Homework 2: The Big Short

This homework is due Sunday February 26, 2016 at 5:00 PM. When complete, submit your code in the R Markdown file and the knitted HTML via GitHub.

Background

This homework is motivated by circumstances surrounding the financial crisis of 2007-2008. We title the homework *The Big Short* because this is the title of a recent book about this topic that was recently made into a movie.

Part of what caused the financial crisis was that the risk of certain securities sold by financial institutions were underestimated. Specifically, the risk of mortgage-backed securities (MBS) and collateralized debt obligations (CDO), whose price dependents on homeowners making their monthly payments, were grossly underestimated. A combination of factors resulted in many more defaults than were expected. This resulted in a crash of the prices of these securities. As a consequence, banks lost so much money that they needed bailouts to avoid default.

Here we present a **very** simplified version of what happened with some of these securities. Hopefully it will help you understand how a wrong assumption about statistical behavior of events can lead to substantial differences between what the model predicts and what actually happens. Specifically, we will see how using an independence assumption can result in misleading conclusions. Before we start with the specific application we ask you about a simple casino game.

Problem 1

In the game of roullete you can bet on several things including black or red. On this bet, if you win, you double your earnings. How does the casino make money on this then? If you look at the possibilities you realize that the chance of red or black are both slightly less than 1/2. There are two green spots, so the of landing on black (or red) is actually 18/38, or 9/19.

Problem 1A

Let's make a quick sampling model for this simple version of roulette. You are going to bet a dollar each time you play and always bet on black. Make a box model for this process using the `sample` function. Write a function `get_outcome` that takes as an argument the number of times you play N and returns your earnings S_N.

```
get_outcome <- function(N){
  x <- sample( c(-1,1), N, replace=TRUE, prob=c(10/19,9/19))
  sum(x)
}
get_outcome(10)
```

```
## [1] 8
```

Problem 1B

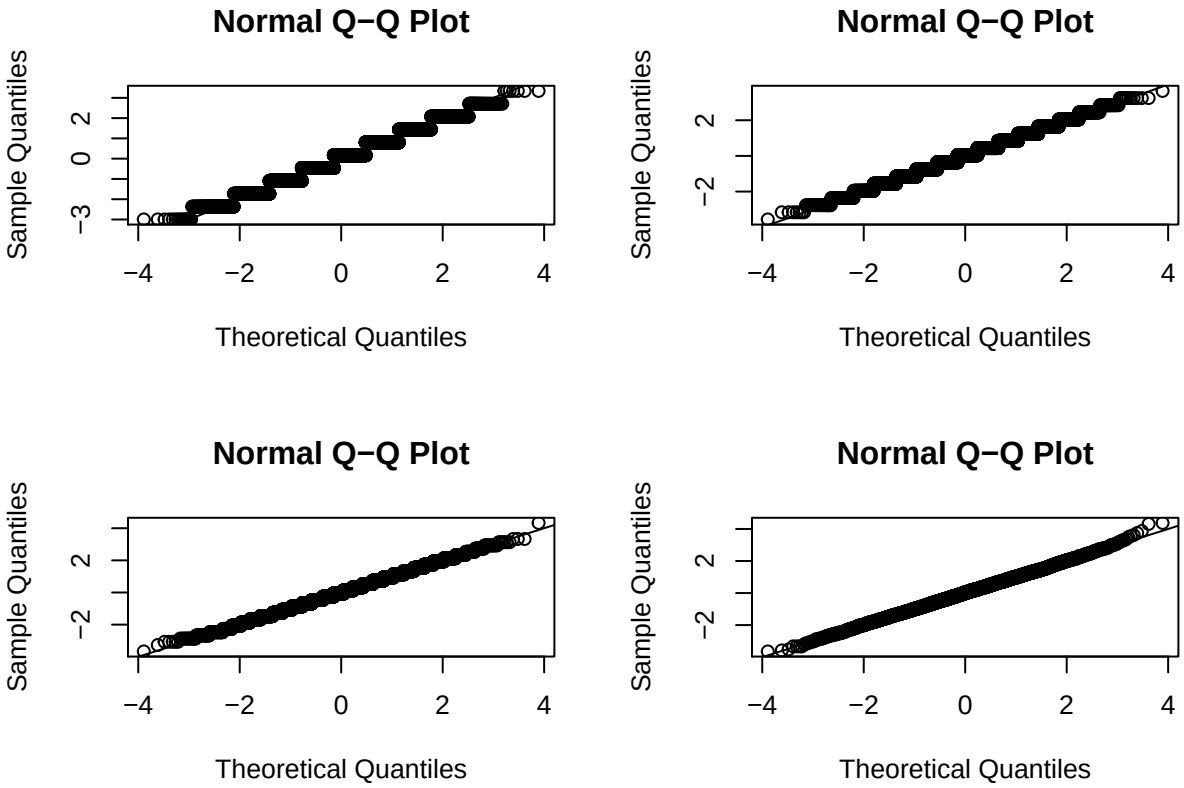
Use Monte Carlo simulation to study the distribution of total earnings S_N for $N = 10, 25, 100, 1000$. That is, study the distribution of earnings for different number of plays. What are the distributions of these two random variables? How does the expected values and standard errors change with N. Then do the same thing for the average winnings S_N/N. What result that you learned in class predicts this?

```
mysd <- function(x){
  sqrt(mean((x-mean(x))^2))
}
B <- 10^4
par(mfrow=c(2,2)) ##don't worry if you don't know what this is. You can remove it.
for(N in c(10, 25, 100, 1000)){
  winnings <- replicate(B,get_outcome(N) )
  z <- (winnings - mean(winnings))/mysd(winnings)
  qqnorm(z)
  abline(0,1)
  cat("Expected value:", mean(winnings),
      "Standard Error:", mysd(winnings),"\n")
}
```

```
## Expected value: -0.5434 Standard Error: 3.155363
```

```
## Expected value: -1.207 Standard Error: 4.994552
```

```
## Expected value: -5.3158 Standard Error: 10.00461
```



