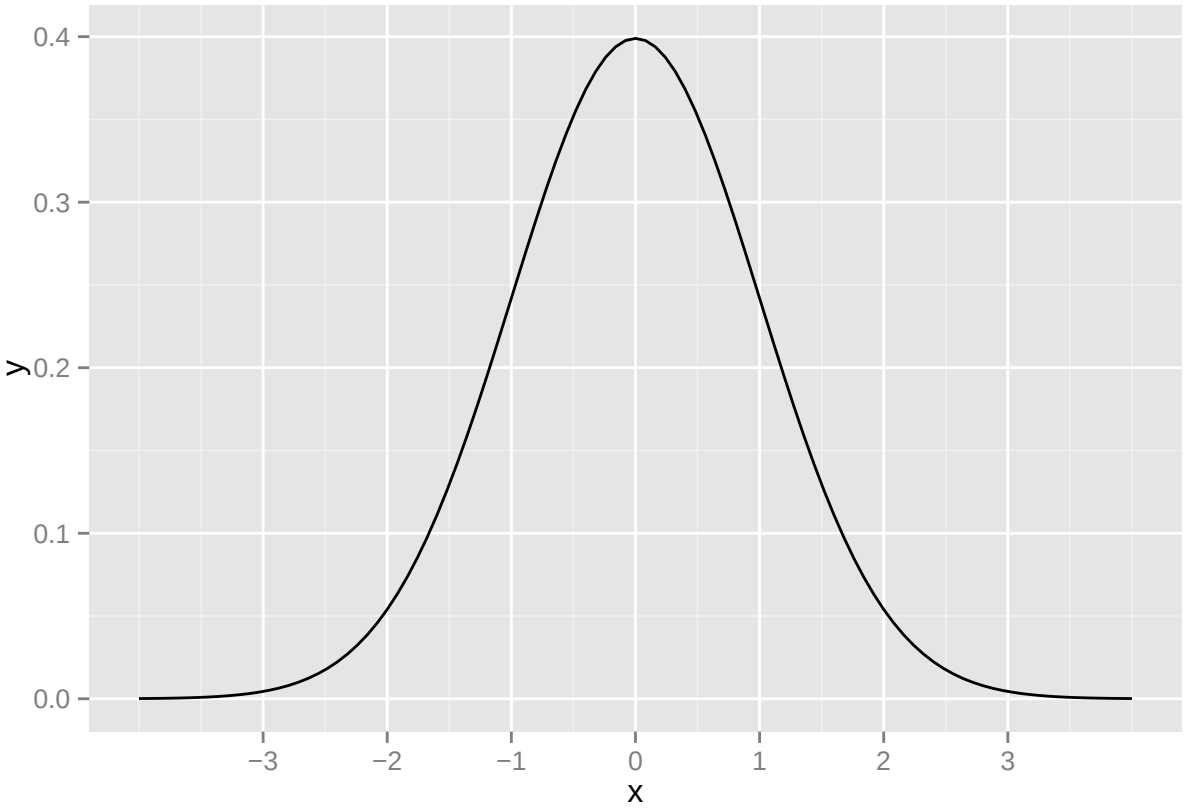
So rather than running a simulation each time we want to conduct a hypotheses test, we could also assume that the null distribution *is* a bell curve. The rest of this assignment will teach you how to work with the "normal distribution", which is just the mathematically correct term for a bell curve.

Normal models

The normal distribution looks like this:

```
## Don't worry too much about the syntax here.
## You won't need to know how to do this on your own.

qplot(c(-4, 4), stat = "function", fun = dnorm,
      geom = "line", xlab = "x") +
    scale_x_continuous(breaks = -3:3)
```



The curve pictured above is called the *standard normal distribution.* It has a mean of 0 and a standard deviation of 1. Mathematically, this is often written as

$$N(\mu = 0, \sigma = 1),$$

or sometimes just

$$N(0, 1).$$

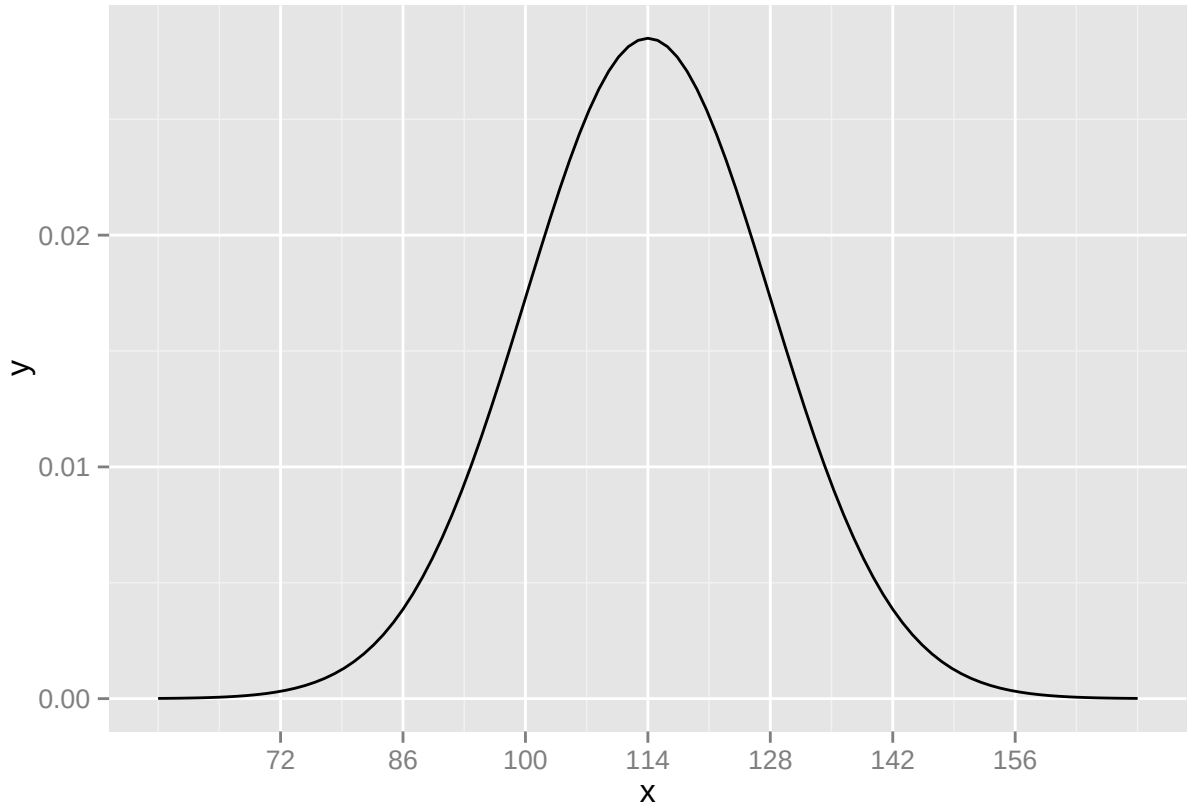
We use this bell curve shape to model data that is unimodal, symmetric, and without outliers. A statistical "model" is a simplification or an idealization. Reality is, of course, never perfectly bell-shaped. Real data is not exactly symmetric with one clear peak in the middle. Nevertheless, an abstract model can give us good answers if used properly.

