Practical 2: Basic Quantitative Genetics

Time schedule of practical session 2:

11:15	Question to lectures, multiple-choice question and follow up on previous multiple choice questions
11:30	Today´s exercise and assignment to groups
12:00	15 minutes break
12:30	Go through excercises using final word
12:50	Repeat multiple-choice questions
13:00	End of practical session 2

Introduction:

The first section of this practical include a simulated data example for a single locus model. In the last section of this practical we use R for explorative data analyses of two quantitative traits, body weight and blood glucose levels, observed in the F2 mouse population. We will be characterizing and investigating the potential effects of a single marker locus. This includes computation of allele and genotype frequencies, evaluating different genetic models, and estimation of the breeding values and genetic variances for the single marker locus.

Furthermore you may also want to explore these `shinyapps` that could help understand some of the basic concepts of quantitative genetics:

https://neyhartj.shinyapps.io/qgshiny/

https://shiny.cnsgenomics.com/Falconer2/

Simulated Data Example for a Single Locus Model

The phenotype is LDL-cholesterol level and we assume that the trait distributions of individuals with 0, 1 or 2 copies of allele T at SNP rs11591147 are Normal distributions with SD=1 and with means of 0.02, -0.40 and -2.00, respectively. Allele T frequency is 4% in Finland. Let's simulate $n = 10,000$ individuals and boxplot them by genotype.

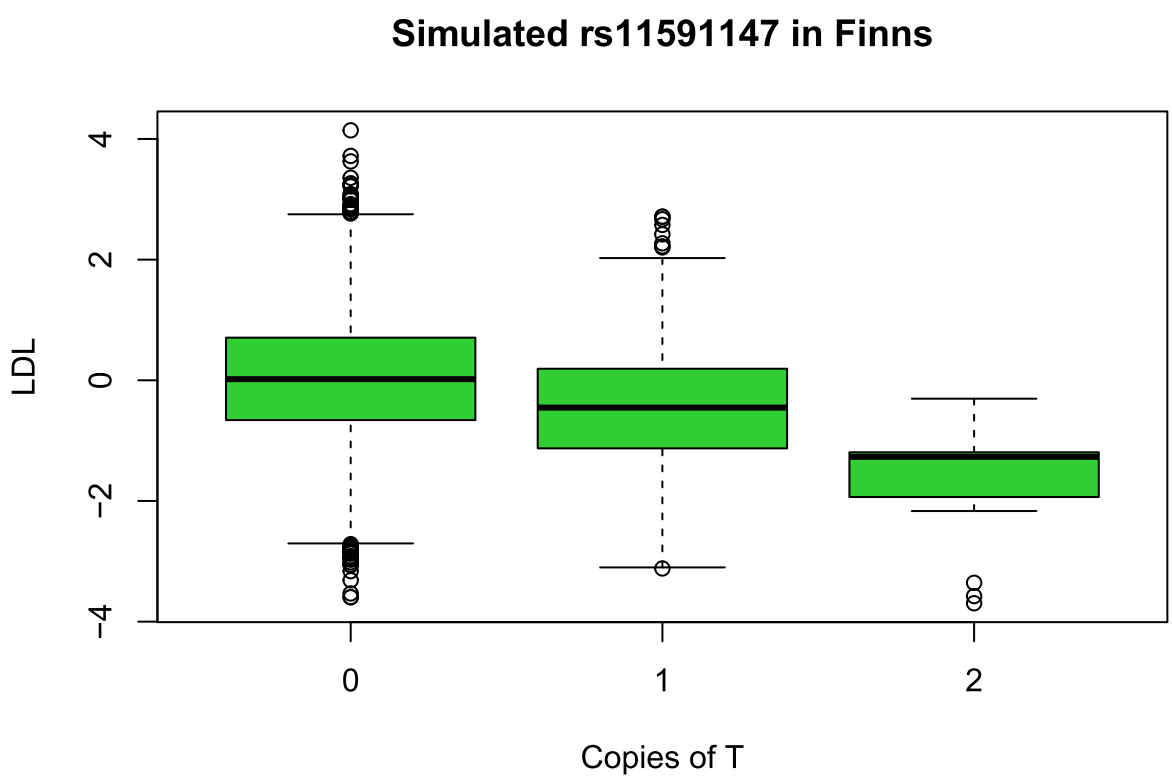
```
n = 10000
f = 0.04
mu = c(0.02, -0.40, -2.00) #mean of each genotype
sigma = c(1, 1, 1) #SD of each genotype

x = rbinom(n, size = 2, p = f) #genotypes for 'n' individuals assuming HWE
table(x)/n #(always check that simulated data is ok before starting to work with it!)
```

```
## x
##      0      1      2
## 0.9244 0.0737 0.0019
```

```
y = rep(NA,n) #make empty phenotype vector
for(ii in 0:2){ #go through each genotype group: 0, 1, 2.
  y[x == ii] = rnorm(sum(x == ii), mu[1+ii], sigma[1+ii]) } #generate trait for group ii

boxplot(y ~ x, main = "Simulated rs11591147 in Finns", ylab = "LDL",
        xlab = "Copies of T", col = "limegreen")
```



