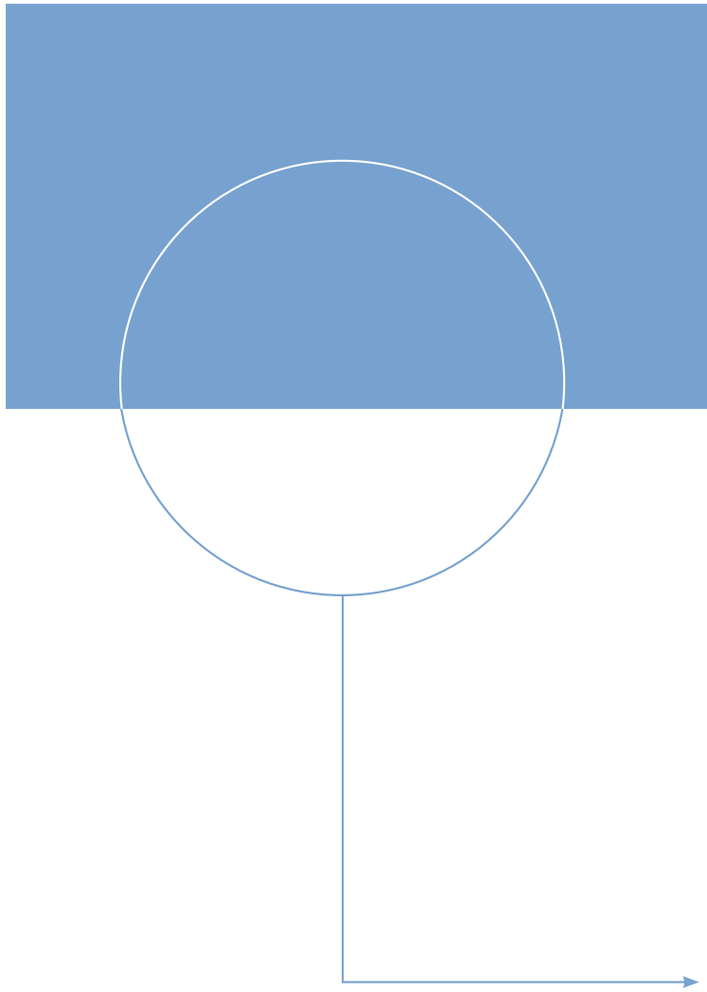


Master's thesis

2019

Kristine Lund Mathisen

NTNU
Norwegian University of
Science and Technology
Faculty of Information Technology and Electrical
Engineering
Department of Mathematical Sciences

Kristine Lund Mathisen

Identifying expression quantitative trait loci in patients with inflammatory bowel disease

Statistical models relating gene expression and single nucleotide polymorphism data

June 2019

NTNU
Norwegian University of
Science and Technology

NTNU
Norwegian University of
Science and Technology

NTNU
Norwegian University of
Science and Technology

Identifying expression quantitative trait loci in patients with inflammatory bowel disease

Statistical models relating gene expression and single nucleotide polymorphism data

Kristine Lund Mathisen

Master of Science in Physics and Mathematics
Submission date: June 2019
Supervisor: Mette Langaas
Co-supervisor: Atle van Beelen Granlund

Norwegian University of Science and Technology
Department of Mathematical Sciences

Summary

This thesis is a pilot study, conducted in order to investigate the possible relationships between genotyped SNPs and the gene expression of a gene. We will use the gene ITGAL as a prototype gene, and analyse the samples from individuals with and without chronic bowel diseases. The selected SNPs are located within the ITGAL gene. The gene expression of ITGAL is measured and preprocessed with two different technologies, microarray and RNASeq. From the microarray data we have 62 samples, and from the RNASeq data we have 75 samples. For all samples, we have the ITGAL gene expression, disease status and SNP status for the selected SNPs. For each sample from patients with chronic bowel disease, it is specified whether the gene expression of ITGAL is measured in inflamed or uninflamed tissue. The data set is compiled by the IBD research group at the Department of Clinical and Molecular Medicine, NTNU. The methods considered in this project are multiple linear models and generalized linear models, where the response is the gene expression of ITGAL, and the covariates are disease status and the interaction between disease status and SNP status. We are interested in how the gene expression of ITGAL is affected when the disease status and SNP status change. There are very few significant results for the microarray data when we correct for multiple testing. For the RNASeq data, no covariates with the interaction term is significant even if we do not correct for multiple testing. We have also compared the two groups of inflamed and uninflamed tissues, regardless of disease. In addition, we have looked at the genotyped SNPs within a distance from ITGAL, and the correlation between the SNPs with significant results. The next step is to expand the pilot study to other genes.

Preface

This project constitutes the course TMA4900 - Industrial Mathematics, Master's Thesis for the Industrial Mathematics program at NTNU. The topic is investigation of the relationship between gene expressions and genotyped SNPs for persons with and without different bowel diseases. I would like to thank my supervisor Mette Langaas at the Department of Mathematical Sciences for excellent guidance in this process, and Atle van Beelen Granlund, researcher at the Department of Clinical and Molecular Medicine for making the data set available to me and for support in understanding the genetic part of the project. I would also like to thank Per Kristian Hove, Oddgeir Lingaas Holmen and Sandor Zeestraten for helping me setting up the connection to the HUNT Cloud and technical support, and my parents for proofreading the project.

Table of Contents

Chapter 1

Introduction

In this project we perform statistical analysis of data from persons with inflammable bowel disease.

1.1 Inflammable bowel disease

Inflammable bowel disease (IBD) is a term to describe multiple chronic bowel diseases, where the two most common are Crohn's disease (CD) and ulcerative colitis (UC). The difference between these diseases is that ulcerative colitis is just in the colon, while Crohn's disease can occur in any part of the gastrointestinal tract. This is illustrated in Figure 1.1. In many cases, it may be difficult to tell the difference between these diseases (Aabakken, 2016).

Crohn's disease may cause inflammations through the whole digestive system, but is most common in the small and large intestine. The inflammation can go straight through the intestinal walls and create false openings (fistulas), which may lead to infections, and narrow areas, which may cause twisting of the stomach (gastric volvulus) (Aabakken, 2018).

Ulcerative colitis usually starts in the lower part of the large intestine and rectum, but might spread through the colon and to the lower part of the small intestine. The inflammation is usually limited to the mucosa. Areas with pus arises in the colon. These areas are called crypt abscesses, and when these breaks, wounds appear where tissue fluid and blood seeps through (Aabakken and Halstensen, 2018).

We will in this thesis look at measurements from patients with Crohn's disease in

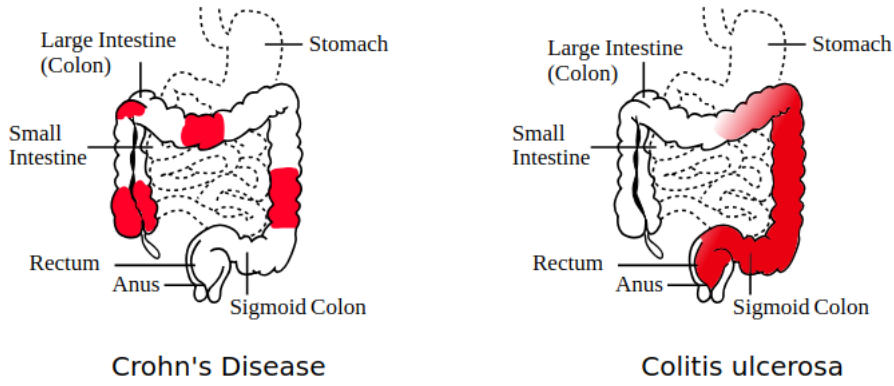


Figure 1.1: This figure shows the difference between Crohn's disease and ulcerative colitis. The red parts represent where the colon is inflamed.
Source: https://commons.wikimedia.org/wiki/File:Crohn%27s_Disease_vs_Colitis_ulcerosa.svg, licensed under CC BY-SA 3.0

inflamed tissue (CDA), patients with Crohn's disease in uninflamed tissue (CDU), patients with ulcerative colitis in inflamed tissue (UCA) and patients with ulcerative colitis in uninflamed tissue (UCU).

1.2 SNP-data

DNA is a self-replicating material which is carrier of genetic information. It describes how the organism will look and function, and these descriptions are inherited from one generation to the next (Martinsen, 2019). SNP (single-nucleotide polymorphism) is a position-based one-base-variation in the DNA. We will study bi-allelic SNPs, which means that there are two base variants. This is illustrated in Figure 1.2. To be classified as a SNP, the least frequent variant (the minor allele) must exist in at least one percent of the population (if not, it is called a SNV). In a genome-wide association study (GWAS) the objective is to search for associations between a phenotype (for example a disease) and a SNP. In this project we will focus on a selection of SNPs located inside and within a distance from the gene ITGAL. As a running example, we will use the SNP with identification number rs11150589, which is located in chromosome 16, position 30 482 494. According to Jostins et al. (2012) this SNP is associated with ulcerative colitis.

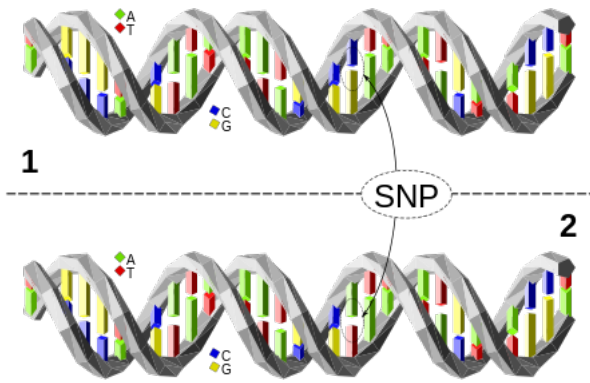


Figure 1.2: SNP is a position-based one-base-variation in the DNA. As illustrated here, the upper base-pair is C/G, while the lower base-pair is A/T.
Source: https://commons.wikimedia.org/wiki/File:Dna-SNP.svg, licensed under CC BY 4.0

1.3 Linkage disequilibrium and genotype correlation

Linkage disequilibrium (LD) is a non-random association of alleles, and is defined as ”the difference between the observed frequency of a particular combination of alleles at two loci and the frequency expected for random association” (Robinson, 2004, pp. 1586). It is a measure for correlation between the SNPs. A haplotype is a combination of alleles in nearby loci in a DNA molecule. Ideally each SNP contribute to two alleles as in Table 1.1.

SNP 1	SNP 2	Haplotype A	Haplotype B
0	0	0-0	0-0
0	1	0-0	0-1
0	2	0-1	0-1
1	0	1-0	0-0
1	1	1-0 or 1-1	0-1 or 0-0
1	2	0-1	1-1
2	0	1-0	1-0
2	1	1-0	1-1
2	2	1-1	1-1

Table 1.1: Table showing connection between SNP status and haplotype.

As we do not have the haplotype data, we do not know if the SNP status 1 at SNP 1 and 1 at SNP 2 is a representation of haplotype 0-1 and 1-0 or 0-0 and 1-1.

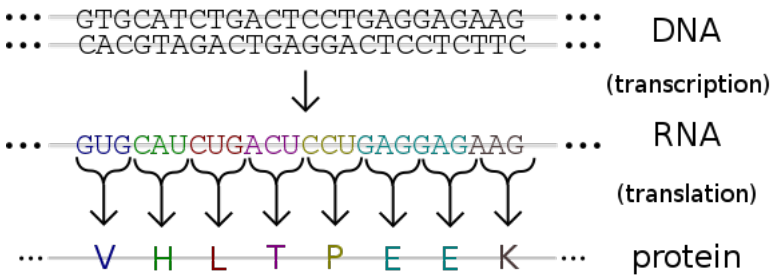


Figure 1.3: Illustration of the transcription from DNA to RNA, and the translation of RNA to protein.
Source: https://commons.wikimedia.org/wiki/File:Genetic_code.svg, licensed under CC BY-SA 3.0

Due to this, we prefer to calculate the genotype correlation coefficient, which is a measure of composite LD. The formula is $\text{Corr}(Y_1, Y_2)$, where Y_1 and Y_2 are the SNP statuses of two SNPs.

