R 기초 및 통계분석

도서출판 신아문화사

서 문

Praise the Lord.

2014년 12월 1일 제주연구원장으로 취임하게 되어 학교를 휴직하고 3년을 가르치는 일에서 떠나게 되었는데 마지막 강의 시간에 학생들에게 다음과 같은 약속을 하였다.

1993년 제주대학교에 부임하여 경제통계학을 강의하면서 2시간 이론강의 및 1시간 실습을 지켜왔고, 그 동안 MSTAT, Excel, SAS, Stata 등 다양한 통계패키지를 가르쳐 왔는데 지금은 빅데이터의 시대이고 빅데이터 처리에 최적화되어 있는 R 언어를 공부해야 할 때이다. 따라서 본인이 3년의 임기를 마치고 학교로 복귀하여 경제통계학을 다시 가르치게 되는 2018년 1학기에는 R 언어로 실습을 하겠다고 하였다. 이 약속은 시대의 변화에 따른 것이기도 했지만 나를 채찍질하기 위한 나와의 약속이기도 하였다.

'눈에 보이지 않으면 곧 잊혀진다(out of sight and out of mind)'는 말이 있듯이 3년 동안 연구원 업무에 집중하다 보니 항상 R 언어를 공부해야 한다는 생각은 있었지만 우선순위에서 밀려 공부를 하지 못했다.

2017년 12월 1일 학교로 다시 돌아와 보니 3년 전에 학생들과 했던 약속이 떠올랐고 3월 초에 개강을 앞두고 마음이 급해졌다. 2018년 새해 들어 R 언어를 집중적으로 공부하고 2018년 1학기 강의안을 마련하였다. 다소 부족했지만 2018년 1학기 강의를 마치고, 2학기에 강의안을 업데이트 하였고, 그 결과물이 『R 기초 및 통계분석』으로 발간되게 되었다.

본 교재에서 사용된 데이터는 본인의 개인 홈페이지(http://kanggc.iptime.org/data/)에서 다운받을 수 있고, 실습을 위한 코드는 http://kanggc.iptime.org/code/에서 다운받을 수 있다. R 언어의 유용한 점 중의 하나는 다양한 packages가 있어 이를 활용하면 다양한 분석이 가능하다는 것이다. 본 교재에서 활용되는 packages들을 한 번에 install할 수 있도록 install- packages.R을 만들어 놓았으므로 본 교재로 실습하기 전에 먼저 install-packages.R을 실행하기를 바란다.

지난 두 학기 동안 R 언어를 가르치면서 강조했던 단어가 R 언어의 열성적인 지지자를 뜻하는 R Enthusiast이었다. 학생들에게 그들의 열정을 쏟아 부어도 아깝지 않은 그 무엇을 소개하고 싶었는데 경제통계학과 관련해서는 그것이 바로 R 언어라고 몇 번을 강조하였다.

본 교재가 나오기까지 많은 도움이 있었다. 무엇보다도 2018년도 제주대학교 국립대학육성사업의 지원이 있었기에 본 교재가 출간될 수 있었다. 처음 대하는 R 언어를 포기하지 않고 굳굳하게 따라와 준 학생들에게도 감사를 드린다. 그리고 항상 기도로 격려를 해 주는 사랑하는 아내와 두 딸 셀라와 셀리에게도 감사의 말을 전한다.

2019년 1월

뉴욕에서 저자

목 차

제 1 장

R 기본사용

1. R 개요
2. RStudio 시작하기
3. 명령어 실행방법
4. 수학 및 통계함수

제1장 R 기본사용

1. R 개요

(1) R이란?

컴퓨터로 통계 및 계량분석이 가능하도록 계산 과정을 정리해 놓은 프로그램을 통계 패키지(또는 소프트웨어)라고 하는데 현재 시중에는 SAS(Statistical Analysis System), SPSS(Statistical Package for the Social Sciences), Stata(Statistics Data), WinRats-32(Regression Analysis for Time Series), EViews(Econometric Views), Limdep(Limited Dependent model) 등 다양한 종류의 통계 소프트웨어가 출시되어 활용되고 있다.

한편, 컴퓨터에 명령을 내리는 데 필요한 '컴퓨터의 언어'를 프로그래밍 언어(programming language)라고 하는데 전통적으로 Basic, Cobol, Fortran, C, C++ 등이 활용되어 왔으나 최근에는 GAUSS(Matrix programming language), Matlab, S-plus, R 등의 사용자가 증가하고 있다.

R은 프로그래밍 언어로 구성된 통계분석 도구로 다양한 분석 기능을 가지고 있는 통계 패키지임에도 불구하고 무료로 제공되고 있어 세계적으로 많은 분석가들이 사용하고 있다.

R은 오클랜드대학교의 Robert Gentleman과 Ross Ihaka에 의해 1995년에 처음으로 개발되었고 현재는 R core team 이 R 프로젝트를 운영하고 있다. R은 데이터의 조작(manipulation)과 연산(calculation), 그리고 그래픽 표현(graphical display)을 통합하는 통합 패키지로 금융공학, 생명공학, 행정학, 의학, 자연과학 등 여러 전문분야에서 활용도가 높아지고 있는데 그 이유는 R이 다음과 같은 장점을 가지고 있기 때문이다.

첫째, R은 간단한 명령어만으로 복잡한 계산을 수행할 수 있는 프로그램이기 때문에 분석을 빠르게 수행할 수 있다.

둘째, R은 Linux, UNIX, MAC OS X, Windows 등 모든 운영체제에서 실행 가능하고, 각종 DBMS(Database Management System) 데이터에 접근이 가능하고, 별도의 패키지를 사용하면 R의 소스를 Java, Python, C, C++ 등의 언어와 호환하여 사용할 수 있다.

셋째, R은 공개 소프트웨어로 모든 소스가 공개되므로 자유로운 수정 및 변경이 가능하여 다양하고 정밀한 분석을 할 수 있다.

넷째, 경제학, 행정학, 의학, 생물학 등 다양한 학문 분야에서 사용되는 수많은 통계분석 방법이 패키지 형태로 공개되므로 사용자가 복잡한 계산식을 일일이 입력하여 분석해야 하는 수고를 들 수 있다.

모든 일에 혜택과 비용이 동시에 발생하듯이 R은 이러한 장점을 가지고 있지만 R을 사용하기 위해서는 R 언어를 배워야 하며, 새로운 기능이 빠르게 추가되고 있기 때문에 지속적으로 새로운 기능을 습득해야 하는 어려움이 있다.

(2) RStudio란

소프트웨어 개발 과정에서 필요한 코딩(coding), 디버깅(debugging), 컴파일(compile)의 과정을 하나로 패키지화한 소프트웨어를 통합개발환경(Integrated Development Environment; IDE)이라고 하는데 RStudio는 R의 통합개발환경 소프트웨어로 RStudio를 사용하기 위해서는 반드시 R이 설치되어 있어야 한다.

- 코딩 : 프로그래밍 언어를 이용하여 구체적인 컴퓨터 프로그램을 만드는 기술
- 디버깅 : 코드상의 오류를 찾아내어 수정하는 과정
- 컴파일 : 컴퓨터가 처리한 언어를 사람이 읽을 수 있는 언어나 그림으로 변환하는 프로그램

RStudio는 기존의 R 개발환경에 새로운 기능들이 추가되어 사용자 효용을 높인 유틸리티 소프트웨어로 다음과 같은 장점을 가지고 있다.

첫째, RStudio 역시 모든 운영체제에서 실행이 가능하며, 모든 R 버전과 호환이 가능하다.

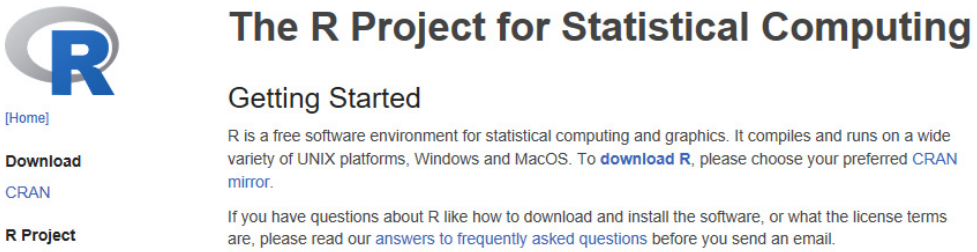
둘째, 코딩작업에 필요한 콘솔(console), 디버깅 작업에 필요한 소스 에디터(source editor), 그리고 데이터 뷰어(data viewer) 및 도표 이력(plot history) 등 통합개발환경의 주요 요소들이 잘 통합되어 편리하고 신속한 작업이 가능하다.

셋째, 표시되는 구문을 종류별로 구분하고(예를 들어 입력문과 출력문, 함수 등) 여러 가지 다른 색으로 강조하여 표시하는 구문 강조(syntax highlight) 기능, 기능과 함수의 첫 글자로 함수를 자동으로 검색하거나 함수에 포함될 요소들을 표시해 주는 코드 완성(code completion), 코드 입력 시 괄호나 따옴표가 자동으로 입력되는 기능 등이 추가되어 수식 입력 과정에서 사용자의 편의를 기하고 있다.

(3) R 및 RStudio 설치

R의 설치파일을 다운로드하기 위해서는 R의 웹페이지(www.r-project.org)에 접속하여 다음과 같은 순서로 진행한다.

첫째, 웹페이지 초기 화면(<그림 1-1>)의 좌측 상단에 있는 CRAN을 클릭한다.



<그림 1-1> R 웹페이지 초기 화면

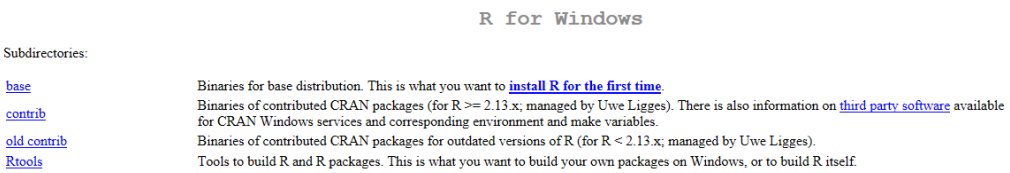
둘째, CRAN 페이지의 국가별 목록에서 대한민국(<그림 1-2>)의 웹페이지 주소 중 하나를 클릭한다.

<그림 1-2> CRAN 페이지 목록(대한민국)

셋째, 운영체제 선택 메뉴(<그림 1-3>)에서 본인의 운영체제에 해당되는 다운로드를 클릭한다.

<그림 1-3> 운영체제 선택 메뉴

넷째, Download R for Windows를 선택하면 세 가지 메뉴(<그림 1-4>)가 나타나는데 base 메뉴를 선택한다,



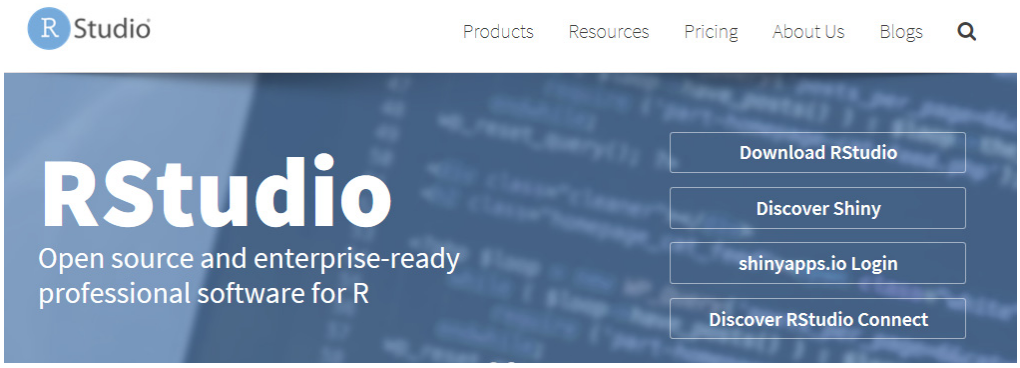
<그림 1-4> 운영 체제별 메뉴

다섯째, Download R 3.4.3 for Windows(<그림 1-5>)를 클릭하여 설치파일을 다운로드하고 설치한다. 단, 설치 시 설치언어 선택은 영문을 권장한다.

<그림 1-5> R 설치프로그램 다운로드

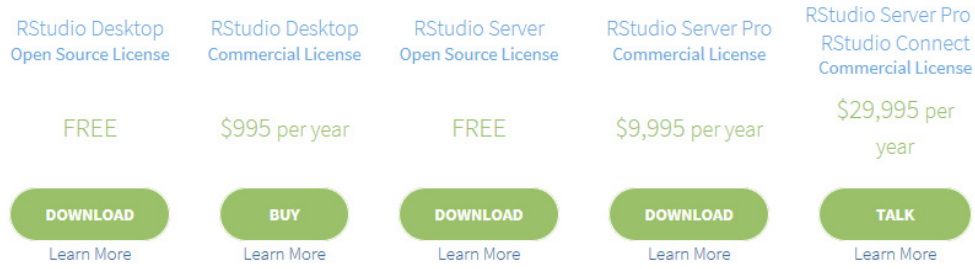
RStudio 설치파일을 다운로드하기 위해서는 RStudio의 웹페이지(www.rstudio.org)에 접속하여 다음과 같은 순서로 진행한다.

첫째, 웹페이지 초기 화면(<그림 1-6>)의 우측 상단에 있는 Download RStudio를 클릭한다.



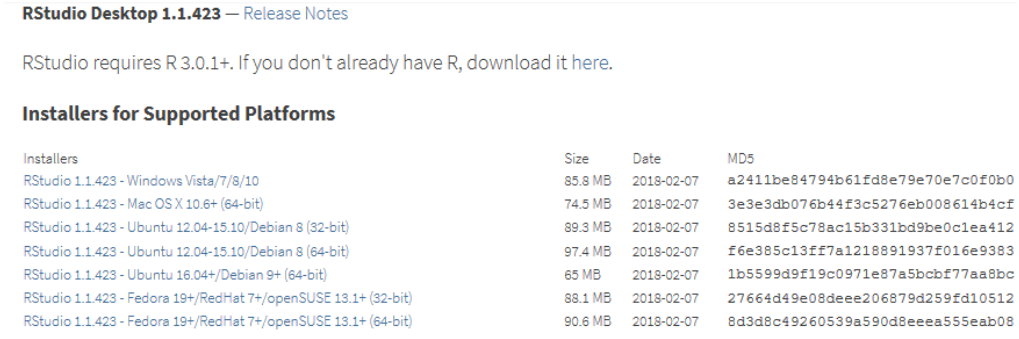
<그림 1-6> RStudio 웹페이지 초기 화면

둘째, RStudio의 다양한 버전(<그림 1-7>) 중 무료인 RStudio Desktop Open Source License의 Download를 클릭한다.



<그림 1-7> RStudio 의 다양한 버전

셋째, 운영체제별로 분류된 RStudio의 설치파일 목록에서 RStudio 1.1.423 - Windows Vista/7/8/10 버전(<그림 1-8>)을 클릭하여 설치파일을 다운로드하고 설치한다.



RStudio Desktop 1.1.423 — Release Notes

RStudio requires R 3.0.1+. If you don't already have R, download it here.

Installers for Supported Platforms

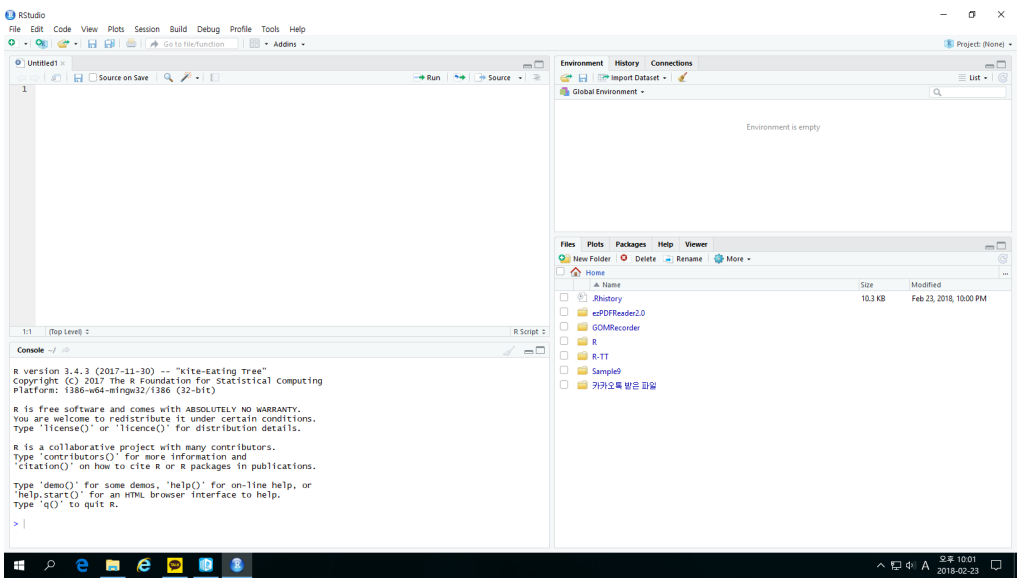
Installers	Size	Date	MD5
RStudio 1.1.423 - Windows Vista/7/8/10	85.8 MB	2018-02-07	a2411be84794b61fd8e79e70e7c0f0b0
RStudio 1.1.423 - Mac OS X 10.6+ (64-bit)	74.5 MB	2018-02-07	3e3e3db076b44f3c5276eb008614b4cf
RStudio 1.1.423 - Ubuntu 12.04-15.10/Debian 8 (32-bit)	89.3 MB	2018-02-07	8515d8f5c78ac15b331bd9be0c1ea412
RStudio 1.1.423 - Ubuntu 12.04-15.10/Debian 8 (64-bit)	97.4 MB	2018-02-07	f6e385c13ff7a1218891937f016e9383
RStudio 1.1.423 - Ubuntu 16.04+/Debian 9+ (64-bit)	65 MB	2018-02-07	1b5599d9f19c0971e87a5bcbf77aa8bc
RStudio 1.1.423 - Fedora 19+/RedHat 7+/openSUSE 13.1+ (32-bit)	88.1 MB	2018-02-07	27664d49e08deee206879d259fd10512
RStudio 1.1.423 - Fedora 19+/RedHat 7+/openSUSE 13.1+ (64-bit)	90.6 MB	2018-02-07	8d3d8c49260539a590d8eeea555eab08

<그림 1-8> RStudio Desktop버전의 OS용 설치파일

2. RStudio 시작하기

RStudio아이콘을 클릭하면 <그림 1-9>와 같이 Sources 창, Console 창, Environment/History 창, Files, Packages/Plots, Help, Viewer 창 등 4개의 창이 나타난다.

- Source 창 : 프로그램 Source를 편집
  프로그램 내의 R 명령어에 커서를 두고 Ctrl-R로 실행
- Console 창 : 명령어를 입력하고 결과를 확인
  상하 화살표를 이용하여 이전 명령어를 편집 및 실행
- Environment/History 창
  - Environment 창 : 변수 또는 객체의 목록과 값 확인
  - History 창 : 명령어 History를 확인 및 검색하고 더블클릭하여 Console 창으로 보냄
- Files, Packages/Plots, Help, Viewer 창
  - Files, Packages 창 : 파일과 폴더 및 패키지 목록
  - Plots, Help, Viewer 창 : 그래프, 도움말, HTML 등 명령어 실행 결과



<그림 1-9> RStudio 4개의 창

3. 명령어 실행방법

R에서 명령어를 실행시키는 방법에는 직접 명령문과 할당 명령문이 있다.

(1) 직접 명령문

Console 창에서 명령어를 직접 입력하여 엔터를 쳐서 실행하거나 print() 함수를 사용하여 실행할 수도 있는데 R을 마치 계산기처럼 사용할 수 있다.

콘솔의 환영 메시지는 Edit>Clear Console(또는 Ctrl+L)을 선택하여 지운 후 (예제 1-1)과 같이 명령어를 입력하여 엔터를 치면 (예제 1-1)의 실행 결과를 보여준다.

(예제 1-1) 직접 명령문 1

```
> 2^3
> 2*3
> 3/3
> 3+3
> 3-3
> q()
```

(예제 1-1)의 실행결과

```
> 2^3
[1] 8
> 2*3
[1] 6
> 3/3
[1] 1
> 3+3
[1] 6
> 3-3
[1] 0
> q()
```

또는 Ctrl+L을 실행하여 console 창의 내용을 지운 후 (예제 1-2)와 같이 명령어를 입력하여 엔터를 치면 (예제 1-2)의 실행 결과를 보여준다.

(예제 1-2) 직접 명령문 2

```
> print(2^.5, digits=5)
> print(2*3, digits=5)
> print(2/3, digits=5)
> print(3+3, digits=5)
> print(3-3, digits=5)
> q()
```

(예제 1-2)의 실행결과

```
> print(2^.5, digits=5)
[1] 1.4142
> print(2*3, digits=5)
[1] 6
> print(2/3, digits=5)
[1] 0.66667
> print(3+3, digits=5)
[1] 6
> print(3-3, digits=5)
[1] 0
```

(2) 할당 명령문

특정한 데이터 또는 연산 결과를 새로운 문자열에 할당하여 하나의 객체를 정의하는 명령문으로 작업 결과의 반환을 요구하지 않는다.

할당 명령문의 형태는 할당 연산자인 <-(또는 ->)를 사용하는 형태와 할당 함수인 assign()을 사용하는 형태가 있는데 모두 동일한 기능을 수행한다.

할당 명령문에 의해 생성된 객체를 제거하려면 rm() 함수를 이용하면 된다.

(예제 1-3)과 같이 명령어를 입력하여 엔터를 치면 (예제 1-3)의 실행 결과를 보여준다.

여기서 x<-c(1,2,3,4,5)는 1부터 5까지 5개의 수치형(numeric) 원소를 결합함수인 c()로 묶어 길이 5인 벡터를 생성한 후 식별문자 'x'에 할당하는 명령문이다.

(예제 1-3) 할당 명령문 1

```
> x<-c(1,2,3,4,5)
> y<-c(1:10)
> z<-x+y
> x
> y
> z
>rm(z)
>z
```

(예제 1-3)의 실행결과

```
> x<-c(1,2,3,4,5)
> y<-c(1:10)
> z<-x+y
> x
[1] 1 2 3 4 5
> y
 [1]  1  2  3  4  5  6  7  8  9 10
> z
 [1]  2  4  6  8 10  7  9 11 13 15
> rm(z)
> z
Error: object 'z' not found
```

한편, (예제 1-4)와 같이 명령어를 입력하여 엔터를 치면 실행 결과를 보여준다.

(예제 1-4) 할당 명령문 2

```
> assign("x", c(1,2,3,4,5))
> assign("y", c(1:10))
> assign("z", x+y)
```

```
> x
> y
> z
> rm(z)
> z
```

(예제 1-4)의 실행결과

```
> assign("x", c(1,2,3,4,5))
> assign("y", c(1:10))
> assign("z",x+y)
> x
[1] 1 2 3 4 5
> y
 [1]  1  2  3  4  5  6  7  8  9 10
> z
 [1]  2  4  6  8 10  7  9 11 13 15
> rm(z)
> z
Error: object 'z' not found
```

(3) 코드 입력 및 실행

Source 창에서 프로그램 Source를 작성, 편집, 저장, 실행, 불러오기 등을 할 수 있다.

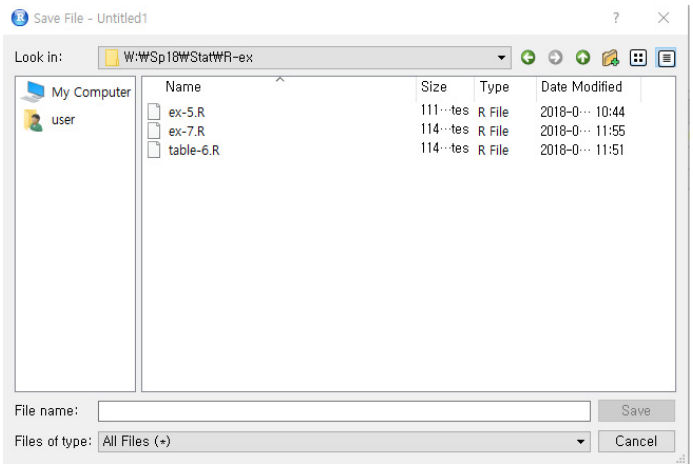
작성된 프로그램을 한 줄씩 실행하는 방법은 Run을 클릭(또는 Ctrl+Enter)하고, 여러 줄 또는 모든 줄을 동시에 실행하는 방법은 여러 줄 또는 모든 줄을 선택하고 Run을 클릭하면 된다.

b1-ch1-5.R과 같이 명령어를 입력하여 모두 선택하고 Run을 클릭하면 다음과 같은 실행 결과를 보여준다.

b1-ch1-5.R의 실행결과

```
> x<-c(1:10)

> x
 [1]  1  2  3  4  5  6  7  8  9 10

> sort(x)
 [1]  1  2  3  4  5  6  7  8  9 10

> sort(x, decreasing=T)
 [1] 10  9  8  7  6  5  4  3  2  1

> mean(x)
[1] 5.5

> median(x)
[1] 5.5

> quantile(x)
   0%   25%   50%   75%  100%
 1.00  3.25  5.50  7.75 10.00

> diff(range(x))
[1] 9

> var(x)
[1] 9.166667

> sd(x)
[1] 3.02765
```

작성된 프로그램 Source를 저장하기 위해서는 File/Save As를 선택한 후 <그림 1-10>과 같이 본인이 원하는 디렉토리에 파일이름(예, b1-ch1-1)을 입력하면 되는데 프로그램의 확장자는 R로 지정이 된다.



<그림 1-10> 프로그램의 저장

저장된 파일을 불러오기 위해서는 File/Open File을 선택한 후 불러 올 파일이 위치한 디렉토리에서 해당 파일을 불러오면 된다.

4. 수학 및 통계함수

(1) 수학함수

R에서는 다양하고 광범위한 내장함수를 제공하고 있어 사용자는 분석 과정에서 빈번하게 사용되는 수식을 단순화한 함수를 사용함으로써 작업의 효율성을 높일 수 있다.

R에서 주로 사용되는 수학함수와 그 기능은 <표 1-1>과 같다.

<표 1-1> 주요 수학함수

함수	기능	함수	기능
sum()	모든 원소의 합	range()	범위 함수
abs()	절댓값 함수	exp()	지수 함수
sqrt()	제곱근 함수	log()	자연로그 함수
max()	최댓값 함수	log10()	상용로그 함수
min()	최솟값 함수	round()	소수점 이하 반올림

b1-ch1-2.R과 같이 수학함수와 관련된 명령어를 입력하여 모두 선택하고 Run을 클릭하면 다음의 실행 결과를 보여준다.

b1-ch1-2.R의 실행결과

```
> a<-c(-3,-2,-1,1,2,3)

> sum(a)
[1] 0

> abs(a)
[1] 3 2 1 1 2 3

> as<-a[4:6]

> sqrt(as)
[1] 1.000000 1.414214 1.732051

> max(a)
[1] 3

> min(a)
[1] -3

> range(a)
[1] -3  3

> exp(a)
[1]      0.04978707      0.13533528      0.36787944      2.71828183      7.38905610
20.08553692

> log(as)
[1] 0.0000000 0.6931472 1.0986123

> log10(as)
[1] 0.0000000 0.3010300 0.4771213
```

(2) 기본 통계함수

기초적인 통계 분석과 관련하여 R에서 주로 사용되는 통계함수와 그 기능은 <표 1-2>와 같다. 기본 통계함수의 사용은 b1-ch1-1.R을 참고하면 된다.

<표 1-2> 기본 통계함수

함수	기능	함수	기능
mean()	산술평균	cor()	상관계수
sort()	오름(내림)차순 정리	cov()	공분산
median()	중앙값	summary()	요약 통계량
quantile()	분위수	cumsum()	누적 합
diff()	원소 사이의 차이	lag()	시차 변수 만들기
var()	분산	sd()	표준편차

b1-ch1-3.R과 같이 기본 통계함수와 관련된 명령어를 입력하여 모두 선택하고 Run을 클릭하면 다음과 같은 실행 결과를 보여준다.

b1-ch1-3.R의 실행결과

```
> x<-c(21,4,13,6,12,7,4,25,22)

> y<-c(-2,4,-3,8,-7,8,-2,-6,5)

> x;y
[1] 21  4 13  6 12  7  4 25 22
[1] -2  4 -3  8 -7  8 -2 -6  5

> cov(x,y)
[1] -19.54167

> cor(x,y)
[1] -0.4123081

> summary(x);summary(y)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   4.00    6.00   12.00   12.67   21.00   25.00
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
-7.0000 -3.0000 -2.0000  0.5556  5.0000  8.0000

> cumsum(1:10);cumprod(1:10)
 [1]  1  3  6 10 15 21 28 36 45 55
 [1]       1       2       6      24     120     720    5040   40320  362880 3628800
```

(3) 확률분포 관련 통계함수

확률분포와 관련된 통계함수와 그 기능은 <표 1-3>과 같다.

<표 1-3> 확률분포 통계함수

분포	R 함수	인수(arguments)
binomial	binom()	size, prob
chi-squared	chisq()	df, ncp
F	f()	df1, df2, ncp
normal	norm()	mean, sd
poison	pois()	lambda
Student's t	t()	df, ncp
uniform	unif()	min, max

우리가 원하는 통계량을 얻기 위해서는 함수의 이름 앞에 <표 1-4>와 같은 접두사를 붙여야 한다.

<표 1-4> 확률분포 접두사

접두사	기능
d	확률밀도함수(PDF)의 확률값, f(x)
p	누적분포함수(CDF)의 확률값, F(x)
q	분위수(quantile) 값, $F^{-1}(x)$
r	무작위 난수 생성

b1-ch1-4.R과 같이 확률분포와 관련된 명령어를 입력하여 모두 선택하고 Run을 클릭하면 다음과 같은 실행 결과를 보여준다.

b1-ch1-4.R의 실행결과

```
> #수익률 평균=40%,표준편차=10%인 정규분포에서 수익률이 60%보다 낮을 확률
> pnorm(60,mean=40,sd=10)
[1] 0.9772499

> #수익률 평균=40%,표준편차=10%인 정규분포에서 수익률이 60%보다 높은 확률(표준화)
> 1-pnorm(2,0,1)
[1] 0.02275013

> #P(Z<1.645)
> pnorm(1.645, 0,1)
[1] 0.9500151

> #P(Z<K)=0.95일 때, K의 값은?
> qnorm(0.95, 0,1)
[1] 1.644854

> #t-통계량이 -3.271, n=16일 때 p의 값은?
> pt(-3.271, 15)
[1] 0.002578269

> #n=16일때, 5% 유의수준에서 기각역(단측)
> qt(p=0.05, df=15)
[1] -1.75305

> round(rnorm(n=20, mean=40, sd=10), digits=2)
 [1] 48.29 51.72 28.27 47.01 44.95 49.13 41.63 36.19 43.55 36.10 30.27 45.50 23.98 41.69
[15] 59.65 56.75 42.91 54.74 43.61 30.84
```

제 2 장

Data set

1. Data set 만들기
2. 데이터관리 기본 명령어

제2장 Data set

1. Data set 만들기

(1) R에서 직접 자료를 입력하기

R에서 숫자나 문자를 직접 입력하여 데이터 파일을 만드는 방법에는 Data Editor 창을 이용하는 방법과 명령어를 이용하는 방법이 있다.

R Source 창에서 다음과 같이 입력하면 Data Editor 창이 뜨는데 <그림 2-1>과 같이 엑셀에서 데이터를 입력하는 방법과 유사하게 데이터를 입력하면 된다.

```
> mydata<-data.frame(age=numeric(), gender=character(), weight=numeric())
> mydata<-edit(mydata)
```

	age	gender	weight
1	25	female	52
2	30	male	65
3	56	male	89
4	61	male	68
5	33	female	55
6	30	male	73
7	85	female	60
8	47	female	49
9	27	male	105
10	70	female	58

<그림 2-1> Data Editor 창 및 데이터 입력

한편, 프로그래밍을 할 경우에는 Data Editor 창을 이용한 데이터 입력방식을 사용할 수 없으므로 이 경우 명령어를 활용할 수 있다. 예를 들어 나이(age), 성별(gender), 몸무게(weight)를 보여주는 데이터는 프로그램 작성 시 다음과 같이 명령어를 입력하면 된다.

```
age<-c(25,30,56,61,33,30,85,47,27,70)
gender<-c("female","male","male","male","female","male","female","female","male","female")
weight<-c(52,65,89,68,55,73,60,49,105,58)
```

(2) 외부에서 작성된 자료를 불러 들여오기

① ASCII-TEXT 파일

외부에서 작성된 ASCII-TEXT 파일을 R로 불러 들여오기 위해서는 b1-ch2-1.R 또는 b1-ch2-2.R을 실행하면 ASCII-TEXT 파일을 불러오고, 이를 행렬로 변경하면 요약통계량을 포함한 다양한 통계 분석이 가능하다.

b1-ch2-1.R의 실행결과

```
> sample1<-"http://kanggc.iptime.org/book/data/sample1.txt"

> sample_dat<-read.delim(sample1,header=T)

> sample_dat
   year       GDP consumption
1  2000  635184.6    413461.2
2  2001  688164.9    460668.2
3  2002  761938.9    515616.0
4  2003  810915.3    535967.4
5  2004  876033.1    562020.2
6  2005  919797.3    602345.4
7  2006  966054.6    643408.0
8  2007 1043257.8    691740.4
9  2008 1104492.2    740804.6
```

```
10  2009  1151707.8      769588.6
11  2010  1265308.0      819821.2
12  2011  1332681.0      873522.7
13  2012  1377456.7      911938.2
14  2013  1429445.4      942267.2
15  2014  1486079.3      972925.0
16  2015  1564123.9     1006005.6
17  2016  1637420.8     1047482.4

> year<-sample_dat$year

> gdp<-sample_dat$GDP

> consumption<-sample_dat$consumption

> min(gdp)
[1] 635184.6

> max(consumption)
[1] 1047482

> mean(gdp)
[1] 1120592

> mean(consumption)
[1] 735857.8

> median(gdp)
[1] 1104492

> median(consumption)
[1] 740804.6

> quantile(gdp)
       0%        25%        50%        75%       100%
 635184.6   876033.1  1104492.2  1377456.7  1637420.8
```

```
> quantile(consumption)
        0%         25%         50%         75%        100%
 413461.2    562020.2    740804.6    911938.2   1047482.4

> var(gdp)
[1] 100191904893

> var(consumption)
[1] 40858333365

> sd(gdp)
[1] 316531

> sd(consumption)
[1] 202134.4

> summary(sample_dat)
      year            GDP          consumption
 Min.   :2000   Min.   : 635185   Min.   : 413461
 1st Qu.:2004   1st Qu.: 876033   1st Qu.: 562020
 Median :2008   Median :1104492   Median : 740805
 Mean   :2008   Mean   :1120592   Mean   : 735858
 3rd Qu.:2012   3rd Qu.:1377457   3rd Qu.: 911938
 Max.   :2016   Max.   :1637421   Max.   :1047482
```

b1-ch2-2.R의 실행결과

```
> sample1<-"http://kanggc.iptime.org/book/data/sample1.txt"

> sample_dat<- as.matrix(read.delim(sample1,header=T),ncol=3)

> sample_dat
      year        GDP  consumption
 [1,] 2000   635184.6     413461.2
 [2,] 2001   688164.9     460668.2
 [3,] 2002   761938.9     515616.0
 [4,] 2003   810915.3     535967.4
```

```
 [5,] 2004  876033.1    562020.2
 [6,] 2005  919797.3    602345.4
 [7,] 2006  966054.6    643408.0
 [8,] 2007 1043257.8    691740.4
 [9,] 2008 1104492.2    740804.6
[10,] 2009 1151707.8    769588.6
[11,] 2010 1265308.0    819821.2
[12,] 2011 1332681.0    873522.7
[13,] 2012 1377456.7    911938.2
[14,] 2013 1429445.4    942267.2
[15,] 2014 1486079.3    972925.0
[16,] 2015 1564123.9   1006005.6
[17,] 2016 1637420.8   1047482.4

> year<-sample_dat[,1]

> year
 [1] 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012
2013 2014 2015
[17] 2016

> gdp<-sample_dat[,2]

> gdp
 [1]  635184.6  688164.9  761938.9  810915.3  876033.1  919797.3  966054.6
1043257.8
 [9] 1104492.2 1151707.8 1265308.0 1332681.0 1377456.7 1429445.4 1486079.3
1564123.9
[17] 1637420.8

> consumption<-sample_dat[,3]

> consumption
 [1]  413461.2  460668.2  515616.0  535967.4  562020.2  602345.4  643408.0
691740.4
 [9]  740804.6  769588.6  819821.2  873522.7  911938.2  942267.2  972925.0
1006005.6
```

```
[17] 1047482.4

> colMeans(sample_dat)
        year          GDP  consumption
      2008.0    1120591.9      735857.8

> summary(sample_dat)
       year          GDP            consumption
 Min.   :2000   Min.   : 635185   Min.   : 413461
 1st Qu.:2004   1st Qu.: 876033   1st Qu.: 562020
 Median :2008   Median :1104492   Median : 740805
 Mean   :2008   Mean   :1120592   Mean   : 735858
 3rd Qu.:2012   3rd Qu.:1377457   3rd Qu.: 911938
 Max.   :2016   Max.   :1637421   Max.   :1047482
```

② CSV 파일

CSV(Comma Separated Value) 파일은 모든 항목을 콤마(,) 단위로 구분하여 저장한 데이터 파일로 엑셀, 워드, 메모장 등 다양한 응용프로그램에서 보기 및 편집이 가능하다는 장점이 있다.

b1-ch2-3.R을 실행하면 CSV 파일을 불러오고, 요약통계량을 포함한 다양한 통계분석이 가능하다.

b1-ch2-3.R의 실행결과

```
> sample1<-("http://kanggc.iptime.org/book/data/csv_sample1.csv")

> sample_dat<-read.csv(sample1,header=T,sep=",")

> sample_dat
  year       gdp  consumption
1 2000  635184.6     413461.2
2 2001  688164.9     460668.2
3 2002  761938.9     515616.0
4 2003  810915.3     535967.4
```

```
5   2004   876033.1      562020.2
6   2005   919797.3      602345.4
7   2006   966054.6      643408.0
8   2007  1043257.8      691740.4
9   2008  1104492.2      740804.6
10  2009  1151707.8      769588.6
11  2010  1265308.0      819821.2
12  2011  1332681.0      873522.7
13  2012  1377456.7      911938.2
14  2013  1429445.4      942267.2
15  2014  1486079.3      972925.0
16  2015  1564123.9     1006005.6
17  2016  1637420.8     1047482.4

> year<-sample_dat$year

> gdp<-sample_dat$GDP

> consumption<-sample_dat$consumption

> summary(sample_dat)
      year           gdp          consumption
 Min.   :2000   Min.   : 635185   Min.   : 413461
 1st Qu.:2004   1st Qu.: 876033   1st Qu.: 562020
 Median :2008   Median :1104492   Median : 740805
 Mean   :2008   Mean   :1120592   Mean   : 735858
 3rd Qu.:2012   3rd Qu.:1377457   3rd Qu.: 911938
 Max.   :2016   Max.   :1637421   Max.   :1047482
```

③ Excel 파일

Excel을 불러오는 방법은 다양한 방법이 있는데 웹상에 있는 xlsx 파일을 불러올 수 있는 방법을 설명하고자 한다. 이를 위해서는 먼저 openxlxs 패키지를 install한 후 library로 불러 와야 하는데 b1-ch2-4.R을 실행하면 Excel 파일을 불러오고, 이를 행렬로 변경하면 요약통계량을 포함한 다양한 통계 분석이 가능하다. 만약에 xls 파일을 위와 같은 방법으로 불러오기 위해서는 xls 파일을 xlsx 파일로 변환시켜 주면 된다.

b1-ch2-4.R의 실행결과

```
> library(openxlsx)
> excel_sample1<-read.xlsx("http://kanggc.iptime.org/book/data/sample1-n.xlsx")
> excel_sample1
   year          gdp  consumption
1  2000   635184.6      413461.2
2  2001   688164.9      460668.2
3  2002   761938.9      515616.0
4  2003   810915.3      535967.4
5  2004   876033.1      562020.2
6  2005   919797.3      602345.4
7  2006   966054.6      643408.0
8  2007  1043257.8      691740.4
9  2008  1104492.2      740804.6
10 2009  1151707.8      769588.6
11 2010  1265308.0      819821.2
12 2011  1332681.0      873522.7
13 2012  1377456.7      911938.2
14 2013  1429445.4      942267.2
15 2014  1486079.3      972925.0
16 2015  1564123.9     1006005.6
17 2016  1637420.8     1047482.4
> excel_sample1_dat<- data.matrix(excel_sample1)
> excel_sample1_dat
   year          gdp  consumption
1  2000   635184.6      413461.2
2  2001   688164.9      460668.2
3  2002   761938.9      515616.0
4  2003   810915.3      535967.4
5  2004   876033.1      562020.2
6  2005   919797.3      602345.4
7  2006   966054.6      643408.0
8  2007  1043257.8      691740.4
9  2008  1104492.2      740804.6
10 2009  1151707.8      769588.6
```

```
11 2010 1265308.0      819821.2
12 2011 1332681.0      873522.7
13 2012 1377456.7      911938.2
14 2013 1429445.4      942267.2
15 2014 1486079.3      972925.0
16 2015 1564123.9     1006005.6
17 2016 1637420.8     1047482.4
> year<-excel_sample1_dat[,1]
> gdp<-excel_sample1_dat[,2]
> consumption<-excel_sample1_dat[,3]
> summary(excel_sample1_dat)
      year            gdp            consumption
 Min.   :2000   Min.   : 635185   Min.   : 413461
 1st Qu.:2004   1st Qu.: 876033   1st Qu.: 562020
 Median :2008   Median :1104492   Median : 740805
 Mean   :2008   Mean   :1120592   Mean   : 735858
 3rd Qu.:2012   3rd Qu.:1377457   3rd Qu.: 911938
 Max.   :2016   Max.   :1637421   Max.   :1047482
```

2. 데이터관리 기본 명령어

(1) 변수의 변환 및 변수명의 변경

기존의 변수를 이용하여 새로운 변수를 만들어 사용할 수 있는데 예를 들어 gdp 및 consumption에 자연로그를 취하여 lgdp 및 lconsumption 변수를 만들 수 있다.

변수명을 변경할 수 있는데 하나의 변수명을 변경하거나(예를 들어, consumption을 cons로) 전체 변수명을 변경할 수 있다.

b1-ch2-5.R을 실행하면 변수명이 바뀐 것을 확인할 수 있다.

b1-ch2-5.R의 실행결과

```
> library(openxlsx)

> excel_sample1<-read.xlsx("http://kanggc.iptime.org/book/data/sample1-n.xlsx")

> excel_sample1_dat<- data.matrix(excel_sample1)

> year<-excel_sample1_dat[,1]

> gdp<-excel_sample1_dat[,2]

> consumption<-excel_sample1_dat[,3]

> lgdp<-log(gdp)

> lconsumption<-log(consumption)

> lgdp; lconsumption
       1        2        3        4        5        6        7        8        9
13.36167 13.44178 13.54362 13.60592 13.68316 13.73191 13.78098 13.85786
13.91490
      10       11       12       13       14       15       16       17
13.95676 14.05083 14.10270 14.13575 14.17280 14.21165 14.26284 14.30863
       1        2        3        4        5        6        7        8        9
12.93232 13.04043 13.15312 13.19183 13.23929 13.30859 13.37453 13.44697
13.51549
      10       11       12       13       14       15       16       17
13.55361 13.61684 13.68029 13.72333 13.75604 13.78806 13.82150 13.86190

> names(excel_sample1)
[1] "year"        "gdp"         "consumption"

> excel_sample1
  year      gdp consumption
1 2000 635184.6    413461.2
2 2001 688164.9    460668.2
3 2002 761938.9    515616.0
```

```
4   2003   810915.3      535967.4
5   2004   876033.1      562020.2
6   2005   919797.3      602345.4
7   2006   966054.6      643408.0
8   2007  1043257.8      691740.4
9   2008  1104492.2      740804.6
10  2009  1151707.8      769588.6
11  2010  1265308.0      819821.2
12  2011  1332681.0      873522.7
13  2012  1377456.7      911938.2
14  2013  1429445.4      942267.2
15  2014  1486079.3      972925.0
16  2015  1564123.9     1006005.6
17  2016  1637420.8     1047482.4

> names(excel_sample1)[3]<-"cons"

> excel_sample1
    year         gdp        cons
1   2000   635184.6   413461.2
2   2001   688164.9   460668.2
3   2002   761938.9   515616.0
4   2003   810915.3   535967.4
5   2004   876033.1   562020.2
6   2005   919797.3   602345.4
7   2006   966054.6   643408.0
8   2007  1043257.8   691740.4
9   2008  1104492.2   740804.6
10  2009  1151707.8   769588.6
11  2010  1265308.0   819821.2
12  2011  1332681.0   873522.7
13  2012  1377456.7   911938.2
14  2013  1429445.4   942267.2
15  2014  1486079.3   972925.0
16  2015  1564123.9  1006005.6
17  2016  1637420.8  1047482.4
```

```
> names(excel_sample1)<-c("T","Y","C")

> excel_sample1
       T           Y           C
1   2000   635184.6   413461.2
2   2001   688164.9   460668.2
3   2002   761938.9   515616.0
4   2003   810915.3   535967.4
5   2004   876033.1   562020.2
6   2005   919797.3   602345.4
7   2006   966054.6   643408.0
8   2007  1043257.8   691740.4
9   2008  1104492.2   740804.6
10  2009  1151707.8   769588.6
11  2010  1265308.0   819821.2
12  2011  1332681.0   873522.7
13  2012  1377456.7   911938.2
14  2013  1429445.4   942267.2
15  2014  1486079.3   972925.0
16  2015  1564123.9  1006005.6
17  2016  1637420.8  1047482.4
```

b1-ch2-6.R은 엑셀 데이터를 불러와서 시계열로 변경하고, 시계열의 시차변수를 만들고, 전년대비 증가율을 구한 후 그림을 그린다. 또한 시계열의 로그를 취한 후 차분 값을 구하면 전년대비 증가율의 근사치를 구할 수 있음을 보여주고 있다.