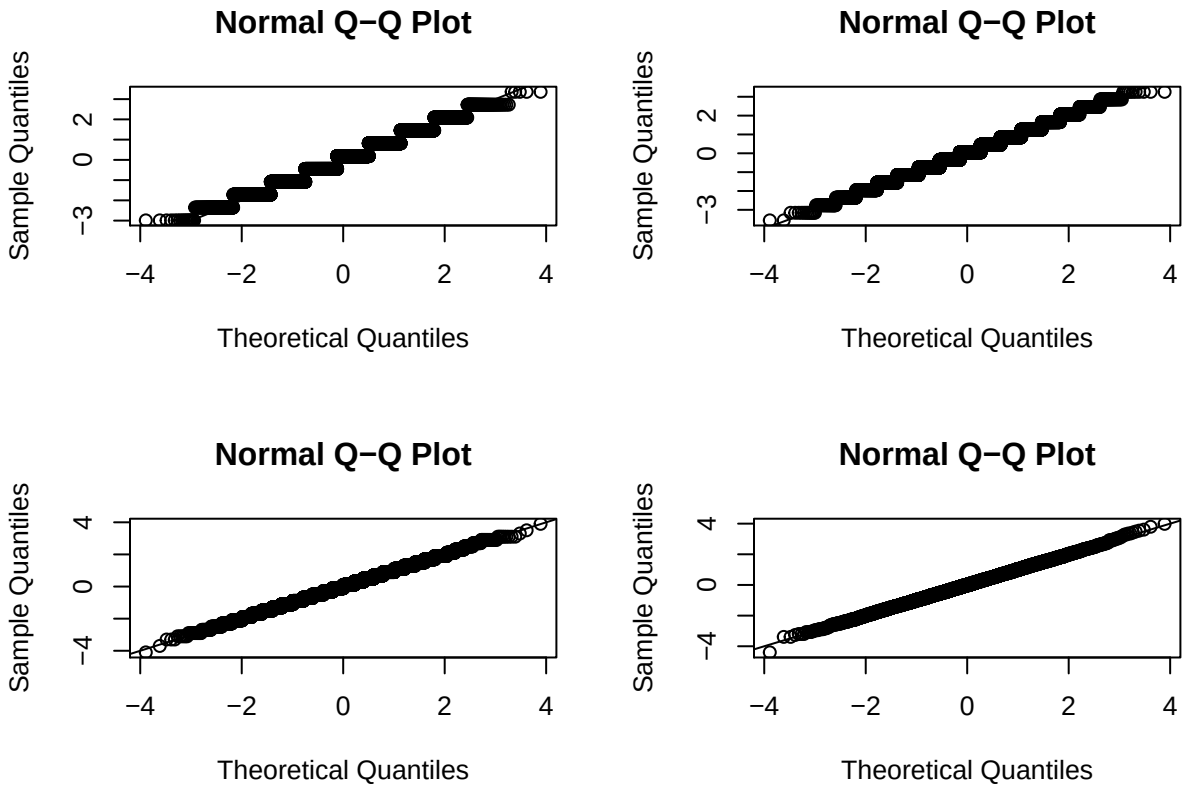
```
## Expected value: -52.6264 Standard Error: 31.66596
```

```
par(mfrow=c(2,2)) ##don't worry if you don't know what this is. You can remove it.
## Try out a few N
for(N in c(10, 25, 100, 1000)){
  ## copy code from above:
  average_winnings <- replicate(B,get_outcome(N) ) / N
  z <- (average_winnings - mean(average_winnings))/mysd(average_winnings)
  qqnorm(z)
  abline(0,1)
  cat("Expected value:", mean(average_winnings),
      "Standard Error:", mysd(average_winnings),"\n")
}
```

```
## Expected value: -0.05858 Standard Error: 0.3151323
```

```
## Expected value: -0.05032 Standard Error: 0.199461
```

```
## Expected value: -0.050258 Standard Error: 0.09996406
```




```
## Expected value: -0.052467 Standard Error: 0.03173256
```

Answer: These are predicted with the central limit theorem that tells us S_N is approximately normal with expected value μN and standard error $\sigma\sqrt{N}$. The average is approximated by the normal as well with expected value μ and standard error $\sigma/\sqrt{N}$. The SE average get's smaller and smaller with N.

Problem 1C

What is the expected value of our sampling model? What is the standard deviation of our sampling model?

Answer: Expectation is $\mu = -1 \times (1-p) + 1 \times p$ which is -1/19. The casino makes, on average, about 5 cents on each bet. In general you can envision this as a box model with an infinite number of tickets (since you can imagine playing this game an infinite number of times) with a fraction $p = 18/38$ tickets labeled 1 and $1 - p = 20/38$ tickets labeled -1.

