EDA 6장 과제

주의사항을 숙지하였고 모든 책임을 지겠습니다.

2019122041 송유진

---

1. 다음은 대학생들의 언어 능력을 측정한 값들이다. 정규분포를 하는가? 직선의 기울기와 절편의 값은? 자료에서 직접 구한 평균과 분산 추정치와 확률도에서 구한 평균과 분산 추정치가 일치하는가?

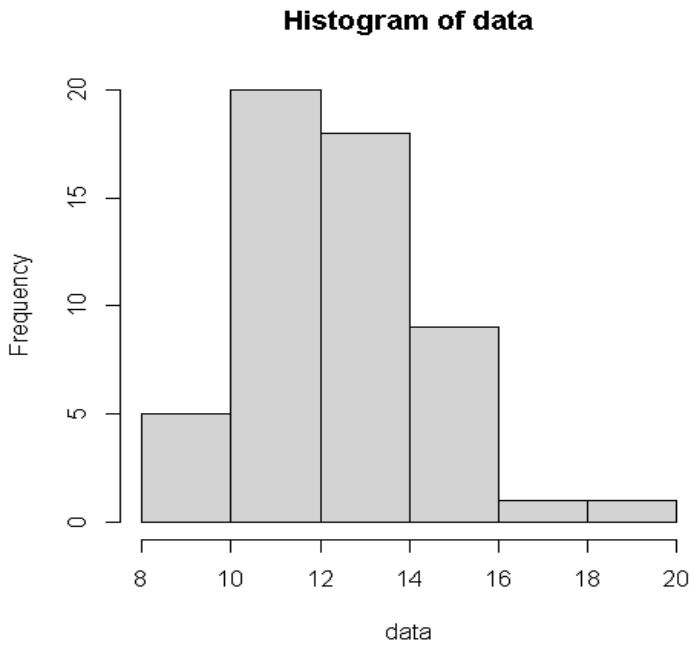
14	11	13	13	13	15	11	16	10
13	14	11	13	12	10	14	10	14
16	14	14	11	11	11	13	12	13
11	11	15	14	16	12	17	9	16
11	19	14	12	12	10	11	12	13
13	14	11	11	15	12	16	15	11

```
> data <- c(14, 11, 13, 13, 13, 15, 11, 16, 10,
+ 13, 14, 11, 13, 12, 10, 14, 10, 14,
+ 16 ,14, 14, 11, 11, 11, 13, 12, 13,
+ 11, 11, 15, 14, 16, 12, 17, 9, 16,
+ 11, 19, 14, 12, 12, 10, 11, 12, 13,
+ 13, 14, 11, 11, 15, 12, 16, 15, 11)
>
> data
 [1] 14 11 13 13 13 15 11 16 10 13 14 11 13 12 10
[16] 14 10 14 16 14 14 11 11 11 13 12 13 11 11 15
[31] 14 16 12 17  9 16 11 19 14 12 12 10 11 12 13
[46] 13 14 11 11 15 12 16 15 11
>
> stem(data)

  The decimal point is at the |

   8 | 0
  10 | 00000000000000000
  12 | 0000000000000000
  14 | 000000000000
  16 | 000000
  18 | 0

> fivenum(data)
[1]  9 11 13 14 19
> source("http://mgimond.github.io/ES218/es218.R")
> lsum(data)
  letter depth lower   mid upper spread
1      M  27.5    13 13.00  13.0    0.0
2      H  14.0    11 12.50  14.0    3.0
3      E   7.5    11 13.25  15.5    4.5
4      D   4.0    10 13.00  16.0    6.0
5      C   2.5    10 13.25  16.5    6.5
```

```
> length(data)
[1] 54
```

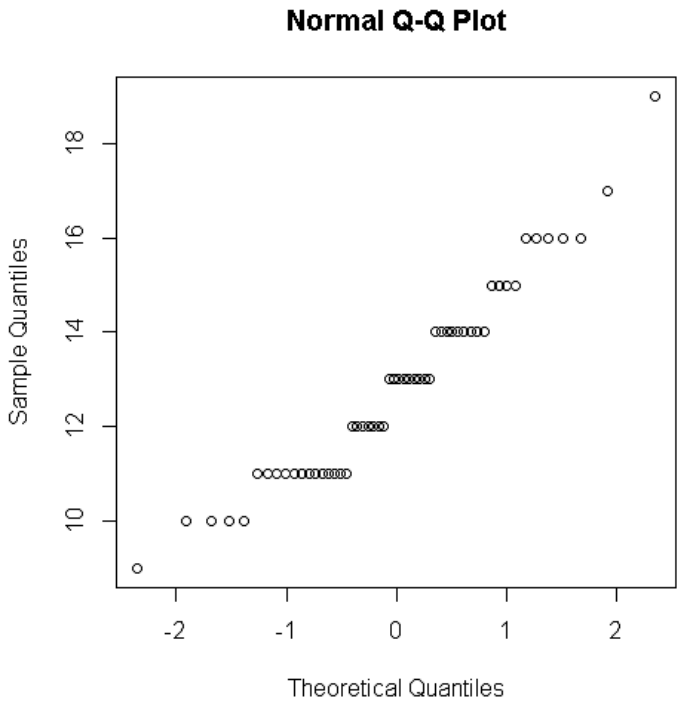
상단의 코드를 통해 데이터를 생성해주었고 줄기잎그림과 문자전시 그리고 fivenum()을 살펴보았다.

> hist(data)



다음으로, histogram을 그려보았다. 오른쪽으로 skewed되어있는 형태를 확인할 수 있었다.

qqnorm(data)



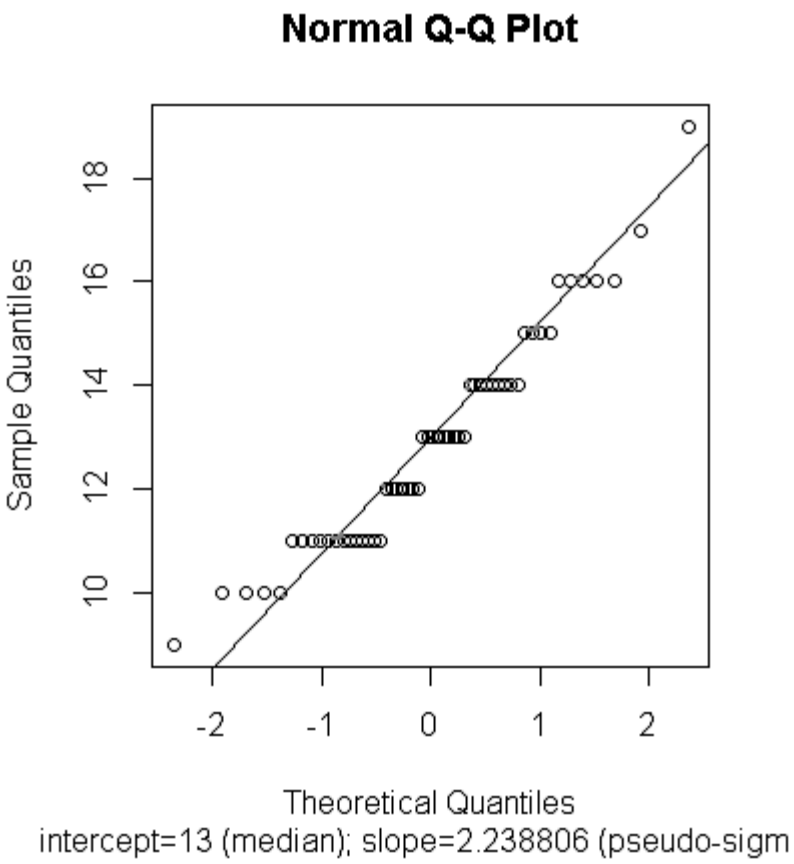
qqnorm 결과 직선의 형태를 띄고 있음을 알 수 있었다.

```
> mean(data)
[1] 12.87037
> var(data)
[1] 4.379106
> sd(data)
[1] 2.092631
>
> x <- fivenum(data)
> x
[1]  9 11 13 14 19
> (pseudosigma <- (x[4]-x[2])/1.34)
[1] 2.238806
```

pseudosigma는 lower hinge, upper hinge는 두 값으로 계산한다. pseudosigma와 sd(data)가 비슷하다는 것은 정규분포한다는 사실을 알려주는데 상단의 결과를 통해 두 값은 유사함을 알 수 있다.