We see that the phenotype varies with genotype in such a way that each additional copy of allele T decreases the level of LDL.

Additive model The simplest way to analyze these data statistically is to use an **additive model**, that makes the assumption that the means of the groups depend additively on the number of allele 1 in the genotype, and that the SDs of the genotype groups are constant. Thus, we fit a linear model $y = \mu + x\beta + \varepsilon$, where y is the phenotype, x is the genotype (0,1 or 2) and parameters to be estimated are

- μ, the mean of genotype 0 and
- β, the effect of each copy of allele 1 on the mean phenotype.

The error terms ε are assumed to have an identical Normal distribution $N(0, \sigma^2)$ where σ^2 is not known and will be estimated from the data. Let's fit this linear model in R using `lm()`.

```
lm.fit = lm(y ~ x)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.6182 -0.6792 -0.0009  0.6861  4.1211
##
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.02150    0.01031   2.087   0.0369 *
## x           -0.52820    0.03614 -14.614   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9918 on 9998 degrees of freedom
## Multiple R-squared:  0.02091,    Adjusted R-squared:  0.02082
## F-statistic: 213.6 on 1 and 9998 DF,  p-value: < 2.2e-16
```

The `summary(lm.fit)` command produced

- parameter estimates (or Coefficients) $\widehat{\mu}$ and $\widehat{\beta}$,
- their standard errors (SE) (estimates for square root of the sampling variance of the parameter estimates),
- t-statistic (estimate/SE) and
- P-value under the null hypothesis that the parameter is 0 and errors are uncorrelated and have distribution $N(0, \sigma^2)$.

Under the assumptions of linear model, sampling distribution of t-statistic is t-distribution and hence q% confidence intervals are determined as $\widehat{\beta} \pm a \times \text{SE}$, where a is the q/2% quantile of t-distribution with $n-2$ degrees of freedom. When σ^2 is known, the t-distribution is replaced by a Gaussian, and same is approximately true when n becomes large, even if estimate $\widehat{\sigma^2}$ is used in computing SE. In these cases, we often talk about z-statistic instead of t-statistic.

