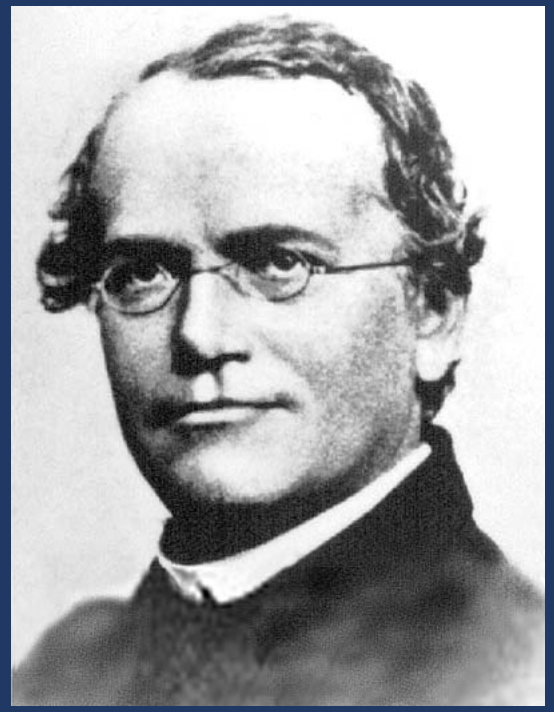
VarianceComponentModels.jl



OpenMendel Workshop

ASHG Annual Meeting 2020

Juhyun Kim

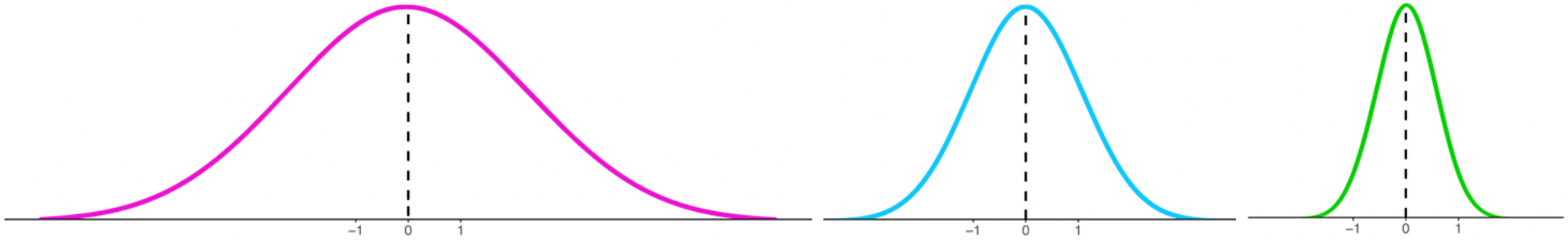
Variance component methods in genetics

Variance component methods are used

- for genetic analysis of quantitative traits (e.g. BMI, cholesterol)
- to assess the strength of genetic effects on a trait
- to characterize the genetic effects on a trait through analyses of gene-gene and gene-environment interaction
- to localize genes influencing a trait through either linkage/association methods
- to explore whether related traits have shared genetic influences in multivariate analyses

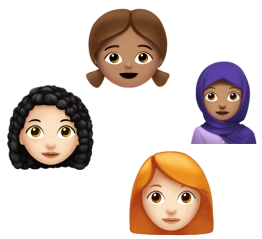
Idea behind variance component methods

Idea: decompose overall variance in a phenotype into particular sources



Phenotypic variance = Genetic variance + Environmental variance

$\sigma_p^2 = \sigma_g^2 + \sigma_e^2$



VarianceComponentModels.jl

Utilities for fitting and testing variance component models

$$Y \sim \text{Normal}(XB, \sigma_1^2 V_1 + \cdots + \sigma_m^2 V_m)$$

In this model, **data** is represented by

- Y: continuously varying quantitative trait(s)
- X: covariates (e.g. sex, age, principal components)
- V_i: structuring matrix corresponding to i-th variance component ($i = 1, \ldots, m$)

and **parameters** are

- B: mean fixed effects coefficient
- σ_i^2: i-th variance component ($i = 1, \ldots, m$)

Feature 1: heritability analysis

Heritability: proportion of the phenotypic variance in a trait that is attributable to genetic effects