Standard deviation is $\sigma = |1 - -1|\sqrt{(9/19)(10/19)}$ which is 0.998614.

Problem 1D

If you play 25 times. Use CLT to approximate the probability that the casino looses money. Then use a Monte Carlo simulation to corroborate.

You sum S is approximately normal with mean $\mu \times N$ and standard error $\sqrt{N}\sigma$ with $N = 25$

$$\Pr(S > 0) = \Pr\left(\frac{S - \mu N}{\sigma\sqrt{N}} > \frac{-\mu N}{\sigma\sqrt{N}}\right)$$

$$1 - \Phi^{-1}(\sqrt{N}\frac{-\mu}{\sigma})$$

```
1-pnorm( sqrt(25)*(1/19)/0.998614)
```

```
## [1] 0.3960737
```

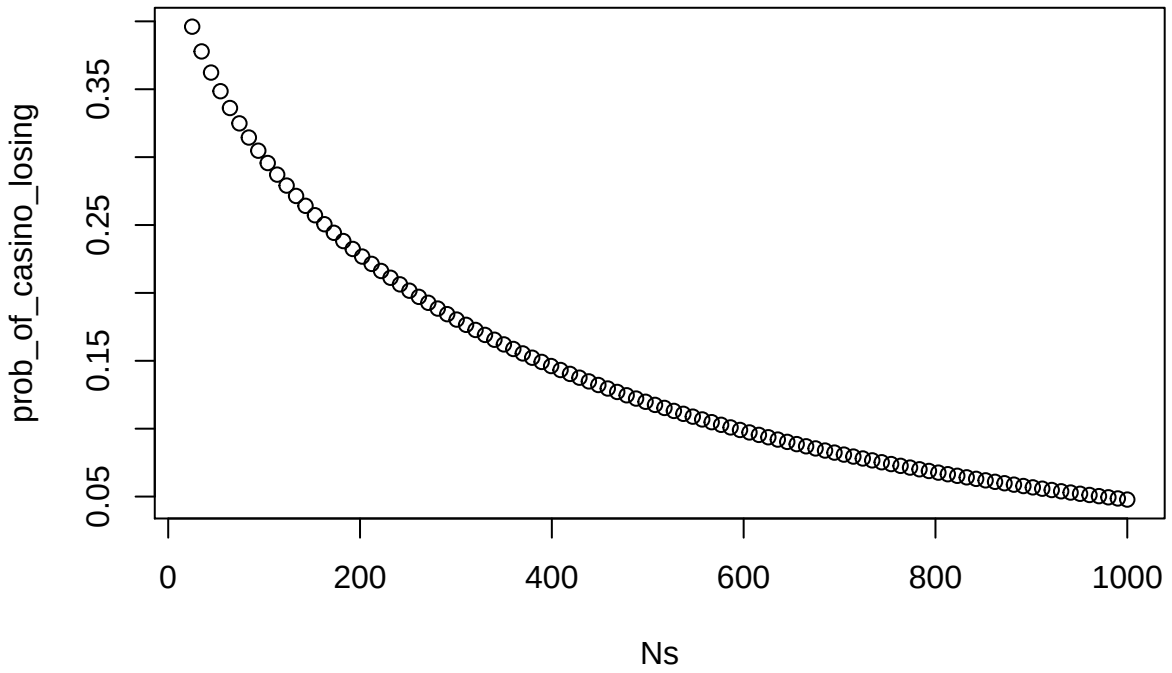
```
B <- 10^5
winnings <- replicate(B,get_outcome(25) )
mean(winnings>0)
```

```
## [1] 0.39245
```

Problem 1E

In general, what is the probability that the casino loses money as a function of N? Make a plot for values ranging from 25 to 1,000. Why does the casino give you free drinks if you keep playing?

```
Ns <-seq(25, 1000, len=100)
prob_of_casino_losing <- 1 - pnorm( sqrt(Ns)*(1/19/0.998614))
plot(Ns, prob_of_casino_losing)
```



Answer: Some of you did simulations to answer this question, but an quickly ran into computational issues with the large sample sizes. This is a perfect example of when an approximate answer is worth the error in approximation. This answer also probably gives you some idea of why casinos try to keep you at the table for long periods of time with free drinks!

Problem 2

You run a bank that has a history of identifying potential homeowners that can be trusted to make payments. In fact, historically, in a given year, only 2% of your customers default. You want to use stochastic models to get an idea of what interest rates you should charge to guarantee a profit this upcoming year.

Problem 2A

Your bank gives out 1,000 loans this year. Create a sampling model and use the function `sample` to simulate the number of foreclosure in a year with the information that 2% of customers default. Also suppose your bank loses $120,000 on each foreclosure. Run the simulation for one year and report your loss.

