Problem -1

```r
Num = 1000
Size = 10000 # dollar amount in thousands
PD = 0.0019
LGD = 0.45
rho = 0.15
L = Num*Size
N = 10000
#Q.1.1
X <- 0.999

Zi =  (qnorm(PD) + sqrt(rho)*qnorm(X))/(sqrt(1 - rho))
WCDR_0.999 <- pnorm(Zi)
creditVar <- L * LGD *(WCDR_0.999 - PD)

print("VaR 99.95 from Closed form Copula model is: ")
print(creditVar)
```

```
[1] "VaR 99.95 from Closed form Copula model is: "
[1] 139047
```

```r
#Q.1.2

N <- 10000 #Number of iterations

creditVar_sims <- function(Num, Size){
  iter_loss <- array(0, dim=c(N)) # Distribution of losses per iteration
  for(iter in 1:N){

    F <- matrix(rnorm(1), 1, Num) # One F Factor value per iteration (same number is needed)
    Z <- matrix(rnorm(Num, mean=0, sd=1), 1, Num) # idiosyncratic errors for all loans
    U <- sqrt(rho) * F + sqrt(1-rho) * Z # Generate the U_i for all loans
    Default <- (U < qnorm(PD)) # Every loan, every iteration, did it default (binary operator)
    loan_loss <- Size * Default * LGD # Total loss on each loan for this iteration
    iter_loss[iter] <- sum(loan_loss)

  }
  hist(iter_loss)
  EL <- mean(iter_loss)
  VaR <- quantile(iter_loss, X)
  return(c(EL, VaR))
}

stats = creditVar_sims(Num, Size)
print( "Expected Loss is ")
print(stats[1])
print( "VaR 99.9% is ")
print(stats[2])
```

```
[1] "Expected Loss is "

8624.25
```

```
[1] "VaR 99.9% is "
   99.9%
144004.5
```

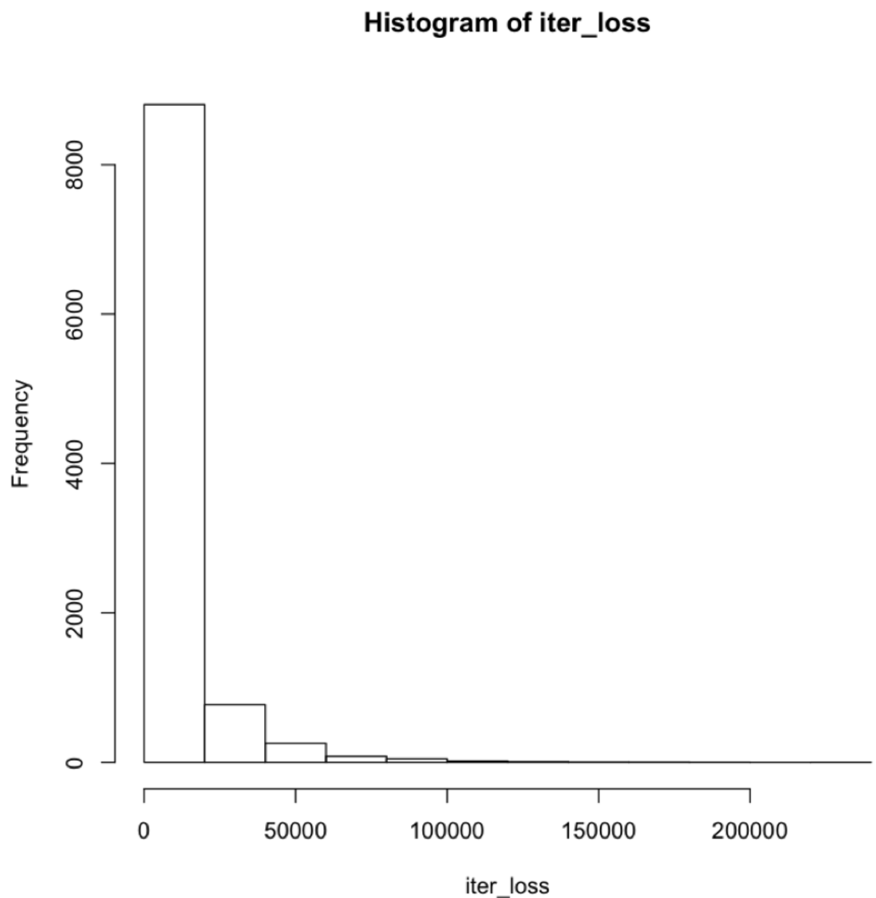

Figure 1: png

```r
#Q.1.3

Num2 = 100
Size2 = 100000

stats_case2 <- creditVar_sims(Num2, Size2)
print( "Expected Loss is ")
print(stats_case2[1])
```

```
print( "VaR 99.9% is ")
print(stats_case2[2])

# VaR increased as Number of loans decreased, it is reasonable as variance increases with decrease in s
```

```
[1] "Expected Loss is "

8437.5
[1] "VaR 99.9% is "
 99.9%
180000
```

