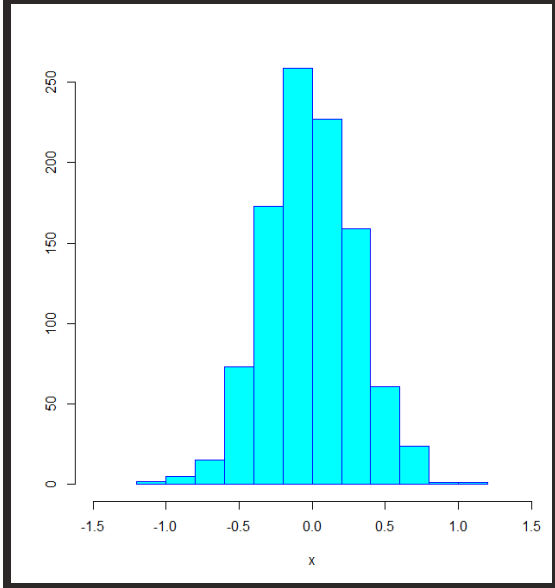
180604

in R 분포에 따른 함수 사용

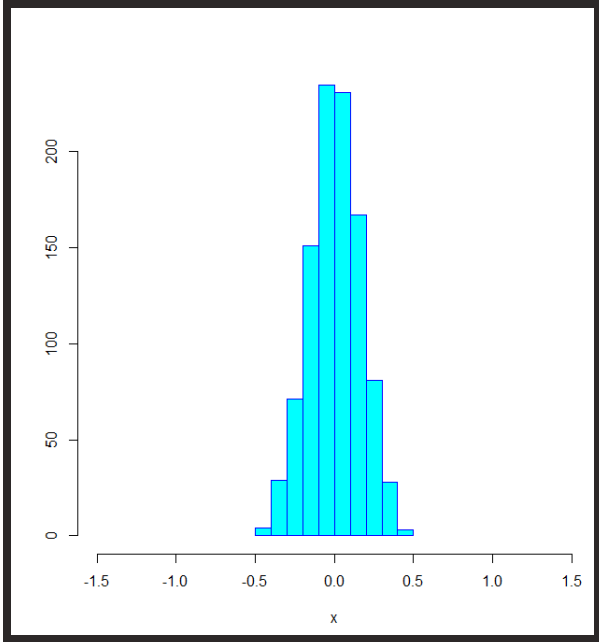
함수	**시작문자**	**함수형**	**함수형태**
확률함수 P(X=x)	d	norm chisq t f	dnrom(x, mean, sd) dchisq(x, df) dt(x, df) df(x, df1, df2)
분포함수 P(X≤ x)	p	norm chisq t f	pnorm(x, mean, sd) pchisq(x, df) pt(x, df) pf(x, df1, df2)
분위수 함수 **P(X≤ x) = q** **함수 원 방향! +** **norm, t-> a/2**	q	norm chisq t f	qnorm(q, mean, sd) qchisq(q, df) qt(q, df) qf(q, df1, df2)
난수생성함수	r	norm chisq t f	rnorm(n, mean, sd) rchisq(n, df) rt(n, df) rf(n, df1, df2)

실습1

```
> #테스트 목적 : 모집단이 정규분포를 따르는 각각 표본 크기가 10, 40 인 표본들을 1000 번 추출했을 때, 각 표본들이 이루는 확률분포가 모집단의 성향을 얼마나 대표하는지 파악

> m10 <- c()
> m40 <- c()


> set.seed(9)
#초기의 값 고정
9를 활용해서 난수를 처음 생성하는 값을 쓰겠다. - 초기값을 토대로 랜덤화.
~ 환경이 달라도 동일한 결과 나옴    for 실습 등 문제 정답 확인용


> for ( i in 1:1000) {
+   m10[i] <- mean(rnorm(10))
+   m40[i] <- mean(rnorm(40))
+ }

> options(digits=4)

> c(mean(m10), sd(m10))
[1] -0.01214  0.30311
> c(mean(m40), sd(m40))
[1] 0.004212 0.160942



> hist(m10, xlim=c(-1.5,1.5), main="", xlab="x", ylab="", col="cyan", border="blue") # border : 막대그래프의 선 색 설정
```

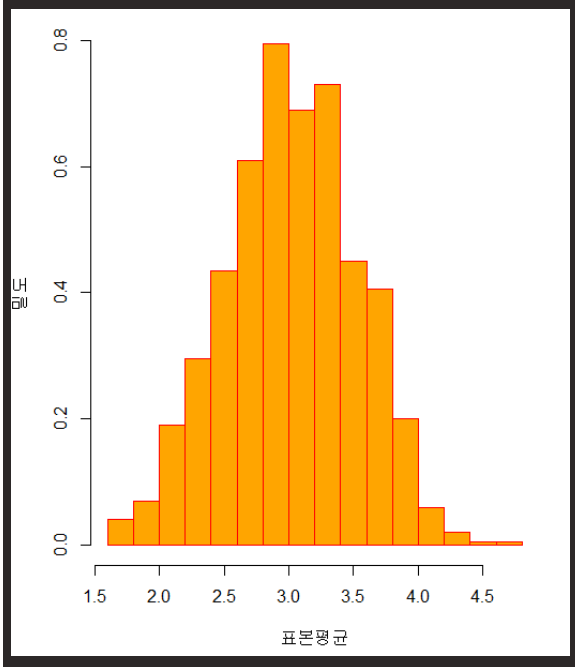


```
> hist(m40, xlim=c(-1.5,1.5), main="", xlab="x", ylab="", col="cyan",
border="blue")
```

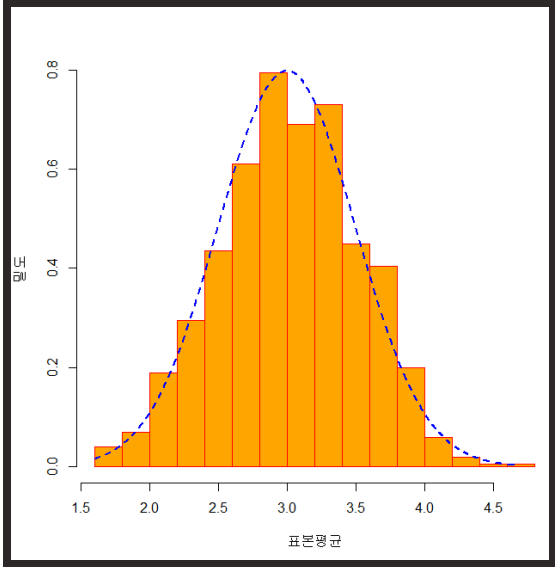


실습 2

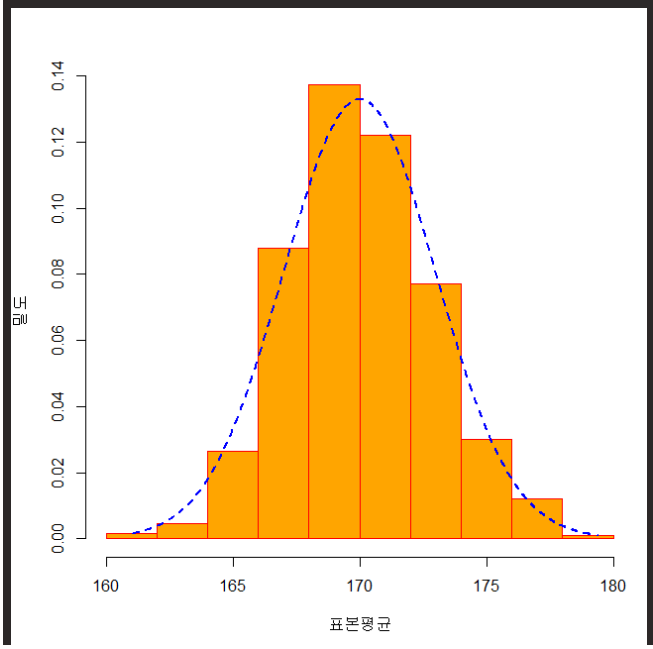
```
> set.seed(9)             # 특정값 쓰면 고정

> n <- 1000

> r.1.mean = rep(NA, n)       # 빈벡터 생성과 동일. NA를 벡터 사이즈만큼 넣어줌.
> r.2.mean = rep(NA, n)

> for (i in 1:n ) {
+ r.1.mean[i] = mean( rnorm(4, mean=3, sd=1) )  # 모집단 평균=3, 표준편차=1
+ r.2.mean[i] = mean( rnorm(4, mean=170, sd=6) ) # 모집단 평균=170, 표준편차=6
+ }

> options(digits=4)
> c(mean(r.1.mean), sd(r.1.mean)) # 표본평균 : 3.0214, 표본표준편차 : 0.5096
[1] 3.0214 0.5096

# 표본분산 s^2 = (0.5096)^2 ,  모집단의 분산 = s^2 * n = (0.5096)^2 * 4 = 1.039
 ~ 모집단의 분산 1과 꽤 근사한 값이다.
(정규분포 모집단과 깊은 관계가 있음을 알아보려는 과정)

> c(mean(r.2.mean), sd(r.2.mean))# 표본평균 : 170.032, 표본표준편차 : 2.835
[1] 170.032   2.835

# s^2 = (2.835)^2, (2.835)^2 * 4 = 32.16  ~ 모집단의 분산 36 -- 조금 차이 남

# s^2 표본분산 = 모집단분산/ n
# 표본표준편차 = sqrt(표본분산/n)


> hist(r.1.mean, prob=TRUE, xlab="표본평균", ylab="밀도", main="",
col="orange", border="red")
# 히스토그램 : prob=TRUE 옵션 사용시 비율에 대한 히스토그램 작성 - 표본의 분포 (cf. 옵
션 없을 시 특정 범위의 빈도수)
```



```
> lines(x1, y1, lty=2, lwd=2, col="blue") # 직선 뿐만 아니라 곡선도 포함

> x1 <- seq(min(r.1.mean), max(r.1.mean), length=1000)
# seq() 사용해서 1000 개 점 나눠줌
> y1 <- dnorm(x=x1, mean=3, sd=(1/sqrt(4)))   # x 를 대입한 정규분포 값 출력
dnorm()

> lines(x1, y1, lty=2, lwd=2, col="blue")    # 정규분포 곡선

# 점선 : 실제 정규분포 곡선 - 표본평균의 분포와 유사함 확인 가능
```