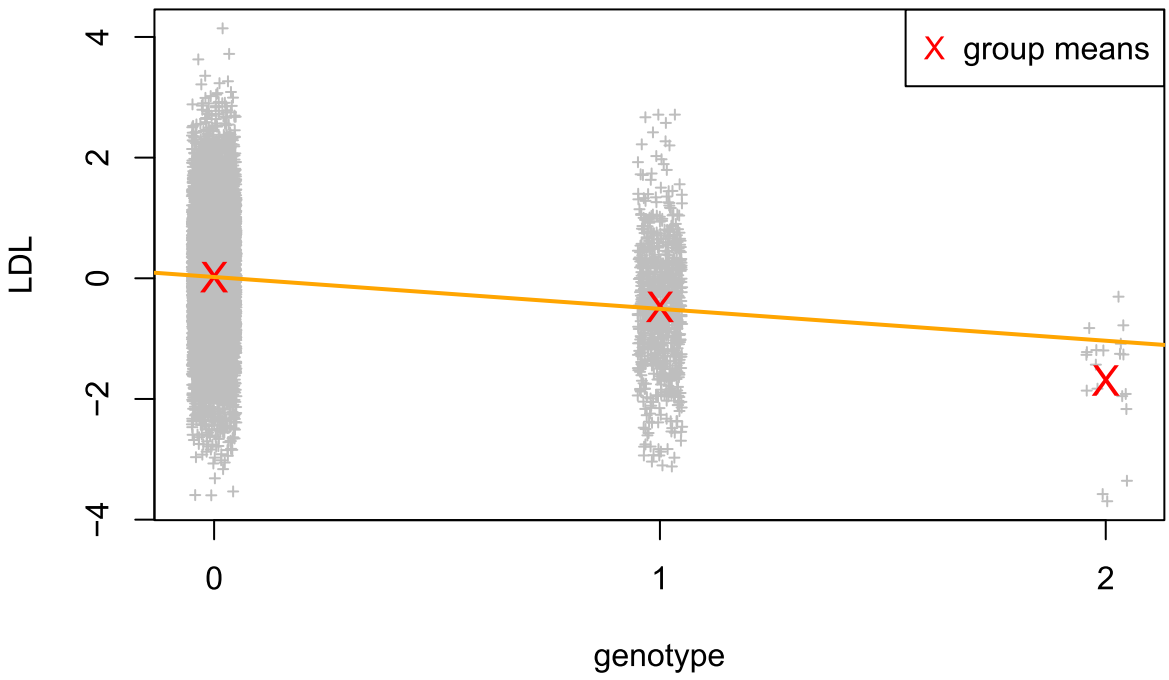
The last paragraph in the output tells about the full model fit. We can measure how much variation in y is left unexplained by the model by computing **residual sum of squares** (RSS):

$$RSS = \sum_{i=1}^{n} \left(y_i - \widehat{\mu} - x_i \widehat{\beta} \right)^2$$

R^2 is the proportion of variance explained by the linear model, i.e., $R^2 = 1 - \frac{RSS}{n-1} / \widehat{\text{Var}}(y)$. Adjusted version penalizes for additional predictors and is defined here as $R^2_{adj} = 1 - \frac{RSS}{n-2} / \widehat{\text{Var}}(y)$. Note that if there is only the intercept parameter μ in the model, then $R^2 = R^2_{adj} = 0$, and if the model explains data perfectly ($RSS = 0$), then $R^2 = R^2_{adj} = 1$. In other cases R^2 values are between 0 and 1 and larger values mean more variance explained by the model.

However, R^2 should not be the only value used to judge how suitable the model is for the data. One should also plot the data and the model fit in different ways to assess this question. For this simple linear model, a scatter plot and a regression line is a good way to assess whether there seem to be deviations from the additivity assumption. Additionally, the differences in residual variation between the groups could indicate interaction effecs between the genetic variant and some other genetic or environmental variable.

```
plot( x + runif(n, -0.05, 0.05), y, xlab = "genotype", ylab = "LDL", xaxt = "n",
      pch = 3, cex = 0.5, col = "gray")
#runif() adds some jitter to x so that all points are not on top of each other
axis(1, at = 0:2, labels = 0:2)
points(0:2, c(mean(y[x==0]), mean(y[x==1]), mean(y[x==2])), col = "red", pch = "X", cex = 1.3)
abline(lm.fit, col = "orange", lwd = 2)
legend("topright", pch = "X", legend ="group means", col = "red")
```



Conclusion: We see a statistically highly significant association between the genotype and phenotype where a copy of allele T decreases LDL levels by 0.45 units. This variant explains about 1.5% of the variation in LDL-cholesterol levels. We also see that individuals homozygous for allele T (genotype 2) have on average lower levels of LDL than the model predicts, which indicates a deviation from the additivity assumption. Let's next fit a full 2-parameter model to quantify this deviation.

Full model Let's add a new parameter γ to the model that describes the residual effect for group 2 after the additive effect is accounted for. The model is $y = \mu + x\beta + z\gamma + \varepsilon$, where z is indicator of genotype 2, i.e., $z_i = 1$ if individual i has genotype 2 and otherwise $z_i = 0$. This is the full model where each genotype group has its own mean (genotype 0: μ; genotype 1: $\mu + \beta$ and genotype 2: $\mu + 2\beta + \gamma$).

```
z = as.numeric( x == 2 ) #z is indicator for genotype group 2
lm.full = lm( y ~ x + z )
summary(lm.full)
```

```
##
## Call:
## lm(formula = y ~ x + z)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.6169 -0.6780  0.0000  0.6854  4.1224
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)  0.02015    0.01031   1.954  0.05070 .
## x           -0.49290    0.03795 -12.990  < 2e-16 ***
## z           -0.72770    0.23910  -3.044  0.00234 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9914 on 9997 degrees of freedom
## Multiple R-squared:  0.02182,    Adjusted R-squared:  0.02162
## F-statistic: 111.5 on 2 and 9997 DF,  p-value: < 2.2e-16
```

It seems that also the new variable is useful (large effect and small P-value). Now the interpretation of coefficients is that genotype 1 has avg. phenotype of -0.40 and genotype 2 has avg. phenotype -0.398*2-1.206 = -2.00.