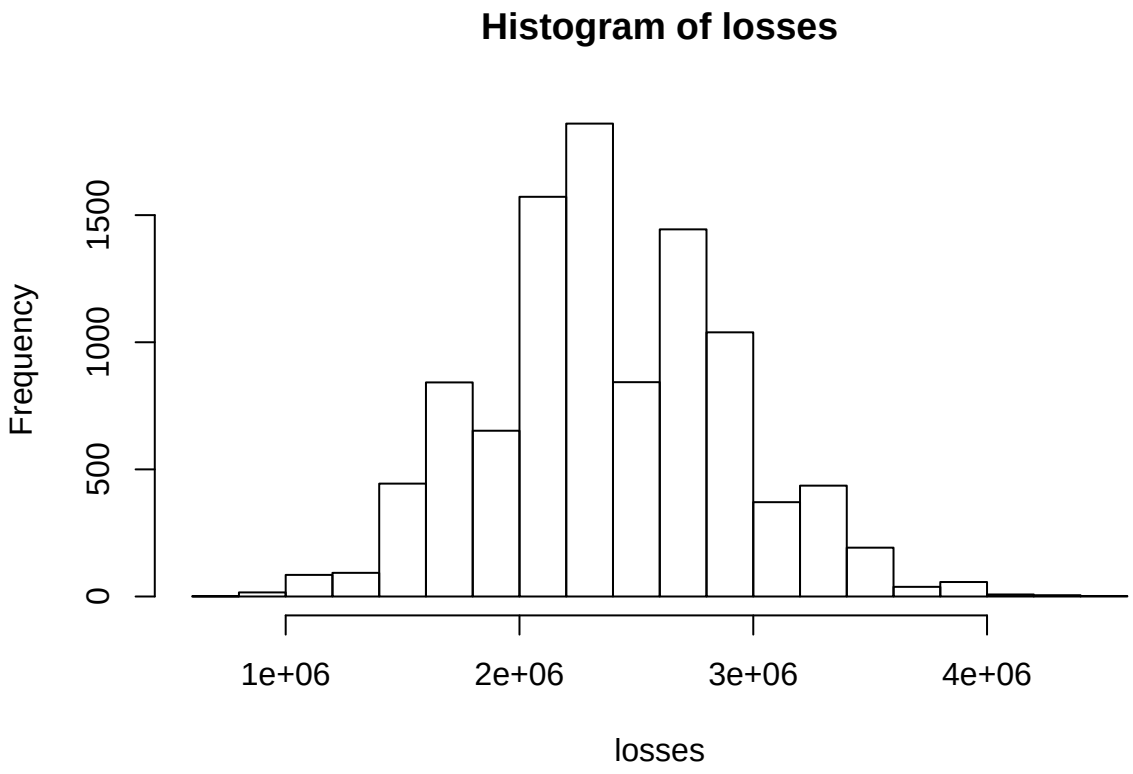
```
N <- 1000
loss_per_foreclosure <- 120000
p <- 0.02
defaults <- sample( c(0,1), N, prob=c(1-p, p), replace = TRUE)
sum(defaults * loss_per_foreclosure)
```

```
## [1] 2640000
```

Problem 2B

Note that the loss you will incur is a random variable. Use Monte Carlo simulation to estimate the distribution of this random variable. Use summaries and visualization to describe your potential losses to your board of trustees.

```r
N <- 1000
loss_per_foreclosure <- 120000
p <- 0.02
B <- 10^4
## Use replicate to generate B random variables
losses <- replicate(B, {
  ## copy code from 2A
  defaults <- sample( c(0,1), N, prob=c(1-p, p), replace = TRUE)
  sum(defaults * loss_per_foreclosure)
})
## Data looks normally distributed:
hist(losses)
```

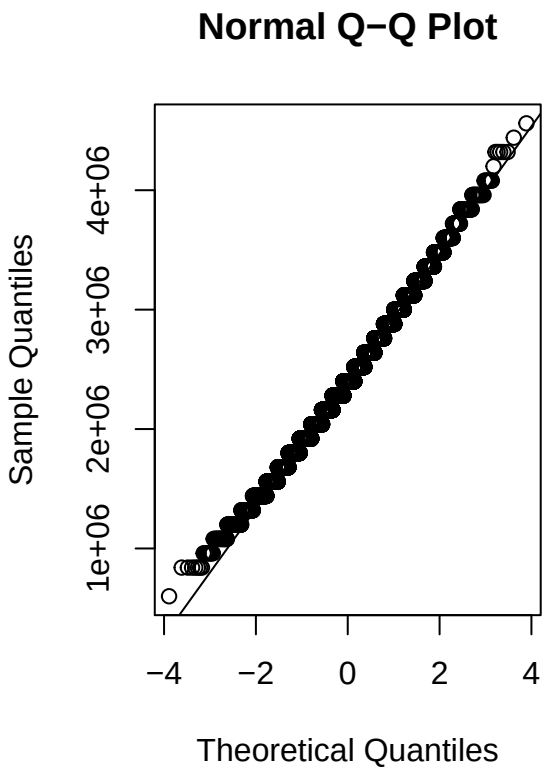



```r
par(mfrow=c(1,2))
qqnorm(losses)
qqline(losses)
mean(losses)
```

```
## [1] 2404752
```

```r
sd(losses)
```

```
## [1] 525314.3
```



Problem 2C

The 1,000 loans you gave out were for $180,000. The way your bank can give out loans and not lose money is by charging an interest rate. If you charge an interest rate of, say, 2% you would earn $3,600 for each loan that doesn't foreclose. At what percentage should you set the interest rate so that your expected profit totals $100,000. Hint: Create a sampling model with expected value 100 so that when multiplied by the 1,000 loans you get an expectation of $100,000. Corroborate your answer with a Monte Carlo simulation.