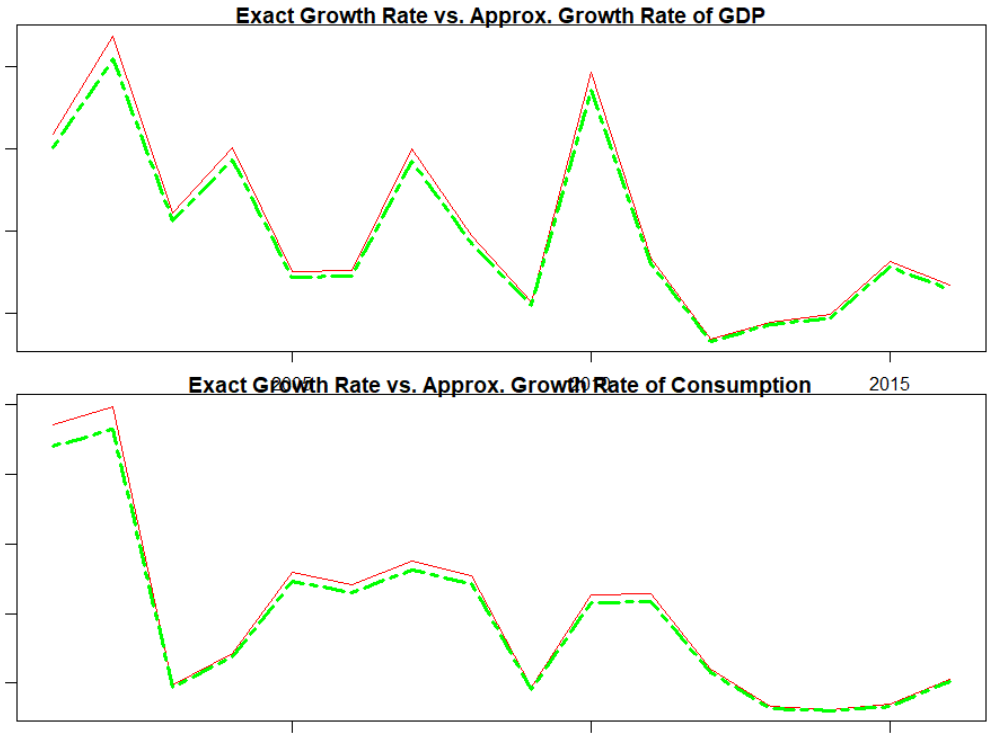
b1-ch2-6.R의 실행결과

```
> library(openxlsx)

> excel_sample1<-read.xlsx("http://kanggc.iptime.org/book/data/sample1-n.xlsx")

> excel_sample1_dat<- data.matrix(excel_sample1)

> year<-excel_sample1_dat[,1]
```

```
> gdp<-excel_sample1_dat[,2]

> consumption<-excel_sample1_dat[,3]

> graphics.off()

> par("mar")
[1] 5.1 4.1 4.1 2.1

> par(mar=c(1,1,1,1))

> y.ts<-ts(gdp, start=2000, end=2016, frequency=1)

> c.ts<-ts(consumption, start=2000, end=2016, frequency=1)

> lagy<-lag(y.ts, k=-1)

> lagc<-lag(c.ts, k=-1)

> gy<-(y.ts-lagy)/lagy

> gc<-(c.ts-lagc)/lagc

> ly.ts<-log(y.ts)

> lc.ts<-log(c.ts)

> gly<-ly.ts-lag(ly.ts, k=-1)

> glc<-lc.ts-lag(lc.ts, k=-1)

> (y<-cbind(gy, gly))
Time Series:
Start = 2001
End = 2016
Frequency = 1
                gy           gly
```

```
2001 0.08340930 0.08011282
2002 0.10720396 0.10183788
2003 0.06427864 0.06229724
2004 0.08030160 0.07724027
2005 0.04995724 0.04874944
2006 0.05029075 0.04906703
2007 0.07991598 0.07688324
2008 0.05869537 0.05703736
2009 0.04274869 0.04186020
2010 0.09863630 0.09406969
2011 0.05324632 0.05187713
2012 0.03359821 0.03304613
2013 0.03774253 0.03704771
2014 0.03961949 0.03885477
2015 0.05251712 0.05118455
2016 0.04686131 0.04579646

> (c<-cbind(gc, glc))
Time Series:
Start = 2001
End = 2016
Frequency = 1
                gc          glc
2001 0.11417516 0.10811437
2002 0.11927847 0.11268426
2003 0.03947007 0.03871104
2004 0.04860893 0.04746445
2005 0.07175045 0.06929324
2006 0.06817119 0.06594801
2007 0.07511936 0.07243169
2008 0.07092863 0.06852615
2009 0.03885505 0.03811919
2010 0.06527202 0.06323018
2011 0.06550392 0.06344785
2012 0.04397768 0.04303811
2013 0.03325774 0.03271666
2014 0.03253621 0.03201811
```

```
2015 0.03400118 0.03343592
2016 0.04122919 0.04040193

> par(mfrow=c(2,1))

> plot(gy, type="l", col="red", main="Exact Growth Rate vs. Approx. Growth Rate of GDP")

> lines(gly, lwd=3, lty=6, col="green")

> plot(gc, type="l", col="red", main="Exact Growth Rate vs. Approx. Growth Rate of Consumption")

> lines(glc, lwd=3, lty=6, col="green")
```



(2) 부분자료 추출

데이터의 일부분을 추출할 수도 있는데 예를 들어, b1-ch2-7.R을 실행하면 2000년대와 2010년대의 자료를 각각 추출할 수 있다.

b1-ch2-7.R의 실행결과

```
> library(openxlsx)

> excel_sample1<-read.xlsx("http://kanggc.iptime.org/book/data/sample1-n.xlsx")

> excel_sample1_dat<- data.matrix(excel_sample1)

> year<-excel_sample1_dat[,1]

> gdp<-excel_sample1_dat[,2]

> consumption<-excel_sample1_dat[,3]

> data1<-excel_sample1_dat[1:10,]

> data1
   year         gdp consumption
1  2000   635184.6      413461.2
2  2001   688164.9      460668.2
3  2002   761938.9      515616.0
4  2003   810915.3      535967.4
5  2004   876033.1      562020.2
6  2005   919797.3      602345.4
7  2006   966054.6      643408.0
8  2007  1043257.8      691740.4
9  2008  1104492.2      740804.6
10 2009  1151707.8      769588.6

> data2<-excel_sample1_dat[11:17,]

> data2
```

	year	gdp	consumption
11	2010	1265308	819821.2
12	2011	1332681	873522.7
13	2012	1377457	911938.2
14	2013	1429445	942267.2
15	2014	1486079	972925.0
16	2015	1564124	1006005.6
17	2016	1637421	1047482.4

제 3 장

기본분석

1. 그림 그리기
2. ggplot2를 이용한 그림 그리기
3. 기술통계량 계산
4. 평균의 계산
5. 두 확률변수의 공분산

제3장 기본분석

1. 그림 그리기

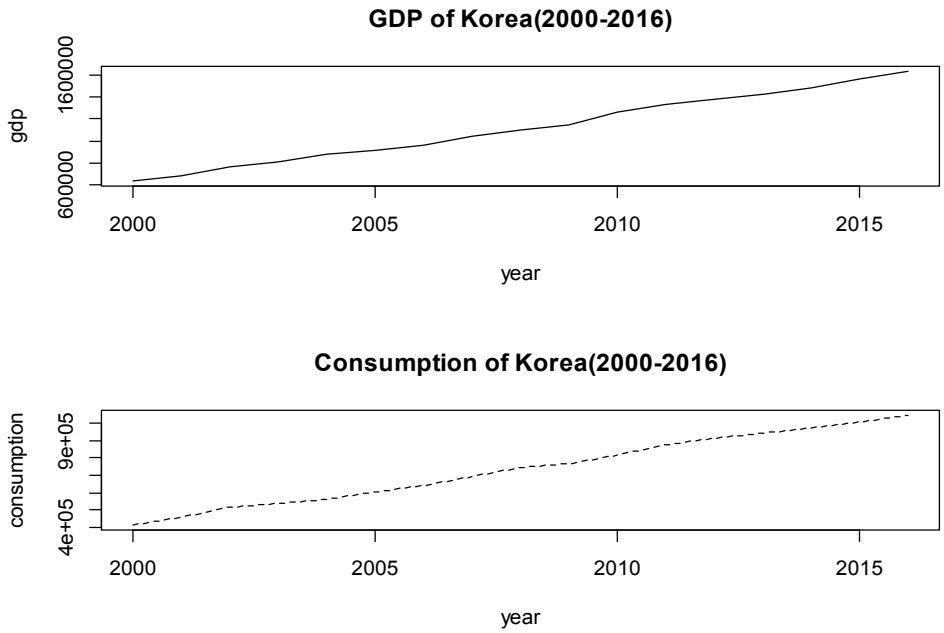
(1) 선 그래프

데이터를 이용하여 다양한 그림을 그릴 수 있는데 예를 들어, b1-ch3-1.R을 실행하면 2000년부터 2016년까지 우리나라의 GDP와 Consumption(소비)에 대한 선 그래프를 그릴 수 있다.

b1-ch3-1.R의 실행결과

```
> csv_sample1<-"http://kanggc.iptime.org/book/data/csv_sample1.csv"

> csv_sample_dat<- as.matrix(read.csv(csv_sample1,header=T),ncol=3)

> year<-csv_sample_dat[,1]

> gdp<-csv_sample_dat[,2]

> consumption<-csv_sample_dat[,3]

> gdp
 [1]  635184.6  688164.9  761938.9  810915.3  876033.1  919797.3
 [7]  966054.6 1043257.8 1104492.2 1151707.8 1265308.0 1332681.0
[13] 1377456.7 1429445.4 1486079.3 1564123.9 1637420.8

> consumption
 [1]  413461.2  460668.2  515616.0  535967.4  562020.2  602345.4
```

```
 [7]  643408.0  691740.4  740804.6  769588.6  819821.2  873522.7
[13]  911938.2  942267.2  972925.0 1006005.6 1047482.4

> par(mfrow=c(2,1)) # 한 페이지에 그림을 위 아래로 나누어 그림

> plot(year, gdp, type="l", main="GDP of Korea(2000-2016)")

> plot(year, consumption, type="l", lty=2,main="Consumption of
Korea(2000-2016)")
```

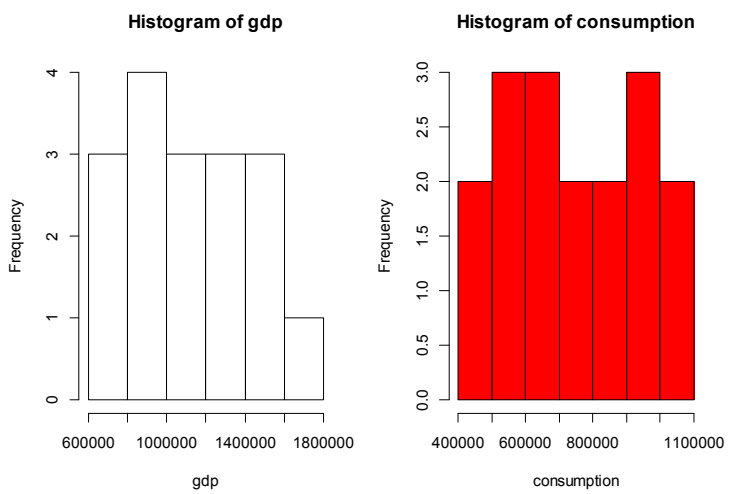


(2) 히스토그램

b1-ch3-2.R을 실행하면 2000년부터 2016년까지 우리나라의 GDP와 Consumption(소비)에 대한 히스토그램을 그릴 수 있다.

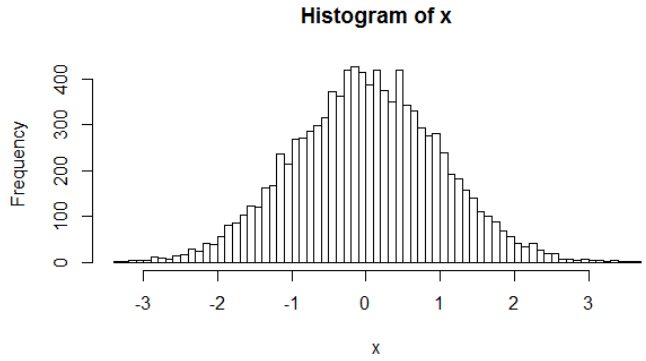
b1-ch3-2.R의 실행결과

```
> csv_sample1<-"http://kanggc.iptime.org/book/data/csv_sample1.csv"

> csv_sample_dat<- as.matrix(read.csv(csv_sample1,header=T),ncol=3)

> year<-csv_sample_dat[,1]

> gdp<-csv_sample_dat[,2]

> consumption<-csv_sample_dat[,3]

> gdp
 [1]  635184.6  688164.9  761938.9  810915.3  876033.1  919797.3
 [7]  966054.6 1043257.8 1104492.2 1151707.8 1265308.0 1332681.0
[13] 1377456.7 1429445.4 1486079.3 1564123.9 1637420.8

> consumption
 [1]  413461.2  460668.2  515616.0  535967.4  562020.2  602345.4
 [7]  643408.0  691740.4  740804.6  769588.6  819821.2  873522.7
[13]  911938.2  942267.2  972925.0 1006005.6 1047482.4

> par(mfrow=c(1,2))  # 한 페이지에 그림을 좌우로 나누어 그림

> hist(gdp)

> hist(consumption, breaks=8, col="red") # 구간의 수를 8개로 함
```



표준정규분포로부터 10,000개의 자료를 만든 후 히스토그램을 그리기 위해서 다음의 명령을 입력하면 된다.

```
>set.seed(1234) # 임의의 정수(예를 들어 1이나 123이나 300이나 12345)를 부여하여 재현 가능한 무작위 번호를 얻을 수 있는 함수
>x<-rnorm(10000) # 표준정규분포로부터 10,000개의 자료를 생성
>hist(x, breaks=100) # 구간의 수를 100으로 하는 히스토그램을 그림
```



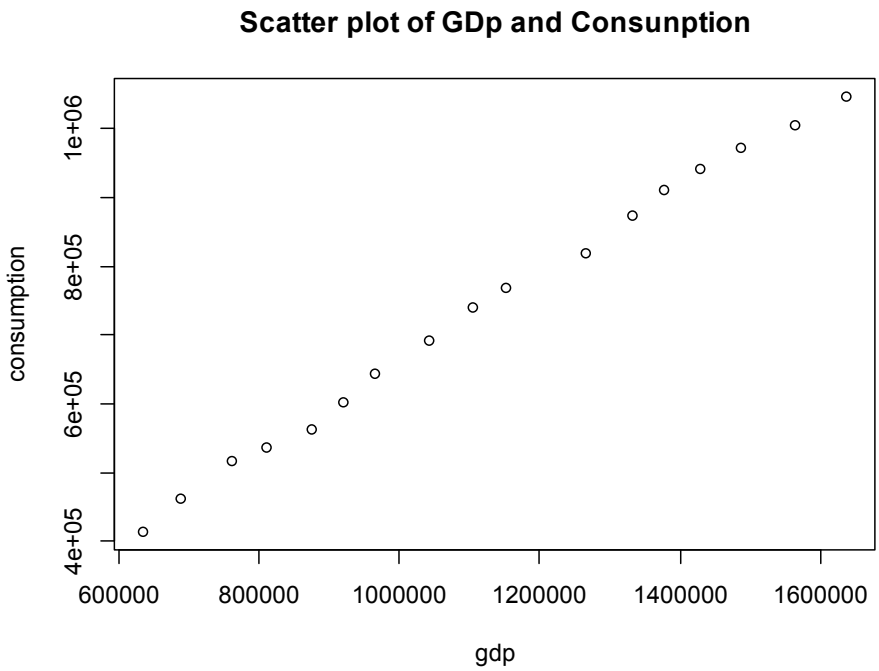
(3) 산포도

b1-ch3-3.R을 실행하면 2000년부터 2016년까지 우리나라의 GDP와 Consumption(소비)에 대한 산포도(scatter plot)를 그릴 수 있다.

b1-ch3-3.R의 실행결과

```
> csv_sample1<-"http://kanggc.iptime.org/book/data/csv_sample1.csv"

> csv_sample_dat<- as.matrix(read.csv(csv_sample1,header=T),ncol=3)

> year<-csv_sample_dat[,1]

> gdp<-csv_sample_dat[,2]
```

```
> consumption<-csv_sample_dat[,3]

> gdp
 [1]   635184.6   688164.9   761938.9   810915.3   876033.1   919797.3
 [7]   966054.6  1043257.8  1104492.2  1151707.8  1265308.0  1332681.0
[13]  1377456.7  1429445.4  1486079.3  1564123.9  1637420.8

> consumption
 [1]   413461.2   460668.2   515616.0   535967.4   562020.2   602345.4
 [7]   643408.0   691740.4   740804.6   769588.6   819821.2   873522.7
[13]   911938.2   942267.2   972925.0  1006005.6  1047482.4

> plot(gdp, consumption, main = "Scatter plot of GDp and Consunption")
```



(4) 상자그래프

상자그래프(box plot)는 최솟값, 아래사분위수(Q_1), 중위수, 위사분위수(Q_3), 최댓값 등 5개 순서통계량을 이용하여 자료를 요약·정리하는 그래프 표현방법으로 두 개 이상의 집단을 상대적으로 비교하기 쉬운 장점이 있다.

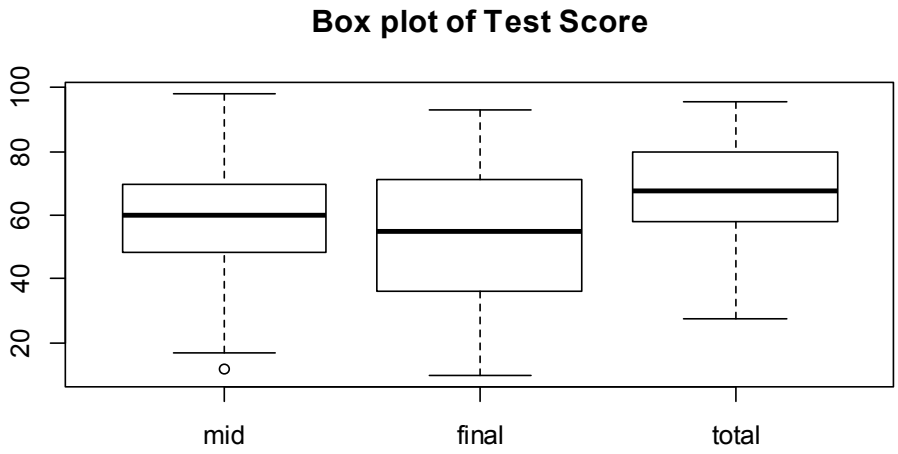
상자그래프는 Q_1과 Q_3를 연결하는 상자를 그리고, 그 상자 안에 중위수를 나타내는 선을 그리며, 최솟값과 Q_1, 그리고 Q_3와 최댓값을 선으로 연결하는 표현방법으로 5개의 순서통계량의 위치를 관찰하여 자료 분포의 특징을 알 수 있다.

b1-ch3-4.R을 실행하면 중간고사(mid), 기말고사(final), 총점(total)에 대한 상자그래프를 그릴 수 있다.

b1-ch3-4.R의 실행결과

```
> library(openxlsx)

>  sample1<-read.xlsx("http://kanggc.iptime.org/book/data/stat-1.xlsx",  sheet=1,
startRow=1, colNames = T)

> mid<-sample1$mid

> final<-sample1$final

> total<-sample1$total

> grade<-sample1$grade

> summary(sample1)
      mid              final            total            grade
 Min.   :12.00   Min.   :10.00   Min.   :27.70   Min.   :1.000
 1st Qu.:48.75   1st Qu.:36.00   1st Qu.:58.17   1st Qu.:2.750
 Median :60.00   Median :55.00   Median :67.75   Median :4.000
 Mean   :59.58   Mean   :52.08   Mean   :67.04   Mean   :3.783
 3rd Qu.:69.25   3rd Qu.:70.50   3rd Qu.:79.45   3rd Qu.:5.000
 Max.   :98.00   Max.   :93.00   Max.   :95.45   Max.   :8.000
```

```
> par(mfrow=c(1,3)) # 한 페이지에 그림을 좌우로 3개 나누어 그림
> boxplot(mid, main="Box plot of mid")
> boxplot(final, main="Box plot of final")
> boxplot(total, main="Box plot of total")
```



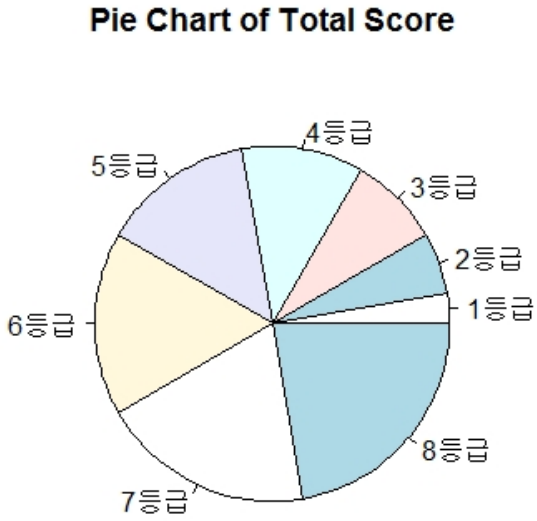
(5) 원그래프(Pie Chart)

원그래프(pie chart)는 전체 자료가 여러 개의 카테고리로 분류되어지는 경우 개개의 카테고리를 전체에 대한 비율에 따라 원 내부의 면적을 분할한 파이(pie)모양의 도표이다. 즉, 각 범주의 관측도수의 상대적인 크기를 원을 분할한 형태로 표현하는 방법이다.

b1-ch3-5.R을 실행하면 총점(total)을 8개의 등급으로 나누어 원그래프를 그릴 수 있다.

b1-ch3-5.R의 실행결과

```
> library(openxlsx)

>  sample1<-read.xlsx("http://kanggc.iptime.org/book/data/stat-1.xlsx",  sheet=1, startRow=1, colNames = T)

> slices<-c(1,2,3,4,5,6,7,8)

> lbls<-c("1등급","2등급","3등급","4등급","5등급","6등급","7등급","8등급")

> pie(slices, labels=lbls, main="Pie Chart of Total Score")>
```



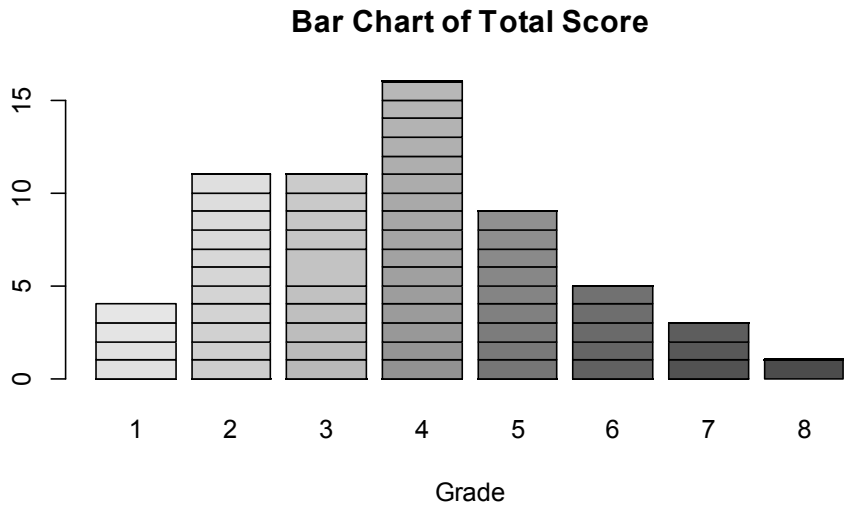
(6) 막대그래프(Bar Chart)

도수분포표는 숫자로 관측된 양적 자료(연속형 자료)를 일정한 구간으로 나눈 후에 각 구간에 속한 개수들의 수를 도수로 나타낸 표이다. 도수분포표에서 각 구간의 관측도수를 막대 형태로 표현하여 그 크기를 비교할 수 있도록 하는 자료의 요약방법이 막대그래프이다.

b1-ch3-6.R을 실행하면 총점(total)을 8개의 구간으로 나누어 막대그래프를 그릴 수 있다.

b1-ch3-6.R의 실행결과

```
> library(openxlsx)

>  sample1<-read.xlsx("http://kanggc.iptime.org/book/data/stat-1.xlsx",  sheet=1, startRow=1, colNames = T)

> mid<-sample1$mid

> final<-sample1$final

> total<-sample1$total

> grade<-sample1$grade

> total
 [1] 95.45 93.00 91.95 90.20 87.45 85.30 84.35 84.30 83.95 83.90 82.85
[12] 82.15 81.45 81.15 80.80 79.00 78.00 77.25 75.85 75.50 75.50 75.15
[23] 74.15 74.10 71.65 71.10 69.90 68.25 68.15 67.95 67.55 67.45 66.75
[34] 64.80 64.65 63.40 63.35 63.30 62.55 62.30 61.15 60.60 59.80 59.65
[45] 58.35 57.65 56.95 55.25 53.20 51.40 51.10 48.95 48.45 47.30 47.00
[56] 40.50 39.65 31.30 30.45 27.70

> grade
 [1] 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3 3 3 3 4 4 4 4 4 4 4 4
[35] 4 4 4 4 4 4 4 4 5 5 5 5 5 5 5 5 5 6 6 6 6 6 7 7 7 8

> counts<-table(total, grade)

> barplot(counts, main="Bar Chart of Total Score", xlab="Grade")
```



(7) 잎-줄기 그래프(Stem and Leaf Plot)

숫자 단위(1단위, 10단위, 100단위,…)를 이용하여 숫자를 두 부분으로 나누어 앞부분은 줄기로, 그리고 뒷부분은 잎으로 하여 자료를 요약·표현하는 방법이 잎-줄기 그래프이다.

잎-줄기 그래프를 그리는 방법은 먼저 숫자를 작은 것부터 크기 순서에 따라 열(columnize)로 나열하고, 관측 값을 그 숫자가 속한 위치의 줄기에 맞추어 잎 부분을 기록하는데 줄기 내의 잎의 값들은 작은 것부터 크기 순서로 정리한다.

이 때 각 줄기에 너무 많은 관측 값이 주어지면 각 줄기에 두 줄을 할당하여 첫 줄에는 잎의 0, 1, 2, 3, 4를 기록하고, 둘째 줄에는 잎의 5, 6, 7, 8, 9를 기록할 수 있다.

b1-ch3-7.R을 실행하면 총점(total)으로 잎-줄기 그래프를 그릴 수 있다.

b1-ch3-7.R의 실행결과
> library(openxlsx)
> sample1<-read.xlsx("http://kanggc.iptime.org/book/data/stat-1.xlsx", sheet=1,

```
startRow=1, colNames = T)

> mid<-sample1$mid

> final<-sample1$final

> total<-sample1$total

> grade<-sample1$grade

> total
 [1] 95.45 93.00 91.95 90.20 87.45 85.30 84.35 84.30 83.95 83.90 82.85
[12] 82.15 81.45 81.15 80.80 79.00 78.00 77.25 75.85 75.50 75.50 75.15
[23] 74.15 74.10 71.65 71.10 69.90 68.25 68.15 67.95 67.55 67.45 66.75
[34] 64.80 64.65 63.40 63.35 63.30 62.55 62.30 61.15 60.60 59.80 59.65
[45] 58.35 57.65 56.95 55.25 53.20 51.40 51.10 48.95 48.45 47.30 47.00
[56] 40.50 39.65 31.30 30.45 27.70

> stem(total)

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

  2 | 8
  3 | 01
  4 | 017789
  5 | 1135788
  6 | 00112333355778888
  7 | 012445666789
  8 | 11123444457
  9 | 0235
>
> stem(total, scale=0.5) # scale=0.5을 통해 줄기의 수를 조정할 수 있음

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

  2 | 801
  4 | 0177891135788
```

```
  6 | 00112333355778888012445666789
  8 | 111234444570235

>
> stem(total, scale=2) # scale=0.5을 통해 줄기의 수를 조정할 수 있음

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

  2 | 8
  3 | 01
  3 |
  4 | 01
  4 | 7789
  5 | 113
  5 | 5788
  6 | 001123333
  6 | 55778888
  7 | 01244
  7 | 5666789
  8 | 111234444
  8 | 57
  9 | 023
  9 | 5
```

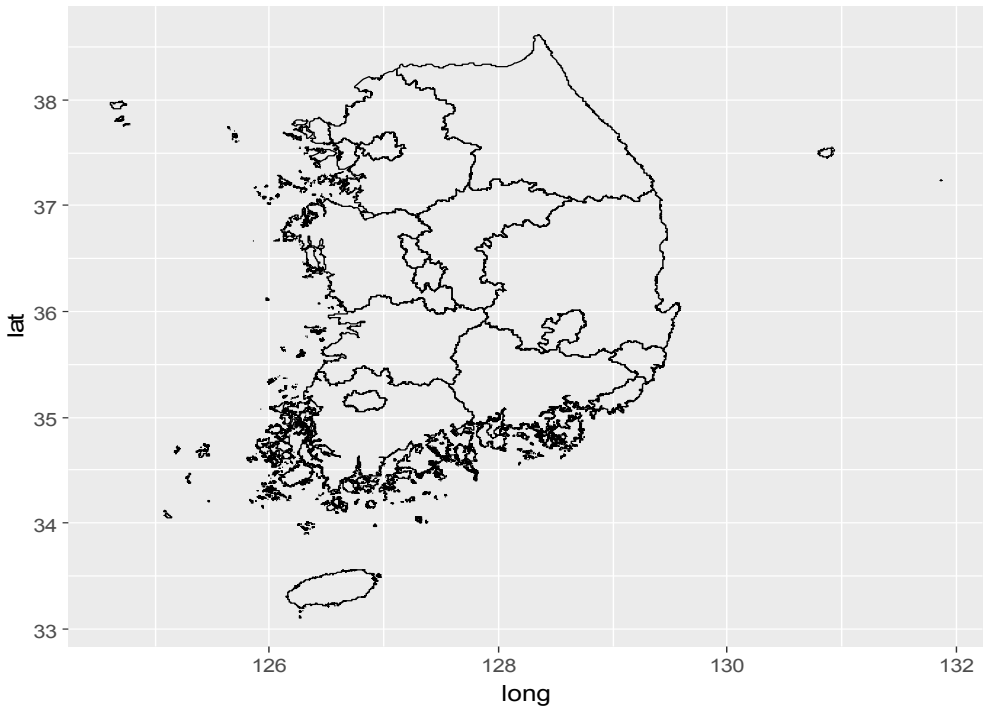
2. ggplot2를 이용한 그림 그리기

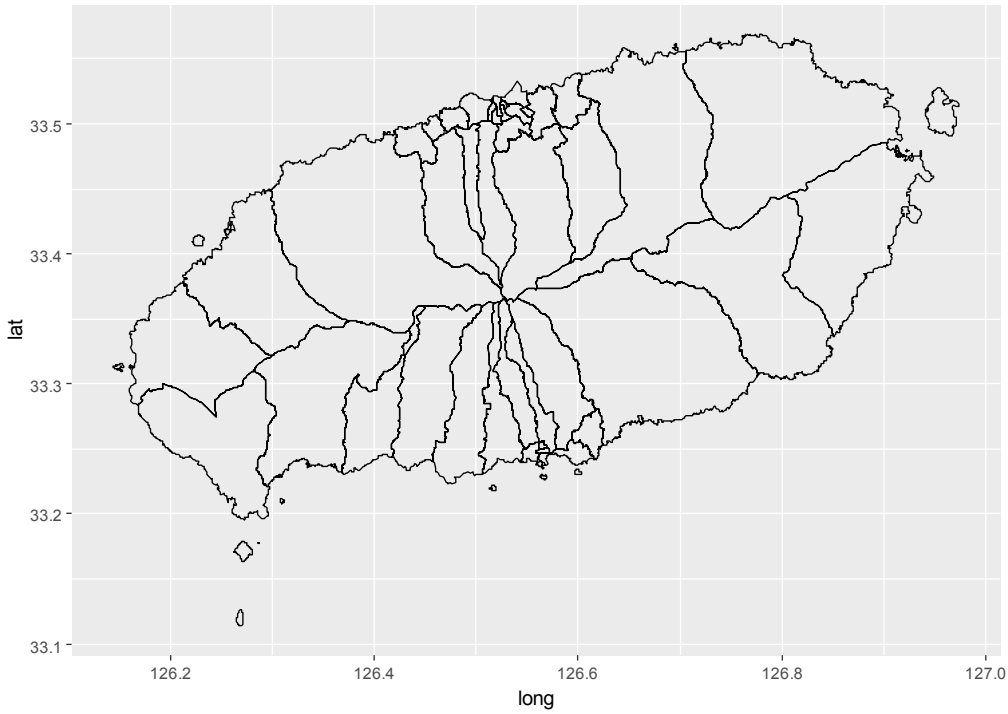
R에서 제공하는 다양한 시각화 관련 패키지 중 하나인 ggplot2 패키지는 히스토그램, 상자그래프 등 다양한 종류의 그래프를 그릴 수 있고 글자 크기, 폰트, 색상 등도 원하는 형태로 설정할 수 있어 ggplot2만으로도 충분히 원하는 시각화가 가능하다.

b1-ch3-8.R을 실행하면 우리나라의 wifi가 설치된 위치를 그림으로 그리거나 제주지역의 행정구역을 그릴 수 있다.

b1-ch3-8.R의 실행결과

```
install.packages("ggplot2")
library(ggplot2)
con1<-url("http://kanggc.iptime.org/book/data/Map.RData")
con2<-url("http://kanggc.iptime.org/book/data/mapJeju.RData")
load(con1)
load(con2)
ggplot() + geom_path(data = map,  aes(x = long,  y = lat,  group = group))
ggplot() + geom_path(data = map.jeju,  aes(x = long,  y = lat,  group = group))
```





3. 기술통계량 계산

데이터의 중심을 측정하는 통계량으로 평균, 중위수, 최빈값 등이 있는데 평균과 중위수는 각각 mean 함수 및 median 함수를 이용하여 구할 수 있으나 최빈값을 구하는 Mode는 기본기능에 없으므로 prettyR이라는 패키지를 통해 구할 수 있다.

한편, 데이터의 흩어짐의 정도를 측정하는 통계량으로 범위, 분산, 표준편차 등이 있다

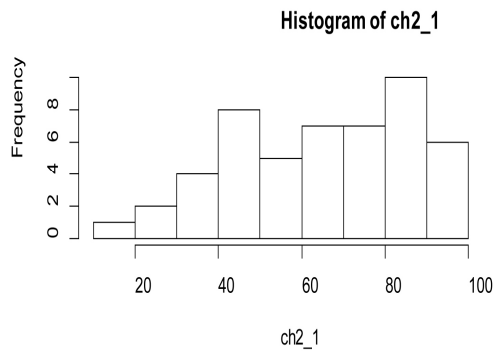
b1-ch3-9.R을 실행하면 데이터의 중심 및 흩어짐의 정도를 측정하는 통계량을 구할 수 있다.

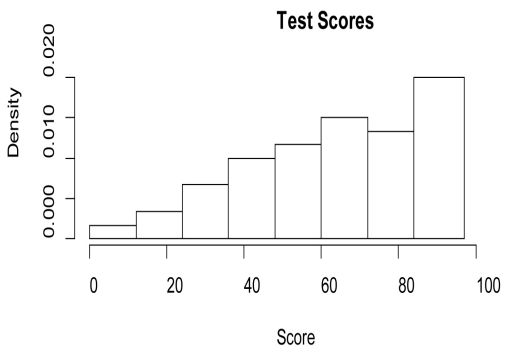
b1-ch3-9.R의 실행결과

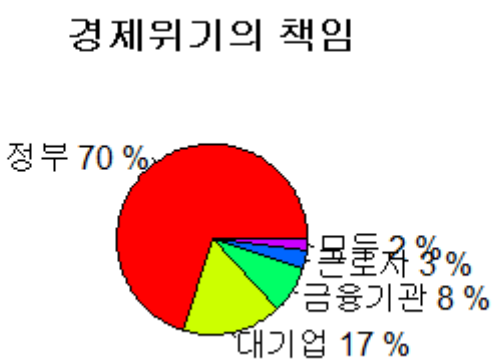
```
> library(prettyR)

> ch2_1<-scan("http://kanggc.iptime.org/book/data/ch2_1.txt")
Read 50 items

> ch2_1
 [1] 65 74 65 36 81 60 43 21 83 64 12 91 60 24 54 69 89 96 86 85 95 85
51
[24] 81 47 62 85 46 49 76 44 72 33 46 49 74 78 48 62 97 31 96 97 88 61
54
[47] 89 77 72 35

> max(ch2_1)
[1] 97

> min(ch2_1)
[1] 12

> mean(ch2_1)
[1] 64.76

> diff(range(ch2_1))
[1] 85

> var(ch2_1)
[1] 490.88

> sd(ch2_1)
[1] 22.15581

> table(ch2_1)[table(ch2_1)[1:length(unique(ch2_1))] == max(table(ch2_1))]
85
 3

> median(ch2_1)
[1] 65
```

```
> Mode(ch2_1)
[1] "85"

> bins<-c(0, 12, 24, 36, 48, 60, 72, 84, 97)

> class<-cut(ch2_1, breaks = bins)

> table(class)
class
 (0,12] (12,24] (24,36] (36,48] (48,60] (60,72] (72,84] (84,97]
      1       2       4       6       7       9       8      13

> table(class)/length(ch2_1)
class
 (0,12] (12,24] (24,36] (36,48] (48,60] (60,72] (72,84] (84,97]
   0.02    0.04    0.08    0.12    0.14    0.18    0.16    0.26

> transform(table(class), Rel_Freq = prop.table(Freq))
     class Freq Rel_Freq
1   (0,12]    1     0.02
2  (12,24]    2     0.04
3  (24,36]    4     0.08
4  (36,48]    6     0.12
5  (48,60]    7     0.14
6  (60,72]    9     0.18
7  (72,84]    8     0.16
8  (84,97]   13     0.26

> hist(ch2_1)

> hist(ch2_1, breaks = bins, main = "Test Scores", xlab = "Score")

> summary(ch2_1)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  12.00   48.25   65.00   64.76   84.50   97.00
```

```
> 수치<-c(70, 17, 8, 3, 2)

> 경제책임<-c("정부", "대기업", "금융기관", "근로자", "모두")

> 경제책임<-paste(경제책임, 수치)

> 경제책임<-paste(경제책임, "%", sep=" ")

> pie(수치, labels=경제책임, col=rainbow(length(경제책임)), main="경제위기의 책임")
```







4. 평균의 계산

평균을 구하는 방법은 산술평균(arithmetic mean), 조화평균(harmonic mean), 기하평균(geometric mean) 등이 있다.

n개의 변수의 산술평균은 변수들의 총합을 변수의 개수 n으로 나눈 값이다.

$$\overline{X}=\frac{X_1+X_2+.....+X_n}{n}=\frac{\sum_{i=1}^{n}X_i}{n}$$

조화평균은 n개의 양수에 대하여 그 역수들을 산술평균한 것의 역수를 말한다. 예를 들어, 두 지점 A, B를 갈 때는 'a' km/h의 속도로, 올 때는 'b' km/h의 속도로 왕복했다면 이 사람의 평균속력은 a와 b의 조화평균에 해당된다.

n개의 양수인 변수의 조화평균은 그 변수의 역수를 산술평균한 것의 역수이다.

$$\frac{1}{H}=\frac{1}{n}(\frac{1}{X_1}+\frac{1}{X_2}+.....+\frac{1}{X_n})=\frac{\sum_{i=1}^{n}\frac{1}{X_i}}{n}$$

예를 들어, 자동차가 처음 10km를 시속 30km로 달리고, 다음 10km를 시속 60km로 달렸을 경우 평균시속은 산술평균인 45km/h가 아니고 조화평균인 40km/h이다.

기하평균은 여러 개의 수를 연속으로 곱해 그 개수의 거듭제곱근으로 구한 수로 흔히 연평균 인구증가율이나 연평균 경제성장률을 구할 때 적용된다.

$$G=\sqrt[n]{X_1X_2....X_n}$$

예를 들어, 2014년부터 2017년까지의 인구변화가 다음의 표와 같을 때 전년도대비비율을 구하면 각각 120%, 130%, 118%가 되는데 이 경우 2014년부터 2017년까지의 연평균 인구증가율은 산술평균인 122.67이 아니고 기하평균인 122.56이다. 그 이유는 2014년부터 산술평균으로 구한 연평균 증가율을 매년 곱하여 2017년 인구를

구해보면 9228명이 되어 실제인구 9204명과 차이가 발생하지만 기하평균으로 구한 연평균 증가율을 매년 곱하여 2017년 인구를 구해보면 9204명이 되어 실제인구와 동일하게 된다.

기하평균의 장점 중의 하나는 최초년도부터 최종년도까지 데이터가 모두 없어도 최초년도와 최종년도의 데이터만 있으면 다음의 식에 의해 연평균 증가율 계산이 가능하다는 점이다.

$$G = ((\frac{\text{최종년도의 값}}{\text{최초년도의 값}})^{(1/n)} - 1) \times 100,$$ n은 경과년도를 나타냄

연도	인구	전년대비비율(%)	$\overline{X}$ × 전년도인구	G × 전년도인구
2014	5000	-	-	-
2015	6000	120	6133	6128
2016	7800	130	7523	7510
2017	9204	118	9228	9204
-	-	-	$\overline{X} = 122.67$	$G = 122.56$

b1-ch3-10.R을 실행하면 산술평균, 조화평균, 기하평균을 각각 구할 수 있다.

b1-ch3-10.R의 실행결과

```
> a<-c(10, 2, 19, 24, 6, 23, 47, 24, 54, 77)

> n<-length(a) # now n is equal to the number of elements in a

> mean(a)
[1] 28.6

> 1/mean(1/a) # compute the harmonic mean
[1] 10.01109

> prod(a)^(1/n) # compute the geometric mean
[1] 18.92809
```

```
> b<-c(30,60)

> mean(b)
[1] 45

> 1/mean(1/b)
[1] 40

> c<-c(120,130,118)

> m<-length(c)

> round(mean(c),digits=2)
[1] 122.67

> round(prod(c)^(1/m),digits=2)
[1] 122.56

> g<-((9204/5000)^(1/3)-1)*100
> round(g, digits=2)
[1] 22.56 # 연평균 증가율이 22.565임을 나타냄
```

5. 두 확률변수의 공분산

두 이산형 확률변수의 확률분포표가 다음과 같이 주어져 있을 경우 이를 이용하여 두 확률변수의 각각의 분산과 두 확률변수의 공분산을 구할 수 있다.

X \ Y	0	1	2	합
1	1/6	1/12	1/12	1/3
3	1/12	1/2	1/12	2/3
합	1/4	7/12	1/6	1

b1-ch3-11.R을 실행하면 두 확률변수의 각각의 분산을 구할 수 있다. 또한 두 확률변수의 공분산 및 상관계수를 각각 구할 수 있다.

b1-ch3-11.R의 실행결과

```
> data<-c(0.167, 0.083, 0.083, 0.333, 0.083, 0.5, 0.083, 0.667, 0.25, 0.583,
0.167, 1)

> mat<-matrix(data, nrow=3, byrow=T)

> rownames(mat)<-c("1", "3", "합계")

> colnames(mat)<-c("0", "1", "2", "합계")

> mat
          0     1     2  합계
1     0.167 0.083 0.083 0.333
3     0.083 0.500 0.083 0.667
합계 0.250 0.583 0.167 1.000

> mu_x<-1*mat[1,4]+3*mat[2,4]

> mu_y<-0*mat[3,1]+1*mat[3,2]+2*mat[3,3]

> mu_x
[1] 2.334

> mu_y
[1] 0.917

> var_x<-1^2*mat[1,4]+3^2*mat[2,4]- mu_x^2

> var_y<-0^2*mat[3,1]+1^2*mat[3,2]+2^2*mat[3,3]-mu_y^2

> var_x
[1] 0.888444
```

```
> var_y
[1] 0.410111

> p_xy<-c(0.167, 0.083, 0.083, 0.083, 0.5, 0.083)

> xy<-c(0,1,2,0,3,6)

> cov_xy<-sum(p_xy*xy)-(mu_x*mu_y)

> cov_xy
[1] 0.106722

> corr_xy<-cov_xy/sqrt(var_x*var_y)

> corr_xy
[1] 0.1768024
```

제 4 장

이론적 확률분포

1. 이론적 확률분포의 관계
2. 베르누이분포
3. 이항분포
4. 포아송분포
5. 균등분포
6. 표준정규분포
7. χ^2-분포
8. t-분포
9. F-분포

제4장 이론적 확률분포

1. 이론적 확률분포의 관계

이론적 확률분포는 <그림 4-1>에 나타나 있는 바와 같이 크게 이산형 확률분포와 연속형 확률분포로 나누어진다.

이산형 확률분포에는 포아송분포와 이항분포가 있는데 이항분포에서 n이 크고, p가 작으면 포아송분포와 유사하게 된다.

이산형 확률분포인 이항분포에서 n이 크고, p가 0.5이면 연속형 확률분포 중 정규분포와 유사하게 된다.

정규분포를 평균이 0, 표준편차가 1이 되게 표준화하면 표준정규분포가 되는데 이 분포가 연속형 확률분포의 출발이 된다.

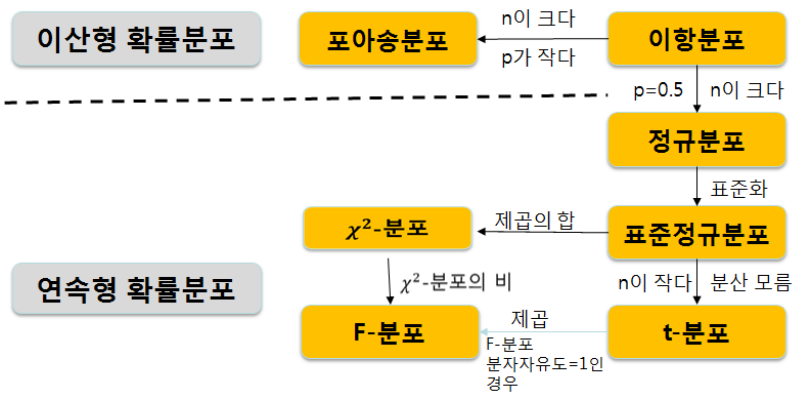
서로 독립적인 표준정규분포의 제곱의 합은 χ^2(카이제곱)-분포에 따르게 되며, χ^2-분포의 모양은 자유도에 의해 결정되는데 이에 대한 자세한 설명은 χ^2-분포에서 한다.

표준정규분포에서 n이 작고, 분산을 모를 경우 t-분포에 따르게 되며, t-분포의 모양은 자유도에 의해 결정되는데 이에 대한 자세한 설명은 t-분포에서 한다.

서로 독립적인 두 개의 χ^2-분포의 비율은 F-분포에 따르게 되며, F-분포의 모양은 분자 및 분모의 자유도에 의해 결정되는데 이에 대한 자세한 설명은 F-분포에서 한다.

자유도가 n인 t-분포에 따르는 t-통계량을 제곱하면 분모의 자유도가 1이고 분자의 자유도가 n인 F-분포와 동일하다.

이 외에도 이산형 확률분포로서 베르누이분포와 연속형 확률분포로서 균등분포 등이 있다.



〈그림 4-1〉 이론적 확률분포의 관계

2. 베르누이분포

실험에서 결과가 둘 중의 하나로 나타나는 실험을 베르누이 시행(Bernoulli trial)이라고 하는데 그 중 하나를 성공(success: s)이라 하고 다른 하나를 실패(failure: f)라고 정의한다.

따라서 베르누이 시행은 실험의 결과가 s 또는 f 인 확률실험이라고 할 수 있으며, 표본공간은 Ω= {s, f}가 된다. 이 실험에서 결과가 s일 확률이 p라면 확률의 기본원리에 의하여 실험결과가 f일 확률은 1-p이다.

베르누이 시행에서 결과가 s이면 '1'이고, 결과가 f이면 '0'이라고 정의된 확률변수를 베르누이 확률변수라고 하고, 베르누이 확률변수 X의 확률분포는 다음과 같다. 베르누이 분포의 모양은 확률 p의 값에 의하여 결정되므로 이 분포의 모수는 p이므로 X~Bernoulli(p)로 정의한다.

$$P(X=x)=p^x(1-p)^{1-x},\ x=0,1$$

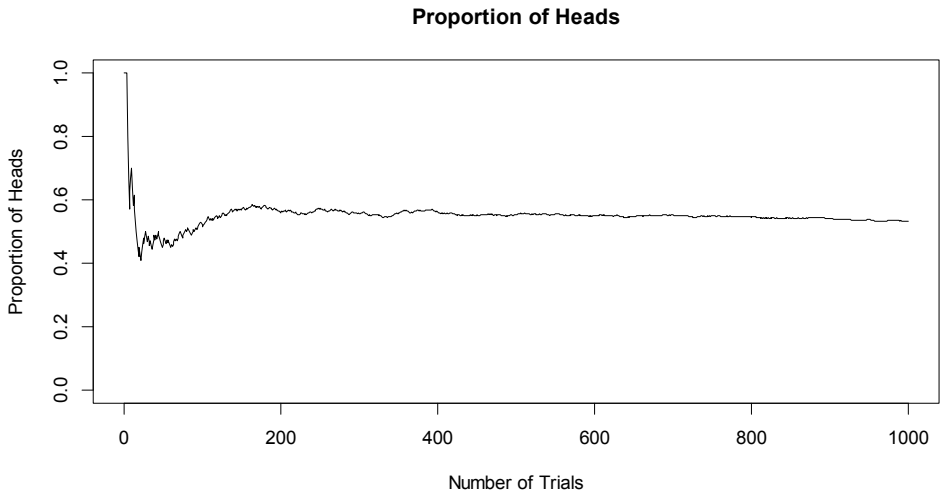
베르누이 분포의 모양은 확률 p의 값에 의하여 결정되므로 이 분포의 모수는 p이며, 베르누이분포의 평균은 p, 분산은 p(1-p)이다.