As an example of this, systolic blood pressure (SBP, measured in millimeters of mercury, or mmHg) is more-or-less normally distributed in women ages 30–44 in the U.S. and Canada, with a mean of 114 and a standard deviation of 14 (from the World Health Organization).

If we were to plot a histogram with the SBP of every woman between the ages of 30 and 44 in the U.S. and Canada, it would have the shape of a normal distribution, but instead of being centered at 0 like the graph above, this one would be centered at 114. Mathematically, we write

$$N(\mu = 114, \sigma = 14).$$

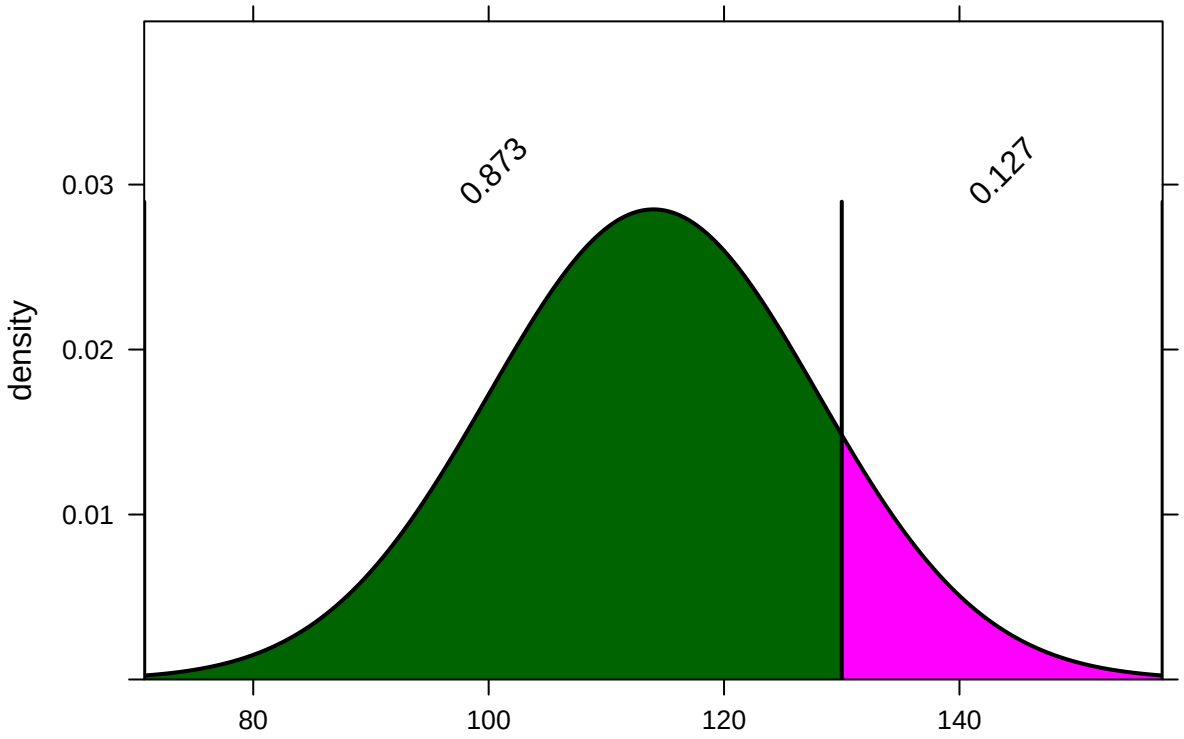
```
qplot(c(58, 170), stat = "function", fun = dnorm,
      args = list(mean = 114, sd = 14), geom = "line", xlab = "x") +
    scale_x_continuous(breaks = c(72, 86, 100, 114, 128, 142, 156))
```



Using this information, we can estimate the percentage of such women who are expected to have any range of SBP without having access to all such data.

For example, what percentage of women ages 30–44 in the U.S. and Canada are expected to have SBP under 130 mmHg? The `pdist` command from the `mosaic` package will not only help you with this calculation, but it also offers a nice visual representation.

```
pdist(dist = "norm", q = 130, mean = 114, sd = 14)
```




```
## [1] 0.873451
```

If you don't need the pretty picture, it's a little easier just to use the `pnorm` function.

```
pnorm(q = 130, mean = 114, sd = 14)
```

```
## [1] 0.873451
```

We illustrate this inline:

The model predicts that 87.3451046% of women ages 30–44 in the U.S. and Canada will have systolic blood pressure under 130 mmHg.

(Ignore the ridiculous number of decimal places that R reports. It's possible to change this, but it's a hassle.)