Note also that the full model above gives the same model fit and is simply a different parameterization of the linear regression that treats the genotype as a factor with three levels.

```
lm.full2 = lm( y ~ as.factor(x) )
summary(lm.full2)
```

```
##
## Call:
## lm(formula = y ~ as.factor(x))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.6169 -0.6780  0.0000  0.6854  4.1224
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.02015    0.01031   1.954   0.0507 .
## as.factor(x)1 -0.49290    0.03795 -12.990  < 2e-16 ***
## as.factor(x)2 -1.71350    0.22767  -7.526 5.67e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9914 on 9997 degrees of freedom
## Multiple R-squared:  0.02182,    Adjusted R-squared:  0.02162
## F-statistic: 111.5 on 2 and 9997 DF,  p-value: < 2.2e-16
```

We should use this latter parameterisation of the full model if we are interested in SE of the genetic effect of genotype 2, because that information is not given by the first parameterisation of the full model. If we instead are interested to quantify how much the data show deviation from the additivity then the first parameterisation is the most suitable. All these models make the same assumption that SD is constant across the genotype groups.

Often quantitative traits is analyzed using the additive model, i.e., a linear regression model with a single parameter for genetic effect. The full model is typically only used for a small group of interesting variants identified by the additive model to check them manually for possible deviations from the additive effects. The main reason for this is that the additive model is usually almost as powerful to find associations as the full model even when deviations from additivity are present in the data, since typically one of the genotype groups is much smaller than the other two and hence does not affect much the statistical model fit. Additionally, our current understanding is that most associations follow well the additive model and the additive model has more power than the full model, for these cases. (But note that our current understanding may be biased in favor of the additive model since we do not usually look very carefully for non-additive effects.)

Let's continue explore our mouse data

The mouse data set can be loaded using the following command:

```
mouse <- readRDS(url("https://github.com/psoerensen/bgcourse/raw/main/data/mouseqtl.rds"))
```

Question 1: How many observations and which variables do we have in the data set? To get a fast overview of the data set you are working with you can use the `str` function:

Answer:

Question 2: How many observations do the two marker variables have in each genotype class? Use the `table` function to explore the two marker variables:

Answer:

Question 2: What are the genotype and allele frequencies for M227? Include the allele and genotype frequencies for M227 in the following table:

Variable	M227
f_{AA}	
f_{AB}	
f_{BB}	
f_{A}	
f_{B}	

Question 3: Does the marker variable M227 potentially influence body weight and glucose? Use the `boxplot` function to visualize the potential effect of the marker variable M227 on the two traits:

Answer:

To best answer these question we can fit a linear model that also include the effect of the marker variable in addition to sire, dam, sex, and reps. This can be done using the `lm` function:

```
fit <- lm(BW~sire+dam+sex+reps+M227, data=mouse)
```

To test the effect of the variables in the model use the `anova` function on the `fit` object from the `lm` function:

```
anova(fit)
```

```
## Analysis of Variance Table
##
## Response: BW
##             Df  Sum Sq Mean Sq   F value    Pr(>F)
## sire        10  1516.8   151.7    9.9249 4.494e-16 ***
## dam         44  2083.4    47.3    3.0982 1.415e-10 ***
## sex          1 20553.6 20553.6 1344.8773 < 2.2e-16 ***
## reps         2  1738.5   869.2   56.8770 < 2.2e-16 ***
## M227         2  1420.8   710.4   46.4848 < 2.2e-16 ***
## Residuals 1114 17025.2    15.3
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Question 4: Based on the linear model results do marker variable M227 influence body weight?

Answer:

The additive effect is modeled by a variable, `add`, with levels that is coded as -1, 0, and 1 (corresponding to -a, 0, a) for the genotypes AA, AB, and BB. The following lines of R code create a the `add` variable, fit the linear model and test the effects:

