

# FQ1

2025-10-27

```
#install.packages("pkgbuild")
#pkgbuild::check_build_tools(debug = TRUE)
#install.packages(c("Rcpp", "RcppEigen", "cli", "rlang", "vctrs", "pillar", "lifecycle"))
#install.packages(c("devtools", "remotes"))
#remotes::install_github("bcm-uga/lfmm",
#                        dependencies = TRUE,
#                        build_vignettes = FALSE,
#                        force = TRUE)
#install.packages("RSpectra")

library(lme4)

## Loading required package: Matrix

library(tibble)
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
## 
##     filter, lag

## The following objects are masked from 'package:base':
## 
##     intersect, setdiff, setequal, union

library(tidyverse)

## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## vforcats    1.0.0      vreadr     2.1.5
## vggplot2    3.5.2      vstringr   1.5.1
## vlubridate  1.9.4      vtidyr     1.3.1
## vpurrr      1.0.4

## -- Conflicts ----- tidyverse_conflicts() --
## xtidyr::expand() masks Matrix::expand()
## xdplyr::filter() masks stats::filter()
## xdplyr::lag()    masks stats::lag()
## xtidyr::pack()   masks Matrix::pack()
## xtidyr::unpack() masks Matrix::unpack()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```

library(lfmm)
library(RSpectra)
load("pl_lt_t.lmer.RData")
load("growth_light_time.lmer.RData")

ranef(pl_lt_t.lmer)$pop

##          (Intercept) mean_light_ly_day2      (Intercept)         week
## BH        0.378246926   0.181480161 -0.0368101285 -1.350670e-02
## CC        0.003850522   0.001847453  0.0027536648  1.010411e-03
## CP2       -0.035821753  -0.017187020 -0.0010121687 -3.713815e-04
## DPR       0.046174477   0.022154183  0.0200917877  7.372258e-03
## FR        -0.037939250  -0.018202980  0.0002511353  9.216487e-05
## IH        -0.060110574  -0.028840622  -0.0002465482 -9.048482e-05
## LV1       -0.007809483  -0.003746934  -0.0006203857 -2.276366e-04
## LV3       -0.094441031  -0.045312129  -0.0046039753 -1.689326e-03
## LVTR      -0.017339276  -0.008319260  -0.0012286187 -4.508142e-04
## LVTR1     -0.104884191  -0.050322682  -0.0039090889 -1.434357e-03
## SC        0.142193502   0.068223422  -0.0071350506 -2.618062e-03
## SQ1       -0.085713634  -0.041124786  -0.0009634537 -3.535201e-04
## SQ2       -0.035590323  -0.017075982  -0.0029852876 -1.095385e-03
## SQ3       -0.060720881  -0.029133443  -0.0010960915 -4.021841e-04
## TM2        0.275835929   0.132344099  0.0312382164  1.146214e-02
## WL1       -0.127071382  -0.060967937  0.0025509083  9.360006e-04
## WL2       -0.032789145  -0.015731996  -0.0032524609 -1.193411e-03
## WR        -0.035271989  -0.016923247  -0.0024615937 -9.032279e-04
## WV        -0.002875235  -0.001379517  -0.0003563765 -1.307643e-04
## Y011      -0.019680101  -0.009442371  -0.0020999292 -7.705231e-04
## Y04       -0.008904257  -0.004272199  -0.0006766998 -2.482998e-04
## Y07       -0.040782978  -0.019567380  -0.0069956354 -2.566902e-03
## Y08       -0.038555871  -0.018498830  -0.0032970665 -1.209786e-03

```

```

pheno_2022 <- ranef(pl_lt_t.lmer)$pop %>%
  as_tibble(rownames = "pop", .name_repair = "unique") %>%
  rename(blup_intercept = `"(Intercept)...1`",
         blup_light = `mean_light_ly_day2`) %>%
  mutate(model = "Y2022")

```

```

## New names:
## * '(Intercept)' -> '(Intercept)...1'
## * '(Intercept)' -> '(Intercept)...3'

```

```

pheno_2023 <- ranef(growth_light_time.lmer)$parent_pop %>%
  as_tibble(rownames = "parent_pop", .name_repair = "unique") %>%
  rename(blup_intercept = `"(Intercept)...1`",
         blup_light = `weekly_avg_SlrW2`) %>%
  mutate(model = "Y2023")