How many women are predicted to have SBP greater than 130? If we count women first with SBP under 130 mmHg, and then count women with SBP over 130 mmHg, that's all women! In other words, all the women have to add up to 100%. Therefore, all we have to do to solve this problem is subtract the number we obtained in the previous question from 1. (Remember that 1 = 100%.)

```
1 - pnorm(q = 130, mean = 114, sd = 14)
```

```
## [1] 0.126549
```

The model predicts that 12.6548954% of women ages 30–44 in the U.S. and Canada will have systolic blood pressure over 130 mmHg.

Now, here's a more complicated question. What percentage of women are predicted to have SBP between 110 mmHg and 130 mmHg?

Recall that the proportion of women predicted to have SBP less than 130 mmHg was 0.873451. But this is also counting women with SBP under 110 mmHg, whom we now want to exclude. The proportion of women with SBP under 110 is 0.3875485. All we have to do, then, is subtract 0.3875485 from 0.873451:

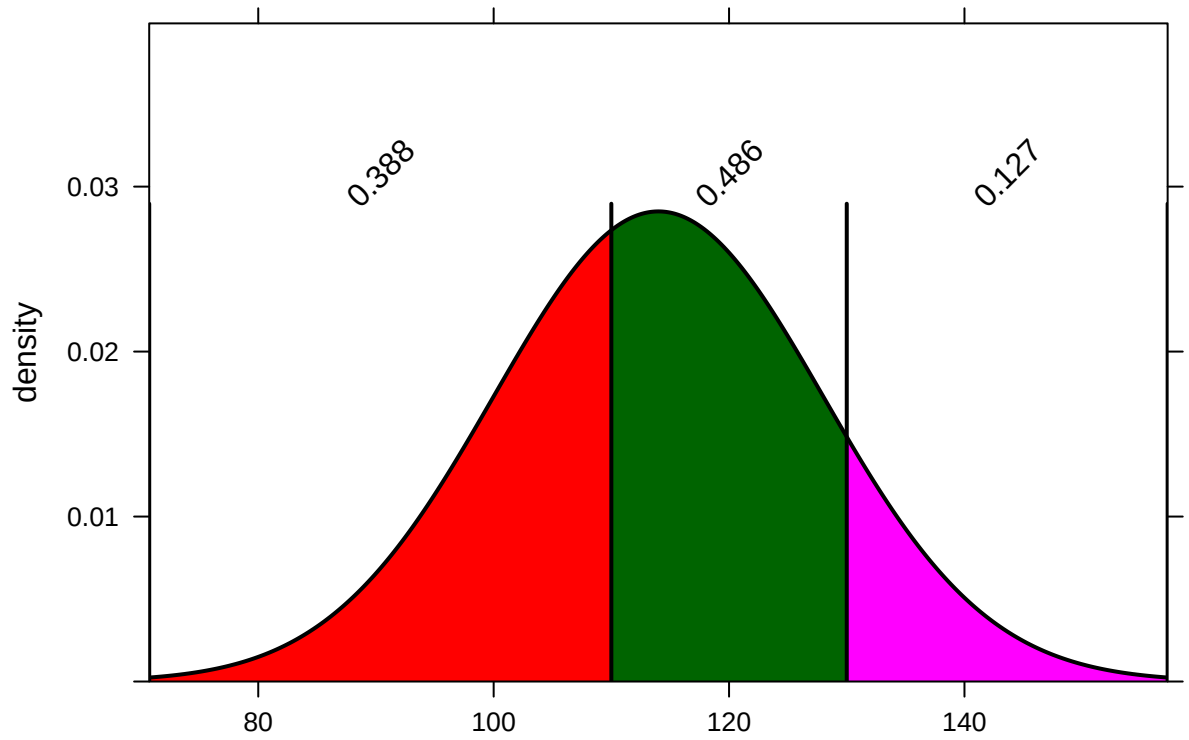
```
pnorm(q = 130, mean = 114, sd = 14) - pnorm(q = 110, mean = 114, sd = 14)
```

```
## [1] 0.4859026
```

The model predicts that 48.5902564% of women ages 30–44 in the U.S. and Canada will have systolic blood pressure between 110 mmHg and 130 mmHg.

The `pdist` command handles this by including both values (110 and 130):

```
pdist(dist = "norm", q = c(110, 130), mean = 114, sd = 14)
```




```
## [1] 0.3875485 0.8734510
```

Notice that the picture is more meaningful than the output below the picture.

Now you try!

IQ scores are standardized so that they have a mean of 100 and a standard deviation of 16.

For each of the following questions, use the `pdist` command to draw the right picture and then use `pnorm` to state your answer in a contextually meaningful full sentence. Don't forget to use the phrase "The model predicts" and report numbers as percentages, not decimals.

Question: What percentage of people would you expect to have IQ scores over 80?

```
## Add code here to draw the model.
```

Question: What percentage of people would you expect to have IQ scores under 90?

```
## Add code here to draw the model.
```

Question: What percentage of people would you expect to have IQ scores between 112 and 132?

```
## Add code here to draw the model.
```

