

전장 유전체 연관분석 실습

(주)바이오투
전임연구원 강윤주

목차

| Part I. 전장 유전체 연관분석 | Part II. 분석 프로그램 설치 | Part II. GAPIT 실전 |
|-----------------------|---------------------|-------------------|
| 1. 전장 유전체 연관분석에 대한 이해 | 1. R 설치 | 1. GAPIT 입력 파일 작성 |
| 2. 데이터 처리 순서 | 2. R studio 설치 | 2. GAPIT 패키지 |
| 3. GWAS 모델 비교 | | 3. GAPIT 명령어 |
| | | 4. GAPIT 결과 확인 |

Part I. 전장 유전체 연관분석

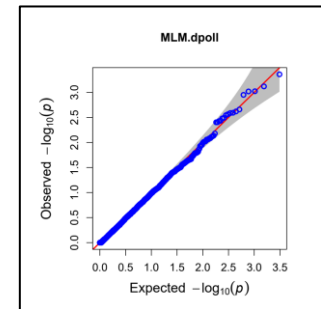
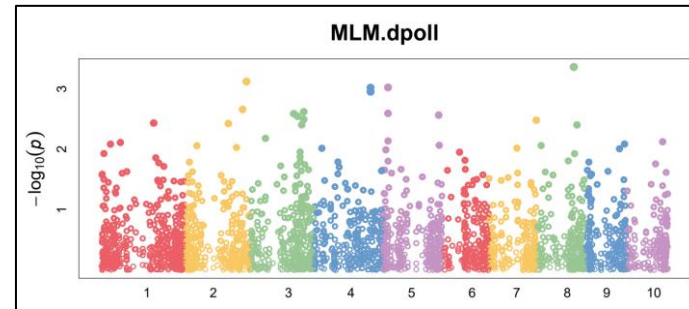
1. 전장 유전체 연관분석에 대한 이해

□ 전장 유전체 연관분석이란?

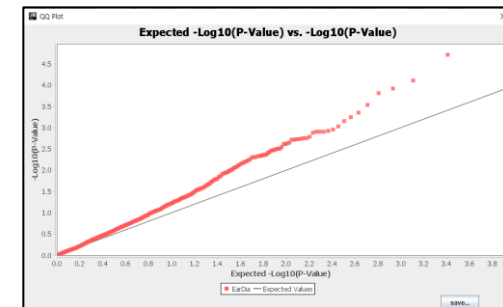
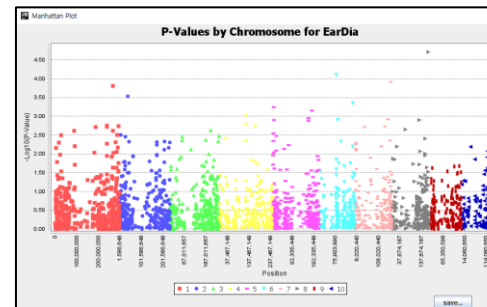
- GWAS (Genome-wide Association Study) 라고 많이 알려진 전장 유전체 연관분석은 전체 게놈(유전체)에 대한 포괄적인 연구를 통해 유전적 변이와 특정 표현형 (질병 또는 형질) 간의 관계를 조사하는 유전학적 분석 방법임.
- 수많은 개별 유전체 데이터와 표현형 데이터를 조합하여 여러가지 통계적인 검정을 통해 특정 표현형과 관련이 있는 유전적 변이의 위치를 분석함.

□ 분석 프로그램

- GAPIT
 - R package를 이용하여 분석
 - 간단한 명령어로 분석이 가능
 - 기본 선형모델, 혼합 선형모델 등에 특화
- TASSEL
 - Windows 기반의 분석 프로그램
 - 사용자의 편의성이 좋음
 - 기본적인 선형모델을 지원 (CLM, MLM)
- PLINK
 - 리눅스 기반의 분석 프로그램
 - 대규모 GWAS 연구에 적합
 - 사용자별 옵션값에 따라 결과의 차이가 크게 나타남



GAPIT 결과 예시



TASSEL 결과 예시

1. 전장 유전체 연관분석에 대한 이해

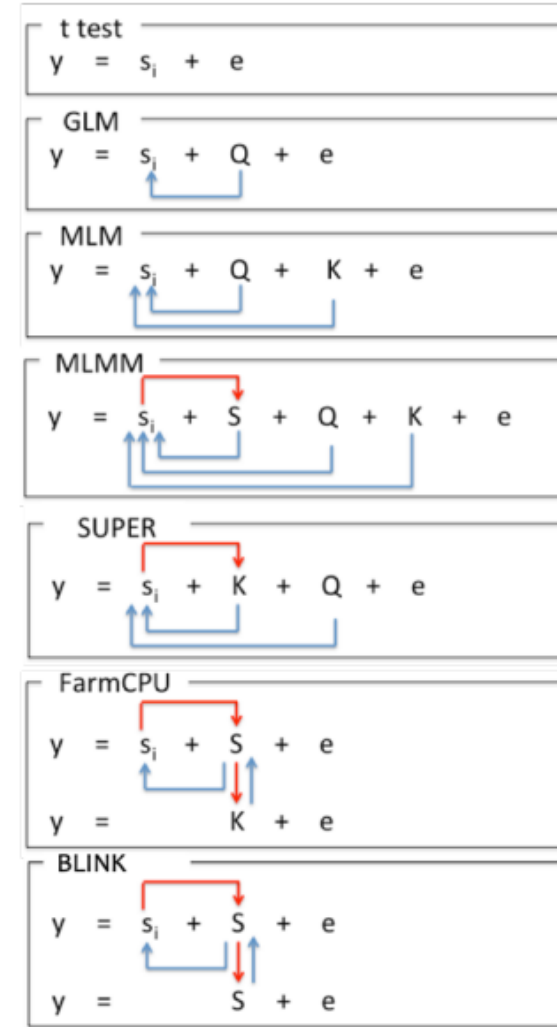
□ GAPIT을 통해 분석하는 이유

- 지속적인 프로그램 업데이트
- 10가지의 GWAS 모델을 구현

| Method | Method paper | Version 1 ¹ | Version 2 ² | Version 3 ³ |
|----------------------------|--|------------------------|------------------------|------------------------|
| General Linear Model (GLM) | Price et al, 2006, <i>Nature Genetics</i> ⁴ | ✓ | ✓ | ✓ |
| Mixed Linear Model (MLM) | Yu et al, 2005, <i>Nature Genetics</i> ⁵ | ✓ | ✓ | ✓ |
| Compression MLM (CMLM) | Zhang et al, 2010, <i>Nature Genetics</i> ⁶ | ✓ | ✓ | ✓ |
| gBLUP | Zhang et al, 2007, <i>J. Anim. Science</i> ⁷ | ✓ | ✓ | ✓ |
| Enriched CMLM | Li et al, 2014, <i>BMC Biology</i> ⁸ | | ✓ | ✓ |
| SUPER | Wang et al, 2014, <i>PLoS One</i> ⁹ | | ✓ | ✓ |
| MLMM | Segura et al, 2012, <i>Nature Genetics</i> ¹⁰ | | | ✓ |
| FarmCPU | Liu et al, 2016, <i>PloS Genetics</i> ¹¹ | | | ✓ |
| cBLUP and sBLUP | Wang et al, 2019, <i>Heredity</i> ¹² | | | ✓ |
| BLINK | Huang et al, 2019, <i>GigaScience</i> ¹³ | | | ✓ |

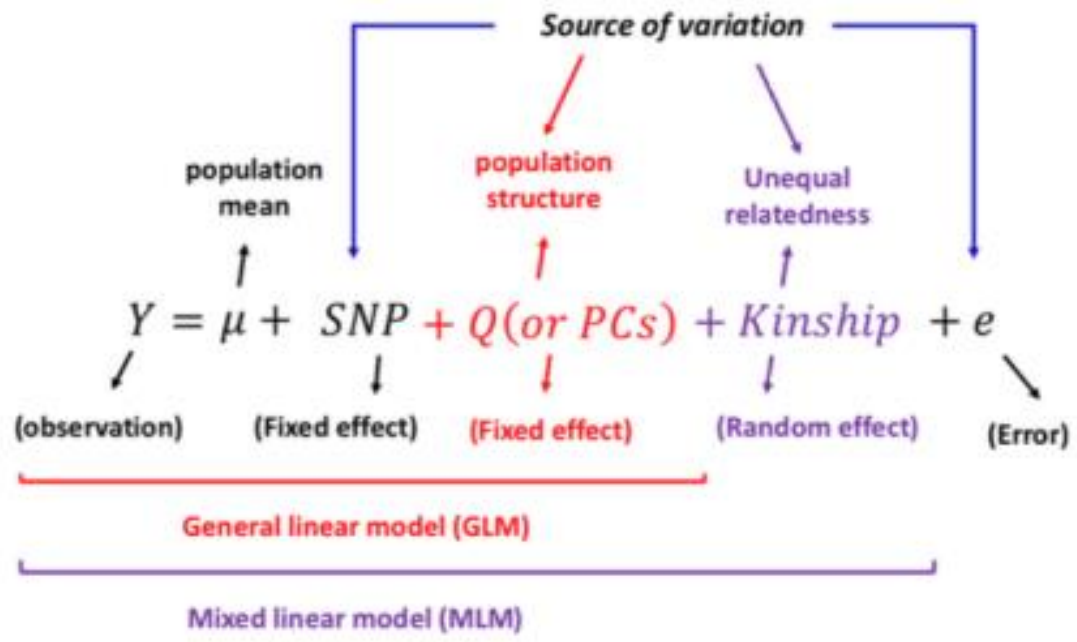
□ 어떤 모델로 분석을 해야하는가?