```
alleles <- c(-1,0,1)
names(alleles) <- c("AA","AB","BB")
mouse$add <- alleles[mouse$M227]
fit <- lm(BW~sire+dam+sex+reps+add, data=mouse)
summary(fit)
```

```
##
## Call:
## lm(formula = BW ~ sire + dam + sex + reps + add, data = mouse)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.8059  -2.3690  -0.0276   2.2767  19.1559
##
## Coefficients: (10 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 35.40587    0.86548  40.909  < 2e-16 ***
## sire28      -0.25065    1.15700  -0.217  0.82853
## sire34      -1.20989    1.20858  -1.001  0.31701
## sire40      -2.51843    1.29093  -1.951  0.05132 .
## sire51      -1.53023    1.16701  -1.311  0.19005
## sire63      -0.68511    1.16245  -0.589  0.55573
## sire69      -0.18677    1.28710  -0.145  0.88465
## sire72      -0.70458    1.17236  -0.601  0.54797
## sire78      -0.59245    1.15907  -0.511  0.60935
```

```
## sire79      -0.37136    1.16245  -0.319  0.74944
## sire85      -2.79819    1.15685  -2.419  0.01573 *
## dam27        2.88501    1.27251   2.267  0.02357 *
## dam29       -2.32129    1.15454  -2.011  0.04461 *
## dam30        0.88628    1.14134   0.777  0.43760
## dam31        0.86749    1.14644   0.757  0.44940
## dam32        2.13550    1.14490   1.865  0.06241 .
## dam33             NA         NA      NA       NA
## dam35       -1.26958    1.13154  -1.122  0.26211
## dam36       -1.98989    1.14191  -1.743  0.08168 .
## dam37       -0.01142    1.26851  -0.009  0.99282
## dam38       -0.91996    1.26827  -0.725  0.46838
## dam39             NA         NA      NA       NA
## dam41       -0.20300    1.28396  -0.158  0.87440
## dam42        1.58519    1.38957   1.141  0.25421
## dam43        0.39593    1.28132   0.309  0.75738
## dam44        0.58490    1.27110   0.460  0.64549
## dam45             NA         NA      NA       NA
## dam46       -1.22983    1.38421  -0.888  0.37448
## dam47       -2.24028    1.26569  -1.770  0.07700 .
## dam48       -2.20983    1.38421  -1.596  0.11067
## dam49       -0.22258    1.27688  -0.174  0.86165
## dam50             NA         NA      NA       NA
## dam52       -1.39191    1.27660  -1.090  0.27581
## dam53        3.14093    1.32917   2.363  0.01830 *
## dam54       -1.35275    1.27659  -1.060  0.28953
## dam55        0.15526    1.14756   0.135  0.89241
## dam56       -2.06169    1.27673  -1.615  0.10663
## dam57             NA         NA      NA       NA
## dam58       -3.59705    1.13507  -3.169  0.00157 **
## dam59       -0.80100    1.16906  -0.685  0.49339
## dam60        1.26519    1.13216   1.118  0.26402
## dam61       -0.43825    1.17477  -0.373  0.70918
## dam62             NA         NA      NA       NA
## dam64        2.44663    1.15465   2.119  0.03432 *
## dam65       -1.23471    1.15474  -1.069  0.28519
## dam66       -1.37101    1.16719  -1.175  0.24039
## dam67       -0.47524    1.17951  -0.403  0.68709
## dam68             NA         NA      NA       NA
## dam70        1.07898    1.29456   0.833  0.40475
## dam71        3.65606    1.20670   3.030  0.00250 **
## dam73        0.67071    1.14211   0.587  0.55715
## dam74       -1.46595    1.12982  -1.298  0.19473
## dam75        1.25837    1.14134   1.103  0.27046
## dam76        1.78894    1.13016   1.583  0.11372
## dam77             NA         NA      NA       NA
## dam80       -1.40069    1.12910  -1.241  0.21504
## dam81        0.63161    1.14208   0.553  0.58035
## dam82       -1.38070    1.12919  -1.223  0.22169
## dam83       -0.42349    1.12955  -0.375  0.70779
## dam84             NA         NA      NA       NA
## dam86       -0.07970    1.31652  -0.061  0.95174
## dam87        2.06670    1.19674   1.727  0.08445 .
## dam88        0.80940    1.18589   0.683  0.49505
```