Solution: The box model must have an expected value of 100 so that the sum of 1,000 outcomes has expected value $100,000. We lose $120,000 with a probability of $p = 0.02$. If we make x on each loan that doesn't foreclose, our expected value is $0.98x - 0.02 \times 120000$. For this to add up to 100 we need This $x = (100 + 0.02 \times 120000)/0.98$. Our interest rate r must be $200000r = (100 + 0.02 \times 120000)/0.98$ or $r = (100 + 0.02 \times 120000)/0.98/200000$.

Check with Monte Carlo simulation

```
N <- 1000
loan <- 180000
p <- 0.02
loss_per_foreclosure <- 120000
interest_rate <- (100+p*loss_per_foreclosure)/(1-p)/loan
cat("interest rate is: ",interest_rate)
```

```
## interest rate is:  0.01417234
```

```
B <- 10000
profit <- replicate(B, {
  ## We add either a loss of -loss_per_foreclosure or gain of interest_rate*loan
```

```
  ## with probabilities p and 1-p respectively for each of N loans
  x <- sample( c(-loss_per_foreclosure, interest_rate*loan ), N, replace=TRUE, prob=c(p, 1-p))
  ## add gains and losses for total profit
  sum(x)
})
mean(profit/N) ##should be about 100
```

```
## [1] 106.7158
```

Note: we'll try to ask you to verify things with a simulation but you'll start noticing that it's generally a good idea to do this to catch mistakes since simulations are quick and easy to write up.

Problem 2D

In problem 2C, you were able to set a very low interest rate. Your customers will be very happy and you are expected to earn $100,000 in profits. However, that is just an expectation. Our profit is a random variable. If instead of a profit your bank loses money, your bank defaults. Under the conditions of Problem 2C, what is the probability that your profit is less than 0?

```
## Run scripts above then:
mean(profit < 0)
```

```
## [1] 0.4365
```

Problem 2E

Note that the probability of losing money is quite high. To what value would you have to raise interest rates in order to make the probability of losing money, and your bank and your job, as low as 0.001? What is the expected profit with this interest rate? Corroborate your answer with a Monte Carlo simulation.

Hint: Use the following short cut. If p fraction of a box are as and $(1-p)$ are bs then , then the SD of the list if $\mid a-b \mid \sqrt{p(1-p)}$

Solution: Let $x = 180000r$ be the earnings we make per loan at an interest rate of r and $l = 120000$ the loss resulting from a foreclosure. The *box* in our box model has an expected value of $\mu = x(1-p) - lp+$ with p the probability of default. The standard error is $\sigma = (x+l)\sqrt{p(1-p)}$. We now that the sum S of the $N =$ 1,000 outcomes is approximately normal with expected value $N\mu$ and standard error $\sqrt{N}\sigma$.

$$\Pr(S < 0) = 0.001$$

$$\Pr\left(\frac{S - N\mu}{\sigma\sqrt{N}} < \frac{-N\mu}{\sigma\sqrt{N}}\right) = 0.001$$

This random variable $(S - N\mu)/(\sigma\sqrt{N})$ is approximately normal with expected value 0 and standard error 1. So this means that to get a probability of 0.001 we need:

$$\sqrt{N}\frac{\mu}{\sigma} = z \text{ with } z = -\Phi^{-1}(0.001)$$

$$\sqrt{N}\frac{x(1-p) - lp}{(x+l)\sqrt{p(1-p)}} = z$$

$$\sqrt{N}x(1-p) - \sqrt{N}lp = zx\sqrt{p(1-p)} + lz\sqrt{p(1-p)}$$

$$x\left\{(1-p)\sqrt{N} - z\sqrt{p(1-p)}\right\} = l\left(z\sqrt{p(1-p)} + \sqrt{N}p\right)$$

$$x = \frac{l\left(z\sqrt{p(1-p)} + \sqrt{N}p\right)}{(1-p)\sqrt{N} - z\sqrt{p(1-p)}}$$

The interest rate is $x/180000$.

Confirm with a Monte Carlo simulation:

```
N <- 1000
loan <- 180000
p <- 0.02
z <- -qnorm(0.001)
loss_per_foreclosure <- 120000
interest_rate <- loss_per_foreclosure*(z*sqrt(p*(1-p))+sqrt(N)*p)/
  ((1-p)*sqrt(N) - z*sqrt(p*(1-p)))/loan
cat("Interest rate is: ", interest_rate,"\n")
```

```
## Interest rate is:  0.02323666
```

```
expectation <- -loss_per_foreclosure*p + interest_rate*loan*(1-p)
cat("Mean profit:" , expectation*N )
```

```
## Mean profit: 1698947
```