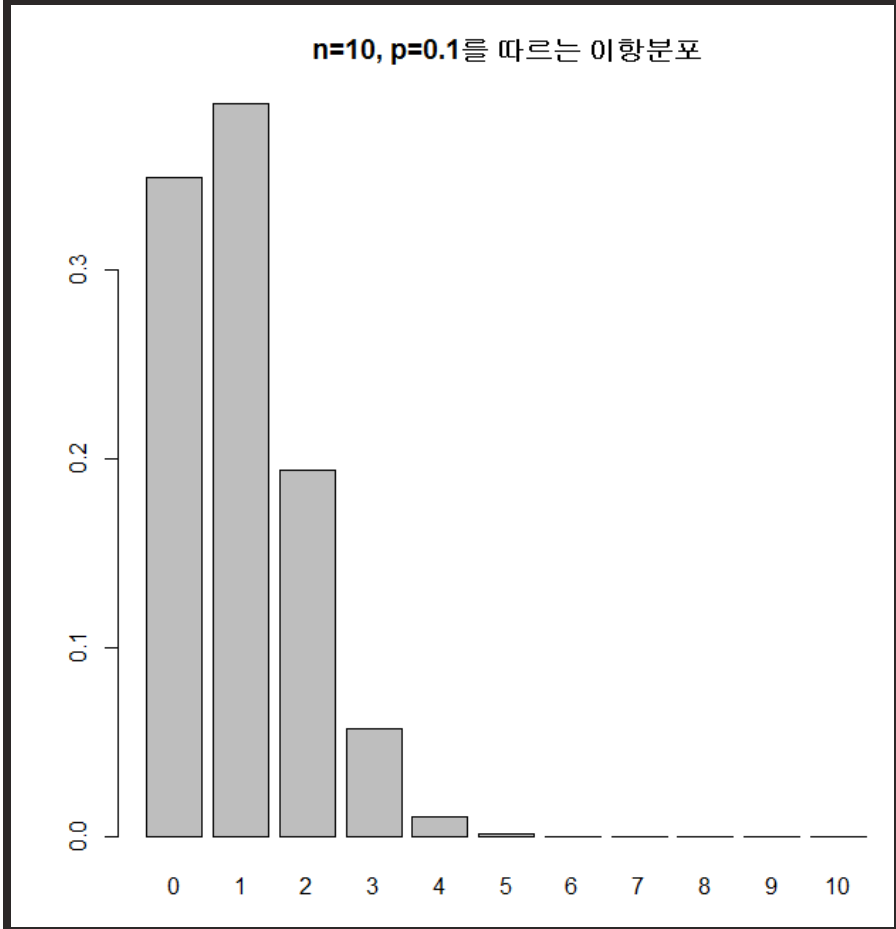



```
> hist(r.2.mean, prob=TRUE, xlab="표본평균", ylab="밀도", main="",
col="orange", border="red")
> x2 <- seq(min(r.2.mean), max(r.2.mean), length=1000)
> y2 <- dnorm( x=x2, mean=170, sd=(6/sqrt(4)) )
> lines(x2, y2, lty=2, lwd=2, col="blue")
```



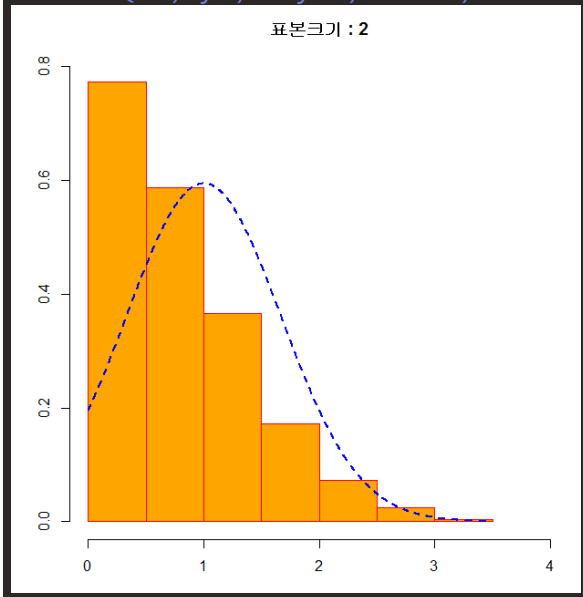
#실습 3

```
# 이항분포 B(n,p) ~ 베르누이시행
> t <- 10; p <- 0.1; x <- 0:10  # t: 시행횟수, p: 성공확률,
> b.p <- dbinom(x, size=t, prob=p)        # dbinom() x에 따른 이항분포 값 출력
> barplot(b.p, names=x, main="n=10, p=0.1를 따르는 이항분포")
```

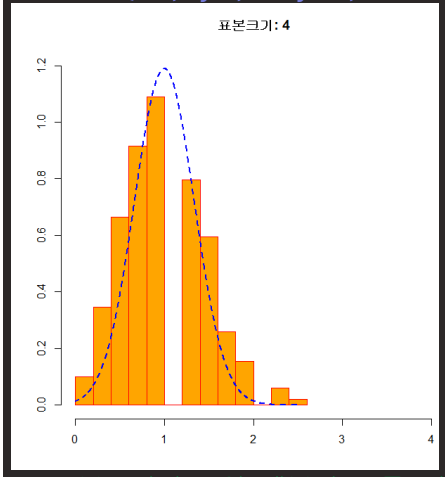



```
# 좌우대칭 x , 꼬리가 긴 형태의 분포

> set.seed(9)
> t <- 10      # 앞에 썼던 t, p와 동일
> p <- 0.1
> x <- 0:10
> n <- 1000

> b.2.mean <- rep(NA, n)
> b.4.mean <- rep(NA, n)
> b.32.mean <- rep(NA, n)


> for(i in 1:n) {
+   b.2.mean[i] <- mean( rbinom(2, size=t, prob=p) )
+   b.4.mean[i] <- mean( rbinom(4, size=t, prob=p) )
+   b.32.mean[i] <- mean( rbinom(32, size=t, prob=p) )
+ }

# 셋 다 모평균 : 1, 모분산 : 0.9, 모표준편차 ~ 0.9487
```

```
> options(digits=4)

> c(mean(b.2.mean), sd(b.2.mean))
 # 모집단이 정규분포가 아님에도 평균이 정규분포처럼 근사함.
[1] 1.0090 0.6763
> c(mean(b.4.mean), sd(b.4.mean))
[1] 1.006 0.481
> c(mean(b.32.mean), sd(b.32.mean))
[1] 0.9989 0.1624


> hist(b.2.mean, prob=T, xlim=c(0, 4), main="표본크기 : 2", ylab="", xlab="",
col="orange", border="red")  # 이항분포를 따르는 집단의 그래프를 확률로
> x1 <- seq(min(b.2.mean), max(b.2.mean), length=1000)
> y1 <- dnorm( x=x1, mean=1, sd=sqrt(0.9)/sqrt(2) )  # 정규분포 값
> lines(x1, y1, lty=2, lwd=2, col="blue")
```