동일한 실험을 무한히 반복할 때 한 사건에 대하여 상대 도수에 의하여 계산된 확률은 고전적 의미의 확률에 접근한다고 할 수 있다. 동전을 던지는 실험은 그 결과가 앞면과 뒷면 밖에 없는데 동전을 던지는 실험을 무한히 반복하면 동전의 앞면이 나오는 확률은 0.5로 점근적으로 수렴한다.

b1-ch4-1.R을 실행하면 동전을 던지는 실험을 번복할수록 동전의 앞면이 나오는 확률이 0.5로 수렴하는 것을 볼 수 있다.

b1-ch4-1.R의 실행결과

```
> set.seed(12345)

> x<-rbinom(1000, 1, .5)

> (table(x))
x
  0   1
469 531

> (mean(x))
[1] 0.531

> cx<-cumsum(x)

> heads<-numeric(1000)

> for (i in 1:1000) {
+     heads[i]<-cx[i]/i
+ }

> plot(heads, type="l", xlab="Number of Trials", ylab="Proportion of Heads",
ylim=c(0,1), main="Proportion of Heads")
```



3. 이항분포

(1) 이항분포의 모양

이항분포(binomial distribution)는 베르누이 시행을 독립적으로 n번 반복했을 때 나타나는 결과에 있어서 성공(s)의 횟수에 대한 분포를 구하는 것이다.

성공의 확률이 p이고 실패의 확률이 q(q = 1-p)인 베르누이 시행을 독립적으로 n번 반복하였을 때 나타나는 성공의 횟수를 확률변수 X 라고 할 때, X를 이항확률변수(binomial random variable)라 하고, X~B(n,p)로 정의하며 확률분포는 다음과 같다

$$P(X=k)=\frac{n!}{k!(n-k)!}p^k q^{n-k},\ k=0,1,...n$$

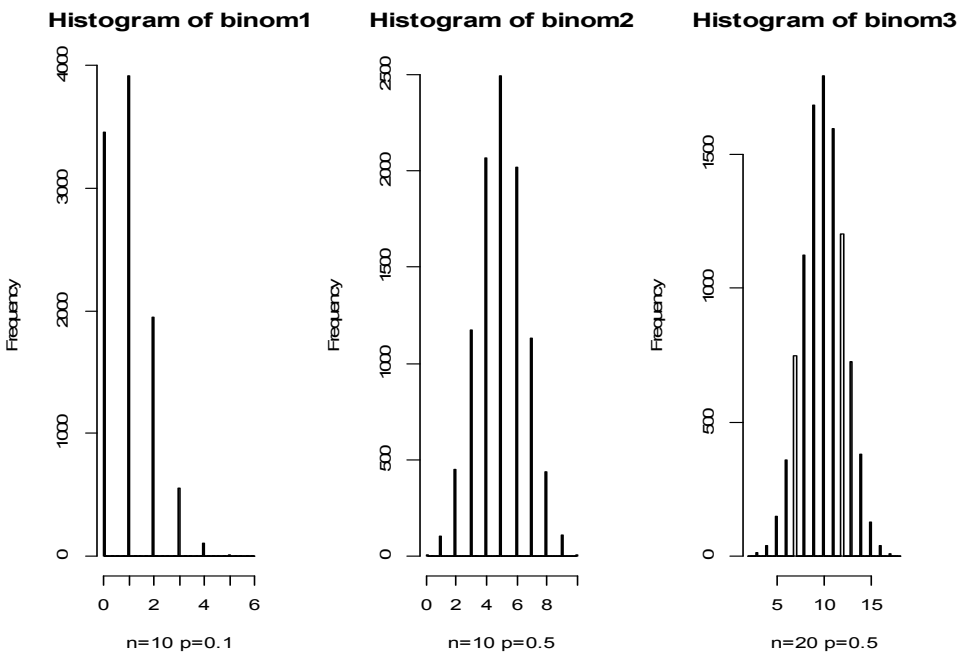
이항분포에서 모수는 각 시행에서 성공이 나타날 확률 p와 시행횟수 n이며, 이항분포의 평균은 np, 분산은 npq이다.

이항분포는 성공확률 p가 0.5이면 평균 $\mu=np=\frac{n}{2}$을 중심으로 좌우대칭인 분포를 가지며, n과 p의 크기에 따라 모양이 결정된다.

b1-ch4-2.R을 실행하면 n과 p에 따라 이항분포의 모양이 변하는 것을 확인할 수 있으며, p=0.5이고 n이 커질 때 좌우대칭의 정규분포와 유사함을 확인할 수 있다.

b1-ch4-2.R의 실행결과

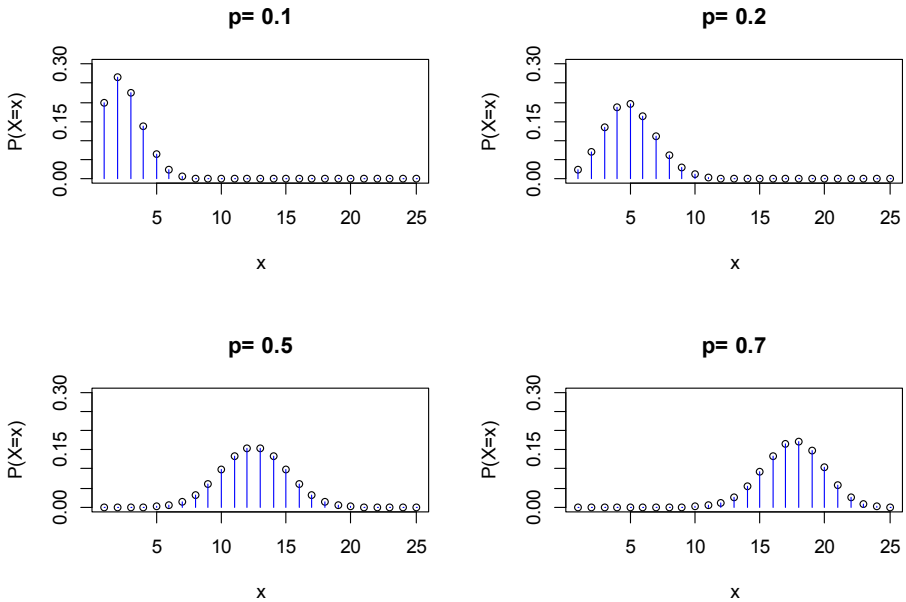
```
set.seed(12345)

r<-10000

binom1<-rbinom(r, 10, 0.1)
binom2<-rbinom(r, 10, 0.5)
binom3<-rbinom(r, 20, 0.5)

par(mfrow=c(1,3))

hist(binom1, breaks=100, xlab="n=10 p=0.1")
hist(binom2, breaks=100, xlab="n=10 p=0.5")
hist(binom3, breaks=100, xlab="n=20 p=0.5")
```



b1-ch4-3.R을 실행하면 n이 주어진 상태에서 p에 따라 이항분포의 모양이 변하는 것을 확인할 수 있으며, p=0.5에 가까울수록 좌우대칭의 정규분포와 유사함을 확인할 수 있다.

b1-ch4-3.R의 실행결과

```
par(mfrow=c(2,2)) # 한 페이지에 그림을 좌우상하 하나씩 4개를 그림
> par(mfrow=c(2,2))

> n<-25 # 시행횟수

> p_list<-c(0.1, 0.2, 0.5, 0.7) # 발생확률

> for (i in 1:length(p_list)) {
+       p_x<-dbinom(x=1:n, n, p_list[i])
+       plot(x=1:n, p_x, xlab="x", ylab="P(X=x)",
+            ylim=c(0, 0.3), xli .... [TRUNCATED]
```



(2) 이항분포의 확률분포표

n이 5이고 성공확률 p가 다음의 표와 같을 때 주어진 성공확률에서 다음과 같이 5번 시행하여 a번 성공하는 누적확률을 구할 수 있다.

$$P(X \leq a) = \sum_{x=0}^{a} P(x)$$

a	0.10	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	0.95
0	.590	.328	.168	.078	.031	.010	.002	.000	.000	.000
1	.919	.737	.528	.337	.187	.087	.031	.007	.000	.000
2	.991	.942	.837	.683	.500	.317	.163	.058	.009	.001
3	1.00	.993	.969	.914	.813	.663	.472	.263	.081	.023
4	1.00	1.00	.998	.990	.969	.922	.832	.672	.410	.226

b1-ch4-4.R을 실행하면 위와 동일한 누적이항확률분포표를 만들 수 있음을 확인할 수 있다.

b1-ch4-4.R의 실행결과

```
> binom11<-rep(NA,5)

> binom12<-rep(NA,5)

> binom13<-rep(NA,5)

> binom11[1]<-pbinom(0, 5, 0.1)

> binom12[1]<-pbinom(0, 5, 0.2)

> binom13[1]<-pbinom(0, 5, 0.3)

> for(i in 2:5) {
+     binom11[i]<-pbinom(i-1, 5, 0.1)
+ }
```

```
> for(i in 2:5) {
+     binom12[i]<-pbinom(i-1, 5, 0.2)
+ }

> for(i in 2:5) {
+     binom13[i]<-pbinom(i-1, 5, 0.3)
+ }

> (binom<-cbind(binom11,binom12, binom13))
      binom11 binom12 binom13
[1,] 0.59049 0.32768 0.16807
[2,] 0.91854 0.73728 0.52822
[3,] 0.99144 0.94208 0.83692
[4,] 0.99954 0.99328 0.96922
[5,] 0.99999 0.99968 0.99757
> source('~/.active-rstudio-document', echo=TRUE)

> binom11<-rep(NA,5)

> binom12<-rep(NA,5)

> binom13<-rep(NA,5)

> binom14<-rep(NA,5)

> binom15<-rep(NA,5)

> binom16<-rep(NA,5)

> binom17<-rep(NA,5)

> binom18<-rep(NA,5)

> binom19<-rep(NA,5)

> binom11[1]<-pbinom(0, 5, 0.1)
```

```
> binom12[1]<-pbinom(0, 5, 0.2)

> binom13[1]<-pbinom(0, 5, 0.3)

> binom14[1]<-pbinom(0, 5, 0.4)

> binom15[1]<-pbinom(0, 5, 0.5)

> binom16[1]<-pbinom(0, 5, 0.6)

> binom17[1]<-pbinom(0, 5, 0.7)

> binom18[1]<-pbinom(0, 5, 0.8)

> binom19[1]<-pbinom(0, 5, 0.9)

> for(i in 2:5) {
+     binom11[i]<-pbinom(i-1, 5, 0.1)
+ }

> for(i in 2:5) {
+     binom12[i]<-pbinom(i-1, 5, 0.2)
+ }

> for(i in 2:5) {
+     binom13[i]<-pbinom(i-1, 5, 0.3)
+ }

> for(i in 2:5) {
+     binom14[i]<-pbinom(i-1, 5, 0.4)
+ }

> for(i in 2:5) {
+     binom15[i]<-pbinom(i-1, 5, 0.5)
+ }
```

```
> for(i in 2:5) {
+     binom16[i]<-pbinom(i-1, 5, 0.6)
+ }

> for(i in 2:5) {
+     binom17[i]<-pbinom(i-1, 5, 0.7)
+ }

> for(i in 2:5) {
+     binom18[i]<-pbinom(i-1, 5, 0.8)
+ }

> for(i in 2:5) {
+     binom19[i]<-pbinom(i-1, 5, 0.9)
+ }

> round((binom<-cbind(binom11,binom12, binom13,binom14,binom15,
binom16,binom17,binom18, binom19)),digits=3)
     binom11 binom12 binom13 binom14 binom15 binom16 binom17 binom18
binom19
[1,]   0.590   0.328   0.168   0.078   0.031   0.010   0.002   0.000   0.000
[2,]   0.919   0.737   0.528   0.337   0.187   0.087   0.031   0.007   0.000
[3,]   0.991   0.942   0.837   0.683   0.500   0.317   0.163   0.058   0.009
[4,]   1.000   0.993   0.969   0.913   0.812   0.663   0.472   0.263   0.081
[5,]   1.000   1.000   0.998   0.990   0.969   0.922   0.832   0.672   0.410
```

4. 포아송분포

(1) 포아송분포의 모양

포아송분포(poison distribution)는 주어진 단위시간, 거리, 영역 등에서 어떤 사건이 발생하는 횟수를 측정하는 확률분포를 말한다. 특정지역에서 제한된 시간 내에 발생하는 교통사고 수의 분포나 자동생산라인에서 특정시간에 발생하는 불량품 수의 분포 등이 대표적인 포아송분포에 따른다.

단위구간 내에서 어떤 사건이 평균 μ회 발생한다고 하고, 확률변수 X를 사건의 발생횟수라고 할 때, X~Poisson (μ)로 표현하며 사건이 k번 발생될 확률은 다음과 같다.

$$P(X=k)=\frac{\mu^k}{k!}e^{-\mu}, k=0,1,2...$$

(예제) 최근 올림픽도로에서는 하루 평균 5건의 교통사고가 발생한다. 교통사고의 발생횟수가 포아송분포를 따른다고 할 때, 다음의 확률을 계산하라

① 어느 날 교통사고가 전혀 일어나지 않을 확률은 얼마인가?

```
> poi1<-ppois(0, 5.0) # 평균이 5인 포아송분포에서 P(x=0)일 확률
> poi1
[1] 0.006737947
```

② 어느 날 교통사고가 3번 이상 일어날 확률은 얼마인가?

```
> poi2<-1-(ppois(2, 5.0)) # 평균이 5인 포아송분포에서 1-P(x<3)일 확률
> poi2
[1] 0.875348
```

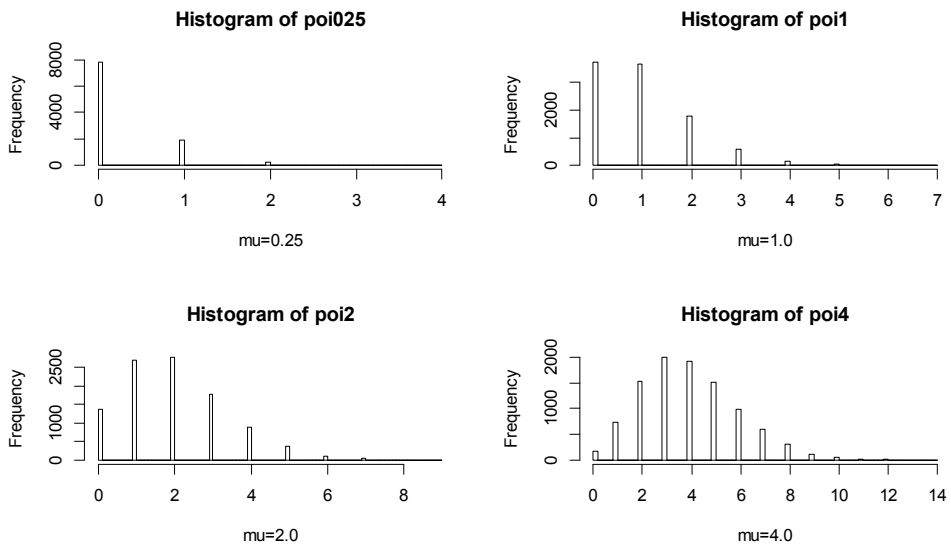
포아송분포에서 모수는 평균 μ이며, 포아송분포의 평균은 μ이고, 분산도 μ이다. 포아송분포의 모양은 평균이 작을 때는 좌우비대칭이나 평균이 증가함에 따라 평균을 중심으로 좌우대칭의 모양으로 변한다. 즉, 정규분포에 가까워지고, 분산은 커진다.

b1-ch4-5.R을 실행하면 평균 μ에 따라 포아송분포의 모양이 변하는 것을 확인할 수 있으며, 평균 μ가 커질 때 좌우대칭의 분포에 가까워지는 것을 확인할 수 있다.

b1-ch4-5.R의 실행결과

```
set.seed(12345)

n<-10000;
```

```
poi025<-rpois(n, 0.25)
poi1<-rpois(n, 1)
poi2<-rpois(n, 2)
poi4<-rpois(n, 4)

par(mfrow=c(2,2))

hist(poi025, breaks=100, xlab="mu=0.25")
hist(poi1, breaks=100, xlab="mu=1.0")
hist(poi2, breaks=100, xlab="mu=2.0")
hist(poi4, breaks=100, xlab="mu=4.0")
```

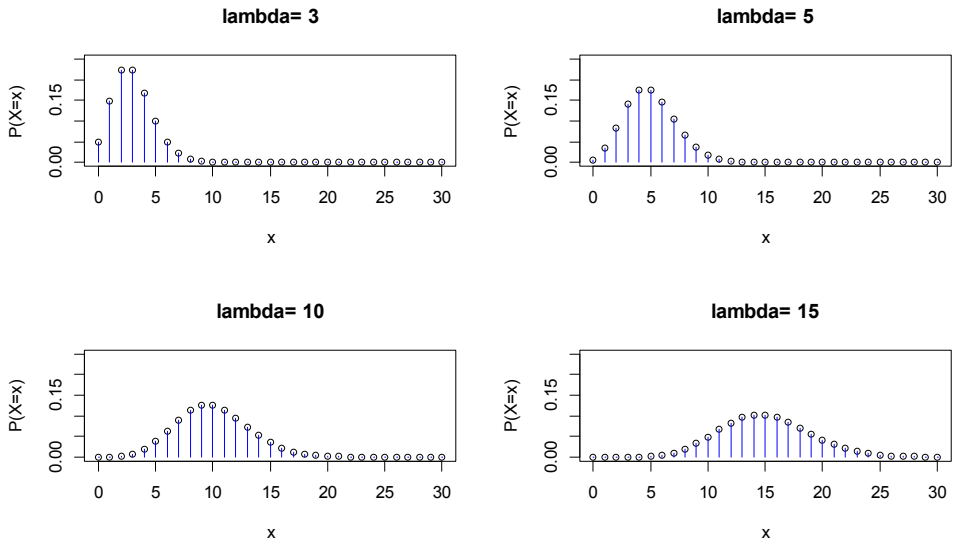


b1-ch4-6.R을 실행하면 평균 μ에 따라 포아송분포의 모양이 변하는 것을 확인할 수 있으며, μ가 커질수록 좌우대칭의 정규분포와 유사함을 확인할 수 있다.

b1-ch4-6.R의 실행결과

```
> par(mfrow=c(2,2)) # 한 페이지에 그림을 좌우상하 하나씩 4개를 그림

> lambda_list<-c(3, 5, 10, 15) # 평균
```

```
> x_list<-30 # 발생횟수를 1부터 x축에 보여줄 최대값

> for (i in 1:length(lambda_list)) {
+         p_x<-dpois(x=0:x_list,lambda_list[i])
+           plot(x=0:x_list, p_x, xlab="x", ylab="P(X=x)", ylim=c(0, 0.25), ....
[TRUNCATED]
```



(2) 포아송분포의 확률분포표

평균 μ가 다음의 표와 같을 때 어떤 사건이 발생되는 횟수의 누적확률을 구할 수 있다.

$$P(X \leq c) = \sum_{k=0}^{c} \frac{\mu^k e^{-\mu}}{k!}, \mu : 기댓값$$

c \ μ	0.02	0.04	0.06	0.08	0.10	0.20	0.30	0.40
0	0.980	0.961	0.942	0.923	0.905	0.819	0.741	0.670
1	1.000	0.999	0.998	0.997	0.995	0.982	0.963	0.938
2	1.000	1.000	1.000	1.000	1.000	0.999	0.996	0.992
3	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.999
4	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

b1-ch4-7.R을 실행하면 위와 동일한 누적포아송확률분포표를 만들 수 있음을 확인할 수 있다.

b1-ch4-7.R의 실행결과

```
> poi11<-rep(NA,5)

> poi12<-rep(NA,5)

> poi13<-rep(NA,5)

> poi14<-rep(NA,5)

> poi15<-rep(NA,5)

> poi16<-rep(NA,5)

> poi17<-rep(NA,5)

> poi18<-rep(NA,5)

> poi11[1]<-ppois(0, 0.02)

> poi12[1]<-ppois(0, 0.04)

> poi13[1]<-ppois(0, 0.06)

> poi14[1]<-ppois(0, 0.08)
```

```
> poi15[1]<-ppois(0, 0.1)

> poi16[1]<-ppois(0, 0.2)

> poi17[1]<-ppois(0, 0.3)

> poi18[1]<-ppois(0, 0.4)

> for(i in 2:5) {
+    poi11[i]<-ppois(i-1, 0.02)
+ }

> for(i in 2:5) {
+    poi12[i]<-ppois(i-1, 0.04)
+ }

> for(i in 2:5) {
+    poi13[i]<-ppois(i-1, 0.06)
+ }

> for(i in 2:5) {
+    poi14[i]<-ppois(i-1, 0.08)
+ }

> for(i in 2:5) {
+    poi15[i]<-ppois(i-1, 0.1)
+ }

> for(i in 2:5) {
+    poi16[i]<-ppois(i-1, 0.2)
+ }

> for(i in 2:5) {
+    poi17[i]<-ppois(i-1, 0.3)
+ }

> for(i in 2:5) {
```

```
+     poi18[i]<-ppois(i-1, 0.4)
+ }

> round((poi<-cbind(poi11,poi12,poi13,poi14,poi15,poi16,poi17,poi18)),digits=3)
      poi11 poi12 poi13 poi14 poi15 poi16 poi17 poi18
[1,]  0.98 0.961 0.942 0.923 0.905 0.819 0.741 0.670
[2,]  1.00 0.999 0.998 0.997 0.995 0.982 0.963 0.938
[3,]  1.00 1.000 1.000 1.000 1.000 0.999 0.996 0.992
[4,]  1.00 1.000 1.000 1.000 1.000 1.000 1.000 0.999
[5,]  1.00 1.000 1.000 1.000 1.000 1.000 1.000 1.000
```

5. 균등분포

균등분포(uniform distribution)는 연속형 분포에서 가장 단순한 분포형태로 특정 구간 내의 값들이 나타날 가능성이 균등한 분포를 말한다.

연속형 확률변수 X가 실수구간 [a,b]에서 나타날 가능성이 균등할 때, X는 균등분포를 따른다고 하며 X~U(a,b)로 표현한다.

X의 확률밀도함수는 다음과 같다

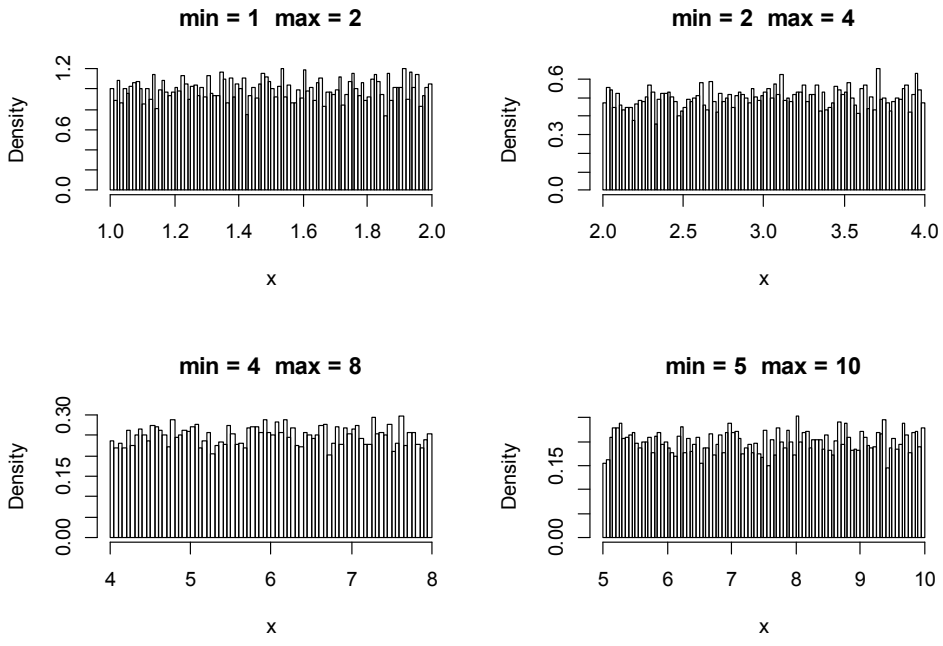
$$f(x)=\begin{cases}\dfrac{1}{b-a}, a \le X \le b \\ 0, \quad \text{다른 곳에서}\end{cases}$$

확률변수 X가 X~U(a,b)라고 할 때, X의 평균은 $\frac{b+a}{2}$, 분산은 $\frac{1}{12}(b-a)^2$ 이다.

b1-ch4-8.R을 실행하면 X~U(1,2), X~U(2,4), X~U(4,8), X~U(5,10)에서 X가 각각 실수구간에서 나타날 가능성이 균등함을 확인할 수 있다.

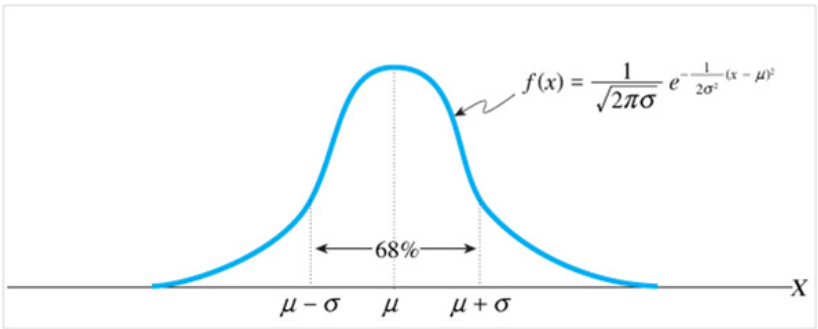
b1-ch4-8.R의 실행결과

```
set.seed(12345)
n<-10000;

min_list<-c(1,2,4,5)
max_list<-c(2,4,8,10)
par(mfrow=c(2,2))

unif1<-runif(n, min=1, max=2)
unif2<-runif(n, min=2, max=4)
unif3<-runif(n, min=4, max=8)
unif4<-runif(n, min=5, max=10)

(munif1<-mean(unif1))
(munif2<-mean(unif2))
(munif3<-mean(unif3))
(munif4<-mean(unif4))

(vunif1<-var(unif1))
(vunif2<-var(unif2))
(vunif3<-var(unif3))
(vunif4<-var(unif4))

for (i in 1:length(min_list)) {
  hist(runif(n, min=min_list[i], max=max_list[i]), freq=F, breaks=100,
xlab="x", main=paste("min =", min_list[i], " max =", max_list[i]))
 }
```



6. 표준정규분포

정규분포란 다음의 <그림 4-2>와 같이 분포의 형태가 종을 엎어 놓은 모양인 분포를 말하며, 분포의 형태는 평균 μ와 분산 σ^2에 의해 결정된다.



<그림 4-2> 정규분포의 모양 및 확률밀도함수

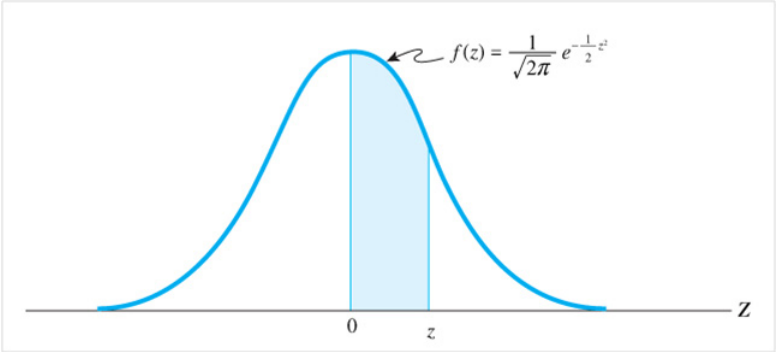
확률변수 X가 평균 μ와 분산 σ^2을 갖는 정규분포를 따른다면 $X \sim N(\mu, \sigma^2)$ 라고 표현하며, X의 확률밀도함수는 다음과 같다.

$$f(x) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{1}{2\sigma^2}(x-\mu)^2} \quad , \; -\infty < x < \infty$$

확률변수 Z가 평균이 0이고 분산이 1인 정규분포를 따를 때 Z는 표준정규분포를 따른다고 하며, $Z \sim N(0,1)$ 로 표현하고, Z의 확률밀도함수는 다음과 같다.

$$f(z) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}z^2} \quad , \; -\infty < z < \infty$$

표준정규분포의 형태는 <그림 4-3>과 같으며, 중심 0에서부터 양의 값 z까지의 확률은 색칠한 부분의 넓이와 같다



<그림 4-3> 표준정규분포의 모양 및 확률밀도함수

z가 0.45이면 0부터 z까지의 확률은 0.1736이 된다.

z	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09
0.0	0.0000	0.0040	0.0080	0.0120	0.0160	0.0199	0.0239	0.0279	0.0319	0.0359
0.1	0.0398	0.0438	0.0478	0.0517	0.0557	0.0596	0.0636	0.0675	0.0714	0.0753
0.2	0.0793	0.0832	0.0871	0.0910	0.0948	0.0987	0.1026	0.1064	0.1103	0.1141
0.3	0.1179	0.1217	0.1255	0.1293	0.1331	0.1368	0.1406	0.1443	0.1480	0.1517
0.4	0.1554	0.1591	0.1628	0.1664	0.1700	0.1736	0.1772	0.1808	0.1844	0.1879

b1-ch4-9.R을 실행하면 위와 동일한 표준정규분포표를 만들 수 있음을 확인할 수 있다.

b1-ch4-9.R의 실행결과

```
> z00<-rep(NA,10)
> z01<-rep(NA,10)
> z02<-rep(NA,10)
> z03<-rep(NA,10)
> z04<-rep(NA,10)

> for(i in 1:10) {
+    z00[i]<-pnorm((i-1)/100, 0, 1)-0.5
+ }

> (z00<-round(z00, digits=4))
 [1] 0.0000 0.0040 0.0080 0.0120 0.0160 0.0199 0.0239 0.0279 0.0319 0.0359

> for(i in 10:19) {
+    z01[i]<-pnorm(i/100, 0, 1)-0.5
+ }

> (z01<-round(z01[10:19], digits=4))
 [1] 0.0398 0.0438 0.0478 0.0517 0.0557 0.0596 0.0636 0.0675 0.0714 0.0753

> for(i in 20:29) {
+    z02[i]<-pnorm(i/100, 0, 1)-0.5
+ }

> (z02<-round(z02[20:29], digits=4))
 [1] 0.0793 0.0832 0.0871 0.0910 0.0948 0.0987 0.1026 0.1064 0.1103 0.1141

> for(i in 30:39) {
+    z03[i]<-pnorm(i/100, 0, 1)-0.5
+ }

> (z03<-round(z03[30:39], digits=4))
```

```
 [1] 0.1179 0.1217 0.1255 0.1293 0.1331 0.1368 0.1406 0.1443 0.1480 0.1517

> for(i in 40:49) {
+    z04[i]<-pnorm(i/100, 0, 1)-0.5
+ }

> (z04<-round(z04[40:49], digits=4))
 [1] 0.1554 0.1591 0.1628 0.1664 0.1700 0.1736 0.1772 0.1808 0.1844 0.1879

> zdist<-rbind(z00,z01,z02,z03,z04)

> (zdist<-round(zdist, digits=4))
      [,1]   [,2]   [,3]   [,4]   [,5]   [,6]   [,7]   [,8]   [,9]  [,10]
z00 0.0000 0.0040 0.0080 0.0120 0.0160 0.0199 0.0239 0.0279 0.0319 0.0359
z01 0.0398 0.0438 0.0478 0.0517 0.0557 0.0596 0.0636 0.0675 0.0714 0.0753
z02 0.0793 0.0832 0.0871 0.0910 0.0948 0.0987 0.1026 0.1064 0.1103 0.1141
z03 0.1179 0.1217 0.1255 0.1293 0.1331 0.1368 0.1406 0.1443 0.1480 0.1517
z04 0.1554 0.1591 0.1628 0.1664 0.1700 0.1736 0.1772 0.1808 0.1844 0.1879
```

7. χ^2-분포

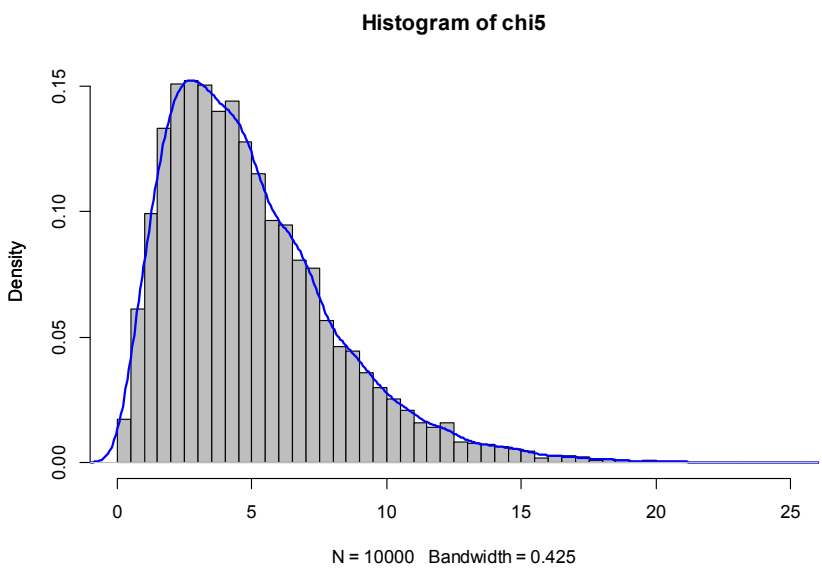
(1) χ^2-분포의 모양

표준정규분포의 제곱의 합이 카이제곱분포에 따른다. 확률변수 $Z_1, Z_2, \ldots, Z_n$ 이 서로 독립적으로 표준정규분포에 따를 때, $Z_1, Z_2, \ldots, Z_n$ 의 제곱합 $\sum_{i=1}^{n} Z_i^2$ 은 자유도가 n인 χ^2-분포를 따른다. 자유도가 n인 χ^2-분포의 평균은 n, 분산은 2n이다.

b1-ch4-10.R을 실행하면 자유도가 5인 χ^2-분포 및 density를 그려주고 있는데 평균이 5에 근사함을 확인할 수 있다.

b1-ch4-10.R의 실행결과

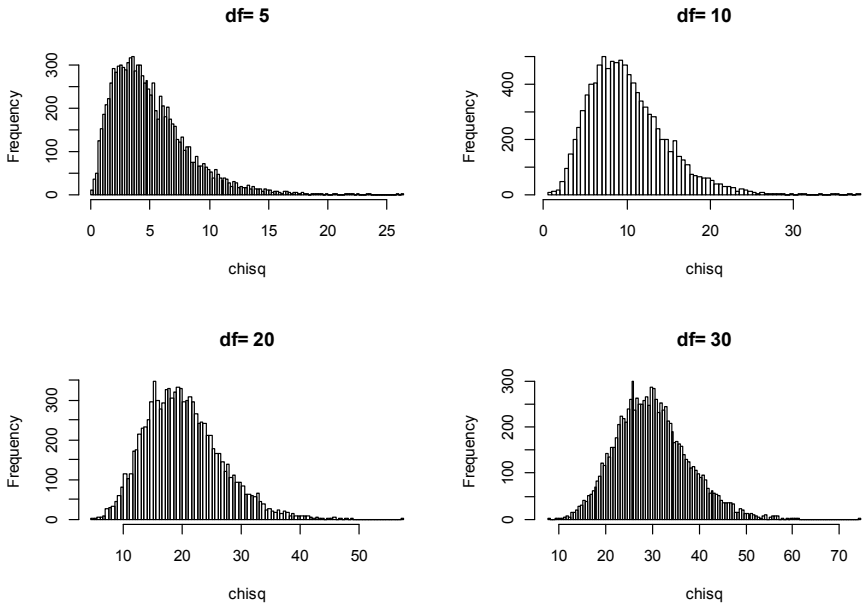
```
> set.seed(12345)
> n<-10000;
> z1<-rnorm(n,0,1)
> z2<-rnorm(n,0,1)
> z3<-rnorm(n,0,1)
> z4<-rnorm(n,0,1)
> z5<-rnorm(n,0,1)
> chi5<-z1^2 + z2^2 + z3^2 + z4^2 + z5^2
> hist(chi5,  freq = F,  col = "grey",  xlab = "",  xlim = c(0,  25),  breaks = 100)
> par(new = T)
> plot(density(chi5),  axes = F,  main = "",  xlim = c(0,  25),  lwd = 2,  col = "blue")
```



b1-ch4-11.R을 실행하면 각각 자유도가 5, 10, 20, 30인 χ^2-분포를 그려주고 있는데 자유도에 따라 χ^2-분포의 모양이 변하는 것을 확인할 수 있다.

b1-ch4-11.R의 실행결과

```
> set.seed(12345)

> n<-10000;

> df_list<-c(5,10,20,30)

> par(mfrow=c(2,2))

> for (i in 1:length(df_list)) {
+         hist(rchisq(n, df=df_list[i], ncp=0), breaks=100, xlab="chisq",
main=paste("df=", df_list[i]))
+ }
```

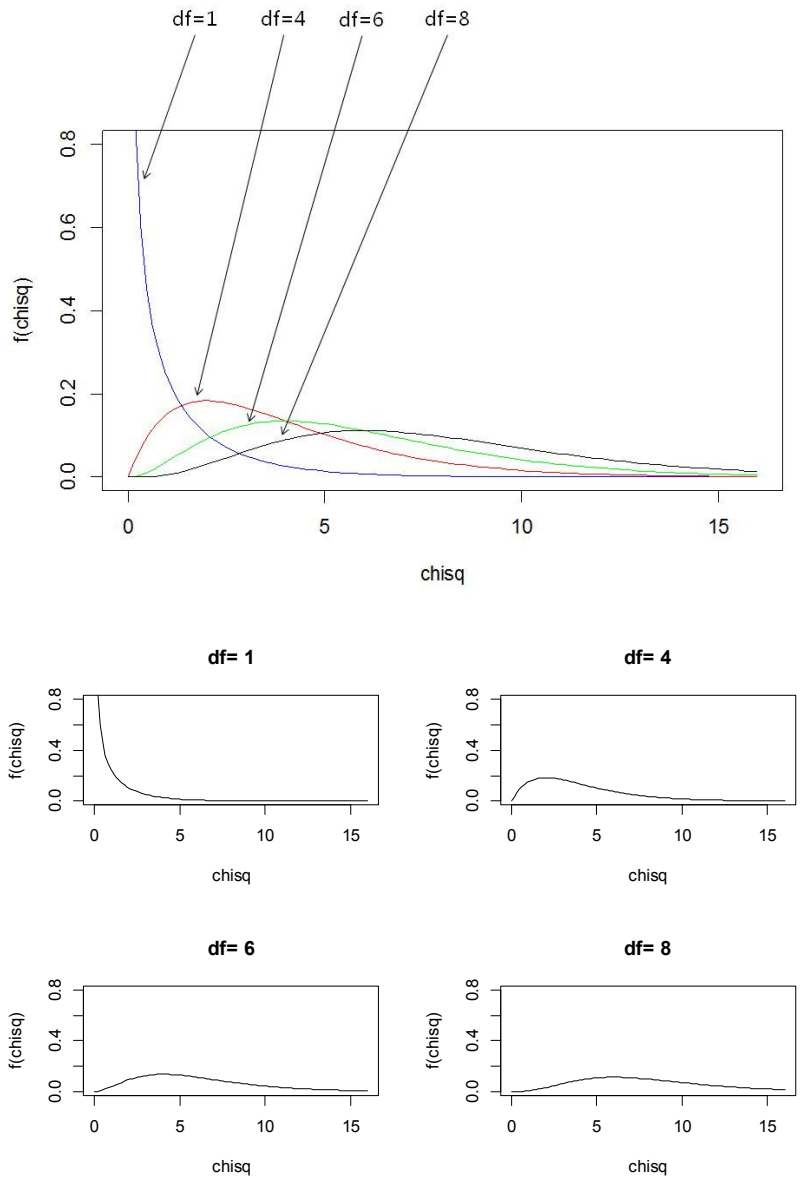


χ^2-분포는 자유도가 모수이므로 자유도의 크기에 따라 분포의 형태가 달라진다.

b1-ch4-12.R을 실행하면 각각 자유도가 1, 4, 6, 8인 χ^2-분포를 그려주고 있는데 자유도에 따라 χ^2-분포의 모양이 변하는 것을 확인할 수 있으며, 자유도가 클수록 정규분포와 근사한 분포 형태를 갖는다.

b1-ch4-12.R의 실행결과

```
> n_list<-c(2,5,7,9) # 표본수(n)

> df_list<-n_list-1 # 자유도

> curve(dchisq(x, 1, ncp=0), add=T, col="blue", xlim=c(0, 16), ylim=c(0, 0.8), xlab="chisq", ylab="f(chisq)")

> curve(dchisq(x, 4, ncp=0), add=T, col="red", xlim=c(0, 16), ylim=c(0, 0.8), xlab="chisq", ylab="f(chisq)")

> curve(dchisq(x, 6, ncp=0), add=T, col="green", xlim=c(0, 16), ylim=c(0, 0.8), xlab="chisq", ylab="f(chisq)")

> curve(dchisq(x, 8, ncp=0), add=T, col="black", xlim=c(0, 16), ylim=c(0, 0.8), xlab="chisq", ylab="f(chisq)")

> par(mfrow=c(2,2))

> for (i in 1:length(df_list)) {
+
+     curve(dchisq(x, df_list[i], ncp=0), add=F, xlim=c(0, 16), ylim=c(0, 0.8), xlab="chisq", ylab="f(chisq)", mai .... [TRUNCATED]
```



(2) χ^2-분포의 확률분포표

χ^2-분포는 자유도 n-1에 따라 형태가 다르므로 각각의 자유도 및 확률에 따른 χ^2의 값이 다음과 같이 주어져 있다.

자유도가 5일 때 $\chi^2_{0.1} = 9.236$인데, 이는 $P(\chi^2 \geqq 9.236) = 0.1$을 의미한다.

자유도	P=0.99	0.95	0.90	0.10	0.05	0.01
1	0.000157	0.00393	0.0158	2.706	3.841	6.635
2	0.0201	0.103	0.211	4.605	5.991	9.210
3	0.115	0.352	0.584	6.251	7.815	11.341
4	0.297	0.711	1.064	7.779	9.488	13.277
5	0.554	1.145	1.610	9.236	11.070	15.086
6	0.872	1.635	2.204	10.645	12.592	16.812
7	1.239	2.167	2.833	12.017	14.067	18.475
8	1.646	2.733	3.490	13.362	15.507	20.090
9	2.088	3.325	4.168	14.684	16.919	21.666
10	2.558	3.940	4.865	15.987	18.307	23.209
11	3.053	4.575	5.578	17.275	19.675	24.725
12	3.571	5.226	6.034	18.549	21.026	26.217
13	4.017	5.892	7.042	19.812	22.362	27.688
14	4.660	6.517	7.790	21.064	23.685	29.141
15	5.229	7.261	8.547	22.307	24.996	30.578
16	5.812	7.962	9.312	23.542	26.296	32.000
17	6.408	8.672	10.085	24.769	27.587	33.409
18	7.015	9.390	10.865	25.989	28.869	34.805
19	7.633	10.117	11.651	27.204	30.144	36.191
20	8.260	10.851	12.443	28.412	31.410	37.566
21	8.897	11.591	13.240	29.615	32.671	38.932
22	9.542	12.338	14.041	30.813	33.924	40.289
23	10.196	13.091	14.848	32.007	35.172	41.638
24	10.856	13.848	15.659	33.196	36.415	42.980
25	11.524	14.611	16.473	34.382	37.652	44.314
26	12.198	15.379	17.292	36.563	38.885	45.642
27	12.879	16.151	18.114	36.741	40.113	46.963
28	13.565	16.928	18.939	37.916	41.337	48.278
29	14.256	17.708	19.768	39.087	42.557	49.588
30	14.953	18.493	30.599	40.256	43.773	50.892

b1-ch4-13.R을 실행하면 위와 동일한 χ^2-분포표를 만들 수 있음을 확인할 수 있다.

b1-ch4-13.R의 실행결과

```
> df<-30

> chi1<-numeric(df)

> chi2<-numeric(df)

> chi3<-numeric(df)

> chi4<-numeric(df)

> chi5<-numeric(df)

> chi6<-numeric(df)

>  for(j in 1:df) {
+    chi1[j]<-qchisq(0.01,j)
+  }

> for(j in 1:df) {
+    chi2[j]<-qchisq(0.05,j)
+ }

> for(j in 1:df) {
+    chi3[j]<-qchisq(0.1,j)
+ }

> for(j in 1:df) {
+    chi4[j]<-qchisq(0.9,j)
+ }

> for(j in 1:df) {
+    chi5[j]<-qchisq(0.95,j)
+ }

> for(j in 1:df) {
+    chi6[j]<-qchisq(0.99,j)
```

```
+ }

> round((chi<-cbind(chi1,chi2,chi3,chi4,chi5, chi6)),digits=4)
          chi1     chi2     chi3     chi4     chi5     chi6
 [1,]  0.0002   0.0039   0.0158   2.7055   3.8415   6.6349
 [2,]  0.0201   0.1026   0.2107   4.6052   5.9915   9.2103
 [3,]  0.1148   0.3518   0.5844   6.2514   7.8147  11.3449
 [4,]  0.2971   0.7107   1.0636   7.7794   9.4877  13.2767
 [5,]  0.5543   1.1455   1.6103   9.2364  11.0705  15.0863
 [6,]  0.8721   1.6354   2.2041  10.6446  12.5916  16.8119
 [7,]  1.2390   2.1673   2.8331  12.0170  14.0671  18.4753
 [8,]  1.6465   2.7326   3.4895  13.3616  15.5073  20.0902
 [9,]  2.0879   3.3251   4.1682  14.6837  16.9190  21.6660
[10,]  2.5582   3.9403   4.8652  15.9872  18.3070  23.2093
[11,]  3.0535   4.5748   5.5778  17.2750  19.6751  24.7250
[12,]  3.5706   5.2260   6.3038  18.5493  21.0261  26.2170
[13,]  4.1069   5.8919   7.0415  19.8119  22.3620  27.6882
[14,]  4.6604   6.5706   7.7895  21.0641  23.6848  29.1412
[15,]  5.2293   7.2609   8.5468  22.3071  24.9958  30.5779
[16,]  5.8122   7.9616   9.3122  23.5418  26.2962  31.9999
[17,]  6.4078   8.6718  10.0852  24.7690  27.5871  33.4087
[18,]  7.0149   9.3905  10.8649  25.9894  28.8693  34.8053
[19,]  7.6327  10.1170  11.6509  27.2036  30.1435  36.1909
[20,]  8.2604  10.8508  12.4426  28.4120  31.4104  37.5662
[21,]  8.8972  11.5913  13.2396  29.6151  32.6706  38.9322
[22,]  9.5425  12.3380  14.0415  30.8133  33.9244  40.2894
[23,] 10.1957  13.0905  14.8480  32.0069  35.1725  41.6384
[24,] 10.8564  13.8484  15.6587  33.1962  36.4150  42.9798
[25,] 11.5240  14.6114  16.4734  34.3816  37.6525  44.3141
[26,] 12.1981  15.3792  17.2919  35.5632  38.8851  45.6417
[27,] 12.8785  16.1514  18.1139  36.7412  40.1133  46.9629
[28,] 13.5647  16.9279  18.9392  37.9159  41.3371  48.2782
[29,] 14.2565  17.7084  19.7677  39.0875  42.5570  49.5879
[30,] 14.9535  18.4927  20.5992  40.2560  43.7730  50.8922
```

8. t-분포

(1) t-분포의 모양

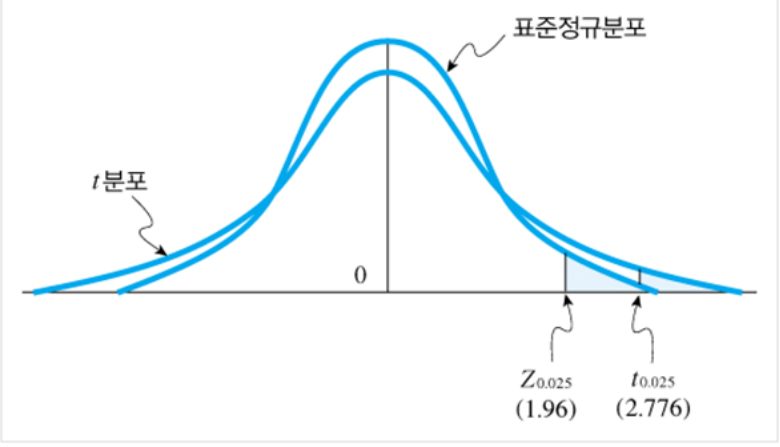
표준정규분포와 χ^2-분포에 의해 t-분포가 도출된다. $Z \sim N(0,1)$ 이고 $V \sim \chi^2(v)$ 이며 서로 독립이면, 다음의 확률변수 T는 자유도가 v인 t-분포에 따른다.

$$T = \frac{Z}{\sqrt{\frac{V}{v}}} \sim t(v)$$

한편, 다음의 확률변수 t는 자유도가 n-1인 t-분포에 따른다.

$$t = \frac{\overline{X} - \mu}{\frac{s}{\sqrt{n}}} \sim t_{n-1}$$

이 확률변수 t는 자유도가 무한히 커지면 표준정규분포에 접근하는데 표준정규분포와 자유도가 4인 t-분포를 비교한 <그림 4-4>를 보면 t-분포가 표준정규분포보다 꼬리부분의 확률이 조금 더 큰 것을 알 수 있다.