- 일반적으로 GLM 또는 BLINK를 사용하여 분석
- 다중 유전자좌 모델은 MLMM, FarmCPU, BLINK 이용
- 단일 유전자좌 모델은 GLM, MLM, CMLM, ECMLM 이용



S_i: Testing marker Q: Population structure K: Kinship
 S: Pseudo QTNs Q: Population structure Arrow: adjustment

1. 전장 유전체 연관분석에 대한 이해



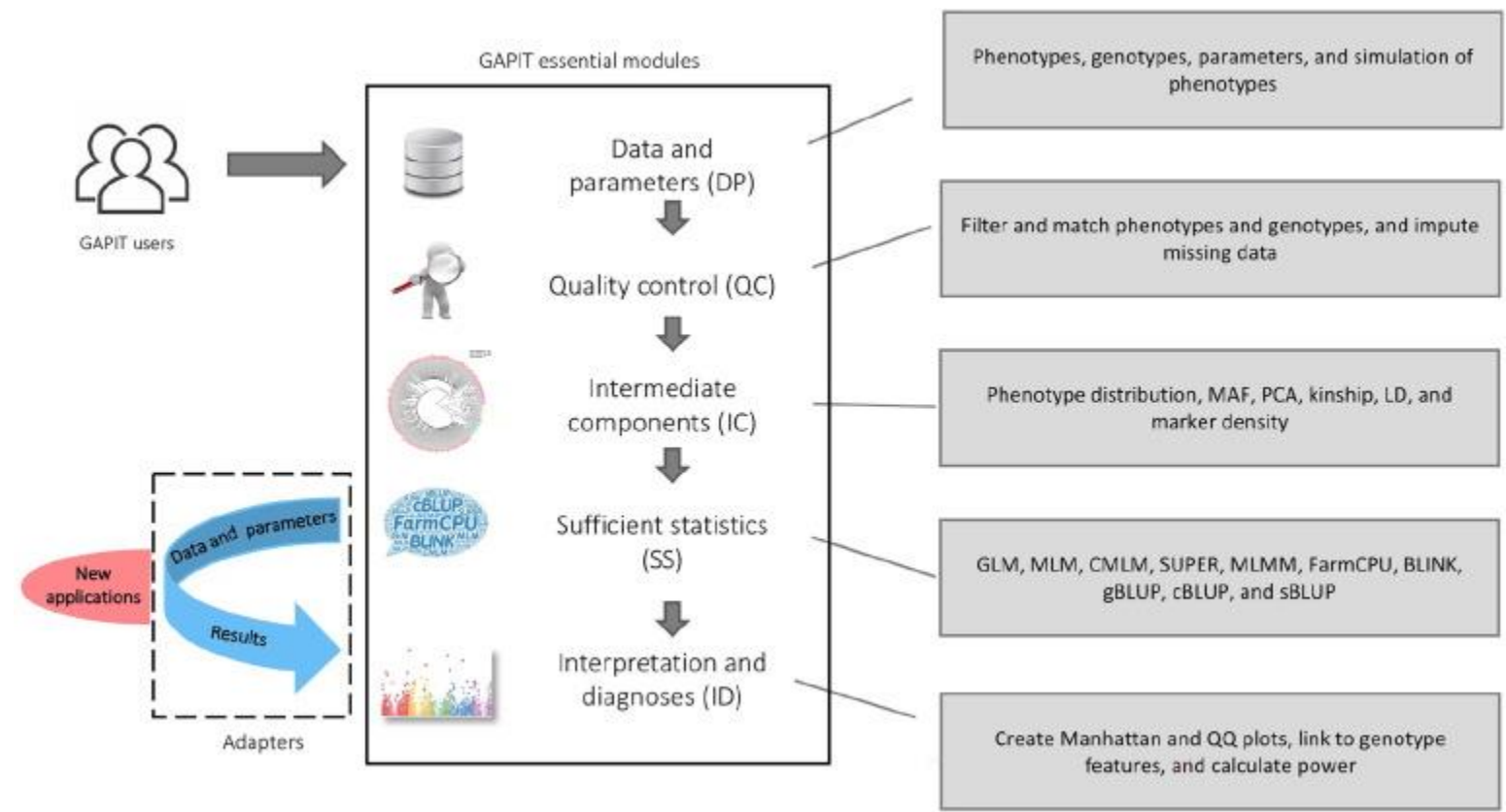
Type I and Type II Error

| Null hypothesis is... | True | False |
|-----------------------|--|---|
| Rejected | Type I error False positive Probability = α | Correct decision True positive Probability = $1 - \beta$ |
| Not rejected | Correct decision True negative Probability = $1 - \alpha$ | Type II error False negative Probability = β |

혼합선형모델은 population structure, kinship 데이터를 동시에 제어함으로써 유형 I, 유형 II의 오류를 줄일 수 있음.

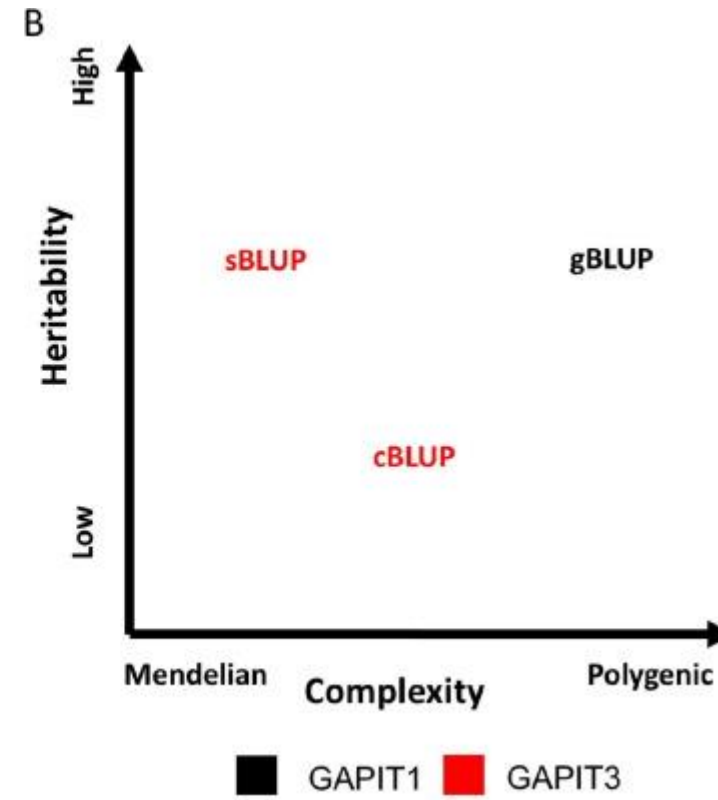
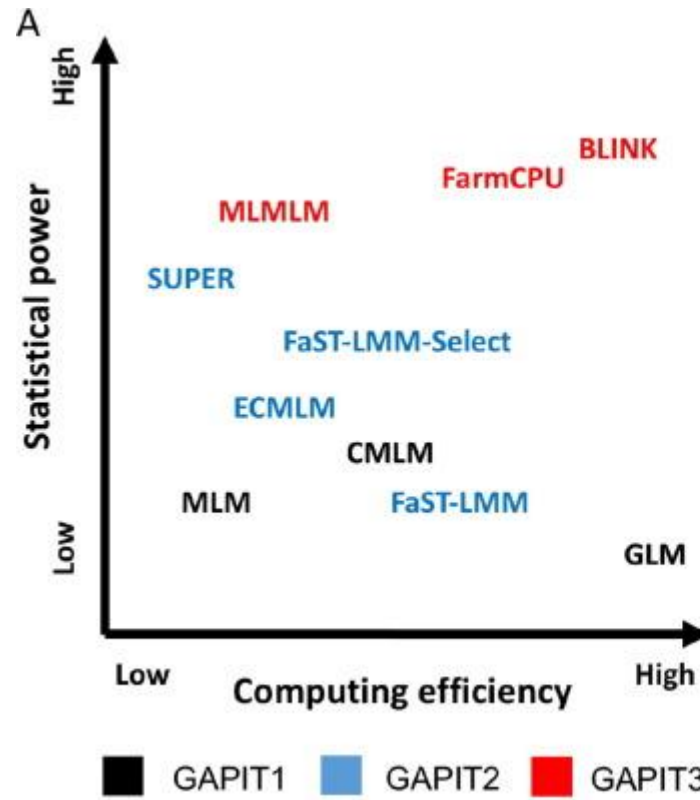
2. GWAS 데이터 처리 순서

□ GAPIT의 데이터 처리 순서



3. GWAS 모델 비교

□ GAPIT의 GWAS 모델 비교



Part II. 분석 프로그램 설치

분석과정 필수 프로그램

□ R (R cran, R gui)

- 통계 계산과 그래픽, 데이터 분석을 위한 프로그래밍 언어이자 소프트웨어 환경
- 오픈소스로 전세계에서 범용적으로 사용되는 언어

□ R studio

- R프로그램 작성 및 실행을 위한 통합 개발환경 프로그램(IDE; Integrated Development Environment)

□ GATK

- Genome Analysis Tool Kit
- NGS 데이터를 이용한 게놈 분석, 변이 탐색, Annotation 등 가능

□ GAPIT

- R 프로그래밍 언어를 기반의 작성된 R package 로 GWAS 분석 수행
- 접근성이 용이하고 비교적으로 쉽게 활용 가능

1. R 설치

□ R 다운로드

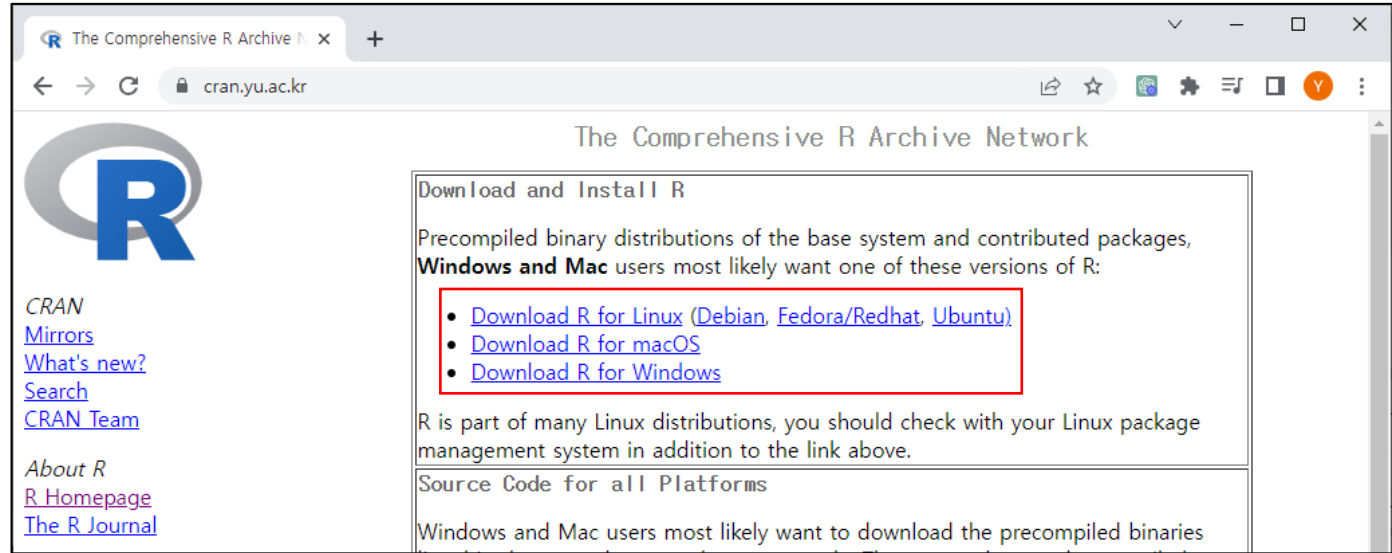
- R 설치 파일 다운로드 (<https://www.r-project.org/>)
- CRAN 클릭 후, Korea 서버 url 클릭

The image shows two browser windows. The top window is the R Project website (r-project.org) with the 'CRAN' link highlighted in a red box. A red arrow points from this link to the 'Korea' mirror entry in the bottom window. The bottom window is the CRAN Mirrors page (cran.r-project.org/mirrors.html) showing a list of mirrors. The 'Korea' mirror entry, with the URL <https://cran.yu.ac.kr/> and 'Yeungnam University' as the provider, is highlighted with a red box.