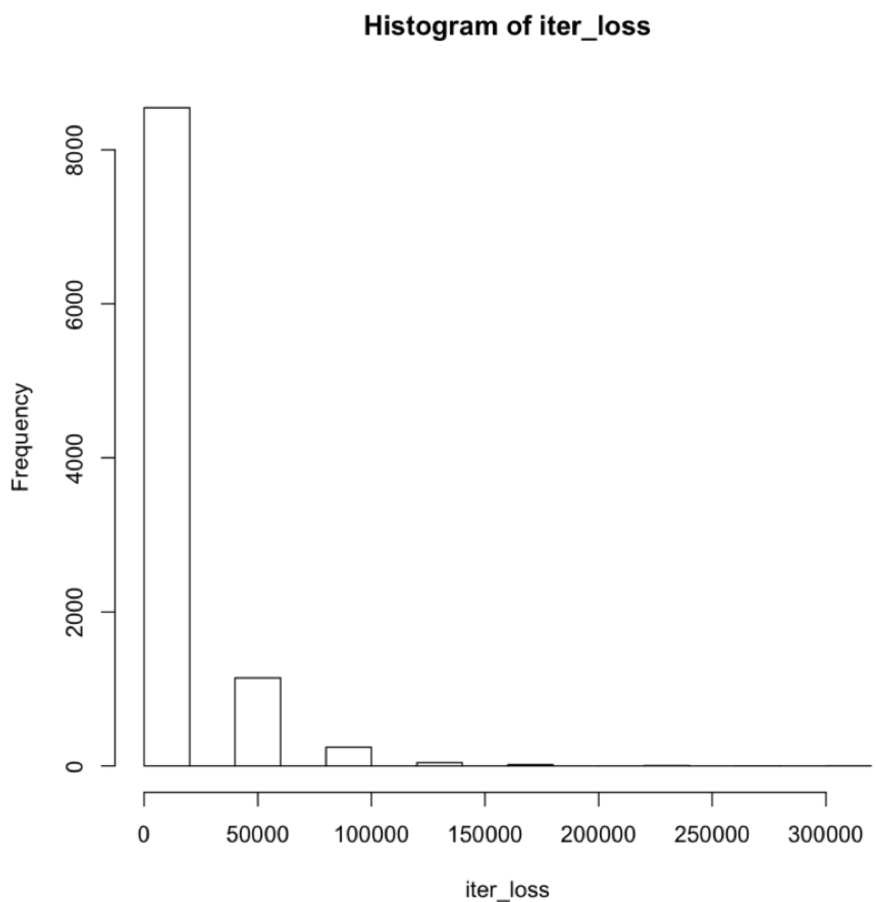

Figure 2: png

```r
#Q.1.4

Num3 = 10
Size3 = 1000000
stats_case3 <- creditVar_sims(Num3, Size3)
print( "Expected Loss is ")
print(stats_case3[1])
print( "VaR 99.9% is ")
print(stats_case3[2])
```

```
[1] "Expected Loss is "

7785
[1] "VaR 99.9% is "
 99.9%
450000
```

Closed form solution and simulation of part 1 & 2 should be same theoratically but due to usage of random number generation there values part away slightly. Whereas parts 3 and 4, due to decrease in sample size, variance increases which results in higher VaR values which is evident in the outputs.

Problem -2

```r
#Q.2.1

#Q.2.1

Num = 1000
problem2.1 <- creditVar_sims(1000, 10000)
print( "Expected Loss is ")
print(problem2.1[1])
print( "VaR 99.9% is ")
print(problem2.1[2])
```

```
[1] "Expected Loss is "

8302.5
[1] "VaR 99.9% is "
   99.9%
139504.5
```

Problem -2

```r
N <- 1000
spread <- c(0.70, 0.88, 1.19, 2.10, 3.39, 4.56, 8.17, 0)
transition <- c(0.05, 0.19, 4.79, 89.41, 4.35, 0.82, 0.2, 0.19)
```

```r
CreditVaR <- function(Num, Size, LGD, PD, rho, alpha, spread, transiton){
  dat <- cbind( spread, transition, rev(cumsum(rev(transition)))/100 )
  colnames(dat)[3] <- "cum"
  iter_loss <- array(0, dim=c(N))
  for(iter in 1:100){
    F <- matrix(rnorm(1), 1, Num)
    Z <- matrix(rnorm(Num, mean=0, sd=1), 1, Num)
    U <- sqrt(rho) * F + sqrt(1-rho) * Z
    Default <- (U < qnorm(PD))
```