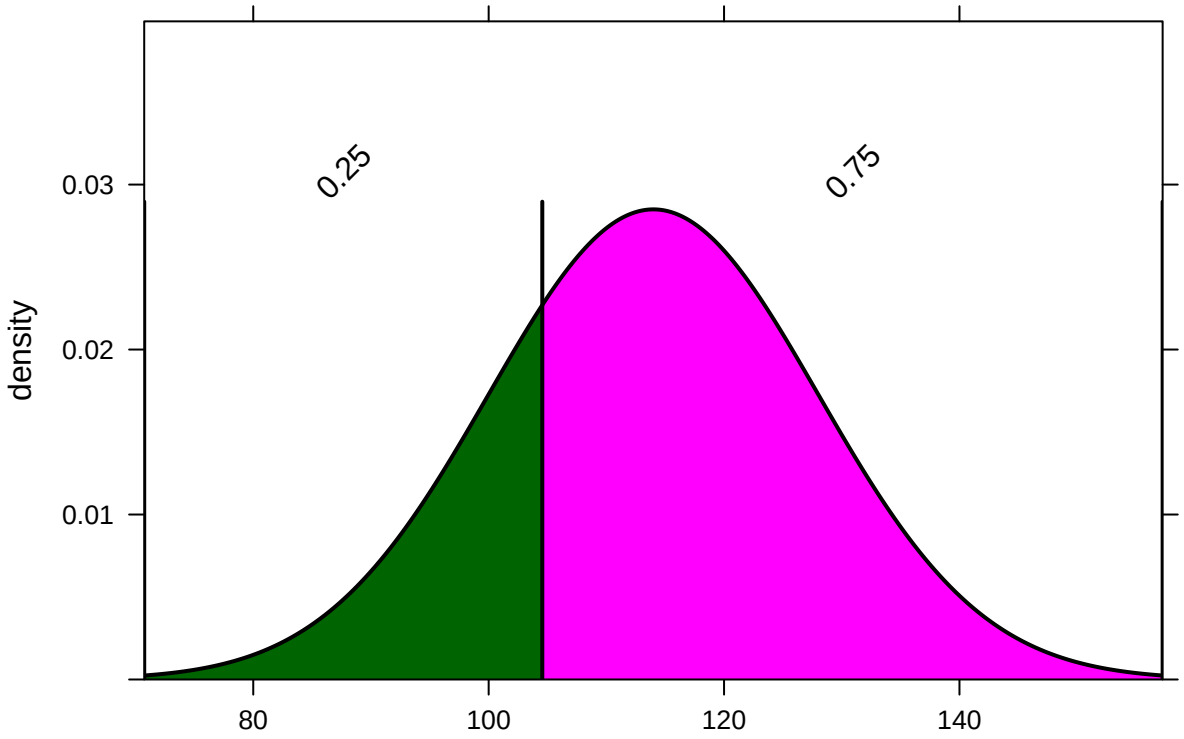
Percentiles

Often, the question is reversed: instead of getting a value and being asked what percentage of the population falls above or below it, we are given a percentile and asked about the value to which it corresponds.

Here is an example using systolic blood pressure. What is the cutoff value of SBP for the lowest 25% of women ages 30–44 in the U.S. and Canada? In other words, what is the 25th percentile of SBP for this group of women?

The commands we need are `qdist` and `qnorm`. They look a lot like `pdist` and `pnorm`. Observe:

```
qdist(dist = "norm", p = 0.25, mean = 114, sd = 14)
```

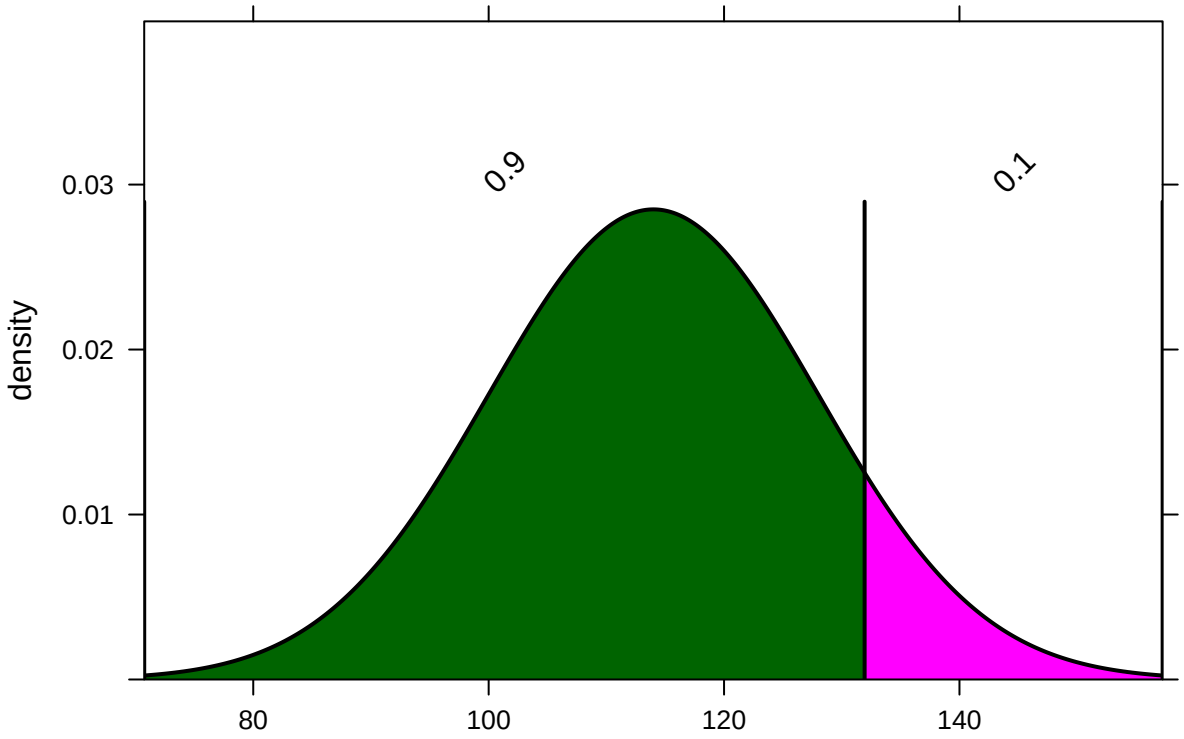



```
## [1] 104.5571
```

The model predicts that the 25th percentile for SBP in women ages 30–44 in the U.S. and Canada is 104.5571435 mmHg.