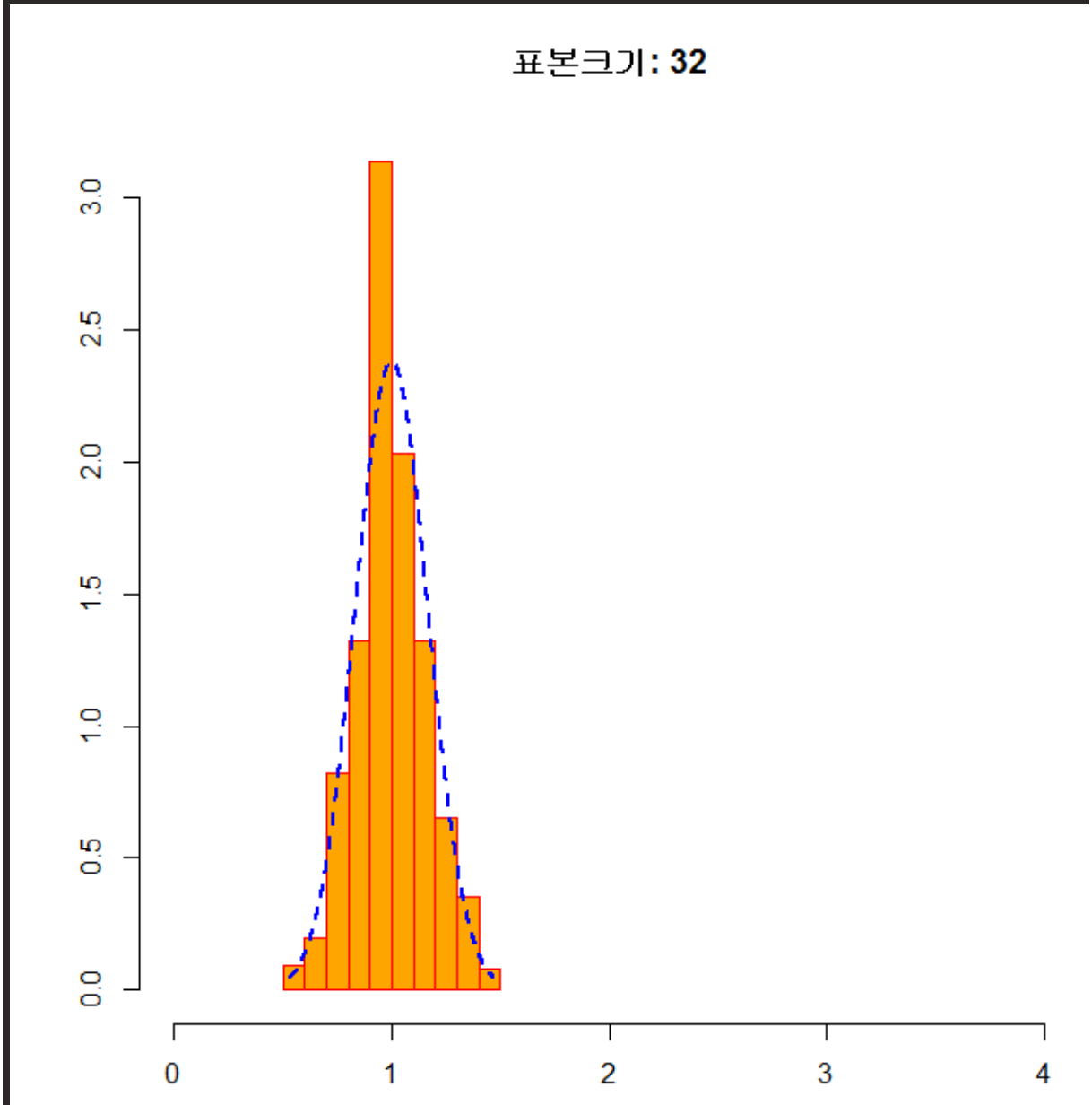



```
# 꽤 차이남..

> hist(b.4.mean, prob=T, xlim=c(0, 4), ylim=c(0, 1.2), main="표본크기: 4",
ylab="", xlab="", col="orange", border="red")
> x2 <- seq(min(b.4.mean), max(b.4.mean), length=1000)
> y2 <- dnorm( x=x2, mean=1, sd=sqrt(0.9)/sqrt(8) )
> lines(x2, y2, lty=2, lwd=2, col="blue")
```




```
 # 표본크기가 2 일 때보다 조금 더 근사해졌지만, 아직 부족...
```

```
> hist(b.32.mean, prob=T, xlim=c(0, 4), main="표본크기: 32", ylab="",
xlab="", col="orange", border="red")
> x3 <- seq(min(b.32.mean), max(b.32.mean), length=1000)
> y3 <- dnorm( x=x3, mean=1, sd=sqrt(0.9)/sqrt(32) )
> lines(x3, y3, lty=2, lwd=2, col="blue")
```




```
# 표본크기가 커질수록 표본들의 분포(이항분포)가 정규분포와 유사함.
```

따라서, 모집단이 정규분포가 아니여도 **표본의 크기가 크면** 표본들의 분포가 정규분포와 근사함
-> 중심극한정리

in 중심극한정리(CLT), 아래의 식 적용 가능
표본평균의 평균 = 모평균 , 표본평균의 분산 = 모분산/샘플사이즈(n)

~ 모집단의 분포와 상관 없음! / 나중에 정규성을 가정으로 하는 경우 by CLT 적용 가능

180607

#점추정 : 표본의 평균이 모집단의 평균일거다라는 추정 방식 (오차가 매우 큼)
#구간추정 : 몇 %의 확률로 모집단의 평균이 특정 구간에 포함되어 있을 것이라는 추정 방식 (표본을 여러 번 samping할 때) (95% 신뢰수준이라고 하면 a값은 0.05)
#구간추정 방식 : 표본이 표준정규분포를 따른다면, P(-1.96<=z<=1.96)을 통해 모집단의 구간 추정 가능

 # 단일모집단의 분포를 알고 있을 경우 표본으로부터 모집단의 평균 추정 ~ 신뢰구간

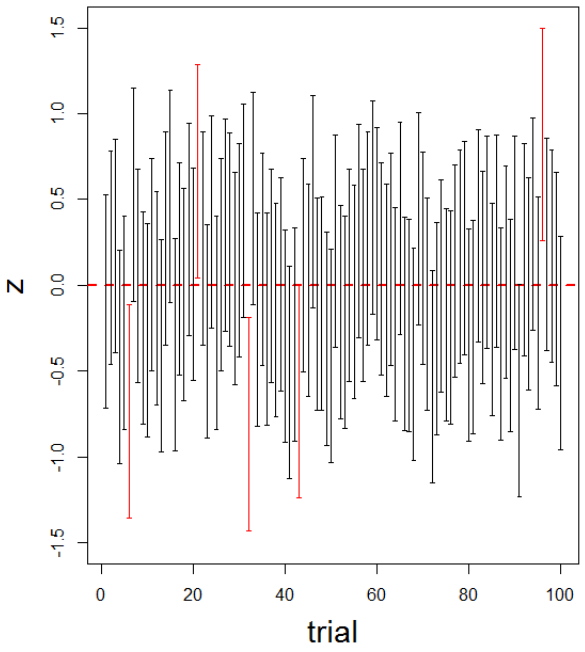
```
> set.seed(9)
> s1<-rnorm(10, mean = 100, sd = 5)

> s1.mean<-mean(s1)
> s1.sd<-sd(s1)
> c(s1.mean, s1.sd)
[1] 98.904907  3.368749

> c(s1.mean - 1.96 * 5 / sqrt(10), s1.mean + 1.96 * 5 / sqrt(10))
[1]  95.80588 102.00394


# qnorm() 사용한 z값이 더 정확함! qnorm이 -이므로 부호 잘 생각해서 쓰기
> c(s1.mean - abs(qnorm(0.025)) * 5 / sqrt(10), s1.mean + abs(qnorm(0.025))
* 5 / sqrt(10))
[1]  95.80593 102.00388



# cf. 99%신뢰구간 ! //// a/2 값인 거 생각하고 쓰기!
> c(s1.mean - abs(qnorm(0.005)) * 5 / sqrt(10), s1.mean + abs(qnorm(0.005))
* 5 / sqrt(10))
[1]  94.83216 102.97765
```

norm, t 잘 분리하기! (t : 원 분포 - 정규분포, n<30)

 # 신뢰구간의 진짜 의미! -> 100번 sampling 해보기 ~ for문, 그래프

+ 모집단 분포 알고 있다고 할 때 시행한 것

```
> set.seed(9)

> n <- 10         #sample size
> x <- 1:100       #100 번 반복
> y <- seq(-3, 3, by=0.01)

> smps <- matrix(rnorm(n * length(x)), ncol=n)  # 100 X 10 형태의 매트릭스 생
성, 각 행(row)이 하나의 sample

> xbar <- apply(smps, 1, mean)    #row 별 연산 (각 sample 의 표본평균 구하기 위해)
> se <- 1 / sqrt(10)              # 표본의 표준편차
> alpha <- 0.05                   # 유의수준 0.05
> z <- qnorm(1 - alpha/2)

> ll <- xbar - z * se          # 각 표본의 신뢰구간의 하한선
> ul <- xbar + z * se          # 각 표본의 신뢰구간의 상한선

> plot(y, xlab="trial", ylab="z", xlim=c(1, 100), ylim=c(-1.5, 1.5),
cex.lab=1.8)

> abline(h=0, col="red", lwd=2, lty=2) #설명선

> l.c <- c()
> l.c <- ifelse(ll * ul > 0, "red", "black")

#모집단의 평균이 신뢰구간에 포함될 때 : 검은색, 그렇지 않으면 빨간색,
  0 포함 안 된 걸 이렇게도 표현하는군!

> arrows(1:length(x), ll, 1:length(x), ul, code=3, angle=90, length=0.02,
col=l.c, lwd=1.5)
#arrows 부터 하면 안됨! plot 부터 그린 다음에 그리기!
```



모집단의 분포 모름 + n<=30 - t통계량 (모표준편차 -> 표본표준편차 , z값 -> not 1.96)

0611

가설검정

키의 평균이 1200mm라는 기존 사실이 현재에도 유지되고 있는지 알아보기

```
> v1<-
c(1196,1340,1232,1184,1295,1247,1201,1182,1192,1287,1159,1160,1243,1264,12
76)
> n<-15         # t통계량, 자유도 n-1
> alpha<-0.05
> xbar<-mean(v1)
> s <- sd(v1)
> #H0 : u = 1220 에 대한 가설검정


> #1. 신뢰구간
> t1<-qt(1-alpha/2, df=n-1)
> c(xbar-t1*s/sqrt(n),xbar+t1*s/sqrt(n))
[1] 1200.526 1260.541


> #2. 통계량
> t_test<-(xbar-1220)/(s/sqrt(n))

> # t_test 값이 +-2.15 사이에 있으므로 채택
```

 # 양쪽검정인지, 단측검정인지 참조해서 alpha/2 , alpha 인지 구분해서 사용하기!!

#3. 유의확률

- 영가설의 타당한 정도를 나타내는 확률 ~ from 검정통계량
- 유의확률이 유의수준 alpha에 비해 크면 영가설 채택, 작으면 영가설 기각
- 1,2번보다 간단

```
> 1-pt(abs(t_test),n-1)
[1] 0.2319981
```

 # 양쪽검정인지, 단측검정인지 참조해서 alpha/2 , alpha 인지 꼭 구분해서 사용하기!!

cf. pt, qt : p(X<=x) ! 따라서

in 우측검정 ; qt(1-alpha, df) / 1-pt(t, df) 형태

좌측검정 ; qt(alpha,df) / pt(t, df) 형태

양측검정 ; 검정통계량이 +,-인지 살펴본 후! 그래도 |t|로 사용한다면

qt(1-alpha/2, df) / 1-pt(|t|, df) 형태로 사용

유의확률 구하기 - 검정통계량 t

양쪽검정 ; P(T > |t|)

단측검정(좌측) ; P(T < t)

단측검정(우측) ; P(T > t)

유의확률과 유의수준 비교 시 주의할 점 !