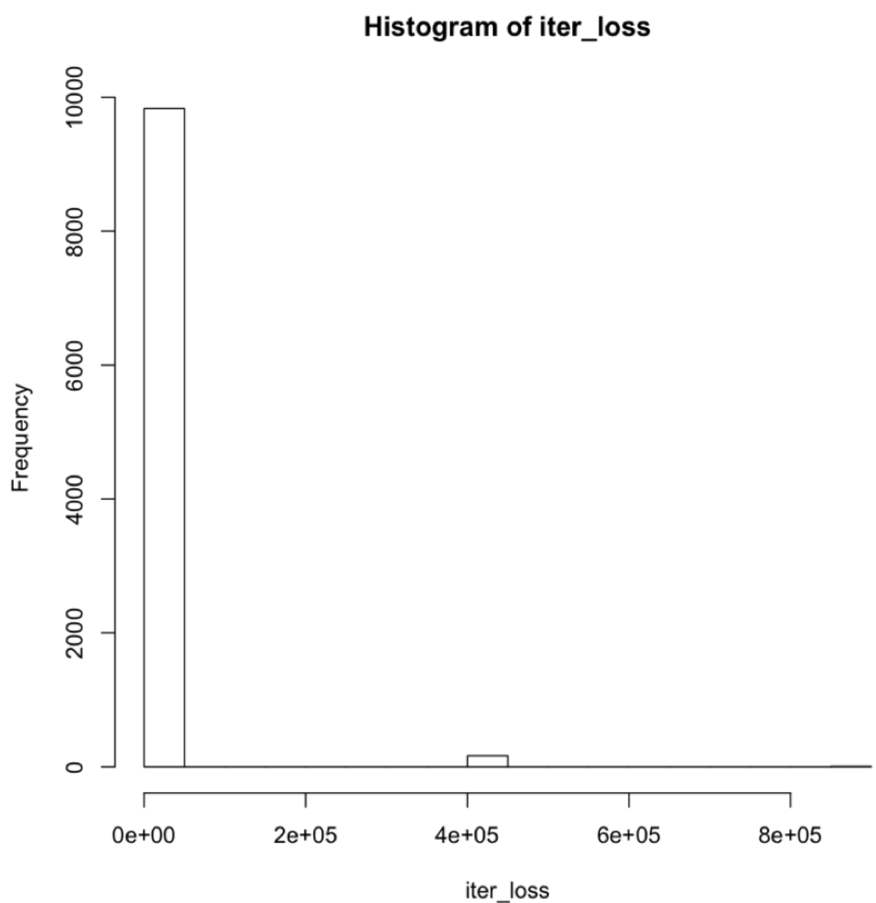
Histogram of iter_loss
Frequency
0
2000
4000
6000
8000
10000
0e+00
2e+05
4e+05
6e+05
8e+05
iter_loss

Figure 3: png

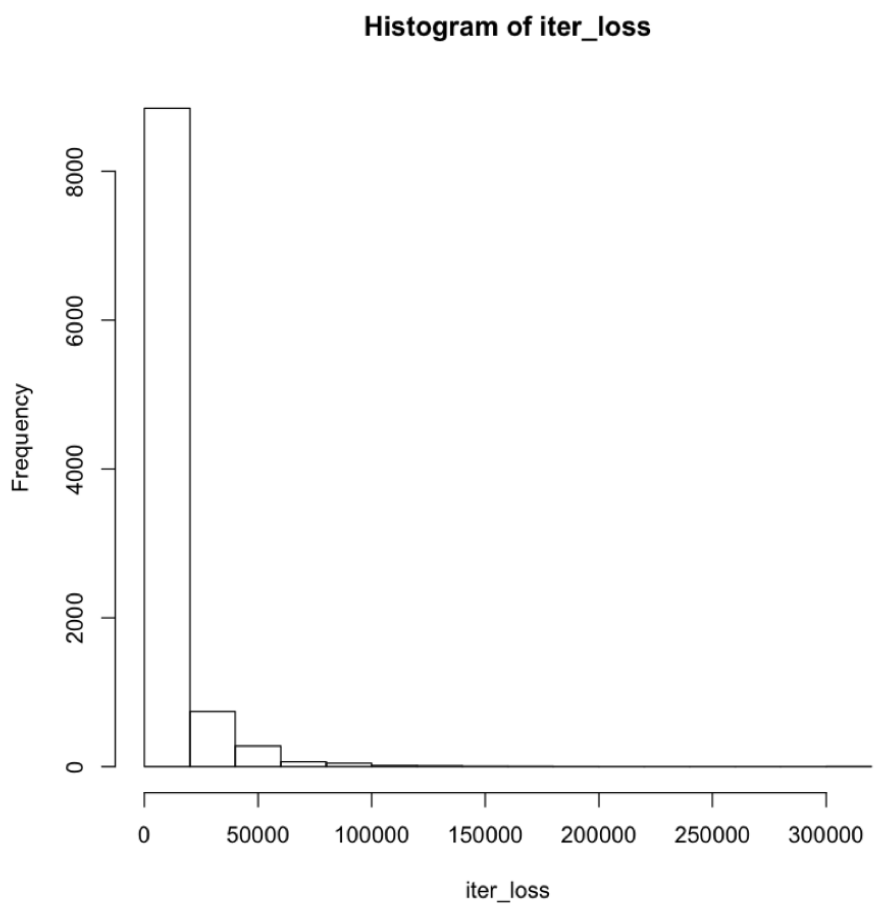
Histogram of iter_loss
Frequency
0
2000
4000
6000
8000
0
50000
100000
150000
200000
250000
300000
iter_loss

