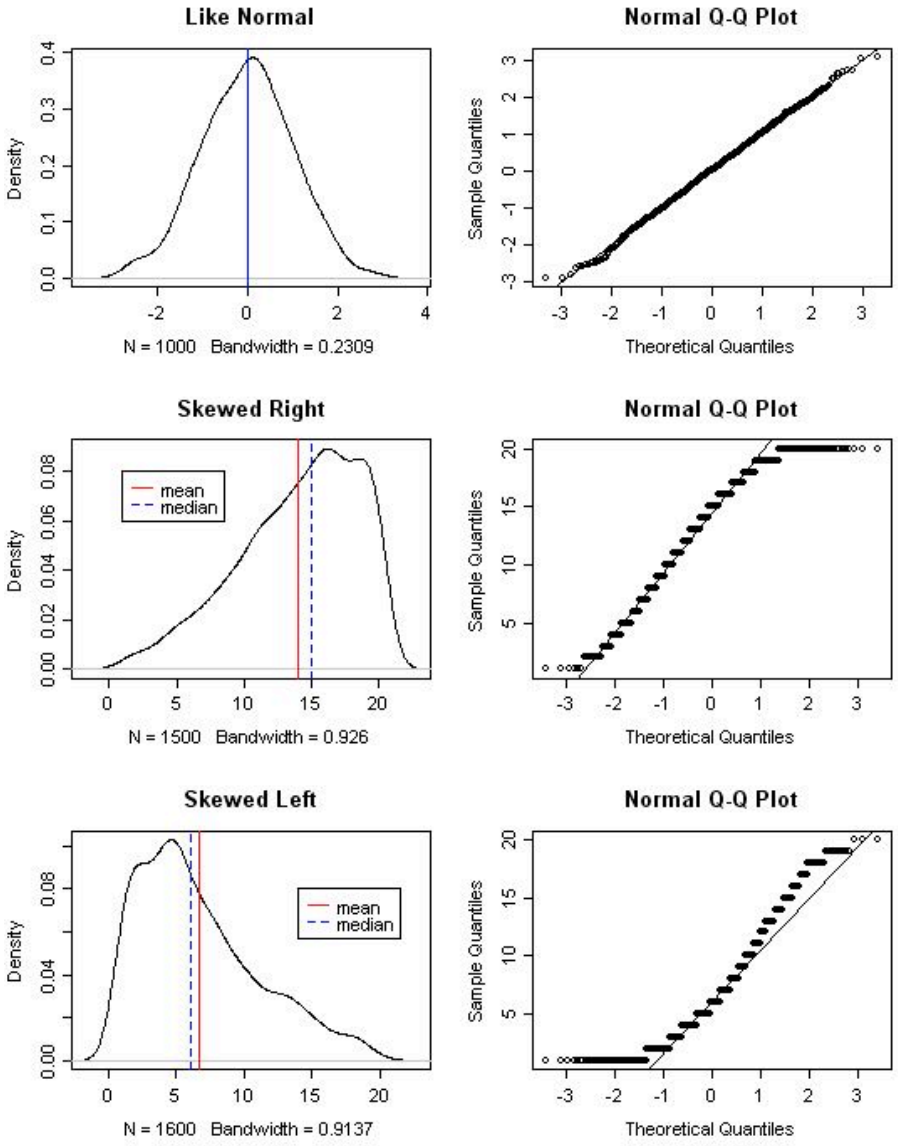
분포에 대해서

유 충현
블로그 모음 13탄(http://blog.naver.com/bdboys) • (주)오픈베이스 • 2010년 11월 7일



분포에 대해서

자료의 분포를 설명하는 통계량에 자료의 중심을 설명하는 대표치와 자료의 퍼짐을 설명하는 산포도(분산)이 있다.

대표치로는 평균, 중위수, 최빈수 등이 있고, 퍼진 정도를 표현하는 분산, 분산의 제곱근인 표준편차 등이 있다. 또한 자료가 어느쪽으로 편중되었는지의 기울기를 나타내는 왜도, 대표치 부근에 자료가 밀집한 정도를 나타내는 첨도 등이 있다.

이번에는 왜도를 중심으로 이야기를 해 본다.

자료의 대칭성을 설면하는 왜도는 0이면 분포가 좌우 대칭을 이루고 음수이면 왼쪽으로 치우친 분포를, 양수이면 오른쪽으로 치우친 분포를 이룬다. 첨도는 3이면 정규분포 곡선과 유사하며, 3보다 크면 정규분포 곡선보다 정점이 높고 뾰족한 분포며 3보다 작으면 정규분포보다 정점이 낮고 퍼진 모양의 분포이다.