1. R 설치

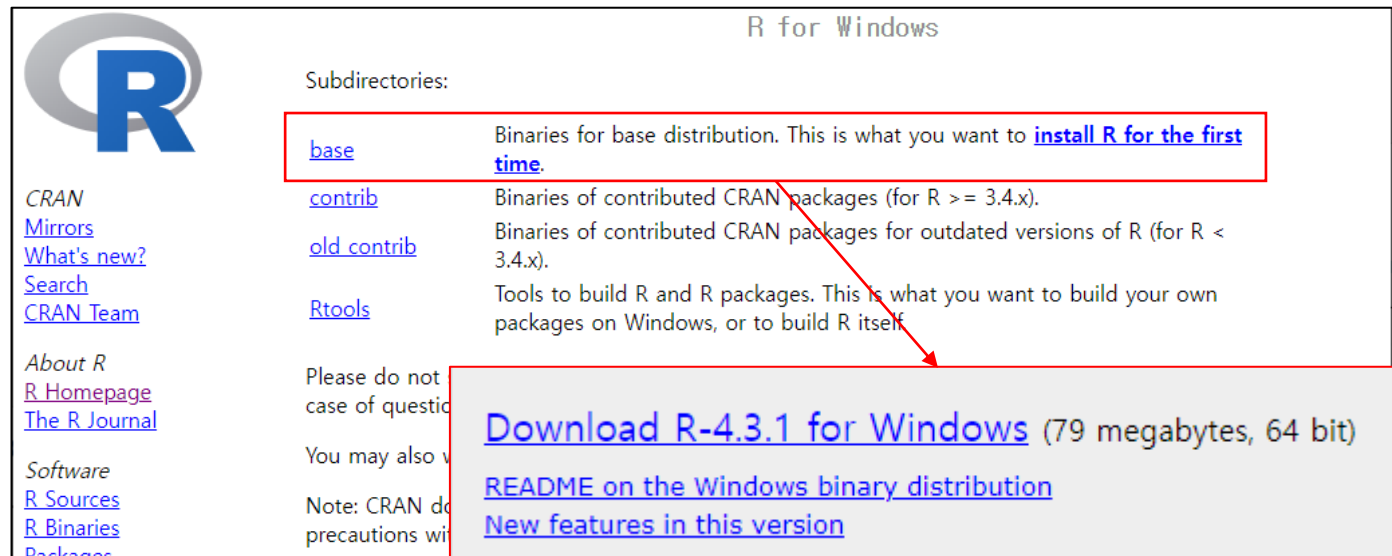
□ R 다운로드

- PC 사양에 맞는 운영체제 선택



- Subdirectories에서 base 선택

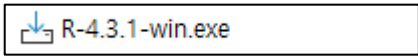
최근 컴퓨터는 대부분 64bit base임.



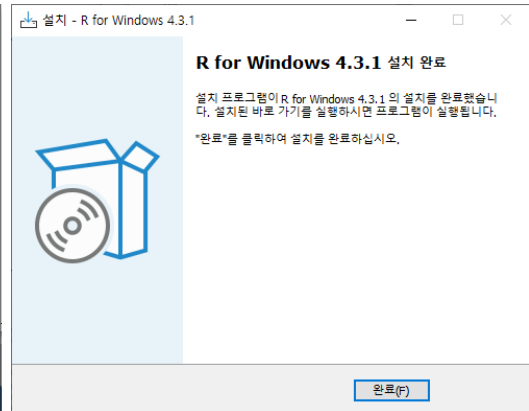
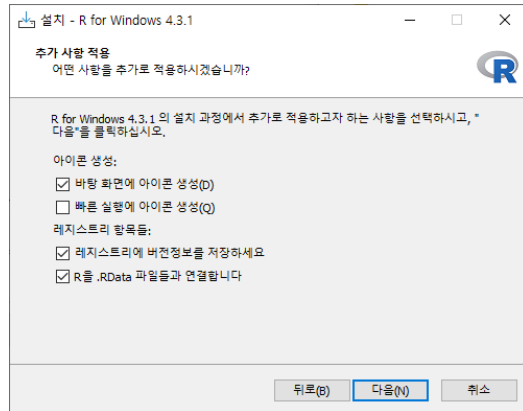
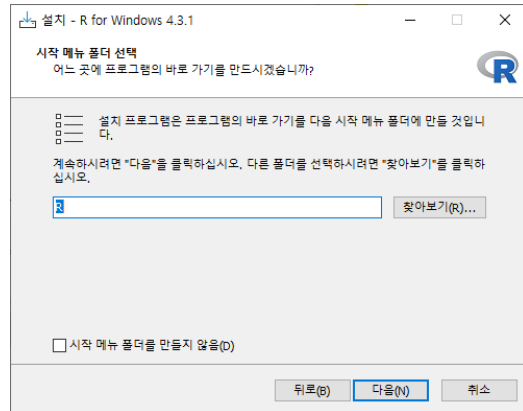
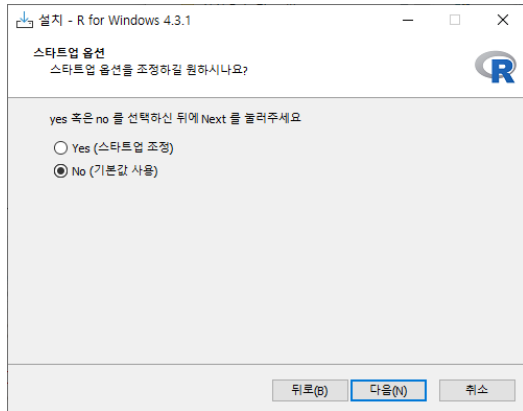
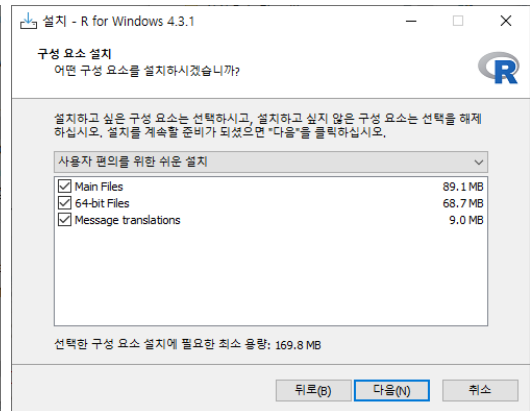
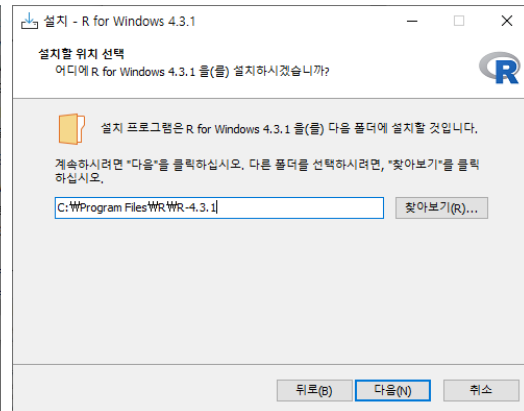
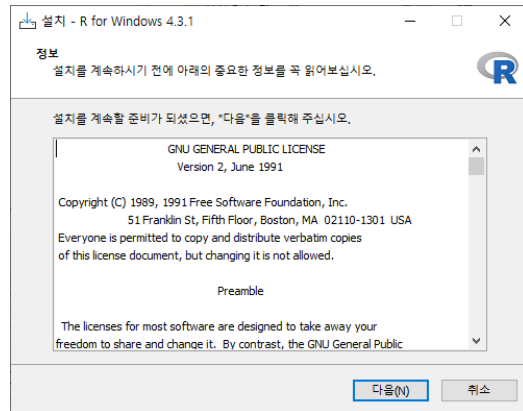
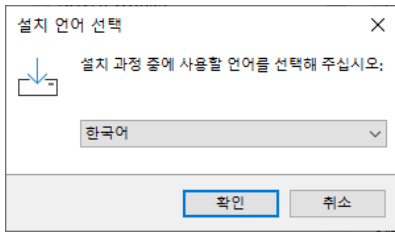
1. R 설치