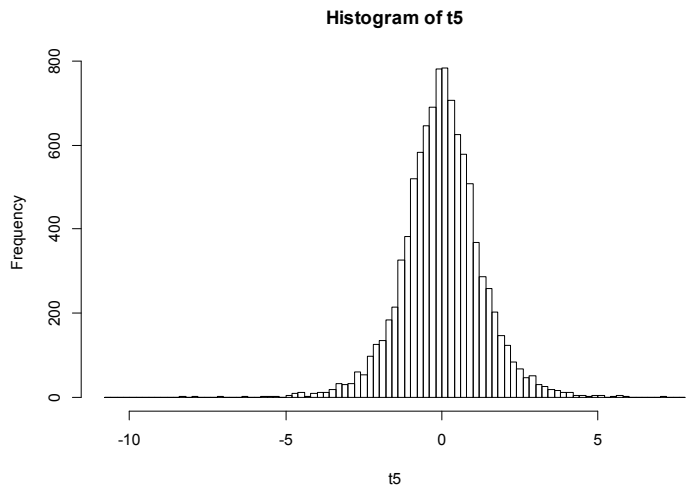


<그림 4-4> 표준정규분포와 자유도가 4인 t-분포의 비교

b1-ch4-14.R을 실행하면 자유도가 5인 t-분포를 그려주고 있는데 좌우 대칭의 분포로 정규분포와 유사함을 확인할 수 있다.

b1-ch4-14.R의 실행결과

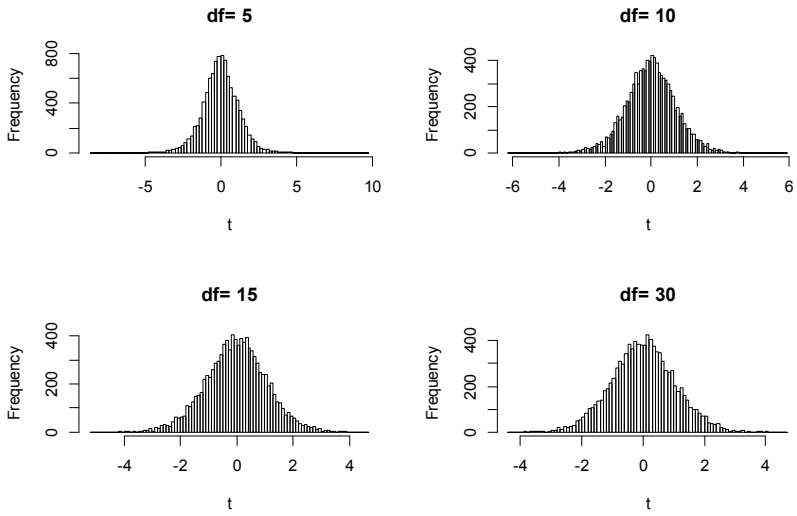
```
set.seed(12345)

n<-10000;

z<-rnorm(n,0,1)
z1<-rnorm(n,0,1)
z2<-rnorm(n,0,1)
z3<-rnorm(n,0,1)
z4<-rnorm(n,0,1)
z5<-rnorm(n,0,1)

chi5<-z1^2 + z2^2 + z3^2 + z4^2 + z5^2

sqchi5<-sqrt(chi5/5)

t5<-z/sqchi5

hist(t5, breaks = 100)
```



b1-ch4-15.R을 실행하면 각각 자유도가 5, 10, 15, 30인 t-분포를 그려주고 있는데 자유도에 따라 t-분포의 모양이 변하는 것을 확인할 수 있다.

b1-ch4-15.R의 실행결과

```
> n<-10000;

> df_list<-c(5,10,15,30)

> par(mfrow=c(2,2))

> for (i in 1:length(df_list)) {
+     hist(rt(n, df=df_list[i], ncp=0), breaks=100, xlab="t", main=paste("df=",
df_list[i]))
```

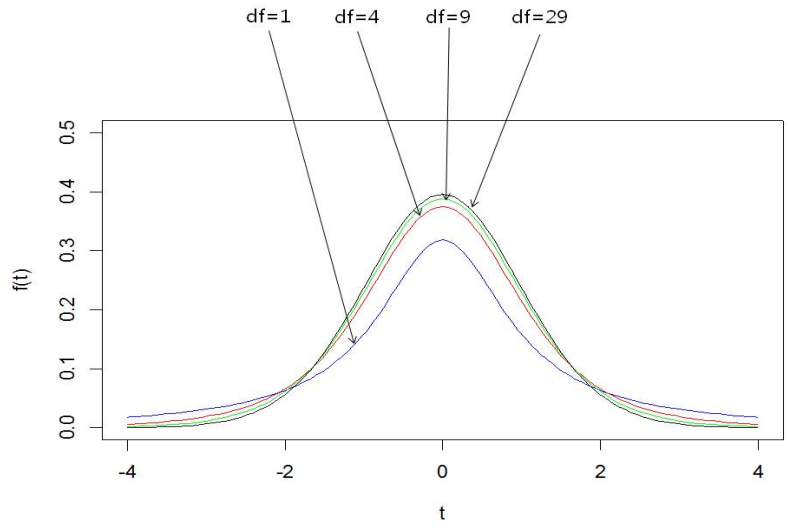


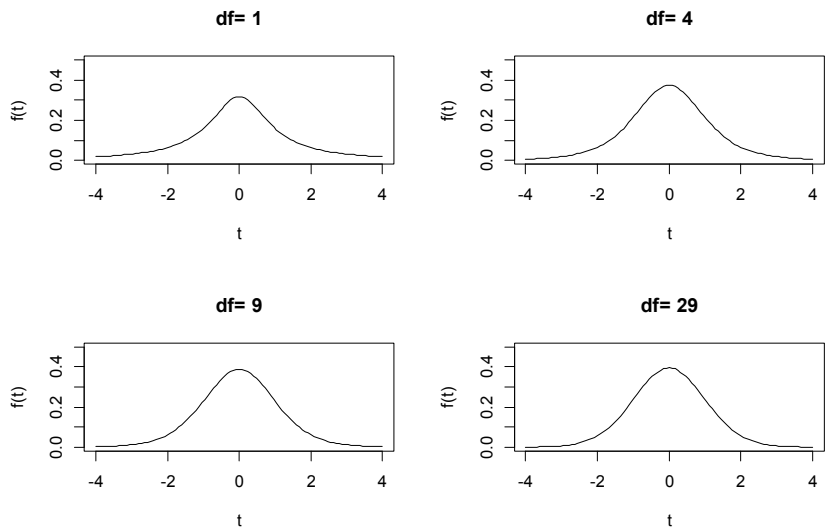
t-분포는 자유도가 모수이므로 자유도의 크기에 따라 분포의 형태가 달라진다.

b1-ch4-16.R을 실행하면 각각 자유도가 1, 4, 9, 29인 t-분포를 그려주고 있는데 자유도에 따라 t-분포의 모양이 변하는 것을 확인할 수 있으며, 자유도가 클수록 정규분포와 근사한 분포 형태를 갖는다.

b1-ch4-16.R의 실행결과

```
> n_list<-c(2,5,10,30) # 표본수(n)

> df_list<-n_list-1 # 자유도

> curve(dt(x, 1, ncp=0), add=T, col="blue", xlim=c(-4, 4), ylim=c(0, 0.5),
xlab="t", ylab="f(t)")

> curve(dt(x, 4, ncp=0), add=T, col="red", xlim=c(-4, 4), ylim=c(0, 0.5),
xlab="t", ylab="f(t)")

> curve(dt(x, 9, ncp=0), add=T, col="green", xlim=c(-4, 4), ylim=c(0, 0.5),
xlab="t", ylab="f(t)")

> curve(dt(x, 29, ncp=0), add=T, col="black", xlim=c(-4, 4), ylim=c(0, 0.5),
xlab="t", ylab="f(t)")

> par(mfrow=c(2,2))

> for (i in 1:length(df_list)) {
+
+     curve(dt(x, df_list[i], ncp=0), xlim=c(-4, 4), ylim=c(0, 0.5), xlab="t",
ylab="f(t)", main=paste("df=", df .... [TRUNCATED]
```





(2) t-분포의 확률분포표

t-분포는 자유도 n-1에 따라 분포의 형태가 다르므로 각각의 자유도 및 확률에 따른 t- 분포의 값이 다음과 같이 주어져 있다.

자유도가 5일 때 $t_{0.05} = 2.015$ 인데 이는 $P(T \geqq 2.015) = 0.05$ 를 의미한다.

v \ p	0.1	0.05	0.025	0.01	0.005
1	3.078	6.314	12.706	31.821	63.657
2	1.886	2.920	4.303	6.965	9.923
3	0.1638	2.353	3.182	4.541	5.841
4	1.533	2.132	2.776	3.747	4.604
5	1.476	2.015	2.571	3.365	4.032
6	1.440	1.943	2.447	3.143	3.707
7	1.415	1.895	2.365	2.998	3.499
8	1.397	1.860	2.306	2.896	3.355
9	1.383	1.833	2.262	2.821	3.250

b1-ch4-17.R을 실행하면 위와 동일한 t-분포표를 만들 수 있음을 확인할 수 있다.

b1-ch4-17.R의 실행결과

```
> t11<-rep(NA,9)

> t12<-rep(NA,9)

> t13<-rep(NA,9)

> t14<-rep(NA,9)

> t15<-rep(NA,9)

> for(i in 1:9) {
+    t11[i]<-qt(0.9, i)
+ }

> for(i in 1:9) {
+    t12[i]<-qt(0.95, i)
+ }

> for(i in 1:9) {
+    t13[i]<-qt(0.975,i)
+ }

> for(i in 1:9) {
+    t14[i]<-qt(0.99, i)
+ }

> for(i in 1:9) {
+    t15[i]<-qt(0.995, i)
+ }

> round((poi<-cbind(t11,t12,t13,t14,t15)), digits=3)
         t11   t12    t13    t14    t15
 [1,] 3.078 6.314 12.706 31.821 63.657
```

```
[2,] 1.886 2.920 4.303 6.965 9.925
[3,] 1.638 2.353 3.182 4.541 5.841
[4,] 1.533 2.132 2.776 3.747 4.604
[5,] 1.476 2.015 2.571 3.365 4.032
[6,] 1.440 1.943 2.447 3.143 3.707
[7,] 1.415 1.895 2.365 2.998 3.499
[8,] 1.397 1.860 2.306 2.896 3.355
[9,] 1.383 1.833 2.262 2.821 3.250
```

9. F-분포

(1) F-분포의 모양

두 개의 독립적인 χ^2-분포에 의해 F-분포가 도출된다. $X_1 \sim \chi^2(v_1)$ 이고 $X_2 \sim \chi^2(v_2)$ 이며 서로 독립이면, 다음의 확률변수 F는 분자 및 분모의 자유도가 각각 v_1, v_2 인 F-분포에 따른다.

$$F = \frac{\frac{X_1}{v_1}}{\frac{X_2}{v_2}} \sim F_{(v_1, v_2)}$$

한편, 다음의 확률변수 F는 분자 및 분모의 자유도가 각각 $n_1 - 1, n_2 - 1$ 인 F-분포에 따른다.

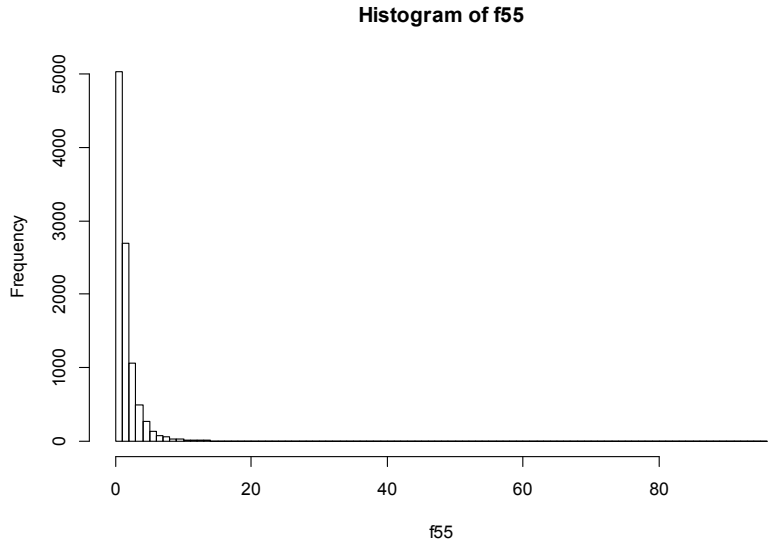
$$F = \frac{\frac{\chi_{n_1-1}}{n_1-1}}{\frac{\chi_{n_2-1}}{n_2-1}} = \frac{\frac{(n_1-1)s_1^2}{(n_1-1)\sigma_1^2}}{\frac{(n_2-1)s_2^2}{(n_2-1)\sigma_2^2}} = \frac{\frac{s_1^2}{\sigma_1^2}}{\frac{s_2^2}{\sigma_2^2}} \sim F_{(n_1-1, n_2-1)}$$

이 확률변수 F는 자유도가 분자 및 분모의 자유도가 같아지면서 커질수록 좌우대칭 분포와 비슷하게 된다.

b1-ch4-18.R을 실행하면 분자 및 분모의 자유도가 각각 5인 F-분포를 그려주고 있는데 좌우 비대칭의 분포임을 확인할 수 있다.

b1-ch4-18.R의 실행결과

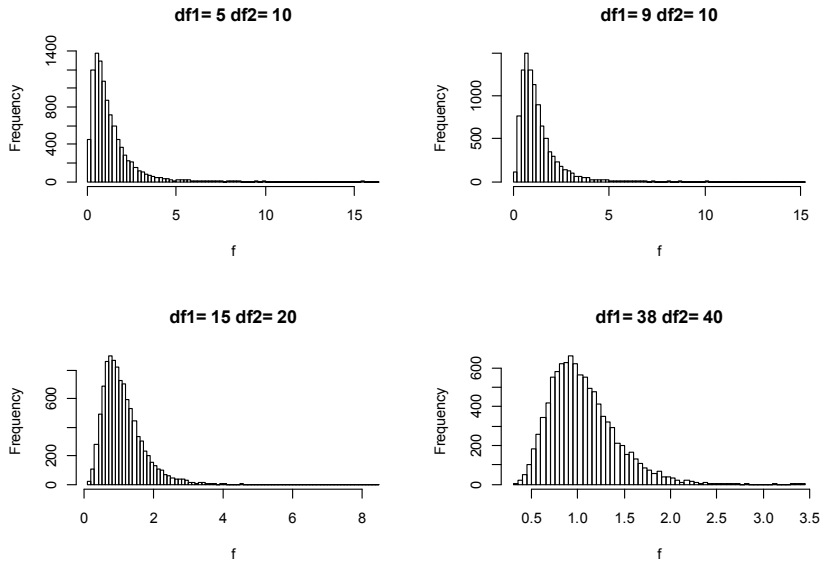
```
> set.seed(12345)
> n<-10000;
> z1<-rnorm(n,0,1)
> z2<-rnorm(n,0,1)
> z3<-rnorm(n,0,1)
> z4<-rnorm(n,0,1)
> z5<-rnorm(n,0,1)
> z6<-rnorm(n,0,1)
> z7<-rnorm(n,0,1)
> z8<-rnorm(n,0,1)
> z9<-rnorm(n,0,1)
> z10<-rnorm(n,0,1)
> chi15<-z1^2 + z2^2 + z3^2 + z4^2 + z5^2
> chi25<-z6^2 + z7^2 + z8^2 + z9^2 + z10^2
> f55<-(chi15/5)/(chi25/5)
> hist(f55, breaks = 100)
```



b1-ch4-19.R을 실행하면 분자 및 분모의 자유도가 각각 (5,10), (9,10), (15,20), (38, 40)인 F-분포를 그려주고 있는데 자유도에 따라 F-분포의 모양이 변하는 것을 확인할 수 있다.

b1-ch4-19.R의 실행결과

```
> set.seed(12345)

> n<-10000;

> df1_list<-c(5,9,15,38)

> df2_list<-c(10,10,20,40)

> par(mfrow=c(2,2))

> for (i in 1:length(df1_list)) {
+     hist(rf(n, df1=df1_list[i], df2=df2_list[i], ncp=0), breaks=100, xlab="f",
main=paste("df1=", df1_list[i], "df2= ..." ... [TRUNCATED]
```

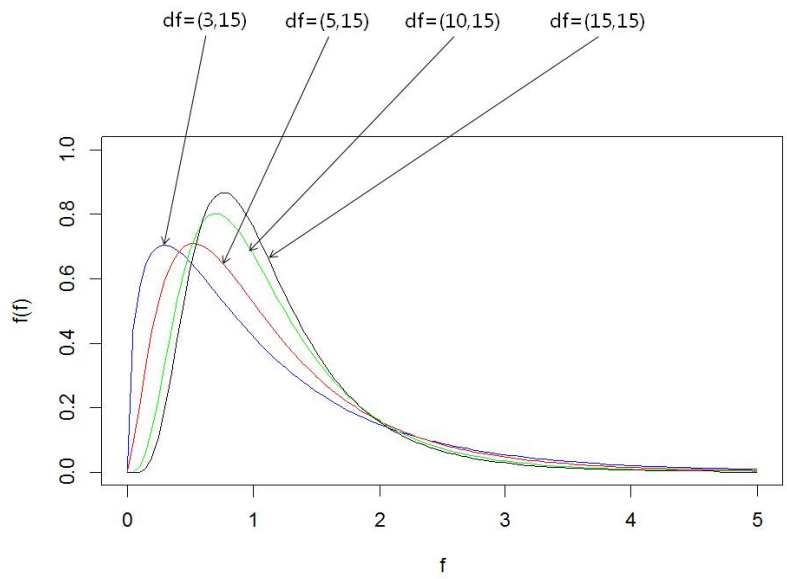


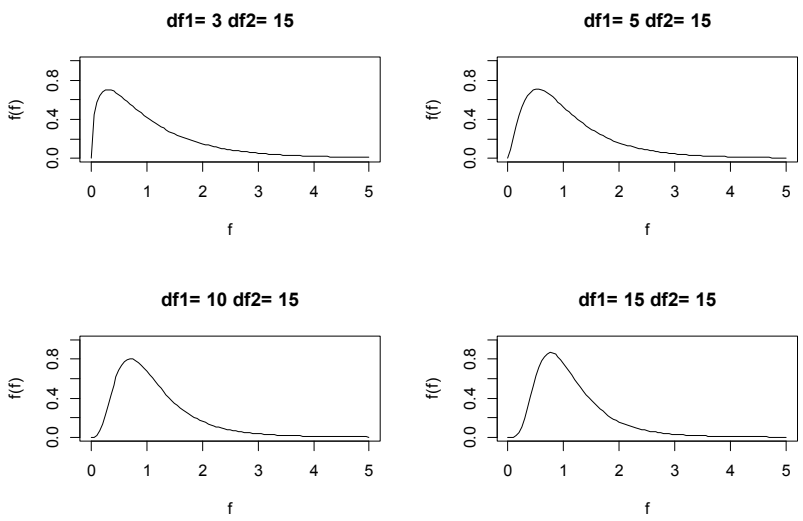
b1-ch4-20.R을 실행하면 자유도가 각각 (3,15), (5,15), (10,15), (15,15)인 F-분포를 그려주고 있는데 분자 및 분모의 자유도가 같아질수록 좌우대칭에 가까운 분포로 변함을 확인할 수 있다.

b1-ch4-20.R의 실행결과

```
> curve(df(x, 3, 15, ncp=0), add=T, col="blue", xlim=c(0,5), ylim=c(0,1), xlab="f", ylab="f(f)")

> curve(df(x, 5, 15, ncp=0), add=T, col="red", xlim=c(0,5), ylim=c(0,1), xlab="f", ylab="f(f)")

> curve(df(x, 10, 15, ncp=0), add=T, col="green", xlim=c(0,5), ylim=c(0,1), xlab="f", ylab="f(f)")

> curve(df(x, 15, 15, ncp=0), add=T, col="black", xlim=c(0,5), ylim=c(0,1), xlab="f", ylab="f(f)")

> df1_list<-c(3,5,10,15)
```

```
> df2_list<-c(15,15,15,15)

> par(mfrow=c(2,2))

> for (i in 1:length(df1_list)) {
+         curve(df(x, df1=df1_list[i], df2=df2_list[i], ncp=0), xlim=c(0,5),
ylim=c(0,1), xlab="f", ylab="f(f)", main=past .... [TRUNCATED]
```





(2) F-분포의 확률분포표

F-분포는 분자의 자유도(v_1) 및 분모의 자유도(v_2)에 따라 분포의 형태가 다르므로 각각의 자유도에 따른 $\alpha = 0.05$ 에 대한 F값이 다음과 같이 주어져 있다

분자의 자유도가 7이고 분모의 자유도가 9인 경우 $F_{0.05} = 3.29$ 이다.

v_2 \ v_1	1	2	3	4	5	6	7	8	9	10
1	161.45	199.50	215.71	224.58	230.16	233.99	236.77	238.88	240.54	241.88
2	18.51	19.00	19.16	19.25	19.30	19.33	19.35	19.37	19.38	19.40
3	10.13	9.55	9.28	9.12	9.01	8.94	8.89	8.85	8.81	8.76
4	7.71	6.94	6.59	6.39	6.26	6.16	6.09	6.04	6.00	5.96
5	6.61	5.79	5.41	5.19	5.05	4.95	4.48	4.82	4.77	4.74
6	5.99	4.74	7.35	4.12	3.94	3.87	3.79	3.73	3.68	3.64
7	5.59	4.74	4.35	4.12	3.97	3.87	3.79	3.73	3.68	3.64
8	5.32	4346	4.07	3.84	3.69	3.58	3.50	3.44	3.39	3.35
9	5.12	4.26	3.86	3.63	3.48	3.37	3.29	3.23	3.18	3.14
10	4.96	4.10	3.71	3.48	3.33	3.22	3.14	3.07	3.02	2.98

b1-ch4-21.R을 실행하면 위와 동일한 F-분포표를 만들 수 있음을 확인할 수 있다.

b1-ch4-21.R의 실행결과

```
> f11<-rep(NA,10)

> f12<-rep(NA,10)

> f13<-rep(NA,10)

> f14<-rep(NA,10)

> f15<-rep(NA,10)

> f16<-rep(NA,10)
```

```
> f17<-rep(NA,10)

> f18<-rep(NA,10)

> f19<-rep(NA,10)

> f110<-rep(NA,10)

> for(i in 1:10) {
+     f11[i]<-qf(0.95, 1, i)
+ }

> for(i in 1:10) {
+     f12[i]<-qf(0.95, 2, i)
+ }

> for(i in 1:10) {
+     f13[i]<-qf(0.95, 3, i)
+ }

> for(i in 1:10) {
+     f14[i]<-qf(0.95, 4, i)
+ }

> for(i in 1:10) {
+     f15[i]<-qf(0.95, 5, i)
+ }

> for(i in 1:10) {
+     f16[i]<-qf(0.95, 6, i)
+ }

> for(i in 1:10) {
+     f17[i]<-qf(0.95, 7, i)
+ }
```

```
> for(i in 1:10) {
+    f18[i]<-qf(0.95, 8, i)
+ }

> for(i in 1:10) {
+    f19[i]<-qf(0.95, 9, i)
+ }

> for(i in 1:10) {
+    f110[i]<-qf(0.95, 10, i)
+ }

> round((poi<-cbind(f11,f12,f13,f14,f15,f16,f17,f18,f19,f110)), digits=2)
       f11    f12    f13    f14    f15    f16    f17    f18    f19    f110
 [1,] 161.45 199.50 215.71 224.58 230.16 233.99 236.77 238.88 240.54 241.88
 [2,]  18.51  19.00  19.16  19.25  19.30  19.33  19.35  19.37  19.38  19.40
 [3,]  10.13   9.55   9.28   9.12   9.01   8.94   8.89   8.85   8.81   8.79
 [4,]   7.71   6.94   6.59   6.39   6.26   6.16   6.09   6.04   6.00   5.96
 [5,]   6.61   5.79   5.41   5.19   5.05   4.95   4.88   4.82   4.77   4.74
 [6,]   5.99   5.14   4.76   4.53   4.39   4.28   4.21   4.15   4.10   4.06
 [7,]   5.59   4.74   4.35   4.12   3.97   3.87   3.79   3.73   3.68   3.64
 [8,]   5.32   4.46   4.07   3.84   3.69   3.58   3.50   3.44   3.39   3.35
 [9,]   5.12   4.26   3.86   3.63   3.48   3.37   3.29   3.23   3.18   3.14
[10,]   4.96   4.10   3.71   3.48   3.33   3.22   3.14   3.07   3.02   2.98
```

제 5 장

표본분포

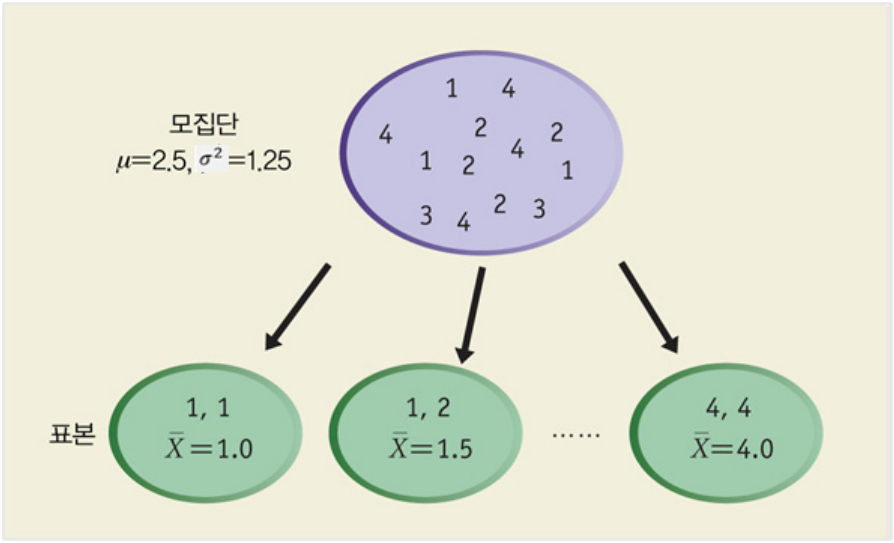
1. 표본평균의 표본분포
2. 중심극한정리
3. 표본분산의 표본분포

제5장 표본분포

1. 표본평균의 표본분포

모집단에서 일정한 크기의 모든 가능한 표본을 추출하였을 때 그 모든 표본으로부터 계산된 통계량 즉, 표본평균 및 표본분산의 확률분포를 표본분포(sampling distribution)이라고 한다.

예를 들어, J제약회사는 많은 종류의 신약을 개발하였다. 이 제약회사가 신약을 개발하기 위해서 1,2,3 혹은 4년의 시간이 걸렸으며 각각의 발생확률은 동등하다고 가정하자. 이때 신약의 평균 개발기간인 모집단의 평균 $\mu = 2.5$ 및 분산 $\sigma^2 = 1.25$ 로 계산된다.



<그림 5-1> 모집단으로부터 표본크기(n)가 2인 임의표본추출

표본크기 2의 모든 가능한 표본과 표본평균은 <표 5-1>과 같고, 표본평균의 표본분포는 <표 5-2>와 같다.

<표 5-1> 표본크기 2인 표본 및 표본평균

표본	표본평균($\overline{X}$)	표본	표본평균($\overline{X}$)
(1,1)	1.0	(3,1)	2.0
(1,2)	1.5	(3,2)	2.5
(1,3)	2.5	(3,3)	3.0
(1,4)	2.5	(3,4)	3.5
(2,1)	1.5	(4,1)	2.5
(2,2)	2.0	(4,2)	3.0
(2,3)	2.5	(4,3)	3.5
(2,4)	3.0	(4,4)	4.0

<표 5-2> 표본평균의 표본분포

$\overline{X}$	1	1.5	2.0	2.5	3.0	3.5	4.0
P($\overline{X}$)	1/16	2/16	3/16	4/16	3/16	2/16	1/16

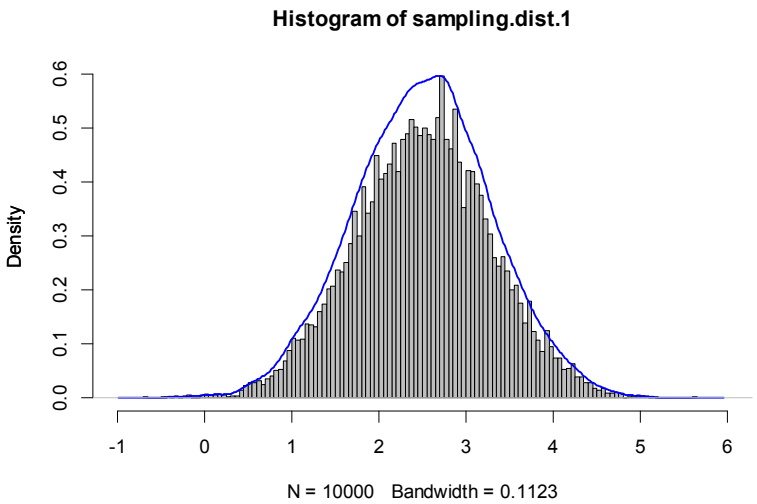
표본평균의 평균을 계산해 보면 2.5로써 모평균과 2.5로 같지만 표본평균의 분산은 0.625로써 모분산과 같지 않으며, 표본평균의 분산은 모분산의 1/2이 된다.

b1-ch5-1.R을 실행하면 표본평균의 평균은 2.502412, 분산은 0.6193981로써 모집단의 이론적인 평균과 표준편차에 근접함을 알 수 있다.

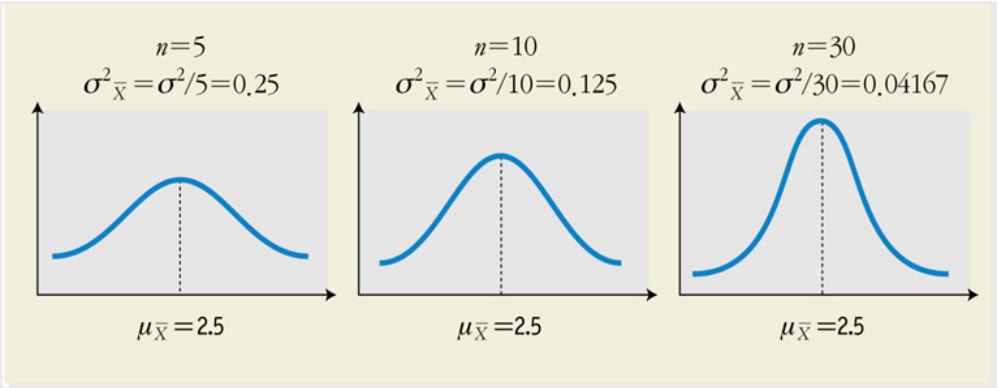
b1-ch5-1.R의 실행결과

```
> sampling.dist.1<-NULL
> for(sample.count in 1:10000){
+     set.seed(sample.count)
+     sample.mean.1<-mean(rnorm(2,2.5,1.118))
+     sampling.dist.1<-c(sampling.dist.1, sample.mean.1)
+ }
> mean(sampling.dist.1)
[1] 2.502412
> var(sampling.dist.1)
[1] 0.6193981
>
>   hist(sampling.dist.1,       freq=F,    col="grey",    xlab="",    xlim=c(-1,    6),
```

```
breaks = 100)
> par(new = T)
> plot(density(sampling.dist.1), axes = F, main = "", xlim = c(-1, 6), lwd = 2,
col = "blue")
```



표본크기에 따른 표본평균의 표본분포를 살펴보면 <그림 5-2>에 나타나 있듯이 표본평균의 평균은 표본크기에 관계없이 모평균과 동일하고, 표본평균의 분산은 모분산을 표본크기로 나눈 값과 같으므로 표본크기가 커짐에 따라 표본평균의 분산은 작아진다.



<그림 5-2> 표본크기(n)에 따른 표본평균의 표본분포

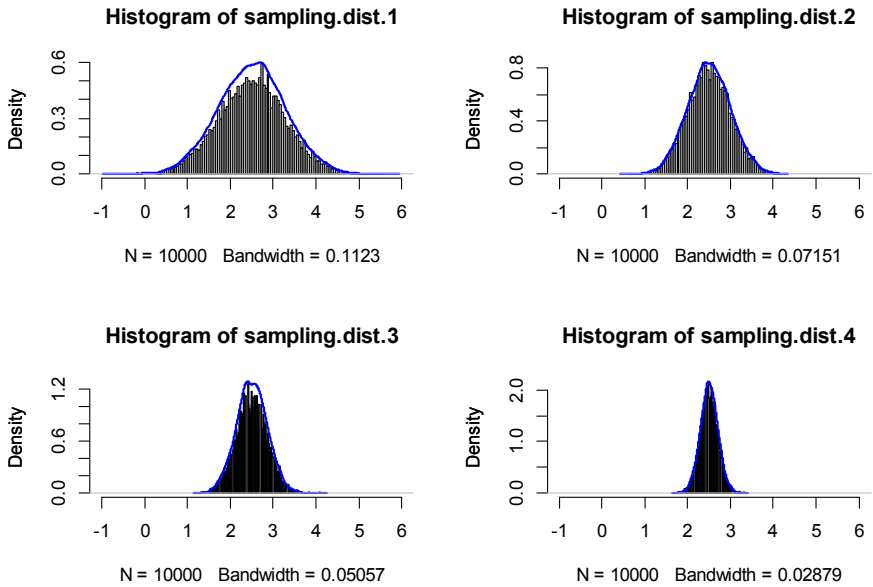
b1-ch5-2.R을 실행하면 <그림 5-1>의 모집단에서 표본크기를 다르게 할 경우 표본평균의 분포를 보여주고 있는데 표본평균의 평균은 모평균과 동일하지만, 표본평균의 분산은 모분산을 표본크기로 나눈 값이 되는 것을 확인할 수 있다.

b1-ch5-2.R의 실행결과

```
> sampling.dist.1<-NULL

> for(sample.count in 1:10000){
+     set.seed(sample.count)
+     sample.mean.1<-mean(rnorm(2,2.5,1.118))
+     sampling.dist.1<-c(sampling.dist.1, sample. .... [TRUNCATED]

> mean(sampling.dist.1)
[1] 2.502412

> var(sampling.dist.1)
[1] 0.6193981

> table(round(sampling.dist.1))

  -1    0    1    2    3    4    5    6
   1   53  968 3913 4055  961   48    1

> sampling.dist.2<-NULL

> for(sample.count in 1:10000){
+     set.seed(sample.count)
+     sample.mean.2<-mean(rnorm(5,2.5,1.118))
+     sampling.dist.2<-c(sampling.dist.2, sample. .... [TRUNCATED]

> mean(sampling.dist.2)
[1] 2.498498

> var(sampling.dist.2)
[1] 0.2513623
```

```
> table(round(sampling.dist.2))

   1    2    3    4
 225 4805 4720  250

> sampling.dist.3<-NULL

> for(sample.count in 1:10000){
+     set.seed(sample.count)
+     sample.mean.3<-mean(rnorm(10,2.5,1.118))
+     sampling.dist.3<-c(sampling.dist.3, sample .... [TRUNCATED]

> mean(sampling.dist.3)
[1] 2.500606

> var(sampling.dist.3)
[1] 0.1256997

> table(round(sampling.dist.3))

   1    2    3    4
  10 5008 4955   27

> sampling.dist.4<-NULL

> for(sample.count in 1:10000){
+     set.seed(sample.count)
+     sample.mean.4<-mean(rnorm(30,2.5,1.118))
+     sampling.dist.4<-c(sampling.dist.4, sample .... [TRUNCATED]

> mean(sampling.dist.4)
[1] 2.502484

> var(sampling.dist.4)
[1] 0.0420068

> table(round(sampling.dist.4))
```

```
   2    3
4992 5008

> par(mfrow=c(2,2))

> hist(sampling.dist.1, freq=F, col="grey", xlab="", xlim=c(-1, 6), breaks=100)

> par(new=T)

> plot(density(sampling.dist.1), axes=F, main="", xlim=c(-1, 6), lwd=2, col="blue")

> hist(sampling.dist.2, freq=F, col="grey", xlab="", xlim=c(-1, 6), breaks=100)

> par(new=T)

> plot(density(sampling.dist.2), axes=F, main="", xlim=c(-1, 6), lwd=2, col="blue")

> hist(sampling.dist.3, freq=F, col="grey", xlab="", xlim=c(-1, 6), breaks=100)

> par(new=T)

> plot(density(sampling.dist.3), axes=F, main="", xlim=c(-1, 6), lwd=2, col="blue")

> hist(sampling.dist.4, freq=F, col="grey", xlab="", xlim=c(-1, 6), breaks=100)

> par(new=T)

> plot(density(sampling.dist.4), axes=F, main="", xlim=c(-1, 6), lwd=2, col="blue")
```



2. 중심극한정리

평균이 μ이고, 분산이 σ^2인 확률분포로부터 표본크기가 n인 확률표본(X_1, X_2 ,..., X_n)을 추출할 때, 표본평균 $\overline{X}= \frac{1}{n}\sum_{i=1}^{n} X_i$ 는 n이 클수록 평균이 μ이고, 분산이 $\frac{\sigma^2}{n}$인 정규분포와 근사한 분포를 갖는다.

즉, $\overline{X}$의 분포는 $\overline{X} \sim N(\mu, \frac{\sigma^2}{n})$ 과 같이 나타낼 수 있는데 이를 중심극한정리(central limit theorem)라고 한다.

관찰된 자료의 모집단이 정규분포에 따를 경우 관찰자료 역시 정규분포에 따르고, 관찰된 자료의 모집단이 실제로 정규분포가 아니면 관찰 자료 역시 정규분포에 따르지 않는다.

그러나 관찰된 자료의 모집단이 실제로 정규분포가 아닌 경우에도 중심극한정리에 의해 표준정규분포를 이용한 추정량의 근사확률을 구할 수 있다.

(실험 1)

확률변수 X가 균등분포 U(0,1)을 따른다고 할 때 X~U(0,1)이며, X는 0과 1사이에서 균등한 분포를 갖는 연속형 확률변수이다. 이러한 확률분포로부터 표본크기 n이 11인 확률표본을 1,000개 추출하는 실험을 해 보자.

1,000개 표본으로부터 구한 표본평균 $\overline{X}_1, \overline{X}_2, ..., \overline{X}_{1000}$ 을 이용하여 히스토그램을 그려보면 균등분포로부터 구한 표본평균의 분포가 정규분포와 근사한 분포를 갖는다는 것을 알 수 있다.

X~U(0,1)에서 X의 평균은 1/2, 분산은 1/12이므로 n=11인 경우 표본평균 $\overline{X}$의 평균은 1/2, 분산은 1/(11x12)이 됨을 다음과 같이 확인할 수 있다.

즉, 표본평균 $\overline{X}$는 근사적으로 $N(\frac{1}{2}, \frac{1}{132})$ 을 따르는데 1,000개 표본의 표본평균과 분산은 각각 0.5019 및 0.0078로서 모집단의 이론적인 평균과 표준편차에 근접함을 알 수 있다

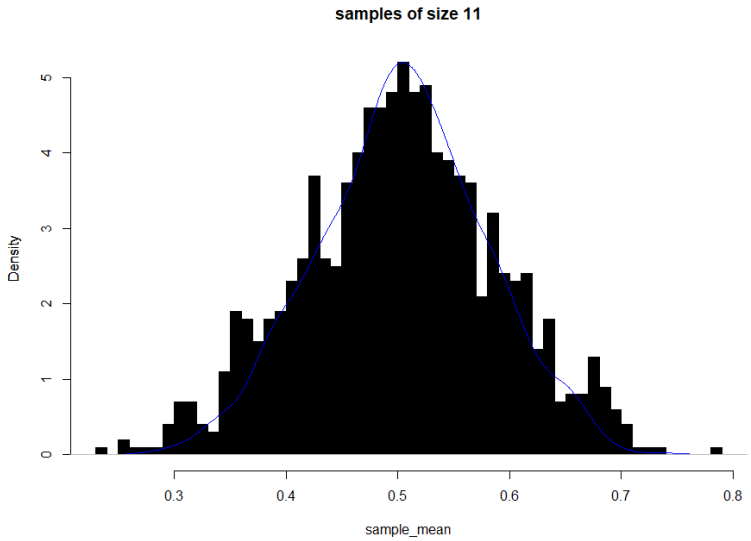
```
> (mean(sample_mean))
[1] 0.5019778

> (var(sample_mean))
[1] 0.007837079
```

b1-ch5-3.R을 실행하면 위의 (실험 1)을 확인할 수 있다.

b1-ch5-3.R의 실행결과

```
> set.seed(23456789)

> sample_size <- 11

> min <- 0

> max <- 1

> n_rep <- 1000
```

```
> sample_mean <- rep(NA, n_rep)

> sample_var <- rep(NA, n_rep)

> #graphics.off()
> #par(mfrow=c(1,2))
>
> for (i in 1:n_rep) {
+         my_sample <- runif(sample_size,min,max)
+         sample_mean[i] <- mean(my_samp .... [TRUNCATED]

> (mean(sample_mean))
[1] 0.5019778

> (var(sample_mean))
[1] 0.007837079

> hist(sample_mean, breaks=40, prob=T, main=paste("samples of size 11"),
col="black")

> par(new=T)

> plot(density(sample_mean), xlab="", axes=F, main="", col="blue")
```



(실험 2)

확률변수 X가 평균이 10, 표준편차가 2인 정규분포를 따른다고 할 때 이러한 확률분포로부터 표본크기가 각각 5, 10, 20, 30인 확률표본을 1,000개 추출하는 실험을 해 보자.

표본평균 $\overline{X}$는 표본크기에 따라 1,000개 표본의 표본평균과 분산은 각각 다음과 같은데 모집단의 이론적인 평균과 분산에 근접함을 알 수 있다.

구분	n=5	n=10	n=20	n=30
표본평균의 평균	10.00494	10.00259	9.998591	9.993856
표본평균의 분산	0.7818611	0.3992823	0.2039642	0.1347668

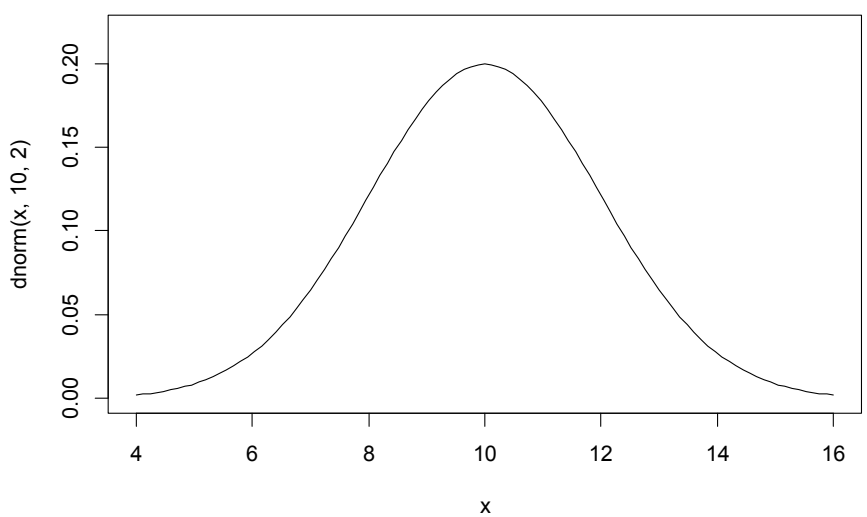
b1-ch5-4.R을 실행하면 위의 (실험 2)를 확인할 수 있다

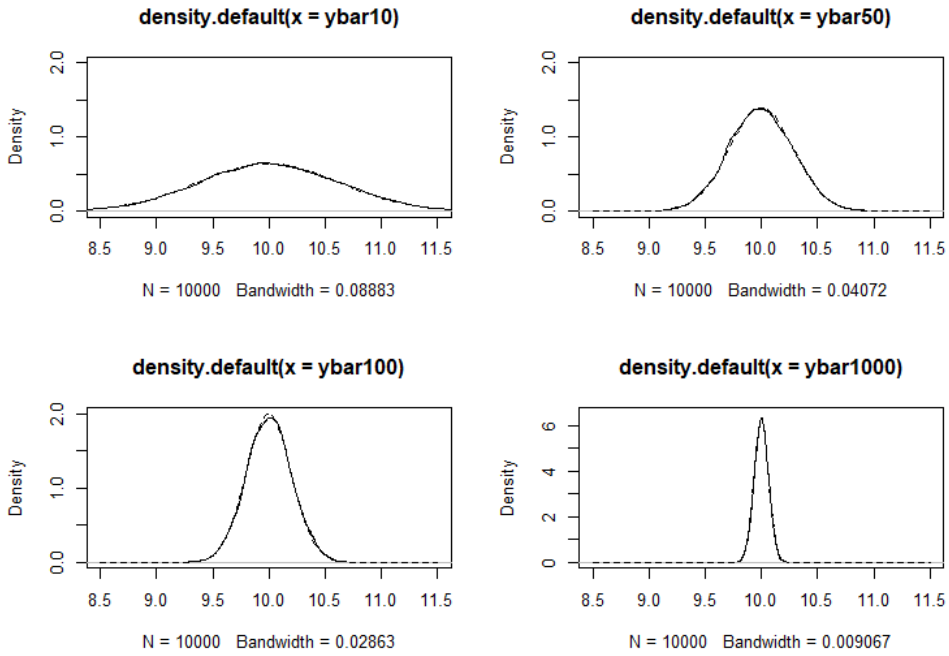
b1-ch5-4.R의 실행결과

```
> set.seed(123456)

> curve(dnorm(x,10,2), xlim = c(4, 16), ylim = c(0.0, 0.22))

> par(mfrow=c(2,2))

> ybar5<-numeric(10000)

> for(j in 1:10000) {
+     sample<-rnorm(5,10,2)
+     ybar5[j] <-mean(sample)
+ }

> mean(ybar5)
[1] 10.00494

> var(ybar5)
[1] 0.7818611

> plot(density(ybar5), xlim = c(7, 13), ylim = c(0.0, 0.5))
```

```
> curve(dnorm(x,10,sqrt(0.78186)), add=T, lty=2)

> ybar10<-numeric(10000)

> for(k in 1:10000) {
+     sample<-rnorm(10,10,2)
+     ybar10[k] <-mean(sample)
+ }

> mean(ybar10)
[1] 10.00259

> var(ybar10)
[1] 0.3992823

> plot(density(ybar10),xlim = c(7.5, 12.5), ylim = c(0.0, 0.8))

> curve(dnorm(x,10,sqrt(0.39928)), add=T, lty=2)

> ybar20<-numeric(10000)

> for(m in 1:10000) {T
+     sample<-rnorm(20,10,2)
+     ybar20[m] <-mean(sample)
+ }

> mean(ybar20)
[1] 9.998591

> var(ybar20)
[1] 0.2039642

> plot(density(ybar20),xlim = c(8.5, 11.5), ylim = c(0.0, 1.0))

> curve(dnorm(x,10,sqrt(0.20396)), add=T, lty=2)
```

```
> ybar30<-numeric(10000)

> for(n in 1:10000) {
+     sample<-rnorm(30,10,2)
+     ybar30[n] <-mean(sample)
+ }

> mean(ybar30)
[1] 9.993856

> var(ybar30)
[1] 0.1347668

> plot(density(ybar30),xlim = c(8.5, 11.5), ylim = c(0.0, 1.2))

> curve(dnorm(x,10,sqrt(0.13477)), add=T, lty=2)
```





(실험 3)

확률변수 X가 자유도가 1인 χ^2-분포를 따른다고 할 때(따라서 평균은 1, 분산은 2가 됨) 이러한 확률분포로부터 표본크기가 각각 2, 10, 20, 30인 확률표본을 1,000개 추출하는 실험을 해 보자.