1.4 Gene expression

Gene expression is the process where the information in a gene's DNA sequence is transcribed and translated to the structures and functions of the cell. Usually the end products are proteins (Alberts et al., 2014, pp. 228). Gene expression includes the transcription from DNA to RNA and (for coded genes) translation of RNA to protein. This is illustrated in Figure 1.3. In this project we will look at RNA measurements of ITGAL, or Integrin alpha L, which is a gene involved in a variety of immune phenomena. We will analyse gene expression data from two technologies: microarray data are from the Illumina human HT-12 expression BeadChips, and RNASeq data from the sequencing (75 cycles single end reads) Illumina HiSeq4000 instrument.

1.5 eQTL

The expression quantitative trait loci (eQTL) is according to Nica and Dermitzakis (2012) "the discovery of genetic variants that explain variation in gene expression levels". An eQTL is a locus which explains a fraction of the genetic variance of a gene expression phenotype. In our case, the phenotype is the gene expression of the ITGAL gene. In the statistical model for discovering eQTLs also disease status and SNP status will be included. Regulatory variants are called *cis* or *trans* acting,

and depends on the distance from the transcriptional start site (TSS). The TSS is the location where transcription starts. The *cis* acting is within 1 megabase (Mb) from TSS, on both sides, while the *trans* acting is within 5 Mb. The term 1 megabase means that we are looking at the nearest 1 000 000 base pairs. However, there are other ways to use the definitions *cis* and *trans*, where the terms describe whether the regulation works directly or through other eQTLs. In this thesis, we will use the terms *local* and *distant cis-acting* to describe the distance from TSS. As discussed in Nica and Dermitzakis (2012), it might be difficult to detect differences between causal and reactive expression changes. The aim of eQTL is to find the association between SNPs and gene expression. In Chapter 3 we will look at SNPs located inside and within a distance from the gene ITGAL and relate these to the gene expression of ITGAL.

1.6 HUNT Cloud

HUNT Cloud is a digital infrastructure where researches can store, access and analyse sensitive data in controlled environments. This includes research unrelated to HUNT (Helseundersøkelsen i Nord-Trøndelag), as in this project. The data set analysed in this thesis contains data on genotyped (and imputed) SNPs for the participants in the study, and this makes it possible to identify them. For their protection, it is necessary to perform the analysis in a safe environment. We have used command line tools in Linux, and run R and Rstudio using X2Go. To access the data, two factor authentication was necessary. More information on HUNT can be found at `https://www.ntnu.no/hunt`.

1.7 Thesis outline

The structure of the thesis is as follows: In Chapter 2 we present the theory of the statistical models used in this project, in order to understand the results. We also present the structure of the data set. In Chapter 3 we look at the model set-up and the connections to genetic models. The results from a running example are shown. In Chapter 4 we look at the process for performing the calculations. The data set is presented properly, the theory from Chapter 2 is applied on the data set and the results are presented. We also expand our data set to investigate how this affects the results. In Chapter 5 we discuss the results and suggest directions for further work.

Chapter 2

Linear and generalized linear models

In this chapter we will present the statistical methods used to perform eQTL analysis. We will use the gene expression of the gene ITGAL and the SNP with identification number rs11150589 located within the ITGAL gene as a running example. From the microarray data we have 62 observations for this gene, and from RNASeq we have 75 observations. For each observation we have the SNP status and the gene expression represented as either a preprocessed value from the microarray technology or count data from the RNASeq technology. The microarray data will be analysed with multiple linear regression and RNASeq with a variant of generalized linear model. We also have information on disease status to include in the model.

2.1 Multiple linear regression model

This presentation is based on Fahrmeir et al. (2013, pp. 73-168).

For our eQTL analysis, we will use the multiple linear regression (MLR) model. We use the following notation:
$\mathbf{Y}$ is a $(n \times 1)$ vector of responses (random variables)
$\mathbf{X}$ is a $(n \times p)$ design matrix with rows $\mathbf{x}_i^T$ for $i = 1, ..., n$
$\boldsymbol{\beta}$ is a $(p \times 1)$ vector of regression parameters including intercept
$\boldsymbol{\varepsilon}$ is a $(n \times 1)$ vector of random errors

The classical normal linear regression model is $\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$. We define the linear

predictor as $\eta_i = \mathbf{x}_i^T \boldsymbol{\beta}$. The following is assumed:
1. $\mathrm{E}(\boldsymbol{\varepsilon}) = \mathbf{0}$
2. $\mathrm{Cov}(\boldsymbol{\varepsilon}) = E\left(\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^T\right) = \sigma^2\mathbf{I}$
3. The design matrix $\mathbf{X}$ has full rank, rank($\mathbf{X}$) = p
4. $\boldsymbol{\varepsilon} \sim N_n\left(0, \sigma^2\mathbf{I}\right)$

It is assumed that the observation pairs are measured from sampling units $\left(\mathbf{x}_i^T, \mathbf{Y}_i\right)$ and that the observation pairs are independent from each other. We assume that our sample is representative of some population of interest.

In the running example, the response vector $\mathbf{Y}$ is the microarray gene expression of ITGAL. The design matrix is $\mathbf{X} = [\mathbf{X}_e\ \mathbf{X}_g]$, where e represents the covariate disease and g represents the interaction between disease status and SNP status. This means that in our running example, the vector $\mathbf{Y}$ and design matrix $\mathbf{X}$ (with additive coding, this is explained in Section 3.1) will be

$$\mathbf{Y} = \begin{bmatrix} 7.25 \\ 7.28 \\ 6.55 \\ 6.64 \\ 6.80 \\ \vdots \\ 7.59 \end{bmatrix}$$

$$\mathbf{X} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 2 & 0 \end{bmatrix}$$

The columns 1-5 in the design matrix $\mathbf{X}$ represent the disease status. Column 1 shows N (which means samples without any bowel disease), column 2 shows "Crohns disease inflamed (CDA)", column 3 shows "Crohns disease uninflamed (CDU)", column 4 shows "ulcerative colitis inflamed (UCA)" and column 5 shows "ulcerative colitis uninflamed (UCU)". The columns 6-10 shows the interaction

between disease status and SNP status, so they show the SNP status multiplied by respectively column 1-5. The observant reader may notice that there is no intercept nor main effect of SNP in this design matrix. This is explained further in Section 3.1.2, and is used because it is easier to make contrasts with this parameterization.

2.1.1 Parameter estimation

In MLR the aim is to estimate the regression parameters $\boldsymbol{\beta}$ and σ^2. We assume a sample of independent random variables $\mathbf{Y} = Y_1, ..., Y_n$. Each Y_i has an univariate normal distribution

$$f\left(y_i; \boldsymbol{\beta}, \sigma^2\right) = \frac{1}{\sqrt{2\pi}\sigma} \cdot \exp\left(-\frac{1}{2\sigma^2}\left(y_i - \mathbf{x}_i^T\boldsymbol{\beta}\right)^2\right).$$

Assuming independent joint distribution for the sample $\mathbf{Y}$

$$f\left(y; \boldsymbol{\beta}, \sigma^2\right) = \prod_{i=1}^{n} f\left(y_i; \boldsymbol{\beta}, \sigma^2\right),$$

which gives the likelihood function

$$L\left(\boldsymbol{\beta}, \sigma^2; y\right) = f\left(y; \boldsymbol{\beta}, \sigma^2\right).$$

It is common to use the natural logarithm of the likelihood, which is called the log-likelihood function $l\left(\boldsymbol{\beta}, \sigma^2; y\right)$. This makes the calculations easier, and is allowed because the log-function is a monotone function so these likelihood functions have optimum at the same point. This gives

$$l\left(\boldsymbol{\beta}, \sigma^2; y\right) = \ln\left(L\left(\boldsymbol{\beta}, \sigma^2; y\right)\right) = \sum_{i=1}^{n} \ln f\left(y_i; \boldsymbol{\beta}, \sigma^2\right).$$

The parameters $\boldsymbol{\beta}$ and σ^2 are unknown, but we may estimate them by maximising the log-likelihood. The maximum likelihood estimates $\hat{\boldsymbol{\beta}}$ and $\hat{\sigma}^2$ are the values of $\boldsymbol{\beta}$ and σ^2 respectively so that

$$\ln L\left(\hat{\boldsymbol{\beta}}, \hat{\sigma}^2; y\right) \geq \ln L\left(\boldsymbol{\beta}, \sigma^2; y\right) \text{ for all } \boldsymbol{\beta}, \sigma^2.$$

To find the maximum likelihood estimates, we want to maximise $l\left(\boldsymbol{\beta}, \sigma^2; y\right)$. This is done by solving this set of equations with respect to the $\boldsymbol{\beta}$-part:

$$\frac{\partial l\left(\boldsymbol{\beta}, \sigma^2; y\right)}{\partial \boldsymbol{\beta}} = 0$$

which leads to the normal equations (in matrix notation)

$$\mathbf{X}^T\mathbf{X}\hat{\boldsymbol{\beta}} = \mathbf{X}^T\mathbf{Y}.$$

The estimator for $\hat{\boldsymbol{\beta}}$ is

$$\hat{\boldsymbol{\beta}} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{Y},$$

with $\mathrm{E}\left(\hat{\boldsymbol{\beta}}\right) = \boldsymbol{\beta}$ and $\mathrm{Cov}\left(\hat{\boldsymbol{\beta}}\right) = \sigma^2\left(\mathbf{X}^T\mathbf{X}\right)^{-1}$. Here $\hat{\boldsymbol{\beta}}$ is multivariate normally distributed, because it is a set of linear combinations of the random variables in $\mathbf{Y}$ which we know are independent and normally distributed.
The maximum likelihood estimator for σ^2 is found by maximising the likelihood inserted the estimate for $\hat{\boldsymbol{\beta}}$, and the formula is

$$\hat{\sigma}^2 = \frac{1}{n}\left(\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}\right)^T\left(\mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\beta}}\right) = \frac{\mathrm{SSE}}{n}.$$

However, this is a biased estimator of σ^2. It is known that

$$\frac{\hat{\sigma}^2\,(n-p)}{\sigma^2} \sim \chi^2_{n-p}$$

so to get unbiased estimator, which is preferred, we use restricted maximum likelihood (REML). This gives

$$s^2 = \frac{1}{n-p}\mathrm{SSE}.$$

In our running example, we get

$$\begin{bmatrix} \widehat{\beta}_{diseaseN} \\ \widehat{\beta}_{diseaseCDA} \\ \widehat{\beta}_{diseaseCDU} \\ \widehat{\beta}_{diseaseUCA} \\ \widehat{\beta}_{diseaseUCU} \\ \widehat{\beta}_{diseaseN:snp} \\ \widehat{\beta}_{diseaseCDU:snp} \\ \widehat{\beta}_{diseaseUCA:snp} \\ \widehat{\beta}_{diseaseUCU:snp} \end{bmatrix} = \begin{bmatrix} 4.589380 \\ 4.705833 \\ 4.591303 \\ 5.032506 \\ 4.707411 \\ 0.075255 \\ 0.087544 \\ 0.181252 \\ 0.047407 \end{bmatrix}$$

and

$$s^2 = 0.2175^2 = 0.04731.$$

2.1.2 Hypothesis testing

In single hypothesis testing, we want to test a null hypothesis against an alternative hypothesis. In the MLR we study

$$H_0 : \boldsymbol{\beta}_j = 0$$
$$H_1 : \boldsymbol{\beta}_j \neq 0.$$

In hypothesis testing there are two different types of errors that can be made. The "type I error" is when the null hypothesis H_0 is rejected, even though H_0 is true. The other type of error, "type II error", is when H_0 is not rejected, even though H_0 is false. This can be illustrated in a table:

	Not reject H_0	Reject H_0
H_0 true	Correct	Type I error
H_0 false	Type II error	Correct

To choose whether H_0 should be rejected or not, we can calculate a p-value. A p-value is defined informally as the probability of our result or a more extreme result, given that H_0 is true. The H_0 is chosen to be rejected at some significance level α if the p-value is smaller than the chosen α. The p-value is based on a test statistic, and for the MLR the test statistic for testing H_0 is

$$T_{0j} = \frac{\hat{\boldsymbol{\beta}}_j - 0}{\hat{\sigma}\sqrt{\left(\mathbf{X}^T\mathbf{X}\right)^{-1}_{[j,j]}}} \sim t_{n-p},$$

where $\left(\mathbf{X}^T\mathbf{X}\right)^{-1}_{[j,j]}$ is the j-th element of the diagonal of the $\left(\mathbf{X}^T\mathbf{X}\right)^{-1}$ matrix. The formula for our two-sided test uses the test statistic T_{0j} and the observed test statistic value t_{0j}. The t-distribution is symmetric around zero, so the p-value can be calculated as

$$p\text{-value} = P\left(T_{0j} > abs\left(t_{0j}\right)\right) + P\left(T_{0j} < -abs\left(t_{0j}\right)\right) = 2 \cdot P\left(T_{0j} > abs\left(t_{0j}\right)\right).$$

In our running example, the p-values are shown in Table 2.1. We choose the significance level to be $\alpha = 0.05$. This means that the diseaseN, diseaseCDA, diseaseCDU, diseaseUCA, diseaseUCU and diseaseUCA:snp are significant, while diseaseN:snp, diseaseCDA:snp, diseaseCDU:snp and diseaseUCU:snp are not significant.