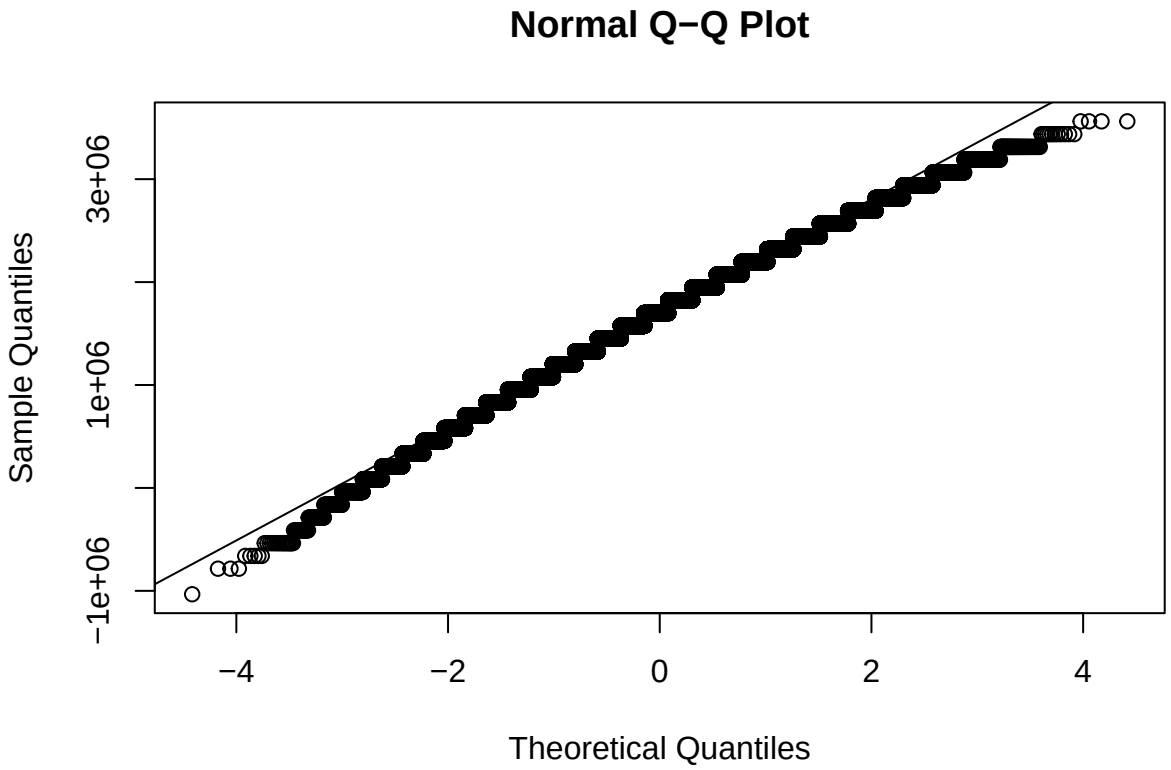
```
B <- 10^5
profit <- replicate(B, {
  x <- sample( c(-loss_per_foreclosure, interest_rate*loan ), N, replace=TRUE, prob=c(p, 1-p))
  sum(x)
})
mean(profit<0)
```

```
## [1] 0.00251
```

Problem 2F

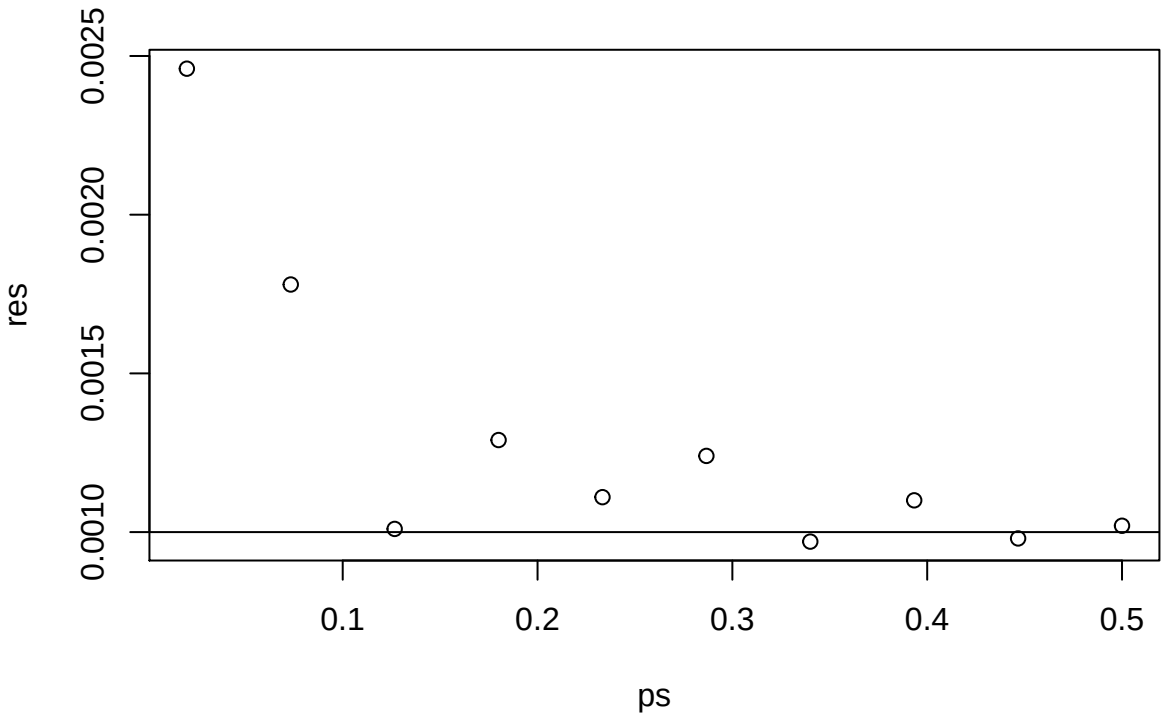
Note that the Monte Carlo simulation gave a slightly higher probability than 0.001. What is a possible reason for this? Hint: See if the disparity is smaller for larger values of p. Also check for probabilities larger than 0.001. Recall we made an assumptions when we calculated the interest rate.

```
## Run script from 2E
##Note that the normal approximation is not that great at the tails
qqnorm(profit)
qqline(profit)
```