```

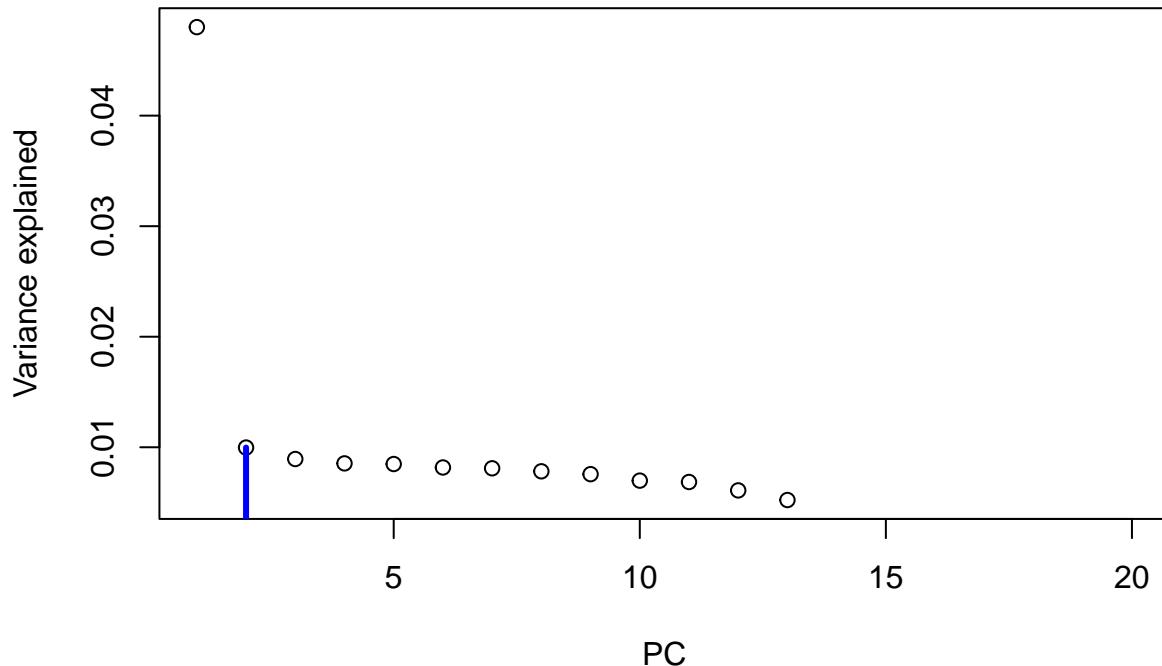
```

## New names:
## * '(Intercept)' -> '(Intercept)...1'
## * '(Intercept)' -> '(Intercept)...3'

```

```
geno_22 <- read.delim("Data/merged_maf_common_UCD2022.tsv", header = TRUE)
geno_23 <- read.delim("Data/merged_maf_common_WL2_2023.tsv", header = TRUE)
```

```
Y <- as.matrix(geno_22[,-1])
pc <- prcomp(Y)
plot(pc$sdev[1:20]^2, xlab = 'PC', ylab = "Variance explained")
points(2,pc$sdev[2]^2, type = "h", lwd = 3, col = "blue")
```



```
Y <- as.matrix(geno_22[,-1])
Y <- t(Y)
X <- matrix(scale(pheno_2022$blup_light), ncol = 1)
rownames(X) <- pheno_2022$pop
colnames(X) <- "blup_light"
common <- intersect(rownames(X), rownames(Y))
Y <- Y[common,]
X <- X[common, , drop = FALSE]
mod.lfmm <- lfmm_ridge(Y = Y,
                        X = X,
                        K = 2)
```

```
## performs association testing using the fitted model:
```

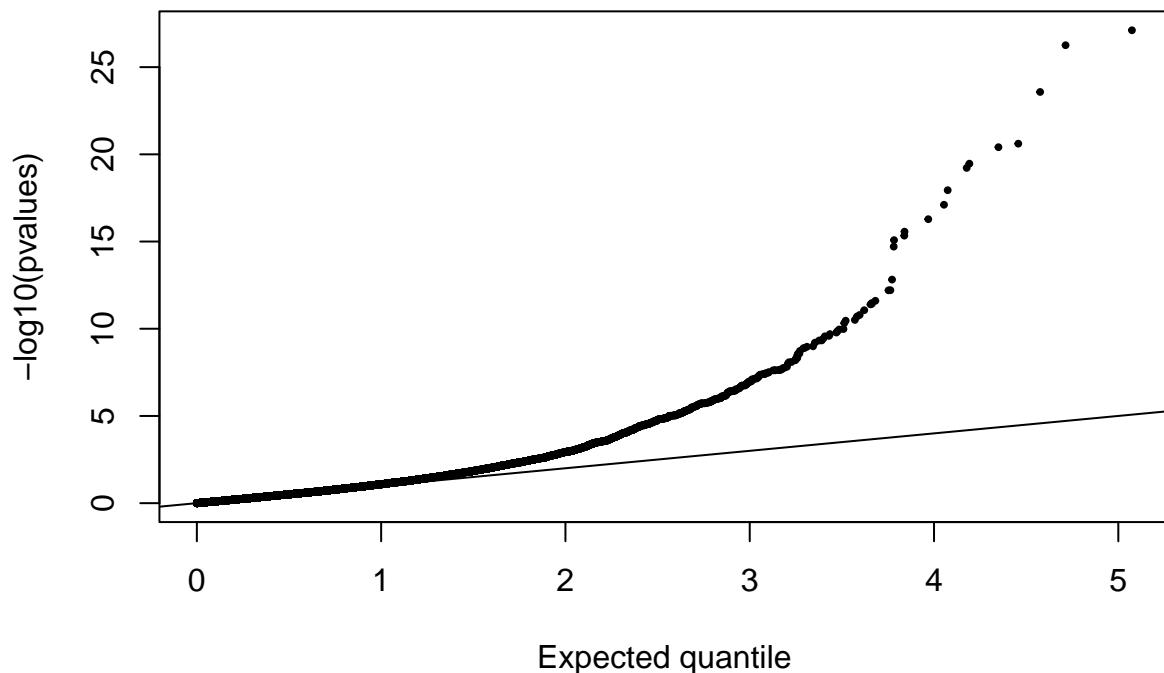
```
pv <- lfmm_test(Y = Y,
                  X = X,
                  lfmm = mod.lfmm,
```

```

    calibrate = "gif")
pvalues <- pv$calibrated.pvalue

#QQ plot
qqplot(rexp(length(pvalues), rate = log(10)),
       -log10(pvalues), xlab = "Expected quantile",
       pch = 19, cex = .4)
abline(0,1)

```



```

## Manhattan plot
plot(-log10(pvalues),
      pch = 19,
      cex = .2,
      xlab = "SNP", ylab = "-Log P",
      col = "grey")

```