	Estimate	Std. Error	t value	Pr(> |t|)
diseaseN	4.589380	0.101404	45.258166	4.90E-44
diseaseCDA	4.705833	0.217488	21.637173	5.67E-28
diseaseCDU	4.591303	0.126986	36.156030	4.99E-39
diseaseUCA	5.032506	0.117950	42.666573	1.03E-42
diseaseUCU	4.707411	0.081622	57.673367	1.66E-49
diseaseN:snp	0.075255	0.082796	0.908918	3.68E-01
diseaseCDU:snp	0.087544	0.108744	0.805049	4.24E-01
diseaseUCA:snp	0.181252	0.079748	2.272795	2.71E-02
diseaseUCU:snp	0.047407	0.063224	0.749829	4.57E-01

Table 2.1: Summary of the results from fitting a MLR to the running example. The number 2.16E-52 is the scientific notation of $2.16 \cdot 10^{-52}$.

2.1.3 Model assessment

There are different ways to measure how well a model fits the data. For the MLR, one way is to look at the total variability in the data, called the sums-of-squares total (SST). The SST can be decomposed into one part that is explained by the regression, sums-of-squares regression (SSR), and one part that is not explained by the regression, sums-of-squares error (SSE). Using $\bar{Y} = \frac{1}{n}\sum_{i=1}^{n} Y_i$ and $\hat{Y}_i = \mathbf{x}_i^T \hat{\beta}$, then we have

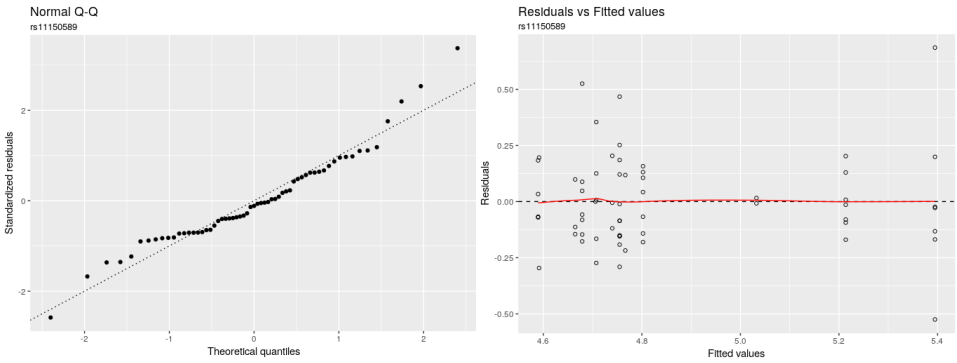
$$\text{SST} = \text{SSR} + \text{SSE}$$

where

$$\text{SST} = \sum_{i=1}^{n} \left(Y_i - \bar{Y}\right)^2$$
$$\text{SSR} = \sum_{i=1}^{n} \left(\hat{Y}_i - \bar{Y}\right)^2$$
$$\text{SSE} = \sum_{i=1}^{n} \left(Y_i - \hat{Y}_i\right)^2$$

We can use this to define the coefficient of determination R^2, which is the ratio between SSR and SST. This gives

$$R^2 = \text{SSR}/\text{SST} = 1 - \text{SSE}/\text{SST}.$$



(a) Normal QQ-plot. **(b)** Residuals vs fitted values.

Figure 2.1: Residual plots for the running example.

.
This R^2 is the squared correlation coefficient between the observed and predicted response. In our running example we have $R^2 = 0.9983$. This means that more than 99 % of the variability in the data is explained.

Another way to determine whether a model fit is good or not, is to look at a normal QQ-plot and the plot showing residuals versus fitted values. The residuals $e_i = Y_i - \hat{Y}_i$ are predicted values for the error terms ε_i. The normal QQ-plot can be used to evaluate the assumption of normality of error terms. The residuals versus fitted values-plot shows if the residuals have non-linear patterns. This can be used to test the assumption of a linear relationship between the response and the covariates. For our running example, plots are shown in Figure 2.1. The normal QQ-plot looks good because the values follow the straight line, while the plot of the residual versus fitted values looks good because there are no clear trends.

The Anderson-Darling normality test can be used to test if a sample comes from a normal distribution, so we have

$$H_0 : \text{The data follow a normal distribution}$$
$$H_1 : \text{The data do not follow a normal distribution}$$

and the test statistic, as reviewed in Das and Imon (2016), is

$$A^2 = -\left(1 + \frac{0.75}{n} + \frac{2.25}{n^2}\right)\left[\frac{\sum_{i=1}^{n}[(2i-1)\log\left(\hat{z}_{(i)}\left(1-\hat{z}_{(n-i+1)}\right)\right)]}{n} + n\right],$$

where $\hat{z}_{(i)} = \Phi\left([y_{(i)} - \hat{\mu}]/\hat{\sigma}\right)$ and $\Phi(\cdot)$ is the cumulative distribution function of an $\mathcal{N}(0,1)$ random variable. To find the distribution of A under H_0, see Table 2 in Stephens (1974). In the R package `nortest` (Gross and Ligges, 2015), a table from another publication of Stephens is used.

2.2 Generalized linear models

In a regression setting, where Y_i is the response and $\mathbf{x}_i$ are covariates, we describe the generalized linear model (GLM) consisting of three ingredients:
1) Distribution of Y_i (random component)
2) Linear predictor η_i (systematic component)
3) Link function g (link between E(Y_i) and linear predictor η_i)

2.2.1 Distribution of Y_i

For a generalized linear model, the distribution of Y_i can be written as a univariate exponential family

$$f(y_i \mid \theta_i) = \exp\left(\frac{y_i\theta_i - b(\theta_i)}{\phi} + c(y_i, \phi)\right)$$

where θ_i is the canonical parameter, ϕ is the nuisance parameter and b and c are known functions. It can be shown that $\mathrm{E}(Y_i) = b'(\theta_i) = \theta$ and $\mathrm{Var}(Y_i) = b''(\theta_i)\cdot\phi$. Further, $\mathrm{V}(\mu_i) = b''(\theta_i)$ is called the variance function. The linear model, presented in Section 2.1, is a GLM with $Y_i \sim N$.

2.2.1.1 Normal distribution

$$f\left(y_i; \mu_i, \sigma^2\right) = \frac{1}{\sqrt{2\pi}\sigma} \cdot \exp\left(-\frac{1}{2\sigma^2}(y_i - \mu_i)^2\right)$$

When $Y_i \sim N\left(\mu_i, \sigma^2\right)$, then $\theta_i = \mu_i$, $b(\theta_i) = \frac{1}{2}\theta_i^2$ and $\phi = \sigma^2$. Thus $\mathrm{E}(Y_i) = b'(\theta_i) = \mu_i$ and $\mathrm{Var}(Y_i) = b''(\theta_i) \cdot \phi = \sigma^2$.

2.2.1.2 Negative binomial distribution

A common parameterization for the probability mass function is

$$f(y_i; \mu_i, \alpha) = \frac{\Gamma\left(y_i + \frac{1}{\alpha}\right)}{\Gamma\left(\frac{1}{\alpha}\right)\Gamma(y_i + 1)}\left(\frac{\mu_i}{\mu_i + \frac{1}{\alpha}}\right)^{y_i}\left(\frac{\frac{1}{\alpha}}{\mu_i + \frac{1}{\alpha}}\right)^{\frac{1}{\alpha}}, y_i = 0, 1, 2...$$

According to Agresti (2015, pp. 247-250), α acts like "a type of dispersion parameter". The reason is that the negative binomial distribution can be seen as a mixture model for count data. Conditional on a parameter λ, let Y be Poisson(λ). The Poisson(λ) has mean and variance λ. Further, let λ have a gamma distribution. Then it can be shown that marginally Y has negative binomial distribution. If we look at Var(Y_i) $= \mu_i + \alpha\mu_i^2$ in the negative binomial distribution, when $\alpha \to 0$ then Var(Y_i) $\to \mu_i$ and the negative binomial distribution can be proven to converge to the Poisson distribution. So in this sense α acts as a "dispersion parameter". When α is fixed, then according to de Jong and Heller (2008, pp. 39), this is an univariate exponential family with

$$\theta_i = \ln\left(\frac{\mu_i}{\alpha\mu_i + 1}\right), b\left(\theta_i\right) = -\frac{1}{\alpha}\ln\left(1 - \alpha e^{\theta_i}\right), \phi = 1$$

Thus,

$$\mathrm{E}\left(Y_i\right) = b'\left(\theta_i\right) = \frac{e^{\theta_i}}{1 - \alpha e^{\theta_i}} = \frac{\mu_i / \left(1 + \alpha\mu_i\right)}{1 - \alpha \cdot \mu_i / \left(1 + \alpha\mu_i\right)} = \mu_i$$

$$\begin{aligned}\mathrm{Var}\left(Y_i\right) = b''\left(\theta_i\right) \cdot \phi &= \frac{e^{\theta_i}}{\left(1 - e^{\theta_i}\right)^2} \cdot 1 = \frac{\mu_i\left(1 + \alpha\mu_i\right)}{\left(1 + \alpha\mu_i\right)^2 - 2\alpha\mu_i\left(1 + \alpha\mu_i\right) + \alpha^2\mu_i^2} \\ &= \mu_i\left(1 + \alpha\mu_i\right).\end{aligned}$$

The variance function is $b''\left(\theta_i\right) = \mathrm{Var}\left(Y_i\right)$.

2.2.2 Linear predictor

The linear predictor is the same as for the linear model,

$$\eta_i = \mathbf{x}_i^T \boldsymbol{\beta}$$

where
$\mathbf{x}_i$ is a $(p \times 1)$ vector
$\boldsymbol{\beta}$ is a $(p \times 1)$ vector

2.2.3 Link function

The link function g is used to connect E(Y_i) = μ_i to the linear predictor η_i. We have $\eta_i = g\left(\mu_i\right)$, and assume that g is monotone and twice differentiable. The inverse function is called the response function: $h\left(\eta_i\right) = \mu_i$. The canonical link is

$$\eta_i = \theta_i \Longleftrightarrow g\left(\mu_i\right) = \theta_i.$$

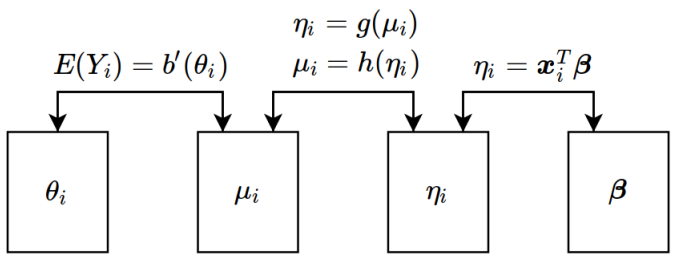


Figure 2.2: The parameters θ_i, μ_i, $\boldsymbol{\beta}$ and the linear predictor η_i are connected.