표본크기에 따라 1,000개 표본의 표본평균의 평균과 분산은 각각 다음과 같은데 모집단의 이론적인 평균과 분산에 근접함을 알 수 있다.

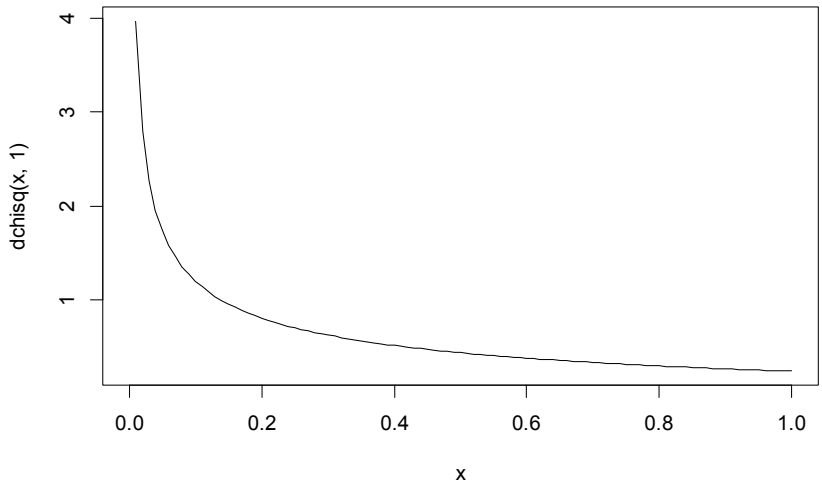
구분	n = 2	n = 10	n = 20	n = 30
표본평균의 평균	0.9873457	1.002666	0.9988942	0.9961374
표본평균의 분산	0.9841301	0.2016694	0.1021074	0.06725091

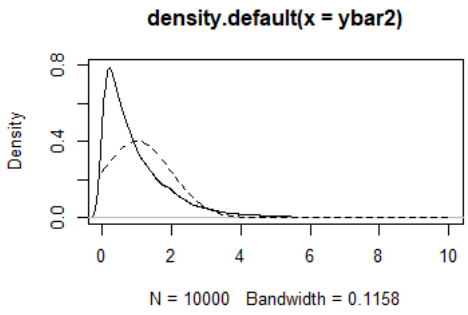
b1-ch5-5.R을 실행하면 위의 (실험 3)을 확인할 수 있다

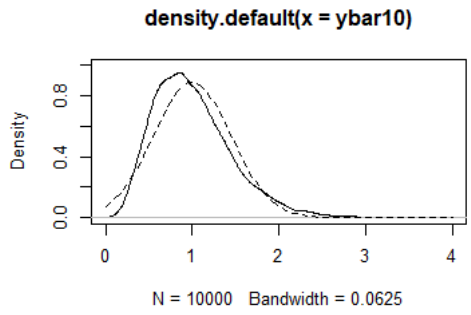
b1-ch5-5.R의 실행결과

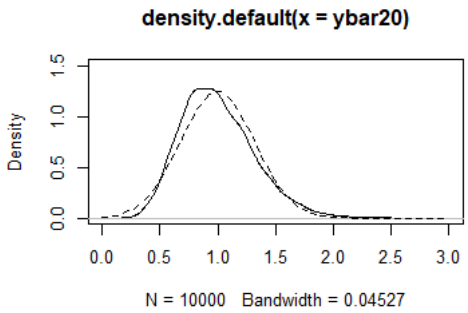
```
> set.seed(123456)

> curve(dchisq(x,1))

> par(mfrow=c(2,2))

> ybar2<-numeric(10000)

> for(j in 1:10000) {
+     sample<-rchisq(2,1)
+     ybar2[j] <-mean(sample)
+ }

> mean(ybar2)
[1] 0.9873457

> var(ybar2)
[1] 0.9841301

> plot(density(ybar2), xlim = c(0, 10), ylim = c(0.0, 0.8))

> curve(dnorm(x,1,sqrt(0.98413)), add=T, lty=2)

> ybar10<-numeric(10000)

> for(k in 1:10000) {
+     sample<-rchisq(10,1)
+     ybar10[k] <-mean(sample)
+ }

> mean(ybar10)
[1] 1.002666

> var(ybar10)
[1] 0.2016694
```

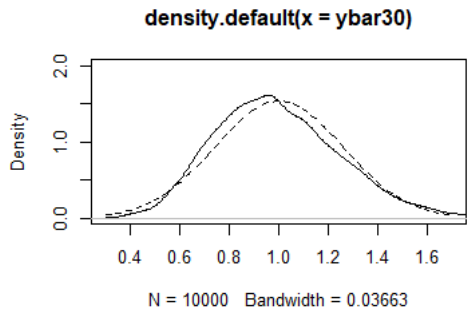
```
> plot(density(ybar10),xlim = c(0, 4), ylim = c(0.0, 1))

> curve(dnorm(x,1,sqrt(0.20167)), add=T, lty=2)

> ybar20<-numeric(10000)

> for(n in 1:10000) {
+     sample<-rchisq(20,1)
+     ybar20[n] <-mean(sample)
+ }

> mean(ybar20)
[1] 0.9988942

> var(ybar20)
[1] 0.1021074

> plot(density(ybar20),xlim = c(0, 3), ylim = c(0.0, 1.5))

> curve(dnorm(x,1,sqrt(0.10241)), add=T, lty=2)

> ybar30<-numeric(10000)

> for(m in 1:10000) {
+     sample<-rchisq(30,1)
+     ybar30[m] <-mean(sample)
+ }

> mean(ybar30)
[1] 0.9961374

> var(ybar30)
[1] 0.06725091

> plot(density(ybar30),xlim = c(0.3, 1.7), ylim = c(0.0, 2.0))

> curve(dnorm(x,1,sqrt(0.06725)), add=T, lty=2)
```


dchisq(x, 1)
x
0.0
0.2
0.4
0.6
0.8
1.0
1
2
3
4


density.default(x = ybar2)
Density
N = 10000 Bandwidth = 0.1158


density.default(x = ybar10)
Density
N = 10000 Bandwidth = 0.0625


density.default(x = ybar20)
Density
N = 10000 Bandwidth = 0.04527


density.default(x = ybar30)
Density
N = 10000 Bandwidth = 0.03663

(실험 4)

확률변수 X가 균등분포 U(0,1)을 따른다고 할 때 즉, X~U(0,1)일 때(따라서 평균은 0.5, 분산은 0.083333이 됨) 이러한 확률분포로부터 표본크기가 각각 2, 10, 20, 30인 확률표본을 1,000개 추출하는 실험을 해 보자.

표본크기에 따라 1,000개 표본의 표본평균의 평균과 분산은 각각 다음과 같은데 모집단의 이론적인 평균과 분산에 근접함을 알 수 있다.

구분	n=2	n=10	n=20	n=30
표본평균의 평균	0.5013918	0.4987733	0.5003994	0.4996023
표본평균의 분산	0.04192942	0.008292283	0.004107226	0.002834952

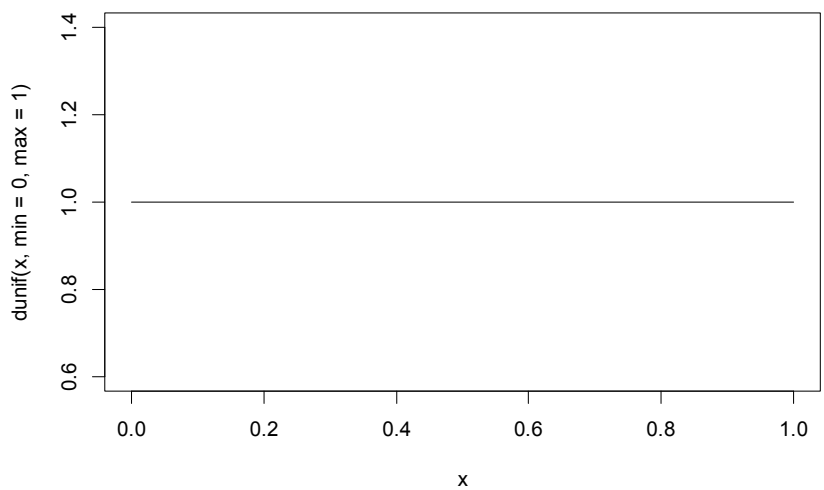
b1-ch5-6.R을 실행하면 위의 (실험 4)를 확인할 수 있다

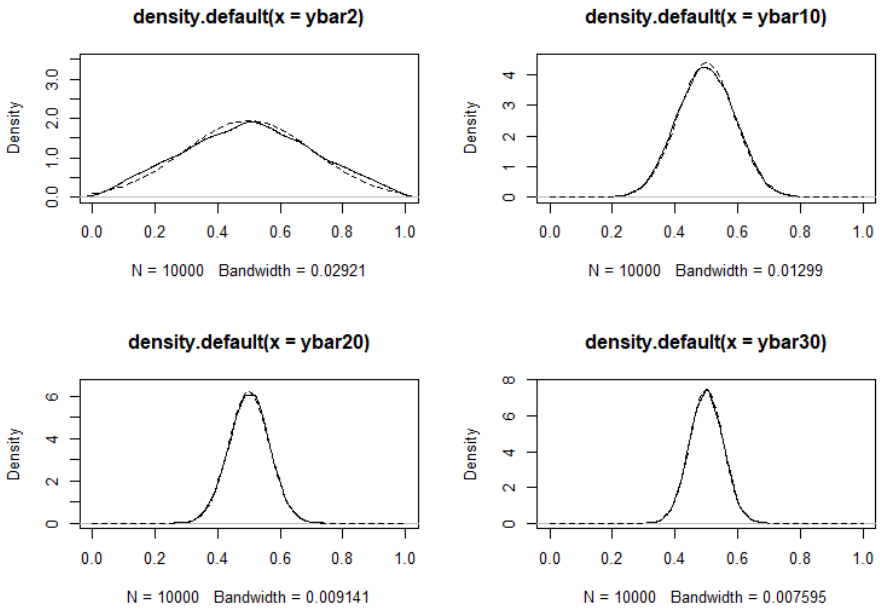
b1-ch5-6.R의 실행결과

```
> set.seed(123456)

> curve(dunif(x,min=0, max=1))

> par(mfrow=c(2,2))

> ybar2<-numeric(10000)

> for(j in 1:10000) {
+     sample<-runif(2,min=0, max=1)
+     ybar2[j] <-mean(sample)
+ }

> mean(ybar2)
[1] 0.5013918

> var(ybar2)
[1] 0.04192942

> plot(density(ybar2), xlim = c(0, 1), ylim = c(0,3.5))
```

```
> curve(dnorm(x,0.5,sqrt(0.04193)), add=T, lty=2)

> ybar10<-numeric(10000)

> for(k in 1:10000) {
+     sample<-runif(10,min=0, max=1)
+     ybar10[k] <-mean(sample)
+ }

> mean(ybar10)
[1] 0.4987733

> var(ybar10)
[1] 0.008292283

> plot(density(ybar10),xlim = c(0, 1), ylim = c(0,4.5))

> curve(dnorm(x,0.5,sqrt(0.00829)), add=T, lty=2)

> ybar20<-numeric(10000)

> for(m in 1:10000) {
+     sample<-runif(20,min=0, max=1)
+     ybar20[m] <-mean(sample)
+ }

> mean(ybar20)
[1] 0.5003994

> var(ybar20)
[1] 0.004107226

> plot(density(ybar20),xlim = c(0, 1), ylim = c(0, 6.5))

> curve(dnorm(x,0.5,sqrt(0.00411)), add=T, lty=2)
```

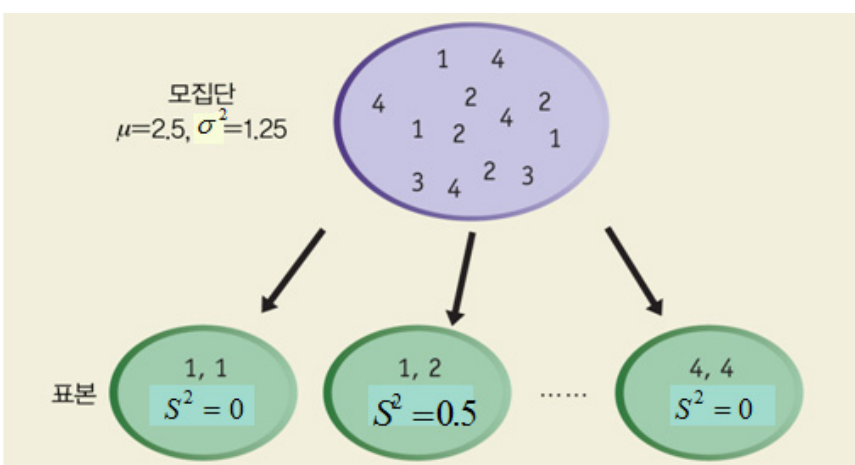
```
> ybar30<-numeric(10000)

> for(n in 1:10000) {
+     sample<-runif(30,min=0, max=1)
+     ybar30[n] <-mean(sample)
+ }

> mean(ybar30)
[1] 0.4996023

> var(ybar30)
[1] 0.002834952

> plot(density(ybar30),xlim = c(0, 1), ylim = c(0, 7.7))

> curve(dnorm(x,0.5,sqrt(0.00283)), add=T, lty=2)
```





3. 표본분산의 표본분포

앞의 예와 동일하게 J제약회사는 많은 종류의 신약을 개발하였다. 이 제약회사가 신약을 개발하기 위해서 1,2,3 혹은 4년의 시간이 걸렸으며 신약의 평균 개발기간인 모집단의 평균 $\mu = 2.5$ 및 분산 $\sigma^2 = 1.25$로 계산된다.



<그림 5-3> 모집단으로부터 표본크기(n)가 2인 임의표본추출

표본크기 2의 모든 가능한 표본과 표본평균 및 표본분산은 <표 5-3>과 같다. 표본평균의 경우 중심극한정리에 의해 표본크기가 증가함에 따라 정규분포를 따르지만, 표본분산의 경우 정규분포를 따르지 않는다.

<표 5-3> 표본크기 2인 표본, 표본평균 및 표본분산

표본	표본평균($\overline{X}$)	표본분산(S^2)	표본	표본평균($\overline{X}$)	표본분산(S^2)
(1,1)	1.0	0.0	(3,1)	2.0	2.0
(1,2)	1.5	0.5	(3,2)	2.5	0.5
(1,3)	2.5	2.0	(3,3)	3.0	0.0
(1,4)	2.5	4.4	(3,4)	3.5	0.5
(2,1)	1.5	0.5	(4,1)	2.5	4.5
(2,2)	2.0	0.0	(4,2)	3.0	2.0
(2,3)	2.5	0.5	(4,2)	3.5	0.5
(2,4)	3.0	2.0	(4,4)	4.0	0.0

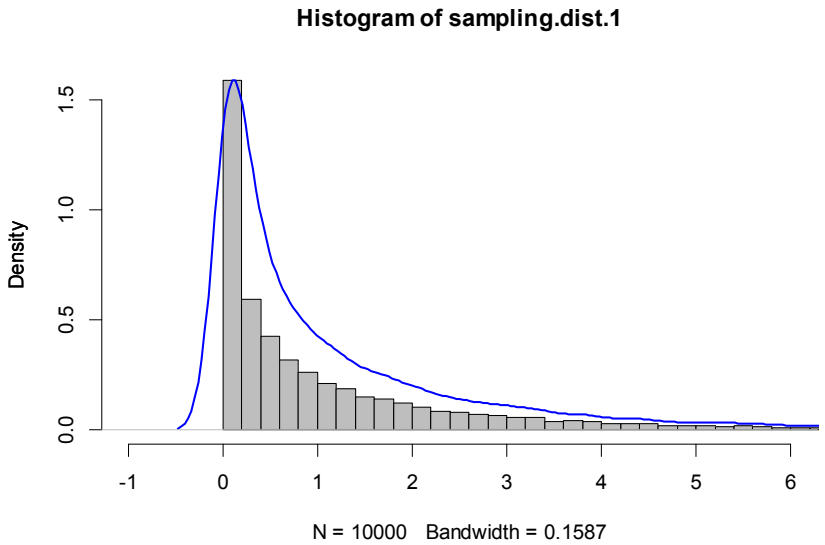
모집단이 정규분포에 따르더라도 표본분산의 표본분포는 다음 히스토그램과 같은 분포를 이루게 된다. 즉, 정규분포로부터 구한 표본분산의 분포는 정규분포에 따르지 않는다.

b1-ch5-7.R을 실행하면 표본분산의 평균은 1.232382, 분산은 3.227298로써 모집단의 이론적인 평균인 $\sigma^2 = 1.25$ 와 분산인 $\frac{2\sigma^4}{n-1} = 3.125$에 근접함을 알 수 있다.

b1-ch5-7.R의 실행결과

```
> sampling.dist.1<-NULL

> for(sample.count in 1:10000){
+     set.seed(sample.count)
+     sample.var.1<-var(rnorm(2,2.5,1.118))
+     sampling.dist.1<-c(sampling.dist.1, sample.va .... [TRUNCATED]

> mean(sampling.dist.1)
[1] 1.232382
```

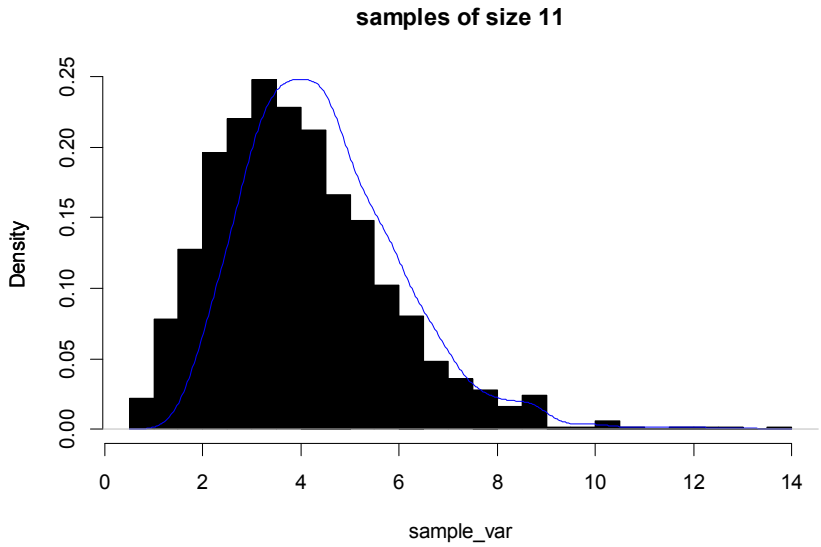
```
> var(sampling.dist.1)
[1] 3.227298

> hist(sampling.dist.1, freq=F, col="grey", xlab="", xlim=c(-1, 6), breaks=100)

> par(new=T)

> plot(density(sampling.dist.1), axes=F, main="", xlim=c(-1, 6), lwd=2, col="blue")
```



평균이 10이고, 표준편차가 2인 정규분포에 따르는 모집단에서 1,000개 표본으로부터 구한 표본분산 $s_1^2, s_2^2, \ldots, s_{1000}^2$ 을 이용하여 히스토그램을 그린 b1-ch5-8.R을 실행해 보면 정규분포에 따르지 않는다는 것을 확인할 수 있다.

b1-ch5-8.R의 실행결과

```
> sample_size <- 11

> n_rep <- 1000

> sample_mean <- rep(NA, n_rep)

> sample_var <- rep(NA, n_rep)

> #graphics.off()
> #par(mfrow=c(1,2))
>
> for (i in 1:n_rep) {
+         my_sample <- rnorm(sample_size,10,2)
+         sample_mean[i] <- mean(my_sample) .... [TRUNCATED]

> (mean(sample_mean))
[1] 10.01389

> (var(sample_mean))
[1] 0.3766244

> hist(sample_var, breaks=40, prob=T, main=paste( "samples of size 11"
),col="black")

> par(new=T)

> plot(density(sample_var), xlab="", axes=F, main="", col="blue")
```



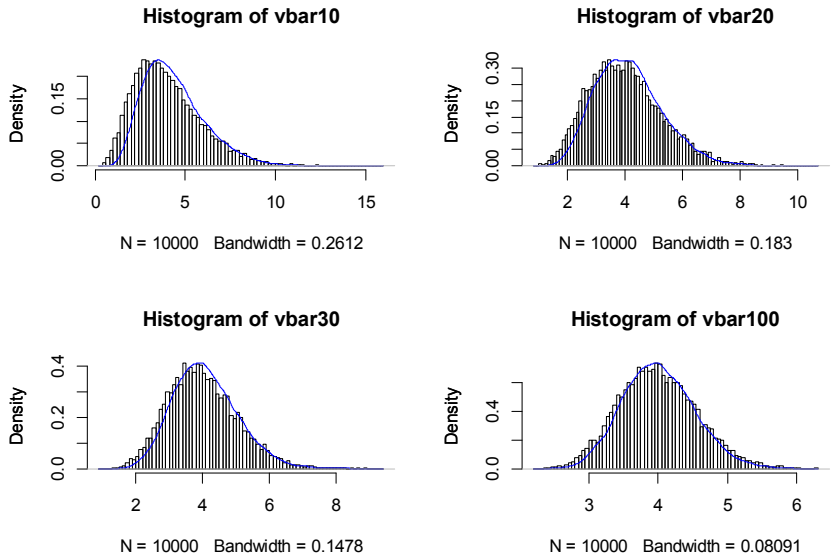
b1-ch5-9.R을 실행하면 표본크기를 다르게 할 경우 표본분산의 분포를 보여주고 있는데 모집단의 이론적인 평균인 σ^2과 분산인 $\frac{2\sigma^4}{n-1}$에 근접함을 알 수 있다.

b1-ch5-9.R의 실행결과

```
> set.seed(123456)

> par(mfrow=c(2,2))

> vbar10<-numeric(10000)

> for(j in 1:10000) {
+     sample10<-rnorm(10,10,2)
+     vbar10[j] <-var(sample10)
+ }

> (mean(vbar10))
[1] 4.020385

> (var(vbar10))
```

```
[1] 3.60356

> hist(vbar10, freq=F, xlab="", breaks=100)

> par(new=T)

> plot(density(vbar10), axes=F, main="", col="blue")

> vbar20<-numeric(10000)

> for(j in 1:10000) {
+     sample20<-rnorm(20,10,2)
+     vbar20[j] <-var(sample20)
+ }

> (mean(vbar20))
[1] 4.002698

> (var(vbar20))
[1] 1.676938

> hist(vbar20, freq=F, xlab="", breaks=100)

> par(new=T)

> plot(density(vbar20), axes=F, main="", col="blue")

> vbar30<-numeric(10000)

> for(j in 1:10000) {
+     sample30<-rnorm(30,10,2)
+     vbar30[j] <-var(sample30)
+ }

> (mean(vbar30))
[1] 3.993987
```

```
> (var(vbar30))
[1] 1.087883

> hist(vbar30, freq=F, xlab="", breaks=100)

> par(new=T)

> plot(density(vbar30), axes=F, main="", col="blue")

> vbar100<-numeric(10000)

> for(j in 1:10000) {
+     sample100<-rnorm(100,10,2)
+     vbar100[j] <-var(sample100)
+ }

> (mean(vbar100))
[1] 3.993146

> (var(vbar100))
[1] 0.3217269

> hist(vbar100, freq=F, xlab="", breaks=100)

> par(new=T)

> plot(density(vbar100), axes=F, main="", col="blue")
```


Histogram of vbar10
Density
0.00
0.15
0
5
10
15
N = 10000 Bandwidth = 0.2612
Histogram of vbar20
Density
0.00
0.15
0.30
2
4
6
8
10
N = 10000 Bandwidth = 0.183
Histogram of vbar30
Density
0.0
0.2
0.4
2
4
6
8
N = 10000 Bandwidth = 0.1478
Histogram of vbar100
Density
0.0
0.4
3
4
5
6
N = 10000 Bandwidth = 0.08091

제 6 장

추정

1. 추정 및 신뢰구간
2. 모평균의 구간추정
3. 모분산의 구간추정

제6장 추 정

1. 추정 및 신뢰구간

(1) 추정

추정량(estimator)은 표본정보를 이용하여 알지 못하는 모수의 참값을 추정하는 방법이며, 알지 못하는 모수가 θ라면 추정량은 $\hat{\theta}$으로 표기한다. 추정치(estimate)는 수치로 계산된 $\hat{\theta}$의 값이다.

추정량에는 모수를 하나의 값으로 추정하는 점추정량과 모수의 값이 빈번히 포함되는 구간을 추정하는 구간추정량이 있다.

점추정(point estimation)은 확률표본의 정보를 이용하여 모수에 대한 특정 값을 지정하는 방법인데, 표본평균은 모평균의 점추정치이고 표본분산은 모분산의 점추정치가 된다. 점추정량은 언제나 표본오차를 수반하기 때문에 전적으로 신뢰할 수 없다.

그러나 구간추정(interval estimation)은 이런 점추정과 달리 모수가 빈번히 포함되는 범위를 제공하여 연구의 목적에 따라 원하는 만큼의 신뢰성을 가지고 모수를 추정할 수 있다

(2) 신뢰구간

θ는 알지 못하는 모수라고 하자. 표본정보에 근거하여 일정한 확률$(1-\alpha)$ 범위 내에 모수가 포함될 가능성이 있는 구간, 즉 다음을 만족하는 확률변수 A와 B를 구할 수 있다.

$$P(A < \theta < B) = 1 - \alpha$$

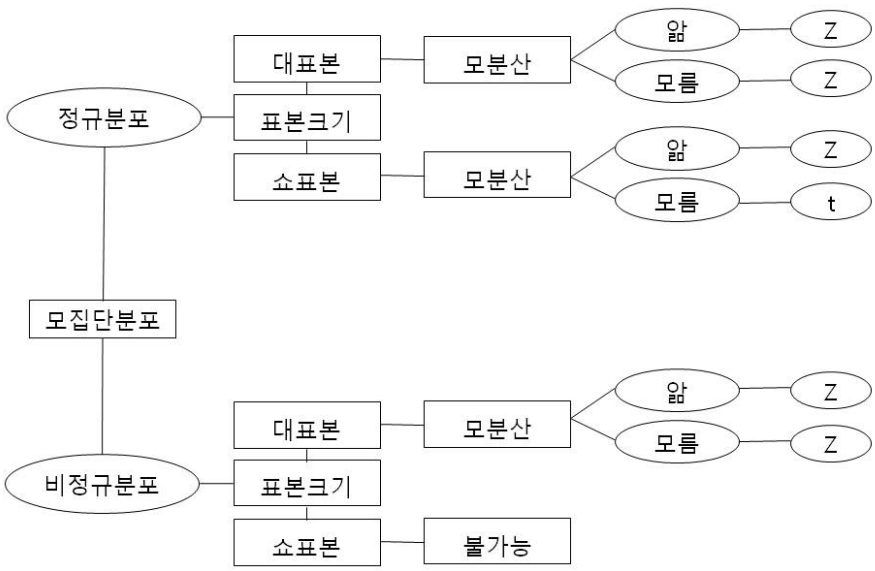
만약 확률변수 A와 B에 대한 측정값을 a와 b라고 하면, 구간 $a < \theta < b$ 는 θ에 대한 $100(1-\alpha)\%$ 신뢰구간이며 $1-\alpha$는 신뢰수준 그리고 α는 유의수준이라고 한다.

2. 모평균 구간추정

<그림 6-1>은 모평균 구간추정을 위한 의사결정 트리를 나타내고 있다.

모집단의 정규분포 여부에 관계없이, 모분산을 알든 모르든 관계없이, 표본의 크기가 30 이상의 대표본이면 Z-통계량을 이용하면 된다.

표본의 크기가 30 미만의 소표본이면, 모집단이 정규분포에 따를 경우에만 모평균 구간추정이 가능하고, 이 경우에 모분산이 알려져 있으면 Z-통계량을 이용하고 모분산을 모를 경우 t-통계량을 이용하면 된다.



<그림 6-1> 모평균 구간추정을 위한 의사결정 트리

평균이 10이고 표준편차가 2인 모집단에서 100개의 표본을 추출하여 모평균(μ)에 대한 신뢰구간을 추정하는 b1-ch6-1.R을 실행해 보면 모평균에 대한 95% 신뢰구간

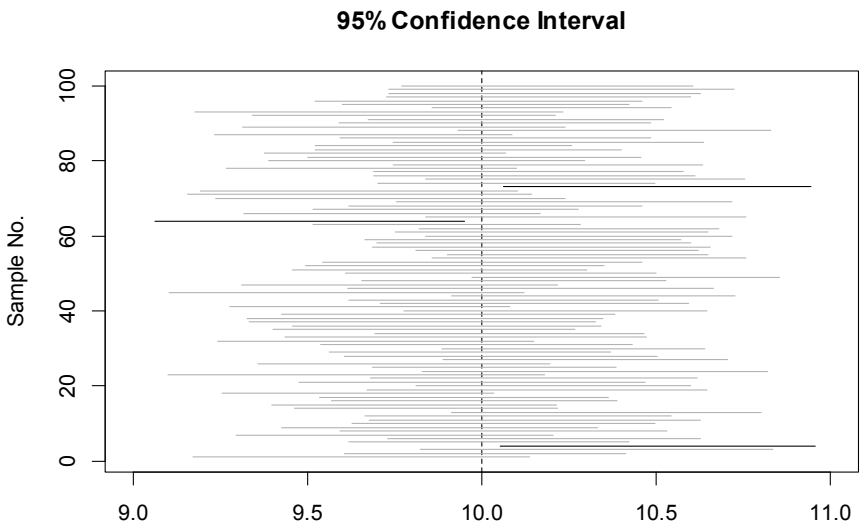
100개 중 95개 이상의 신뢰구간이 모평균 10을 포함하고 있는 것을 확인할 수 있다.

모평균 μ가 확률변수가 아니고 고정된 상수이므로 μ에 대한 95% 신뢰구간의 의미는 크기가 동일한 100개의 서로 다른 표본에 의해 동일한 공식으로 100개의 신뢰구간을 구했을 때 그 중에서 95개의 구간이 모평균 μ를 포함한다고 볼 수 있다는 것이다.

b1-ch6-1.R의 실행결과

```
> set.seed(12343)

> #par(mfrow = c(1,2))
>
> CIlower<-numeric(100)

> CIupper<-numeric(100)

> pvalue1<-numeric(100)

> for(j in 1:100) {
+     sample<-rnorm(80,10,2)
+     testres1<-t.test(sample,mu = 10)
+     CIlower[j]<-testres1$conf.int[1]
+     CIupper[j]<-testres1$conf.i .... [TRUNCATED]

> testres1$conf.int[1]
[1] 9.769371

> testres1$conf.int[2]
[1] 10.60363

> testres1$p.value
[1] 0.3762031

> reject1<-pvalue1<=0.05

> table(reject1)
```

```
reject1
FALSE  TRUE
   97     3

> color<-rep(gray(.7),100)

> color[reject1]<-"black"

> plot(0, xlim=c(9,11), ylim=c(1,100), ylab="Sample No.", xlab="",
main="95% Confidence Interval")

> abline(v=10, lty=2)

> for(j in 1:100) {
+   lines(c(CIlower[j], CIupper[j]), c(j,j), col=color[j], lwd=1)
+ }
```



(1) Z-통계량 이용

모집단의 정규분포 여부에 관계없이, 모분산을 알든 모르든 관계없이, 표본의 크기가 30 이상의 대표본인 경우 또는 모집단이 정규분포에 따르고, 모분산이 알려져 있으며, 표본의 크기가 30 미만의 소표본인 경우 Z-통계량을 이용하여 모평균에 대한 신뢰구간을 구할 수 있다.

평균 μ와 분산 σ^2을 가지는 모집단이 있으며, μ는 모르고 σ^2만 안다고 가정하자. 그리고 이 모집단으로부터 $x_1, x_2, ..., x_n$ 을 표본추출하였으며 σ^2은 알지만 μ는 모르기 때문에 신뢰구간을 통하여 μ를 추정할 수 있으며 Z-통계량을 이용하면 된다.

신뢰수준이 $1-\alpha$가 되는 표준정규분포의 신뢰구간은 다음과 같다.

$$P(-z_{\alpha/2} < Z < z_{\alpha/2}) = 1-\alpha$$

여기서 확률변수 $Z = \dfrac{\overline{X}-\mu}{\dfrac{\sigma}{\sqrt{n}}}$ 이므로 위의 확률등식을 다음과 같이 다시 쓸 수 있다.

$$P(-z_{\alpha/2} < \frac{\overline{X}-\mu}{\frac{\sigma}{\sqrt{n}}} < z_{\alpha/2}) = 1-\alpha$$

$$P(-z_{\alpha/2}\frac{\sigma}{\sqrt{n}} < \overline{X}-\mu < z_{\alpha/2}\frac{\sigma}{\sqrt{n}}) = 1-\alpha$$

$$P(\overline{X}-z_{\alpha/2}\frac{\sigma}{\sqrt{n}} < \mu < \overline{X}+z_{\alpha/2}\frac{\sigma}{\sqrt{n}}) = 1-\alpha$$

따라서 $a = \overline{X}-z_{\alpha/2}\dfrac{\sigma}{\sqrt{n}}$ 부터 $b = \overline{X}+z_{\alpha/2}\dfrac{\sigma}{\sqrt{n}}$ 까지의 임의구간이 모평균 μ를 포함할 확률은 $1-\alpha$가 되므로 μ에 대한 $100(1-\alpha)\%$의 신뢰구간은 다음과 같다.

$$(\overline{X}-z_{\alpha/2}\frac{\sigma}{\sqrt{n}} \ , \ \overline{X}+z_{\alpha/2}\frac{\sigma}{\sqrt{n}} \ )$$

(연습 1)

한 기업에서 현대인들이 대기오염에 시달린다는 사실에 착안하여 언제나 신선한 산소를 마실 수 있는 휴대용 산소제품을 개발하기로 하였다. 따라서 이 기업의 연구팀은 일반인들의 산소 소비량을 측정하기 위해 임의로 35명을 선정, 분당 산소 소비량을 조사하여 다음 자료를 얻었다. 이 연구팀은 일반인들의 모집단은 정규분포하며 분산이 0.36이라는 사실을 알고 있다고 가정하자. 모집단 평균의 95% 신뢰구간을 구하라.

0.360	1.189	0.614	0.788	0.273	2.464	0.517	1.827	0.537	0.374	0.449	0.262
0.448	0.971	0.372	0.898	0.411	0.348	1.925	0.550	0.622	0.610	0.319	0.406
0.413	0.767	0.385	0.674	0.521	0.603	0.533	0.662	1.177	0.307	1.499	

b1-ch6-2.R을 실행해 보면 모평균에 대한 95% 신뢰구간은 (0.5176519, 0.9152052)이다. 그러므로 μ는 95% 신뢰수준에서 (0.5176519,0.9152052)에 속하게 된다.

b1-ch6-2.R의 실행결과

```
> data1<-"http://kanggc.iptime.org/book/data/chap10-2.csv"

> data1_dat<-as.matrix(read.csv(data1,header=T), ncol=1)

> var1<-data1_dat[,1]

> xbar=mean(var1)

> z<-qnorm(0.025, 0, 1, lower.tail=F)

> LCL1<-xbar-z*(0.6/sqrt(35))

> UCL1<-xbar+z*(0.6/sqrt(35))

> LCL1;UCL1
[1] 0.5176519
[1] 0.9152052
```

(2) t-통계량 이용

모집단이 정규분포에 따르고, 표본의 크기가 30 미만의 소표본이며, 모분산이 알려져 있지 않은 경우 t-통계량을 이용하여 모평균에 대한 신뢰구간을 구할 수 있다.

모분산을 모르고 표본의 크기가 충분히 크지 않을 때, 확률변수 $t=\frac{\overline{X}-\mu}{\frac{s}{\sqrt{n}}}$ 는 자유도 n-1인 t-분포(Student's t-distribution)에 따른다.

신뢰수준이 $1-\alpha$가 되는 t-분포의 신뢰구간은 다음과 같다.

$$P(-t_{(n-1,\alpha/2)} < t_{(n-1)} < t_{(n-1,\alpha/2)}) = 1-\alpha$$

여기서 확률변수 $t=\frac{\overline{X}-\mu}{\frac{s}{\sqrt{n}}}$ 이므로 위의 확률등식을 다음과 같이 다시 쓸 수 있다.

$$P(-t_{(n-1,\alpha/2)} < \frac{\overline{X}-\mu}{\frac{s}{\sqrt{n}}} < t_{(n-1,\alpha/2)}) = 1-\alpha$$

$$P(-t_{(n-1,\alpha/2)}\frac{s}{\sqrt{n}} < \overline{X}-\mu < t_{(n-1,\alpha/2)}\frac{s}{\sqrt{n}}) = 1-\alpha$$

$$P(\overline{X}-t_{(n-1,\alpha/2)}\frac{s}{\sqrt{n}} < \mu < \overline{X}+t_{(n-1,\alpha/2)}\frac{s}{\sqrt{n}}) = 1-\alpha$$

따라서 $a=\overline{X}-t_{(n-1,\alpha/2)}\frac{s}{\sqrt{n}}$ 부터 $b=\overline{X}+t_{(n-1,\alpha/2)}\frac{s}{\sqrt{n}}$ 까지의 임의구간이 모평균 μ를 포함할 확률은 $1-\alpha$ 가 되므로 μ에 대한 $100(1-\alpha)\%$의 신뢰구간은 다음과 같다.

$$(\overline{X}-t_{(n-1,\alpha/2)}\frac{s}{\sqrt{n}}\ ,\ \overline{X}+t_{(n-1,\alpha/2)}\frac{s}{\sqrt{n}}\ )$$

(연습 2)

한 피자 체인점의 지배인은 피자 배달시간이 오래 걸린다는 소비자들의 불평을 확인해 보기 위해 피자 배달주문 중 임의로 20개를 선정하여 배달시간(단위 : 분)을 측정하였더니 다음과 같았다. 모집단이 정규분포를 한다고 가정하고 모평균 배달시간에 대한 95% 신뢰구간을 구하라.

14	10	9	10	11	16	15	8	6	18
17	4	12	15	14	15	9	8	7	16

b1-ch6-3.R을 실행해 보면 모평균에 대한 95% 신뢰구간은 (9.808949, 13.5910)이다. 그러므로 μ는 95% 신뢰수준에서 (9.808949, 13.5910)에 속하게 된다.

b1-ch6-3.R의 실행결과

```
> library(foreign)

> time<-read.dta(file = "http://kanggc.iptime.org/book/data/chap10-2.dta")

> n<-length(time$var1)

> s<-sd(time$var1)

> t19<-qt(0.025, df = 19, lower.tail = F)

> average<-mean(time$var1)

> LCL<-average-t19*(s/sqrt(n))

> UCL<-average + t19*(s/sqrt(n))

> LCL;UCL
[1] 9.808949
[1] 13.5910
```

3. 모분산 구간추정

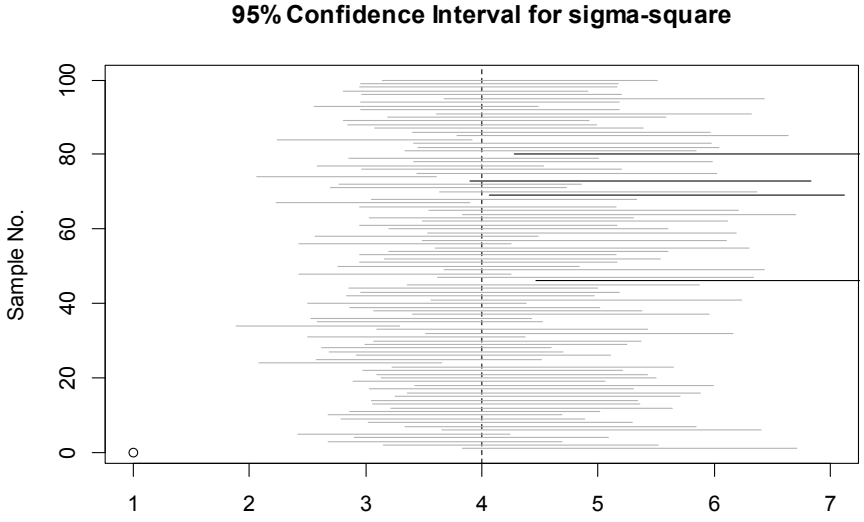
모평균이 10이고 표준편차가 2인 모집단에서 100개의 표본을 추출하여 모분산(σ^2)에 대한 신뢰구간을 추정하는 b1-ch6-4.R을 실행해 보면 모분산에 대한 95% 신뢰구간 100개 중 95개 이상의 신뢰구간이 모분산 4를 포함하고 있는 것을 확인할 수 있다.

b1-ch6-4.R의 실행결과

```
> set.seed(12345)

> CIlower<-numeric(100)

> CIupper<-numeric(100)

> pvalue1<-numeric(100)

> for(j in 1:100) {
+     sample<-rnorm(100,10,2)
+     s2<-var(sample)
+     chi<-(99*s2)/4
+     pvalue1[j]<-pchisq(chi, 0.95, df=99, lower.tail=F)
+     chi_ .... [TRUNCATED]

> CIlower
  [1] 3.831738 3.153297 2.680453 2.905234 2.421398 3.657560
  [7] 3.335122 3.025787 2.788475 2.679994 2.863353 3.220437
 [13] 3.059344 3.051541 3.257024 3.358969 3.028908 3.421953
 [19] 2.892686 3.138877 3.097010 2.978539 3.224138 2.086591
 [25] 2.576973 2.918803 2.682981 2.625174 2.997243 3.068458
 [31] 2.496968 3.519971 3.098808 1.882539 2.581418 2.527020
 [37] 3.402880 3.069659 2.863192 2.501846 3.562625 2.840043
 [43] 2.959190 2.853818 3.356341 4.461403 3.619685 2.428544
 [49] 3.673904 2.761155 2.950630 3.162173 2.946491 3.197005
 [55] 3.599023 2.430203 3.484282 2.563818 3.535696 3.199001
 [61] 2.951949 3.490446 3.027803 3.827879 3.545298 2.944418
```

```
 [67] 2.227769 3.046476 4.067420 3.637768 2.699178 2.775919
 [73] 3.901155 2.063341 3.439670 2.970812 2.586364 3.417553
 [79] 2.858123 4.282643 3.336033 3.447982 3.409942 2.238515
 [85] 3.790552 3.408887 3.076498 2.850088 2.812755 3.191278
 [91] 3.609685 2.960486 2.560492 2.957969 3.674383 2.969041
 [97] 2.806176 2.949856 2.954241 3.144629

> CIupper
  [1] 6.707636 5.519993 4.692259 5.085748 4.238771 6.402729
  [7] 5.838286 5.296782 4.881355 4.691455 5.012433 5.637524
 [13] 5.355525 5.341864 5.701572 5.880032 5.302245 5.990288
 [19] 5.063782 5.494750 5.421461 5.214072 5.644003 3.652675
 [25] 4.511111 5.109500 4.696684 4.595489 5.246813 5.371478
 [31] 4.371059 6.161874 5.424607 3.295472 4.518893 4.423667
 [37] 5.956901 5.373582 5.012150 4.379598 6.236542 4.971627
 [43] 5.180201 4.995741 5.875431 7.809894 6.336428 4.251280
 [49] 6.431340 4.833531 5.165215 5.535532 5.157970 5.596506
 [55] 6.300257 4.254183 6.099398 4.488083 6.189401 5.600000
 [61] 5.167524 6.110189 5.300310 6.700880 6.206210 5.154340
 [67] 3.899813 5.332998 7.120209 6.368082 4.725038 4.859376
 [73] 6.829153 3.611974 6.021303 5.200544 4.527551 5.982585
 [79] 5.003278 7.496967 5.839881 6.035853 5.969262 3.918625
 [85] 6.635538 5.967416 5.385554 4.989212 4.923859 5.586481
 [91] 6.318921 5.182469 4.482260 5.178063 6.432179 5.197444
 [97] 4.912342 5.163861 5.171537 5.504819

> pvalue1
  [1] 0.059365461 0.445263806 0.836304403 0.667558618
  [5] 0.950396583 0.112775148 0.295535011 0.560355653
  [9] 0.762139944 0.836587466 0.702893025 0.387075359
 [13] 0.529854093 0.536949332 0.356642919 0.278058096
 [17] 0.557521927 0.234824428 0.678284751 0.458071934
 [21] 0.495683276 0.603036470 0.383951114 0.995207162
 [25] 0.892743695 0.655838906 0.834741009 0.868303202
 [29] 0.586204653 0.521571287 0.926280445 0.176434103
 [33] 0.494058478 0.999351346 0.890624839 0.914690263
 [37] 0.247456105 0.520479744 0.703026500 0.924479633
```

```
 [41] 0.154458251 0.721927270 0.620322274 0.710738072
 [45] 0.279955249 0.003081599 0.128240912 0.948408171
 [49] 0.106559718 0.782262224 0.627920030 0.437426138
 [53] 0.631580980 0.407074407 0.137322176 0.947938374
 [57] 0.196418993 0.898853957 0.168091943 0.405356954
 [61] 0.626751454 0.192862620 0.558525412 0.060268861
 [65] 0.163136320 0.633411882 0.985429029 0.541554870
 [69] 0.021943166 0.120669129 0.824520124 0.771495324
 [73] 0.044942131 0.996088025 0.223456829 0.609957335
 [77] 0.888235164 0.237702517 0.707205799 0.007847395
 [81] 0.294856592 0.218246940 0.242732095 0.984268856
 [85] 0.069605521 0.243433826 0.514269323 0.713784971
 [89] 0.743550819 0.412016516 0.132578477 0.619169279
 [93] 0.900361295 0.621408161 0.106381603 0.611540571
 [97] 0.748650567 0.628604973 0.624718737 0.452952109

> reject1<-pvalue1<=0.05

> table(reject1)
reject1
FALSE  TRUE
   96     4

> color<-rep(gray(.7),100)

> color[reject1]<-"black"

> plot(0, xlim=c(1,7), ylim=c(1,100), ylab="Sample No.", xlab="", main="95% 
Confidence Interval for sigma-square")

> abline(v=4, lty=2)

> for(j in 1:100) {
+     lines(c(CIlower[j], CIupper[j]), c(j,j), col=color[j], lwd=1)
+ }
```



확률변수 $\chi^2_{(n-1)} = \frac{(n-1)s^2}{\sigma^2}$ 는 자유도가 n-1인 χ^2-분포를 따른다.

신뢰수준이 $1-\alpha$가 되는 $\chi^2_{(n-1)}$ 의 신뢰구간은 다음과 같다.

$$P(\chi^2_{(n-1,1-\alpha/2)} < \chi^2_{(n-1)} < \chi^2_{(n-1,\alpha/2)}) = 1-\alpha$$

여기서 확률변수 $\chi^2_{(n-1)} = \frac{(n-1)s^2}{\sigma^2}$ 이므로 위의 확률등식을 다음과 같이 다시 쓸 수 있다.

$$P(\chi^2_{(n-1,1-\alpha/2)} < \frac{(n-1)s^2}{\sigma^2} < \chi^2_{(n-1,\alpha/2)}) = 1-\alpha$$

$$P(\frac{(n-1)s^2}{\chi^2_{(n-1,\alpha/2)}} < \sigma^2 < \frac{(n-1)s^2}{\chi^2_{(n-1,1-\alpha/2)}}) = 1-\alpha$$

따라서 $a = \frac{(n-1)s^2}{\chi^2_{(n-1,\alpha/2)}}$ 부터 $b = \frac{(n-1)s^2}{\chi^2_{(n-1,1-\alpha/2)}}$ 까지의 임의구간이 모분산 σ^2를 포함할 확률은 $1-\alpha$ 가 되므로 σ^2에 대한 $100(1-\alpha)\%$의 신뢰구간은 다음과 같다.

$$(\frac{(n-1)s^2}{\chi^2_{(n-1,\alpha/2)}} \ , \ \frac{(n-1)s^2}{\chi^2_{(n-1,1-\alpha/2)}})$$

(연습 3)

K제약회사에서 두통약을 개발하였으며 이 약의 효과를 알아보기 위해 두통환자 10명을 임의로 선정하여 두통약을 복용하게 한 후 두통 억제 시간(단위 : 분)을 측정하였다. 두통 억제 시간이 정규분포할 때, 모분산에 대한 99%의 신뢰구간을 구하라.

