Introduction to Probability and Information Theory
Homework #4

Verzani pg. 47

#6.3
```
> x = sample(0:10,10,replace=TRUE)
> x
 [1] 0 2 7 3 1 8 7 1 9 3
> max(x)
[1] 9
> min(x)
[1] 0
```

#6.7
```
> qnorm(0.05)
[1] -1.644854
```

#6.8 There's no[1] function built in to R to answer this question directly, but we can solve it using two basic facts. First, the sum rule tells us that:

$$P(-z \leq Z \leq z) = P(-z \leq Z \leq 0) + P(0 < Z \leq z)$$

and

$$P(Z \leq z) = P(Z \leq 0) + P(0 < Z \leq z)$$

Second, the standard normal distribution is symmetric around zero, which means:

$$P(Z \leq 0) = P(Z > 0) = 0.5$$

and

$$P(-z \leq Z \leq 0) = P(0 < Z \leq z)$$

Putting these together, we get:

$$\begin{aligned} P(-z \leq Z \leq z) &= P(-z \leq Z \leq 0) + P(0 < Z \leq z) \\ &= 2 \times P(0 < Z \leq z) \\ &= 2 \times (P(Z \leq z) - P(Z \leq 0)) \\ &= 2 \times (P(Z \leq z) - 0.5) \\ &= 2 \times P(Z \leq z) - 1 \end{aligned}$$

So, to find the z such that $P(-z \leq Z \leq z) = 0.05$, we can find z such that:

$$\begin{aligned} 2 \times P(Z \leq z) - 1 &= 0.05 \\ P(Z \leq z) &= 0.525 \end{aligned}$$

Now we've got something R can solve for us:

[1] that I'm aware of

```
> qnorm(0.525)
[1] 0.06270678
```

And just to confirm that that answer makes sense, we can check:

```
> pnorm(qnorm(0.525))-pnorm(-qnorm(0.525))
[1] 0.05
```

Verzani pg. 53

#7.8 The library that has `simple.sim` is now called UsingR, and can be downloaded from CRAN, the sample place you downloaded R itself from. Following the instructions in the problem, we need to load the library, define the bootstrap function, load the `faithful` data set, and run the simulation:

```
> library(UsingR)
> bootstrap = function(data,n=length(data)) {
+   boot.sample=sample(data,n,replace=TRUE)
+   median(boot.sample)
+ }
> data(faithful)
> d = simple.sim(100,bootstrap,faithful[['eruptions']])
```

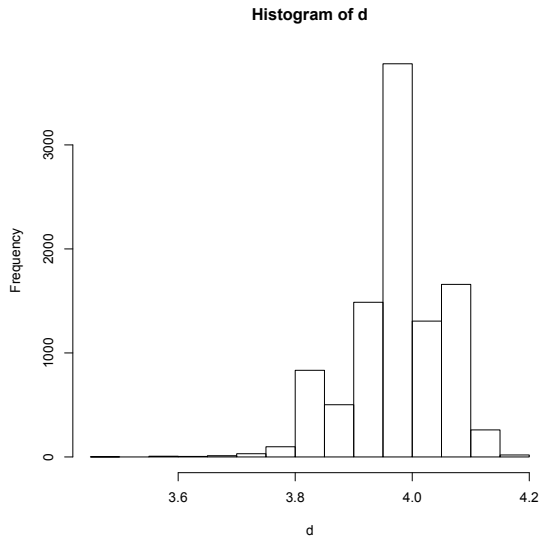
So now the question is: are the values in *d* normally distributed? We can check the summary statistics:

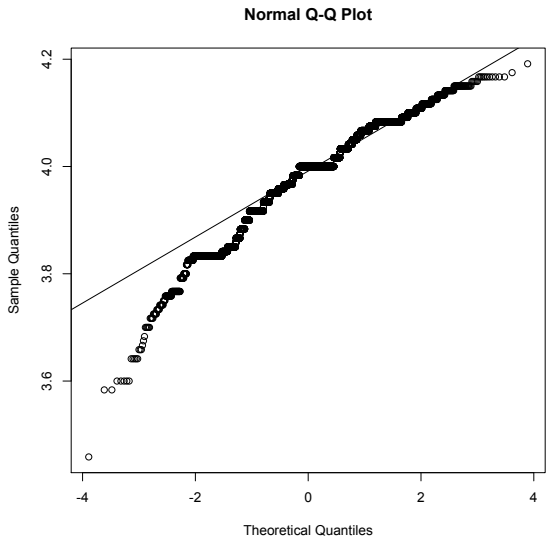
```
> summary(d)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  3.458   3.950   4.000   3.983   4.033   4.192
```

which suggest that they're not normally distributed. The normal distribution is symmetric, but there seems to be a wider range of values in *d* that fall below the mean than fall above it. We can also look at a histogram and a qq-plot for the data:

```
> hist(d)
> qqnorm(d)
> qqline(d)
```

which gives the result:





Both of these plot indicate the same thing as the summary statistics. The distribution of values in *d* is not normal, but is skewed toward the left (there are more small values and fewer large values in *d* than we'd expected if they were normally distributed).

Verzani pg. 65

#9.1 The R function `rnorm` is supposed to draw a random sample from a normally distributed population with a given mean and standard deviation. When they ask "Did it get it right?", what they're asking is: does the population that `rnorm` draws from really have the specified mean? (One might also ask whether the population is normally distributed and whether is has the specified standard deviation, but the *z*-test won't tell you that.)

So, to get to it. First generate the random sample:

```
> x = rnorm(15,10,5)
> x
 [1] 12.208735  9.105011  6.687132 10.234824 13.855025 14.880171  8.317338
 [8]  7.862318 13.361448 -1.598729 14.578183 10.057801 14.736326 14.957778
[15]  7.602090
```

The *z*-test function defined on pg. 62 is part of UsingR, so we can use like so:

```
> library(UsingR)
> simple.z.test(x,5)
[1]  7.92606 12.98667
```

By the *z*-test, the 95% confidence interval for the mean of the population that this sample is drawn from is $(7.93, 12.99)$. Since the confidence interval includes 10 (which is supposed to be the true mean of that population), we have no reason to think there's anything wrong with the way `rnorm` is implemented.

#9.2 Suppose we draw a sample from a population, find the sample mean, and construct a 95% confidence interval around it (like we did above). We would expect that if we were to repeat that experiment over and over again, 95% of the time the confidence interval we computed would include the true population mean (in this case, 10)--that's what a "95% confidence interval" means.

Typically we can't repeat experiments in real life, but for this problem they want us to simulate a repeated experiment, to see if this interpretation of a 95% confidence interval holds. We can do this by drawing 100 samples, computing the 95% c.i.'s for each sample, and counting how many of those c.i.'s include the population mean 10:

```
> r = replicate(100,simple.z.test(rnorm(15,10,5),5))
> sum(r[1,] <= 10 & r[2,] >= 10)/100
[1] 0.94
```

It's not quite the predicted 95%, but it's close. If we increase the number of replications, we get even closer:

```
> r = replicate(10000,simple.z.test(rnorm(15,10,5),5))
> sum(r[1,] <= 10 & r[2,] >= 10)/10000
[1] 0.9485
```

Finally, for those of you who are still following all this, here's a brain bender: the example R commands given in this problem actually calculate something slightly different. What are those commands doing, and why is my version better?