2.2.3.1 Normal distribution

For the normal distribution the canonical link is $g(\mu_i) = \mu_i$.

2.2.3.2 Negative binomial distribution

For the negative binomial distribution, the canonical link is $g(\mu_i) = \theta_i = \ln\left(\frac{\mu_i}{\mu_i+\alpha}\right)$, but we will use $g(\mu_i) = \ln(\mu_i)$.

2.2.4 Parameter estimation

In GLM, the aim is to estimate $\boldsymbol{\beta}$ and ϕ. We might as well estimate the parameters μ_i, η_i or θ_i, as they are all connected as shown in Figure 2.2.
The log-likelihood function is written on the form

$$l(\boldsymbol{\beta}) = \sum_{i=1}^{n} l_i(\boldsymbol{\beta}) = \sum_{i=1}^{n} \frac{1}{\phi}\left(y_i\theta_i - b(\theta_i)\right) + \sum_{i=1}^{n} c(y_i, \phi).$$

To estimate the unknown parameters, we set $\boldsymbol{s}(\boldsymbol{\beta}) = \frac{\partial l}{\partial \boldsymbol{\beta}} = 0$. This is called the score function, and $\boldsymbol{s}(\boldsymbol{\beta})$ is a $(p \times 1)$ vector. This gives

$$\boldsymbol{s}(\boldsymbol{\beta}) = \frac{\partial l}{\partial \boldsymbol{\beta}} = \sum_{i=1}^{n} \frac{\partial l_i}{\partial \boldsymbol{\beta}} = \sum_{i=1}^{n} s_i(\boldsymbol{\beta})$$

where

$$l_i = \frac{y_i\theta_i - b(\theta_i)}{\phi} + c(y_i, \phi).$$

To find the score function, each component is calculated by using the chain rule:

$$s_i(\boldsymbol{\beta}) = \frac{\partial l_i}{\partial \boldsymbol{\beta}} = \frac{\partial l_i}{\partial \theta_i} \cdot \frac{\partial \theta_i}{\partial \mu_i} \cdot \frac{\partial \mu_i}{\partial \eta_i} \cdot \frac{\partial \eta_i}{\partial \boldsymbol{\beta}} = \left(y_i - b'(\theta_i)\right) \cdot \frac{1}{b''(\theta_i)} \cdot h'(\eta_i) \cdot \mathbf{x}_i$$
$$= (y_i - \mu_i) \cdot \frac{1}{\text{Var}(Y_i)} \cdot h'(\eta_i) \cdot \mathbf{x}_i.$$

The score function is

$$s(\boldsymbol{\beta}) = \sum_{i=1}^{n} \frac{(y_i - \mu_i) \cdot \mathbf{x}_i \cdot h'(\eta_i)}{\text{Var}(Y_i)}.$$

2.2.4.1 Normal distribution

For the normal distribution, the log-likelihood function is

$$l(\boldsymbol{\beta}) = \left(\sqrt{2\pi}\right)^{-\frac{n}{2}} \cdot \sigma^{-\frac{n}{2}} \cdot \exp\left(-\frac{1}{2}\sum_{i=1}^{n} \frac{(y_i - \mu_i)^2}{\sigma^2}\right)$$

and the score function is

$$s(\boldsymbol{\beta}) = \sum_{i=1}^{n} \frac{\left(y_i - \mathbf{x}_i^T\boldsymbol{\beta}\right) \cdot \mathbf{x}_i}{\sigma^2}.$$

This is easy to solve. The $\hat{\boldsymbol{\beta}}$ is as shown in Section 2.1.1,

$$\hat{\boldsymbol{\beta}} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{Y}.$$

2.2.4.2 Negative binomial distribution

For the negative binomial distribution, the log-likelihood function is

$$L(\mu, \alpha,) = \sum_{i=1}^{n}\left[\log\Gamma\left(y_i + \frac{1}{\alpha}\right) - \log\Gamma\left(\frac{1}{\alpha}\right) - \log\Gamma(y_i + 1)\right]$$
$$+ \sum_{i=1}^{n}\left[y_i \log\left(\frac{\alpha\mu_i}{1 + \alpha\mu_i}\right) - \left(\frac{1}{\alpha}\right)\log(1 + \alpha\mu_i)\right].$$

Using the log-link, $\eta_i = \ln(\mu_i)$ and $\eta_i = \mathbf{x}_i^T\boldsymbol{\beta}$, the score function is

$$\boldsymbol{s}(\boldsymbol{\beta}) = \sum_{i=1}^{n} \frac{\left(y_i - \exp\left(\mathbf{x}_i^T\boldsymbol{\beta}\right)\right) \cdot \mathbf{x}_i \cdot \exp\left(\mathbf{x}_i^T\boldsymbol{\beta}\right)}{\exp\left(\mathbf{x}_i^T\boldsymbol{\beta}\right)\left(1 + \alpha\exp\left(\mathbf{x}_i^T\boldsymbol{\beta}\right)\right)} \tag{2.1}$$

Newton-Raphson and Fisher scoring
For the negative binomial distribution the equation $\boldsymbol{s}(\boldsymbol{\beta}) = 0$ does not have a closed form solution, but can be solved using Newton's method (Quarteroni et al., 2007, pp. 311):

$$\boldsymbol{\beta}^{(t+1)} = \boldsymbol{\beta}^{(t)} - H\left(\boldsymbol{\beta}^{(t)}\right)^{-1} \boldsymbol{s}\left(\boldsymbol{\beta}^{(t)}\right)$$

where $H(\boldsymbol{\beta})$ is the Hessian on the log-likelihood, also called the observed Fisher information matrix,

$$H(\boldsymbol{\beta}) = \frac{\partial l(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^T}.$$

In statistics we use the expected information matrix $F^{-1}(\hat{\boldsymbol{\beta}}^{(t)})$ instead of the observed Fisher information matrix $H\left(\hat{\boldsymbol{\beta}}^{(t)}\right)^{-1}$, which gives the Fisher scoring method:

$$\boldsymbol{\beta}^{(t+1)} = \boldsymbol{\beta}^{(t)} + F^{-1}\left(\boldsymbol{\beta}^{(t)}\right) \boldsymbol{s}\left(\boldsymbol{\beta}^{(t)}\right). \tag{2.2}$$

The expected Fisher information matrix is in general given as

$$F(\boldsymbol{\beta})_{[h,l]} = \sum_{i=1}^{n} \frac{x_{ih} x_{il} [h'(\eta_i)]^2}{\mathrm{Var}(Y_i)},$$

for an exponential family GLM model. This can be rewritten into

$$F(\boldsymbol{\beta}) = \mathbf{X}^T \mathbf{W} \mathbf{X} \tag{2.3}$$

where $\mathbf{W} = \mathrm{diag}\left(\frac{h'(\eta_i)^2}{\mathrm{Var}(Y_i)}\right)$. In our case, the negative binomial distribution with the log-link, $\mathbf{W} = \mathrm{diag}\left(\frac{\exp(\mathbf{x}_i^T \boldsymbol{\beta})^2}{\exp(\mathbf{x}_i^T \boldsymbol{\beta})(1+\alpha \exp(\mathbf{x}_i^T \boldsymbol{\beta}))}\right)$. To find $\hat{\boldsymbol{\beta}}$, insert (2.1) and (2.3) into (2.2). This is possible as long as α is fixed.

2.2.5 Properties of parameter estimators

It can be shown that asymptotically

$$\hat{\boldsymbol{\beta}} \approx N_p\left(\boldsymbol{\beta}, F^{-1}(\boldsymbol{\beta})\right)$$

and even

$$\hat{\boldsymbol{\beta}} \approx N_p\left(\boldsymbol{\beta}, \hat{F}^{-1}\left(\hat{\boldsymbol{\beta}}\right)\right)$$

where $\hat{F}^{-1}\left(\hat{\boldsymbol{\beta}}\right)$ is as in (2.3), but inserting $\hat{\boldsymbol{\beta}}$ into $\mathbf{W}$ to get $\hat{\mathbf{W}}$.

2.2.6 Wald test

For the GLM we want to test the multivariate hypotheses

$$H_0 : \boldsymbol{C\beta} = d$$
$$H_1 : \boldsymbol{C\beta} \neq d$$

where $\boldsymbol{C}$ is a $(r \times p)$ matrix. An estimator for $\boldsymbol{C\beta}$ is $\boldsymbol{C}\hat{\boldsymbol{\beta}}$, where

$$\mathrm{E}\left(\boldsymbol{C}\hat{\boldsymbol{\beta}}\right) = \boldsymbol{C\beta}$$

$$\mathrm{Cov}\left(\boldsymbol{C}\hat{\boldsymbol{\beta}}\right) = \boldsymbol{C} \cdot \mathrm{Cov}\left(\hat{\boldsymbol{\beta}}\right)\boldsymbol{C}^T = \boldsymbol{C}F^{-1}\left(\boldsymbol{\beta}\right)\boldsymbol{C}^T$$

The Wald test statistic is

$$w = \left(\boldsymbol{C}\hat{\boldsymbol{\beta}} - \boldsymbol{d}\right)^T \left[\boldsymbol{C}F^{-1}\left(\hat{\boldsymbol{\beta}}\right)\boldsymbol{C}^T\right]^{-1}\left(\boldsymbol{C}\hat{\boldsymbol{\beta}} - \boldsymbol{d}\right)$$

which is asymptotically χ^2-distributed with r degrees of freedom (Fahrmeir et al., 2013, pp. 286).

2.3 Several SNPs and multiple testing

In this project we want to test multiple SNPs, and for this we may use multiple hypothesis testing. We assume that we have m hypothesis tests, which gives m p-values, and then we choose a cut-off on the p-values at a local significance level α_{loc} to decide if we want to reject each null hypothesis. The null hypotheses with p-value lower than α_{loc} are rejected, and this gives R rejected null hypotheses. The number of false null hypotheses that are rejected hypotheses is called S, and the number of true null hypotheses that are rejected is called V (type I error). The number of true null hypotheses that is not rejected is called U, and the number of false null hypotheses that are not rejected is called T (type II error). This gives the Table 2.2. The only quantities that are observed are R and m.

	Not reject H_0	Reject H_0	Total
H_0 true	U	V	m_0
H_0 false	T	S	$m - m_0$
Total	$m - R$	R	m

Table 2.2: Multiple testing set-up (Benjamini and Hochberg, 1995).

2.3.1 Familywise error rate

The familywise error rate (FWER) is defined as the probability of one or more false positive findings

$$\text{FWER} = P(V > 0)$$

where V is the number of type I errors in the m hypothesis tests. Here V is an unobservable random variable. To control the FWER we find a local significance level cut-off α_{loc} to be used on each of the m hypothesis tests.

2.3.2 Bonferroni method

The Bonferroni method is a method for controlling the FWER. We have

$$\text{FWER} = P(R_1 \cup \cdots \cup R_m) \leq \sum_{j=1}^{m} P(R_j) = \sum_{j=1}^{m} \alpha_{loc} = m\alpha_{loc}$$

where R_j is the event where we reject the hypothesis number j. Setting the FWER to α, we solve

$$m\alpha_{loc} = \alpha$$

which gives the local significance level

$$\alpha_{loc} = \frac{\alpha}{m}$$

for the Bonferroni method.