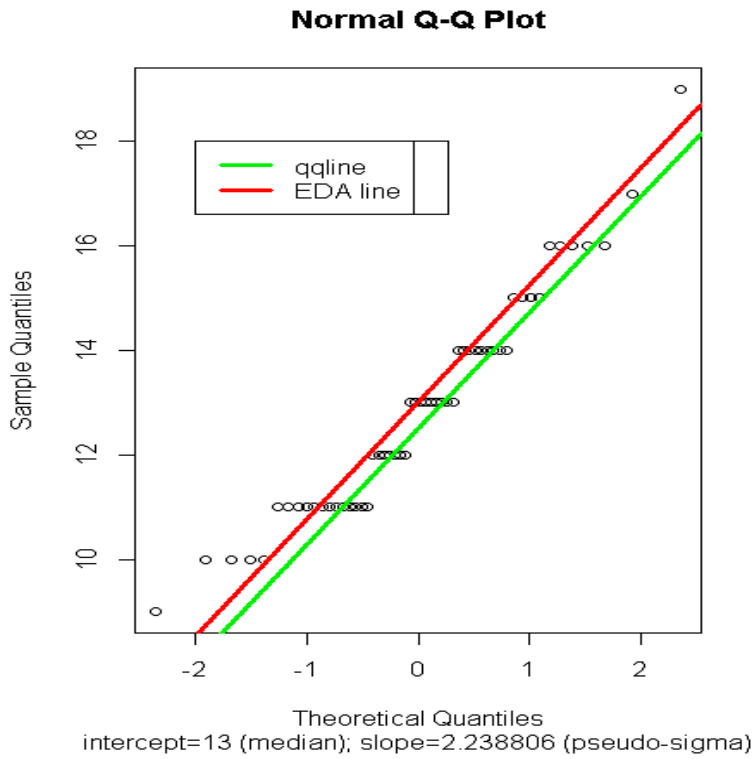
또한, 상단의 qqnorm 결과에서 보조선을 통해 더 자세하게 살펴보았다.

```
qqnorm(data)
abline(x[3],pseudosigma)
title(sub="intercept=13 (median); slope=2.238806 (pseudo-sigma)")
```



median과 pseudosigma를 활용해 그린 plot의 결과는 상단의 plot과 같다.

```
qqnorm(score, ylab="Scores of Students",main="Normal Prob Plot")
score.x = quantile(data, probs=c(0.25,0.75), names = FALSE,na.rm = TRUE)
score.y = qnorm(c(0.25,0.75))
slope = diff(score.x)/diff(score.y)
int = score.x[1] - slope * score.y[1]
abline(int, slope, col="green2", lwd=3)
pseudosigma = (x[4]-x[2])/1.34
abline(x[3],pseudosigma, col="red", lwd=3)
legend(-2, 18, c("qqline", "EDA line"), col=c("green2", "red"), lty=1, lwd=3)
```



추가적으로, slope와 int 변수를 통해 제1사분위수와 제3사분위수를 통과하는 직선을 하나 그렸다. (초록색)
해당 경우에는 slope, 즉 직선의 기울기는 2.223903 그리고 int, 즉 절편은 12.5 였다.
또한, 위에서 그려준 바와 같이 절편으로는 자료의 중위값(13)을, pseudosigma(2.238806)를 기울기로 하여 빨간색 직선을 그려주었다. (빨간색)

-> 두 직선을 비교해 보았을 때, 빨간색 직선이 조금 더 데이터에 적합한 선임을 알 수 있었다.
최종적으로 자료에서 직접 구한 평균과 분산 추정치와 확률도에서 구한 평균과 분산 추정치가 일치하는지 결론을 내려보았다.

```
> mean(data)
[1] 12.87037
> var(data)
[1] 4.379106
> sd(data)
[1] 2.092631
>
> x <- fivenum(data)
> x
[1]  9 11 13 14 19
> (pseudosigma <- (x[4]-x[2])/1.34)
[1] 2.238806
```

```
> slope
[1] 2.223903
> pseudosigma
[1] 2.238806
>
> int
[1] 12.5
> x[3]
[1] 13
```

mean()을 통해 직접 구한 평균은 12.87였고 variance는 4.379106이었다. 또한, fivenum()을 통해 median은 13임을 알 수 있었고 이를 통해 pseudosigma를 구할 수 있었고 2.223903라는 값이 도출되었다. 따라서 분산 추정치는 (4.945745) pseudosigma^2임을 알 수 있다.
즉, 자료에서 직접 구한 평균과 분산 추정치와 확률도에서 구한 평균과 분산 추정치는 거의 유사하다는 결론을 내릴 수 있었다.

---

2. The frequency table gives the yield strength of a Bofores steel. The observed values are obtained as routine tests. x=yield strength in 1.275 kg/mm^2

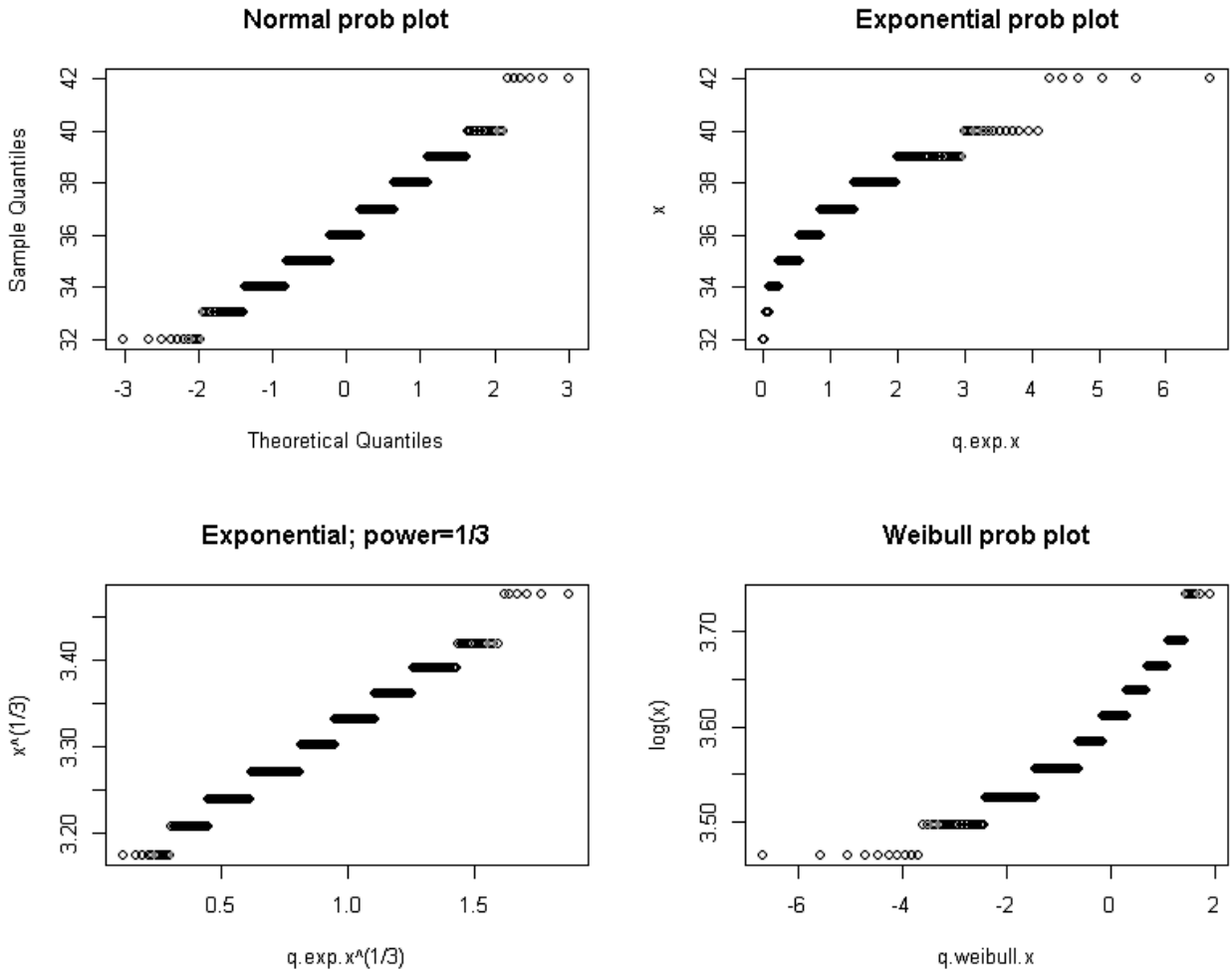
x	frequency
32	10
33	33
34	81
35	161
36	224
37	289
38	336
39	369
40	383
42	389