□ R 설치

- 다운로드 된 파일



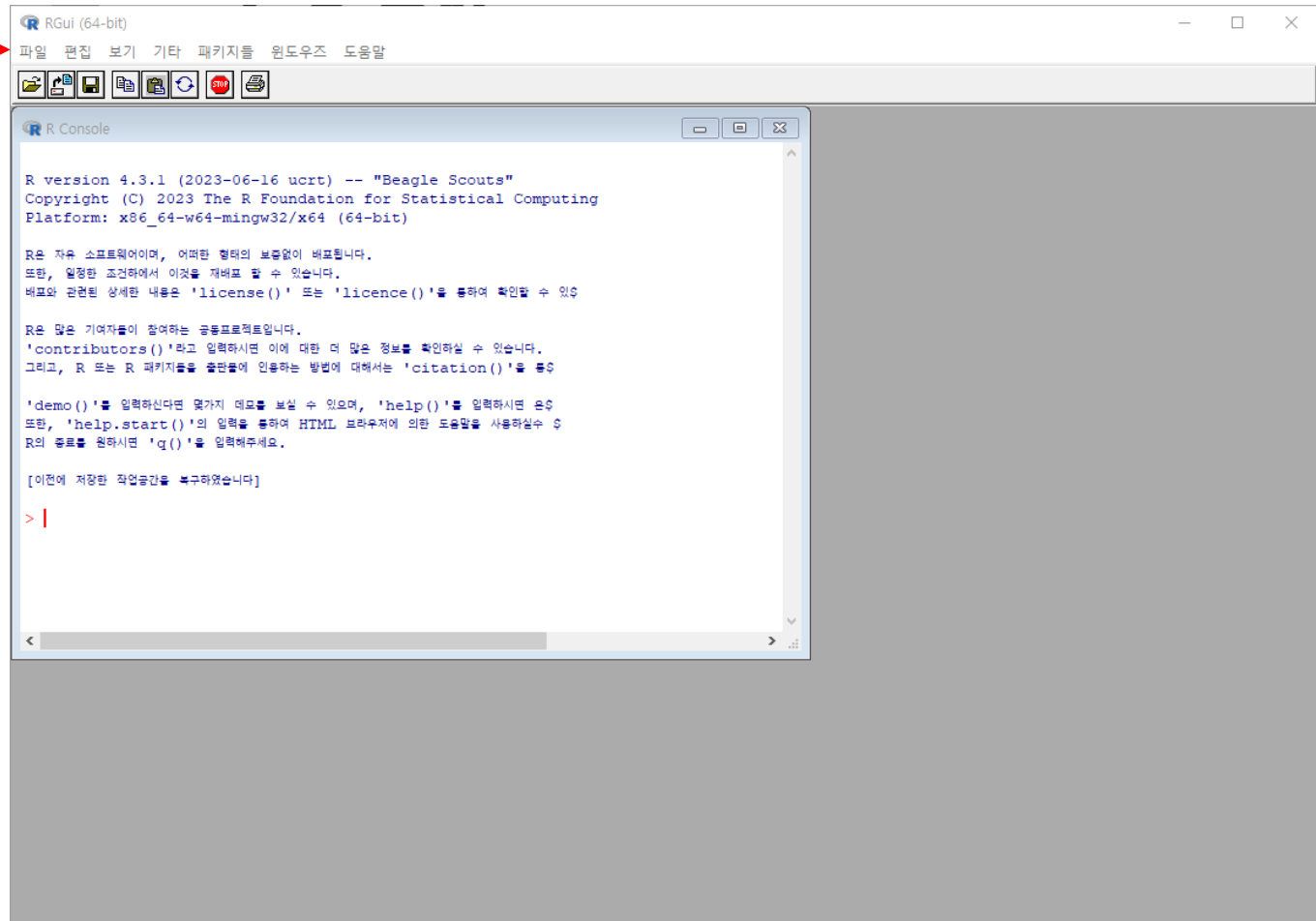
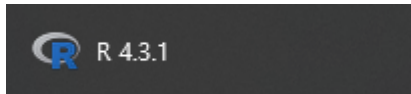
- 다운로드 된 파일 실행



1. R 설치

□ R 실행

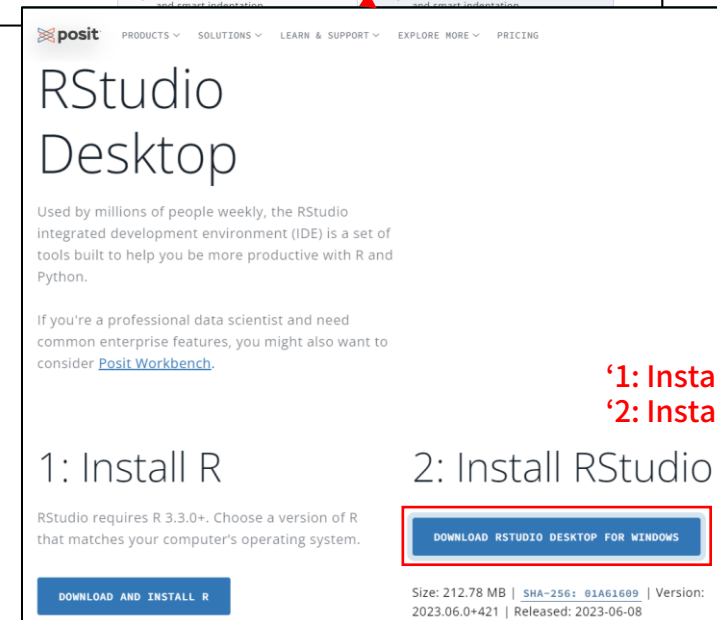
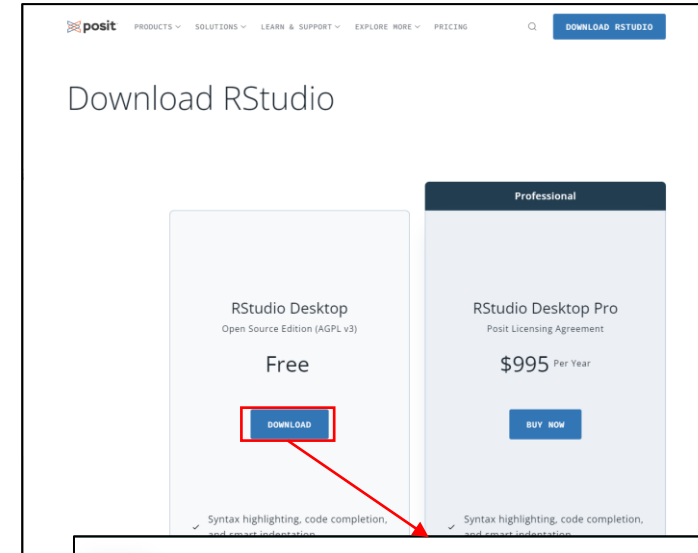
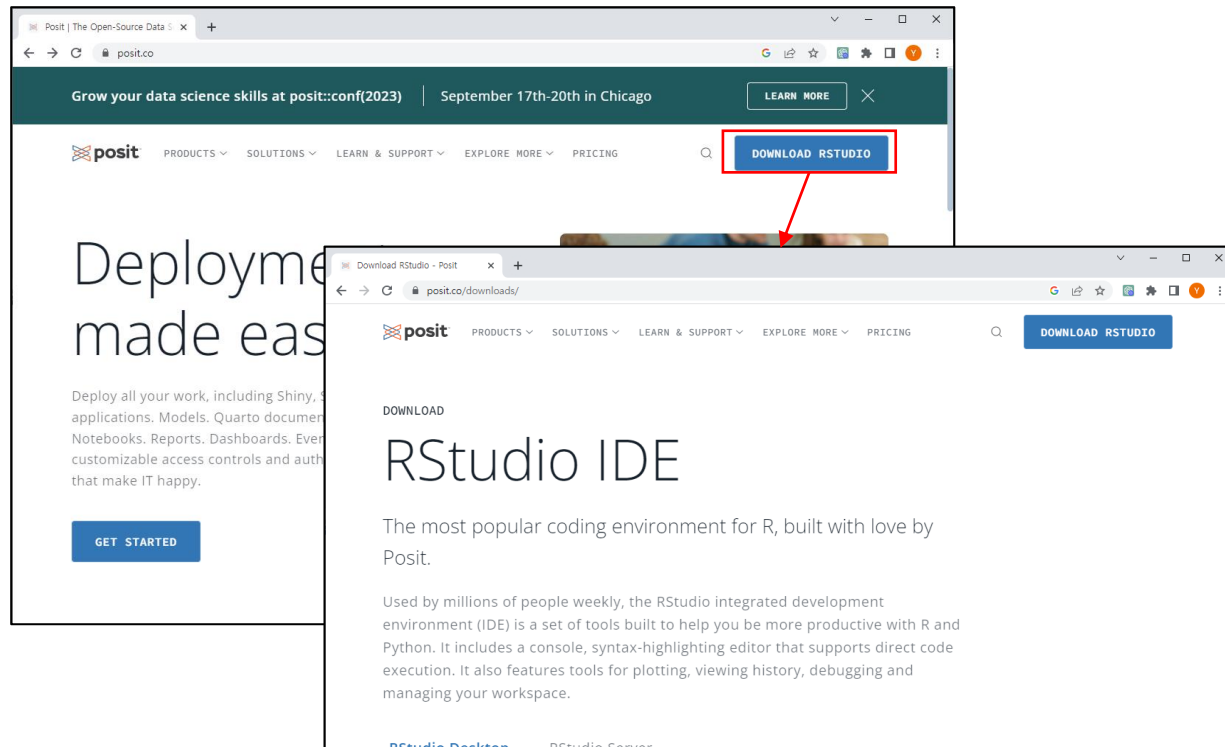
- 설치된 프로그램 실행 (시작메뉴에서 확인 가능)



2. R studio 설치

□ R studio 다운로드

- R을 효과적으로 편리하게 사용할 수 있게 도와주는 통합 개발 환경
- R studio (<https://posit.co/>)
- Download RSTUDIO 클릭
- Rstudio Desktop Free 버전 다운로드