What if we asked about the highest 10% of women? All you have to do is remember that the top 10% is actually the 90th percentile.

```
qdist(dist = "norm", p = 0.9, mean = 114, sd = 14)
```

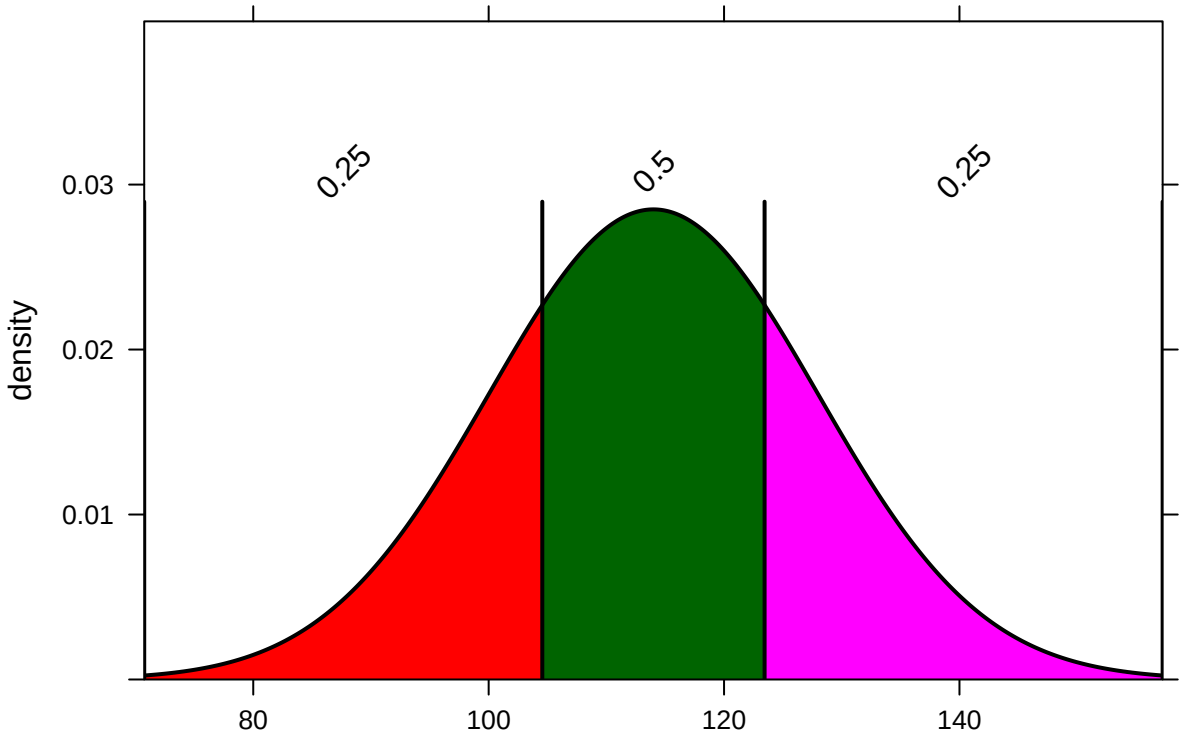



```
## [1] 131.9417
```

The model predicts that the top 10% of SBP in women ages 30–44 in the U.S. and Canada have SBP higher than 131.9417219 mmHg.

Finally, what if we want the middle 50%? This is trickier. The middle 50% lies between the 25th percentile and the 75th percentile. Observe the graph below.

```
qdist(dist = "norm", p = c(0.25, 0.75), mean = 114, sd = 14)
```




```
## [1] 104.5571 123.4429
```

Therefore, the model predicts that the middle 50% of SBP for women ages 30–44 in the U.S. and Canada lies between 104.5571435 mmHg and 123.4428565 mmHg.

Now you try!

We'll continue to use IQ scores (mean of 100 and standard deviation of 16).

For each of the following questions, use the `qdist` command to draw the right picture and then use `qnorm` to state your answer in a contextually meaningful full sentence. Don't forget to use the phrase "The model predicts".

Question: What cutoff value bounds the highest 5% of IQ scores?

```
## Add code here to draw the model.
```

Question: What cutoff value bounds the lowest 30% of IQ scores?

```
## Add code here to draw the model.
```

Question: What cutoff values bound the middle 80% of IQ scores?

```
## Add code here to draw the model.
```

z-scores

Sometimes it is easier to refer to a value in terms of how many standard deviations it lies from the mean. For example, a systolic blood pressure of 100 is 14 mmHg below the mean (114 mmHg), but since the standard deviation is 14 mmHg, this means that 100 is one standard deviation below the mean. This distance from the mean in terms of standard deviations is called a *z-score*.

We calculate z-scores using the following formula:

$$z = \frac{x - \mu}{\sigma}.$$

In our example, if we wanted to know the z-score for an SBP of 100, we just plug all the numbers into the formula above:

$$z = \frac{100 - 114}{14} = -1.$$

What is the z-score for an SBP of 132? In the graph of the nomal model $N(\mu = 114, \sigma = 14)$, we can see that 132 lies between 128 and 142, which are 1 and 2 standard deviations above the mean, respectively. The exact z-score is

$$z = \frac{132 - 114}{14}$$

which comes out to 1.2857143.

The `scale` function from R also computes z-scores. Just note that the function takes arguments `center` and `scale`, not `mean` and `sd`.

```
scale(x = 100, center = 114, scale = 14)
```

```
##      [,1]
## [1,]   -1
## attr(,"scaled:center")
## [1] 114
## attr(,"scaled:scale")
## [1] 14
```

```
scale(x = 132, center = 114, scale = 14)
```

```
##          [,1]
## [1,] 1.285714
## attr(,"scaled:center")
## [1] 114
## attr(,"scaled:scale")
## [1] 14
```

Also note that the function spits about a bunch of crap we don't care about. This goes away if you use the command inline:

The z-score for 100 is -1 and the z-score for 132 is 1.2857143.