1. Additive genetic effect (σ_a^2), unique/unshared environmental effect (σ_e^2)

$$\Omega = \sigma_a^2(2\Phi) + \sigma_e^2 I, \qquad h^2 = \frac{\sigma_a^2}{\sigma_a^2 + \sigma_e^2}$$

```
data1 = VarianceComponentVariate(pheno, X, (2Φgrm, Matrix(1.0I, nobs, nobs)))
```

```
# heritability and its standard error
h, hse = heritability(trait_model.Σ, Σcov)
[h[1], hse[1]]
```

```
2-element Array{Float64,1}:
 0.5735231845909365   ← heritability estimate
 0.2564683288674781   ← standard error
```

Feature 1: heritability analysis

2. Additive genetic effect (σ_a^2), unique/unshared environmental effect (σ_e^2), shared household effect (σ_h^2)

$$\Omega = \sigma_a^2(2\Phi) + \sigma_h^2\Psi_h + \sigma_e^2 I$$

```
data2 = VarianceComponentVariate(pheno, X, (2Φgrm, Ψhousehold, Matrix(1.0I, nobs, nobs)))
```

3. Additive genetic effect (σ_a^2), unique/unshared environmental effect (σ_e^2), shared household effect (σ_h^2), sibling effect (σ_{SP}^2), maternal effect (σ_M^2)

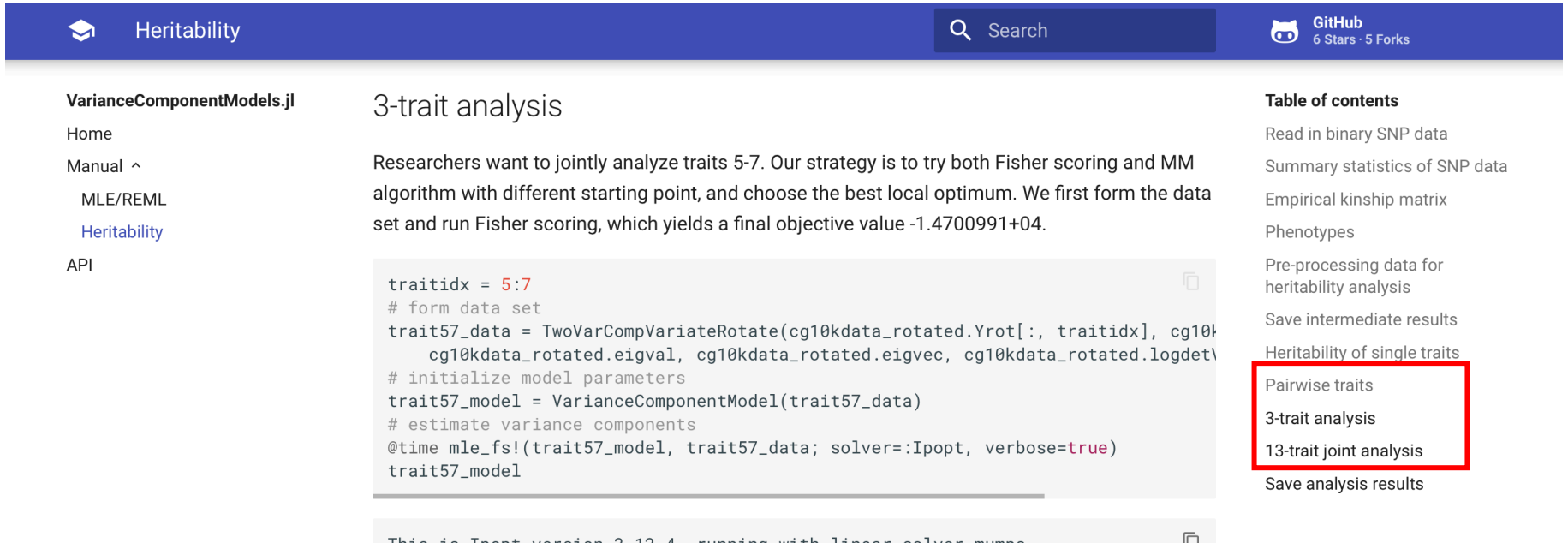
$$\Omega = \sigma_a^2(2\Phi) + \sigma_h^2\Psi_h + \sigma_{SP}^2\Delta + \sigma_M^2\Lambda + \sigma_e^2 I$$

```
data3 = VarianceComponentVariate(pheno, X,
        (2Φgrm, Ψhousehold, Δsp, Λm, Matrix(1.0I, nobs, nobs)))
```