자료가 정규분포를 하는가? 자료가 지수분포 또는 weibull분포를 하는가? 어떤 분포가 가장 잘 맞는가? 가장 잘 맞는 분포의 parameter 추정값은? 자료에서 직접 구한 값과 일치하는가?

```
> x = c(rep(32,10),rep(33,23),rep(34,48),rep(35,80),rep(36,63),
+       rep(37,65),rep(38,47),rep(39,33),rep(40,14),rep(42,6))
> x
  [1] 32 32 32 32 32 32 32 32 32 32 33 33 33 33 33 33 33 33 33 33
 [21] 33 33 33 33 33 33 33 33 33 33 33 33 33 33 34 34 34 34 34 34 34
 [41] 34 34 34 34 34 34 34 34 34 34 34 34 34 34 34 34 34 34 34 34 34
 [61] 34 34 34 34 34 34 34 34 34 34 34 34 34 34 34 34 34 34 34 34 34
 [81] 34 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35
[101] 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35
[121] 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35
[141] 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35 35
[161] 35 36 36 36 36 36 36 36 36 36 36 36 36 36 36 36 36 36 36 36 36
[181] 36 36 36 36 36 36 36 36 36 36 36 36 36 36 36 36 36 36 36 36 36
[201] 36 36 36 36 36 36 36 36 36 36 36 36 36 36 36 36 36 36 36 36 36
[221] 36 36 36 36 37 37 37 37 37 37 37 37 37 37 37 37 37 37 37 37 37
[241] 37 37 37 37 37 37 37 37 37 37 37 37 37 37 37 37 37 37 37 37 37
[261] 37 37 37 37 37 37 37 37 37 37 37 37 37 37 37 37 37 37 37 37 37
[281] 37 37 37 37 37 37 37 37 37 38 38 38 38 38 38 38 38 38 38 38
[301] 38 38 38 38 38 38 38 38 38 38 38 38 38 38 38 38 38 38 38 38 38
[321] 38 38 38 38 38 38 38 38 38 38 38 38 38 38 38 38 39 39 39 39
[341] 39 39 39 39 39 39 39 39 39 39 39 39 39 39 39 39 39 39 39 39 39
[361] 39 39 39 39 39 39 39 39 39 40 40 40 40 40 40 40 40 40 40 40
[381] 40 40 40 42 42 42 42 42 42
```

상단의 코드를 통해 데이터를 생성해주었다.
해당 데이터를 통해 Normal Probability Plot, Exponential Probability Plot, Weibull Probability Plot 그리고 원점에 많은 데이터가 몰려있기 때문에 Exponential Prob Plot에서 1/3승하여 재표현을 해주었다.

```
par(mfrow=c(2,2))
qqnorm(x, main='Normal prob plot')
n = length(x)
i = 1:n
q.exp.x = -log(1-(i-0.5)/n)
plot(q.exp.x, x, main="Exponential prob plot")
plot(q.exp.x^(1/3), x^(1/3),main="Exponential; power=1/3")
q.weibull.x = log(q.exp.x)
plot(q.weibull.x, log(x), main="Weibull prob plot")
```

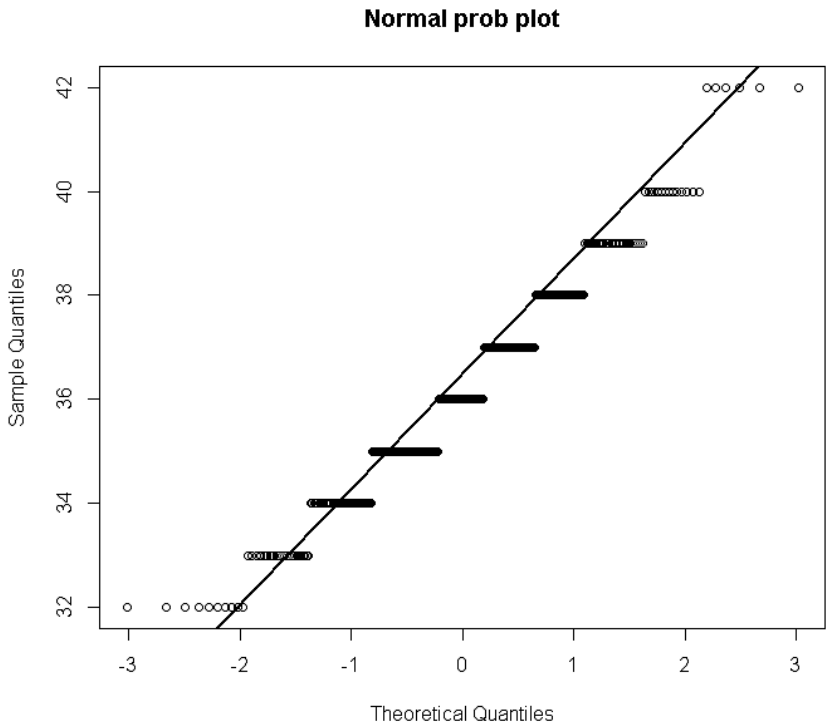


4개의 plot을 종합적으로 살펴보면 Normal Prob Plot이 가장 직선을 형태를 띄고 있음을 확인했다. 따라서, 해당 데이터는 정규분포를 따른다는 것을 알 수 있다.

```
> score.x = quantile(x, probs=c(0.25,0.75), names = FALSE,na.rm = TRUE)
> score.y = qnorm(c(0.25,0.75))
>
> slope = diff(score.x)/diff(score.y)
> int = score.x[1] - slope * score.y[1]
>
> slope
[1] 2.223903
> int
[1] 36.5
```

상단의 코드를 통해 slope와 int를 계산했다. slope는 2.223903 그리고 절편(int)은 36.5임을 확인했다.

```
qqnorm(x, main='Normal prob plot')
abline(int, slope, lwd=2)
```



해당 값을 토대로 상단의 plot을 그려보았다.

```
> mean(x)
[1] 36.1671
> var(x)
[1] 4.170461
> sd(x)
[1] 2.042171
```

다음으로 fivenum()을 통해 pseudosigma를 구해보았다.

```
> five <- fivenum(x)
> five
[1] 32 35 36 38 42
> (pseudosigma <- (five[4]-five[2])/1.34)
[1] 2.238806
```

다음으로, 정규분포의 parameter 추정값을 살펴보자. 그 전에 자료에서 직접 구한 값을 살펴보면 mean값은 36.1671, variance는 4.170461임을 알 수 있었다.
또한, fivenum()을 통해 median은 36임을 알 수 있었고 이를 통해 pseudosigma를 구할 수 있었고 2.238806라는 값이 도출되었다. 따라서 분산 추정치는 (5.012252) pseudosigma^2임을 알 수 있다. 즉, 자료에서 직접 구한 평균과 분산 추정치와 확률도에서 구한 평균과 분산 추정치는 유사하다는 결론을 내릴 수 있었다.

---

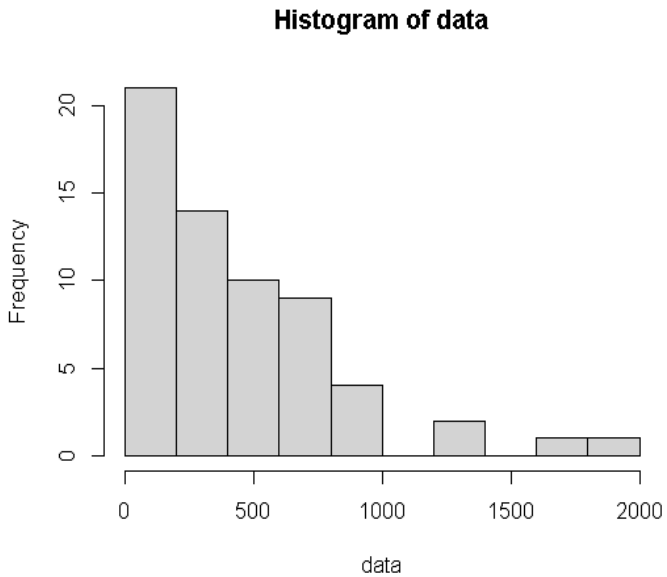
3. [QUAKES.DAT] 1907년부터 1977년 사이에 발생한 Richter 규모 7.5 이상 또는 1000명 이상 사망자를 낸 지진들의 발생시간 자료이다. 지수분포 또는 weibull분포를 하는지 알아보고 파라미터 추정값이 표본에서 직접 구한 값과 일치하는지 비교하여라. 총 63건의 지진이 있었고, 그 중 62개가 기록되었다.

```
> setwd("D:\\2022-1(3-2)\\2022-01_탐자분\\Data")
> data <- read.table("QUAKES.DAT", header=FALSE, fill = TRUE)
> data=unlist(data)
> names(data)=NULL
> data <- data[!is.na(data)]
> data <- sort(data)
>
> length(data)
[1] 62
```

```
> data
 [1]    9   30   33   36   38   40   40   44   46   76   82
[12]   83   92   99  121  129  139  145  150  157  194  203
[23]  209  220  246  263  280  294  304  319  328  335  365
[34]  375  384  402  434  436  454  460  556  562  567  584
[45]  599  638  667  695  710  721  735  736  759  780  832
[56]  840  887  937 1336 1354 1617 1901
>
```