```
## dam89        0.14869    1.19829   0.124  0.90127
## dam90             NA         NA      NA       NA
## sexMale      8.43092    0.22884  36.843  < 2e-16 ***
## reps2       -0.37618    0.27188  -1.384  0.16675
## reps3        2.67459    0.29815   8.971  < 2e-16 ***
## add          1.89345    0.19868   9.530  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.911 on 1115 degrees of freedom
##   (3 observations deleted due to missingness)
## Multiple R-squared:  0.6153, Adjusted R-squared:  0.5953
## F-statistic: 30.75 on 58 and 1115 DF,  p-value: < 2.2e-16
```

```
confint(fit,parm="add")
```

```
##        2.5 %   97.5 %
## add 1.503616 2.283276
```

The regression coefficient for the variable `add` is 1.89. The coefficient corresponds to the allele substitution effect (α). Previously we have estimated allele and genotype frequencies for M227. The following table summarizes all genotypic values, all breeding values and the dominance deviations.

Genotyp A_iA_j	Genotypic value GV_{ij}	Breeding Value BV_{ij}	Dominance Deviation D_{ij}
A_1A_1	a	$2q\alpha$	$-2q^2d$
A_1A_2	d	$(q-p)\alpha$	$2pqd$
A_2A_2	$-a$	$-2p\alpha$	$-2p^2d$

Question 5: What are the breeding values for body weight based on the M227 locus?

Answer:

The formula below shows that σ^2_G consists of two components. The first component σ^2_A is called the **genetic additive variance** and the second component σ^2_D is termed **dominance variance**. Here σ^2_A corresponds to the variance of the breeding values. The variance of breeding values is also called the additive genetic variance, because as we have already seen the breeding values are additive in the number of favorable alleles. In populations where there is no additive genetic variance, individuals all have the same breeding value. Therefore, they will produce offspring with the same expected advantage (zero), and selection cannot generate any improvement over generations. Because σ^2_D corresponds to the variance of the dominance deviation effects it is called dominance variance.

$$\begin{aligned}\sigma^2_G &= 2pq\alpha^2 + (2pqd)^2 \\ &= \sigma^2_A + \sigma^2_D\end{aligned}$$

Question 6: What is the additive genetic variance for body weight based on M227 locus?

Answer:

Now we will fit the full genetic model to locus M227 including both additive and dominance effects. The additive effect is modeled as previously shown by a variable `add` that is coded as -1, 0, and 1 (corresponding to -a, 0, a) for the genotypes AA, AB, and BB. The dominance effect is modeled by a variable `dom` that is coded as 0, 1, and 0 (corresponding to 0,d,0) for the genotypes AA, AB, and BB. The corresponding R code is shown below:

```
alleles <- c(-1,0,1)
names(alleles) <- c("AA","AB","BB")
mouse$add <- alleles[mouse$M227]
mouse$dom <- as.numeric(mouse$add==1)
fit <- lm(BW~sire+dam+sex+reps+add+dom, data=mouse)
summary(fit)
```

```
##
## Call:
## lm(formula = BW ~ sire + dam + sex + reps + add + dom, data = mouse)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.9839  -2.3945   0.0148   2.3158  19.4394
##
## Coefficients: (10 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 35.56973    0.87257  40.764  < 2e-16 ***
## sire28      -0.15502    1.15837  -0.134  0.89357
## sire34      -1.20627    1.20801  -0.999  0.31823
## sire40      -2.51132    1.29033  -1.946  0.05187 .
## sire51      -1.57196    1.16682  -1.347  0.17818
## sire63      -0.64376    1.16225  -0.554  0.57977
## sire69      -0.18947    1.28649  -0.147  0.88294
## sire72      -0.66402    1.17215  -0.567  0.57117
## sire78      -0.57003    1.15863  -0.492  0.62282
## sire79      -0.36138    1.16191  -0.311  0.75584
## sire85      -2.70261    1.15822  -2.333  0.01980 *
## dam27        2.70533    1.27805   2.117  0.03450 *
## dam29       -2.29161    1.15418  -1.985  0.04733 *
## dam30        0.86666    1.14088   0.760  0.44763
## dam31        0.81303    1.14653   0.709  0.47839
## dam32        2.05909    1.14560   1.797  0.07254 .
## dam33             NA         NA      NA       NA
## dam35       -1.39066    1.13414  -1.226  0.22039
## dam36       -1.96089    1.14155  -1.718  0.08612 .
## dam37       -0.10649    1.26964  -0.084  0.93317
## dam38       -1.02966    1.26997  -0.811  0.41767
## dam39             NA         NA      NA       NA
## dam41       -0.16125    1.28368  -0.126  0.90006
## dam42        1.68688    1.39072   1.213  0.22540
```