왜도가 1이면 중위수, 평균(산술평균), 최빈수가 같은 값을 가지며 자료가 왼쪽으로 치우친 분포라면 왜도는 음수이며 최빈수<중위수<평균의 관계가 자료가 오른쪽으로 치우친 분포라면 왜도는 양수이며 평균<중위수<최빈수의 관계가 성립함을 배운적이 있다. 그러면 이를 보여줄 데이터를 만들고 그림으로 그려보면서 왜도의 값에 따른 평균, 중위수, 최빈수의 관계를 이해해 보자.

다음과 같은 프로그램을 만들어 보았다.

```
> # 최빈수를 구하는 함수
> get.mode <- function(x) {
+ tbl=table(x)
+  as.numeric(names(tbl[which(tbl==max(tbl))]))
+ }

> # 왜도를 구하는 함수
> get.skewness <- function(x) {
+  sum((x-mean(x))^3)/(length(x)-1) / sd(x)^3
+ }

> # 첨도를 구하는 함수
> get.kurtosis<- function(x) {
+  sum((x-mean(x))^4)/(length(x)-1) / sd(x)^4  - 3
+ }

> ############################
> # 왜도가 0에 근사한 데이터를 생성
> ############################

> x=round(rnorm(1000),5)  # 정규난수 1000개 생성
```

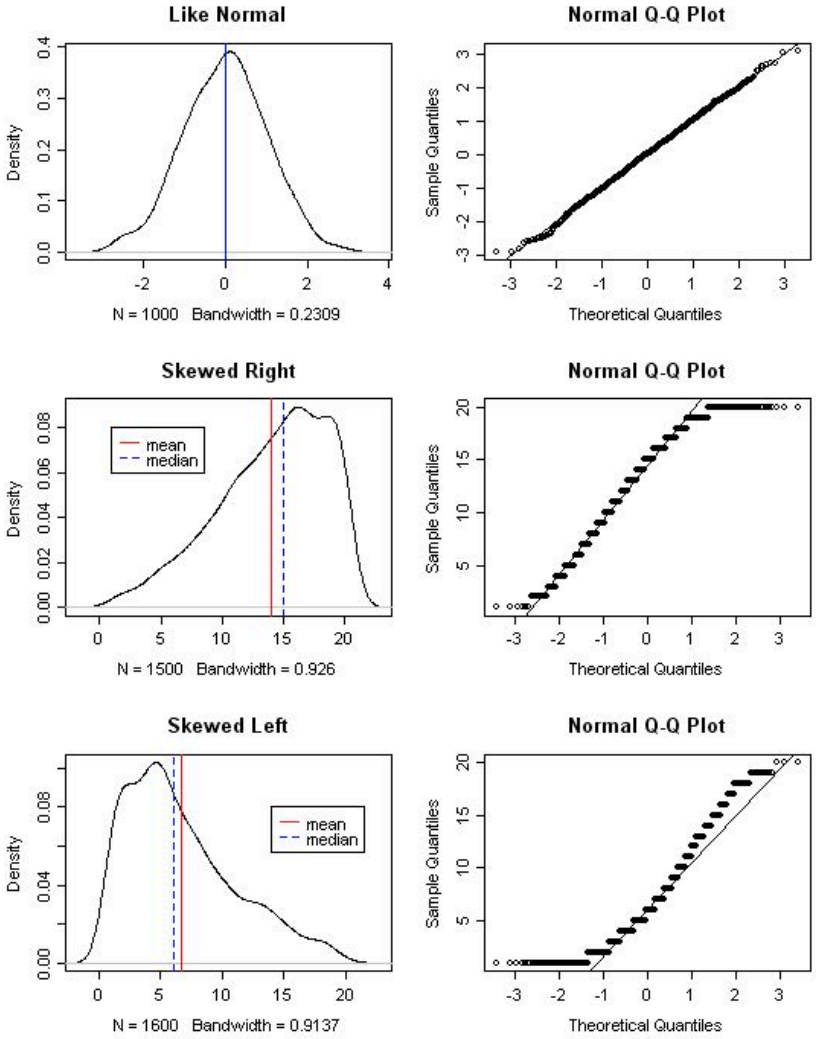
```
> x.mean=mean(x)            # 평균
> x.median=median(x)        # 중위수
> x.mode=get.mode(x)        # 최빈수

> x.mean
[1] -0.00464817
> sd(x)     # 표준편차
[1] 1.024234
> x.median
[1] 0.015635
> x.mode
[1] 0.19721 1.30967
> summary(x)
     Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
-2.916000 -0.684500  0.015640 -0.004648  0.684200  3.067000
> get.skewness(x) # 왜도
[1] -0.04879586
> get.kurtosis(x) # 첨도
[1] -0.08690496


> ##########################
> # 왜도가 음수인 데이터를 생성
> ##########################
> y=numeric(0)
> for(i in 1:15) {
+  y <<- c(y,sample(i:20,100,replace=T))
+ }

> y.mean=mean(y)
> y.median=median(y)
> y.mode=get.mode(y)

> y.mean
[1] 14.026
> sd(y)
[1] 4.44205
> y.median
[1] 15
> y.mode
[1] 16
> summary(y)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   1.00   11.00   15.00   14.03   18.00   20.00
> get.skewness(y)
[1] -0.660706
> get.kurtosis(y)
[1] -0.2496434

> ##########################
> # 왜도가 양수인 데이터를 생성
> ##########################
> z=numeric(0)
```

```
> for(i in 5:20) {
+   z <<- c(z,sample(1:i,100,replace=T))
+ }

> z.mean=mean(z)
> z.median=median(z)
> z.mode=get.mode(z)

> z.mean
[1] 6.6575
> sd(z)
[1] 4.440019
> z.median
[1] 6
> z.mode
[1] 5
> summary(z)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.000   3.000   6.000   6.658   9.000  20.000
> get.skewness(z)
[1] 0.8242766
> get.kurtosis(z)
[1] -0.03099681
```