Chapter 3

Analysing gene expression data

In this chapter, we will apply the theory from Chapter 2 on the dataset briefly presented in Chapter 1. We will start by looking at design matrices and their connections to genetic models, interactions between covariates and introduce the term contrasts. Then we present the two technologies and use them both to analyse one SNP.

3.1 Design matrices and contrasts

There are different ways to look at the effect of a SNP. In this thesis we will look at additive SNP effects and relation between additive SNP effect and disease. For completeness next we give a presentation of the additive, dominant, recessive and codominant genetic models.

3.1.1 Genetic models

Let the most common variant of the SNP be called a, while the least common is called A. The SNP statuses are represented by numbers, where $aa = 0$, $Aa = 1$ and $AA = 2$. First, the additive model assumes that the mean increase in the response when comparing two different SNP statuses is linear dependent on the number of as. The dominant model assumes that the change in the response depends on whether there is any A in the SNP or not, regardless of how many. The recessive model assumes that the only change in response is when the SNP is AA. The codominant model assumes that the response changes, but not linearly, with SNP status. This is illustrated in Figure 3.1.

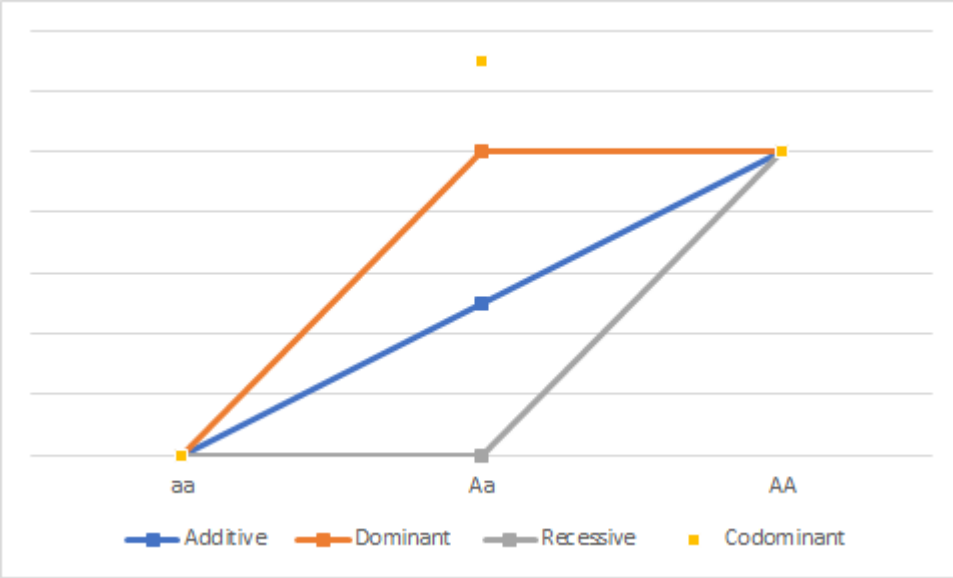


Figure 3.1: This figure shows different genetic models. On the vertical axis we have gene expression on some scale.

The genetic models can be represented as codings in design matrices. We will now consider a hypothetical data set with three individuals with SNP statuses $= [aa\ Aa\ AA]^T$. The additive model is linear, where the SNP status is a linear factor:

$$\mathbf{X}_{add} = \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix}$$

The recessive model represents whether the SNP status is AA or not:

$$\mathbf{X}_{res} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

The dominant model represents whether the SNP status is aa or not:

$$\mathbf{X}_{dom} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$$

The codominant model can be regarded as one factor with three levels. The SNP status aa is the reference level. The first column represents whether the SNP status is Aa or not, and the second column whether the status is AA or not. The model is then:

$$\mathbf{X}_{cod} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

As mentioned above, we will use the additive encoding in this thesis.

3.1.2 Interactions

We now examine two different covariates (SNP status and disease status), to study their effect in the linear predictor. These effects are not necessarily additive, and there might be an additional effect because the effect of the SNP on the response varies for different diseases. This means that we have to consider models with an additional interactive effect as well. Consider 15 individuals, where these 15 have all possible combinations of the 5 diseases statuses and the 3 SNP statuses. For our running example with the additive model, the $\boldsymbol{\beta}$ and design matrix for the microarray data will be:

$$\boldsymbol{\beta} = \begin{bmatrix} \beta_{diseaseN} \\ \beta_{diseaseCDA} \\ \beta_{diseaseCDU} \\ \beta_{diseaseUCA} \\ \beta_{diseaseUCU} \\ \beta_{diseaseN:snp} \\ \beta_{diseaseCDA:snp} \\ \beta_{diseaseCDU:snp} \\ \beta_{diseaseUCA:snp} \\ \beta_{diseaseUCU:snp} \end{bmatrix}$$

$$\mathbf{X}_{micro} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 2 \end{bmatrix} \tag{3.1}$$

This means that in row 1 the sample has disease status N and SNP status 0, in row 5 the sample has disease status CDA and SNP status 1 and in row 15 the sample has disease status UCU and SNP status 2. The RNASeq data does not contain any samples with disease status N, so the $\boldsymbol{\beta}$ and the design matrix are

$$\boldsymbol{\beta} = \begin{bmatrix} \beta_{diseaseCDA} \\ \beta_{diseaseCDU} \\ \beta_{diseaseUCA} \\ \beta_{diseaseUCU} \\ \beta_{diseaseCDA:snp} \\ \beta_{diseaseCDU:snp} \\ \beta_{diseaseUCA:snp} \\ \beta_{diseaseUCU:snp} \end{bmatrix}$$

$$\mathbf{X}_{RNASeq} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 2 \end{bmatrix} \tag{3.2}$$

This means that in row 1 the sample has disease status CDA and SNP status 0, in row 5 the sample has disease status CDU and SNP status 1 and in row 12 the sample has disease status UCU and SNP status 2.

We have chosen a parameterization with main effect of disease and interaction between SNP and disease. Instead, we could have chosen a parametrization with intercept, dummy variable coding of main effect of disease, linear (additive) main effect of SNP and dummy variable coding of the interaction between SNP and disease. In both of these two parameterizations we would be able to estimate the same number of parameters, 10 for the microarray data and 8 for the RNASeq data.

3.1.3 Contrasts of interest

Assume we are interested in looking at the difference between inflamed and uninflamed tissues, independent of the disease status. Then the data are grouped into two groups: CDA+UCA (inflamed) and CDU+UCU (uninflamed). To study this, we multiply the $\boldsymbol{\beta}$ vector with a vector $\boldsymbol{C}$. The $\boldsymbol{C}$ vector can be written in different ways, depending on which elements of $\boldsymbol{\beta}$ we want the difference between. In our case, and for the RNASeq data, there are four interesting $\boldsymbol{C}$ vectors, called $\boldsymbol{C}_0$, $\boldsymbol{C}_1$, $\boldsymbol{C}_2$ and $\boldsymbol{C}_3$. The first is

$$\boldsymbol{C}_0 = \begin{bmatrix} 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

and shows the difference

$$\left(\beta_{diseaseCDA} + \beta_{diseaseUCA} - \left(\beta_{diseaseCDU} + \beta_{diseaseUCU}\right)\right).$$

This is the difference between inflamed and uninflamed tissues when the SNP status equals 0. The second $\boldsymbol{C}$ vector is $\boldsymbol{C}_1$

$$\boldsymbol{C}_1 = \begin{bmatrix} 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \end{bmatrix}$$

and shows the difference

$$\begin{aligned} &\beta_{diseaseCDA} + \beta_{diseaseCDA:snp} + \beta_{diseaseUCA} + \beta_{diseaseUCA:snp} \\ &\quad - (\beta_{diseaseCDU} + \beta_{diseaseCDU:snp} + \beta_{diseaseUCU} + \beta_{diseaseUCU:snp}) . \end{aligned}$$

This is the difference between inflamed and uninflamed tissues when the SNP status equals 1. The third $\boldsymbol{C}$ vector is $\boldsymbol{C}_2$

$$\boldsymbol{C}_2 = \begin{bmatrix} 1 & -1 & 1 & -1 & 2 & -2 & 2 & -2 \end{bmatrix}$$

and shows the difference

$$\begin{aligned} &\beta_{diseaseCDA} + 2\beta_{diseaseCDA:snp} + \beta_{diseaseUCA} + 2\beta_{diseaseUCA:snp} \\ &\quad - (\beta_{diseaseCDU} + 2\beta_{diseaseCDU:snp} + \beta_{diseaseUCU} + 2\beta_{diseaseUCU:snp}) . \end{aligned}$$

This is the difference between inflamed and uninflamed tissues when the SNP status equals 2. The fourth $\boldsymbol{C}$ vector is

$$\boldsymbol{C}_3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 \end{bmatrix}$$

and shows the difference

$$(\beta_{diseaseCDA:snp} + \beta_{diseaseUCA:snp} - (\beta_{diseaseCDU:snp} + \beta_{diseaseUCU:snp})) .$$

This is the difference between the effects of the change in SNP status (from 0 to 1 or from 1 to 2) for inflamed and uninflamed tissues. See above that $\boldsymbol{C}_3 = \boldsymbol{C}_1 - \boldsymbol{C}_0$. Since we have additive coding of SNP, $\boldsymbol{C}_3$ can also be interpreted as $\boldsymbol{C}_2 - \boldsymbol{C}_1$.

For the microarray data, the $\boldsymbol{\beta}$ includes N, so we have

$$\boldsymbol{C}_0 = \begin{bmatrix} 0 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

and

$$\boldsymbol{C}_3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 \end{bmatrix}$$

For both microarray data and RNASeq data, we do not have complete data, and the contrast vectors will be adjusted. There are only one observation for CDA in

our microarray data set, so we can not estimate $\beta_{diseaseCDA:snp}$. This means that the contrast vectors change to

$$\boldsymbol{C}_0 = \begin{bmatrix} 0 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\boldsymbol{C}_3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & -1 \end{bmatrix}$$

and similarly for other types of missing data.

3.2 Microarray and LM

For the microarray data, we will use LM. We have recieved data which are already preprocessed, by so-called quantile normalization and transformed to the $\log_2$-scale. As we will see in Section 3.3, the RNASeq data will be analysed on natural log scale. Hence, the preprocessed microarray data was divided by $\log_2(\exp(1))$ to more easily compare results across technologies. Using the methods presented in Chapter 2 with design matrix (3.1), the fitted model for our running example rs11150589 is presented in Figure 3.2 and Table 3.1.

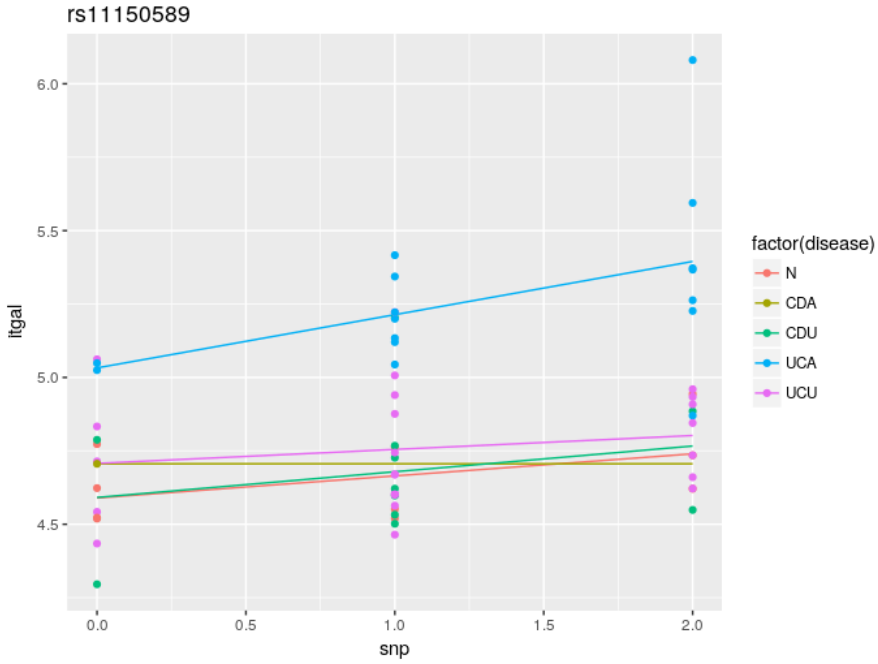


Figure 3.2: Fitted model for rs11150589. The observed values are represented as dots.