```
## dam43        0.47784    1.28198   0.373  0.70941
## dam44        0.58816    1.27049   0.463  0.64350
## dam45             NA         NA      NA       NA
## dam46       -1.26492    1.38377  -0.914  0.36086
## dam47       -2.25783    1.26514  -1.785  0.07459 .
## dam48       -2.19786    1.38358  -1.589  0.11245
## dam49       -0.16798    1.27684  -0.132  0.89535
## dam50             NA         NA      NA       NA
## dam52       -1.38567    1.27600  -1.086  0.27774
## dam53        3.14725    1.32855   2.369  0.01801 *
## dam54       -1.36294    1.27601  -1.068  0.28569
## dam55        0.24780    1.14882   0.216  0.82926
## dam56       -2.03902    1.27622  -1.598  0.11039
## dam57             NA         NA      NA       NA
## dam58       -3.75045    1.13955  -3.291  0.00103 **
## dam59       -0.80097    1.16851  -0.685  0.49320
## dam60        1.15561    1.13419   1.019  0.30848
## dam61       -0.58849    1.17887  -0.499  0.61774
## dam62             NA         NA      NA       NA
## dam64        2.43267    1.15414   2.108  0.03527 *
## dam65       -1.22671    1.15421  -1.063  0.28810
## dam66       -1.42396    1.16722  -1.220  0.22274
## dam67       -0.47535    1.17895  -0.403  0.68688
## dam68             NA         NA      NA       NA
## dam70        0.93928    1.29760   0.724  0.46930
## dam71        3.50353    1.21079   2.894  0.00388 **
## dam73        0.60463    1.14250   0.529  0.59676
## dam74       -1.47393    1.12930  -1.305  0.19211
## dam75        1.24478    1.14084   1.091  0.27546
## dam76        1.72917    1.13039   1.530  0.12637
## dam77             NA         NA      NA       NA
## dam80       -1.42111    1.12865  -1.259  0.20825
## dam81        0.64809    1.14159   0.568  0.57035
## dam82       -1.40261    1.12875  -1.243  0.21427
## dam83       -0.40456    1.12909  -0.358  0.72018
## dam84             NA         NA      NA       NA
## dam86        0.01084    1.31741   0.008  0.99344
## dam87        2.01013    1.19682   1.680  0.09332 .
## dam88        0.72611    1.18675   0.612  0.54076
## dam89        0.24076    1.19943   0.201  0.84095
## dam90             NA         NA      NA       NA
## sexMale      8.43211    0.22873  36.865  < 2e-16 ***
## reps2       -0.37123    0.27177  -1.366  0.17223
## reps3        2.67819    0.29802   8.987  < 2e-16 ***
## add          2.15633    0.27014   7.982 3.55e-15 ***
## dom         -0.75295    0.52454  -1.435  0.15145
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.909 on 1114 degrees of freedom
##   (3 observations deleted due to missingness)
## Multiple R-squared:  0.616,  Adjusted R-squared:  0.5957
## F-statistic: 30.29 on 59 and 1114 DF,  p-value: < 2.2e-16
```

```
confint(fit,parm="add")
```

```
##        2.5 %   97.5 %
## add 1.626286 2.686375
```

```
confint(fit,parm="dom")
```

```
##         2.5 %   97.5 %
## dom -1.782151 0.276259
```

The results from the linear model analysis suggest that only the additive genetic effect, `add`, is significantly different from 0. However in the following exercise we will be using the both the additive effect (`add`) and dominance effect (`dom`) estimated for locus M227, and the allele frequency of the positive allele (B) to explore the effect of changes in allele frequency.

Task: Use the following shinyapp, https://shiny.cnsgenomics.com/Falconer2/, to illustrate the relationship between p, a,d and α.