66	37	18	31	85	63	73	83	65	80

b1-ch6-5.R을 실행해 보면 모분산에 대한 99% 신뢰구간은 (208.8612, 2839.822)이다. 그러므로 σ^2는 99% 신뢰수준에서 (208.8612, 2839.822)에 속하게 된다.

b1-ch6-5.R의 실행결과

```
> time<-c(66,37,18,31,85,63,73,83,65,80)

> df<-length(time)-1

> s.sq<-var(time)

> u.chi<-qchisq(0.005, df=df, lower.tail=F)

> u.chi
[1] 23.58935

> l.chi<-qchisq(0.995, df=df, lower.tail=F)

> l.chi
[1] 1.734933

> LCL<-(df*s.sq/u.chi)

> UCL<-(df*s.sq/l.chi)

> LCL;UCL
[1] 208.8612
[1] 2839.822
```

제 7 장

가설검정

1. 가설검정의 기초개념
2. 단일집단에 대한 가설검정
3. 두 집단에 대한 가설검정

제7장 가설검정

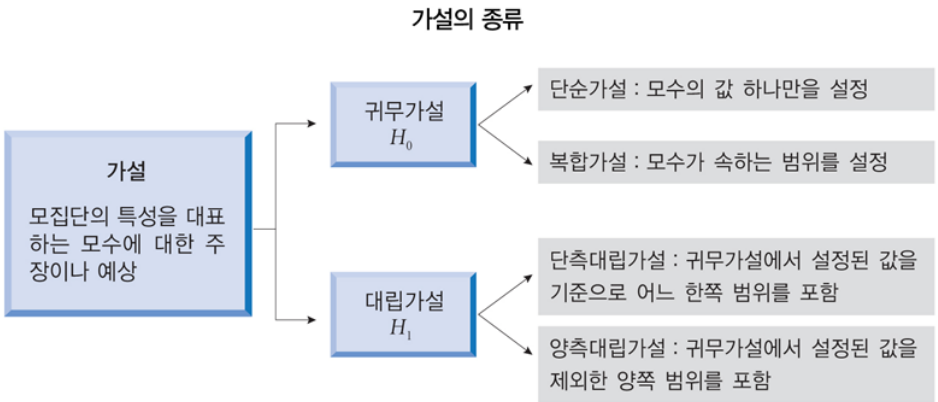
1. 가설검정의 기초개념

(1) 가설의 종류

가설이란 모집단의 특성을 대표하는 모수에 대한 주장이나 예상을 의미하는데 가설의 종류에는 귀무가설(Null Hypothesis : H_0)과 대립가설(Alternative Hypothesis : H_1)이 있다.

귀무가설에는 모수의 값 하나만을 설정하는 단순가설과 모수가 속하는 범위를 설정하는 복합가설이 있다.

대립가설에는 귀무가설에서 설정된 값을 기준으로 어느 한쪽 범위를 포함하는 단측대립가설과 귀무가설에서 설정된 값을 제외한 양쪽 범위를 포함하는 양측대립가설이 있다.



(2) 가설설정

일반적으로 통계분석에서는 모집단의 모수에 대하여 관심이 있으므로 가설은 모수에 대하여 설정한다.

“모수가 특정한 값이다”, “두 모수의 값이 같다” 등과 같이 간단하고 구체적인 경우를 귀무가설로 설정한다.

“모수가 특정한 값이 아니다”, “한 모수의 값이 다른 모수의 값보다 크다”, “두 모수의 값이 다르다” 등과 같이 모수에 대한 관심의 영역 중에서 귀무가설로 지정되지 않은 모든 경우를 포괄적으로 대립가설로 설정한다.

(3) 가설검정

모집단에 대한 어떤 가설을 설정한 뒤에 표본관찰을 통하여 그 가설의 채택여부를 결정하는 분석방법을 가설검정이라고 한다.

가설검정이란 두 가설 H_0와 H_1 중에서 하나를 선택하는 과정이므로 H_0를 채택(accept)하면 H_1을 기각(reject)하게 되고 H_0를 기각하면 H_1을 채택하게 된다.

따라서 가설검정이란 ‘두 가설 중에서 귀무가설 H_0를 채택하든지 또는 기각하는 과정’ 이라고 이해할 수 있다.

(4) 검정통계량

검정통계량은 가설검정에서 관찰된 표본으로부터 구하는 통계량으로 분포가 가설에서 주어지는 모수에 의존한다.

귀무가설이 옳다는 전제하에서 구한 검정통계량의 값이 나타날 가능성이 크면 귀무가설을 채택하고 나타날 가능성이 작으면 귀무가설을 기각한다.

$$\text{검정통계량} = \frac{\text{표본통계량} - \text{모수의 귀무가설의 값}}{\text{표본통계량의 표준오차}}$$

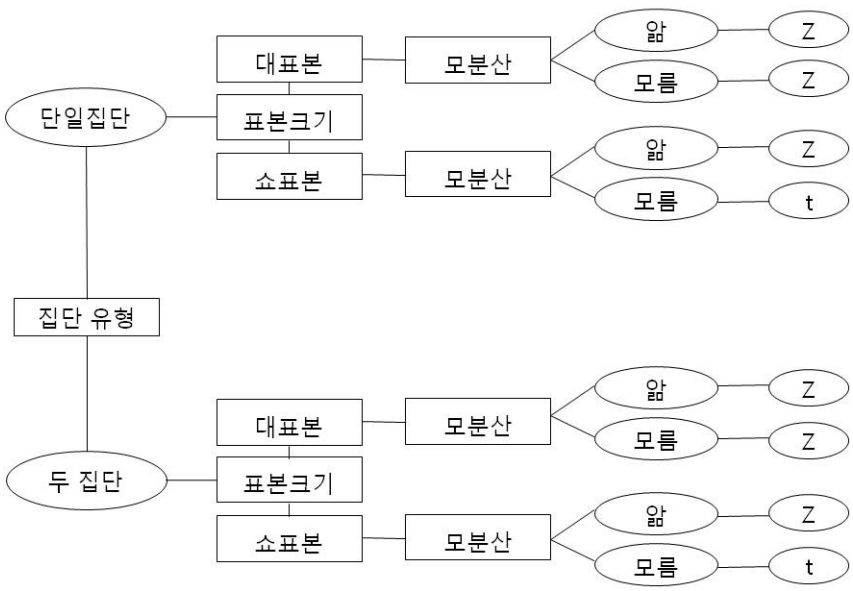
가설검정은 단일집단 모평균 및 모분산에 대한 가설검정이 있고, 두 집단의 모평균

및 모분산에 대한 가설검정이 있는데 이와 관련된 검정통계량으로 정리해 보면 <표 7-1>과 같다.

<표 7-1> 가설검정의 유형에 따른 검정통계량

구분	단일집단	두 집단
모평균	Z-검정통계량, t-검정통계량	Z-검정통계량, t-검정통계량
모분산	χ^2-검정통계량	F-검정통계량

한편, 모평균 가설검정을 위한 의사결정 트리를 그려보면 <그림 7-1>과 같다.



<그림 7-1> 모평균 가설검정을 위한 의사결정 트리

(5) 유의수준과 신뢰수준

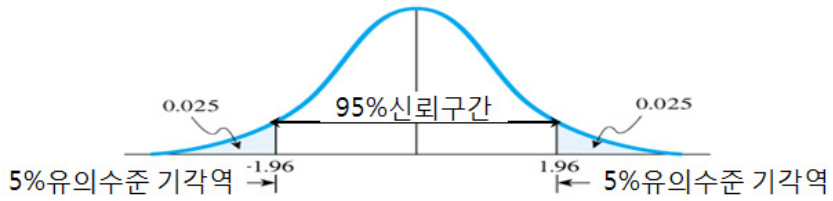
유의수준 α란 귀무가설이 옳은데도 불구하고 이를 기각하는 확률의 크기를 말하며 검정통계량을 구하는 것과는 무관하게 검정을 실시하는 사람의 판단에 따라 결정되

는데 일반적으로 유의수준은 1%, 5%, 10% 중 하나를 정한다.

가설검정에서 유의수준 α%는 구간추정에서 신뢰수준 $(100-\alpha)$%와 동일한 의미를 갖는다.

예를 들어, 표준정규분포에서 95% 신뢰구간은 $P(-1.96 \leq Z \leq 1.96) = 0.95$ 에 의하여 (-1.96, 1.96)임을 알 수 있는데, 표준정규분포를 이용하는 검정에서 5% 유의수준 하에서의 기각역은 $(-\infty, -1.96)$과 $(1.96, \infty)$로 위의 95% 신뢰구간과 반대가 된다.

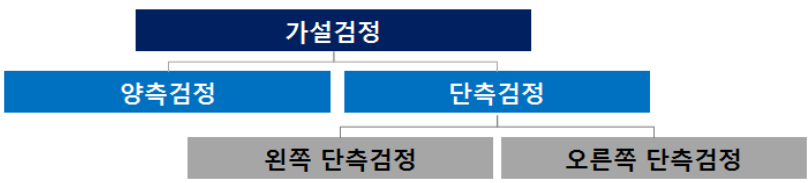
이와 같이 5% 유의수준 하에서의 기각역은 95% 신뢰수준 하에서의 신뢰구간과 반대의 의미를 가지고 있기 때문에 유의수준 α%하에서의 검정은 $(100-\alpha)$%의 신뢰수준 하에서의 검정이라고도 한다.



(6) 기각역

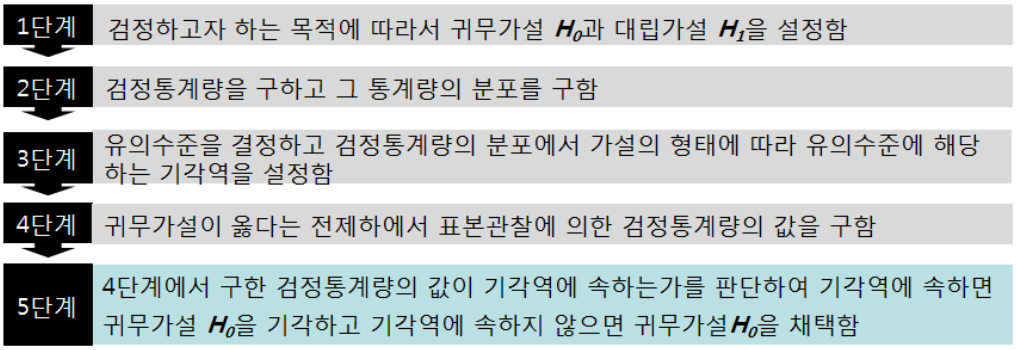
기각역이란 가설검정에서 유의수준 α가 정해졌을 때, 검정통계량의 분포에서 이 유의수준의 크기에 해당하는 영역을 말하는데, 이 영역의 위치는 대립가설의 형태에 따라 달라진다.

대립가설의 형태는 가설검정의 목적에 의하여 결정되는데 가설검정은 대립가설의 형태에 따라 양측검정과 단측검정으로 나누어지고, 단측검정은 다시 왼쪽 단측검정과 오른쪽 단측검정으로 분류된다.



(7) 가설검정 단계

가설검정 과정을 단계적으로 설명하면 다음과 같다



(8) 제1종의 오류 및 제2종의 오류

제1종 오류(type Ⅰ error)는 귀무가설 H_0가 옳은데도 불구하고 H_0를 기각하는 오류를 말한다. 이것이 나타날 확률을 제1종 오류의 크기라고 하는데, 이는 앞에서 정의된 유의수준 α와 같다.

제2종 오류(type Ⅱ error)는 귀무가설 H_0가 옳지 않은데도 H_0를 채택하는 오류를 말한다. 이것이 나타날 확률을 제2종 오류의 크기라고 하는데 이를 β로 표현한다.

가설검정결과 / 정확한 사실	H_0가 사실이라고 판정	H_0가 사실이 아니라고 판정
H_0가 사실임	옳은 결정	제1종의 오류(α)
H_0가 사실이 아님	제2종의 오류(β)	옳은 결정

b1-ch7-1.R을 실행하면 H_0이 사실이 아님에도 불구하고 H_0을 채택하는 제2종의 오류를 보여준다. 평균이 10이고 표준편차가 2인 모집단($H_0 : \mu_0 = 10$ 이 사실인 귀무가설)에서 100개의 표본을 추출하여 $H_0 : \mu_0 = 9.5$, $H_0 : \mu_0 = 9.3$, $H_0 : \mu_0 = 9.2$ 의 가설(H_0가 사실이 아님)을 검정하기 위한 신뢰구간을 살펴보면 귀무가설의 값이 사실에서 멀어질수록 신뢰구간이 귀무가설이 사실이 아닌 모평균을 포함하는 경우는

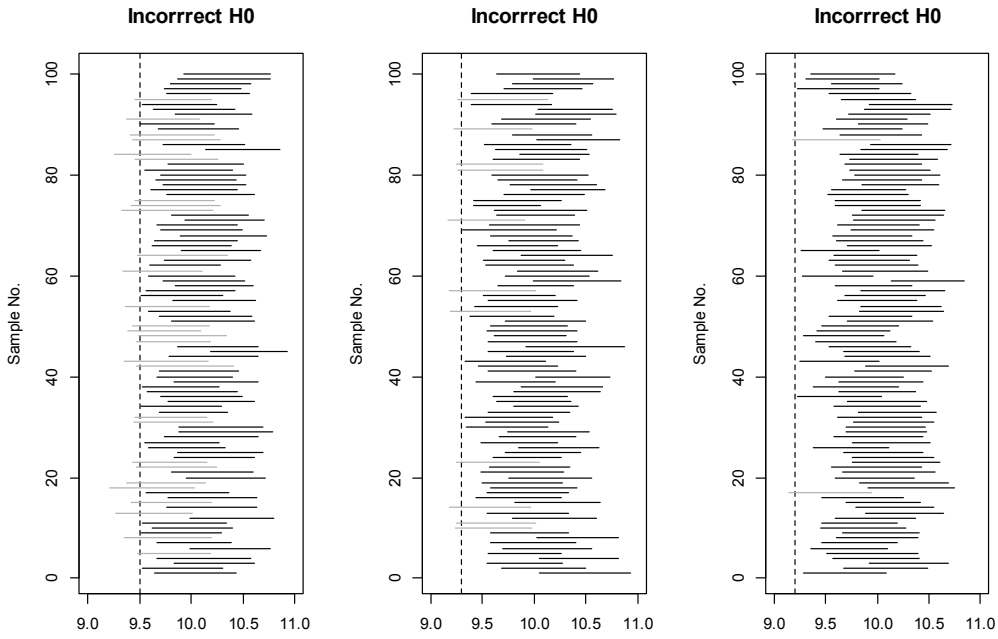
점점 적어지는 즉, 제2종의 오류가 작아지는 것을 확인할 수 있다.

b1-ch7-1.R의 실행결과

```
> set.seed(123456)

> par(mfrow=c(1,3))

> CIlower<-numeric(100)

> CIupper<-numeric(100)

> pvalue1<-numeric(100)

> pvalue2<-numeric(100)

> for(j in 1:1000) {
+     sample<-rnorm(100,10,2)
+     testres1<-t.test(sample,mu=10)
+     CIlower[j]<-testres1$conf.int[1]
+     CIupper[j]<-testres1$conf .... [TRUNCATED]

> reject2<-pvalue2<=0.05

> table(reject2)
reject2
FALSE  TRUE
  294   706

> color<-rep(gray(.7),100)

> color[reject2[1:100]]<-"black"

> plot(0, xlim=c(9,11), ylim=c(1,100), ylab="Sample No.", xlab="",
main="Incorrrect H0")

> abline(v=9.5, lty=2)
```

```
> for(j in 1:100) {
+     lines(c(CIlower[j], CIupper[j]), c(j,j), col=color[j], lwd=1)
+ }

> set.seed(12345)

> CIlower<-numeric(100)

> CIupper<-numeric(100)

> pvalue1<-numeric(100)

> pvalue2<-numeric(100)

> for(j in 1:100) {
+     sample<-rnorm(100,10,2)
+     testres1<-t.test(sample,mu=10)
+     CIlower[j]<-testres1$conf.int[1]
+     CIupper[j]<-testres1$conf. .... [TRUNCATED]

> reject2<-pvalue2<=0.05

> table(reject2)
reject2
FALSE  TRUE
   11    89

> color<-rep(gray(.7),100)

> color[reject2[1:100]]<-"black"

>   plot(0,   xlim=c(9,11),   ylim=c(1,100),   ylab="Sample   No.",   xlab="",
main="Incorrrect H0")

> abline(v=9.3, lty=2)

> for(j in 1:100) {
+     lines(c(CIlower[j], CIupper[j]), c(j,j), col=color[j], lwd=1)
```

```
+ }

> set.seed(1234)

> CIlower<-numeric(100)

> CIupper<-numeric(100)

> pvalue1<-numeric(100)

> pvalue2<-numeric(100)

> for(j in 1:100) {
+     sample<-rnorm(100,10,2)
+     testres1<-t.test(sample,mu=10)
+     CIlower[j]<-testres1$conf.int[1]
+     CIupper[j]<-testres1$conf. .... [TRUNCATED]

> reject2<-pvalue2<=0.05

> table(reject2)
reject2
FALSE  TRUE
    2    98

> color<-rep(gray(.7),100)

> color[reject2[1:100]]<-"black"

> plot(0, xlim=c(9,11), ylim=c(1,100), ylab="Sample No.", xlab="",
main="Incorrrect H0")

> abline(v=9.2, lty=2)

> for(j in 1:100) {
+     lines(c(CIlower[j], CIupper[j]), c(j,j), col=color[j], lwd=1)
+ }
```



2. 단일집단에 대한 가설검정

(1) 모평균에 대한 가설검정

(연습 1)

어떤 아이스크림 회사의 영업부 사원은 체인점의 여름 판매량보다 겨울 판매량이 평균 34.5%감소한다고 한다. 전국 체인점 중 15개를 표본추출하여 판매 감소량을 조사해 보니 다음과 같이 나타났다. 모집단이 정규분포한다고 가정하고 판매량의 감소가 평균 34.5%라는 귀무가설을 10% 유의수준에서 양측검정하라.

33.46	33.38	32.73	32.15	33.99	34.10	33.97	34.34
22.95	33.85	34.23	34.05	34.13	34.45	34.19	

b1-ch7-2.R을 실행해 보면 계산된 검정통계량의 값은 -4.3136이고 10% 유의수

준 하에서 자유도가 14인 t-분포의 임계치는 -1.7631과 1.7631이므로 10% 유의수준 하에서 모평균이 34.5라는 귀무가설을 기각한다. 또한 계산된 검정통계량의 p-value가 0.0007145로써 0.1보다 작으므로 귀무가설을 기각한다는 것을 알 수 있다.한편, 귀무가설에 대한 90% 신뢰구간은 (33.51136, 34.08464)이고 귀무가설의 값이 이 신뢰구간에 포함되지 않으므로 귀무가설이 기각됨을 확인할 수 있다.

b1-ch7-2.R의 실행결과

```
> library(foreign)

> time<-read.dta(file="http://kanggc.iptime.org/book/data/chap11-2-1.dta")

> time$var1
 [1] 33.46 33.38 32.73 32.15 33.99 34.10 33.97 34.34 33.95
[10] 33.85 34.23 34.05 34.13 34.45 34.19

> t.test(time$var1, mu=34.5, conf.level=0.9)

        One Sample t-test

data:  time$var1
t = -4.3136, df = 14, p-value = 0.0007145
alternative hypothesis: true mean is not equal to 34.5
90 percent confidence interval:
 33.51136 34.08464
sample estimates:
mean of x
   33.798


> (t14<-qt(0.05, df=14, lower.tail=T))
[1] -1.76131

> (t14<-qt(0.05, df=14, lower.tail=F))
[1] 1.76131
```

(2) 모분산에 대한 가설검정

(연습 2)

한 기술연구소에서 휴대전화 배터리 무게의 분산이 62g이라는 주장에 대한 양측검정을 하려고 한다. 휴대전화 7개를 무작위 선정하여 조사한 무게가 다음과 같으며, 휴대전화 배터리 무게는 정규분포한다고 하자. 5% 유의수준에서 휴대전화 배터리 무게의 분산이 62g이라는 귀무가설을 양측검정하라.

36	37	38	39	39	44	47

b1-ch7-3.R을 실행해 보면 계산된 검정통계량의 값은 1.548387이고 5% 유의수준 하에서 자유도가 인 χ^2-분포의 임계치는 1.237344와 14.44938이므로 5% 유의수준 하에서 모분산이 62g이라는 귀무가설을 허용한다. 또한 계산된 검정통계향의 p-value가 0.9562158로써 0.975보다 작으므로 귀무가설을 기각한다는 것을 확인할 수 있다.

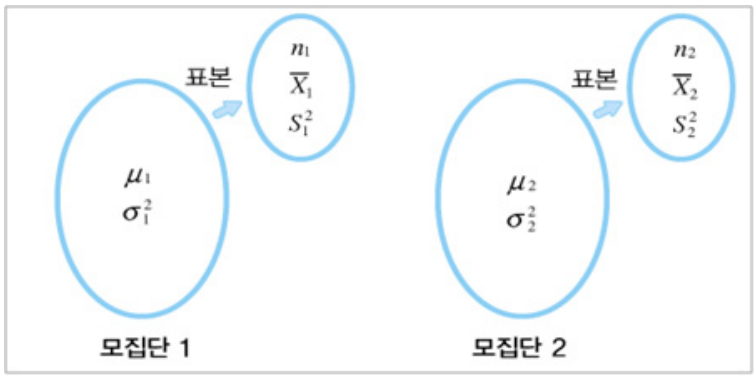
b1-ch7-3.R의 실행결과

```
> x<-c(36,37,38,39,39,44,47)

> xbar<-mean(x)

> xbar
[1] 40

> s.sq<-var(x)

> s.sq
[1] 16

> df<-6

> q=length(x)-1
```

```
> chi<-(df*s.sq)/62

> chi
[1] 1.548387

> pchisq(chi, df = q,lower.tail = F)
[1] 0.9562158

> UCV<-qchisq(0.025, df = q, lower.tail = F)

> UCV
[1] 14.44938

> LCV<-qchisq(0.975, df = q, lower.tail = F)

> LCV
[1] 1.237344
```

3. 두 집단에 대한 가설검정

사로 독립적인 두 모집단의 모수와 각각의 모집단에서 추출한 표본의 표본통계량은 다음과 같다.



모수와통계량 / 집단	모수		표본의 크기	통계량	
	모평균	모분산		표본평균	표본분산
집단 1	μ_1	σ_1^2	n_1	$\overline{X}_1$	s_1^2
집단 2	μ_2	σ_2^2	n_2	$\overline{X}_2$	s_2^2

(1) 모평균에 대한 가설검정

① 두 모집단의 분산이 알려져 있지 않지만 같다고 가정할 경우

이 경우 두 표본의 크기의 합이 30 이상이면 즉, $n_1+n_2 \geqq 30$ 이면 Z-검정통계량을 이용하면 되고, 두 표본의 크기의 합이 30 미만이면 즉, $n_1+n_2 < 30$ 이면 t-검정통계량은 다음과 같다.

$$t = \frac{\overline{X}_1 - \overline{X}_2}{S_p\sqrt{\frac{1}{n_1}+\frac{1}{n_2}}} \sim t_{(n_1+n_2-2)} \text{ , 단, } S_p = \sqrt{\frac{(n_{1-1})S_1^2+(n_{2-1})S_2^2}{n_1+n_2-2}}$$

(연습 3)

대기업과 중소기업에 근무하는 근로자의 직장 만족도가 다음과 같이 조사되었다. 두 모집단의 분산은 동일하다고 가정하고, 중소기업 근로자들의 평균 직장 만족도가 대기업 근로자보다 높다는 귀무가설($H_0 : \mu_1 \geqq \mu_2$) 및 대립가설($H_1 : \mu_1 < \mu_2$)을 설정하고 1% 유의수준에서 단측검정하라.

표본 1(중소기업 근로자, $n_1 = 10$)									
41	45	42	62	68	54	52	55	44	60
표본 1(대기업 근로자, $n_2 = 17$)									
74	74	70	52	76	91	71	78	76	78
83	50	52	66	65	53	72			

등분산 가정 하에 두 독립표본 평균차이($\mu_1 - \mu_2$)에 대한 가설검정을 하는 b1-ch7-4.R을 실행해 보면 계산된 검정통계량의 값은 -3.9421이고 1% 유의수준

하에서 자유도가 25인 t-분포의 임계치는 -2.485107과 2.485107이므로 1% 유의수준 하에서 귀무가설을 기각한다. 또한 계산된 검정통계향의 p-value가 0.0002874로써 0.01보다 작으므로 귀무가설을 기각한다는 것을 알 수 있다. 한편 귀무가설에 대한 99% 신뢰구간은 $(-\infty, -6.346238)$이고 계산된 검정통계량의 값은 이 신뢰구간에 포함되지 않으므로 귀무가설이 기각됨을 확인할 수 있다.

b1-ch7-4.R의 실행결과

```
> small<-c(41,45,42,62,68,54,52,55,44,60)

> large<-c(74,74,70,52,76,91,71,78,76,78,83,50,52,66,65,53,72)

> (mean(small))
[1] 52.3

> (mean(large))
[1] 69.47059

> (var(small))
[1] 85.12222

> (var(large))
[1] 138.7647

> t.test(small, large, alternative = "less", var.equal = T, conf.level = 0.99)

        Two Sample t-test

data:  small and large
t = -3.9421, df = 25, p-value = 0.0002874
alternative hypothesis: true difference in means is less than 0
99 percent confidence interval:
      -Inf -6.346238
sample estimates:
mean of x mean of y
 52.30000  69.47059
```

```
> (t25<-qt(0.01, df=25, lower.tail=F))
[1] 2.485107
```

② 두 모집단의 분산이 알려져 있지 않지만 다르다고 가정할 경우

이 경우 두 표본의 크기의 합이 30 이상이면 즉, $n_1+n_2 \geqq 30$ 이면 Z-검정통계량을 이용하면 되고, 두 표본의 크기의 합이 30 미만이면 즉, $n_1+n_2 < 30$ 이면 t-검정통계량은 다음과 같다.

$$t = \frac{\overline{X}_1 - \overline{X}_2}{\sqrt{\frac{S_1^2}{n_1} + \frac{S_2^2}{n_2}}} \sim t_{(n_1+n_2-2)}$$

(연습 4)

통계학을 수강하는 경제학과 학생과 경영학과 학생들의 중간고사 성적이 각각 다음과 같았다. 두 모집단의 분산은 다르다고 가정하고, 경제학과 학생과 경영학과 학생의 모평균이 동일하다는 귀무가설을 5% 유의수준에서 양측검정하라.

	학생 1	학생 2	학생 3	학생 4	학생 5	학생 5
경제학과	75	71	52	46	70	83
경영학과	82	73	59	48	68	93

두 모집단의 분산이 알려져 있지 않지만 다르다고 기정하고, 두 독립표본 평균차이 $(\mu_1-\mu_2)$에 대한 가설검정을 하는 b1-ch7-5.R을 실행해 보면 계산된 검정통계량의 값은 -0.4953이고 5% 유의수준 하에서 자유도가 10인 t-분포의 임계치는 -1.812461과 1.812461이므로 5% 유의수준 하에서 귀무가설을 기각할 수 없다. 또한 계산된 검정통계량의 p-value가 0.6313으로써 0.05보다 크므로 귀무가설을 허용한다는 것을 알 수 있다. 한편, 귀무가설에 대한 95% 신뢰구간은 (-23.86736, 15.20069)이고 귀무가설의 값 0이 이 신뢰구간에 포함되므로 귀무가설이 허용됨을 확인할 수 있다.

b1-ch7-5.R의 실행결과

```
> econ<-c(75,71,52,46,70,83)

> mgt<-c(82,73,59,48,68,93)

> t.test(econ, mgt, conf.level=0.95)

        Welch Two Sample t-test

data:  econ and mgt
t = -0.4953, df = 9.8507, p-value = 0.6313
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -23.86736  15.20069
sample estimates:
mean of x mean of y
 66.16667  70.50000


> (mean(econ))
[1] 66.16667

> (mean(mgt))
[1] 70.5

> (var(econ))
[1] 201.3667

> (var(mgt))
[1] 257.9

> (t10<-qt(0.05, df=10, lower.tail=F))
[1] 1.812461
```

③ 독립이 아닌 짝진 표본의 경우

쌍번호	1	2	3	4	...	n
X	x_1	x_2	x_3	x_4	...	x_n
Y	y_1	y_2	y_3	y_4	...	y_n
D	d_1	d_2	d_3	d_4	...	d_n

이 경우 n개의 쌍 $(x_1,y_1),(x_2,y_2),...,(x_n,y_n)$ 으로 관측된 표본에서 $d_i = x_i - y_i$, $i = 1,2,...,n$ 이라고 할 때, 두 집단 평균 μ_X, μ_Y의 동일성에 대한 검정은 $\mu_D = \mu_X - \mu_Y$ 이므로 귀무가설은 $H_0 : \mu_D = 0$ 와 같다.

이 경우 짝진 표본의 크기가 30 이상이면 Z-검정통계량을 이용하면 되고, 짝진 표본의 크기가 30 미만이면 t-검정통계량을 이용하면 된다.

짝진 표본의 차이를 나타내는 d_i의 평균은 $\bar{d} = \frac{1}{n}\sum_{i=1}^{n} d_i$ 이고, 표본분산은 $S_d^2 = \frac{1}{n-1}\sum_{i=1}^{n}(d_i - \bar{d})^2$ 라고 할 때, 귀무가설 하에서 검정통계량은 다음과 같다.

$$t = \frac{\bar{d}}{\frac{S_d}{\sqrt{n}}} \sim t_{(n-1)}$$

(연습 5)

통계학을 수강하는 경영학부 학생들을 대상으로 보충수업이 학생들에게 도움이 되는지 알아보기 위해 6명을 임의로 선정하였다. 보충수업 전에 시험을 보게 하고 보충수업을 수강한 후 다시 시험을 보게 하였으며, 그 결과는 다음과 같다. 보충수업이 학생들의 성적 향상에 도움이 되는지 5% 유의수준에서 검정하라.

학생	보충수업 전(X_1)	보충수업 후(X_2)	점수 차이($d = X_1 - X_2$)
1	75	82	-7
2	71	73	-2
3	52	59	-7
4	46	48	-2
5	70	69	1
6	83	93	-10

이 경우 귀무가설은 $H_0 : \mu_1 - \mu_2 = 0$ 이고, 대립가설은 $H_1 : \mu_1 < \mu_2$ 가 된다. 독립이 아닌 짝진 두 표본 평균차이($\mu_1 - \mu_2$)에 대한 가설검정을 하는 b1-ch7-6.R을 실행해 보면 계산된 검정통계량의 값은 -2.6656이고 5% 유의수준 하에서 자유도가 5인 t-분포의 임계치는 -2.015048과 2.015048이므로 5% 유의수준 하에서 귀무가설을 기각한다. 또한 계산된 검정통계량의 p-value가 0.02229로써 0.05보다 작으므로 귀무가설을 기각한다는 것을 알 수 있다. 한편 귀무가설에 대한 95% 신뢰구간은 $(-\infty, -1.098207)$이고, 귀무가설의 값 0은 이 신뢰구간에 포함되지 않으므로 귀무가설이 기각됨을 확인할 수 있다.

b1-ch7-6.R의 실행결과

```
> library(foreign)

> data<-read.dta(file="http://kanggc.iptime.org/book/data/chap11-3-2.dta")

> data$var1
[1] 75 71 52 46 70 83

> data$var2
[1] 82 73 59 48 69 93

> d<-data$var1-data$var2

> d
[1]  -7  -2  -7  -2   1 -10

> (var(d))
[1] 17.1

> t.test(d, mu=0, alternative="less", conf.level=0.95)

        One Sample t-test

data:  d
t = -2.6656, df = 5, p-value = 0.02229
alternative hypothesis: true mean is less than 0
```

```
95 percent confidence interval:
       -Inf -1.098207
sample estimates:
mean of x
      -4.5


> (t5<-qt(0.05, df=5, lower.tail=F))
[1] 2.015048
```

(2) 모분산에 대한 가설검정

두 집단의 모분산이 동일하다는 가설검정의 귀무가설은 $H_0 : \sigma_1^2 = \sigma_2^2$ 이고, 대립가설은 일반적으로 $H_1 : \sigma_1^2 > \sigma_2^2$ 의 단측검정을 한다.

귀무가설 하에서 검정통계량은 다음과 같다.

$$F = \frac{S_1^2}{S_2^2} \sim F_{(n_{1-1}, n_{2-1})}$$

(연습 6)

대기업과 중소기업에 근무하는 근로자의 직장 만족도가 다음과 같이 조사되었다. 두 모집단의 모분산은 동일성 여부를 5% 유의수준에서 단측검정하라.

표본 1(중소기업 근로자, $n_1 = 10$)									
41	45	42	62	68	54	52	55	44	60
표본 1(대기업 근로자, $n_2 = 17$)									
74	74	70	52	76	91	71	78	76	78
83	50	52	66	65	53	72			

두 독립표본의 동분산($\sigma_1^2 = \sigma_2^2$)에 대한 가설검정을 하는 b1-ch7-8.R을 실행해 보면 계산된 검정통계량의 값은 1.6302이고 5% 유의수준 하에서 분자의 자유도가 16

이고, 분모의 자유도가 9인 F-분포의 임계치는 2.988966이므로 5% 유의수준 하에서 귀무가설을 허용한다. 또한 계산된 검정통계량의 p-value가 0.231로써 0.05보다 크므로 귀무가설을 허용한다는 것을 알 수 있다. 한편 귀무가설에 대한 95% 신뢰구간은 (0.5454, ∞)이고, 귀무가설의 값 1은 이 신뢰구간에 포함되므로 귀무가설이 허용됨을 확인할 수 있다.

b1-ch7-8.R의 실행결과

```
> small<-c(41,45,42,62,68,54,52,55,44,60)

> large<-c(74,74,70,52,76,91,71,78,76,78,83,50,52,66,65,53,72)

> (mean(small))
[1] 52.3

> (mean(large))
[1] 69.47059

> (var(small))
[1] 85.12222

> (var(large))
[1] 138.7647

> var.test(large, small, alternative = "greater", conf.level = 0.95)

        F test to compare two variances

data:  large and small
F = 1.6302, num df = 16, denom df = 9, p-value =
0.231
alternative hypothesis: true ratio of variances is greater than 1
95 percent confidence interval:
 0.5454      Inf
sample estimates:
ratio of variances
```

```
              1.630182

> (f<-qf(0.05, df1 = 16, df2 = 9, lower.tail = F))
[1] 2.988966
```

참 고 문 헌

강기춘(2010), 『계량경제학 : 이론과 실습』, 온누리.

김영우(2017), 『Do it! 쉽게 배우는 R 데이터 분석』, 이지스퍼브리싱.

박범조(2013), 『응용 계량경제학 : R 활용』, 시그마프레스.

이용구 · 김삼용(2016), 『통계학의 이해 : EXCEL 실습 -제8판-』, 율곡출판사.

이재길(2017), 『R 프로그램에 기반한 시계열 자료 분석』, 황소걸음 아카데미.

폴 티터 지음 · 이제원 옮김(2012), 『R Cookbook : 데이터 분석과 통계 그래픽스를 위한 실전 예제』, 인사이트.

한치록(2017), 『계량경제학 강의』, 박영사.

Cowpertwait, P.S.P and A.V. Metcalfe(2009), *Introductory Time Series with R*, Springer.

Heiss, F.(2016), *Using R for Introductory Econometrics*, John Wiley & Sons,Inc.

Schmuller, J.(2017), *Statistical Analysis with R For Dummies*, John Wiley & Sons,Inc.

부록 1

R 코드

제1장 R 기본사용

b1-ch1-1.R

```
x<-c(1:10)
x
sort(x)
sort(x, decreasing=T)
mean(x)
median(x)
quantile(x)
diff(range(x))
var(x)
sd(x)
```

b1-ch1-2.R

```
a<-c(-3,-2,-1,1,2,3)
sum(a)
abs(a)
as<-a[4:6]
sqrt(as)
max(a)
min(a)
range(a)
exp(a)
log(as)
log10(as)
```

b1-ch1-3.R

```
x<-c(21,4,13,6,12,7,4,25,22)
y<-c(-2,4,-3,8,-7,8,-2,-6,5)
x;y
cov(x,y)
cor(x,y)
summary(x);summary(y)
cumsum(1:10);cumprod(1:10)
```

b1-ch1-4.R

```
#수익률 평균=40%,표준편차=10%인 정규분포에서 수익률이 60%보다 낮을 확률
pnorm(60,mean=40,sd=10)

#수익률 평균=40%,표준편차=10%인 정규분포에서 수익률이 60%보다 높은 확률
(표준화)
1-pnorm(2,0,1)

#P(Z<1.645)
pnorm(1.645, 0,1)

#P(Z<K)=0.95일 때, K의 값은?
qnorm(0.95, 0,1)

#t-통계량이 -3.271, n=16일 때 p의 값은?
pt(-3.271, 15)

#n=16일때, 5% 유의수준에서 기각역(단측)
qt(p=0.05, df=15)

round(rnorm(n=20, mean=40, sd=10), digits=2)
```

제2장 Data set

bi-ch2-1.R

```
sample1<-"http://kanggc.iptime.org/book/data/sample1.txt"
sample_dat<-read.delim(sample1,header = T)
sample_dat
year<-sample_dat$year
gdp<-sample_dat$GDP
consumption<-sample_dat$consumption
min(gdp)
max(consumption)
mean(gdp)
mean(consumption)
median(gdp)
median(consumption)
quantile(gdp)
quantile(consumption)
var(gdp)
var(consumption)
sd(gdp)
sd(consumption)
summary(sample_dat)
```

b1-ch2-2.R

```
sample1<-"http://kanggc.iptime.org/book/data/sample1.txt"
sample_dat<-  as.matrix(read.delim(sample1,header = T),ncol = 3)
sample_dat
year<-sample_dat[,1]
year
gdp<-sample_dat[,2]
gdp
```

```
consumption<-sample_dat[,3]
consumption
colMeans(sample_dat)
summary(sample_dat)
```

b1-ch2-3.R

```
sample1<-("http://kanggc.iptime.org/book/data/csv_sample1.csv")
sample_dat<-read.csv(sample1,header=T,sep=",")
sample_dat
year<-sample_dat$year
gdp<-sample_dat$GDP
consumption<-sample_dat$consumption
summary(sample_dat)
```

b1-ch2-4.R

```
library(openxlsx)
excel_sample1<-read.xlsx("http://kanggc.iptime.org/book/data/sample1-n.xlsx")
excel_sample1
excel_sample1_dat<- data.matrix(excel_sample1)
excel_sample1_dat
year<-excel_sample1_dat[,1]
gdp<-excel_sample1_dat[,2]
consumption<-excel_sample1_dat[,3]
summary(excel_sample1_dat)
```

b1-ch2-5.R

```
library(openxlsx)
excel_sample1<-read.xlsx("http://kanggc.iptime.org/book/data/sample1-n.xlsx")
excel_sample1_dat<- data.matrix(excel_sample1)
year<-excel_sample1_dat[,1]
```

```
gdp<-excel_sample1_dat[,2]
consumption<-excel_sample1_dat[,3]
lgdp<-log(gdp)
lconsumption<-log(consumption)
lgdp; lconsumption
names(excel_sample1)
excel_sample1
names(excel_sample1)[3]<-"cons"
excel_sample1
names(excel_sample1)<-c("T","Y","C")
excel_sample1
```

bi-ch2-6.R

```
library(openxlsx)
excel_sample1<-read.xlsx("http://kanggc.iptime.org/book/data/sample1-n.xlsx")
excel_sample1_dat<- data.matrix(excel_sample1)
year<-excel_sample1_dat[,1]
gdp<-excel_sample1_dat[,2]
consumption<-excel_sample1_dat[,3]

graphics.off()
par("mar")
par(mar=c(1,1,1,1))

y.ts<-ts(gdp, start=2000, end=2016, frequency=1)
c.ts<-ts(consumption, start=2000, end=2016, frequency=1)

lagy<-lag(y.ts, k=-1)
lagc<-lag(c.ts, k=-1)

gy<-(y.ts-lagy)/lagy
gc<-(c.ts-lagc)/lagc

ly.ts<-log(y.ts)
lc.ts<-log(c.ts)
```

```
gly<-ly.ts-lag(ly.ts, k=-1)
glc<-lc.ts-lag(lc.ts, k=-1)

(y<-cbind(gy, gly))
(c<-cbind(gc, glc))


par(mfrow=c(2,1))

plot(gy, type="l", col="red", main="Exact Growth Rate vs. Approx. Growth Rate of GDP")
lines(gly, lwd=3, lty=6, col="green")

plot(gc, type="l", col="red", main="Exact Growth Rate vs. Approx. Growth Rate of Consumption")
lines(glc, lwd=3, lty=6, col="green")
```

b1-ch2-7.R

```
library(openxlsx)
excel_sample1<-read.xlsx("http://kanggc.iptime.org/book/data/sample1-n.xlsx")
excel_sample1_dat<- data.matrix(excel_sample1)
year<-excel_sample1_dat[,1]
gdp<-excel_sample1_dat[,2]
consumption<-excel_sample1_dat[,3]
data1<-excel_sample1_dat[1:10,]
data1
data2<-excel_sample1_dat[11:17,]
data2
```

제3장 기본분석

b1-ch3-1.R

```
csv_sample1<-"http://kanggc.iptime.org/book/csv_sample1.csv"
csv_sample_dat<- as.matrix(read.csv(csv_sample1,header=T),ncol=3)
year<-csv_sample_dat[,1]
gdp<-csv_sample_dat[,2]
consumption<-csv_sample_dat[,3]
par(mfrow=c(2,1))
plot(year, gdp, type="l", main="GDP of Korea(2000-2016)")
plot(year, consumption, type="l", lty=2,main="Consumption of Korea
(2000-2016)")
```

b1-ch3-2.R

```
csv_sample1<-"http://kanggc.iptime.org/book/csv_sample1.csv"
csv_sample_dat<- as.matrix(read.csv(csv_sample1,header=T),ncol=3)
year<-csv_sample_dat[,1]
gdp<-csv_sample_dat[,2]
consumption<-csv_sample_dat[,3]
par(mfrow=c(1,2))
hist(gdp)
hist(consumption, breaks=8, col="red")
```

b1-ch3-3.R

```
csv_sample1<-"http://kanggc.iptime.org/book/csv_sample1.csv"
csv_sample_dat<- as.matrix(read.csv(csv_sample1,header=T),ncol=3)
year<-csv_sample_dat[,1]
gdp<-csv_sample_dat[,2]
```

```
consumption<-csv_sample_dat[,3]
plot(gdp, consumption, main = "Scatter plot of GDp and Consunption")
```

b1-ch3-4.R

```
library(openxlsx)

sample1<-read.xlsx("http://kanggc.iptime.org/book/data/stat-1.xlsx", sheet = 1,
startRow = 1, colNames = T)

mid<-sample1$mid
final<-sample1$final
total<-sample1$total
grade<-sample1$grade

summary(sample1)

par(mfrow = c(1,3))
boxplot(mid, main = "Box plot of mid")
boxplot(final, main = "Box plot of final")
boxplot(total, main = "Box plot of total")
```

b1-ch3-5.R

```
library(openxlsx)

sample1<-read.xlsx("http://kanggc.iptime.org/book/data/stat-1.xlsx", sheet = 1,
startRow = 1, colNames = T)

slices<-c(1,2,3,4,5,6,7,8)
lbls<-c("1등급","2등급","3등급","4등급","5등급","6등급","7등급","8등급")
pie(slices, labels = lbls, main = "Pie Chart of Total Score")
```

b1-ch3-6.R

```
library(openxlsx)
sample1<-read.xlsx("http://kanggc.iptime.org/book/data/stat-1.xlsx",  sheet = 1,
startRow = 1,  colNames  =  T)

mid<-sample1$mid
final<-sample1$final
total<-sample1$total
grade<-sample1$grade

counts<-table(total,  grade)
barplot(counts,  main = "Bar  Chart  of  Total  Score",  xlab = "Grade")
```

b1-ch3-7.R

```
library(openxlsx)
sample1<-read.xlsx("http://kanggc.iptime.org/book/data/stat-1.xlsx",  sheet = 1,
startRow = 1,  colNames  =  T)

mid<-sample1$mid
final<-sample1$final
total<-sample1$total
grade<-sample1$grade

total
stem(total)
stem(total,  scale = 0.5)
stem(total,  scale = 2)
```

b1-ch3-8.R

```
install.packages("ggplot2")
library(ggplot2)
con1<-url("http://kanggc.iptime.org/book/data/Map.RData")
```

```
con2<-url("http://kanggc.iptime.org/book/data/mapJeju.RData")
load(con1)
load(con2)
ggplot() + geom_path(data = map, aes(x = long, y = lat, group = group))
ggplot() + geom_path(data = map.jeju, aes(x = long, y = lat, group = group))
```

b1-ch3-9.R

```
install.packages("prettyR")
library(prettyR)
ch2_1<-scan("http://kanggc.iptime.org/book/data/ch2_1.txt")
ch2_1
max(ch2_1)
min(ch2_1)
mean(ch2_1)
diff(range(ch2_1))
var(ch2_1)
sd(ch2_1)
table(ch2_1)[table(ch2_1)[1:length(unique(ch2_1))] == max(table(ch2_1))]
median(ch2_1)
Mode(ch2_1)
bins<-c(0, 12, 24, 36, 48, 60, 72, 84, 97)
class<-cut(ch2_1, breaks = bins)
table(class)
table(class)/length(ch2_1)
transform(table(class), Rel_Freq = prop.table(Freq))
hist(ch2_1)
hist(ch2_1, breaks = bins, main = "Test Scores", xlab = "Score")
summary(ch2_1)
수치<-c(70, 17, 8, 3, 2)
경제책임<-c("정부", "대기업", "금융기관", "근로자", "모두")
경제책임<-paste(경제책임, 수치)
경제책임<-paste(경제책임, "%", sep = " ")
pie(수치, labels = 경제책임, col = rainbow(length(경제책임)), main = "경제위기의 책
임")
```

bi-ch3-10.R

```
a<-c(10, 2, 19, 24, 6, 23, 47, 24, 54, 77)
n<-length(a) # now n is equal to the number of elements in a

mean(a)
1/mean(1/a) # compute the harmonic mean
prod(a)^(1/n) # compute the geometric mean

b<-c(30,60)
mean(b)
1/mean(1/b)

c<-c(120,130,118)
m<-length(c)
round(mean(c),digits=2)
round(prod(c)^(1/m),digits=2)

g<-((9204/5000)^(1/3)-1)*100
round(g, digits=2)
```

b1-ch3-11.R

```
data<-c(0.167, 0.083, 0.083, 0.333, 0.083, 0.5, 0.083, 0.667, 0.25, 0.583, 0.167, 1)
mat<-matrix(data, nrow=3, byrow=T)
rownames(mat)<-c("1", "3", "합계")
colnames(mat)<-c("0", "1", "2", "합계")
mat
mu_x<-1*mat[1,4] + 3*mat[2,4]
mu_y<-0*mat[3,1] + 1*mat[3,2] + 2*mat[3,3]
mu_x
mu_y
var_x<-1^2*mat[1,4] + 3^2*mat[2,4]- mu_x^2
var_y<-0^2*mat[3,1] + 1^2*mat[3,2] + 2^2*mat[3,3]-mu_y^2
var_x
var_y
p_xy<-c(0.167, 0.083, 0.083, 0.083, 0.5, 0.083)
xy<-c(0,1,2,0,3,6)
```

```
cov_xy<-sum(p_xy*xy)-(mu_x*mu_y)
cov_xy
corr_xy<-cov_xy/sqrt(var_x*var_y)
corr_xy
```