	Estimate	Std. Error	t value	Pr(> |t|)
diseaseN	4.589380	0.101404	45.258166	4.90E-44
diseaseCDA	4.705833	0.217488	21.637173	5.67E-28
diseaseCDU	4.591303	0.126986	36.156030	4.99E-39
diseaseUCA	5.032506	0.117950	42.666573	1.03E-42
diseaseUCU	4.707411	0.081622	57.673367	1.66E-49
diseaseN:snp	0.075255	0.082796	0.908918	3.68E-01
diseaseCDU:snp	0.087544	0.108744	0.805049	4.24E-01
diseaseUCA:snp	0.181252	0.079748	2.272795	2.71E-02
diseaseUCU:snp	0.047407	0.063224	0.749829	4.57E-01

Table 3.1: Summary of the fitted model for rs11150589.

There are only one observation of a sample with CDA, so there are no estimated effect of SNP for CDA. Out of the covariates representing interaction between SNP status and disease status, $\beta_{diseaseUCA:snp}$ is the only significant one (at level 0.05). To look at the effect of the difference between inflamed and uninflamed tissues, we use the contrast vector $\boldsymbol{C}_0$ which is presented in Section 3.1.3. We study

$$H_0 : \boldsymbol{C}_0\boldsymbol{\beta} = 0$$
$$H_1 : \boldsymbol{C}_0\boldsymbol{\beta} \neq 0.$$

We have

$$\begin{aligned}\boldsymbol{C}_0\hat{\boldsymbol{\beta}} &= \hat{\beta}_{diseaseCDA} + \hat{\beta}_{diseaseUCA} - (\hat{\beta}_{diseaseCDU} + \hat{\beta}_{diseaseUCU}) \\ &= 4.705833 + 5.032506 - (4.591303 + 4.707411) = 0.439625\end{aligned}$$

and the Wald test gives a p-value of

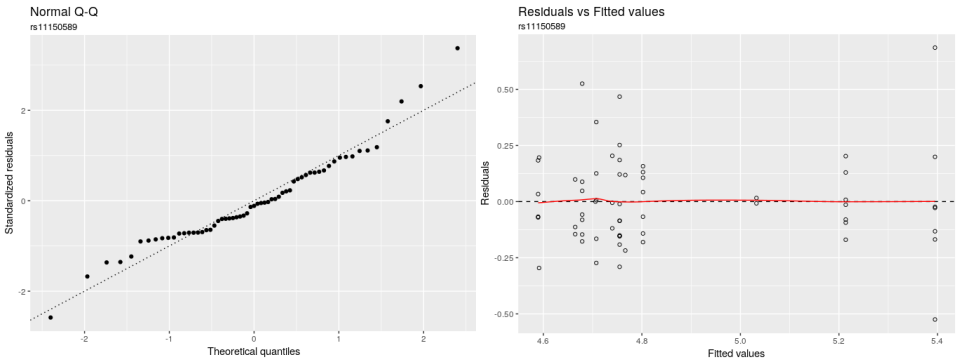
$$W_{disease} = 0.01582864$$

This means that there is a significant effect of the difference between inflammable and un-inflammable tissues. When we look at the contrast vector $\boldsymbol{C}_3$, we have

$$H_0 : \boldsymbol{C}_3\boldsymbol{\beta} = 0$$
$$H_1 : \boldsymbol{C}_3\boldsymbol{\beta} \neq 0$$

where

$$\begin{aligned}\boldsymbol{C}_3\hat{\boldsymbol{\beta}} &= \hat{\beta}_{diseaseUCA:snp} - (\hat{\beta}_{diseaseCDU:snp} + \hat{\beta}_{diseaseUCU:snp}) \\ &= 0.181252 - (0.087544 + 0.047407) = 0.046301\end{aligned}$$



(a) Normal QQ-plot. **(b)** Residuals vs fitted values.

Figure 3.3: Residual plots for rs11150589.

The Wald test gives a p-value of

$$W_{SNPi} = 0.3832292$$

which means that there are no significant effect of the difference between inflammable and un-inflammable tissues when we are looking at the interaction between disease status and SNP status. The normal QQ-plot is shown in Figure 3.3a and the plot for the residuals vs fitted values is shown in Figure 3.3b, and they look good. The Anderson-Darling normality test has a p-value = 0.001265. This means that even though the plots look good, we have to show caution when interpreting p-values, because the error terms are not necessarily normally distributed.

3.3 RNASeq and GLM

We want to compare the gene expressions for patients with different SNP statuses and disease statuses. When analysing RNASeq data, the gene expression is represented as count data. We used ESNG-number for ITGAL to find correct transcript. For preprocessing, the transcript expression values were generated by quasi alignment using Salmon and the Ensemble (GRCh38) human transcriptome. Aggregation of transcript to gene expression was performed using tximport. Gene expression values with CPM (counts per million) below one in more than three samples were filtered out before differential expression analysis. The preprocessing was performed at the group of Atle van Beelen Granlund, and we received preprocessed count data.

According to Holmes and Huber (2018), there are challenges in using count data. This is because their distribution is not symmetric, and the variance and distribution shape of the data in different parts of the dynamic range are very different.

To not deviate too much from the notation used in Love et al. (2014a), we use the index i for the gene and j for the sample in this section. To compare our observations, due to the challenges in using count data, the data from different samples have to be scaled with a factor. To calculate this factor, we use the data from all available genes. In our case we have 58 051 genes. This scaling factor is called a size factor, and is a different number for each sample. The size factor s_j is calculated using the R function `estimateSizeFactors` from the R package `DESeq2` (Love et al., 2014b), which uses the "median ratio method" as described in Love et al. (2014a). For our data set, the size factors are

$$\boldsymbol{s} = \begin{bmatrix} 1.0467966 \\ 1.0031062 \\ 1.0885007 \\ 0.8742161 \\ 1.0313613 \\ \vdots \\ 1.2620621 \end{bmatrix}$$

The distribution of the size factors for all samples in our data set are presented in Figure 3.4.

Let K_{ij} be the RNASeq count for gene i and sample j, and m the number of samples, then we have

$$\hat{s}_j = \text{median}_i \frac{K_{ij}}{(\Pi_{v=1}^{m} K_{iv})^{1/m}}$$

where the denominator gives the geometric mean. Further, we assume

$$K_{ij} \sim \text{NB}(\mu_{ij}, \alpha_i)$$

where $\text{NB}(\mu_{ij}, \alpha_i)$ denotes the negative binomial distribution with mean μ_{ij} and dispersion parameter α_i. In Chapter 2, we saw that α can be regarded as a kind of dispersion parameter, and following Love et al. (2014a) we refer to α as a dispersion parameter here. We further assume that

$$\begin{aligned} \mu_{ij} &= s_j q_{ij} \\ \log(q_{ij}) &= \mathbf{x}_j^T \boldsymbol{\beta}_i \end{aligned}$$

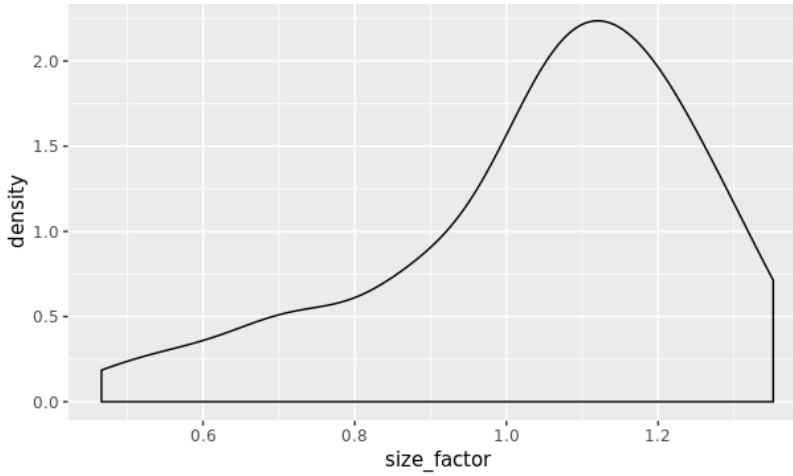


Figure 3.4: Distribution of size factors for all samples in our data set. The range of the data is 0.46-1.35.

In this thesis we are only looking at one gene (ITGAL), so we use the negative binomial model for one gene, omitting the i subscript. The observed count of the ITGAL RNASeq gene expression is K_j, where

$$K_j \sim \text{NB}(\mu_j, \alpha)$$

with

$$\mu_j = s_j q_j$$

$$\log(q_j) = \mathbf{x}_j^T \boldsymbol{\beta}$$

Observe that μ_j is a product. Using a log-link this can be written as

$$\log(\mu_j) = \log(s_j) + \log(\mathbf{x}_j^T \boldsymbol{\beta})$$

where the term $\log(s_j)$ is called an offset and is considered a constant and not estimated in the GLM routine. In the GLM, we also assume that α is known. In GLM we have used $\phi = 1$ for negative binomial, but it can also be estimated in R. To estimate the dispersion factor α, the R function `estimateDispersions` is used. This function calculates dispersion estimates for negative binomial distributed data, borrowing strength across all genes, see Love et al. (2014a) for details on the estimation. The dispersion estimate for ITGAL is

$$\hat{\alpha}_{ITGAL} = 0.2202738.$$

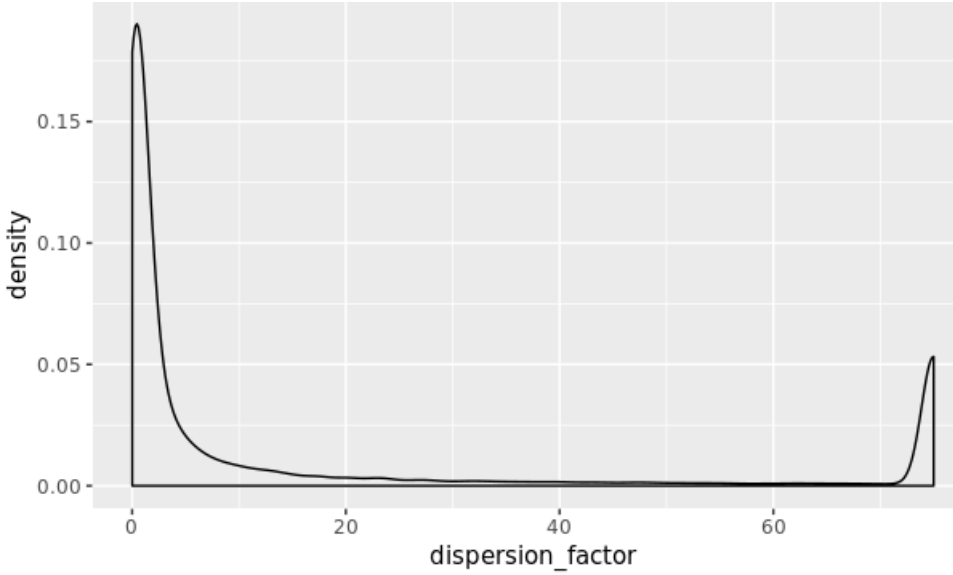


Figure 3.5: The distribution of dispersion factors for all genes in our data set.

The distribution of the dispersion factors for all genes in our data set are presented in Figure 3.5.

To perform hypothesis testing for many genes simultaneously, the R function `nbinomWaldTest` was considered, but it will not be used. This is because we have a large enough sample size to perform hypothesis testing without borrowing strength from other observations. We will instead use the R function `glm` with the negative binomial family from the R package `MASS` (Venables and Ripley, 2002). For comparison, we also show the result of running the `nbinomWaldTest`.