Figure 4: png

```
    prob <- pnorm(U)
    prob <- as.data.frame(prob)
    next_rate <- indexer(prob, dat)
    iter_loss[iter] <- func(next_rate, spread, LGD)
    }
  hist(iter_loss)
  EL <- mean(iter_loss)
  VaR <- quantile(iter_loss, X)
  return(c(EL, VaR))
  }
```

```
CreditVaR(Num, Size, LGD, PD, rho, alpha, spread, transition)
```

```
<dt>1</dt>
    <dd>1115.47372565731</dd>
<dt>99.9%</dt>
    <dd>59850.1659545031</dd>
```

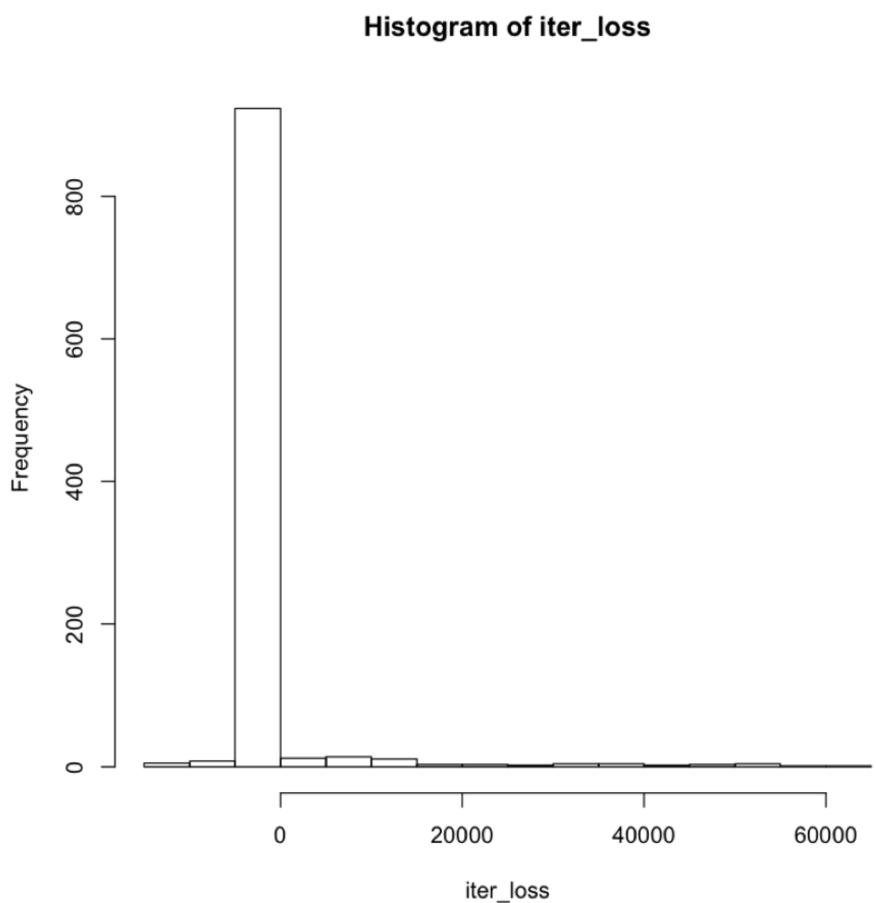
Histogram of iter_loss
Frequency
0
200
400
600
800
0
20000
40000
60000
iter_loss

Figure 5: png