- 양쪽검정 : 유의확률은 한 쪽의 확률만 구한 것,
  유의확률을 유의수준의 반(alpha/2)과 비교하거나
  유의확률에 두 배를 한 2*유의확률과 유의수준을 비교해야 함.

통계량 먼저 구한 후 그림 그리고,
참조할 함수 / t값 (특히 양쪽 검정일 때 양/음수) 구하고 판단하기 !!

예제2 : 단일모집단의 평균 검정 - 여아 신생아 몸무게의 평균 검정

내용 :

H0 - 여아 신생아의 몸무게 = 2800g

H1 - 여아 신생아의 몸무게 > 2800g

```
> v2<-
c(3837,3334,2208,1745,2576,3208,3746,3523,3430,3480,3116,3428,2184,2383,35
00,3866,3542,3278)
> x_bar<-mean(v2)
> x_sd<-sd(v2)
> n<-18
> alpha<-0.05
 # 1. 신뢰구간
> qt(1-alpha,n-1)
[1] 1.739607
> x_bar-qt(1-alpha,n-1)*x_sd/sqrt(n)
[1] 2873.477                    # 95% 신뢰구간 : (2873.477, 무한대)

> #2. 통계량 검정
> t_test <- (x_bar-2800)/(x_sd/sqrt(n))
> t_test
[1] 2.233188
> qt(1-alpha,n-1)
[1] 1.739607                   # 임계값(1.74) < 통계량(2.23) -- 기각

> #3. 유의확률 검정
> 1-pt(t_test,n-1)
[1] 0.01963422
```

결론

- 여아 신생아의 몸무게 평균이 2800(g)보다 증가하였는지 알아보기 위해,
- 18명의 신생아로부터 측정한 모무게의 평균과 표준편차는 3132.44+-631.583(g)으로 조사되었으며,
- 이로부터 구한 검정통계량은 2.233(유의확률 0.02)으로 나타났습니다.

- 따라서 "여아 신생아의 몸무게의 평균이 2800(g_보다 크다."는 유의수준 0.05 하에서 통계적으로 유의한 결론입니다.
- 이로부터 여아 신생아의 평균 체중은 기존에 알려진 2800(g) 보다 증가한 것으로 여겨집니다.

+ t.test() 함수로 풀어보기

```
> data <-
read.table("http://www.amstat.org/publications/jse/datasets/babyboom.dat.t
xt",header=F)
> str(data)
'data.frame':   44 obs. of  4 variables:
 $ V1: int  5 104 118 155 257 405 407 422 431 708 ...
 $ V2: int  1 1 2 2 2 1 1 2 2 2 ...
 $ V3: int  3837 3334 3554 3838 3625 2208 1745 2846 3166 3520 ...
 $ V4: int  5 64 78 115 177 245 247 262 271 428 ...
> names(data) <- c("time", "gender", "weight", "minutes")

> tmp<-subset(data, gender==1)
> weight <- tmp[,3]      #tmp[[3]]과 같은 뜻으로 쓰임!

> t.test(weight, mu=2800, alternative="greater")
        One Sample t-test

data:  weight
t = 2.2332, df = 17, p-value = 0.01963     #t 통계량, 자유도, 유의확률
alternative hypothesis: true mean is greater than 2800  #대립가설
95 percent confidence interval:
 2873.477      Inf         #95% 신뢰구간
sample estimates:
mean of x
 3132.444
```

t.test (sample, mu, alternative, conf.level)

sample ; 검정할 데이터

mu ; 영가설의 평균값

alternative =

"two.sided" (기본값) 양쪽검정 / **"greater"** (오른쪽) 한쪽검정 / **"less"** (왼쪽) 한쪽검정

#conf.leve = 신뢰수준

0.95 (기본값)

0612

모집단이 두 개인 경우

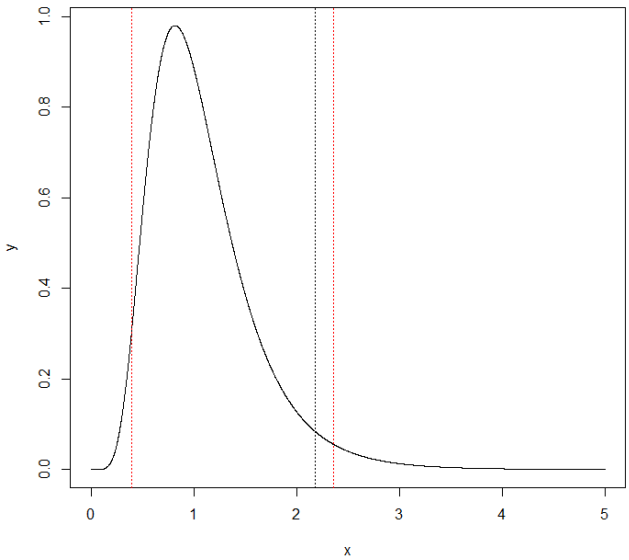
1. 서로 독립인 두 집단 / 2. 대응을 이루는 두 집단

1. 서로 독립인 두 모집단 : 평균 차이 검정

독립검정

- 1. 분산비 비교

```
> #독립검정 - 1. 분산의 비 비교
> data <-
read.table("http://www.amstat.org/publications/jse/datasets/babyboom.dat.t
xt",header=F)
> names(data) <- c("time", "gender", "weight", "minutes")
################ 간단하게 등분산 검정
> var.test(data$weight ~ data$gender)
        F test to compare two variances
data:  data$weight by data$gender
F = 2.1771, num df = 17, denom df = 25, p-value = 0.07526
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.9225552 5.5481739
sample estimates:
ratio of variances
          2.177104
# 95%는 모수에 대한 분산 비율의 신뢰구간!!       not 임계값!

#################### 직접 등분산 검정
> data_f <- data[data$gender==1,3]
> data_m <- data[data$gender==2,3]
> var_f<-sd(data_f)^2
> var_m<-sd(data_m)^2
> n<-length(data_f)
> m<-length(data_m)
> alpha<-0.05
> test_f<-var_f/var_m
> #1. 상한, 하한 임계값 구하기   양측검정 ! ---- alpha/2 로 사용해줘야 함
> qf(alpha/2,n-1,m-1)
[1] 0.3924002
> qf(1-alpha/2,n-1,m-1)
[1] 2.359863
> #cf. 자유도 17, 25 인 F 분포 그래프 그리기
> x<-seq(0,5,by=0.001)
> y<-df(x,n-1,m-1)
> plot(x,y,type='l')
> #위에서 구한 값 그래프에 그리기
> abline(v=test_f,lty=3)
> abline(v=qf(alpha/2,n-1,m-1),lty=3,col=2)
> abline(v=qf(1-alpha/2,n-1,m-1),lty=3,col=2)
# 신뢰구간은 또 다른 식 ! 임계값에 더 추가 수행 있어야 함(모수를 가지고 하는 거니까!)
> #2. 유의확률
> 1-pf(test_f,n-1,m-1)
[1] 0.03763131
> #p-value : 0.0376... 양측검정이므로 바로 비교할 값은 alpha/2 인 0.025.. !
> #해석접근 1. 0.025 보다 p-value 큼 - 귀무가설 채택
> #해석접근 2. (0.0376)*2 가 유의수준 a=0 ㅋ.05 보다 크므로 귀무가설 채택
(var.test 에서 나타나는 p-value 는 *2 값)
```



2. 등분산 검정 O -> T통계량

```
> t.test(data$weight ~ data$gender, mu=0, alternative="less", var.equal=T)
        Two Sample t-test
data:  data$weight by data$gender
t = -1.5229, df = 42, p-value = 0.06764
alternative hypothesis: true difference in means is less than 0
95 percent confidence interval:
     -Inf 25.37242
sample estimates:
mean in group 1 mean in group 2
      3132.444        3375.308
```