Using the methods presented in Chapter 2 with design matrix (3.2), the fitted model for our running example rs11150589 is presented in Figure 3.6 and Table 3.2. We want to plot the observed data. For plotting, it is recommended to use normalised data, transforming the data to get the same variance. There are many ways to do this. In Love et al. (2014a) it is recommended to use regularized logarithm transformation (`rlog`), which must be calculated based on all genes simultaneously. This operation is time-consuming when we have many genes. Another way to transform the data is to use a variance-stabilising transformation (`vst`), which stabilise the variance, but this transformation does not take the size factor into account. This means that `rlog` might be better, but due to this being very time consuming and we are only interested in the ITGAL gene, we have chosen to use `vst` in this thesis because it is faster. According to Love et al. (2014a) the value of rlog(K_{ij})

for large counts is approximately equal to $\log_2\left(\frac{K_{ij}}{s_j}\right)$. The `vst` values are used for plotting the observed values from the RNASeq data in Figure 3.6, Chapter 4 and Appendix C. The data shown are `vst` values divided by $\log(\exp(1))$, because we work on the natural log scale, while the DESeq2 uses log2 scale. Table 3.3 shows the results using the R function `nbinomWaldTest`, which uses almost 4 minutes and perform the calculations for all genes, but for one SNP. As we can see, the results are similar to the results from `glm` and we prefer using `glm` due to running time and because we are only interested in the results for ITGAL.

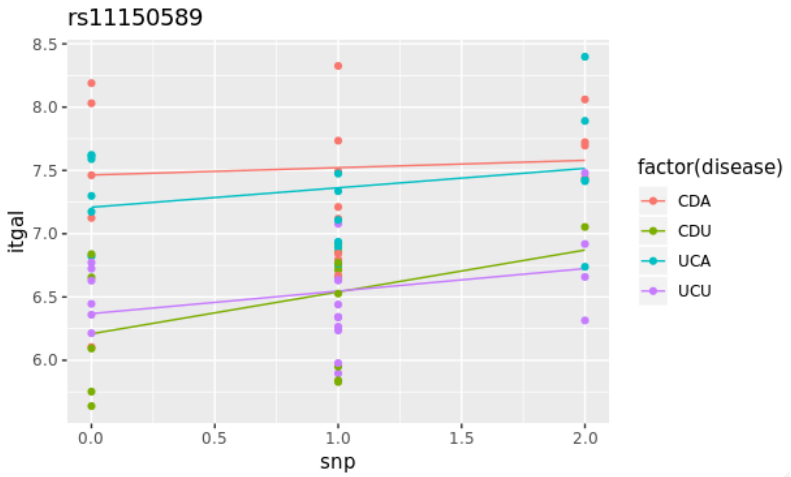


Figure 3.6: Fitted model for rs11150589.

	Estimate	Std. Error	z value	Pr(> |z|)
diseaseCDA	7.464120	0.185182	40.306983	0.00E+00
diseaseCDU	6.209766	0.196842	31.546945	1.97E-218
diseaseUCA	7.209524	0.171076	42.142231	0.00E+00
diseaseUCU	6.369039	0.171791	37.074384	7.27E-301
diseaseCDA:snp	0.057709	0.157115	0.367307	7.13E-01
diseaseCDU:snp	0.330731	0.209419	1.579282	1.14E-01
diseaseUCA:snp	0.153356	0.142041	1.079659	2.80E-01
diseaseUCU:snp	0.177812	0.150583	1.180825	2.38E-01

Table 3.2: Summary of the fitted model for rs11150589 using the R function `glm`.

None of the covariates representing interaction between SNP status and disease status are significant at level 0.05.

	Estimate	Std. Error	z value	Pr(> |z|)
diseaseCDA	7.464142	0.182841	40.823126	0.00E+00
diseaseCDU	6.209770	0.194369	31.948423	5.68E-224
diseaseUCA	7.209495	0.168915	42.681150	0.00E+00
diseaseUCU	6.369016	0.169635	37.545339	0.00E+00
diseaseCDA.rs11150589	0.057687	0.155128	0.371867	7.10E-01
diseaseCDU.rs11150589	0.330730	0.206784	1.599404	1.10E-01
diseaseUCA.rs11150589	0.153386	0.140246	1.093696	2.74E-01
diseaseUCU.rs11150589	0.177833	0.148692	1.195985	2.32E-01

Table 3.3: Summary of the fitted model for rs11150589 using the R function `nbinomWaldTest`.

To look at the effect of the disease groups, we use the contrast vector $\boldsymbol{C}_0$ which is presented in Section 3.1.3. We study

$$H_0 : \boldsymbol{C}_0\boldsymbol{\beta} = 0$$
$$H_1 : \boldsymbol{C}_0\boldsymbol{\beta} \neq 0.$$

We have

$$\begin{aligned}\boldsymbol{C}_0\hat{\boldsymbol{\beta}} &= \hat{\beta}_{diseaseCDA} + \hat{\beta}_{diseaseUCA} - (\hat{\beta}_{diseaseCDU} + \hat{\beta}_{diseaseUCU}) \\ &= 7.464120 + 7.209524 - (6.209766 + 6.369039) = 2.094839\end{aligned}$$

and the Wald test gives a p-value of

$$W_{disease} = 1.522387E - 08.$$

This means that there is an significant effect of the difference between inflammable and un-inflammable tissues. When we look at the contrast vector $\boldsymbol{C}_3$, we have

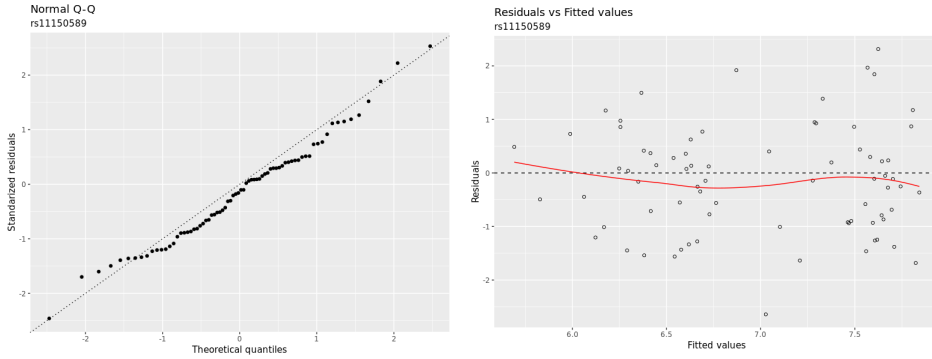
$$H_0 : \boldsymbol{C}_3\boldsymbol{\beta} = 0$$
$$H_1 : \boldsymbol{C}_3\boldsymbol{\beta} \neq 0$$

where

$$\begin{aligned}\boldsymbol{C}_3\hat{\boldsymbol{\beta}} &= \hat{\beta}_{diseaseCDA:snp} + \hat{\beta}_{diseaseUCA:snp} - (\hat{\beta}_{diseaseCDU:snp} + \hat{\beta}_{diseaseUCU:snp}) \\ &= 0.057709 + 0.153356 - (0.330731 + 0.177812) = -0.297478\end{aligned}$$

The Wald test gives a p-value of

$$W_{SNPi} = 0.382014$$



(a) Normal QQ-plot.

(b) Residuals vs fitted values.

Figure 3.7: Residual plots for rs11150589.

which means that there are no significant effect of the difference between inflammable and un-inflammable tissues when we are looking at the interaction between disease status and SNP status.

The normal QQ-plot is shown in Figure 3.7a and the plot for the residuals vs fitted values is shown in Figure 3.7b, and they look good. Even though we have fitted a GLM model with negative binomial distribution, it is not unusual to study residual plots for trends. We have plotted the deviance residuals, as explained in Dunn and Smyth (2018, pp. 297-300).

Chapter 4

eQTL analyses

4.1 Process overview

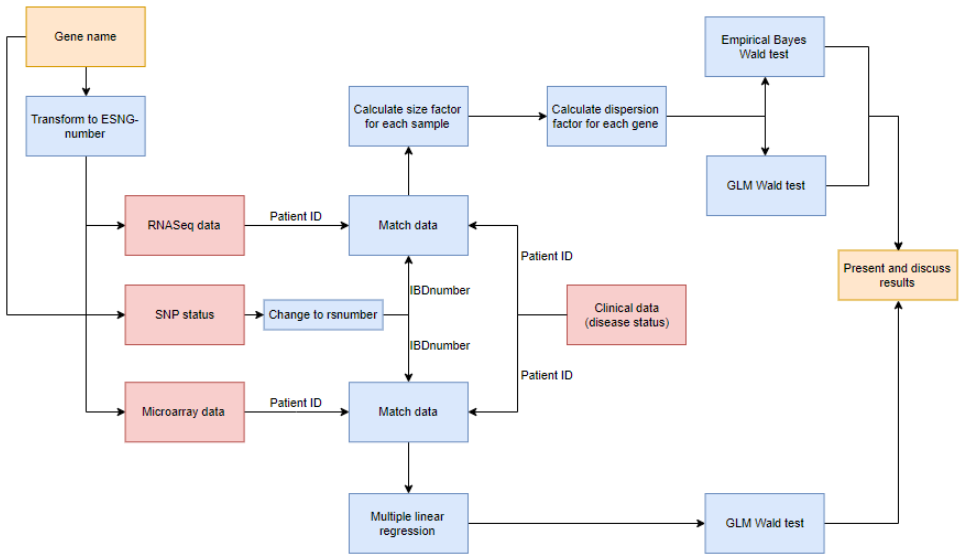


Figure 4.1: Flowchart explaining the operations performed on the gene expression and SNP data.

To perform the analysis presented in Chapter 3 and this chapter, we have executed a series of operations on the gene expression and SNP data. This is presented in the flowchart in Figure 4.1. We will now explain the process. The starting point is that the researcher want to perform an eQTL analysis for a specific gene. The

gene name, in our analysis ITGAL, is related to our RNASeq data and microarray data through an ESNG-number. We have used the R package `biomaRt` (Durinck et al., 2005) to extract the appropriate ESNG-number. We also used `biomaRt` to find the exact chromosomal position of the ITGAL gene, and from this the positions of SNPs inside and within a distance from the gene. We have used the stand-alone tool `vcftools` (Danecek et al., 2011) to extract SNP data at these locations (commands are presented in Appendix B). To identify a SNP other than by chromosomal position, we have used the rs coding scheme number, also obtained from `biomaRt`. The rs numbers are presented in Section 4.2.

The RNASeq data and the microarray data are connected to the clinical data, with information on disease status. From here, the data are treated differently. For the microarray data we use multiple linear regression, as presented in Section 3.2 and GLM Wald test. For the RNASeq data we use generalized linear models, as presented in Section 3.3, and GLM Wald test.

4.2 SNPs inside the ITGAL gene

In this pilot study, we have analysed the gene expression of ITGAL. This is a gene at chromosome 16, position 30 472 658 - 30 523 185. We now look at SNPs inside the ITGAL gene, positions 30 482 494 - 30 518 041. In our data, this interval contains nine genotyped SNPs, and we have transformed their chromosomal position to rs coding scheme number, as shown in Table 4.1. We have performed nine separate analyses, one for each genotyped SNP, for both the microarray and RNASeq data.

Chromosomal position	rs number
16:30482494	rs11150589
16:30490515	rs34166708
16:30490776	rs146094039
16:30492823	rs1064524
16:30493000	rs11574941
16:30495412	rs2285459
16:30500761	rs3087438
16:30516565	rs34838942
16:30518041	rs2230433

Table 4.1: Table showing connection between chromosomal position and rs number.