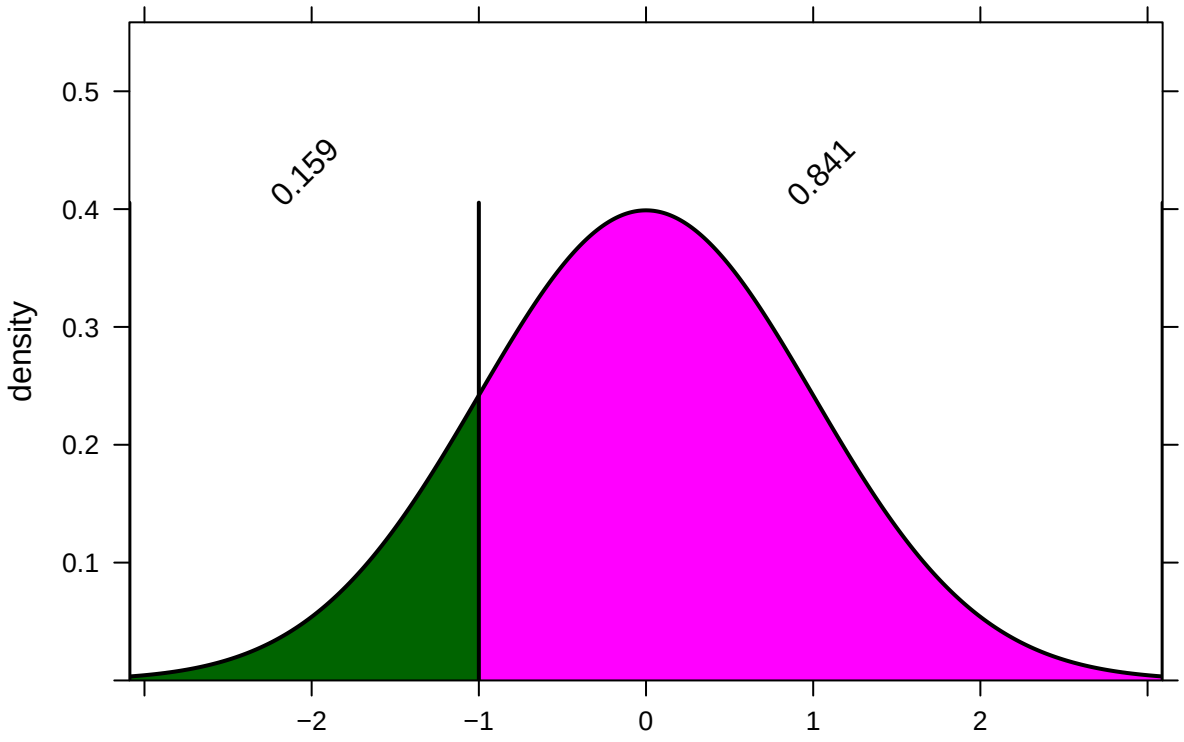
Now you try!

If IQ scores have a mean of 100 and a standard deviation of 16, what are the z-scores for the following IQ scores?

- 80
- 102
- 130

If you are working with z-scores, that also makes the commands easier when working with the normal model. The default settings for `pdist`, `pnorm`, `qdist`, and `qnorm` is `mean = 0` and `sd = 1`. That saves you some typing. So, for example, we calculated above that an SBP of 100 has a z-score of -1. What percentage of women are expected to have SBP lower than 100?

```
pdist(dist = "norm", q = -1)
```




```
## [1] 0.1586553
```

The model predicts that 15.8655254% of women ages 30–44 in the U.S. and Canada will have SBP less than 100.

Question: Albert Einstein supposedly had an IQ of 160. Calculate the z-score for his IQ and then use that z-score to figure out what percentage of the population is smarter than Einstein.

QQ plots

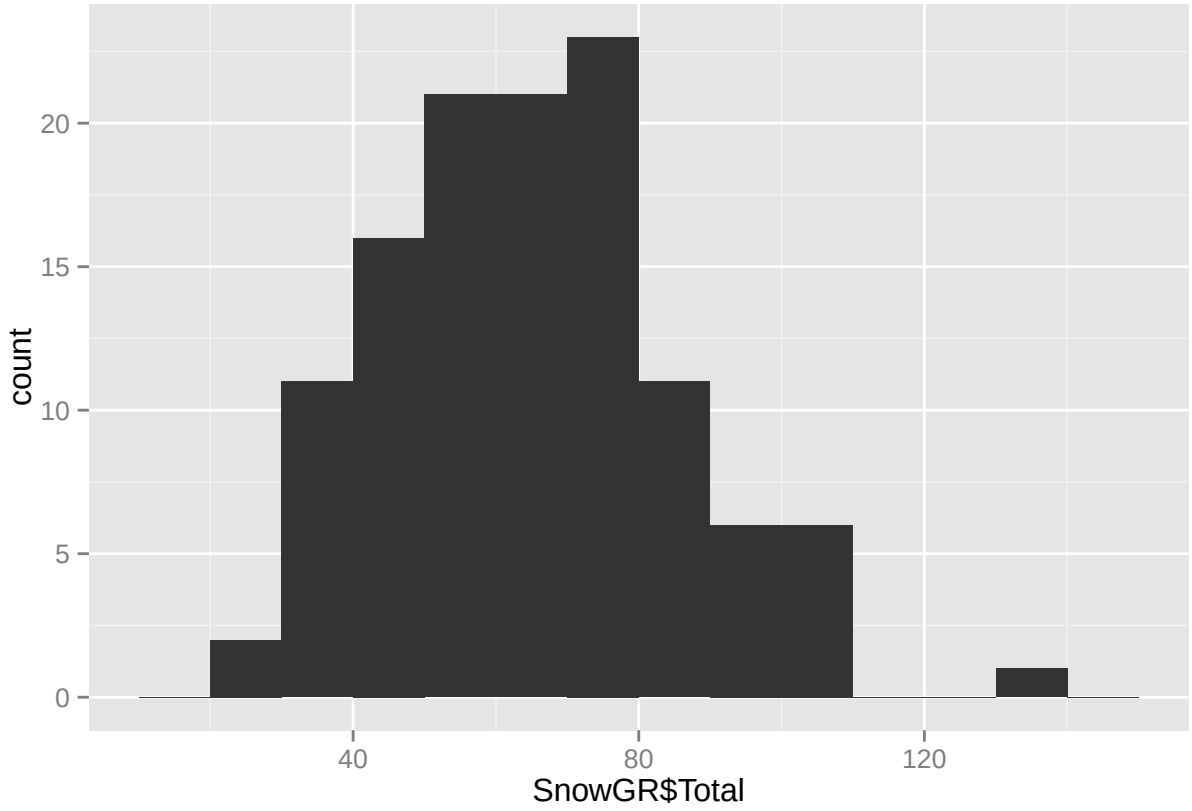
All of the work we do with normal models assumes that a normal model is appropriate. When we want to summarize data using a normal model, this means that the data distribution should be reasonably unimodal, symmetric, and with no serious outliers.