#cf. oneway.test에서도 var.equl=T 쓰임 ! 등분산성 가정

직접 해보기

```
> data <-
read.table("http://www.amstat.org/publications/jse/datasets/babyboom.dat.t
xt",header=F)
> names(data) <- c("time", "gender", "weight", "minutes")
> data_f <- data[data$gender==1,3]
> data_m <- data[data$gender==2,3]
> mean_f <- mean(data_f)
> mean_m <- mean(data_m)
> var_f<-sd(data_f)^2
> var_m<-sd(data_m)^2
> n<-length(data_f)
> m<-length(data_m)
> var_fm <-((n-1)*var_f + (m-1)*var_m)/(n+m-2)
> sd_fm <- sqrt(var_fm)
> t_test <- (mean(data_f)-mean(data_m))/(sd_fm * sqrt(1/n + 1/m))
> t_test
[1] -1.522856
> #임계값
> qt(alpha, n+m-2) #상한값!, 하한값은 마이너스 무한대. 기각역에 포함되지 않는다.
[1] -1.681952
> pt(t_test, n+m-2) # 유의확률. 유의수준 0.05 보다 큰 값으로 귀무가설 채택.
[1] 0.06764459
```

2. 대응을 이루는 두 집단 - 대응표본 평균 비교

```
> t.test(data$Prior,data$Post, paired=T, alternative = "less")

        Paired t-test

data:  data$Prior and data$Post
t = -4.1849, df = 16, p-value = 0.0003501
alternative hypothesis: true difference in means is less than 0
95 percent confidence interval:
     -Inf -4.233975
sample estimates:
mean of the differences
            -7.264706
```

```
> #----------직접해보기--------#

> #서로 대응인 두 모집단(같은 집단) - 처리 이전 / 이후 비교
> data <- read.csv("01.anorexia.csv")
> d1 <- data$Prior - data$Post
> n <- length(d1)
> dmean <- mean(d1)
> dsd <- sd(d1)
> alpha=0.05


> #검정통계량
> t_test<-dmean/(dsd/sqrt(n))
> t_test
[1] -4.184908


> #1.임계값
> qt(alpha,n-1) #임계값의 상한값, 마이너스 무한대가 하한값
[1] -1.745884
> # 검정통계량 값이 기각역에 속하므로 귀무가설 기각.


> #유의확률
> pt(t_test,n-1)
[1] 0.0003501266
> #유의확률이 유의수준 0.05 보다 작으므로, 귀무가설 기각.
```

0614

모비율에 대한 추론

[정리] 모비율의 근사적 신뢰구간

: 표본크기 n이 클 때, 모비율 p에 대한 100(1-α)% 근사 신뢰구간

$$\left[\hat{p} \pm z_{1-\alpha/2}\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}\right] \qquad \hat{p} = \frac{X}{n}$$

[정리] 모비율의 검정 (표본이 큰 경우)

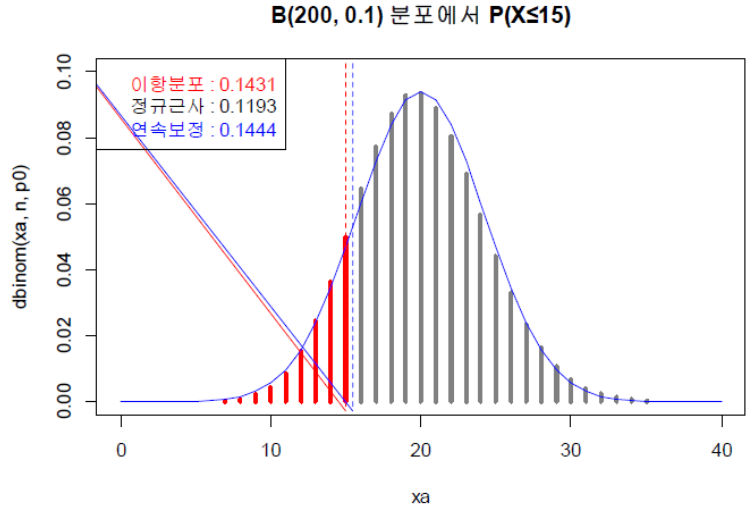
$$H_0 : p = p_0$$

$$Z_0 \equiv \frac{X - np_0}{\sqrt{np_0(1-p_0)}} = \frac{X/n - p_0}{\sqrt{p_0(1-p_0)/n}} \overset{a}{\sim} N(0,1) \,\Big|\, H_0$$

① $H_1 : p > p_0 \Rightarrow$ 기각역: $Z_0 > z_{1-\alpha}$

② $H_1 : p < p_0 \Rightarrow$ 기각역: $Z_0 < z_\alpha = -z_{1-\alpha}$

③ $H_1 : p \neq p_0 \Rightarrow$ 기각역: $|Z_0| > z_{1-\alpha/2}$



n이 클수록 이항분포는 정규분포에 근사함.

독립표본!

->not 대응표본

모비율 차이에 대한 추론
(표본이 큰 경우)

[정리] 모비율 차이의 신뢰구간 (표본이 큰 경우)

: 표본크기가 클 때, 두 모집단의 모비율 차이에 대한 100(1-α)% 근사 신뢰구간

$$\left[(\hat{p}_1 - \hat{p}_2) \pm z_{1-\alpha/2} \sqrt{\frac{\hat{p}_1(1-\hat{p}_1)}{n_1} + \frac{\hat{p}_2(1-\hat{p}_2)}{n_2}} \right]$$

$$X \sim B(n_1, p_1) \quad Y \sim B(n_2, p_2) \qquad \hat{p}_1 = \frac{X}{n_1},\ \hat{p}_2 = \frac{Y}{n_2}$$

$$E(\hat{p}_1 - \hat{p}_2) = p_1 - p_2 \qquad Var(\hat{p}_1 - \hat{p}_2) = \frac{p_1(1-p_1)}{n_1} + \frac{p_2(1-p_2)}{n_2}$$

$$\Rightarrow \frac{(\hat{p}_1 - \hat{p}_2) - (p_1 - p_2)}{\sqrt{p_1(1-p_1)/n_1 + p_2(1-p_2)/n_2}} \overset{a}{\sim} N(0,1)$$

$$\Rightarrow 1 - \alpha \simeq P\left(-z_{1-\alpha/2} < \frac{(\hat{p}_1 - \hat{p}_2) - (p_1 - p_2)}{\sqrt{p_1(1-p_1)/n_1 + p_2(1-p_2)/n_2}} < z_{1-\alpha/2} \right)$$

중간 var 계산 -> not -, but +!

문제 풀어보기

[예 1] 초콜릿 한 개의 무게는 모표준편차 5(g)인 정규분포를 따른다고 한다. 랜덤하게 샘플링한 초콜릿 50개 무게의 표본평균이 199.5(g)였을 때, 95% 신뢰구간을 구하시오.

[예 2] A사 K모델 자동차의 연비는 평균 12.5(km/l), 표준편차 0.5(km/l)로 알려져 있는데, 새로 개발된 엔진을 장착한 40대의 자동차 연비를 측정한 결과 표본평균이 12.64(km/l)로 나왔다. 연비가 개선되었는지 유의수준 5%에서 검정하시오.

[예 3] 프로세스에서 10개의 제품을 랜덤 샘플링하여 검사한 결과 2개의 불량품이 발견되었다. 공정 불량률이 0.1보다 크다고 할 수 있는지 유의수준 10%에서 검정하시오.

[예 4] 두 개의 라인에서 생산한 제품의 불량률을 비교하기 위하여 라인1과 라인2에서 각각 150개와 250개의 표본을 검사한 결과 각각 12개와 10개의 불량품이 발견. 생산라인 1의 불량률이 생산라인 2보다 크다고 할 수 있는지 유의수준 5%에서 검정.

```
> #사용자 정의 함수 만들어서 문제 풀기
#z 분포를 따르는 신뢰구간 구하기 (벡터가 들어오는 경우 가정했음)
> z_conf<-function(x_bar, sig, n=length(x_bar), alpha=0.05){
+   x_bar<-mean(x_bar)
+   ll <- x_bar - qnorm(1-alpha/2)*sig/sqrt(n)
+   hh <- x_bar + qnorm(1-alpha/2)*sig/sqrt(n)
+   cat((1-alpha)*100, "% 신뢰구간 => [", ll, ",", hh, "]")
+ }


# 원래 임계값은 qnorm(alpha/2)로 음의값, 계산적인 편리함 때문에 양수값 쓰고(대칭이기 때문) 마이너스 사용

# cat 함수는 ,로 이어져서 출력 가능한 함수! 편하다!! 깔끔

#t 분포를 따르는 신뢰구간 구하기
> t_conf<-function(x, sig=sd(x), n=length(x), alpha=0.05){
+   mean_x<-mean(x)
+   ll <- mean_x - qt(1-alpha/2,n-1)*sig/sqrt(n)
```

```
+   hh <- mean_x + qt(1-alpha/2,n-1)*sig/sqrt(n)
+   cat((1-alpha)*100, "% 신뢰구간 => [", ll, ",", hh, "]")
+ }


> #z 검정함수 만들어보기
> z_val<-function(mh1, mu, sigma, n, alpha=0.05, both=TRUE){
+   z_test <- (mh1-mu)/(sigma/sqrt(n))
+   p_value <- 1-pnorm(abs(z_test))  #z_test가 0보다 작을 때 다른 결과 나옴
+
+   if (both==TRUE){
+     p_value <- p_value*2
+   }
+   if (p_value > alpha) {
+     cat("h0 : u =",mu, "채택")
+   }
+   else {
+     cat("h0 : u =", mu, "기각")}
+
+ }

> # t 검정 만들기
> t_val<-function(mh1, mu, sd, n, alpha=0.05, both=TRUE){
+   t_test <- (mh1-mu)/(sd/sqrt(n))
+   p_value <- 1-pt(abs(t_test),n-1)
+
+   if (both==TRUE){
+     p_value <- p_value*2
+   }
+   if (p_value > alpha) {
+     cat("h0 : u =",mu, "채택")
+   }
+   else {
+     cat("h0 : u =", mu, "기각")}
+
+ }




> #3. 프로세스에서 10개의 제품을 랜덤 샘플링하여 검사한 결과 2개의 불량품이 발견되었다.
공정 불량률이 0.1보다 크다고 할 수 있는지 유의수준 10%에서 검정하시오

> n=10
> x=2
> pz_test<-(2/10-0.1)/(sqrt(0.1*0.9/10))
> pz_test
[1] 1.054093

> qnorm(0.9)
[1] 1.281552

> 1-pnorm(pz_test)    #유의확률
[1] 0.1459203



> #귀무가설 채택. 공정불량률이 0.1보다 크지 않음.
```

```
> #4.
> n1=150
> x1=12
> p1<-x1/n1
> n2=250
> x2=10
> p2<-x2/n2
> alpha=0.05
> p12<-(x1+x2)/(n1+n2)
> #생산라인 1의 불량률이 생산라인 2보다 큰가?
> #H0 p1=p2 -> p1-p2=0
> #H1 pq-p2>0
> #2 그냥 풀기
> (p1-p2)/sqrt(p1*(1-p1)/n1+p2*(1-p2)/n2)  #1.576
[1] 1.575895

> #2-2 검정통계량 제대로 사용
p0를 (x1+x2)/(n1+n2)로 보기 ! 그리고  *(1/n1+1/n2) 를 sqrt 한 값을 쓰기!
> (p1-p2)/sqrt(p12*(1-p12)*(1/n1+1/n2))  #1.699
[1] 1.698824
> #기각역 1.65

> qnorm(0.95)
[1] 1.644854
```