Feature 2: multivariate analysis

When multiple traits are known to be correlated, we want to know whether they are influenced by the same genes

- e.g. hypertension, abdominal obesity, high triglyceride levels, low HDL cholesterol levels



Feature 3: association analysis

- Test whether the mean trait values differ by genotype
  - Likelihood ratio test
- Set up the null model without SNPs

```
X = [ones(nobs) age sex PC1 PC2 PC3 PC4]
nulldata = VarianceComponentVariate(pheno, X,
                                    (2Φgrm, Matrix(1.0I, nobs, nobs)))
nullmodel = VarianceComponentModel(nulldata)
@time nulllogl, nullmodel, = fit_mle!(nullmodel, nulldata; algo=:FS)
```

- Set up the full model with SNPs

```
snp_mat = convert(Vector{Float64}, EUR_subset[:, 10:20])
Xalt = [ones(nobs) age sex PC1 PC2 PC3 PC4 snp_mat]
altdata = VarianceComponentVariate(pheno, Xalt,
                                   (2Φgrm, Matrix(1.0I, nobs, nobs)))
altmodel = VarianceComponentModel(altdata)
@time altlogl, altmodel, = fit_mle!(altmodel, altdata; algo=:FS)
```

Feature 3: association analysis

- Likelihood ratio test (LRT) to test the goodness-of-fit between two models

```
using Distributions
LRT = 2(altlogl - nulllogl)
```

```
11.639045364202502
```

- The associated *p*-value

```
pval = ccdf(Chisq(11), LRT)
```

```
0.3913771524949879
```

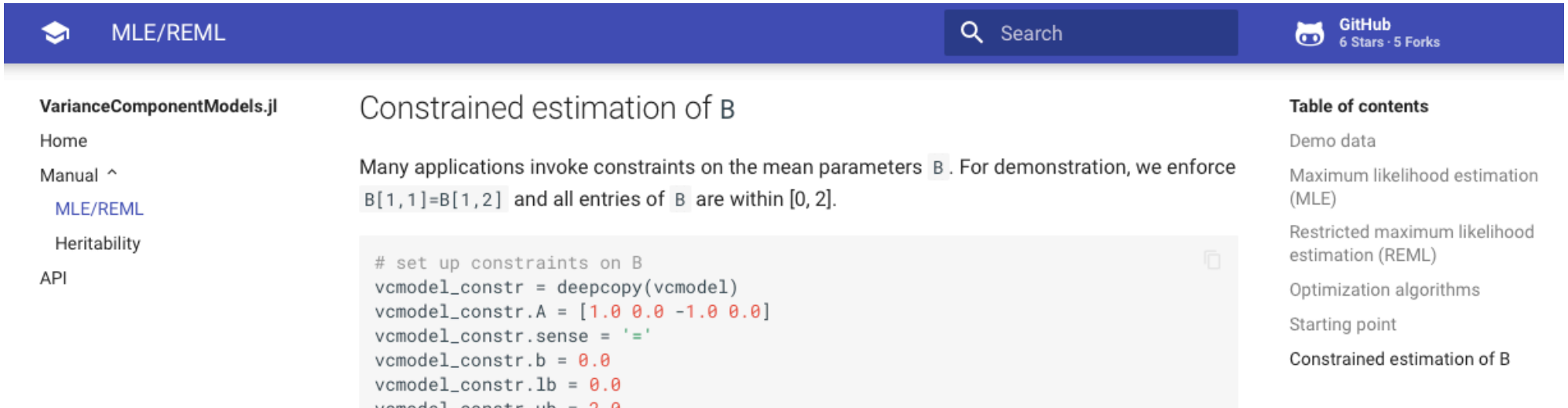
Other features

- Choice of algorithms
  - Fisher-scoring algorithm
  - Minorization-maximization (MM) algorithm

```
fit_mle!(model, data; algo=:FS)
```

```
fit_mle!(model, data; algo=:MM)
```

- Constrained estimation of the mean parameters



For more information

- VarianceComponentModels.jl documentation: https://openmendel.github.io/VarianceComponentModels.jl/latest/
- Jupyter notebook tutorial: https://github.com/OpenMendel/ASHG-OpenMendelWorkshop-2020-Oct/tree/master/05-VarianceComponent-Kim