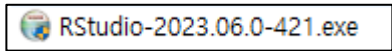


‘1: Install R’은 이미 했으므로,
‘2: Install Rstudio’ 다운로드 진행

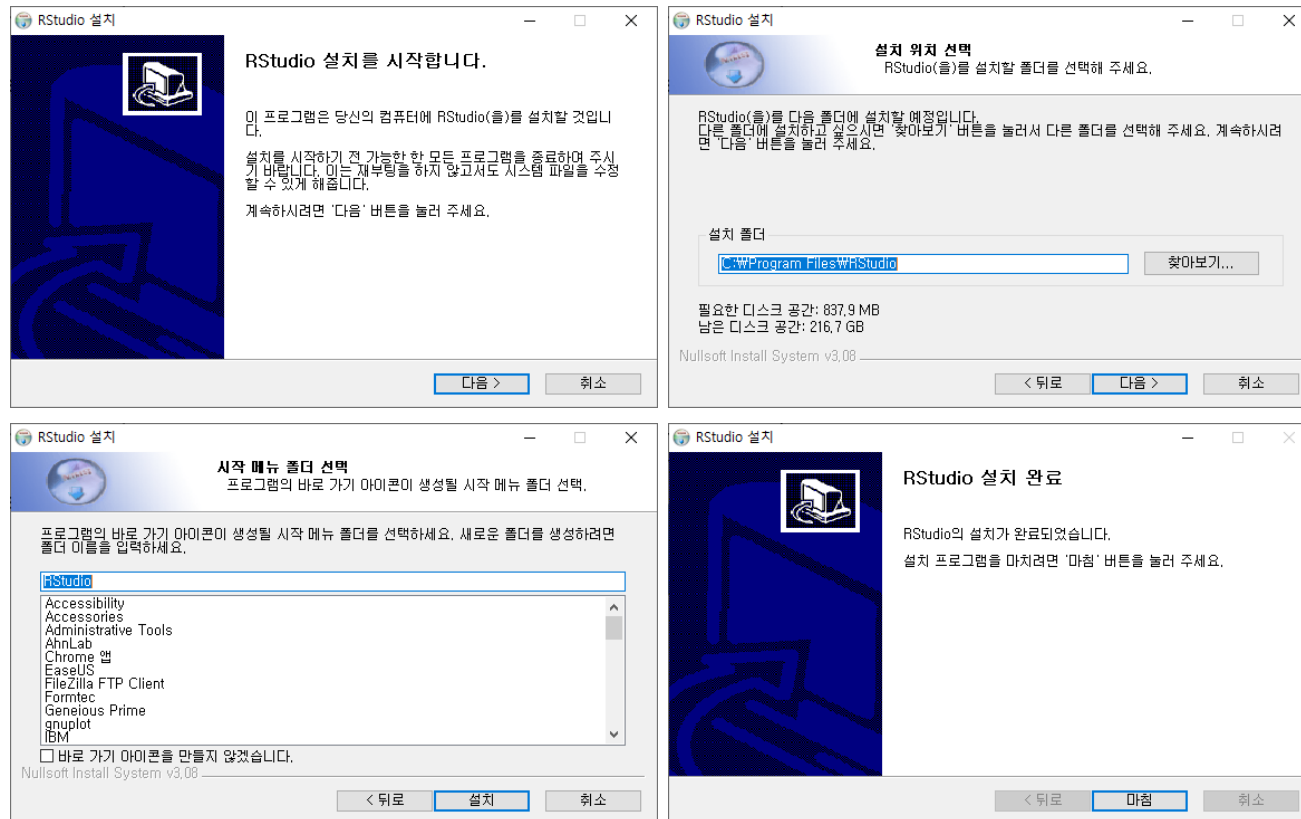
2. R studio 설치

□ R studio 설치

- 다운로드 된 파일



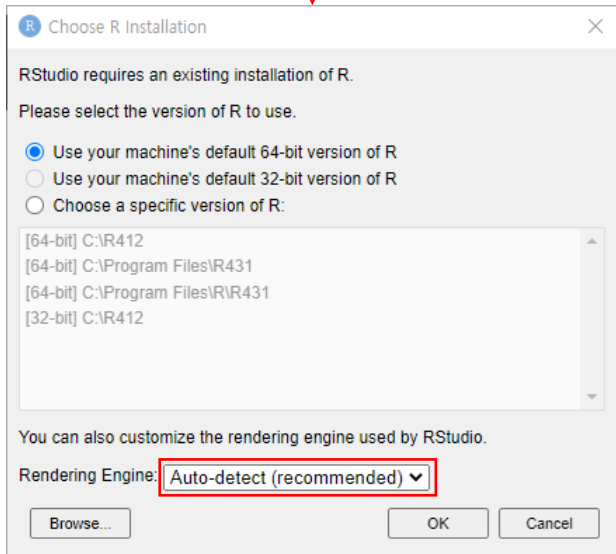
- 다운로드 된 파일 실행



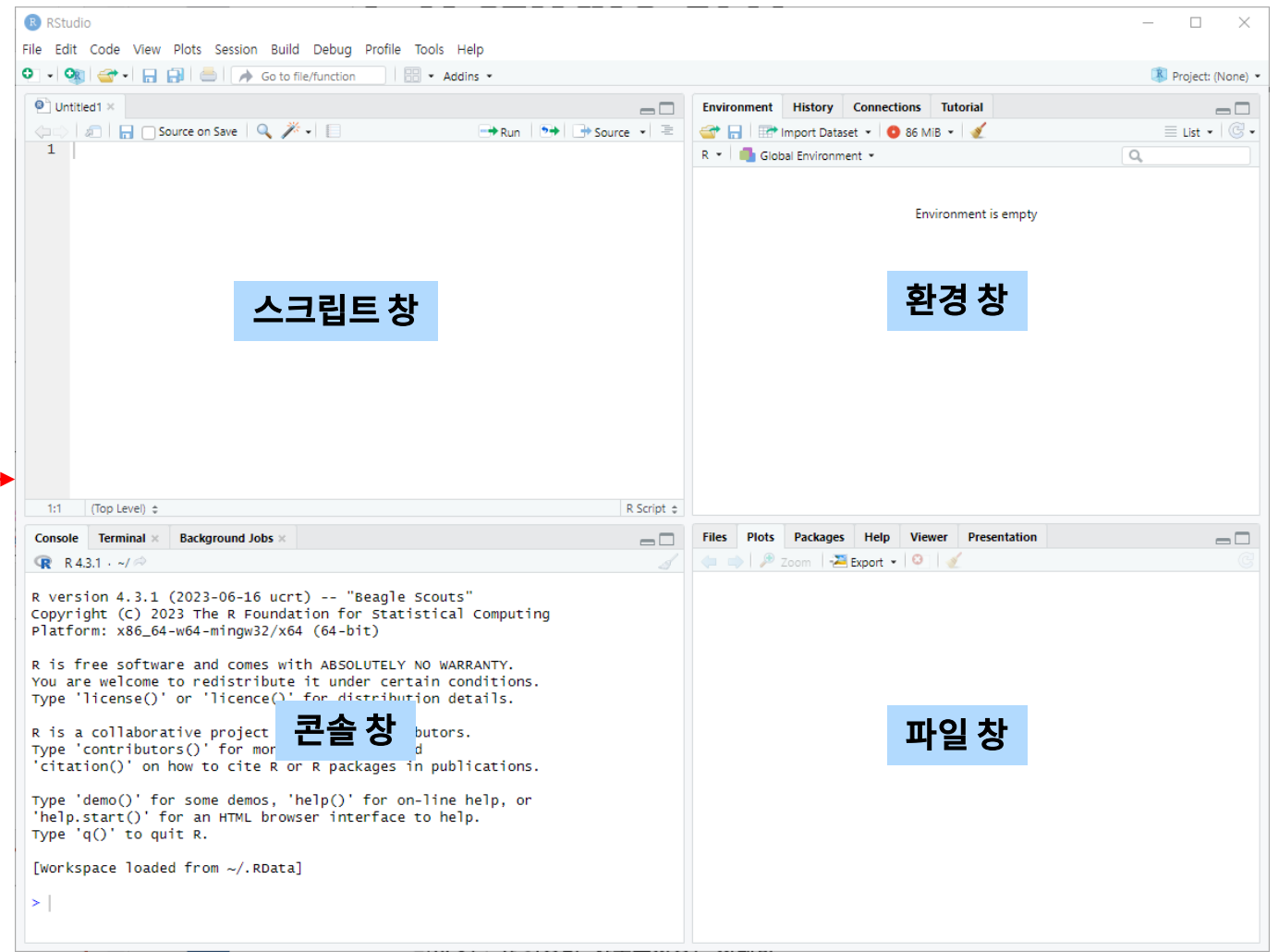
2. R studio 설치

□ R studio 실행

- 설치된 프로그램 실행 (시작메뉴에서 확인 가능)



설치되어 있는 R 버전 리스트가 보이고 선택옵션이 있음.
특별한 경우가 없으면, 자동옵션으로 선택함.



2. R studio 설치

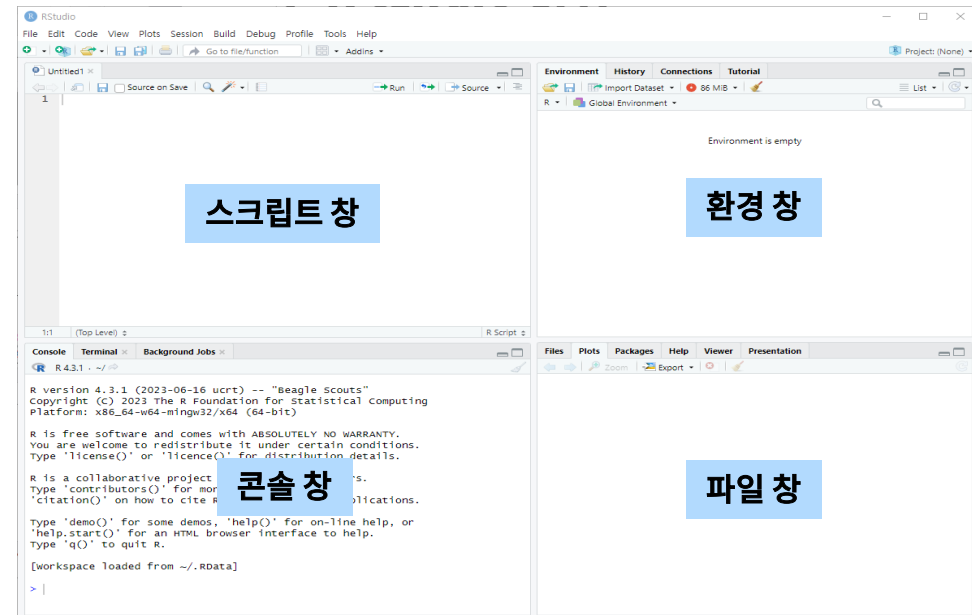
□ R studio 화면 설명

- 스크립트 창
 - 코드를 작성하는 창
 - 긴 코드 작성에 용이
 - 필요한 부분만 선택하여 실행 가능
 - 별도의 코드 파일(.R)로 저장/불러오기 가능
 - 함수에 대한 자동완성 기능 제공

- 콘솔 창
 - 코드 실행 및 결과
 - 오류 확인

- Environment/History/Connections 창
 - Environment : 입력된 데이터 세트 확인
 - History : 실행한 명령어, 결과 등 확인
 - Connections : DB 서버와 연결 관리

- Files/Plots/Packages/Help/Viewer창
 - Files : 파일 탐색기
 - Plots : 그래프 출력
 - Packages : 패키지 관리
 - Help : 도움말