제4장 이론적 확률분포

b1-ch4-1.R

```
set.seed(12345)

x<-rbinom(1000, 1, .5)
(table(x))
(mean(x))

cx<-cumsum(x)

heads<-numeric(1000)

for (i in 1:1000) {
    heads[i]<-cx[i]/i
}

plot(heads, type="l", xlab="Number of Trials", ylab="Proportion of Heads",
ylim=c(0,1), main="Proportion of Heads")
```

b1-ch4-2.R

```
set.seed(12345)
```

```
r<-10000

binom1<-rbinom(r, 10, 0.1)
binom2<-rbinom(r, 10, 0.5)
binom3<-rbinom(r, 20, 0.5)

par(mfrow=c(1,3))

hist(binom1, breaks=100, xlab="n=10 p=0.1")
hist(binom2, breaks=100, xlab="n=10 p=0.5")
hist(binom3, breaks=100, xlab="n=20 p=0.5")
```

b1-ch4-3.R

```
par(mfrow=c(2,2))

n<-25 # 시행횟수

p_list<-c(0.1, 0.2, 0.5, 0.7) # 발생확률

for (i in 1:length(p_list)) {
    p_x<-dbinom(x=1:n, n, p_list[i])
    plot(x=1:n, p_x, xlab="x", ylab="P(X=x)", ylim=c(0, 0.3), xlim=c(1,n),
main=paste("p=", p_list[i]))
    x_seq<-seq(1,n,1)
    lines(x_seq, p_x, type="h", col="blue")
}
```

b1-ch4-4.R

```
binom11<-rep(NA,5)
binom12<-rep(NA,5)
binom13<-rep(NA,5)
binom14<-rep(NA,5)
binom15<-rep(NA,5)
binom16<-rep(NA,5)
```

```
binom17<-rep(NA,5)
binom18<-rep(NA,5)
binom19<-rep(NA,5)

binom11[1]<-pbinom(0, 5, 0.1)
binom12[1]<-pbinom(0, 5, 0.2)
binom13[1]<-pbinom(0, 5, 0.3)
binom14[1]<-pbinom(0, 5, 0.4)
binom15[1]<-pbinom(0, 5, 0.5)
binom16[1]<-pbinom(0, 5, 0.6)
binom17[1]<-pbinom(0, 5, 0.7)
binom18[1]<-pbinom(0, 5, 0.8)
binom19[1]<-pbinom(0, 5, 0.9)

for(i in 2:5) {
  binom11[i]<-pbinom(i-1, 5, 0.1)
}
for(i in 2:5) {
  binom12[i]<-pbinom(i-1, 5, 0.2)
}
for(i in 2:5) {
  binom13[i]<-pbinom(i-1, 5, 0.3)
}
for(i in 2:5) {
  binom14[i]<-pbinom(i-1, 5, 0.4)
}
for(i in 2:5) {
  binom15[i]<-pbinom(i-1, 5, 0.5)
}
for(i in 2:5) {
  binom16[i]<-pbinom(i-1, 5, 0.6)
}
for(i in 2:5) {
  binom17[i]<-pbinom(i-1, 5, 0.7)
}
for(i in 2:5) {
  binom18[i]<-pbinom(i-1, 5, 0.8)
```

```
}
for(i in 2:5) {
  binom19[i]<-pbinom(i-1, 5, 0.9)
}

round((binom<-cbind(binom11,binom12, binom13,binom14,binom15,
binom16,binom17,binom18, binom19)),digits=3)
```

b1-ch4-5.R

```
set.seed(12345)

n<-10000;

poi025<-rpois(n, 0.25)
poi1<-rpois(n, 1)
poi2<-rpois(n, 2)
poi4<-rpois(n, 4)

par(mfrow=c(2,2))

hist(poi025, breaks=100, xlab="mu=0.25")
hist(poi1, breaks=100, xlab="mu=1.0")
hist(poi2, breaks=100, xlab="mu=2.0")
hist(poi4, breaks=100, xlab="mu=4.0")
```

b1-ch4-6.R

```
par(mfrow=c(2,2))

lambda_list<-c(3, 5, 10, 15) # 평균
x_list<-30 # 발생횟수를 1부터 x축에 보여줄 최대값

for (i in 1:length(lambda_list)) {
```

```
    p_x<-dpois(x=0:x_list,lambda_list[i])
    plot(x=0:x_list, p_x, xlab="x", ylab="P(X=x)", ylim=c(0, 0.25),
         xlim=c(0,x_list), main=paste("lambda=", lambda_list[i]))
    x_seq<-seq(0,x_list,1)
    lines(x_seq, p_x, type="h", col="blue")
}
```

b1-ch4-7.R

```
poi11<-rep(NA,5)
poi12<-rep(NA,5)
poi13<-rep(NA,5)
poi14<-rep(NA,5)
poi15<-rep(NA,5)
poi16<-rep(NA,5)
poi17<-rep(NA,5)
poi18<-rep(NA,5)

poi11[1]<-ppois(0, 0.02)
poi12[1]<-ppois(0, 0.04)
poi13[1]<-ppois(0, 0.06)
poi14[1]<-ppois(0, 0.08)
poi15[1]<-ppois(0, 0.1)
poi16[1]<-ppois(0, 0.2)
poi17[1]<-ppois(0, 0.3)
poi18[1]<-ppois(0, 0.4)

for(i in 2:5) {
  poi11[i]<-ppois(i-1, 0.02)
}
for(i in 2:5) {
  poi12[i]<-ppois(i-1, 0.04)
}
for(i in 2:5) {
  poi13[i]<-ppois(i-1, 0.06)
}
```

```
for(i in 2:5) {
  poi14[i]<-ppois(i-1, 0.08)
}
for(i in 2:5) {
  poi15[i]<-ppois(i-1, 0.1)
}
for(i in 2:5) {
  poi16[i]<-ppois(i-1, 0.2)
}
for(i in 2:5) {
  poi17[i]<-ppois(i-1, 0.3)
}
for(i in 2:5) {
  poi18[i]<-ppois(i-1, 0.4)
}

round((poi<-cbind(poi11,poi12,poi13,poi14,poi15,poi16,poi17,poi18)),digits=3)
```

b1-ch4-8.R

```
set.seed(12345)
n<-10000;

min_list<-c(1,2,4,5)
max_list<-c(2,4,8,10)
par(mfrow=c(2,2))

unif1<-runif(n, min=1, max=2)
unif2<-runif(n, min=2, max=4)
unif3<-runif(n, min=4, max=8)
unif4<-runif(n, min=5, max=10)

(munif1<-mean(unif1))
(munif2<-mean(unif2))
(munif3<-mean(unif3))
(munif4<-mean(unif4))
```

```
(vunif1<-var(unif1))
(vunif2<-var(unif2))
(vunif3<-var(unif3))
(vunif4<-var(unif4))

for (i in 1:length(min_list)) {
  hist(runif(n, min=min_list[i], max=max_list[i]), freq=F, breaks=100,
xlab="x", main=paste("min =", min_list[i], " max =", max_list[i]))
 }
```

b1-ch4-9.R

```
z00<-rep(NA,10)
z01<-rep(NA,10)
z02<-rep(NA,10)
z03<-rep(NA,10)
z04<-rep(NA,10)

for(i in 1:10) {
  z00[i]<-pnorm((i-1)/100, 0, 1)-0.5
}
(z00<-round(z00, digits=4))

for(i in 10:19) {
  z01[i]<-pnorm(i/100, 0, 1)-0.5
}
(z01<-round(z01[10:19], digits=4))

for(i in 20:29) {
  z02[i]<-pnorm(i/100, 0, 1)-0.5
}
(z02<-round(z02[20:29], digits=4))

for(i in 30:39) {
  z03[i]<-pnorm(i/100, 0, 1)-0.5
```

```
}
(z03<-round(z03[30:39], digits=4))

for(i in 40:49) {
  z04[i]<-pnorm(i/100, 0, 1)-0.5
}
(z04<-round(z04[40:49], digits=4))

zdist<-rbind(z00,z01,z02,z03,z04)
(zdist<-round(zdist, digits=4))
```

b1-ch4-10.R

```
set.seed(12345)

n<-10000;

z1<-rnorm(n,0,1)
z2<-rnorm(n,0,1)
z3<-rnorm(n,0,1)
z4<-rnorm(n,0,1)
z5<-rnorm(n,0,1)

chi5<-z1^2+z2^2+z3^2+z4^2+z5^2

hist(chi5,  freq=F, col="grey", xlab="", xlim=c(0, 25), breaks=100)
par(new=T)
plot(density(chi5), axes=F, main="", xlim=c(0, 25), lwd=2, col="blue")
```

b1-ch4-11.R

```
set.seed(12345)

n<-10000;
```

```
df_list<-c(5,10,20,30)
par(mfrow=c(2,2))

for (i in 1:length(df_list)) {
  hist(rchisq(n, df=df_list[i], ncp=0), breaks=100, xlab="chisq",
main=paste("df=", df_list[i]))
}
```

b1-ch4-12.R

```
n_list<-c(2,5,7,9) # 표본수(n)
df_list<-n_list-1 # 자유도

curve(dchisq(x, 1, ncp=0), add=T, col="blue", xlim=c(0, 16), ylim=c(0, 0.8),
xlab="chisq", ylab="f(chisq)")
curve(dchisq(x, 4, ncp=0), add=T, col="red", xlim=c(0, 16), ylim=c(0, 0.8),
xlab="chisq", ylab="f(chisq)")
curve(dchisq(x, 6, ncp=0), add=T, col="green", xlim=c(0, 16), ylim=c(0,
0.8), xlab="chisq", ylab="f(chisq)")
curve(dchisq(x, 8, ncp=0), add=T, col="black", xlim=c(0, 16), ylim=c(0,
0.8), xlab="chisq", ylab="f(chisq)")

par(mfrow=c(2,2))

for (i in 1:length(df_list)) {

  curve(dchisq(x, df_list[i], ncp=0), add=F, xlim=c(0, 16), ylim=c(0, 0.8),
xlab="chisq", ylab="f(chisq)", main=paste("df=", df_list[i]))

}
```

b1-ch4-13.R

```
df<-30

chi1<-numeric(df)
chi2<-numeric(df)
chi3<-numeric(df)
chi4<-numeric(df)
chi5<-numeric(df)
chi6<-numeric(df)

 for(j in 1:df) {
  chi1[j]<-qchisq(0.01,j)
 }

for(j in 1:df) {
  chi2[j]<-qchisq(0.05,j)
}

for(j in 1:df) {
  chi3[j]<-qchisq(0.1,j)
}

for(j in 1:df) {
  chi4[j]<-qchisq(0.9,j)
}

for(j in 1:df) {
  chi5[j]<-qchisq(0.95,j)
}

for(j in 1:df) {
  chi6[j]<-qchisq(0.99,j)
}

round((chi<-cbind(chi1,chi2,chi3,chi4,chi5,  chi6)),digits=4)
```

b1-ch4-14.R

```
set.seed(12345)

n<-10000;

z<-rnorm(n,0,1)
z1<-rnorm(n,0,1)
z2<-rnorm(n,0,1)
z3<-rnorm(n,0,1)
z4<-rnorm(n,0,1)
z5<-rnorm(n,0,1)

chi5<-z1^2+z2^2+z3^2+z4^2+z5^2

sqchi5<-sqrt(chi5/5)

t5<-z/sqchi5

hist(t5, breaks=100)
```

b1-ch4-15.R

```
set.seed(12345)

n<-10000;
df_list<-c(5,10,15,30)
par(mfrow=c(2,2))

for (i in 1:length(df_list)) {
  hist(rt(n, df=df_list[i], ncp=0), breaks=100, xlab="t", main=paste("df=",
df_list[i]))
 }
```

b1-ch4-16.R

```
n_list<-c(2,5,10,30) # 표본수(n)
df_list<-n_list-1 # 자유도

curve(dt(x, 1, ncp=0), add=T, col="blue", xlim=c(-4, 4), ylim=c(0, 0.5),
xlab="t", ylab="f(t)")
curve(dt(x, 4, ncp=0), add=T, col="red", xlim=c(-4, 4), ylim=c(0, 0.5),
xlab="t", ylab="f(t)")
curve(dt(x, 9, ncp=0), add=T, col="green", xlim=c(-4, 4), ylim=c(0, 0.5),
xlab="t", ylab="f(t)")
curve(dt(x, 29, ncp=0), add=T, col="black", xlim=c(-4, 4), ylim=c(0, 0.5),
xlab="t", ylab="f(t)")

par(mfrow=c(2,2))

for (i in 1:length(df_list)) {

  curve(dt(x, df_list[i], ncp=0), xlim=c(-4, 4), ylim=c(0, 0.5), xlab="t",
ylab="f(t)", main=paste("df=", df_list[i]))

  }
```

b1-ch4-17.R

```
t11<-rep(NA,9)
t12<-rep(NA,9)
t13<-rep(NA,9)
t14<-rep(NA,9)
t15<-rep(NA,9)

for(i in 1:9) {
  t11[i]<-qt(0.9, i)
}
for(i in 1:9) {
  t12[i]<-qt(0.95, i)
}
```

```
for(i in 1:9) {
  t13[i]<-qt(0.975,i)
}
for(i in 1:9) {
  t14[i]<-qt(0.99, i)
}
for(i in 1:9) {
  t15[i]<-qt(0.995, i)
}

round((poi<-cbind(t11,t12,t13,t14,t15)), digits=3)
```

b1-ch4-18.R

```
set.seed(12345)

n<-10000;

z1<-rnorm(n,0,1)
z2<-rnorm(n,0,1)
z3<-rnorm(n,0,1)
z4<-rnorm(n,0,1)
z5<-rnorm(n,0,1)
z6<-rnorm(n,0,1)
z7<-rnorm(n,0,1)
z8<-rnorm(n,0,1)
z9<-rnorm(n,0,1)
z10<-rnorm(n,0,1)

chi15<-z1^2+z2^2+z3^2+z4^2+z5^2
chi25<-z6^2+z7^2+z8^2+z9^2+z10^2

f55<-(chi15/5)/(chi25/5)

hist(f55, breaks=100)
```

b1-ch4-19.R

```
set.seed(12345)

n<-10000;
df1_list<-c(5,9,15,38)
df2_list<-c(10,10,20,40)
par(mfrow=c(2,2))

for (i in 1:length(df_list)) {
  hist(rf(n, df1=df1_list[i], df2=df2_list[i], ncp=0), breaks=100, xlab="f",
main=paste("df1=", df1_list[i], "df2=", df2_list[i]))
}
```

b1-ch4-20.R)

```
curve(df(x, 3, 15, ncp=0), add=T, col="blue", xlim=c(0,5), ylim=c(0,1),
xlab="f", ylab="f(f)")
curve(df(x, 5, 15, ncp=0), add=T, col="red", xlim=c(0,5), ylim=c(0,1),
xlab="f", ylab="f(f)")
curve(df(x, 10, 15, ncp=0), add=T, col="green", xlim=c(0,5), ylim=c(0,1),
xlab="f", ylab="f(f)")
curve(df(x, 15, 15, ncp=0), add=T, col="black", xlim=c(0,5), ylim=c(0,1),
xlab="f", ylab="f(f)")

df1_list<-c(3,5,10,15)
df2_list<-c(15,15,15,15)
par(mfrow=c(2,2))


for (i in 1:length(df1_list)) {
  curve(df(x, df1=df1_list[i], df2=df2_list[i], ncp=0), xlim=c(0,5),
ylim=c(0,1), xlab="f", ylab="f(f)", main=paste("df1=", df1_list[i], "df2=",
df2_list[i]))
}
```

b1-ch4-21.R

```
f11<-rep(NA,10)
f12<-rep(NA,10)
f13<-rep(NA,10)
f14<-rep(NA,10)
f15<-rep(NA,10)
f16<-rep(NA,10)
f17<-rep(NA,10)
f18<-rep(NA,10)
f19<-rep(NA,10)
f110<-rep(NA,10)

for(i in 1:10) {
  f11[i]<-qf(0.95, 1, i)
}
for(i in 1:10) {
  f12[i]<-qf(0.95, 2, i)
}
for(i in 1:10) {
  f13[i]<-qf(0.95, 3, i)
}
for(i in 1:10) {
  f14[i]<-qf(0.95, 4, i)
}
for(i in 1:10) {
  f15[i]<-qf(0.95, 5, i)
}
for(i in 1:10) {
  f16[i]<-qf(0.95, 6, i)
}
for(i in 1:10) {
  f17[i]<-qf(0.95, 7, i)
}
for(i in 1:10) {
  f18[i]<-qf(0.95, 8, i)
}
for(i in 1:10) {
```

```
  f19[i]<-qf(0.95, 9, i)
}
for(i in 1:10) {
  f110[i]<-qf(0.95, 10, i)
}

round((poi<-cbind(f11,f12,f13,f14,f15,f16,f17,f18,f19,f110)), digits=2)
```

제5장 표본분포

b1-ch5-1.R

```
sampling.dist.1<-NULL
for(sample.count in 1:10000){
  set.seed(sample.count)
  sample.mean.1<-mean(rnorm(2,2.5,1.118))
  sampling.dist.1<-c(sampling.dist.1, sample.mean.1)
}
mean(sampling.dist.1)
var(sampling.dist.1)

hist(sampling.dist.1, freq=F, col="grey", xlab="", xlim=c(-1, 6),
breaks=100)
par(new=T)
plot(density(sampling.dist.1), axes=F, main="", xlim=c(-1, 6), lwd=2,
col="blue")
```

b1-ch5-2.R

```
sampling.dist.1<-NULL
```

```
for(sample.count in 1:10000){
  set.seed(sample.count)
  sample.mean.1<-mean(rnorm(2,2.5,1.118))
  sampling.dist.1<-c(sampling.dist.1, sample.mean.1)
}
mean(sampling.dist.1)
var(sampling.dist.1)
table(round(sampling.dist.1))

sampling.dist.2<-NULL
for(sample.count in 1:10000){
  set.seed(sample.count)
  sample.mean.2<-mean(rnorm(5,2.5,1.118))
  sampling.dist.2<-c(sampling.dist.2, sample.mean.2)
}
mean(sampling.dist.2)
var(sampling.dist.2)
table(round(sampling.dist.2))

sampling.dist.3<-NULL
for(sample.count in 1:10000){
  set.seed(sample.count)
  sample.mean.3<-mean(rnorm(10,2.5,1.118))
  sampling.dist.3<-c(sampling.dist.3, sample.mean.3)
}
mean(sampling.dist.3)
var(sampling.dist.3)
table(round(sampling.dist.3))

sampling.dist.4<-NULL
for(sample.count in 1:10000){
  set.seed(sample.count)
  sample.mean.4<-mean(rnorm(30,2.5,1.118))
  sampling.dist.4<-c(sampling.dist.4, sample.mean.4)
}
mean(sampling.dist.4)
var(sampling.dist.4)
```

```
table(round(sampling.dist.4))

par(mfrow = c(2,2))

hist(sampling.dist.1, freq = F, col = "grey", xlab = "", xlim = c(-1, 6), breaks = 100)
par(new = T)
plot(density(sampling.dist.1), axes = F, main = "", xlim = c(-1, 6), lwd = 2, col = "blue")

hist(sampling.dist.2, freq = F, col = "grey", xlab = "", xlim = c(-1, 6), breaks = 100)
par(new = T)
plot(density(sampling.dist.2), axes = F, main = "", xlim = c(-1, 6), lwd = 2, col = "blue")

hist(sampling.dist.3, freq = F, col = "grey", xlab = "", xlim = c(-1, 6), breaks = 100)
par(new = T)
plot(density(sampling.dist.3), axes = F, main = "", xlim = c(-1, 6), lwd = 2, col = "blue")

hist(sampling.dist.4, freq = F, col = "grey", xlab = "", xlim = c(-1, 6), breaks = 100)
par(new = T)
plot(density(sampling.dist.4), axes = F, main = "", xlim = c(-1, 6), lwd = 2, col = "blue")
```

b1-ch5-3.R

```
#sampling distribution of sample mean of 1000 samples with 11 size samples from uniform dist.

set.seed(23456789)

sample_size <- 11
```

```
min <- 0
max <- 1
n_rep <- 1000

sample_mean <- rep(NA, n_rep)
sample_var <- rep(NA, n_rep)

#graphics.off()
#par(mfrow=c(1,2))

for (i in 1:n_rep) {
       my_sample <- runif(sample_size,min,max)
       sample_mean[i] <- mean(my_sample)
       sample_var[i] <- var(my_sample)
    }

(mean(sample_mean))
(var(sample_mean))

hist(sample_mean, breaks=40, prob=T, main=paste( "samples of size 11"
),col="black")
par(new=T)
plot(density(sample_mean), xlab="", axes=F, main="", col="blue")
```

b1-ch5-4.R

```
set.seed(123456)

curve(dnorm(x,10,2), xlim = c(4, 16), ylim = c(0.0, 0.22))

par(mfrow=c(2,2))

ybar5<-numeric(10000)
for(j in 1:10000) {
  sample<-rnorm(5,10,2)
```

```
  ybar5[j] <-mean(sample)
}
mean(ybar5)
var(ybar5)
plot(density(ybar5), xlim = c(7, 13), ylim = c(0.0, 0.5))
curve(dnorm(x,10,sqrt(0.78186)), add=T, lty=2)

ybar10<-numeric(10000)
for(k in 1:10000) {
  sample<-rnorm(10,10,2)
  ybar10[k] <-mean(sample)
}
mean(ybar10)
var(ybar10)
plot(density(ybar10),xlim = c(7.5, 12.5), ylim = c(0.0, 0.8))
curve(dnorm(x,10,sqrt(0.39928)), add=T, lty=2)


ybar20<-numeric(10000)
for(m in 1:10000) {T
  sample<-rnorm(20,10,2)
  ybar20[m] <-mean(sample)
}
mean(ybar20)
var(ybar20)
plot(density(ybar20),xlim = c(8.5, 11.5), ylim = c(0.0, 1.0))
curve(dnorm(x,10,sqrt(0.20396)), add=T, lty=2)

ybar30<-numeric(10000)
for(n in 1:10000) {
  sample<-rnorm(30,10,2)
  ybar30[n] <-mean(sample)
}
mean(ybar30)
var(ybar30)
plot(density(ybar30),xlim = c(8.5, 11.5), ylim = c(0.0, 1.2))
curve(dnorm(x,10,sqrt(0.13477)), add=T, lty=2)
```

b1-ch5-5.R

```
set.seed(123456)

curve(dchisq(x,1))

par(mfrow=c(2,2))

ybar2<-numeric(10000)
for(j in 1:10000) {
  sample<-rchisq(2,1)
  ybar2[j] <-mean(sample)
}
mean(ybar2)
var(ybar2)
plot(density(ybar2), xlim = c(0, 10), ylim = c(0.0, 0.8))
curve(dnorm(x,1,sqrt(0.98413)), add=T, lty=2)

ybar10<-numeric(10000)
for(k in 1:10000) {
  sample<-rchisq(10,1)
  ybar10[k] <-mean(sample)
}
mean(ybar10)
var(ybar10)
plot(density(ybar10),xlim = c(0, 4), ylim = c(0.0, 1))
curve(dnorm(x,1,sqrt(0.20167)), add=T, lty=2)

ybar20<-numeric(10000)
for(n in 1:10000) {
  sample<-rchisq(20,1)
  ybar20[n] <-mean(sample)
}
mean(ybar20)
var(ybar20)
plot(density(ybar20),xlim = c(0, 3), ylim = c(0.0, 1.5))
curve(dnorm(x,1,sqrt(0.10241)), add=T, lty=2)
```

```
ybar30<-numeric(10000)
for(m in 1:10000) {
  sample<-rchisq(30,1)
  ybar30[m] <-mean(sample)
}
mean(ybar30)
var(ybar30)
plot(density(ybar30),xlim = c(0.3, 1.7), ylim = c(0.0, 2.0))
curve(dnorm(x,1,sqrt(0.06725)), add=T, lty=2)
```

b1-ch5-6.R

```
set.seed(123456)

curve(dunif(x,min=0, max=1))

par(mfrow=c(2,2))

ybar2<-numeric(10000)
for(j in 1:10000) {
  sample<-runif(2,min=0, max=1)
  ybar2[j] <-mean(sample)
}
mean(ybar2)
var(ybar2)
plot(density(ybar2), xlim = c(0, 1), ylim = c(0,3.5))
curve(dnorm(x,0.5,sqrt(0.04193)), add=T, lty=2)

ybar10<-numeric(10000)
for(k in 1:10000) {
  sample<-runif(10,min=0, max=1)
  ybar10[k] <-mean(sample)
}
mean(ybar10)
var(ybar10)
plot(density(ybar10),xlim = c(0, 1), ylim = c(0,4.5))
curve(dnorm(x,0.5,sqrt(0.00829)), add=T, lty=2)
```

```
ybar20<-numeric(10000)
for(m in 1:10000) {
  sample<-runif(20,min=0, max=1)
  ybar20[m] <-mean(sample)
}
mean(ybar20)
var(ybar20)
plot(density(ybar20),xlim = c(0, 1), ylim = c(0, 6.5))
curve(dnorm(x,0.5,sqrt(0.00411)), add=T, lty=2)

ybar30<-numeric(10000)
for(n in 1:10000) {
  sample<-runif(30,min=0, max=1)
  ybar30[n] <-mean(sample)
}
mean(ybar30)
var(ybar30)
plot(density(ybar30),xlim = c(0, 1), ylim = c(0, 7.7))
curve(dnorm(x,0.5,sqrt(0.00283)), add=T, lty=2)
```

b1-ch5-7.R

```
sampling.dist.1<-NULL
for(sample.count in 1:10000){
  set.seed(sample.count)
  sample.var.1<-mean(rnorm(2,2.5,1.118))
  sampling.dist.1<-c(sampling.dist.1, sample.var.1)
}
mean(sampling.dist.1)
var(sampling.dist.1)

hist(sampling.dist.1, freq=F, col="grey", xlab="", xlim=c(-1, 6),
breaks=100)
par(new=T)
plot(density(sampling.dist.1), axes=F, main="", xlim=c(-1, 6), lwd=2,
col="blue")
```

b1-ch5-8.R

```
#sampling distribution of sample mean of 1000 samples with 11 size samples from normal dist.

set.seed(23456789)

sample_size <- 11

n_rep <- 1000

sample_mean <- rep(NA, n_rep)
sample_var <- rep(NA, n_rep)

#graphics.off()
#par(mfrow=c(1,2))

for (i in 1:n_rep) {
      my_sample <- rnorm(sample_size,10,2)
      sample_mean[i] <- mean(my_sample)
      sample_var[i] <- var(my_sample)
   }

(mean(sample_mean))
(var(sample_mean))

hist(sample_var, breaks=40, prob=T, main=paste( "samples of size 11" ),col="black")
par(new=T)
plot(density(sample_var), xlab="", axes=F, main="", col="blue")
```

b1-ch5-9.R

```
set.seed(123456)

par(mfrow=c(2,2))
```

```
vbar10<-numeric(10000)
for(j in 1:10000) {
  sample10<-rnorm(10,10,2)
  vbar10[j] <-var(sample10)
}

(mean(vbar10))
(var(vbar10))

hist(vbar10, freq=F, xlab="", breaks=100)
par(new=T)
plot(density(vbar10), axes=F, main="", col="blue")

vbar20<-numeric(10000)
for(j in 1:10000) {
  sample20<-rnorm(20,10,2)
  vbar20[j] <-var(sample20)
}

(mean(vbar20))
(var(vbar20))


hist(vbar20, freq=F, xlab="", breaks=100)
par(new=T)
plot(density(vbar20), axes=F, main="", col="blue")

vbar30<-numeric(10000)
for(j in 1:10000) {
  sample30<-rnorm(30,10,2)
  vbar30[j] <-var(sample30)
}

(mean(vbar30))
(var(vbar30))

hist(vbar30, freq=F, xlab="", breaks=100)
```

```
par(new = T)
plot(density(vbar30), axes = F, main = "", col = "blue")

vbar100<-numeric(10000)
for(j in 1:10000) {
  sample100<-rnorm(100,10,2)
  vbar100[j] <-var(sample100)
}

(mean(vbar100))
(var(vbar100))

hist(vbar100, freq = F, xlab = "", breaks = 100)
par(new = T)
plot(density(vbar100), axes = F, main = "", col = "blue")
```

제6장 추 정

b1-ch6-1.R

```
set.seed(12343)

#par(mfrow = c(1,2))

CIlower<-numeric(100)
CIupper<-numeric(100)
pvalue1<-numeric(100)

for(j in 1:100) {
  sample<-rnorm(80,10,2)
  testres1<-t.test(sample,mu = 10)
  CIlower[j]<-testres1$conf.int[1]
```

```
  CIupper[j]<-testres1$conf.int[2]
  pvalue1[j]<-testres1$p.value
    }
testres1$conf.int[1]
testres1$conf.int[2]
testres1$p.value

reject1<-pvalue1<=0.05
table(reject1)

color<-rep(gray(.7),100)
color[reject1]<-"black"

plot(0, xlim=c(9,11), ylim=c(1,100), ylab="Sample No.", xlab="", main="95% Confidence Interval")
abline(v=10, lty=2)
for(j in 1:100) {
  lines(c(CIlower[j], CIupper[j]), c(j,j), col=color[j], lwd=1)
}
```

b1-ch6-2.R

```
data1<-"http://kanggc.iptime.org/book/data/chap10-2.csv"

data1_dat<-as.matrix(read.csv(data1,header=T), ncol=1)

var1<-data1_dat[,1]

xbar=mean(var1)

z<-qnorm(0.025, 0, 1, lower.tail=F)

LCL1<-xbar-z*(0.6/sqrt(35))

UCL1<-xbar+z*(0.6/sqrt(35))

LCL1;UCL1
```

b1-ch6-3.R

```
library(foreign)

time<-read.dta(file = "http://kanggc.iptime.org/book/data/chap10-2.dta")

n<-length(time$var1)

s<-sd(time$var1)

t19<-qt(0.025, df = 19, lower.tail = F)

average<-mean(time$var1)

LCL<-average-t19*(s/sqrt(n))

UCL<-average + t19*(s/sqrt(n))

LCL;UCL
```

b1-ch6-4.R

```
set.seed(12345)

CIlower<-numeric(100)
CIupper<-numeric(100)
pvalue1<-numeric(100)

for(j in 1:100) {
  sample<-rnorm(100,10,2)
  s2<-var(sample)
  chi<-(99*s2)/4
  pvalue1[j]<-pchisq(chi, 0.95, df = 99, lower.tail = F)
  chi_u<-qchisq(0.975, df = 99)
  chi_l<-qchisq(0.025, df = 99)
  CIlower[j]<-(99*s2)/chi_u
  CIupper[j]<-(99*s2)/chi_l
#   pvalue1[j]<-pvalue1
```

```
}

CIlower
CIupper
pvalue1

reject1<-pvalue1<=0.05
table(reject1)

color<-rep(gray(.7),100)
color[reject1]<-"black"

plot(0, xlim=c(1,7), ylim=c(1,100), ylab="Sample No.", xlab="", main="95% Confidence Interval for sigma-square")
abline(v=4, lty=2)
for(j in 1:100) {
  lines(c(CIlower[j], CIupper[j]), c(j,j), col=color[j], lwd=1)
}
```

b1-ch6-5.R

```
time<-c(66,37,18,31,85,63,73,83,65,80

df<-length(time)-1

s.sq<-var(time)

u.chi<-qchisq(0.005, df=df, lower.tail=F)
u.chi
l.chi<-qchisq(0.995, df=df, lower.tail=F)
l.chi
LCL<-(df*s.sq/u.chi)

UCL<-(df*s.sq/l.chi)

LCL;UCL
```

제7장 가설검정

b1-ch7-1.R

```
set.seed(123456)

par(mfrow = c(1,3))

CIlower<-numeric(100)
CIupper<-numeric(100)
pvalue1<-numeric(100)
pvalue2<-numeric(100)

for(j in 1:1000) {
  sample<-rnorm(100,10,2)
  testres1<-t.test(sample,mu = 10)
  CIlower[j]<-testres1$conf.int[1]
  CIupper[j]<-testres1$conf.int[2]
  pvalue1[j]<-testres1$p.value
  pvalue2[j]<-t.test(sample, mu = 9.5)$p.value
}

reject2<-pvalue2<= 0.05
table(reject2)

color<-rep(gray(.7),100)
color[reject2[1:100]]<-"black"

plot(0, xlim = c(9,11), ylim = c(1,100), ylab = "Sample No.", xlab = "",
main = "Incorrrect H0")
abline(v = 9.5, lty = 2)
for(j in 1:100) {
  lines(c(CIlower[j], CIupper[j]), c(j,j), col = color[j], lwd = 1)
}
```

```
set.seed(12345)

CIlower<-numeric(100)
CIupper<-numeric(100)
pvalue1<-numeric(100)
pvalue2<-numeric(100)

for(j in 1:100) {
  sample<-rnorm(100,10,2)
  testres1<-t.test(sample,mu=10)
  CIlower[j]<-testres1$conf.int[1]
  CIupper[j]<-testres1$conf.int[2]
  pvalue1[j]<-testres1$p.value
  pvalue2[j]<-t.test(sample, mu=9.3)$p.value
}

reject2<-pvalue2<=0.05
table(reject2)

color<-rep(gray(.7),100)
color[reject2[1:100]]<-"black"

plot(0, xlim=c(9,11), ylim=c(1,100), ylab="Sample No.", xlab="",
main="Incorrrect H0")
abline(v=9.3, lty=2)
for(j in 1:100) {
  lines(c(CIlower[j], CIupper[j]), c(j,j), col=color[j], lwd=1)
}

set.seed(1234)

CIlower<-numeric(100)
CIupper<-numeric(100)
pvalue1<-numeric(100)
pvalue2<-numeric(100)
```

```
for(j in 1:100) {
  sample<-rnorm(100,10,2)
  testres1<-t.test(sample,mu=10)
  CIlower[j]<-testres1$conf.int[1]
  CIupper[j]<-testres1$conf.int[2]
  pvalue1[j]<-testres1$p.value
  pvalue2[j]<-t.test(sample, mu=9.2)$p.value
}

reject2<-pvalue2<=0.05
table(reject2)

color<-rep(gray(.7),100)
color[reject2[1:100]]<-"black"

plot(0, xlim=c(9,11), ylim=c(1,100), ylab="Sample No.", xlab="",
main="Incorrrect H0")
abline(v=9.2, lty=2)
for(j in 1:100) {
  lines(c(CIlower[j], CIupper[j]), c(j,j), col=color[j], lwd=1)
}
```

b1-ch7-2.R

```
library(foreign)

time<-read.dta(file="http://kanggc.iptime.org/book/data/chap11-2-1.dta")

time$var1

t.test(time$var1, mu=34.5, conf.level=0.9)

(t14<-qt(0.05, df=14, lower.tail=T))
(t14<-qt(0.05, df=14, lower.tail=F))
```

b1-ch7-3.R

```
x<-c(36,37,38,39,39,44,47)

xbar<-mean(x)
xbar

s.sq<-var(x)
s.sq

df = 6
q = length(x)-1

chi<-(df*s.sq)/62
chi

pchisq(chi, df = q,lower.tail = F)

UCV<-qchisq(0.025, df = q, lower.tail = F)
UCV

LCV<-qchisq(0.975, df = q, lower.tail = F)
LCV
```

b1-ch7-4.R

```
small<-c(41,45,42,62,68,54,52,55,44,60)
large<-c(74,74,70,52,76,91,71,78,76,78,83,50,52,66,65,53,72)

(mean(small))
(mean(large))

(var(small))
(var(large))

t.test(small, large, alternative = "less", var.equal = T, conf.level = 0.99)

(t25<-qt(0.01, df = 25, lower.tail = F))
```

b1-ch7-5.R

```
econ<-c(75,71,52,46,70,83)
mgt<-c(82,73,59,48,68,93)

t.test(econ, mgt, conf.level=0.95)

(mean(econ))
(mean(mgt))

(var(econ))
(var(mgt))

(t10<-qt(0.05, df=10, lower.tail=F))
```

b1-ch7-6.R

```
library(foreign)

data<-read.dta(file="http://kanggc.iptime.org/book/data/chap11-3-2.dta")

data$var1
data$var2

d<-data$var1-data$var2
d

(var(d))

t.test(d, mu=0, alternative="less", conf.level=0.95)

(t5<-qt(0.05, df=5, lower.tail=F))
```

b1-ch7-7.R

```
small<-c(41,45,42,62,68,54,52,55,44,60)
large<-c(74,74,70,52,76,91,71,78,76,78,83,50,52,66,65,53,72)

(mean(small))
(mean(large))

(var(small))
(var(large))

var.test(large, small, alternative = "greater", conf.level = 0.95)

(f<-qf(0.05, df1 = 16, df2 = 9, lower.tail = F))
```

부록 2

주요통계표

<표 1> 누적이항확률분포표

$$P(X \le a) = \sum_{x=0}^{a} P(x)$$

(a) n = 5

a	0.01	0.05	0.10	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	0.95	0.99
0	.951	.774	.590	.328	.168	.078	.031	.010	.002	.000	.000	.000	.000
1	.999	.977	.919	.737	.528	.337	.187	.087	.031	.007	.000	.000	.000
2	1.00	.999	.991	.942	.837	.683	.500	.317	.163	.058	.009	.001	.000
3	1.00	1.00	1.00	.993	.969	.914	.813	.663	.472	.263	.081	.023	.001
4	1.00	1.00	1.00	1.00	.998	.990	.969	.922	.832	.672	.410	.226	.049

(b) n = 10

a	0.01	0.05	0.10	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	0.95	0.99
0	.904	.599	.349	.107	.028	.006	.001	.000	.000	.000	.000	.000	.000
1	.996	.914	.736	.376	.149	.046	.011	.002	.000	.000	.000	.000	.000
2	1.00	.988	.930	.678	.383	.167	.055	.012	.002	.000	.000	.000	.000
3	1.00	.999	.987	.879	.650	.382	.172	.055	.011	.001	.000	.000	.000
4	1.00	1.00	.998	.967	.850	.633	.377	.166	.047	.006	.000	.000	.000
5	1.00	1.00	1.00	.994	.953	.834	.623	.367	.150	.033	.002	.000	.000
6	1.00	1.00	1.00	.999	.989	.945	.828	.618	.350	.121	.013	.001	.000
7	1.00	1.00	1.00	1.00	.998	.988	.945	.833	.617	.322	.070	.012	.000
8	1.00	1.00	1.00	1.00	1.00	.998	.989	.954	.851	.624	.264	.086	.004
9	1.00	1.00	1.00	1.00	1.00	1.00	.999	.994	.972	.893	.651	.401	.096

(c) n = 15

a	0.01	0.05	0.10	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	0.95	0.99
0	.860	.463	.206	.035	.005	.000	.000	.000	.000	.000	.000	.000	.000
1	.990	.829	.549	.167	.035	.005	.000	.000	.000	.000	.000	.000	.000
2	1.00	.964	.816	.398	.127	.027	.004	.000	.000	.000	.000	.000	.000
3	1.00	.995	.944	.648	.297	.091	.018	.002	.000	.000	.000	.000	.000
4	1.00	.999	.987	.836	.514	.217	.059	.009	.001	.000	.000	.000	.000
5	1.00	1.00	.998	.939	.722	.403	.151	.034	.004	.000	.000	.000	.000
6	1.00	1.00	1.00	.982	.869	.610	.304	.095	.015	.001	.000	.000	.000
7	1.00	1.00	1.00	.996	.950	.787	.500	.213	.050	.004	.000	.000	.000
8	1.00	1.00	1.00	.999	.985	.905	.696	.390	.131	.018	.000	.000	.000
9	1.00	1.00	1.00	1.00	.996	.966	.849	.597	.278	.061	.002	.002	.000
10	1.00	1.00	1.00	1.00	.999	.991	.941	.783	.485	.164	.013	.013	.000
11	1.00	1.00	1.00	1.00	1.00	.998	.982	.909	.703	.352	.056	.056	.000
12	1.00	1.00	1.00	1.00	1.00	1.00	.996	.973	.873	.602	.184	.184	.000
13	1.00	1.00	1.00	1.00	1.00	1.00	1.00	.995	.965	.833	.451	.451	.010
14	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	.995	.965	.794	.794	.140

(d) n = 20

a	0.01	0.05	0.10	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	0.95	0.99
0	.818	.358	.122	.012	.001	.000	.000	.000	.000	.000	.000	.000	.000
1	.983	.736	.392	.069	.008	.001	.000	.000	.000	.000	.000	.000	.000
2	.999	.925	.677	.206	.035	.004	.000	.000	.000	.000	.000	.000	.000
3	1.00	.984	.867	.411	.107	.016	.001	.000	.000	.000	.000	.000	.000
4	1.00	.997	.957	.630	.238	.051	.006	.000	.000	.000	.000	.000	.000
5	1.00	1.00	.989	.804	.416	.126	.021	.002	.000	.000	.000	.000	.000
6	1.00	1.00	.998	.913	.608	.250	.058	.006	.000	.000	.000	.000	.000
7	1.00	1.00	1.00	.968	.772	.416	.132	.021	.001	.000	.000	.000	.000
8	1.00	1.00	1.00	.990	.887	.596	.252	.057	.005	.000	.000	.000	.000
9	1.00	1.00	1.00	.997	.952	.755	.412	.128	.017	.001	.000	.000	.000
10	1.00	1.00	1.00	.999	.983	.872	.588	.245	.048	.003	.000	.000	.000
11	1.00	1.00	1.00	1.00	.995	.943	.748	.404	.113	.010	.000	.000	.000
12	1.00	1.00	1.00	1.00	.999	.979	.868	.584	.228	.032	.000	.000	.000
13	1.00	1.00	1.00	1.00	1.00	.994	.942	.750	.392	.087	.002	.000	.000
14	1.00	1.00	1.00	1.00	1.00	.998	.979	.874	.584	.196	.011	.000	.000
15	1.00	1.00	1.00	1.00	1.00	1.00	.994	.949	.762	.370	.043	.003	.000
16	1.00	1.00	1.00	1.00	1.00	1.00	.999	.984	.893	.589	.133	.016	.000
17	1.00	1.00	1.00	1.00	1.00	1.00	1.00	.996	.965	.794	.323	.075	.001
18	1.00	1.00	1.00	1.00	1.00	1.00	1.00	.999	.992	.931	.608	.264	.017
19	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	.999	.988	.878	.642	.182

(e) n = 25

a	0.01	0.05	0.10	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	0.95	0.99
0	.778	.277	.072	.004	.000	.000	.000	.000	.000	.000	.000	.000	.000
1	.974	.642	.271	.027	.002	.000	.000	.000	.000	.000	.000	.000	.000
2	.998	.873	.537	.098	.009	.000	.000	.000	.000	.000	.000	.000	.000
3	1.00	.966	.764	.234	.033	.002	.000	.000	.000	.000	.000	.000	.000
4	1.00	.993	.902	.421	.090	.009	.000	.000	.000	.000	.000	.000	.000
5	1.00	.999	.967	.617	.193	.029	.002	.000	.000	.000	.000	.000	.000
6	1.00	1.00	.991	.780	.341	.074	.007	.000	.000	.000	.000	.000	.000
7	1.00	1.00	.998	.891	.512	.154	.022	.001	.000	.000	.000	.000	.000
8	1.00	1.00	1.00	.953	.677	.274	.054	.004	.000	.000	.000	.000	.000
9	1.00	1.00	1.00	.983	.811	.425	.115	.013	.000	.000	.000	.000	.000
10	1.00	1.00	1.00	.994	.902	.586	.212	.034	.002	.000	.000	.000	.000
11	1.00	1.00	1.00	.998	.956	.732	.345	.078	.006	.000	.000	.000	.000
12	1.00	1.00	1.00	1.00	.983	.846	.500	.154	.017	.000	.000	.000	.000
13	1.00	1.00	1.00	1.00	.994	.922	.655	.268	.044	.002	.000	.000	.000
14	1.00	1.00	1.00	1.00	.998	.966	.788	.414	.098	.006	.000	.000	.000
15	1.00	1.00	1.00	1.00	1.00	.987	.885	.575	.189	.017	.000	.000	.000
16	1.00	1.00	1.00	1.00	1.00	.996	.946	.726	.323	.047	.000	.000	.000
17	1.00	1.00	1.00	1.00	1.00	.999	.978	.846	.488	.109	.002	.000	.000
18	1.00	1.00	1.00	1.00	1.00	1.00	.993	.926	.659	.220	.009	.000	.000
19	1.00	1.00	1.00	1.00	1.00	1.00	.998	.971	.807	.383	.033	.001	.000
20	1.00	1.00	1.00	1.00	1.00	1.00	1.00	.991	.910	.579	.098	.007	.000
21	1.00	1.00	1.00	1.00	1.00	1.00	1.00	.998	.967	.766	.236	.034	.000
22	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	.991	.902	.463	.127	.002
23	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	.998	.973	.729	.358	.026
24	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	.996	.928	.723	.222

<표 2> 누적포아송확률분포표

$$P(X \le c) = \sum_{k=0}^{c} \frac{\mu^k e^{-\mu}}{k!}, \mu : 기댓값$$

μ \ c	0.02	0.04	0.06	0.08	0.10	0.15	0.20	0.25	0.30	0.35	0.40
0	0.980	0.961	0.942	0.923	0.905	0.861	0.819	0.779	0.741	0.705	0.670
1	1.000	0.999	0.998	0.997	0.995	0.990	0.982	0.974	0.963	0.951	0.938
2	1.000	1.000	1.000	1.000	1.000	0.999	0.999	0.998	0.996	0.994	0.992
3	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.999
4	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

μ \ c	0.45	0.50	0.55	0.60	0.65	0.70	0.75	0.80	0.85	0.90	0.95
0	0.638	0.607	0.577	0.549	0.522	0.497	0.472	0.449	0.427	0.407	0.387
1	0.925	0.910	0.894	0.878	0.861	0.844	0.827	0.809	0.791	0.772	0.754
2	0.989	0.986	0.982	0.977	0.972	0.966	0.959	0.953	0.945	0.937	0.929
3	0.999	0.998	0.998	0.997	0.996	0.994	0.993	0.991	0.989	0.987	0.984
4	1.000	1.000	1.000	1.000	0.999	0.999	0.999	0.999	0.998	0.998	0.997
5	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

μ \ c	1.0	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2.0
0	0.368	0.333	0.301	0.273	0.247	0.223	0.202	0.183	0.165	0.150	0.135
1	0.736	0.699	0.663	0.627	0.592	0.558	0.525	0.493	0.463	0.434	0.406
2	0.920	0.900	0.879	0.857	0.833	0.809	0.783	0.757	0.731	0.704	0.677
3	0.981	0.974	0.966	0.957	0.946	0.934	0.921	0.907	0.891	0.875	0.857
4	0.996	0.995	0.992	0.989	0.986	0.981	0.976	0.970	0.964	0.956	0.947
5	0.999	0.999	0.998	0.998	0.997	0.996	0.994	0.992	0.990	0.987	0.983
6	1.000	1.000	1.000	1.000	0.999	0.999	0.999	0.998	0.997	0.997	0.995
7	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.999	0.999	0.999
8	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

μ \ c	2.2	2.4	2.6	2.8	3.0	3.2	3.4	3.6	3.8	4.0	4.2
0	0.111	0.091	0.074	0.061	0.050	0.041	0.033	0.027	0.022	0.050	0.015
1	0.355	0.308	0.267	0.231	0.199	0.171	0.147	0.126	0.107	0.092	0.078
2	0.623	0.570	0.518	0.469	0.423	0.380	0.340	0.303	0.269	0.238	0.210
3	0.819	0.779	0.736	0.692	0.647	0.603	0.558	0.515	0.473	0.433	0.395
4	0.928	0.904	0.877	0.848	0.815	0.781	0.744	0.706	0.668	0.629	0.590
5	0.975	0.964	0.951	0.935	0.916	0.895	0.871	0.844	0.816	0.785	0.753
6	0.993	0.988	0.983	0.976	0.966	0.955	0.942	0.927	0.909	0.889	0.867
7	0.998	0.997	0.995	0.992	0.988	0.983	0.977	0.969	0.960	0.949	0.936
8	1.000	0.999	0.999	0.998	0.996	0.994	0.992	0.988	0.984	0.979	0.972
9	1.000	1.000	1.000	0.999	0.999	0.998	0.997	0.996	0.994	0.992	0.989
10	1.000	1.000	1.000	1.000	1.000	1.000	0.999	0.999	0.998	0.997	0.996
11	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.999	0.999	0.999
12	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