We can, of course, use a histogram to check this. But a histogram can be highly sensitive to the choice of bins. Furthermore, for small sample sizes, histograms look "chunky", making it hard to test this assumption.

An easier way to check normality is to use a *quantile-quantile plot*, typically called a *QQ plot* or a *normal probability plot*. We won't get into the technicalities of how this plot works. Suffice it to say that if data is normally distributed, the points of a QQ plot should lie along a diagonal line.

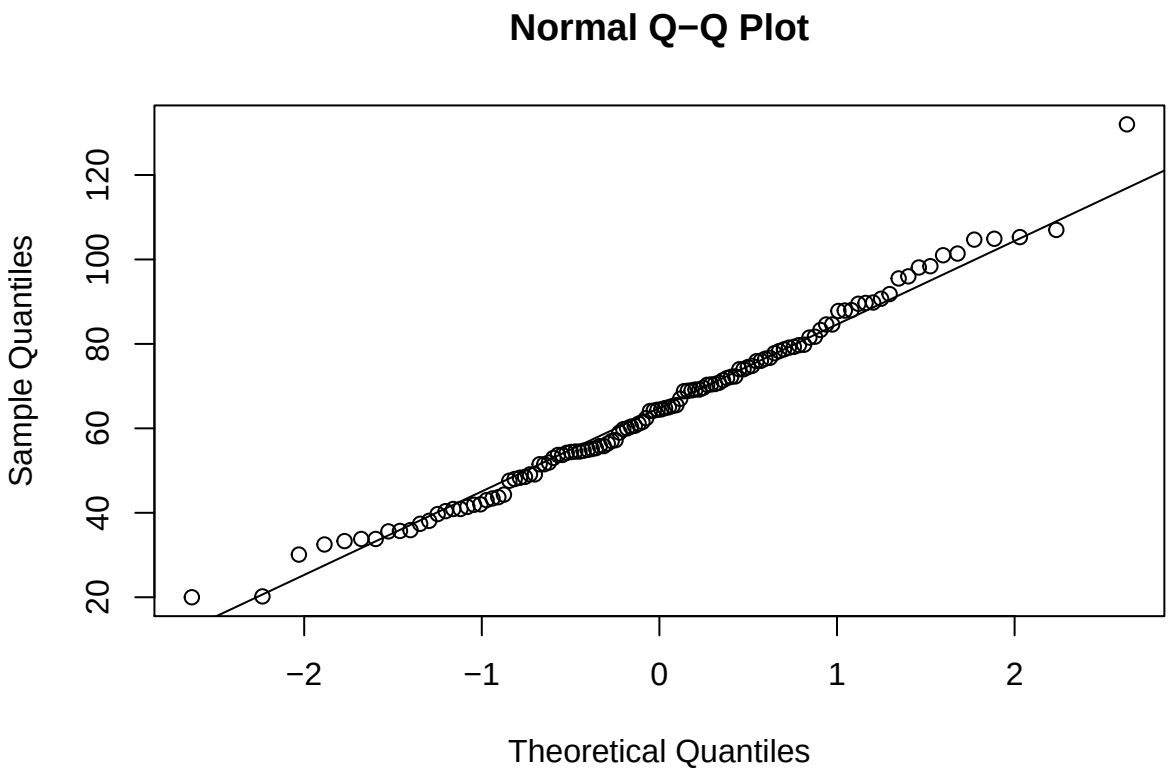
Here is an example. The total snowfall in Grand Rapids, Michigan has been recorded every year since 1893. A histogram (with reasonable binning) shows that the data is nearly normal.

```
qplot(SnowGR$Total, binwidth = 10)
```



Here is the QQ plot for the same data. The `qqnorm` command creates the actual QQ plot. The `qqline` commands just adds a reference line to make it easy to see whether the dots are truly lined up.