Also note that as the approximation is much better for value of p closer 0.5

```
N <- 1000
loan <- 180000
p <- 0.02
loss_per_foreclosure <- 120000
z <- -qnorm(0.001)
B <- 10^5
ps <- seq(0.02,0.5,len=10)
res <- sapply(ps,function(p){
  ##same code as above but changing p and therefore the interest_rate:
  interest_rate <- loss_per_foreclosure*(z*sqrt(p*(1-p))+sqrt(N)*p)/
  ((1-p)*sqrt(N) - z*sqrt(p*(1-p)))/loan
  profit <- replicate(B, {
    x <- sample( c(-loss_per_foreclosure, interest_rate*loan ), N, replace=TRUE, prob=c(p, 1-p))
    sum(x)
    })
  mean(profit<0) ##should be about 0.001
})
plot(ps,res)
abline(h=0.001)
```



When p is close to 0 or 1, the central limit theorem needs larger N for the approximation to work.

Problem 3

We were able to set interest rate of about 2% that guaranteed a very low probability of having a loss. Furthermore, the expected average was over $1 million. Now other financial companies noticed the success of our business. They also noted that if we increase the number loans we give, our profits increase. However, the pool of reliable borrowers was limited. So these other companies decided to give loans to less reliable borrowers but at a higher rate.

Problem 3A

The pool of borrowers they found had a much higher default rate, estimated to be $p = 0.05$. What interest rate would give these companies the same expected profit as your bank (Answer to 2E)?

```
##From 2E
N <- 1000
loan <- 180000
loss_per_foreclosure <- 120000
p <- 0.02
z <- -qnorm(0.001)
interest_rate <- loss_per_foreclosure*(z*sqrt(p*(1-p))+sqrt(N)*p)/
  ((1-p)*sqrt(N) - z*sqrt(p*(1-p)))/loan

## expectation per loan is therefore:
expectation <- -loss_per_foreclosure*p + interest_rate*loan*(1-p)

p2 <- 0.05
interest_rate2 <- (expectation +
```

```
                      loss_per_foreclosure*p2)/(1-p2)/loan
cat("New interest rate is:", interest_rate2)
```

```
## New interest rate is: 0.04502308
```

Problem 3B

At the interest rate calculated in 3A what is the probability of negative profits? Use both the normal approximation and then confirm with a Monte Carlo simulation.

Answer: Normal approximation tells us the average profit S/N has expected value $\mu =$`expectation` and standard deviation $\sigma/\sqrt{N}$ with $\sigma =$ `(interest_rate2*loan + loss_per_foreclosure)*sqrt(p2*(1-p2))`. So we compute

$$\Pr(S/N > 0) = \Pr\left(\sqrt{N}(S/N - \mu)/\sigma > -\sqrt{N}\mu/\sigma\right) = \Pr\left(Z > -\sqrt{N}\mu/\sigma\right)$$

```
## From 3A we have the expectation is
N <- 1000
loan <- 180000
loss_per_foreclosure <- 120000
p <- 0.02
z <- -qnorm(0.001)
interest_rate <- loss_per_foreclosure*(z*sqrt(p*(1-p))+sqrt(N)*p)/
  ((1-p)*sqrt(N) - z*sqrt(p*(1-p)))/loan

expectation <- -loss_per_foreclosure*p + interest_rate*loan*(1-p)

p2 <- 0.05
interest_rate2 <- (expectation +
                     loss_per_foreclosure*p2)/(1-p2)/loan

## to compute normal approximation:
mu  = expectation
sigma = (interest_rate2*loan + loss_per_foreclosure)*sqrt(p2*(1-p2))
pnorm(-sqrt(N)*mu/sigma)
```

```
## [1] 0.0271593
```

```
## Now confirm with Monte Carlo
B <- 10^4
profit <- replicate(B, {
  x <- sample( c(-loss_per_foreclosure, interest_rate2*loan ), N, replace=TRUE, prob=c(p2, 1-p2))
  sum(x)
})
mean(profit<0)
```

```
## [1] 0.0278
```

Problem 3C

Note that the probability is much higher now. This is because the standard deviation grew. The companies giving out the loans did not want to raise interest rates much more since it would drive away clients. Instead they used a statistical approach. They increased N. How large does N need to be for this probability to be 0.001? Use the central limit approximation and then confirm with a Monte Carlo simulation.