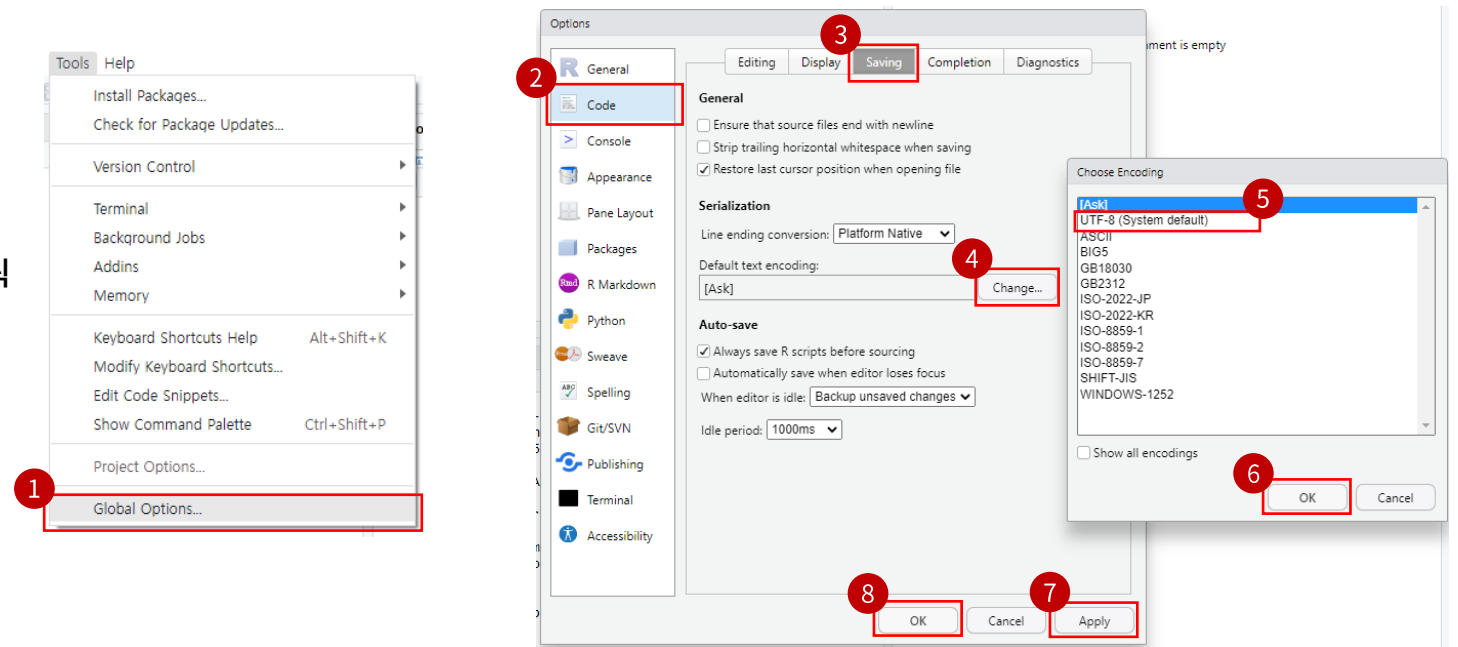


2. R studio 설치

□ R studio 환경설정

- 한글 문제 이슈로 encoding 설정이 필요함.
- UTF-8로 설정

- Tools에서 Global Options 선택
- Code로 이동
- Saving 탭 클릭
- 'Default text encoding' 에서 Change 클릭
- 'UTF-8' 선택
- OK 클릭
- Apply 클릭
- OK 클릭



Part III. GAPIT 실전

GWAS - 분석에 필요한 파일 1. HapMap

□ 파일 확장자명은 '*.hmp.txt' 와 같이 저장되어야 하며 파일 형식은 아래와 같음.

| A | B | C | D | E | F | G | H | I | J | K | L | M | N | O |
|-----------|---------|-------|--------|--------|----------|--------|----------|-----------|-------|--------|----|----|----|----|
| rs | alleles | chrom | pos | strand | assembly | center | protLSID | assayLSID | panel | QCcode | 1 | 2 | 3 | 4 |
| chr01_102 | G/A | 1 | 102663 | + | NA | NA | NA | NA | NA | NA | AG | AG | AG | GG |
| chr01_102 | A/C | 1 | 102714 | + | NA | NA | NA | NA | NA | NA | AC | AC | AC | AA |
| chr01_133 | C/T | 1 | 133318 | + | NA | NA | NA | NA | NA | NA | CC | CC | CC | CC |
| chr01_133 | A/G | 1 | 133405 | + | NA | NA | NA | NA | NA | NA | AA | AA | AA | AA |
| chr01_344 | G/A | 1 | 344263 | + | NA | NA | NA | NA | NA | NA | AA | AA | AA | AA |
| chr01_344 | C/A | 1 | 344362 | + | NA | NA | NA | NA | NA | NA | AA | AA | AA | AA |
| chr01_387 | A/G | 1 | 387783 | + | NA | NA | NA | NA | NA | NA | GG | GG | GG | GG |
| chr01_388 | G/A | 1 | 388166 | + | NA | NA | NA | NA | NA | NA | AA | AA | AA | AA |

SNP 정보

Ref 정보

Genotype



[header 설명]

rs : 마커명

alleles: 염기서열 (biallele)

chrom : 염색체 번호 (숫자 only!)

pos : 염색체 내 위치

| | | | | | | | | | | |
|----------|----|----|----|----|----|----|----|----|----|----|
| Genotype | AA | CC | GG | TT | AG | CT | CG | AT | GT | AC |
| Code | A | C | G | T | R | Y | S | W | K | M |

* 위 그림에서 center, protLSID, assayLSID, panel, QCcode 는 'NA' 처리하여 분석 진행해도 무관함.

* HapMap 파일은 GAPIT, TASSEL 모두 동일한 파일을 사용함.

GWAS - 분석에 필요한 파일 2. 형질정보

□ 파일 확장자명은 '*.txt' 와 같이 저장되어야 하며 파일 형식은 아래와 같음.

1. GAPIT format

| Taxa | Pheno1 | Pheno2 | 형질명 |
|------|--------|--------|-----|
| 1 | 10.8 | 30.1 | |
| 2 | 8.7 | 27.7 | |
| 3 | 10.3 | 27.1 | |
| 4 | 8.4 | 28.2 | |
| 5 | 8 | 29.7 | |
| 6 | 13.2 | 27.6 | |
| 7 | 9.6 | NaN | |
| 8 | 6.5 | 21 | |
| 9 | 9 | 26.8 | |
| 10 | 8.4 | 28.2 | |
| 11 | 8.2 | 24.6 | |
| 12 | 5.5 | 27.7 | |
| 13 | NaN | 24.8 | |

샘플명 형질값

2. TASSEL format

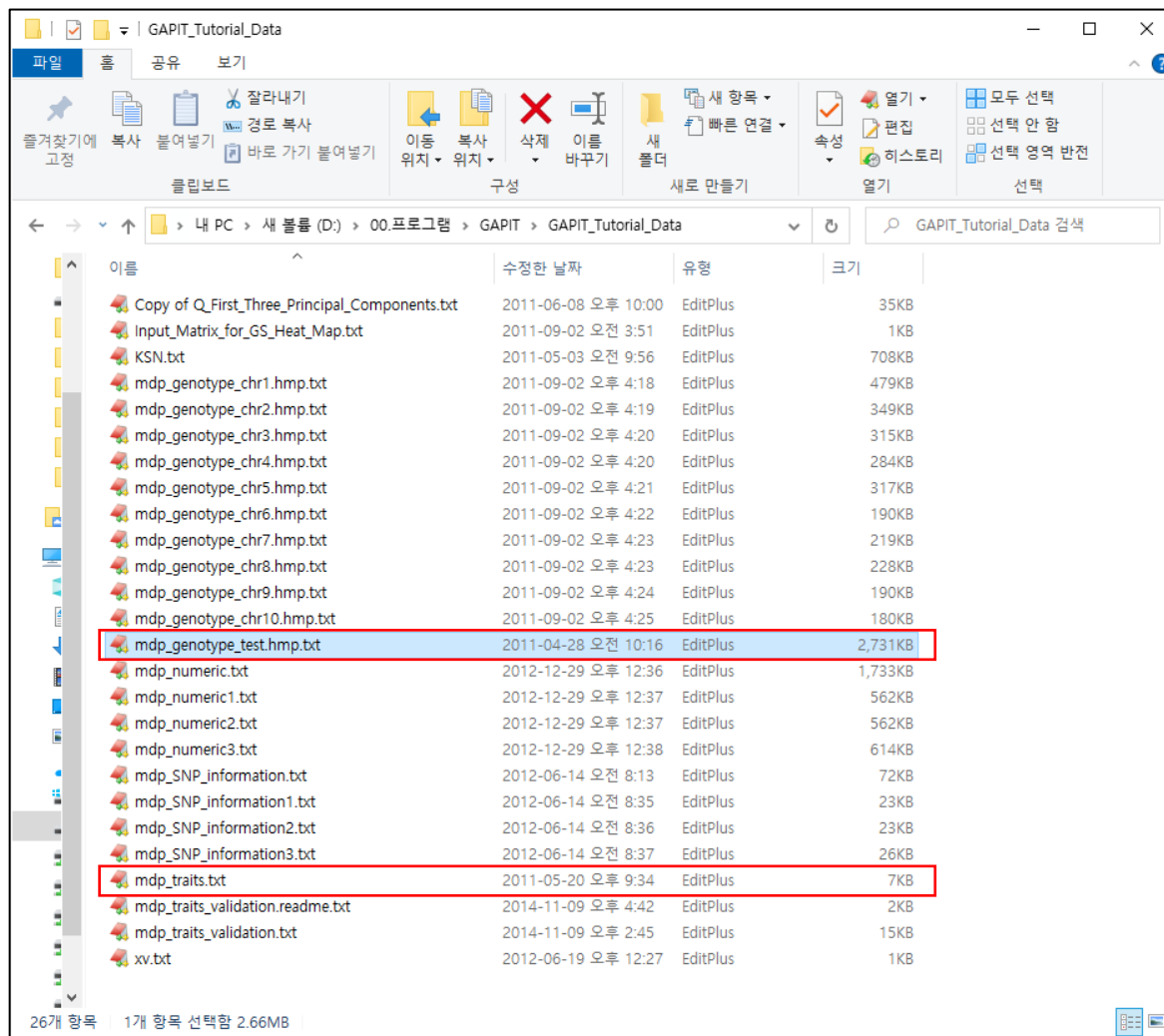
| <Trait> | Pheno1 | Pheno2 | 형질명 |
|---------|--------|--------|-----|
| 1 | 10.8 | 30.1 | |
| 2 | 8.7 | 27.7 | |
| 3 | 10.3 | 27.1 | |
| 4 | 8.4 | 28.2 | |
| 5 | 8 | 29.7 | |
| 6 | 13.2 | 27.6 | |
| 7 | 9.6 | NA | |
| 8 | 6.5 | 21 | |
| 9 | 9 | 26.8 | |
| 10 | 8.4 | 28.2 | |
| 11 | 8.2 | 24.6 | |
| 12 | 5.5 | 27.7 | |
| 13 | NA | 24.8 | |

샘플명 형질값

- * 위 그림과 같이 GAPIT과 TASSEL의 포맷은 거의 유사하나 Missing data의 표기나 1행, 1열의 표시 방법에 차이가 있음.
- * TASSEL에서 Missing data는 '-999' 또는 'NA'로 표기 가능함.
- * 모든 형질은 Tab (\t)으로 구분되어야함.