데이터를 불러온 후 적절하게 전처리를 해주었다.

```
> source("http://mgimond.github.io/ES218/es218.R")
> lsum(data)
  letter depth lower    mid  upper spread
1      M  31.5 331.5 331.50  331.5    0.0
2      H  16.0 129.0 398.00  667.0  538.0
3      E   8.5  45.0 425.50  806.0  761.0
4      D   4.5  37.0 586.75 1136.5 1099.5
5      C   2.5  31.5 758.50 1485.5 1454.0
>
> fivenum(data)
[1]    9.0  129.0  331.5  667.0 1901.0
>
> stem(data)

  The decimal point is 2 digit(s) to the right of the |

   0 | 13344444588890234569
   2 | 01256890234788
   4 | 034566678
   6 | 0470124468
   8 | 3494
  10 |
  12 | 45
  14 |
  16 | 2
  18 | 0
```

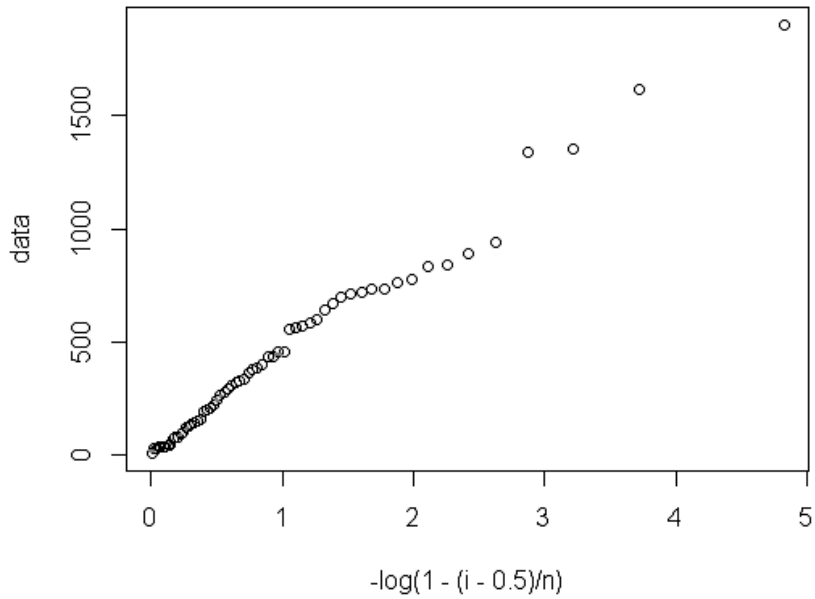


줄기잎그림과 histogram을 종합적으로 살펴본 결과, right skewed의 형태를 볼 수 있었다.

(1) 지수분포

QUAKES.DAT가 지수분포에 적합한지 알아보기 위하여 지수분포와 QUAKES data의 probability plot을 그려주었다.

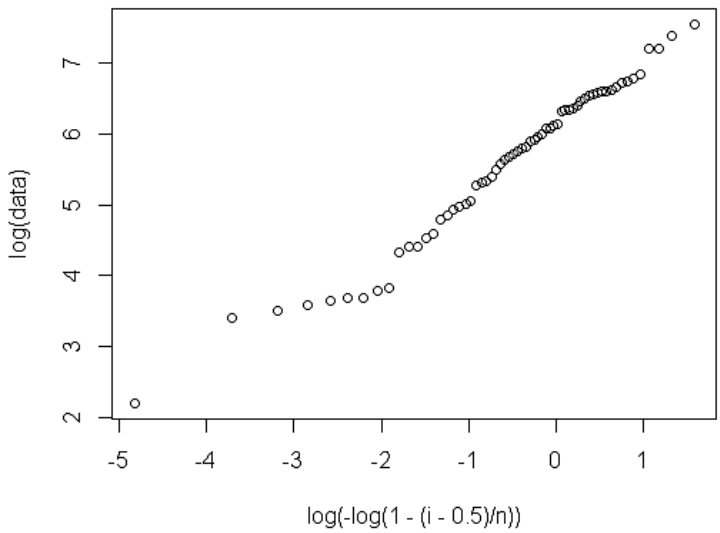
```
n <- length(data)
i <- 1:n
plot(-log(1-(i-0.5)/n), data)
```



그 결과, 선형의 형태를 살펴보기 어려웠다. 따라서, weibull분포를 시도해보기로 했다.

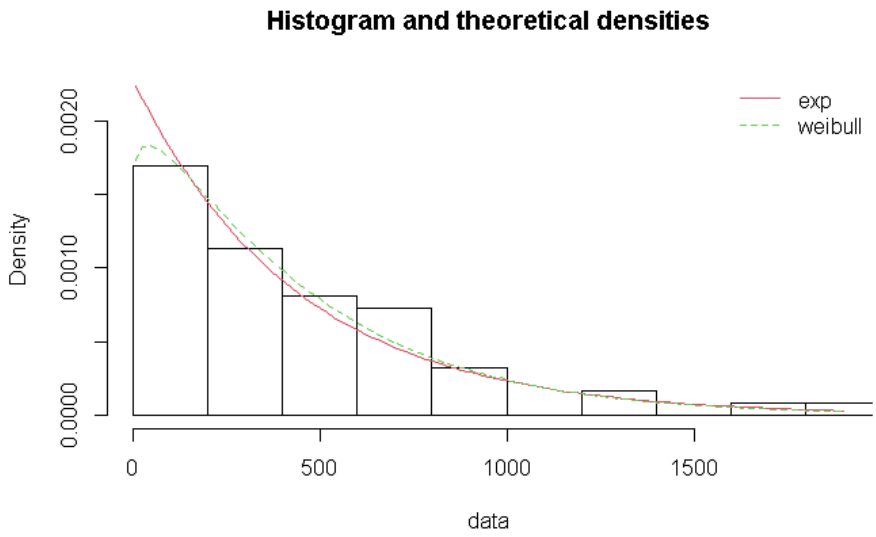
(2) weibull분포
weibull 분포와 적합한지 확인하기 위하여 아래의 코드를 활용해서 probability plot을 그려주었다.

```
plot(log(-log(1-(i-0.5)/n)), log(data))
```

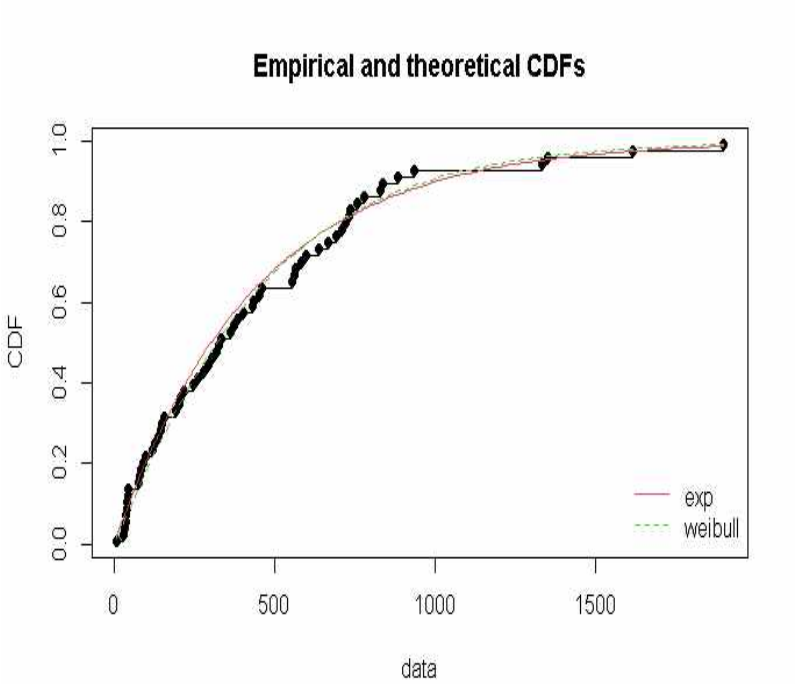


지수분포에 비해 데이터가 몰려있는 부분(지수분포의 경우 0에 몰려있음을 확인)이 적었고 조금 더 직선의 형태를 지니고 있음을 확인했기 때문에 최종적으로 weibull분포를 선택하고자 했다.
더 정확한 결론을 위해 아래의 코드를 실행하고 결과를 분석해보았다.

```
library(fitdistrplus)
f_e <- fitdist(data, "exp")
f_w <- fitdist(data, "weibull")
denscomp(list(f_e, f_w))
```

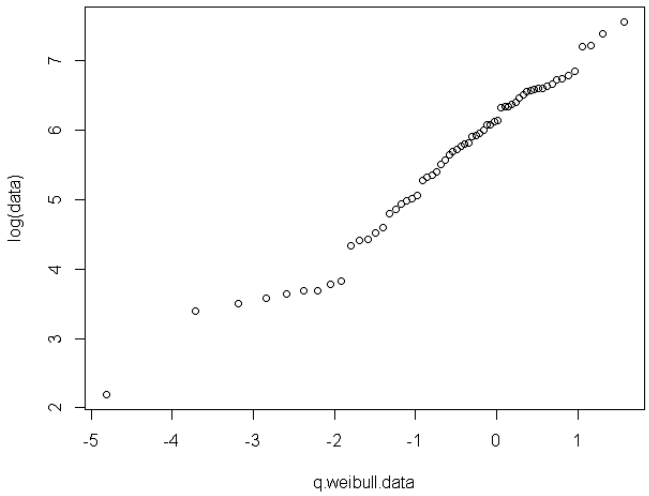



```
cdfcomp (list(f_e, f_w))
```



그 결과, weibull 분포가 데이터에 조금 더 fitting하고 있음을 확인할 수 있었다.
최종적으로 weibull 분포를 그리고 파라미터를 추정해보았다.