Answer: From before

$$\sqrt{N}\frac{\mu}{\sigma} = z \text{ with } z = -\Phi^{-1}(0.001)$$

which implies that with the new default probability p_2 we have

$$\sqrt{N} = z\frac{(x+l)\sqrt{p_2(1-p_2)}}{x(1-p_2) - lp_2}$$

```
## From 3B we have the new interest rate is computed like this
N <- 1000
loan <- 180000
loss_per_foreclosure <- 120000
p <- 0.02
z <- -qnorm(0.001)
interest_rate <- loss_per_foreclosure*(z*sqrt(p*(1-p))+sqrt(N)*p)/
  ((1-p)*sqrt(N) - z*sqrt(p*(1-p)))/loan

expectation <- -loss_per_foreclosure*p + interest_rate*loan*(1-p)

p2 <- 0.05
interest_rate2 <- (expectation +
                     loss_per_foreclosure*p2)/(1-p2)/loan


## Now let's compute the new N
z <- -qnorm(0.001)
N2 <- (z*(interest_rate2*loan + loss_per_foreclosure)*sqrt(p2*(1-p2)) /
  (interest_rate2*loan*(1-p2) - loss_per_foreclosure*p2 ))^2
cat("N is :",N2)
```

```
## N is : 2578.947
```

```
### And confirm that the probability goes down to close to 0.001
B <- 10^4
profit <- replicate(B, {
  x <- sample( c(-loss_per_foreclosure, interest_rate2*loan ), N2, replace=TRUE, prob=c(p2, 1-p2))
  sum(x)
})
mean(profit<0)
```

```
## [1] 0.0011
```

So by increasing the number of loans we were able to reduce our risk! Now, for this to work, all the assumptions in our model need to be approximately correct, including the assumption that the probability of default was **independent**. This turned out to be false and the main reason for the under estimate of risk.

Problem 3D

Define the following matrix of outcomes for two borrowers using our previous box model:

```
loan <- 180000
loss_per_foreclosure <- 120000
p2 <- 0.05
interest_rate2 <- 0.05
B <- 10^5
outcomes1 <- replicate(B,{
  sample( c(-loss_per_foreclosure, interest_rate2*loan ), 2, replace=TRUE, prob=c(p2, 1-p2))
})
```

We can confirm independence by computing the probability of default for the second conditioned on the first defaulting:

```
sum( outcomes1[1,] < 0 & outcomes1[2,]<0)/sum(outcomes1[1,]<0)
```

```
## [1] 0.05108453
```

This quantity is about the same as the probability of default 0.05.

Now we create a new model. Before generating each set of defaults, we assume that an random event occurred that makes all default probabilities go up or go down by 4 points. We could see how this would happen if, for example, demand for houses decreases and all house prices drop.

```
B <- 10^5
outcomes2 <- replicate(B,{
  add <- sample( c(-0.04,0.04), 1)
  sample( c(-loss_per_foreclosure, interest_rate2*loan ), 2, replace=TRUE, prob=c(p2+add, 1-(p2+add)))
})
```

Note that the outcomes are no longer independent as demonstrated by this result not being equal to 0.05

```
sum( outcomes2[1,] < 0 & outcomes2[2,]<0)/sum(outcomes2[1,]<0)
```

```
## [1] 0.083183
```

Generate a simulation with correlated outcomes such as those above. This time use the interest rate calculated in 3A. What is the expected earnings under this model compared to the previous?

```
### Compute the interest rate from 3A
N <- 1000
loan <- 180000
loss_per_foreclosure <- 120000
p <- 0.02
z <- -qnorm(0.001)
interest_rate <- loss_per_foreclosure*(z*sqrt(p*(1-p))+sqrt(N)*p)/
  ((1-p)*sqrt(N) - z*sqrt(p*(1-p)))/loan

## expectation per loan is therefore:
expectation <- -loss_per_foreclosure*p + interest_rate*loan*(1-p)
```

```
p2 <- 0.05
interest_rate2 <- (expectation +
                     loss_per_foreclosure*p2)/(1-p2)/loan
## And the N we need to get the same low probability
z <- -qnorm(0.001)
N2 <- (z*(interest_rate2*loan + loss_per_foreclosure)*sqrt(p2*(1-p2)) /
  (interest_rate2*loan*(1-p2) - loss_per_foreclosure*p2 ))^2


## Now run the simulation using the code given above for correlated data
B <- 10^5
profit2 <- replicate(B,{
  add <- sample( c(-0.04,0.04), 1)
  x <- sample( c(-loss_per_foreclosure, interest_rate2*loan ), N2, replace=TRUE, prob=c(p2+add, 1-(p2+a
  sum(x)
})
```

Answer:

Note that expected profits are about the same:

```
## Without the added probabitiliy to induce correlation we get:
B <- 10^5
profit <- replicate(B,{
  add <- 0
  x <- sample( c(-loss_per_foreclosure, interest_rate2*loan ), N2, replace=TRUE, prob=c(p2+add, 1-(p2+a
  sum(x)
})

mean(profit)
```

```
## [1] 4384716
```

```
mean(profit2)
```

```
## [1] 4370193
```

What is the probability of losing $1 million compared to the previous?

```
mean(profit < 0)
```

```
## [1] 0.00115
```

```
mean(profit2 < 0)
```

```
## [1] 0.50048
```

What is the probability of losing $10 million compared to the previous?

```r
mean(profit < -10^6)
```

```
## [1] 9e-05
```

```r
mean(profit2 < -10^6)
```

```
## [1] 0.50048
```

Problem 4

Read this wikipedia page about the financial crisis. Write a paragraph describing how what you learned in this homework help explain the conditions that led to the crisis.

Example answer: If the creators of the securities assumed independence when computing the probability but the reality was that events were not independent, then they grossly underestimated the chances of losing money. This resulted in them selling securities at to low a price since the risk was much higher than what they calculated. This led to billions of dollars of losses once defaults started happening at a much higher rate than expected.

Of course this was not the only reason for the underestimation of the risk, but we hope this gave you some idea of the types of analysis that could help explain things.