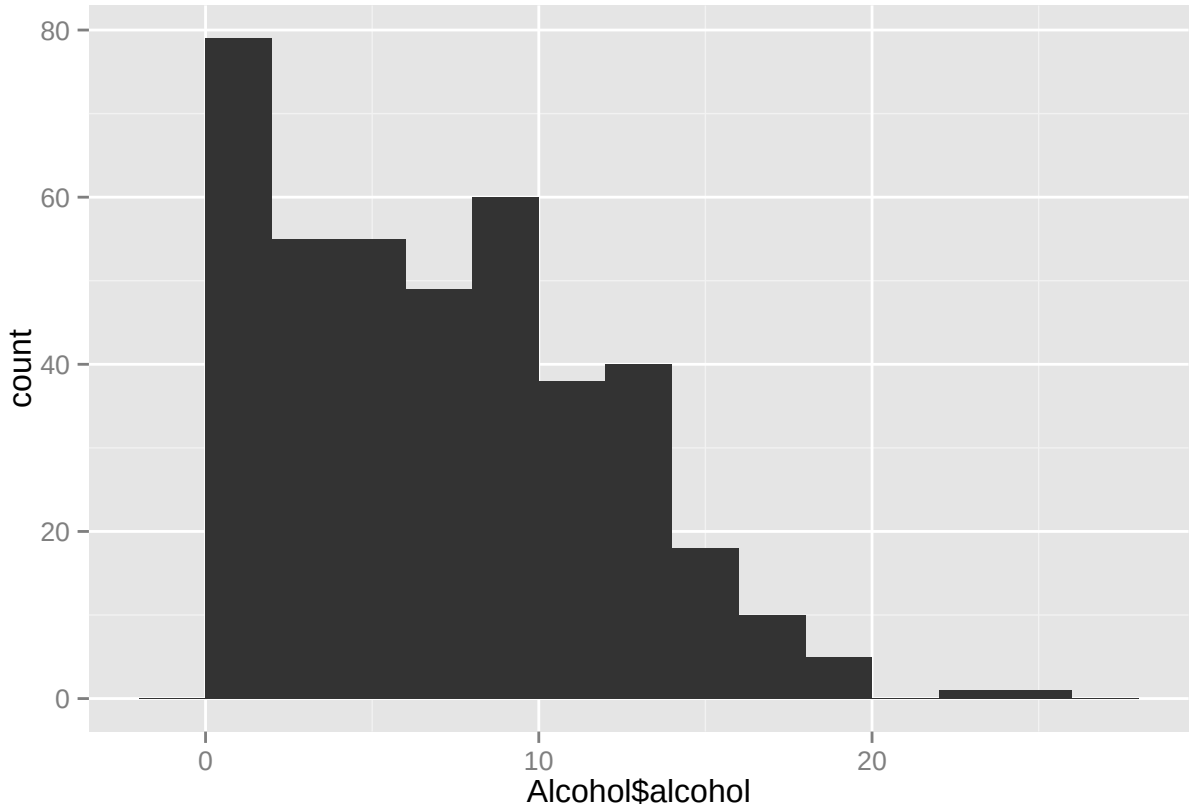
```
qqnorm(SnowGR$Total)
qqline(SnowGR$Total)
```



Other than a few points here and there, the bulk of the data is lined up nicely.

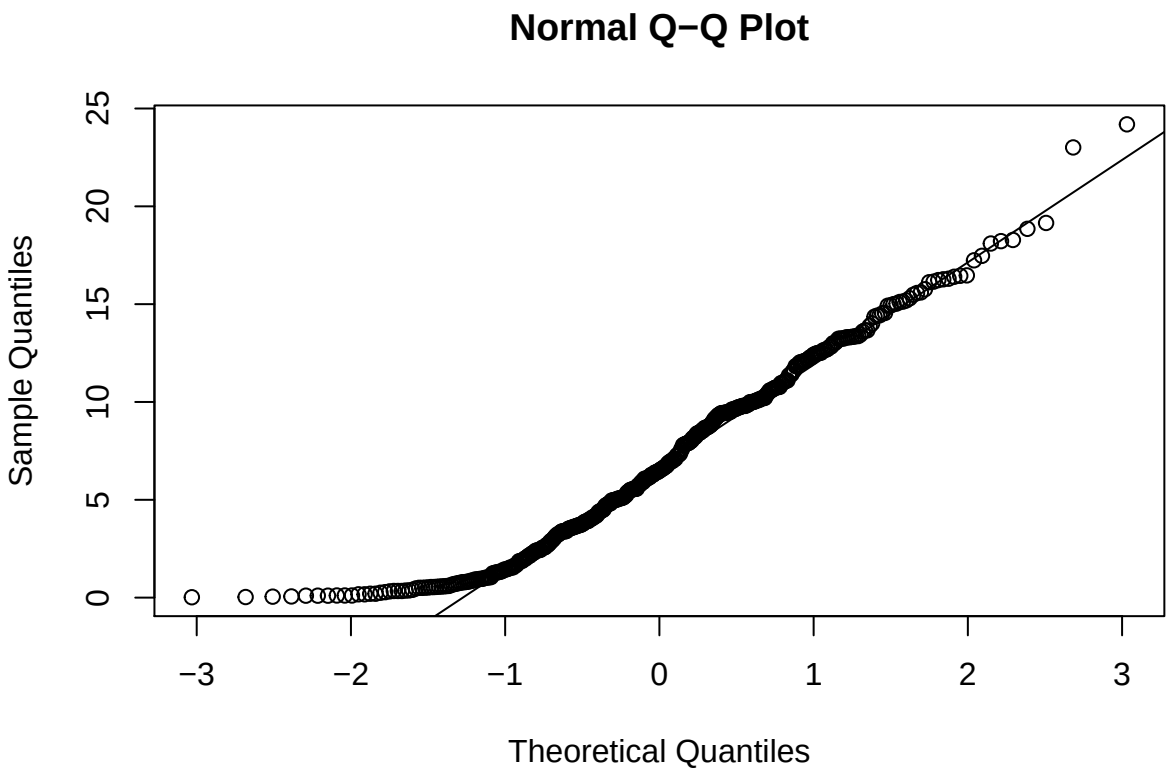
Contrast that with skewed data. For example, the `Alcohol` dataset contains per capita consumption (in liters) of alcohol for various countries over several years. The alcohol comsumption variable is highly skewed, as one can see in the histogram.

```
qplot(Alcohol$alcohol, binwidth = 2)
```



It is also apparent in the QQ plot that the data is not normally distributed.

```
qqnorm(Alcohol$alcohol)
qqline(Alcohol$alcohol)
```



The path of dots is sharply curved, indicating a lack of normality.

Now you try!

Find a data set with a numerical variable that is nearly normal in its distribution. (It can be something we've already seen in a past assignment, or if you're really ambitious, you're welcome to find a new data set.) Plot both a histogram and a QQ plot to demonstrate that the data is nearly normal. No need for a written response. Just plot the graphs.

```
## Add code here to plot a histogram and a QQ plot.
```

Now find a data set with a numerical variable that is skewed in its distribution. Plot both a histogram and a QQ plot to demonstrate that the data is not nearly normal. Again, no need for a written response. Just plot the graphs.

```
## Add code here to plot a histogram and a QQ plot.
```