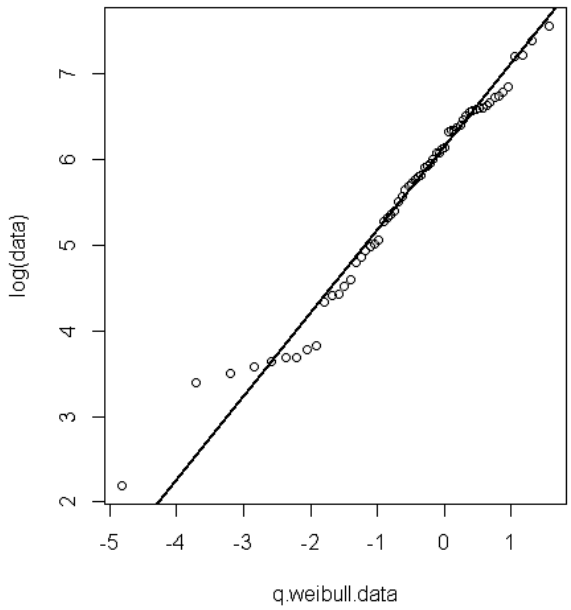
```
n <- length(data)
i <- 1:n
q.exp.data <- -log(1-(i-0.5)/n)
q.weibull.data <- log(q.exp.data)
plot(q.weibull.data, log(data), main="Weibull prob plot")


line(qqplot(q.weibull.data,log(data)))
```



와이블 분포 확률 Plot은 위와 같다. 또한,

```
>
> line(qqplot(q.weibull.data,log(data)))

Call:
line(qqplot(q.weibull.data, log(data)))

Coefficients:
[1]  6.1451  0.9716
```

상단의 결과를 바탕으로 abline(6.1451,0.9716, lwd=2) 코드를 실행하여 보조선을 그어주었다.



a=e^(intercept*(-b)),b=1/slope

또한, 위의 식을 활용해서 계산해본다면 다음과 같은 추정값을 얻을 수 있다.

```
> (b=1/0.9716)
[1] 1.02923
> (a=exp(6.1451*(-b)))
[1] 0.001791464
```

상단의 추정값을 아래 코드 실행을 통해 표본에서 직접 구한 값과 비교해보면 거의 유사함을 알 수 있다.

```
> (mean.data <- mean(data))
[1] 437.2097
> (var.data <- var(data))
[1] 159941.8
> (shape.data <- mean.data^2/var.data)
[1] 1.195136
> (scale.data <- mean.data/var.data)
[1] 0.002733554
>
```

---

4. 아래 자료는 주어진 시간에 발생하는 방사능 물질의 배출 건수를 조사한 것이다. Poisson 분포에 적합하는지 판단하여라

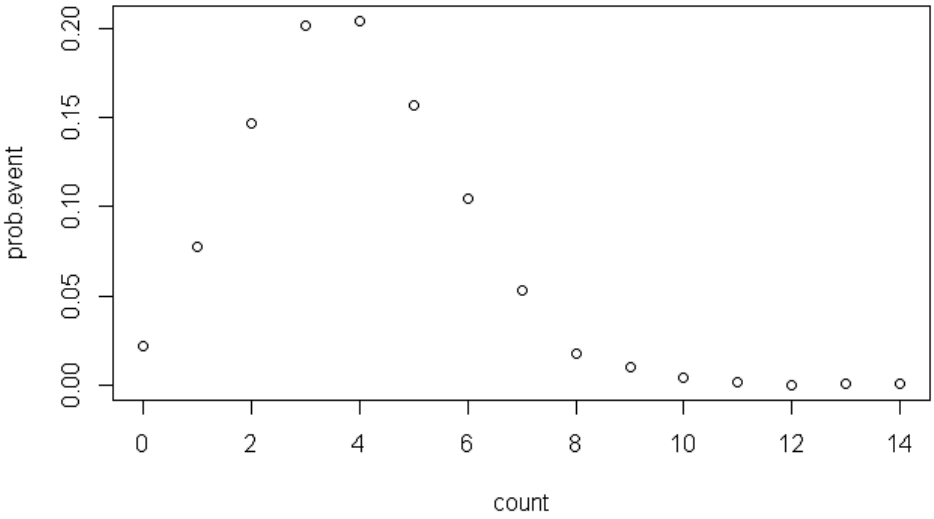
count	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
frequency	57	203	383	525	532	408	272	139	46	27	10	4	0	1	1

```
> count <- c(0, 1, 2, 3, 4, 5, 6, 7,8, 9, 10, 11, 12, 13, 14)
> frequency <- c(57,203,383,525,532,408,272,139,46,27,10,4,0,1,1)
> (n <- sum(frequency))
[1] 2608
> data <- rep(count[1],frequency[1])
> is.vector(data)
[1] TRUE
> for ( i in 2:15 ) data <- c(data, rep(count[i],frequency[i]))
> f1 <- fitdist(data, "pois")
> summary(f1)
Fitting of the distribution ' pois ' by maximum likelihood
Parameters :
       estimate Std. Error
lambda 3.872316 0.03853289
Loglikelihood:  -5353.423   AIC:  10708.85   BIC:  10714.71
```

문제에서 제시한 방향대로 상단의 코드를 실행했고 lambda의 추정치는 3.872316임을 알 수 있었다.

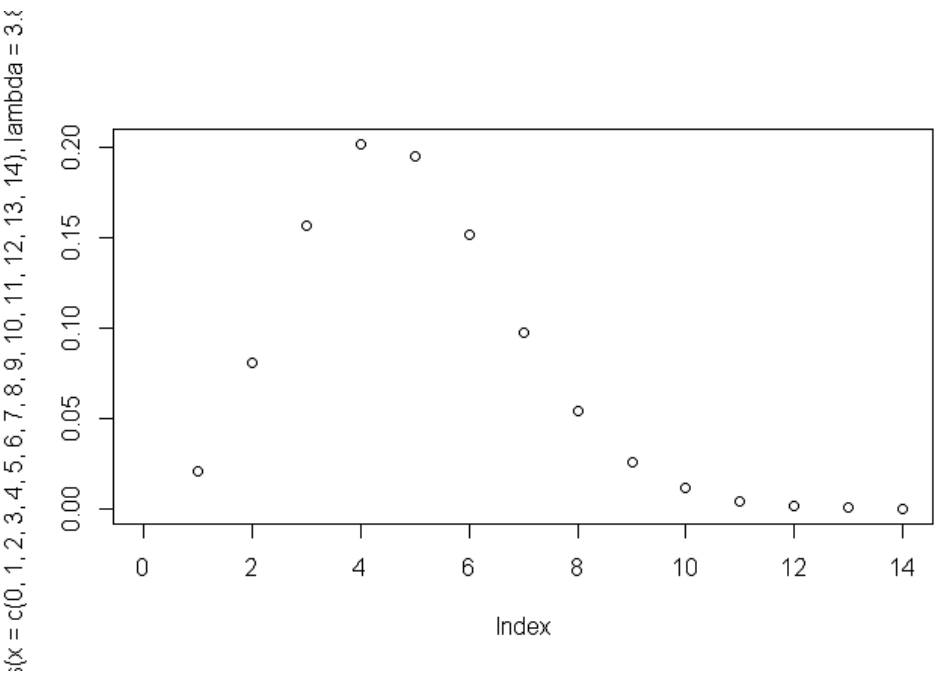
우선, 아래의 코드를 실행하여 분포를 살펴보자.

```
sum.freq<-sum(frequency)
prob.event<-frequency/sum.freq
plot(prob.event~count, data=data)
```



그리고 위에서 추정한 lambda 값을 통해 plot을 그려보자.

```
plot(dpois(x=c(0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14), lambda = 3.872316), xlim=c(0, 14))
```



두 개의 plot을 비교했을 때 유사한 양상을 띄고 있음을 확인할 수 있었다. 수치적으로 확인해보기 위해 gofstat(Goodness-of-fit statistics)도 실행해보았다.