모집단이 세 개 이상일 경우 평균 비교 검정

ANOVA for 집단간 차이 알아보기

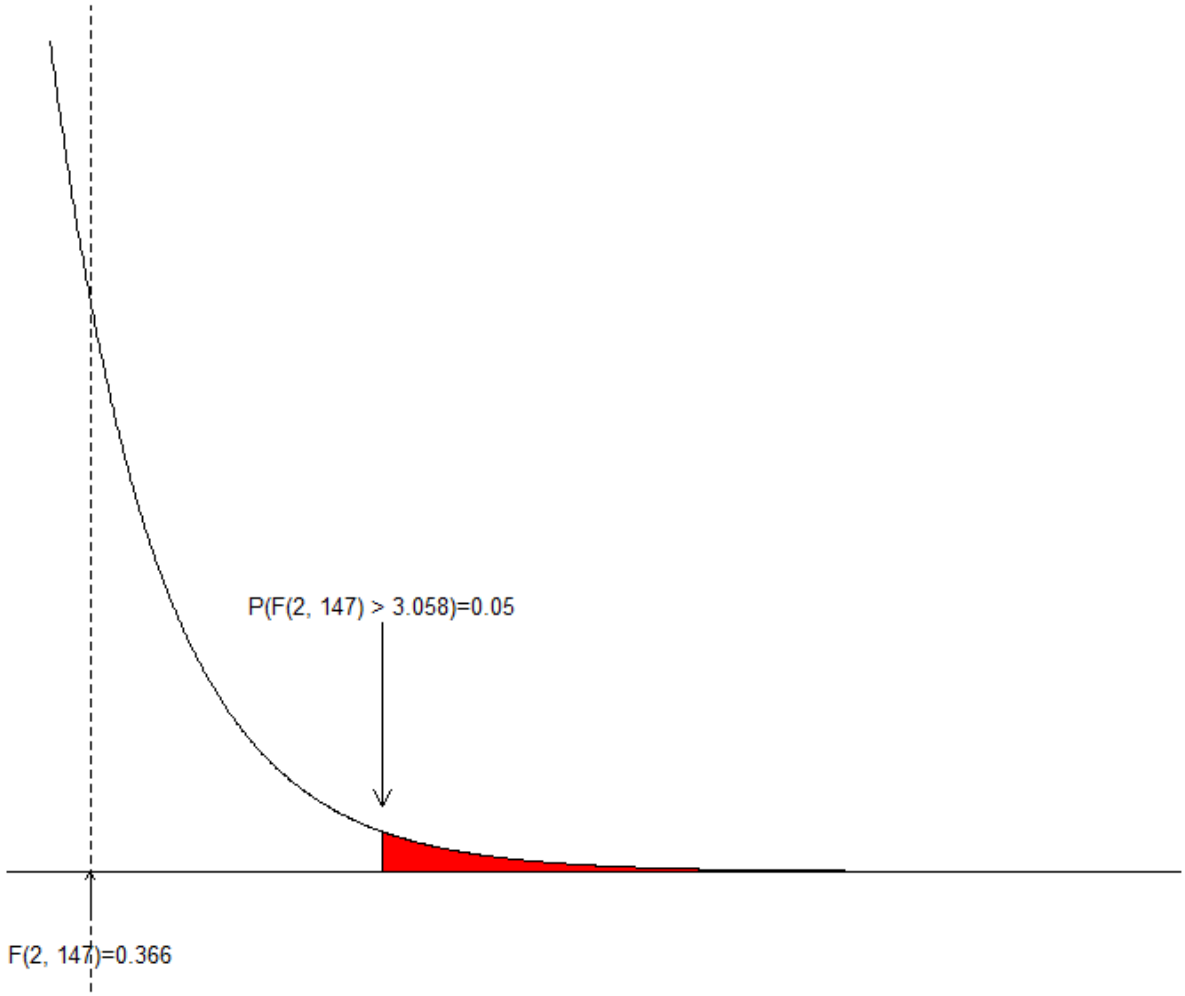
영가설 : 모든 처리의 평균이 같다.

$$\sum(y_{ij} - \bar{y}_{..})^2 =$$

```
ad <- read.csv("age.data.csv", header=T)
ad$scale<-factor(ad$scale)

y1 <- ad$age[ad$scale=="1"]
y2 <- ad$age[ad$scale=="2"]
y3 <- ad$age[ad$scale=="3"]

y1.mean <- mean( y1 )
y2.mean <- mean( y2 )
y3.mean <- mean( y3 )

sse.1 <- sum( (y1 - y1.mean)^2 )
sse.2 <- sum( (y2 - y2.mean)^2 )
sse.3 <- sum( (y3 - y3.mean)^2 )

(sse <- sse.1 + sse.2 + sse.3)
(dfe <- (length(y1)-1) + (length(y2)-1) + (length(y3)-1))

y <- mean(ad$age)

sst.1 <- length(y1) * sum((y1.mean - y)^2)
 #length(y1)을 곱하는 이유 ? -> SSR 시그마 i, j에서 j에 대한 부분 포함 X -> j만큼 곱해주기
sst.2 <- length(y2) * sum((y2.mean - y)^2)
sst.3 <- length(y3) * sum((y3.mean - y)^2)

(sst <- sst.1 + sst.2 + sst.3)
dft <- 2

( tsq <- sum( (ad$age - y)^2 ) )
( ss <- sst + sse )

(mst <- sst / dft)
```

```
mse <- sse / dfe
(f.t <- mst / mse)

alpha <- 0.05
(tol <- qf(1-alpha, 2, 147))

(p.value <- 1 - pf(f.t, 2, 147))
```

```
# 그래프 그리기
x <- seq(0, 10, by=0.01)
yf <- df(x, 2, 147)
par(mar=c(2, 1, 1, 1)) # 하, 좌, 상, 우 (아래부터 시계방향) 여백 지정!!
plot(x, yf, type="l", ylim=c(-0.1, 1), xlab="", ylab="", axes=F)
abline(h=0)
tol.r <- round(tol, 2)
polygon(c(tol.r, x[x>=tol.r], 6), c(0, yf[x>=tol.r], 0), col="red") # 기각역을 색으로 표현,
#x의 범위 지정(from, to, 6-밀집 정도), y축 범위, color 지정
arrows(tol, 0.3, tol, 0.08, length=0.1)
text(tol, 0.32, paste("P(F(2, 147) > ", round(tol, 3),")=0.05", sep=""), cex=0.8)
abline(v=f.t, lty=2)
arrows(f.t, -0.05, f.t, 0, length=0.05)
text(f.t, -0.1, paste("F(2, 147)=", round(f.t, 3),sep=""), cex=0.8)
```



anova 모형

ow <- lm(age~scale, data=ad)

anova(ow) # scale변수가 숫자여서 factor 처리 안 되어있으면 이상한 결과 나옴 ! factor로 바꿔주기

oneway.test(age~scale, data=ad, var.equal=TRUE)

```
> ### H0 채택되었으므로 나이에 따른 효과 X, 그럼 score변수 하기!
> sc<-lm(score~scale, data=ad)
> anova(sc)
Analysis of Variance Table

Response: score
          Df Sum Sq Mean Sq F value Pr(>F)
scale      2     72   36.06  0.1539 0.8575
Residuals 147  34454  234.38
> ##역시 지역에 따른 score 차이도 없당!!!!
```