GWAS 예제 데이터

□ 예제 데이터는 GAPIT에서 제공하는 샘플 데이터로 진행



1. GAPIT 입력 파일 작성

□ VCF 파일에서 hapmap 파일 변환

- 1단계 : GATK 프로그램으로 Variants table 작성

```
gatk VariantsToTable -V <vcf 파일> -R <reference 서열> -F CHROM -F POS -F REF -F ALT -GF GT -O <output.table>
```

실행명령어 1: `gatk VariantsToTable -V combinedflt_snps.PASS.vcf -R reference.fa -F CHROM -F POS -F REF -F ALT -GF GT -O combinedflt_snps.PASS.vcf.table`

- 2단계 : R package 설치 (“optparse”) – 필요한 경우에만 진행

```
> R # R command 실행
> install.packages("optparse") # Package 설치 (온라인 사용 가능한 경우)
> install.packages("optparse_1.7.3.tar.gz", repos=NULL) # Package 설치 (폐쇄망, 오프라인의 경우)
> library("optparse") # 설치된 package 불러오기
> quit() # R command 종료
```

- 3단계 : Variants table 을 hapmap 파일로 변환

```
Rscript convert-vcf-to-hapmap.R -i <output.table> -o <vcf_converted_to_hapmap.txt>
```

실행명령어 2: `Rscript convert-vcf-to-hapmap.R -i combinedflt_snps.PASS.vcf.table -o vcf_converted_to_hapmap.txt`

2. GAPIT 패키지

□ GAPIT 분석에 필요한 기본 R package

- 패키지 리스트

```
"gplots", "LDheatmap", "genetics", "ape", "bigmemory", "EMMREML", "scatterplot3d", "lme4",
"rgl", "multtest", "BiocManager", "snpStats"
```

- R package 설치 (<http://zzlab.net/GAPIT/GAPIT.library.R> 파일 참고)

```
if(!require(gplots)) install.packages("gplots")
if(!require(LDheatmap)) install.packages("LDheatmap")
if(!require(genetics)) install.packages("genetics")
if(!require(ape)) install.packages("ape")
if(!require(compiler)) install.packages("compiler")
if(!require(grid)) install.packages("grid")
if(!require(bigmemory)) install.packages("bigmemory")
if(!require(EMMREML)) install.packages("EMMREML")
if(!require(scatterplot3d)) install.packages("scatterplot3d")
if(!require(lme4)) install.packages("lme4")
if(!require(rgl)) install.packages("rgl")

if(!'multtest'%in% installed.packages()[,"Package"]){
  if (!requireNamespace("BiocManager", quietly = TRUE))
    install.packages("BiocManager")
  BiocManager::install("multtest")
  BiocManager::install("snpStats")
}
```

2. GAPIT 패키지

□ GAPIT 분석에 필요한 기본 R package

- R package 설치 (수동 설치 - 폐쇄망, 오프라인 상황에서 설치할 때 필요함.)

```

setwd("dir") # downloaded_package 폴더 경로 확인 후 입력
getwd() # 설정된 경로 확인
pkPath <- getwd() # 경로를 변수명에 저장

FileName = list.files() # 설정된 경로 내 파일 리스트

# package 수동으로 설치하기

for(i in range(2)){
  for(j in 1:length(FileName)){
    pkName <- paste(pkPath,"/",FileName[j], sep="")
    install.packages(
      pkName,
      repos = NULL,
      dependencies = FALSE
    )
  }
}

```

R studio 스크립트 창에 작성한 결과

```

setwd("dir") # downloaded_package 폴더 경로로 변경
getwd() # 설정된 경로 확인
pkPath <- getwd() # 경로를 변수명에 저장

FileName = list.files() # 설정된 경로 내 파일 리스트

# package 수동으로 설치하기
for(i in range(2)){
  for(j in 1:length(FileName)){
    pkName <- paste(pkPath,"/",FileName[j], sep="")
    install.packages(
      pkName,
      repos=NULL,
      dependencies = FALSE
    )
  }
}

```

2. GAPIT 패키지

□ GAPIT에 필요한 Function 불러오기

- 방법 1: Zhiwu Zhang Lab 웹에서 GAPIT source 가져오기

```
source("http://zzlab.net/GAPIT/gapit_functions.txt")
```

- GAPIT에 필요한 대부분의 패키지는 방법 1의 명령어를 통해 설치됨.
- R의 버전 차이로 일부 패키지는 수동 설치가 필요함.
예시) `install.packages("lme4")`
`install.packages("statmode")`

- 방법 2: GitHub에서 GAPIT source 가져오기

```
install.packages("devtools")
devtools::install_github("jiabowang/GAPIT3", force=TRUE)
library(GAPIT3)
```

- 방법 3: 다운로드 파일에서 GAPIT source 가져오기

```
source("파일경로/파일명")
```

2. GAPIT 패키지

□ GAPIT에 필요한 Function 불러오기

- 방법 1 또는 2를 통해 function 을 불러오면 환경 창에 다음과 같은 function 이 저장된 것을 확인할 수 있음.

The screenshot shows the R Studio Environment pane with the 'Functions' tab selected. It lists various functions installed from the GAPIT package, including 'Blink', 'Blink.BICselection', 'Blink.cor', 'Blink.LDRemove', 'Blink.LDRemoveBlock', 'Blink.LM', 'Blink.Pred', 'Blink.ptor', 'Blink.rtop', 'Blink.SUB', 'circle.plot', 'Densitplot', and several 'emma.delta' and 'emma.eigen' functions. Each function entry includes its name and a brief description of its parameters and return value.

| Function Name | Description |
|--------------------------|--|
| Blink | Large function (574.4 kb) |
| Blink.BICselection | function (Y, Psort = NULL, CV = NULL, GD = NULL, orientation = NULL, BIC.method = "even") |
| Blink.cor | function (Y, GD, w = NULL, orientation = "row", ms = ms, n = ny, m = nm) |
| Blink.LDRemove | function (GDneo = NULL, LD = 0.7, Porder = NULL, bound = FALSE, model = "A", orientation = "row", bl... |
| Blink.LDRemoveBlock | function (GDneo = NULL, LD = NULL, Porder = NULL, bound = FALSE, model = "A", orientation = NULL) |
| Blink.LM | function (y, w = NULL, GDP, orientation = "col") |
| Blink.Pred | function (GD = NULL, Y = NULL, CV = NULL, orientation = "col") |
| Blink.ptor | function (p, df) |
| Blink.rtop | function (r, df) |
| Blink.SUB | function (GM = NULL, GLM = NULL, QTN = NULL, method = "mean", useapply = TRUE, model = "A", no.cv = ... |
| circle.plot | function (myr, type = "l", x = NULL, lty = 1, lwd = 1, col = "black", add = TRUE, n.point = 1000) |
| Densitplot | function (map, col = c("darkblue", "white", "red"), main = "SNP Density", bin = 1e+06, band = 3, wid... |
| emma.delta.ML.dLL.w.Z | function (logdelta, lambda, etas.1, xi.1, n, etas.2.sq) |
| emma.delta.ML.dLL.wo.Z | function (logdelta, lambda, etas.1, xi.1, n, etas.2.sq) |
| emma.delta.ML.LL.w.Z | function (logdelta, lambda, etas.1, xi.1, n, etas.2.sq) |
| emma.delta.ML.LL.wo.Z | function (logdelta, lambda, etas.1, xi.1, n, etas.2.sq) |
| emma.delta.REML.dLL.w.Z | function (logdelta, lambda, etas.1, n, t1, etas.2.sq) |
| emma.delta.REML.dLL.wo.Z | function (logdelta, lambda, etas.1, n, t, etas.2.sq) |
| emma.delta.REML.LL.w.Z | function (logdelta, lambda, etas.1, n, t, etas.2.sq) |
| emma.delta.REML.LL.wo.Z | function (logdelta, lambda, etas.1, n, t, etas.2.sq) |
| emma.eigen.L | function (Z, K, complete = TRUE) |
| emma.eigen.L.w.Z | function (Z, K, complete = TRUE) |
| emma.eigen.L.wo.Z | function (K) |
| emma.eigen.R | function (Z, K, X, complete = TRUE) |
| emma.eigen.R.w.Z | function (Z, K, X, complete = TRUE) |
| emma.eigen.R.wo.Z | function (K, X) |
| emma.kinship | function (snps, method = "additive", use = "all") |
| emma.ML.LRT | function (ys, xs, K, Z = NULL, X0 = NULL, ngrids = 100, llim = -10, ulim = 10, esp = 1e-10, ponly = ... |
| emma.MLE | function (y, X, K, Z = NULL, ngrids = 100, llim = -10, ulim = 10, esp = 1e-10, eig.L = NULL, eig.R = ... |
| emma.REML.t | function (ys, xs, K, Z = NULL, X0 = NULL, ngrids = 100, llim = -10, ulim = 10, esp = 1e-10, ponly = ... |

3. GAPIT 실행

□ GAPIT 파일 입력

- 1단계: GAPIT 실행 경로 설정

```
setwd("폴더경로")
```

* 폴더경로: GAPIT 결과 파일이 저장되는 폴더 위치

- 2단계: GAPIT 입력 데이터 불러오기

```
myY <- read.table("mdp_traits.txt", head=TRUE) # 형질 파일 입력
myG <- read.table("mdp_genotype_test.hmp.txt", head=FALSE) # 유전형 파일 입력
```

- 3단계: 입력된 데이터 확인

```
View(myY)
View(myG)
```

| | Taxa | EarHT | dpoll | EarDia |
|---|-------|-------|-------|----------|
| 1 | 811 | 59.50 | NaN | NaN |
| 2 | 4226 | 65.50 | 59.5 | 32.21933 |
| 3 | 4722 | 81.13 | 71.5 | 32.42100 |
| 4 | 33-16 | 64.75 | 64.5 | NaN |
| 5 | 38-11 | 92.25 | 68.5 | 37.89700 |

| | V1 | V2 | V3 | V4 | V5 | V6 | V7 | V8 | V9 | V10 | V11 | V12 | V13 | V14 |
|---|------------|---------|-------|---------|--------|----------|--------|----------|-----------|----------|--------|-------|-------|------|
| 1 | rs | alleles | chrom | pos | strand | assembly | center | protLSID | assayLSID | panel | QCcode | 33-16 | 38-11 | 4226 |
| 2 | PZB00859.1 | A/C | 1 | 157104 | + | AGPv1 | Panzea | NA | NA | maize282 | NA | CC | CC | CC |
| 3 | PZA01271.1 | C/G | 1 | 1947984 | + | AGPv1 | Panzea | NA | NA | maize282 | NA | CC | GG | CC |
| 4 | PZA03613.2 | G/T | 1 | 2914066 | + | AGPv1 | Panzea | NA | NA | maize282 | NA | GG | GG | GG |
| 5 | PZA03613.1 | A/T | 1 | 2914171 | + | AGPv1 | Panzea | NA | NA | maize282 | NA | TT | TT | TT |

3. GAPIT 실행

□ GAPIT 실행

▪ 1단계 : GAPIT 명령어 작성 (GLM)

```
myGAPIT <- GAPIT(
Y=myY,
G=myG,
PCA.total=3,
model="GLM"
)
```

R studio 스크립트 창에 작성한 결과

```
myGAPIT <- GAPIT(
Y=myY,
G=myG,
PCA.total=3,
model="GLM"
)
```

- model 을 입력하지 않는 경우, default 인 MLM으로 실행됨.
- model 에는 GLM, MLM, MLMM, BLINK, CMLM, ECMLM, FarmCPU, SUPER 등 다양한 통계모델을 선택하여 입력할 수 있음.