참고 :
https://www.rdocumentation.org/packages/fitdistrplus/versions/1.1-8/topics/gofstat

```
> gofstat(fitdist(data,"pois"))$chisqpvalue
[1] 0.1236787
```

p-value가 0.617로 귀무가설을 지지한다. 즉, 포아송 분포임을 확인할 수 있었다.

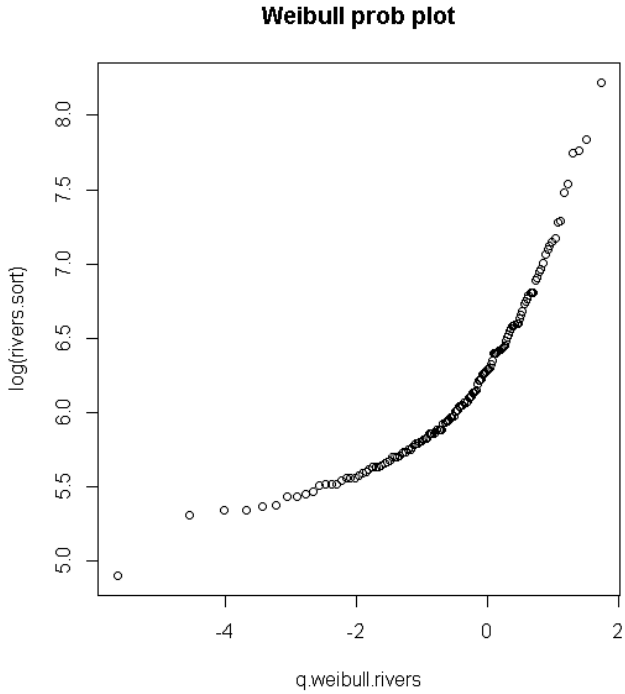
 # 동영상 수업 중 추가 HW

```
> stem(rivers)

  The decimal point is 2 digit(s) to the right of the |

   0 | 4
   2 | 011223334555566667778888899900001111223333444555556666888888999
   4 | 1111223334455667790012333445 67
   6 | 000112233578012234468
   8 | 045790018
  10 | 04507
  12 | 1471
  14 | 56
  16 | 7
  18 | 9
  20 |
  22 | 25
  24 | 3
  26 |
  28 |
  30 |
  32 |
  34 |
  36 | 1
```

rivers 데이터의 stem을 그려봄으로써 대략적인 분포를 살펴보았다.

다음으로, weibull 분포를 확인해보았다.

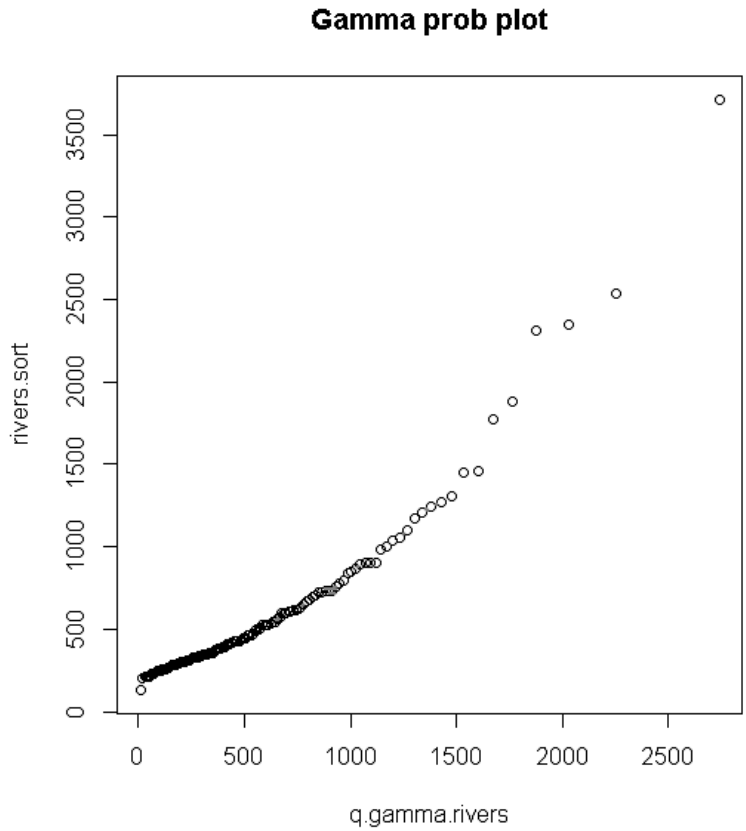
```
q.exp.rivers <- -log(1-(i-0.5)/n.rivers)
q.weibull.rivers <- log(q.exp.rivers)
plot(q.weibull.rivers, log(rivers.sort), main="Weibull prob plot")
```



그 후, 감마분포를 확인해보았다.

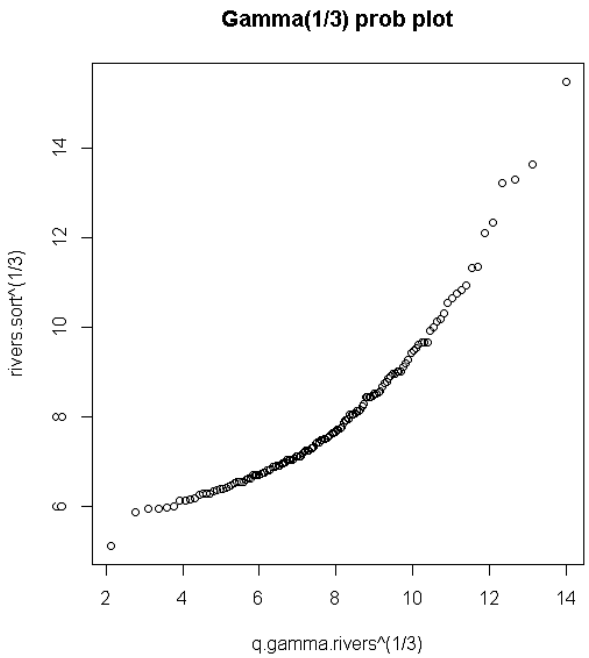
```
> rivers.sort <- sort(rivers)
> (n.rivers <- length(rivers))
[1] 141
> i <- 1:n.rivers
> mean.rivers <- mean(rivers)
> mean.rivers
[1] 591.1844
> var.rivers <- var(rivers)
> var.rivers
[1] 243908.4
> shape.rivers <- mean.rivers^2/var.rivers
> shape.rivers
[1] 1.432911
> scale.rivers <- mean.rivers/var.rivers
> scale.rivers
[1] 0.002423797
```

```
> q.gamma.rivers <- qgamma((i-0.5)/n.rivers, shape.rivers, scal
e.rivers)
> plot(q.gamma.rivers, rivers.sort, main="Gamma prob plot")
```



변환 전 자료들은 원점에 많이 몰려 있음을 확인할 수 있었다. 원점 근처에 자료들이 분포하고 있는 형태를 확인하기 어렵다. 따라서 1/3승을 취하여 데이터가 퍼지도록 했다.

```
plot(q.gamma.rivers^(1/3), rivers.sort^(1/3), main="Gamma(1/3) prob plot")
```



그 결과, 상단의 그래프 결과를 얻을 수 있었다. 이를 살펴보면 점들이 convex한 양상을 띄고

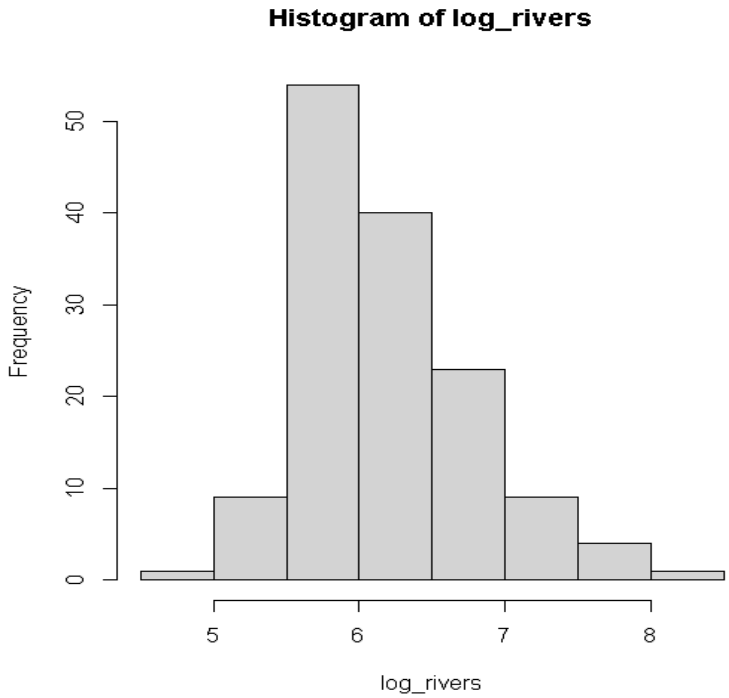
있음을 확인할 수 있었다. 선형의 형태가 아니기 때문에 감마분포를 따르지 않는다고 결론 내릴 수 있다.

weibull 분포 그리고 감마분포 모두 적합하지 않음을 확인했다.