20180619

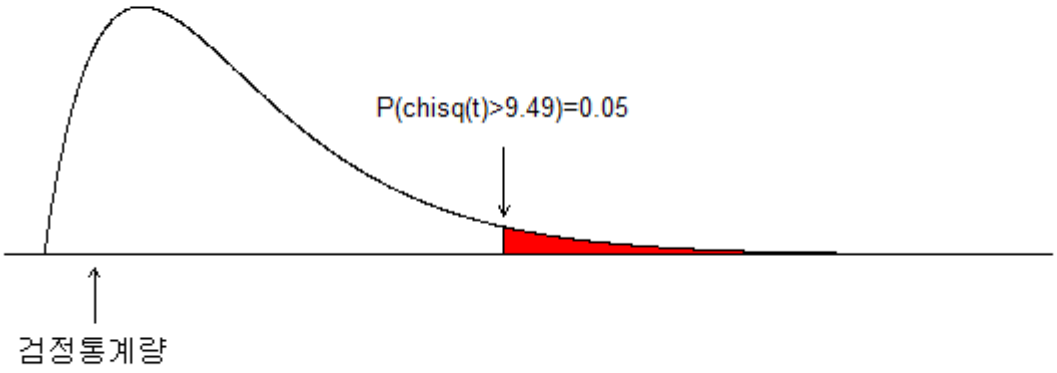
#범주형 자료분석

#범주의 개수 = k

#(관찰도수-기대도수)^2/기대도수의 범주별 총합 ~ chisq(k-1) 분포

#chisq.test(x(범주별 관찰도수 벡터), p(범주별 기대확률 벡터))

```
x=c(32,65,47,38,18)
chisq.test(x,p=c(0.15,0.3,0.25,0.2,0.1)) #p= 꼭 써줘야 함 !

x
e1=200*c(0.15,0.3,0.25,0.2,0.1)
e1

chisq = (x-e1)^2/e1
test_chisq=sum(chisq)

alpha=0.05
k=5

# 임계값
ll=qchisq(1-alpha,k-1)
ll
#유의확률
1-pchisq(test_chisq,k-1)

#그래프 그리기
x=seq(0,20,by=0.01)
y=dchisq(x,k-1)
plot(x,y,type='l',ylim=c(-0.1,0.5), xlab="", ylab="", axes=F)
abline(h=0)
polygon(c(ll, x[x>=ll], 6), c(0, y[x>=ll], 0), col="red")
arrows(ll,y[x=ll] + 0.01 + 0.05 ,ll, y[x=ll] + 0.01,length=0.05)
# arrow - (x,y)시작점 ~ (x,y)끝점 (화살표 표시 있는 부분) 를 잇는 화살표
# 바로 위에 조금만 떼서 표현하고 싶어서 y[x=11] + 0.01 로 표현
arrows(test_chisq,-0.05,test_chisq,-0.01,length=0.05)

text(ll, y[x=ll] + 0.01 + 0.08, paste("P(chisq(t)>",round(ll,2),")=0.05",sep=""),cex=0.8)
# text - x축 위치, y축 위치
text(test_chisq, -0.07, paste("검정통계량",sep=""),cex=0.8)
```


P(chisq(t)>9.49)=0.05
검정통계량

0621

교차분석 : 두 개의 범주형 변수 (factor)간의 연관성을 분석

-동질성 검정 : 하나의 팩터를 고정했을 때, 다른 팩터 간의 분포가 동일한지 분석

-독립성 검정 : 두 변수가 서로 연관성이 있는지 분석

동질성검정

```
> sns.c <- read.csv("snsbyage.csv", header=T, stringsAsFactors=FALSE)
> str( sns.c )
'data.frame':  1439 obs. of  2 variables:
 $ age    : int  1 1 1 1 1 1 1 1 1 1 ...
 $ service: chr  "F" "F" "F" "F" ...
> sns.c <- transform(sns.c, age.c = factor(age, levels=c(1, 2, 3),
labels=c("20 대", "30 대", "40 대")))
> sns.c <- transform(sns.c, service.c = factor(service, levels=c("F", "T",
"K", "C", "E"), ordered=TRUE))
> c.tab <- table(sns.c$age.c, sns.c$service.c)
> c.tab

       F   T   K   C   E
 20 대 207 117 111  81  16
 30 대 107 104 236 109  15
 40 대  78  76 133  32  17
> (a.n <- margin.table(c.tab, margin=1))

20 대 30 대 40 대
 532  571  336
> (s.n <- margin.table(c.tab, margin=2)) # 각 사별 이용률을 알기 위해 필수적인 자
료

  F   T   K   C   E
392 297 480 222  48
> (s.p <- s.n / margin.table(c.tab))   # 각사별/total - 각 사별 이용률 출력(for
기대도수 구하기)

       F        T        K        C        E
0.2724114 0.2063933 0.3335650 0.1542738 0.0333565
> #table!! 1x3, 1x5 형태로 보이지만 3x1, 5x1 형태임
> ## 기대도수
> (expected <- a.n %*% t(s.p))

            F        T       K       C       E
 20 대 144.92286 109.80125 177.4566 82.07366 17.74566
 30 대 155.54691 117.85059 190.4656 88.09034 19.04656
 40 대  91.53023  69.34816 112.0778 51.83600 11.20778
> # %/*% -- 행렬 곱!(inner product) cf. python dot 함수
> #inner product ~ (3x1)*(1x5)=3x5
> #따라서 5x1 인 s.p 를 t()를 통해 꼭 전치시켜줘야 함 !!
> (o.e <- c.tab-expected)  # 편차 : 관찰도수 - 기대도수

             F         T         K         C         E
 20 대  62.077137   7.198749 -66.456567  -1.073662  -1.745657
 30 대 -48.546908 -13.850591  45.534399  20.909659  -4.046560
 40 대 -13.530229   6.651842  20.922168 -19.835997   5.792217
> (t.t <- sum((o.e)^2 / expected ))  # 검정통계량
```

```
[1] 102.752
> qchisq(0.95, df=8)
[1] 15.50731
> 1-pchisq(t.t, df=8)
[1] 0
> chisq.test(c.tab)  # 도수만 있는 교차테이블 넣으면 됨

        Pearson's Chi-squared test

data:  c.tab
X-squared = 102.75, df = 8, p-value < 2.2e-16

> addmargins(chisq.test(c.tab)$expected)

             F        T       K        C       E  Sum
  20대 144.92286 109.80125 177.4566  82.07366 17.74566  532
  30대 155.54691 117.85059 190.4656  88.09034 19.04656  571
  40대  91.53023  69.34816 112.0778  51.83600 11.20778  336
  Sum  392.00000 297.00000 480.0000 222.00000 48.00000 1439
```

연습문제

	A	B	C
1	라인	등급	제품수
2	1	1	20
3	1	2	16
4	1	3	29
5	1	4	21
6	1	5	14
7	2	1	14
8	2	2	22
9	2	3	26
10	2	4	25
11	2	5	13
12	3	1	18
13	3	2	24
14	3	3	32
15	3	4	18
16	3	5	8

#세 개의 라인으로부터 제품을 생산, 제품을 5등급으로 나누어 관리, 각 생산라인에서 100개씩 랜덤샘플링 한 결과 제품의 등급이 생산라인과 상관없이 일정한 분포를 따르는지 유의수준 0.05에서 검정하시오

```
(product <- read.csv("라인별_등급표.csv", header=T, stringsAsFactors=FALSE))

product.c <- transform(product, line.c = factor(라인, levels=c(1, 2, 3), labels=c("line1", "line2", "line3")))
product.c <- transform(product.c, grade.c = factor(등급, levels=c(1,2,3,4,5), labels=c("1등급", "2등급", "3등급", "4등급", "5등급")))

product.c

# 교차 테이블 작성
(product.c.t <- xtabs(제품수~line.c+grade.c, data=product.c))

(line.n <- margin.table(product.c.t, margin=1))
(grade.n <- margin.table(product.c.t, margin=2))

(grade.p <- grade.n / margin.table(product.c.t))
(expected <- line.n %*% t(grade.p))
```

```
(o.e <- product.c.t-expected)  # 편차 : 관찰도수 - 기대도수
(t.t <- sum(  (o.e)^2 / expected ))  # 검정통계량


qchisq(0.95, df=8) # df=(r-1)*(k-1) = 2 * 4

1-pchisq(t.t, df=8)


chisq.test(product.c.t)
```