> # 이하 그래프 출력

```
> par(mfrow=c(3,2))
> plot(density(x), main="Like Normal")
> abline(v=x.mean,col="red")
> abline(v=x.median,col="blue")
> #abline(v=x.mode,col="black")
> qqnorm(x)
> qqline(x)
>
> plot(density(y), main="Skewed Right")
> abline(v=y.mean,col="red", lty=1)
> abline(v=y.median,col="blue", lty=2)
> #abline(v=y.mode,col="black")
> #legend(locator(1),legend=c("mean","median"),col=c
("red","blue"),lty=1:2)
> legend(1,0.08,legend=c("mean","median"),col=c("red","blue"),lty=1:2)
> qqnorm(y)
> qqline(y)
>
> plot(density(z), main="Skewed Left")
> abline(v=z.mean,col="red", lty=1)
> abline(v=z.median,col="blue", lty=2)
> #abline(v=z.mode,col="black")
> #legend(locator(1),legend=c("mean","median"),col=c
("red","blue"),lty=1:2)
> legend(14,0.08,legend=c("mean","median"),col=c("red","blue"),lty=1:2)
> qqnorm(z)
> qqline(z)
```

결과를 실행시키면 다음과 같은 그래프를 얻을 수 있다. 왼쪽의 그림들이 분포의 그림이고, 오른쪽은 정규분포에 근사하는지를 검증하기 위한 Q-Q Plot이다.
첫번째 그림은 정규분포에 근사하고,
두번째 그림은 오른쪽으로 치우친 분포고,
세번째 그림은 왼쪽으로 치우친 분포의 그래프임을 알 수 있다.



이항분포의 정규분포 근사

이항분포 B(n,p)에서 p값에 관계없이 n값이 충분히 커지면 이 분포는 정규분포에 근사하게 된다. 이것을 그래프로 그려 정규분포로 근사하는 것을 추적해 보았다. p=0.2이고 n=10,20,30,40,50인 이항분포를 한 좌표에 그려 보았다.

```
> jpeg("binom.jpg",550,550)
> plot(dbinom(1:20,10,0.2),type="l", main="이항분포의 정규분포 근사; B(n, 0.2)", xlab="X", ylab="Probability")
> lines(dbinom(1:20,20,0.2), lty=2, col=2)
> lines(dbinom(1:20,30,0.2), lty=3, col=3)
> lines(dbinom(1:20,40,0.2), lty=4, col=4)
> lines(dbinom(1:20,50,0.2), lty=5, col=5)
> legend(10,0.27,legend=c("B(10,0.2)","B(20,0.2)","B(30,0.2)","B(40,0.2)","B(50,0.2)"),lty=1:5, col=1:5)
> dev.off()
```