따라서, Rivers 데이터를 로그변환을 취한 뒤 분포를 살펴보았다.

```
> log_rivers <- log(rivers.sort)
> stem(log_rivers)

  The decimal point is 1 digit(s) to the left of the |

  48 | 1
  50 |
  52 | 15578
  54 | 44571222466689
  56 | 023334677000124455789
  58 | 00122366666999933445777
  60 | 122445567800133459
  62 | 112666799035
  64 | 000113345812578889
  66 | 003683579
  68 | 0019156
  70 | 079357
  72 | 89
  74 | 84
  76 | 56
  78 | 4
  80 |
  82 | 2
```

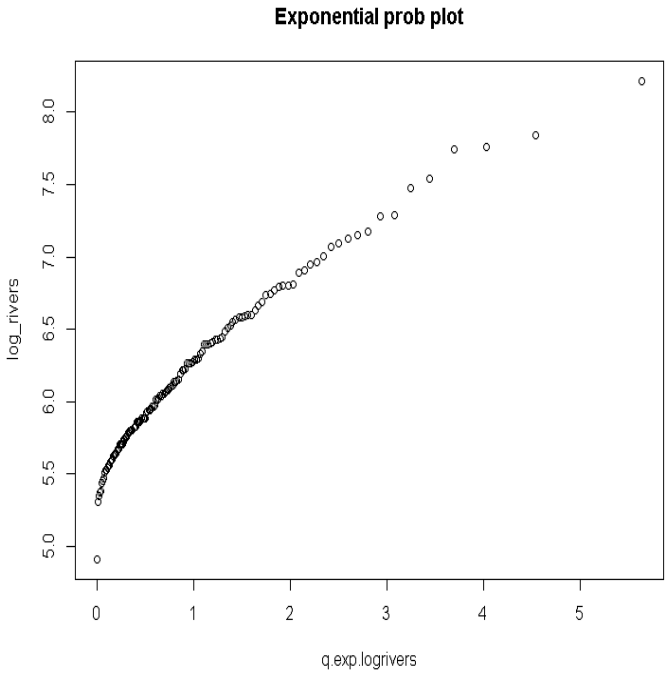
hist(log_rivers)



줄기잎그림과 히스토그램을 살펴본 결과 오른쪽으로 skewed되어있는 형태를 확인할 수 있었다.

```
> (n.logrivers <- length(log_rivers))
[1] 141
> i <- 1:n.logrivers
> q.exp.logrivers <- -log(1-(i-0.5)/n.logrivers)
> plot(q.exp.logrivers, log_rivers, main="Exponential prob plot")
```

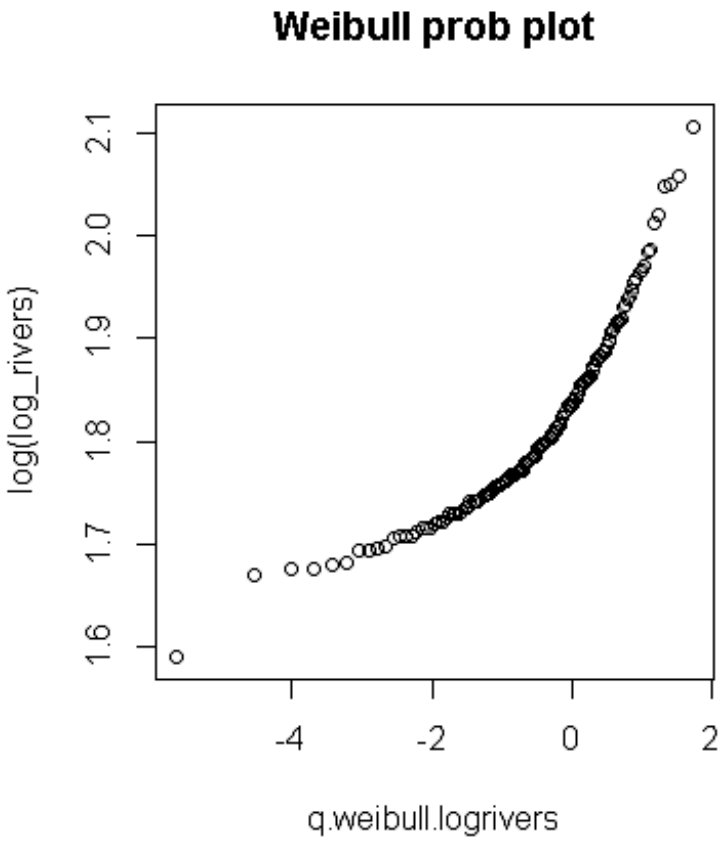
가장 먼저 지수분포를 살펴보았고 그 결과 아래의 그래프를 결과로 얻을 수 있었다.



결과를 통해 곡선의 양상을 확인할 수 있었기 때문에 지수분포는 적합하지 않음을 결론지었다.

다음으로, weibull 분포를 살펴보았다.

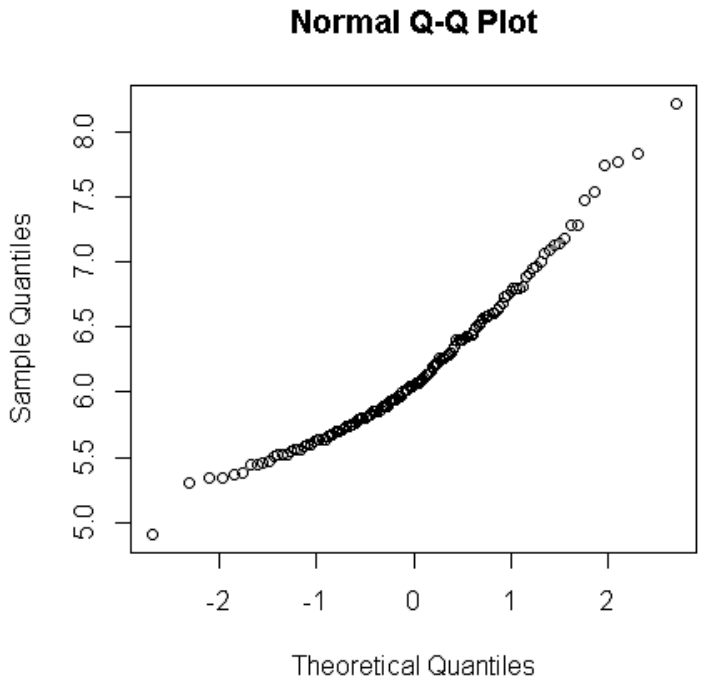
```
q.weibull.logrivers <- log(q.exp.logrivers)
plot(q.weibull.logrivers, log(log_rivers), main="Weibull prob plot")
```



그 결과 곡선의 형태가 더욱 심해졌음을 확인할 수 있었다.

다음으로, 정규분포를 시도해보았다.

qqnorm(log_rivers)



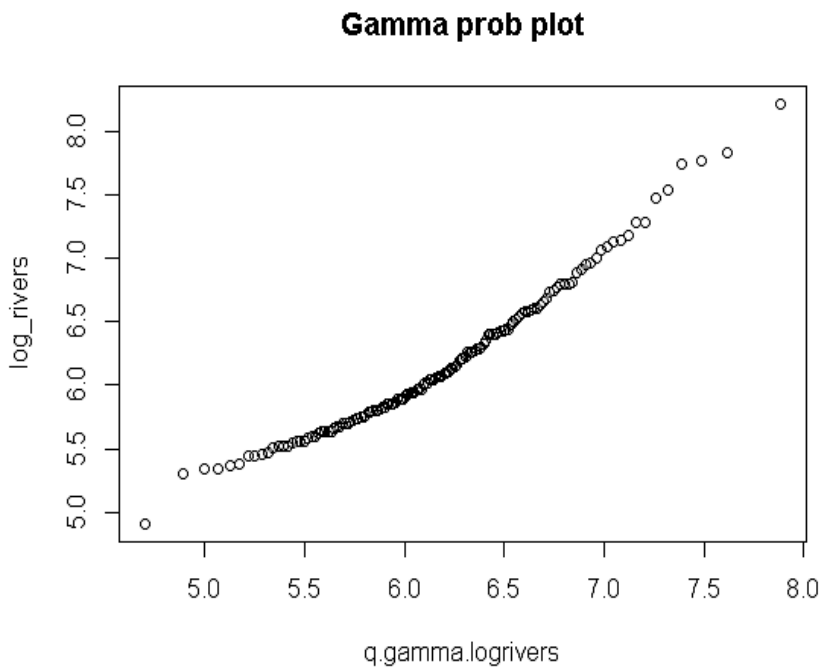
정규분포도 마찬가지로 곡선의 형태를 보인다.
정규분포인 경우 lsum을 살펴보았을 때 mid값이 증가해야한다. 그러나 log_rivers의 경우 분위수의 증가에 따라 mid값이 증가하는 것을 알 수 있다. 따라서 정규분포도 적합하지 않다.

```
> lsum(log_rivers)
  letter depth    lower      mid    upper    spread
1      M  71.0 6.052089 6.052089 6.052089 0.0000000
2      H  36.0 5.736572 6.129333 6.522093 0.7855205
3      E  18.5 5.570206 6.209506 6.848806 1.2786002
4      D   9.5 5.459549 6.310149 7.160748 1.7011989
5      C   5.0 5.370638 6.456161 7.541683 2.1710451
```