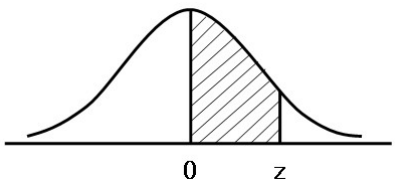
μ \ c	4.4	4.6	4.8	5.0	5.2	5.4	5.6	5.8	6.0	6.2	6.4
0	0.012	0.010	0.009	0.007	0.006	0.005	0.004	0.003	0.002	0.002	0.002
1	0.066	0.056	0.048	0.040	0.034	0.029	0.024	0.021	0.017	0.015	0.012
2	0.185	0.163	0.143	0.125	0.109	0.095	0.082	0.072	0.062	0.054	0.046
3	0.359	0.326	0.294	0.265	0.238	0.213	0.191	0.170	0.151	0.134	0.119
4	0.551	0.513	0.476	0.440	0.406	0.373	0.342	0.313	0.285	0.259	0.235
5	0.720	0.686	0.651	0.616	0.581	0.546	0.512	0.478	0.446	0.414	0.384
6	0.844	0.818	0.791	0.762	0.732	0.702	0.670	0.638	0.606	0.574	0.542
7	0.921	0.905	0.887	0.867	0.845	0.822	0.797	0.771	0.744	0.716	0.687
8	0.964	0.955	0.944	0.932	0.918	0.903	0.886	0.867	0.847	0.826	0.803
9	0.985	0.980	0.975	0.968	0.960	0.951	0.941	0.929	0.916	0.902	0.886
10	0.994	0.992	0.990	0.986	0.982	0.977	0.972	0.965	0.957	0.949	0.939
11	0.998	0.997	0.996	0.995	0.993	0.990	0.988	0.984	0.980	0.975	0.969
12	0.999	0.999	0.999	0.998	0.997	0.996	0.995	0.993	0.991	0.989	0.986
13	1.000	1.000	1.000	0.999	0.999	0.999	0.998	0.997	0.996	0.995	0.994
14	1.000	1.000	1.000	1.000	1.000	1.000	0.999	0.999	0.999	0.998	0.997
15	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.999	0.999	0.999
16	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

μ \ c	6.6	6.8	7.0	7.2	7.4	7.6	7.8	8.0	8.5	9.0	9.5
0	0.001	0.001	0.001	0.001	0.001	0.001	0.000	0.000	0.000	0.000	0.000
1	0.010	0.009	0.007	0.006	0.005	0.004	0.004	0.003	0.002	0.001	0.001
2	0.040	0.034	0.030	0.025	0.022	0.019	0.016	0.014	0.009	0.006	0.004
3	0.105	0.093	0.082	0.072	0.063	0.055	0.048	0.042	0.030	0.021	0.015
4	0.213	0.192	0.173	0.156	0.140	0.125	0.112	0.100	0.074	0.055	0.040
5	0.355	0.327	0.301	0.276	0.253	0.231	0.210	0.191	0.150	0.116	0.089
6	0.511	0.480	0.450	0.420	0.392	0.365	0.338	0.313	0.256	0.207	0.165
7	0.658	0.628	0.599	0.569	0.539	0.510	0.481	0.453	0.386	0.324	0.269
8	0.780	0.755	0.729	0.703	0.676	0.648	0.620	0.593	0.523	0.456	0.392
9	0.869	0.850	0.830	0.810	0.788	0.765	0.741	0.717	0.653	0.587	0.522
10	0.927	0.915	0.901	0.887	0.871	0.854	0.835	0.816	0.763	0.706	0.645
11	0.963	0.955	0.947	0.937	0.926	0.915	0.902	0.888	0.849	0.803	0.752
12	0.982	0.978	0.973	0.967	0.961	0.954	0.945	0.936	0.909	0.876	0.836
13	0.992	0.990	0.987	0.984	0.980	0.976	0.971	0.966	0.949	0.926	0.898
14	0.997	0.996	0.994	0.993	0.991	0.989	0.986	0.983	0.973	0.959	0.940
15	0.999	0.998	0.998	0.997	0.996	0.995	0.993	0.992	0.986	0.978	0.967
16	0.999	0.999	0.999	0.999	0.998	0.998	0.997	0.996	0.993	0.989	0.982
17	1.000	1.000	1.000	1.000	0.999	0.999	0.999	0.998	0.997	0.995	0.991
18	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.999	0.999	0.998	0.996
19	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.999	0.999	0.998
20	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.999
21	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

μ \ c	10.0	10.5	11.0	11.5	12.0	12.5	13.0	13.5	14.0	14.5	15.0
0	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
2	0.003	0.002	0.001	0.001	0.001	0.000	0.000	0.000	0.000	0.000	0.000
3	0.010	0.007	0.005	0.003	0.002	0.002	0.001	0.001	0.000	0.000	0.000
4	0.029	0.021	0.015	0.011	0.008	0.005	0.004	0.003	0.002	0.001	0.001
5	0.067	0.050	0.038	0.028	0.020	0.015	0.011	0.008	0.006	0.004	0.003
6	0.130	0.102	0.079	0.060	0.046	0.035	0.026	0.019	0.014	0.010	0.008
7	0.220	0.179	0.143	0.114	0.090	0.070	0.054	0.041	0.032	0.024	0.018
8	0.333	0.279	0.232	0.191	0.155	0.125	0.100	0.079	0.062	0.048	0.037
9	0.458	0.397	0.341	0.289	0.242	0.201	0.166	0.135	0.109	0.088	0.070
10	0.583	0.521	0.460	0.402	0.347	0.297	0.252	0.211	0.176	0.145	0.118
11	0.697	0.639	0.579	0.520	0.462	0.406	0.353	0.304	0.260	0.220	0.185
12	0.792	0.742	0.689	0.633	0.576	0.519	0.463	0.409	0.358	0.311	0.268
13	0.864	0.825	0.781	0.733	0.682	0.628	0.573	0.518	0.464	0.413	0.363
14	0.917	0.888	0.854	0.815	0.772	0.725	0.675	0.623	0.570	0.518	0.466
15	0.951	0.932	0.907	0.878	0.844	0.806	0.764	0.718	0.669	0.619	0.568
16	0.973	0.960	0.944	0.924	0.899	0.869	0.835	0.798	0.756	0.711	0.664
17	0.986	0.978	0.968	0.954	0.937	0.916	0.890	0.861	0.827	0.790	0.749
18	0.993	0.988	0.982	0.974	0.963	0.948	0.930	0.908	0.883	0.853	0.819
19	0.997	0.994	0.991	0.986	0.979	0.969	0.957	0.942	0.923	0.901	0.875
20	0.998	0.997	0.995	0.992	0.988	0.983	0.975	0.965	0.952	0.936	0.917
21	0.999	0.999	0.998	0.996	0.994	0.991	0.986	0.980	0.971	0.960	0.947
22	1.000	0.999	0.999	0.998	0.997	0.995	0.992	0.989	0.983	0.976	0.967
23	1.000	1.000	1.000	0.999	0.999	0.998	0.996	0.994	0.991	0.986	0.981
24	1.000	1.000	1.000	1.000	0.999	0.999	0.998	0.997	0.995	0.992	0.989
25	1.000	1.000	1.000	1.000	1.000	0.999	0.999	0.998	0.997	0.996	0.994
26	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.999	0.999	0.998	0.997
27	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.999	0.999	0.998
28	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.999	0.999
29	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000

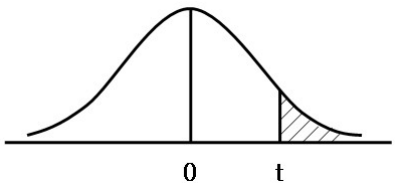
〈표 3〉 표준정규분포표

$$P(0 \leqq Z \leqq z) = \int_0^z \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2} dx$$



z	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09
0.0	0.0000	0.0040	0.0080	0.0120	0.0160	0.0199	0.0239	0.0279	0.0319	0.0359
0.1	0.0398	0.0438	0.0478	0.0517	0.0557	0.0596	0.0636	0.0675	0.0714	0.0753
0.2	0.0793	0.0832	0.0871	0.0910	0.0948	0.0987	0.1026	0.1064	0.1103	0.1141
0.3	0.1179	0.1217	0.1255	0.1293	0.1331	0.1368	0.1406	0.1443	0.1480	0.1517
0.4	0.1554	0.1591	0.1628	0.1664	0.1700	0.1736	0.1772	0.1808	0.1844	0.1879
0.5	0.1915	0.1950	0.1985	0.2019	0.2054	0.2088	0.2123	0.2157	0.2190	0.2224
0.6	0.2257	0.2291	0.2324	0.2357	0.2389	0.2422	0.2454	0.2486	0.2518	0.2549
0.7	0.2580	0.2612	0.2642	0.2673	0.2704	0.2734	0.2764	0.2794	0.2823	0.2852
0.8	0.2881	0.2910	0.2939	0.2967	0.2995	0.3023	0.3051	0.3078	0.3106	0.3133
0.9	0.3159	0.3186	0.3212	0.3238	0.3264	0.3289	0.3315	0.3340	0.3365	0.3389
1.0	0.3413	0.3438	0.3461	0.3435	0.3508	0.3531	0.3554	0.3577	0.3599	0.3621
1.1	0.3643	0.3665	0.3686	0.3708	0.3729	0.3749	0.3770	0.3790	0.3810	0.3830
1.2	0.3849	0.3869	0.3888	0.3907	0.3925	0.3944	0.3962	0.3980	0.3997	0.4015
1.3	0.4032	0.4049	0.4066	0.4082	0.4099	0.4115	0.4131	0.4147	0.4162	0.4177
1.4	0.4192	0.4207	0.4222	0.4236	0.4251	0.4265	0.4279	0.4292	0.4306	0.4319
1.5	0.4332	0.4345	0.4357	0.4370	0.4382	0.4394	0.4406	0.4418	0.4429	0.4441
1.6	0.4452	0.4463	0.4474	0.4484	0.4495	0.4505	0.4515	0.4525	0.4535	0.4545
1.7	0.4554	0.4564	0.4573	0.4582	0.4591	0.4599	0.4608	0.4616	0.4625	0.4633
1.8	0.4641	0.4649	0.4656	0.4664	0.4671	0.4678	0.4686	0.4693	0.4699	0.4706
1.9	0.4713	0.4719	0.4726	0.4732	0.4738	0.4744	0.4750	0.4756	0.4761	0.4767
2.0	0.4772	0.4778	0.4783	0.4788	0.4793	0.4798	0.4803	0.4808	0.4812	0.4817
2.1	0.4821	0.4826	0.4830	0.4834	0.4838	0.4842	0.4846	0.4850	0.4854	0.4857
2.2	0.4861	0.4864	0.4868	0.4871	0.4875	0.4878	0.4881	0.4884	0.4887	0.4890
2.3	0.4893	0.4896	0.4898	0.4901	0.4904	0.4906	0.4909	0.4911	0.4913	0.4916
2.4	0.4918	0.4920	0.4922	0.4925	0.4927	0.4929	0.4931	0.4932	0.4934	0.4936
2.5	0.4938	0.4940	0.4941	0.4943	0.4945	0.4946	0.4948	0.4949	0.4951	0.4952
2.6	0.4953	0.4955	0.4956	0.4957	0.4959	0.4960	0.4961	0.4962	0.4963	0.4964
2.7	0.4965	0.4966	0.4967	0.4968	0.4969	0.4970	0.4971	0.4972	0.4973	0.4974
2.8	0.4974	0.4975	0.4976	0.4977	0.4977	0.4978	0.4979	0.4979	0.4980	0.4981
2.9	0.4981	0.4982	0.4982	0.4983	0.4984	0.4985	0.4985	0.4985	0.4986	0.4986
3.0	0.4986	0.4987	0.4987	0.4988	0.4988	0.4984	0.4989	0.4989	0.4990	0.4990

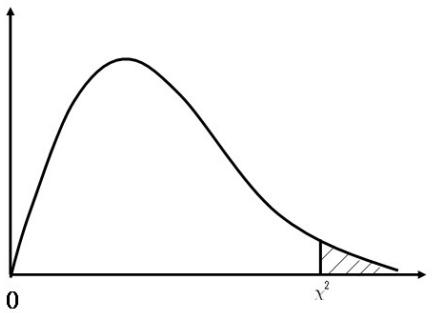
〈표 4〉 t-분포표



v / p	0.1	0.05	0.025	0.01	0.005
1	3.078	6.314	12.706	31.821	63.657
2	1.886	2.920	4.303	6.965	9.923
3	0.1638	2.353	3.182	4.541	5.841
4	1.533	2.132	2.776	3.747	4.604
5	1.476	2.015	2.571	3.365	4.032
6	1.440	1.943	2.447	3.143	3.707
7	1.415	1.895	2.365	2.998	3.499
8	1.397	1.860	2.306	2.896	3.355
9	1.383	1.833	2.262	2.821	3.250
10	1.372	1.812	2.228	2.764	3.169
11	1.363	1.796	2.201	2.718	3.103
12	1.356	1.782	2.179	2.681	3.055
13	1.350	1.771	2.160	2.650	3.012
14	1.345	1.761	2.145	2.624	2.977
15	1.341	1.753	2.131	2.602	2.947
16	1.337	1.746	2.120	2.583	2.921
17	1.333	1.740	2.110	2.567	2.898
18	1.330	1.734	2.101	2.552	2.878
19	1.328	1.729	2.093	2.539	2.816
20	1.325	1.725	2.086	2.528	2.845
21	1.323	1.721	2.080	2.518	2.831
22	1.321	1.717	2.074	2.508	2.819
23	1.319	1.714	2.069	2.500	2.807
24	1.318	1.711	2.064	2.492	2.797
25	1.316	1.708	2.060	2.485	2.787
26	1.315	1.706	2.056	2.479	2.779
27	1.314	1.703	2.052	2.473	2.771
28	1.313	1.701	2.048	2.467	2.763
29	1.311	1.699	2.045	2.462	2.756
30	1.310	1.697	2.042	2.457	2.750
40	1.303	1.684	2.021	2.423	2.704
60	1.296	1.671	2.000	2.390	2.660
120	1.289	1.658	1.980	2.358	2.617
∞	1.282	1.645	1.960	2.326	2.576

〈표 5〉 χ^2-분포표

$P(X \geqq \chi_\alpha^2) = \alpha$



자유도	P=0.99	0.98	0.95	0.90	0.80	0.20	0.10	0.05	0.02	0.01
1	0.000157	0.000628	0.00393	0.0158	0.0642	1.642	2.706	3.841	5.412	6.635
2	0.0201	0.0404	0.103	0.211	0.446	3.219	4.605	5.991	7.824	9.210
3	0.115	0.185	0.352	0.584	1.005	4.642	6.251	7.815	9.837	11.341
4	0.297	0.429	0.711	1.064	1.649	5.989	7.779	9.488	11.668	13.277
5	0.554	0.752	1.145	1.610	2.343	7.289	9.236	11.070	13.388	15.086
6	0.872	1.134	1.635	2.204	3.070	8.558	10.645	12.592	15.033	16.812
7	1.239	1.564	2.167	2.833	3.822	9.803	12.017	14.067	16.622	18.475
8	1.646	2.032	2.733	3.490	4.594	11.030	13.362	15.507	18.168	20.090
9	2.088	2.532	3.325	4.168	5.380	12.242	14.684	16.919	19.679	21.666
10	2.558	3.059	3.940	4.865	6.179	13.442	15.987	18.307	21.161	23.209
11	3.053	3.609	4.575	5.578	5.989	14.631	17.275	19.675	22.618	24.725
12	3.571	4.178	5.226	6.034	7.807	15.812	18.549	21.026	24.054	26.217
13	4.017	4.765	5.892	7.042	8.634	16.985	19.812	22.362	25.472	27.688
14	4.660	5.368	6.517	7.790	9.467	18.151	21.064	23.685	26.873	29.141
15	5.229	5.985	7.261	8.547	10.307	19.311	22.307	24.996	28.259	30.578
16	5.812	6.614	7.962	9.312	11.152	20.465	23.542	26.296	29.633	32.000
17	6.408	7.255	8.672	10.085	12.002	21.615	24.769	27.587	30.995	33.409
18	7.015	7.906	9.390	10.865	12.857	22.760	25.989	28.869	32.346	34.805
19	7.633	8.567	10.117	11.651	13.716	23.900	27.204	30.144	33.687	36.191
20	8.260	9.237	10.851	12.443	14.578	25.038	28.412	31.410	35.020	37.566
21	8.897	9.915	11.591	13.240	15.445	26.171	29.615	32.671	36.343	38.932
22	9.542	10.600	12.338	14.041	16.314	27.301	30.813	33.924	37.659	40.289
23	10.196	11.293	13.091	14.848	17.187	26.429	32.007	35.172	38.968	41.638
24	10.856	11.992	13.848	15.659	18.062	29.553	33.196	36.415	40.270	42.980
25	11.524	12.697	14.611	16.473	18.940	30.675	34.382	37.652	41.566	44.314
26	12.198	13.409	15.379	17.292	19.820	31.795	36.563	38.885	42.856	45.642
27	12.879	14.125	16.151	18.114	20.703	32.912	36.741	40.113	44.141	46.963
28	13.565	14.847	16.928	18.939	21.588	34.027	37.916	41.337	45.419	48.278
29	14.256	15.574	17.708	19.768	22.475	36.139	39.087	42.557	46.693	49.588
30	14.953	16.306	18.493	30.599	23.364	36.250	40.256	43.773	47.962	50.892

<표 6> F-분포표(5% 유의수준)

v1 \ v2	1	2	3	4	5	6	7	8	9	10	12	15	20	24	30	40	∞
1	161.45	199.50	215.71	224.58	230.16	233.99	236.77	238.88	240.54	241.88	243.91	245.95	248.01	249.05	250.09	241.14	254.31
2	18.51	19.00	19.16	19.25	19.30	19.33	19.35	19.37	19.38	19.40	19.41	19.43	19.45	19.45	19.46	19.47	19.50
3	10.13	9.55	9.28	9.12	9.01	8.94	8.89	8.85	8.81	8.76	8.74	8.70	8.66	8.64	8.62	8.59	8.53
4	7.71	6.94	6.59	6.39	6.26	6.16	6.09	6.04	6.00	5.96	5.91	5.86	5.80	5.77	5.75	5.72	5.63
5	6.61	5.79	5.41	5.19	5.05	4.95	4.48	4.82	4.77	4.74	4.68	4.62	4.56	4.53	4.50	4.46	4.37
6	5.99	4.74	7.35	4.12	3.94	3.87	3.79	3.73	3.68	3.64	3.57	3.51	3.44	3.41	3.38	3.34	3.23
7	5.59	4.74	4.35	4.12	3.97	3.87	3.79	3.73	3.68	3.64	3.57	3.51	3.44	3.41	3.38	3.34	3.23
8	5.32	4346	4.07	3.84	3.69	3.58	3.50	3.44	3.39	3.35	3.28	3.22	3.15	3.12	3.08	3.04	2.93
9	5.12	4.26	3.86	3.63	3.48	3.37	3.29	3.23	3.18	3.14	3.07	3.01	2.94	2.90	2.84	2.83	2.71
10	4.96	4.10	3.71	3.48	3.33	3.22	3.14	3.07	3.02	2.98	2.91	2.85	2.77	2.74	2.70	2.66	2.54
11	4.84	3.98	3.59	3.36	3.20	3.09	3.01	2.95	2.90	2.85	2.79	2.72	2.65	2.61	2.57	2.53	2.40
12	4.75	3.89	3.49	3.26	3.11	3.00	2.91	2.85	2.80	2.75	2.69	2.62	2.54	2.51	2.47	2.43	2.30
13	4.67	3.81	3.41	3.18	3.03	2.92	2.83	2.77	2.71	2.67	2.60	2.53	2.46	2.42	2.38	2.34	2.21
14	4.60	3.74	3.34	3.11	2.96	2.85	2.76	2.80	2.65	2.60	2.53	2.47	2.39	2.35	2.31	2.27	2.13
15	4.54	3.68	3.29	3.06	2.90	2.79	2.71	2.64	2.59	2.54	2.48	2.40	2.33	2.29	2.25	2.20	2.07
16	4.49	3.63	3.24	3.01	2.85	2.74	2.66	2.59	2.54	2.49	2.42	2.35	2.28	2.24	2.19	2.15	2.01
17	4.45	3.59	3.20	2.96	2.81	2.70	2.61	2.55	2.49	2.45	2.38	2.31	2.23	2.19	2.15	2.10	1.96
18	4.41	3.55	3.16	2.93	2.77	2.66	2.58	2.51	2.46	2.41	2.34	2.27	2.19	2.15	2.11	2.06	1.92
19	4.38	3.52	3.13	2.90	2.74	2.63	2.54	2.48	2.42	2.38	2.31	2.23	2.16	2.11	2.07	2.03	1.88
20	4.35	3.49	3.10	2.87	2.71	2.60	2.51	2.45	2.39	2.35	2.28	2.20	2.12	2.08	2.04	1.99	1.84
21	4.32	3.47	3.07	2.84	2.68	2.57	2.49	2.42	2.37	2.32	2.25	2.18	2.10	2.05	2.01	1.96	1.81
22	4.30	3.44	3.05	2.82	2.66	2.55	2.46	2.40	2.34	2.30	2.23	2.15	2.07	2.03	1.98	1.94	1.78
23	4.28	3.42	3.03	2.80	2.64	2.53	2.44	2.37	2.32	2.27	2.20	2.13	2.05	2.01	1.96	1.91	1.76
24	4.26	3.40	3.01	2.78	2.62	2.51	2.42	2.36	2.30	2.25	2.18	2.11	2.03	1.98	1.94	1.89	1.73
25	4.24	3.39	2.99	2.76	2.60	2.49	2.40	2.34	2.28	2.24	2.16	2.09	2.01	1.96	1.92	1.87	1.71
26	4.23	3.37	2.98	2.74	2.59	2.47	2.39	2.32	2.27	2.22	2.15	2.07	1.99	1.95	1.90	1.85	1.69
27	4.21	3.35	2.96	2.73	2.57	2.46	2.37	2.31	2.25	2.20	2.13	2.06	1.97	1.93	1.88	1.84	1.67
28	4.20	3.34	2.95	2.71	2.56	2.45	2.36	2.29	2.24	2.19	2.12	2.04	1.96	1.91	1.87	1.82	1.65
29	4.18	3.33	2.93	2.70	2.55	2.43	2.35	2.28	2.22	2.18	2.10	2.03	1.94	1.90	1.85	1.81	1.64
30	4.17	3.32	2.92	2.69	2.53	2.42	2.33	2.27	2.21	2.16	2.09	2.01	1.93	1.89	1.84	1.79	1.62
40	4.08	3.23	2.84	2.61	2.45	2.34	2.25	2.18	2.12	2.08	2.00	1.92	1.84	1.79	1.74	1.69	1.51
60	4.00	3.15	2.76	2.53	2.37	2.25	2.17	2.10	2.04	1.99	1.92	1.84	1.75	1.70	1.65	1.59	1.39
120	3.92	3.07	2.68	2.45	2.29	2.18	2.09	2.02	1.95	1.91	1.83	1.75	1.66	1.61	1.55	1.50	1.25
∞	3.84	3.00	2.60	2.37	2.21	2.10	2.01	1.94	1.88	1.83	1.75	1.67	1.57	1.52	1.46	1.39	1.00

v_1: 분자의 자유도 v_2: 분모의 자유도.

<표 6(계속)> F-분포표(1% 유의수준)

v1 \ v2	1	2	3	4	5	6	7	8	9	10	12	15	20	24	30	40	∞
1	4052.18	4999.50	5403.35	5624.58	5763.65	5858.99	5928.36	5981.07	6022.47	6055.87	6106.31	6157.28	6208.73	6234.63	6260.65	6286.78	6365.86
2	98.50	99.00	99.17	99.25	99.30	99.33	99.36	99.37	99.39	99.40	99.42	99.43	99.45	99.46	99.47	99.47	99.50
3	34.12	30.82	29.46	82.71	28.24	27.91	27.67	27.49	27.35	27.23	27.05	26.87	26.69	26.60	26.50	26.41	26.13
4	21.20	18.00	16.69	15.98	15.52	15.21	14.98	14.80	14.66	14.55	14.37	14.20	14.02	13.93	13.84	13.75	13.46
5	16.26	13.27	12.06	11.39	10.97	10.67	10.46	10.29	10.16	10.05	9.89	9.72	9.55	9.47	9.38	9.29	9.02
6	13.75	10.92	9.78	9.15	8.75	8.47	8.26	8.10	7.98	7.87	7.72	7.56	7.40	7.31	7.23	7.14	6.88
7	12.25	9.55	8.45	7.85	7.46	7.19	6.99	6.84	6.72	6.62	6.47	6.31	6.16	6.07	5.99	5.91	5.65
8	11.26	8.65	7.59	7.01	6.63	6.37	6.18	6.03	5.91	5.81	5.67	5.52	5.36	5.28	5.20	5.12	4.86
9	10.56	8.02	6.99	6.42	6.06	5.80	5.61	5.47	5.35	5.26	5.11	4.96	4.81	4.73	4.65	4.57	4.31
10	10.04	7.56	6.55	5.99	5.64	5.39	5.20	5.06	4.94	4.85	4.71	4.56	4.41	4.33	4.25	4.17	3.91
11	9.65	7.21	6.22	5.67	5.32	5.07	4.89	4.74	4.63	4.54	4.40	4.25	4.10	4.02	3.94	3.86	3.60
12	9.33	6.93	5.95	5.41	5.06	4.82	4.64	4.50	4.39	4.30	4.16	4.01	3.86	3.78	3.70	3.62	3.36
13	9.07	6.70	5.74	5.21	4.86	4.62	4.44	4.30	4.19	4.10	3.96	3.82	3.66	3.59	3.51	3.43	3.17
14	8.86	6.51	5.56	5.04	4.70	4.46	4.28	4.14	4.03	3.94	3.80	3.66	3.51	3.43	3.35	3.27	3.00
15	8.68	6.36	5.42	4.89	4.56	4.32	4.14	4.00	3.89	3.80	3.67	3.52	3.37	3.29	3.21	3.13	2.87
16	8.53	6.23	5.29	4.77	4.44	4.20	4.03	3.89	3.78	3.69	3.55	3.41	3.26	3.18	3.10	3.02	2.75
17	8.40	6.11	5.19	4.67	4.34	4.10	3.93	3.79	3.68	3.59	3.46	3.31	3.16	3.08	3.00	2.92	2.65
18	8.29	6.01	5.09	4.58	4.25	4.01	3.84	3.71	360	3.51	3.37	3.23	3.08	3.00	2.92	2.84	2.57
19	8.18	5.93	5.01	4.50	4.17	3.94	3.77	3.63	3.52	3.43	3.30	3.15	3.00	2.92	2.84	2.76	2.49
20	8.10	5.85	4.94	4.43	4.10	3.87	3.70	3.56	3.46	3.37	3.23	3.09	2.94	2.86	2.78	2.69	2.42
21	8.02	5.78	4.87	4.37	4.04	3.81	3.64	3.51	3.40	3.31	3.17	3.03	2.88	2.80	2.72	2.64	2.36
22	7.95	5.72	4.82	4.31	3.99	3.76	3.59	3.45	3.35	3.26	3.12	2.98	2.83	2.75	2.67	2.58	2.31
23	7.88	5.66	4.76	4.26	3.94	3.71	3.54	3.41	3.30	3.21	3.07	2.93	2.78	2.70	2.62	2.54	2.26
24	7.82	5.61	4.72	4.22	3.90	3.67	3.50	3.36	3.26	3.17	3.03	2.89	2.74	2.66	2.58	2.49	2.21
25	7.77	5.57	4.68	4.18	3.86	3.63	3.46	3.32	3.22	3.13	2.99	2.85	2.70	2.62	2.54	2.45	2.17
26	7.72	5.53	4.64	4.14	3.82	3.59	3.42	3.29	3.18	3.09	2.96	2.81	2.66	2.58	2.50	2.42	2.13
27	7.68	5.49	4.60	4.11	3.78	3.56	3.39	3.26	3.15	3.06	2.93	2.78	2.63	2.55	2.47	2.38	2.10
28	7.64	5.45	4.57	4.07	3.75	3.53	3.36	3.23	3.12	3.03	2.90	2.75	2.60	2.52	2.44	2.35	2.06
29	7.60	5.42	4.54	4.04	3.73	3.50	3.33	3.20	3.09	3.00	2.87	2.73	2.57	2.49	2.41	2.33	2.03
30	7.56	5.39	4.51	4.02	3.70	3.47	3.30	3.17	3.07	2.98	2.84	2.70	2.55	2.47	2.39	2.30	2.01
40	7.31	5.18	4.31	3.83	3.51	3.20	3.12	2.99	2.89	2.80	2.66	2.52	2.37	2.29	2.20	2.11	1.80
60	7.08	4.98	4.13	3.65	3.34	3.12	2.95	2.82	2.72	2.63	2.50	2.35	2.20	2.12	2.03	1.94	1.60
120	6.85	4.79	3.95	3.48	3.17	2.96	2.79	2.66	2.56	2.47	2.34	2.19	2.03	1.95	1.86	1.76	1.38
∞	6.63	4.61	3.78	3.32	3.02	2.80	2.64	2.51	2.41	2.32	2.18	2.04	1.88	1.79	1.70	1.59	1.00

v_1: 분자의 자유도 v_2: 분모의 자유도.

<표 7> Durbin-Watson (5% 유의수준)

n	k'=1		k'=2		k'=3		k'=4		k'=5	
	dL	dU	dL	dU	dL	dU	dL	dU	dL	dU
15	1.077	1.361	0.946	1.543	0.814	1.750	0.685	1.977	0.562	2.220
16	1.106	1.371	0.982	1.539	0.857	1.728	0.734	1.935	0.615	2.157
17	1.133	1.381	1.015	1.536	0.897	1.710	0.779	1.900	0.664	2.104
18	1.158	1.391	1.046	1.535	0.933	1.696	0.820	1.872	0.710	2.060
19	1.180	1.410	1.074	1.536	0.967	1.685	0.859	1.848	0.752	2.023
20	1.201	1.411	1.100	1.537	0.998	1.676	0.894	1.828	0.792	1.991
21	1.221	1.420	1.125	1.538	1.026	1.669	0.927	1.812	0.829	1.964
22	1.239	1.429	1.147	1.541	1.053	1.664	0.958	1.797	0.863	1.940
23	1.257	1.437	1.168	1.543	1.078	1.660	0.986	1.785	0.895	1.920
24	1.273	1.446	1.188	1.546	1.101	1.656	1.013	1.775	0.925	1.902
25	1.288	1.454	1.206	1.550	1.123	1.654	1.038	1.767	0.953	1.886
26	1.302	1.461	1.224	1.553	1.143	1.652	1.062	1.759	0.979	1.873
27	1.316	1.469	1.240	1.556	1.162	1.651	1.084	1.753	1.004	1.861
28	1.328	1.476	1.255	1.560	1.181	1.650	1.104	1.747	1.028	1.850
29	1.341	1.483	1.270	1.563	1.198	1.650	1.124	1.743	1.050	1.841
30	1.352	1.489	1.284	1.567	1.214	1.650	1.143	1.739	1.071	1.833
31	1.363	1.496	1.297	1.570	1.229	1.650	1.160	1.735	1.090	1.825
32	1.373	1.502	1.309	1.574	1.244	1.650	1.177	1.732	1.109	1.819
33	1.383	1.508	1.321	1.577	1.258	1.651	1.193	1.730	1.127	1.813
34	1.393	1.514	1.333	1.580	1.271	1.652	1.208	1.728	1.144	1.808
35	1.402	1.519	1.343	1.584	1.283	1.653	1.222	1.726	1.160	1.803
36	1.411	1.525	1.354	1.587	1.295	1.654	1.236	1.724	1.175	1.799
37	1.419	1.530	1.364	1.590	1.307	1.655	1.249	1.723	1.190	1.795
38	1.427	1.535	1.373	1.594	1.318	1.656	1.261	1.722	1.204	1.792
39	1.435	1.540	1.382	1.597	1.328	1.658	1.273	1.722	1.218	1.789
40	1.442	1.544	1.391	1.600	1.338	1.659	1.285	1.721	1.230	1.786
45	1.475	1.566	1.430	1.615	1.383	1.666	1.336	1.720	1.287	1.776
50	1.503	1.585	1.462	1.628	1.421	1.674	1.378	1.721	1.335	1.771
55	1.528	1.601	1.490	1.641	1.452	1.681	1.414	1.724	1.374	1.768
60	1.549	1.616	1.514	1.652	1.480	1.689	1.444	1.727	1.408	1.767
65	1.567	1.629	1.536	1.662	1.503	1.696	1.471	1.731	1.438	1.767
70	1.583	1.641	1.554	1.672	1.525	1.703	1.494	1.735	1.464	1.768
75	1.598	1.652	1.571	1.680	1.543	1.709	1.515	1.739	1.487	1.770
80	1.611	1.662	1.586	1.688	1.560	1.715	1.534	1.743	1.507	1.772
85	1.624	1.671	1.600	1.696	1.575	1.721	1.550	1.747	1.525	1.774
90	1.636	1.679	1.612	1.703	1.589	1.726	1.566	1.751	1.542	1.776
95	1.645	1.687	1.623	1.709	1.602	1.732	1.579	1.755	1.557	1.778
100	1.654	1.694	1.634	1.715	1.613	1.736	1.592	1.758	1.571	1.780
150	1.720	1.746	1.706	1.760	1.693	1.774	1.679	1.778	1.665	1.802
200	1.758	1.778	1.748	1.789	1.738	1.799	1.728	1.810	1.718	1.820

k'= 상수항을 제외한 설명변수의 수

n	k' = 6		k' = 7		k' = 8		k' = 9		k' = 10	
	dL	dU	dL	dU	dL	dU	dL	dU	dL	dU
15	0.447	2.472	0.343	2.727	0.251	2.979	0.175	3.216	0.111	3.438
16	0.502	2.388	0.398	2.624	0.304	2.860	0.222	3.090	0.155	3.304
17	0.554	2.318	0.451	2.537	0.356	2.757	0.272	2.975	0.198	3.184
18	0.603	2.257	0.502	2.461	0.407	2.667	0.321	2.873	0.244	3.073
19	0.649	2.206	0.459	2.396	0.456	2.589	0.369	2.783	0.290	2.974
20	0.692	2.162	0.595	2.339	0.502	2.521	0.416	2.704	0.336	2.885
21	0.732	2.124	0.637	2.290	0.547	2.460	0.461	2.633	0.380	2.806
22	0.769	2.090	0.677	2.246	0.588	2.407	0.504	2.571	0.424	2.734
23	0.804	2.061	0.715	2.208	0.628	2.360	0.545	2.514	0.465	2.670
24	0.837	2.035	0.751	2.174	0.666	2.318	0.584	2.464	0.506	2.613
25	0.868	2.012	0.784	2.144	0.702	2.280	0.621	2.419	0.544	2.560
26	0.897	1.992	0.816	2.117	0.735	2.246	0.657	2.379	0.581	2.513
27	0.925	1.974	0.845	2.093	0.767	2.216	0.691	2.342	0.616	2.470
28	0.951	1.958	0.874	2.071	0.798	2.188	0.723	2.309	0.650	2.431
29	0.975	1.944	0.900	2.052	0.826	2.164	0.753	2.278	0.682	2.396
30	0.998	1.931	0.926	2.034	0.854	2.141	0.782	2.251	0.712	2.363
31	1.020	1.920	0.950	2.018	0.879	2.120	0.810	2.226	0.741	2.333
32	1.041	1.909	0.972	2.004	0.904	2.102	0.836	2.203	0.769	2.306
33	1.061	1.900	0.994	1.991	0.927	2.085	0.861	2.181	0.795	2.281
34	1.080	1.891	1.015	1.979	0.950	2.069	0.885	2.162	0.821	2.257
35	1.097	1.884	1.034	1.967	0.971	2.054	0.908	2.144	0.845	2.236
36	1.114	1.877	1.053	1.957	0.991	2.041	0.930	2.127	0.868	2.216
37	1.131	1.870	1.071	1.948	1.011	2.029	0.951	2.112	0.891	2.198
38	1.146	1.864	1.088	1.939	1.029	2.017	0.970	2.098	0.912	2.180
39	1.161	1.859	1.104	1.932	1.047	2.007	0.990	2.085	0.932	2.164
40	1.175	1.854	1.120	1.924	1.064	1.997	1.008	2.072	0.945	2.149
45	1.238	1.835	1.189	1.895	1.139	1.958	1.089	2.002	1.038	2.088
50	1.291	1.822	1.246	1.875	1.201	1.930	1.156	1.986	1.110	2.044
55	1.334	1.814	1.294	1.861	1.253	1.909	1.212	1.959	1.170	2.010
60	1.372	1.808	1.335	1.850	1.298	1.894	1.260	1.939	1.222	1.984
65	1.404	1.805	1.370	1.843	1.336	1.882	1.301	1.923	1.266	1.964
70	1.433	1.802	1.401	1.837	1.369	1.873	1.337	1.910	1.305	1.948
75	1.458	1.801	1.428	1.834	1.399	1.867	1.369	1.901	1.339	1.935
80	1.480	1.801	1.453	1.831	1.425	1.861	1.397	1.893	1.369	1.925
85	1.500	1.801	1.474	1.829	1.448	1.857	1.422	1.886	1.396	1.916
90	1.518	1.801	1.494	1.827	1.469	1.854	1.445	1.881	1.420	1.909
95	1.535	1.802	1.512	1.827	1.489	1.852	1.465	1.877	1.442	1.903
100	1.550	1.803	1.528	1.826	1.506	1.850	1.484	1.874	1.462	1.898
150	1.651	1.817	1.637	1.832	1.622	1.847	1.608	1.862	1.594	1.877
200	1.707	1.831	1.697	1.841	1.686	1.852	1.675	1.863	1.665	1.874

<표 7 (계속)> Durbin-Watson (1% 유의수준)

n	k' = 1		k' = 2		k' = 3		k' = 4		k' = 5	
	dL	dU	dL	dU	dL	dU	dL	dU	dL	dU
15	0.811	1.070	0.700	1.252	0.591	1.464	0.488	1.704	0.391	1.967
16	0.844	1.086	0.737	1.252	0.633	1.446	0.532	1.663	0.437	1.900
17	0.874	1.102	0.772	1.255	0.672	1.432	0.574	1.630	0.480	1.847
18	0.902	1.118	0.805	1.259	0.708	1.422	0.613	1.604	0.522	1.803
19	0.928	1.132	0.835	1.265	0.742	1.415	0.650	1.584	0.561	1.767
20	0.952	1.147	0.863	1.271	0.773	1.411	0.685	1.567	0.598	1.737
21	0.975	1.161	0.890	1.277	0.803	1.408	0.718	1.554	0.633	1.712
22	0.997	1.174	0.914	1.284	0.831	1.407	0.748	1.543	0.667	1.691
23	1.018	1.187	0.938	1.291	0.858	1.407	0.777	1.534	0.698	1.673
24	1.037	1.199	0.960	1.298	0.882	1.407	0.805	1.528	0.728	1.658
25	1.055	1.211	0.981	1.305	0.906	1.409	0.831	1.523	0.756	1.645
26	1.072	1.222	1.001	1.312	0.928	1.411	0.855	1.518	0.783	1.635
27	1.089	1.233	1.019	1.319	0.949	1.413	0.878	1.515	0.808	1.626
28	1.104	1.244	1.037	1.325	0.969	1.415	0.900	1.513	0.832	1.618
29	1.119	1.254	1.054	1.332	0.988	1.418	0.921	1.512	0.855	1.611
30	1.133	1.263	1.070	1.339	1.006	1.421	0.941	1.511	0.877	1.606
31	1.147	1.273	1.085	1.345	1.023	1.425	0.960	1.510	0.897	1.601
32	1.160	1.282	1.100	1.352	1.040	1.428	0.979	1.510	0.917	1.597
33	1.172	1.291	1.114	1.358	1.055	1.432	0.996	1.510	0.936	1.594
34	1.184	1.299	1.128	1.364	1.070	1.435	1.012	1.511	0.954	1.591
35	1.195	1.307	1.140	1.370	1.085	1.439	1.028	1.512	0.971	1.589
36	1.206	1.315	1.153	1.376	1.098	1.442	1.043	1.513	0.988	1.588
37	1.217	1.323	1.165	1.382	1.112	1.446	1.058	1.514	1.004	1.586
38	1.227	1.330	1.176	1.388	1.124	1.449	1.072	1.515	1.019	1.585
39	1.237	1.337	1.187	1.393	1.137	1.453	1.085	1.517	1.034	1.584
40	1.246	1.344	1.198	1.398	1.148	1.457	1.098	1.518	1.048	1.584
45	1.288	1.276	1.245	1.423	1.201	1.474	1.156	1.528	1.111	1.584
50	1.324	1.403	1.285	1.446	1.245	1.491	1.205	1.538	1.164	1.587
55	1.356	1.427	1.320	1.466	1.284	1.506	1.247	1.548	1.209	1.592
60	1.383	1.449	1.350	1.484	1.317	1.520	1.283	1.558	1.249	1.598
65	1.407	1.468	1.377	1.500	1.346	1.534	1.315	1.568	1.283	1.604
70	1.429	1.485	1.400	1.515	1.372	1.546	1.343	1.578	1.313	1.611
75	1.448	1.501	1.422	1.529	1.395	1.557	1.368	1.587	1.340	1.617
80	1.466	1.515	1.441	1.541	1.416	1.568	1.390	1.595	1.364	1.624
85	1.482	1.528	1.458	1.553	1.435	1.578	1.411	1.603	1.386	1.630
90	1.496	1.540	1.474	1.563	1.452	1.587	1.429	1.611	1.406	1.636
95	1.510	1.552	1.489	1.573	1.468	1.596	1.446	1.618	1.425	1.642
100	1.522	1.562	1.503	1.583	1.482	1.604	1.462	1.625	1.441	1.647
150	1.611	1.637	1.598	1.651	1.584	1.665	1.571	1.679	1.557	1.693
200	1.664	1.684	1.653	1.693	1.643	1.704	1.633	1.715	1.623	1.725

k' = 상수항을 제외한 설명변수의 수

n	k' = 6		k' = 7		k' = 8		k' = 9		k' = 10	
	dL	dU	dL	dU	dL	dU	dL	dU	dL	dU
15	0.303	2.244	0.226	2.530	0.161	2.817	0.107	3.101	0.068	3.374
16	0.349	2.153	0.269	2.416	0.200	2.681	0.142	2.944	0.094	3.201
17	0.393	2.078	0.313	2.319	0.241	2.566	0.179	2.811	0.127	3.053
18	0.435	2.015	0.355	2.238	0.282	2.467	0.216	2.679	0.160	2.925
19	0.476	1.963	0.396	2.169	0.322	2.381	0.255	2.597	0.196	2.813
20	0.515	1.918	0.436	2.110	0.362	2.308	0.294	2.510	0.232	2.714
21	0.552	1.881	0.474	2.059	0.400	2.244	0.331	2.434	0.268	2.625
22	0.587	1.849	0.510	2.015	0.437	2.188	0.368	2.367	0.304	2.548
23	0.620	1.821	0.545	1.977	0.473	2.140	0.404	2.308	0.340	2.479
24	0.652	1.797	0.578	1.944	0.507	2.097	0.439	2.255	0.375	2.417
25	0.682	1.766	0.610	1.915	0.540	2.059	0.473	2.209	0.409	2.362
26	0.711	1.759	0.640	1.889	0.572	2.026	0.505	2.168	0.441	2.313
27	0.738	1.743	0.669	1.867	0.602	1.997	0.536	2.131	0.473	2.269
28	0.764	1.729	0.696	1.847	0.630	1.970	0.566	2.098	0.504	2.229
29	0.788	1.718	0.723	1.830	0.658	1.947	0.595	2.068	0.533	2.193
30	0.812	1.707	0.748	1.814	0.684	1.925	0.622	2.041	0.562	2.160
31	0.834	1.698	0.772	1.800	0.710	1.906	0.649	2.017	0.589	2.131
32	0.856	1.690	0.794	1.788	0.734	1.889	0.674	1.995	0.615	2.104
33	0.876	1.683	0.816	1.776	0.757	1.874	0.698	1.975	0.641	2.080
34	0.896	1.677	0.837	1.766	0.779	1.860	0.722	1.957	0.665	2.057
35	0.914	1.671	0.857	1.757	0.800	1.847	0.744	1.940	0.689	2.037
36	0.932	1.666	0.877	1.749	0.821	1.836	0.766	1.925	0.711	2.018
37	0.950	1.662	0.895	1.742	0.841	1.825	0.787	1.911	0.733	2.001
38	0.966	1.658	0.913	1.735	0.860	1.816	0.807	1.899	0.754	1.985
39	0.982	1.655	0.930	1.729	0.878	1.807	0.826	1.887	0.774	1.970
40	0.997	1.652	0.946	1.724	0.895	1.799	0.844	1.876	0.789	1.956
45	1.065	1.643	1.019	1.704	0.974	1.768	0.927	1.834	0.881	1.902
50	1.123	1.639	1.081	1.692	1.039	1.748	0.997	1.805	0.955	1.864
55	1.172	1.638	1.134	1.685	1.095	1.734	1.057	1.785	1.018	1.837
60	1.214	1.639	1.179	1.682	1.144	1.726	1.108	1.771	1.072	1.817
65	1.251	1.642	1.218	1.680	1.186	1.720	1.153	1.761	1.120	1.802
70	1.283	1.645	1.253	1.680	1.223	1.716	1.192	1.754	1.162	1.792
75	1.313	1.646	1.284	1.682	1.256	1.716	1.227	1.746	1.199	1.785
80	1.338	1.653	1.312	1.683	1.285	1.714	1.259	1.745	1.232	1.777
85	1.362	1.657	1.337	1.685	1.312	1.714	1.287	1.743	1.262	1.773
90	1.383	1.661	1.360	1.687	1.336	1.714	1.312	1.741	1.228	1.769
95	1.403	1.666	1.381	1.690	1.358	1.715	1.336	1.741	1.313	1.767
100	1.421	1.670	1.400	1.693	1.378	1.717	1.357	1.741	1.335	1.765
150	1.543	1.708	1.530	1.722	1.515	1.737	1.501	1.752	1.486	1.767
200	1.613	1.735	1.603	1.746	1.592	1.757	1.582	1.768	1.571	1.779

<표 8> Dickey-Fuller t-검정치 분포표

n	Probability of a Smaller Value							
	0.01	0.025	0.05	0.10	0.90	0.95	0.975	0.99
				$\hat{\tau}$				
25	-2.66	-2.26	-1.95	-1.60	0.92	1.33	1.70	2.16
50	-2.62	-2.25	-1.95	-1.61	0.91	1.31	1.66	2.08
100	-2.60	-2.24	-1.95	-1.61	0.90	1.29	1.64	2.03
250	-2.58	-2.23	-1.95	-1.62	0.89	1.28	1.63	2.01
500	-2.58	-2.23	-1.95	-1.62	0.89	1.28	1.62	2.00
∞	-2.58	-2.23	-1.95	-1.62	0.89	1.28	1.62	2.00
				$\hat{\tau}_{\mu}$				
25	-3.75	-3.33	-3.00	-2.63	-0.37	0.00	0.34	0.72
50	-3.58	-3.22	-2.93	-2.60	-0.40	-0.03	0.29	0.66
100	-3.51	-3.17	-2.89	-2.58	-0.42	-0.05	0.26	0.63
250	-3.46	-3.14	-2.88	-2.57	-0.42	-0.06	0.24	0.62
500	-3.44	-3.13	-2.87	-2.57	-0.43	-0.07	0.24	0.61
∞	-3.43	-3.12	-2.86	-2.57	-0.44	-0.07	0.23	0.60
				$\hat{\tau}_{\tau}$				
25	-4.38	-3.95	-3.60	-3.24	-1.14	-0.80	-0.50	-0.15
50	-4.15	-9.80	-3.50	-3.18	-1.19	-0.87	-0.58	-0.24
100	-4.04	-3.73	-3.45	-3.15	-1.22	-0.90	-0.62	-0.28
250	-3.99	-3.69	-3.43	-3.13	-1.23	-0.92	-0.64	-0.31
500	-3.98	-3.68	-3.42	-3.13	-1.24	-0.93	-0.65	-0.32
∞	-3.96	-3.66	-3.41	-3.12	-1.25	-0.94	-0.66	-0.33

R 기초 및 통계분석

인 쇄 ‖ 2019년 1월 25일
발 행 ‖ 2019년 1월 25일

지은이 ‖ 강 기 춘

펴낸곳 ‖ 도서출판 신아문화사
찍은곳 ‖ 일신옵셋인쇄사

ISBN ‖ 978-89-97074-84-6 93320

본 교재는 2018년도 제주대학교 국립대학 육성사업에 의해 지원받았음