-참고

```
##########시작 전 기본 다지기#################
> sns.c <- read.csv("snsbyage.csv", header=T, stringsAsFactors=FALSE)
> str( sns.c )
'data.frame':  1439 obs. of  2 variables:
 $ age    : int  1 1 1 1 1 1 1 1 1 1 ...
 $ service: chr  "F" "F" "F" "F" ...

> sns.c <- transform(sns.c, age.c = factor(age, levels=c(1, 2, 3),
+                                          labels=c("20 대", "30 대", "40 대")))
> #transform() for 연산 결과를 새로운 컬럼에 저장(원본 맨 뒤에 있는 함수)
> #age.c 컬럼을 추가 --factor 화한 벡터를
> #label 설정 in factor() -- 실제데이터를 변경하기보다 labeling 과정 (빠름, 원본 X)

> sns.c <- transform(sns.c, service.c = factor(service, levels=c("F", "T",
"K", "C", "E"), ordered=TRUE))
> #levels -- 순서지정
> #ordered=T 순서형 자료로 설정, order 안 하면 알파벳 순으로 자동 정렬됨,
> # 1,2,3,4,5 를 F,T,K,C,E 로 라벨링을 하고 시은데 ordered 를 사용하지 않으면 알파벳
순으로 C-1 로 잘못 팩터화가 될 수 있음


> #범주형 자료에서 위의 과정들 권장

> sns.c
     age service age.c service.c
1      1       F  20 대         F
2      1       F  20 대         F
3      1       F  20 대         F
4      1       F  20 대         F
5      1       F  20 대         F
6      1       F  20 대         F
7      1       F  20 대         F
8      1       F  20 대         F
9      1       F  20 대         F
10     1       F  20 대         F
...
```

```
#factor가 한 개일 때
> age.c.tab <- table(sns.c$age.c)
> #table()은 각 범주별로 몇 개 있는지 빈도수 요약해줌
> #만약 교차표 형태로 존재하면 rowsum()으로 해도 됨
> #아직 데이터프레임형태가 아니어서 table()을 통해 구했음
> str(age.c.tab)
 'table' int [1:3(1d)] 532 571 336
 - attr(*, "dimnames")=List of 1
  ..$ : chr [1:3] "20대" "30대" "40대"

> age.c.tab
20대 30대 40대
 532  571  336
> margin.table(age.c.tab)
[1] 1439
> #margin.table() -- 총합 total count


> addmargins(age.c.tab)
20대 30대 40대  Sum
 532  571  336 1439
> #margin을 구해서 total count를 마지막 합으로 저장

> prop.table(age.c.tab)

     20대      30대      40대
0.3697012 0.3968033 0.2334955
> #각각 margin의 기준으로 비율을 구해줌
> #사마다의 비율을 알 때 필요한 함수 prop.table()
> #------------함수, 단일 factor에 대한 예시 설명이었음---------------#
```

```
> #실전 - 두 개의 factor
> #교차테이블 만드는 과정

> #1. table()함수

> c.tab <- table(sns.c$age.c, sns.c$service.c)
> #table()에 factor를 두 개 쓰면 알아서 교차테이블을 만들어줌
> str(c.tab)
 'table' int [1:3, 1:5] 207 107 78 117 104 76 111 236 133 81 ...
 - attr(*, "dimnames")=List of 2
  ..$ : chr [1:3] "20대" "30대" "40대"
  ..$ : chr [1:5] "F" "T" "K" "C" ...
> c.tab

      F   T   K   C   E
 20대 207 117 111  81  16
 30대 107 104 236 109  15
 40대  78  76 133  32  17
> #먼저 쓴 factor - 행, 뒤에 쓴 factor - 열 -- 각각의 도수를 구해줌 (관찰도수에 대한
테이블)


> #잠깐 함수 살펴보기
> margin.table(c.tab) #총합 total count
[1] 1439
> margin.table(c.tab, margin=1) #행끼리 연산-- 연령별

20대 30대 40대
 532  571  336
> margin.table(c.tab, margin=2) #열끼리 연산-- 회사별

  F   T   K   C   E
392 297 480 222  48
> #table에 총합 넣기
> addmargins(c.tab)    #도수만 가진 테이블의 행/열/전체 합 알아서 count 출력됨
        F   T   K   C   E  Sum
 20대  207 117 111  81  16  532
 30대  107 104 236 109  15  571
 40대   78  76 133  32  17  336
 Sum   392 297 480 222  48 1439


> #참고
> addmargins(c.tab, margin=1) # 행이 추가됨 (앞과 조금 다르다)-- 열끼리 연산 값

      F   T   K   C   E
 20대 207 117 111  81  16
 30대 107 104 236 109  15
 40대  78  76 133  32  17
 Sum  392 297 480 222  48
> addmargins(c.tab, margin=2) # 열이 추가됨  -- 행끼리 연산 값

      F   T   K   C   E Sum
 20대 207 117 111  81  16 532
 30대 107 104 236 109  15 571
 40대  78  76 133  32  17 336
```

```
> apply(c.tab, 1, mean)
 20대  30대  40대
106.4 114.2  67.2
> apply(c.tab, 2, mean)
      F       T       K       C       E
130.6667  99.0000 160.0000  74.0000  16.0000
> prop.table(c.tab) # 전체비율

              F         T         K         C         E
  20대 0.14384990 0.08130646 0.07713690 0.05628909 0.01111883
  30대 0.07435719 0.07227241 0.16400278 0.07574705 0.01042391
  40대 0.05420431 0.05281445 0.09242530 0.02223767 0.01181376
> prop.table(c.tab, margin=1) # 행별비율 ex. 20대 내 회사 비율

              F         T         K         C         E
  20대 0.38909774 0.21992481 0.20864662 0.15225564 0.03007519
  30대 0.18739054 0.18213660 0.41330998 0.19089317 0.02626970
  40대 0.23214286 0.22619048 0.39583333 0.09523810 0.05059524
> prop.table(c.tab, margin=2) # 열별비율 ex. 회사 내 20대 비율

             F        T        K        C        E
  20대 0.5280612 0.3939394 0.2312500 0.3648649 0.3333333
  30대 0.2729592 0.3501684 0.4916667 0.4909910 0.3125000
  40대 0.1989796 0.2558923 0.2770833 0.1441441 0.3541667




> #테이블을 구하는 두 번째 방법
> ## 2. xtabs() 함수
> xt.age <- xtabs(~age.c, data=sns.c)
> #우측 formula에 하나만 쓰면 단일 팩터에 대한 테이블 생성

> str(xt.age)
 'xtabs' int [1:3(1d)] 532 571 336
 - attr(*, "dimnames")=List of 1
  ..$ age.c: chr [1:3] "20대" "30대" "40대"
 - attr(*, "call")= language xtabs(formula = ~age.c, data = sns.c)
> xt.age
age.c
20대 30대 40대
 532  571  336
> xt.sns <- xtabs(~age.c+service.c, data=sns.c)
> #우측 formula에 두 개 작성-> cross table작성
> # formula 작성 방법 ~  행 factor col + 열로 표현하고 싶은 factor col(순서대로) -
- 도수만 출력함
> xt.sns
     service.c
age.c   F   T   K   C   E
  20대 207 117 111  81  16
  30대 107 104 236 109  15
  40대  78  76 133  32  17
```

```
> #cf. 이미 요약된 자료일 때 테이블 구성
> #"xtab.count.csv 는 이미 각 범주형 변수별로 몇 개가 해당하는지 요약된 자료 (앞 자료
는 요약 X)
> #count 변수에 group 별, result 별 개수(도수) 표현되어 있음.
> s.data <- read.csv("xtab.count.csv", header=T)
> xt.s.data <- xtabs(count~group+result, data=s.data)
> #formula 작성 방법-- 도수를 가지는 col ~ 행 factor col + 열 factor col
> xt.s.data
```

0622 독립성 검정

- 결과적으론 정규성 검정과 비슷함
- H0(독립이다) 하에 p11 = n.. * (p1.=n1. / n..) * (p.1 = n.1 / n..)
  = n1. * n.1 / n.. (똑같당!)

-->실전 코드

```
data(UCBAdmissions) # 3차원 함수 (dept가 층인)
(ucba.tab<-apply(UCBAdmissions,c(1,2),sum))
#3차원일 때 apply 함수를 사용하면 층간 각 원소끼리 더할 수 있게 해줌

#행별, 열별 도수의 합
(a.n<-margin.table(ucba.tab, margin=1)) # 행별 합
(g.n<-margin.table(ucba.tab, margin=2)) # 열별 합

(a.p<-a.n/margin.table(ucba.tab)) # 각 비율(합격과 불합격 비율)
(g.p<-g.n/margin.table(ucba.tab)) # 각 비율(남,녀 비율)

(expected <- margin.table(ucba.tab)*(a.p %*% t(g.p))) # 기대도수
addmargins(expected)

## chi-square statistic
chi_test<-sum((ucba.tab-expected)^2/expected) #연속성 수정 X
chi_con_test<-sum((abs(ucba.tab-expected)-0.5)^2/expected)
#연속성 수정 (2*2 일 때 / 이항분포->정규화 과정에서 빠지는 값을을 약 0.5로 봄. 연속성 수정한 검정통계량이 적합함)

qchisq(0.95,1)
1-pchisq(chi_test,1)
1-pchisq(chi_con_test,1)

###############
chisq.test(ucba.tab) # 2*2 테이블의 경우 자동으로 연속성 수정됨
chisq.test(ucba.tab,correct=F) # 연속성 수정 하고 싶지 않을 때
```

->> 카이스퀘어 단점 -> 동일성 / 독립성 구분 불가능 + 행/열을 전치해도 값이 같다.
(설명변수 / 종속변수가 뒤바뀌어도 똑같이 나옴 -> 서로 구분이 어려움)
~~> 우도비검정(상대비율) 등으로 보완

연습문제

```
dre<-read.csv("선호과목_장래희망.csv")
dre.tab<-xtabs(학생수 ~ 선호과목 + 장래희망, data=dre)
#확인용
addmargins(dre.tab)

chisq.test(dre.tab)
# 중학생들의 선호 교과목과 장래희망은 서로 독립이 아니다.


#상세한 각각 값들
g1.n<-margin.table(dre.tab, margin=1) # 교과 합
fu.n<-margin.table(dre.tab, margin=2) # 장래희망 합

g1.p<-g1.n/margin.table(dre.tab)
fu.p<-fu.n/margin.table(dre.tab)

df=12
expected<-margin.table(dre.tab)*(g1.p %*% t(fu.p))
c_test<-sum((dre.tab-expected)^2/expected)

c_test
1-pchisq(c_test,df)
qchisq(0.95,df)
```