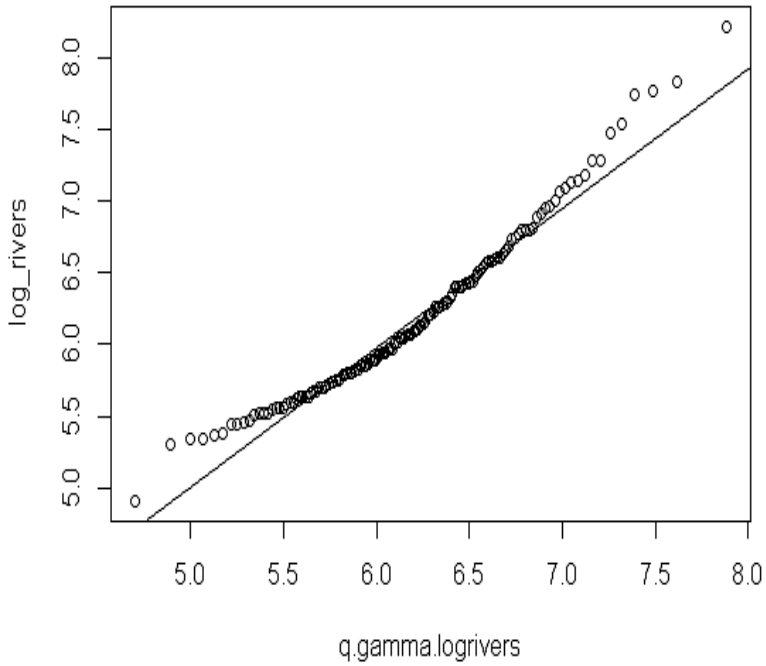
마지막으로 감마분포를 시도해보았다.

```
> mean.logrivers <- mean(log_rivers)
> mean.logrivers
[1] 6.175879
> var.logrivers <- var(log_rivers)
> var.logrivers
[1] 0.3498534
> shape.logrivers <- mean.logrivers^2/var.logrivers
> shape.logrivers
[1] 109.0213
> scale.logrivers <- mean.logrivers/var.logrivers
> scale.logrivers
[1] 17.65276
>
> q.gamma.logrivers <- qgamma((i-0.5)/n.logrivers, shape.logrivers,
 scale.logrivers)
```

plot(q.gamma.logrivers, log_rivers, main="Gamma prob plot")



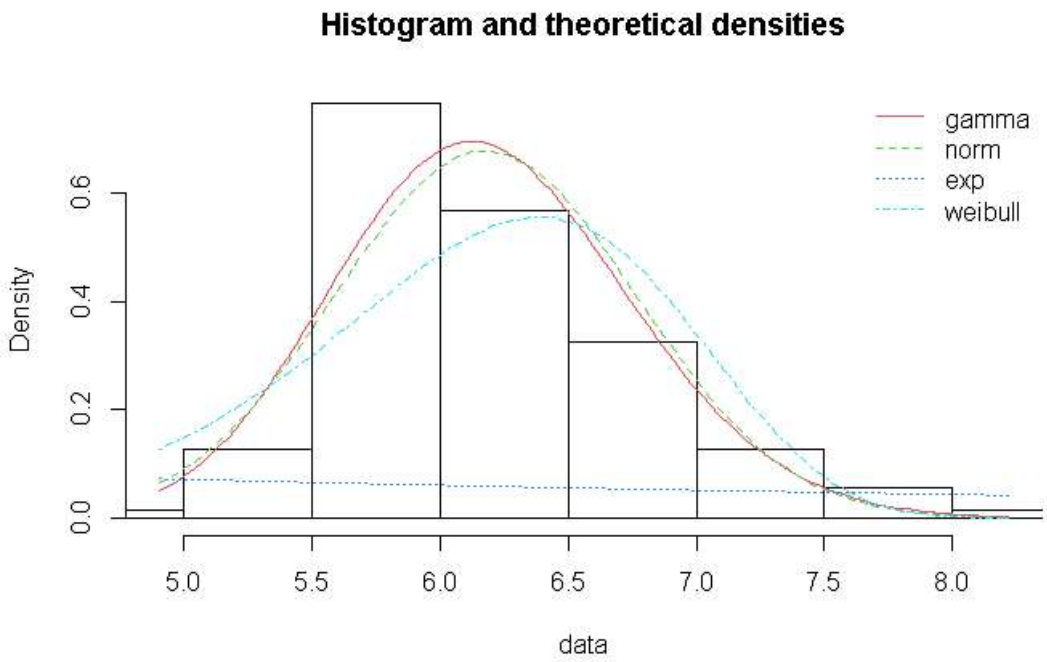
abline(line(qqplot(q.gamma.logrivers, log_rivers)))



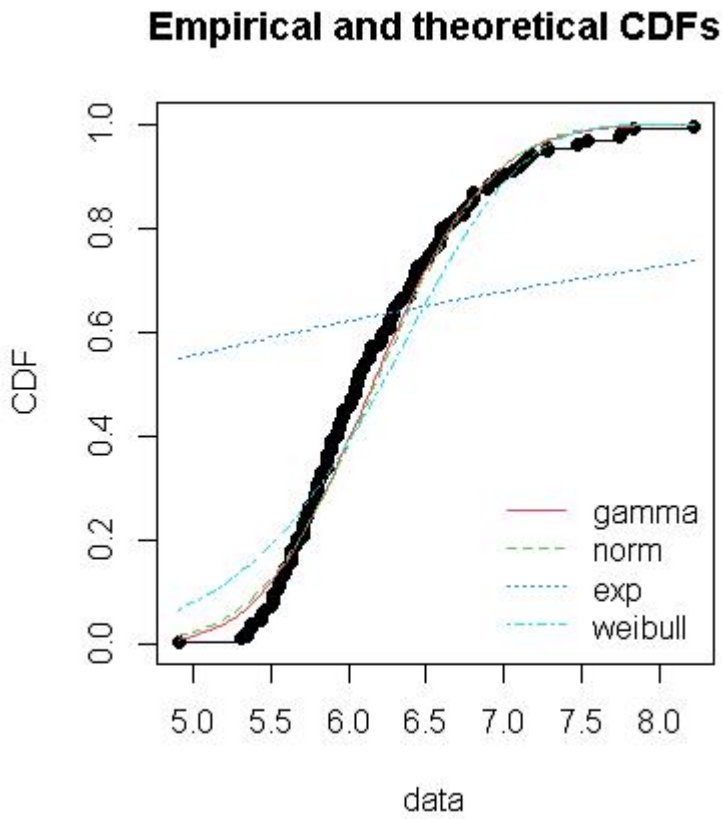
이전에 시도해본 결과들과 유사한 prob plot을 확인할 수 있었다. 또한, Tukey의 robust line estimation을 활용해서 선을 그렸을 때도 직선과 거의 fitting되지 않음을 확인했다.

비록 4개의 분포 모두 적합하지 않았지만, 감마분포가 그나마 직선의 형태를 띄고 있었다. 더 정확한 분포를 살펴보기 위해 아래의 코드를 수행해보았다.

```
> library(fitdistrplus)
> f_g <- fitdist(log_rivers, "gamma")
> f_n <- fitdist(log_rivers, "norm")
> f_e <- fitdist(log_rivers, "exp")
> f_w <- fitdist(log_rivers, "weibull")
> denscomp(list(f_g, f_n, f_e, f_w))
```



cdfcomp(list(f_g, f_n, f_e, f_w))



그 결과, 감마분포와 정규분포가 log_rivers의 데이터에 가장 적합한 분포라는 결론을 내릴 수 있었다.