▪ 2단계 : GAPIT 수행

- 작성된 GAPIT 명령어 전체 드래그 → **CTRL** + **ENTER**

▪ GAPIT 실행하면 아래 콘솔 창에 아래와 같이 출력됨.

```
Console Terminal Background Jobs
R 3.6.3 · d:/00.프로그램/GAPIT/
> myGAPIT <- GAPIT(
+ Y=myY,
+ G=myG,
+ PCA.total=3,
+ model="GLM"
+ )
[1] "----- welcome to GAPIT -----"
[1] "GLM"
[1] "-----Processing traits-----"
[1] "Phenotype provided!"
[1] "The 1 model in all."
[1] "GLM"
[1] "GAPIT.DP in process..."
[1] "Converting genotype..."
[1] "Converting HapMap format to numerical under model of Middle"
[1] "Perform numericalization"
[1] "succesfully finished converting HapMap which has bits of 2"
[1] "Converting genotype done."
[1] "GAPIT will filter marker with MAF setting !!"
[1] "The markers will be filtered by SNP.MAF: 0"
maf_index
TRUE
3093
[1] "calculating kinship..."
[1] "Number of individuals and SNPs are 281 and 3093"
[1] "calculating ZHANG relationship defined by zhiwu zhang..."
[1] "subtracting mean..."
[1] "Getting x'X..."
[1] "Adjusting..."
[1] "Adjustment by the minimum diagonal"
[1] "Calculating kinship with Zhang method: done"
[1] "kinship calculated"
[1] "Creating heat map for kinship..."
[1] "Kinship heat map created"
[1] "Adding IDs to kinship..."
```

3. GAPIT 실행

□ 다양한 통계 모델로 GAPIT 실행 명령어

▪ GLM

```
myGAPIT <- GAPIT(
Y=myY,
G=myG,
PCA.total=3,
model="GLM"
)
```

R studio 스크립트 창에 작성한 결과

```
myGAPIT <- GAPIT(
Y=myY,
G=myG,
PCA.total=3,
model="GLM"
)
```

▪ MLM

```
myGAPIT <- GAPIT(
Y=myY,
G=myG,
PCA.total=3
)
```

R studio 스크립트 창에 작성한 결과

```
myGAPIT <- GAPIT(
Y=myY,
G=myG,
PCA.total=3
)
```

▪ CMLM

```
myGAPIT <- GAPIT(
Y=myY,
G=myG,
PCA.total=3,
kinship.cluster=c("average", "complete", "ward"),
kinship.group=c("Mean", "Max"),
model="CMLM"
)
```

R studio 스크립트 창에 작성한 결과

```
myGAPIT <- GAPIT(
Y=myY,
G=myG,
PCA.total=3,
kinship.cluster=c("average", "complete", "ward"),
kinship.group=c("Mean", "Max"),
model="CMLM"
)
```

4. GAPIT 결과 확인

□ GWAS 실행이 완료되면 아래와 같이 결과 파일이 생성됨.

- 결과파일은 앞서 설정한 폴더위치에 저장됨.

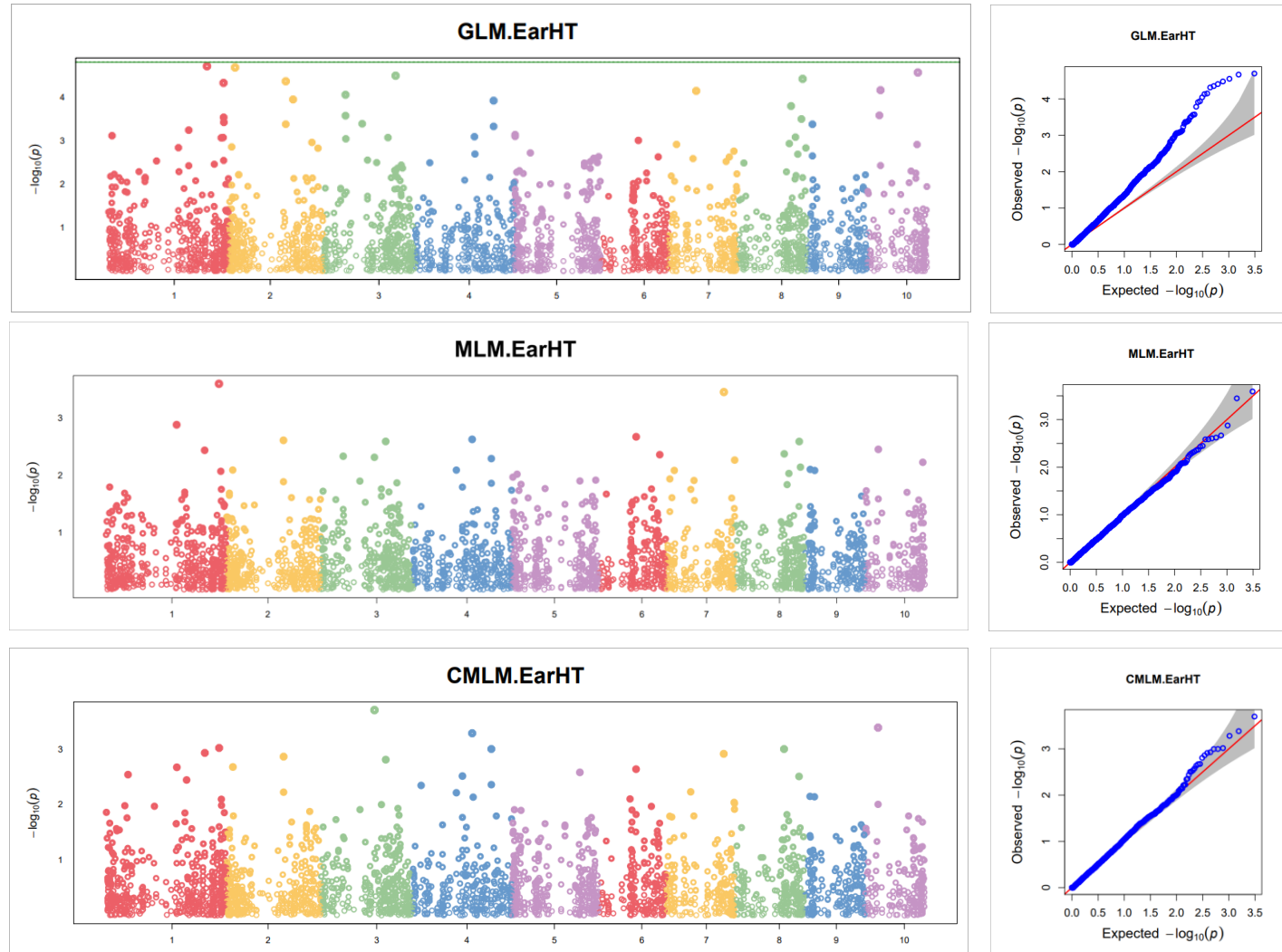
| 이름 | 수정한 날짜 |
|--|--------------------|
| GAPIT.Genotype.Distance_R_Chro.pdf | 2023-07-05 오후 2:17 |
| GAPIT.Genotype.Distance.Rsquare.csv | 2023-07-05 오후 2:17 |
| GAPIT.Association.Vairance_markers.GLM.EarDia.csv | 2023-07-05 오후 2:18 |
| GAPIT.Association.Vairance_markers.GLM.dpoll.csv | 2023-07-05 오후 2:18 |
| GAPIT.Association.Significant_SNPs.GLM.EarDia.pdf | 2023-07-05 오후 2:18 |
| GAPIT.Association.QQs_Tracks.pdf | 2023-07-05 오후 2:18 |
| GAPIT.Association.QQs_Symphysic.pdf | 2023-07-05 오후 2:18 |
| GAPIT.Association.QQ.GLM.EarHT.pdf | 2023-07-05 오후 2:17 |
| GAPIT.Association.QQ.GLM.EarDia.pdf | 2023-07-05 오후 2:18 |
| GAPIT.Association.QQ.GLM.dpoll.pdf | 2023-07-05 오후 2:18 |
| GAPIT.Association.PVE.GLM.EarDia.csv | 2023-07-05 오후 2:18 |
| GAPIT.Association.PVE.GLM.dpoll.csv | 2023-07-05 오후 2:18 |
| GAPIT.Association.Prediction_results.GLM.EarHT.csv | 2023-07-05 오후 2:17 |
| GAPIT.Association.Prediction_results.GLM.EarDia.csv | 2023-07-05 오후 2:18 |
| GAPIT.Association.Prediction_results.GLM.dpoll.csv | 2023-07-05 오후 2:18 |
| GAPIT.Association.Optimum.GLM.EarHT.pdf | 2023-07-05 오후 2:17 |
| GAPIT.Association.Optimum.GLM.EarDia.pdf | 2023-07-05 오후 2:18 |
| GAPIT.Association.Optimum.GLM.dpoll.pdf | 2023-07-05 오후 2:18 |
| GAPIT.Association.Manhattans_Wide.pdf | 2023-07-05 오후 2:18 |
| GAPIT.Association.Manhattans_Symphysic_Traitsnames.csv | 2023-07-05 오후 2:18 |
| GAPIT.Association.Manhattans_Symphysic_Legend.pdf | 2023-07-05 오후 2:18 |
| GAPIT.Association.Manhattans_Symphysic.pdf | 2023-07-05 오후 2:18 |
| GAPIT.Association.Manhattans_High.pdf | 2023-07-05 오후 2:18 |
| GAPIT.Association.Manhattans_Circular.pdf | 2023-07-05 오후 2:18 |
| GAPIT.Association.Manhattan_Geno.GLM.EarHT.pdf | 2023-07-05 오후 2:17 |
| GAPIT.Association.Manhattan_Geno.GLM.EarDia.pdf | 2023-07-05 오후 2:18 |
| GAPIT.Association.Manhattan_Geno.GLM.dpoll.pdf | 2023-07-05 오후 2:18 |

QQ plot

Manhattan plot

4. GAPIT 결과 확인

□ GWAS 모델 별 결과 비교



감사합니다.



대전시 유성구 테크노2로 187, B동 412호



bi@bioto.co.kr



042-710-0077



070-7585-5344