4.2.1 Microarray data

Our data set contains nine genotyped SNPs. The number of observations for each disease and SNP status is shown for each genotyped SNP in Table 4.2 for the microarray data. There is only one sample with CDA in this study, which makes it hard to estimate the general effect for samples with this disease. For the other diseases there are between 10 and 24 samples, which makes it possible to fit linear models. The ITGAL samples sorted by their SNP status and disease status are shown in Figure 4.2. We notice that the samples with UCA have higher ITGAL values than the others.

	CDA			CDU			N			UCA			UCU		
	0	1	2	0	1	2	0	1	2	0	1	2	0	1	2
rs11150589	1	0	0	2	7	2	4	3	3	2	7	7	5	12	7
rs34166708	1	0	0	11	0	0	10	0	0	16	0	0	24	0	0
rs146094039	1	0	0	11	0	0	10	0	0	16	0	0	24	0	0
rs1064524	1	0	0	11	0	0	10	0	0	14	2	0	19	5	0
rs11574941	1	0	0	8	3	0	6	3	1	12	3	1	20	4	0
rs2285459	1	0	0	2	7	2	4	3	3	2	7	7	4	11	9
rs3087438	1	0	0	4	6	1	5	5	0	5	7	4	8	12	4
rs34838942	1	0	0	11	0	0	10	0	0	16	0	0	24	0	0
rs2230433	0	1	0	5	4	2	4	3	3	8	7	1	18	5	1

Table 4.2: Number of observations for each disease status and SNP status for each genotyped SNP for the microarray data.

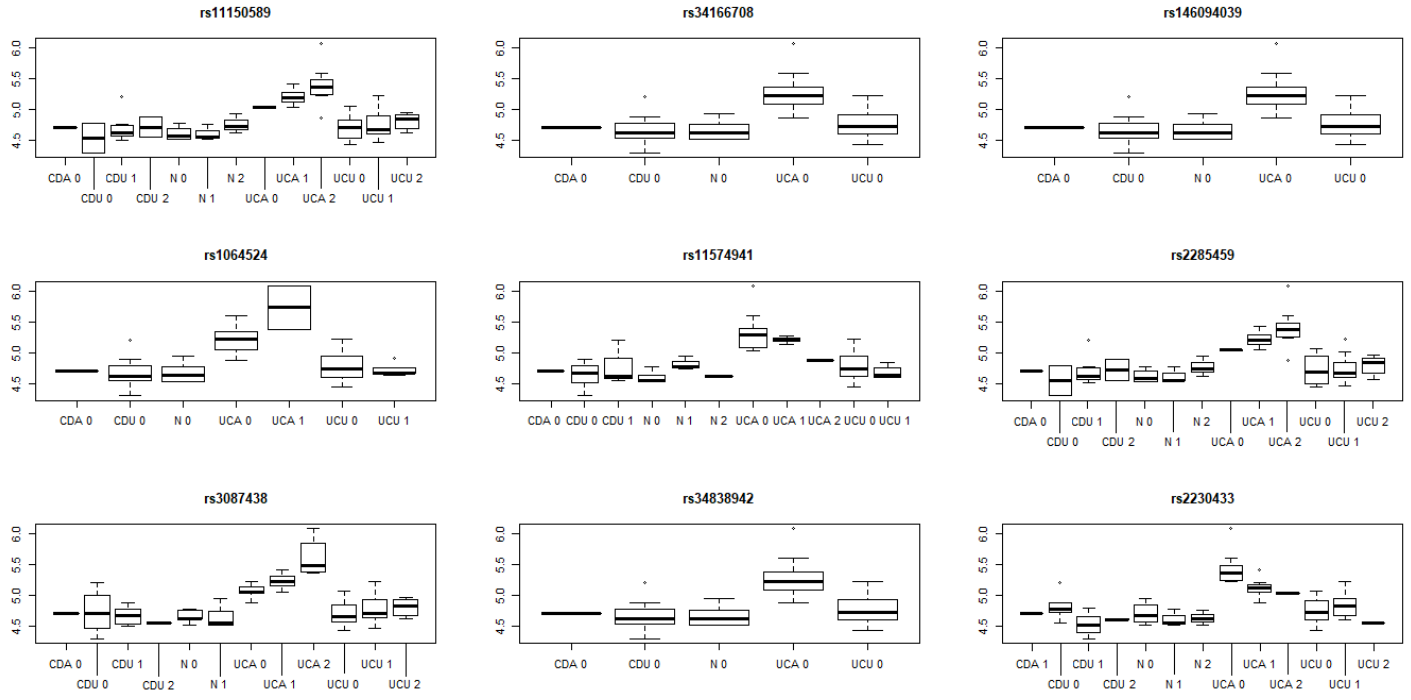


Figure 4.2: Microarray gene expression of ITGAL from samples sorted by their SNP status and disease status, with one plot for each genotyped SNP.

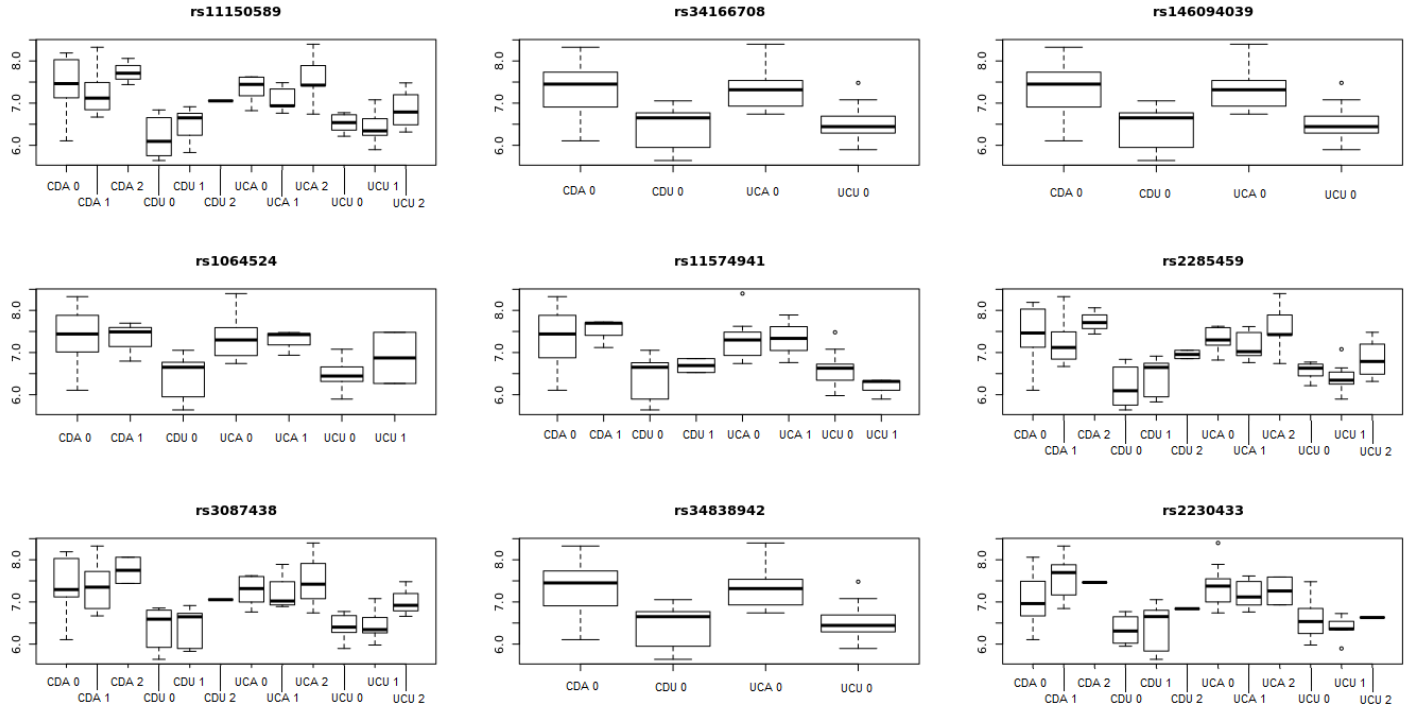


Figure 4.3: RNASeq gene expression of ITGAL from samples sorted by their SNP status and disease status, with one plot for each genotyped SNP.

4.2.2 RNASeq data

The number of observations for each disease and SNP status is shown for each genotyped SNP in Table 4.3 for the RNASeq data. There are no normal samples (labeled N) in this study, so the parameters $\beta_{diseaseN}$ and $\beta_{diseaseN:snp}$ will not be estimated here. For the other diseases there are between 17 and 20 samples, which makes it possible to fit linear models. The ITGAL values from samples sorted by their SNP status and disease status are shown in Figure 4.3. We notice that the samples with CDA and UCA, the inflamed tissues, have higher gene expression of ITGAL than the others.

	CDA			CDU			N			UCA			UCU		
	0	1	2	0	1	2	0	1	2	0	1	2	0	1	2
rs11150589	5	9	4	5	11	1	0	0	0	6	9	5	6	10	4
rs34166708	18	0	0	17	0	0	0	0	0	20	0	0	20	0	0
rs146094039	18	0	0	17	0	0	0	0	0	20	0	0	20	0	0
rs1064524	15	3	0	17	0	0	0	0	0	17	3	0	18	2	0
rs11574941	15	3	0	15	2	0	0	0	0	17	3	0	17	3	0
rs2285459	5	9	4	5	10	2	0	0	0	5	10	5	5	11	4
rs3087438	6	10	2	8	8	1	0	0	0	8	8	4	8	9	3
rs34838942	18	0	0	17	0	0	0	0	0	20	0	0	20	0	0
rs2230433	6	11	1	4	12	1	0	0	0	12	6	2	12	7	1

Table 4.3: Number of observations for each disease status and SNP status for each genotyped SNP for RNASeq data.

4.2.3 Minor allele frequency

We study biallelic SNPs, so there is a frequency for the most common allele (the major allele) and for the least common allele (the minor allele) (Nica and Dermitzakis, 2012). The minor allele frequencies (MAF) for the nine genotyped SNPs in this project are found by using `vcftools` and are presented in Table 4.4. The MAF influences the sample size for the number of persons with each genotype for the SNPs. If the two copies of a SNP that one person receives from his/her parents are inherited independently, then we would expect on average the following:

SNP status	Number of samples with this SNP status
0	$n \cdot (1 - \text{MAF})^2$
1	$n \cdot 2 \cdot \text{MAF}(1 - \text{MAF})$
2	$n \cdot \text{MAF}^2$

Here n is the total number of samples. The MAF for the nine genotyped SNPs are presented in Table 4.4. This explains the fact that we only have observed SNP status 0 for SNPs rs34166708, rs146094039 and rs34838942.

	MAF
rs11150589	0.4771
rs34166708	0.0115
rs146094039	0.0112
rs1064524	0.0601
rs11574941	0.1035
rs2285459	0.4784
rs3087438	0.3911
rs34838942	0.0059
rs2230433	0.2952

Table 4.4: Minor allele frequency for the nine genotyped SNPs.

4.3 Results from SNPs inside the ITGAL gene

4.3.1 Microarray data

For the microarray data, the $\boldsymbol{C}\hat{\boldsymbol{\beta}}$ and results from the Wald test are shown in Table 4.5, using the contrast vectors $\boldsymbol{C}_0$ and $\boldsymbol{C}_3$ for all SNPs. As there are only one observation of CDA, we can not estimate an interaction between SNP and the disease CDA. The contrast $\boldsymbol{C}_3$ is between the groups UCA and UCU+CDU. Looking at the difference in effect of the interaction between disease and SNP for these disease groups, this is significant at level 0.05 for two SNPs (when we do not correct for multiple testing): rs1064524 and rs3087438. When we correct for multiple testing using the Bonferroni method (with m = 13), only the result for rs1064524 is significant.

The plots and results for the microarray data are presented in Section 3.2 and Appendix C. The only significant covariate with interaction at level 0.05 (when we do not correct for multiple testing) is $\beta_{diseaseUCA:snp}$, which is significant for almost all SNPs. The exceptions are rs34166708, because there are no interaction terms here, and rs11574941, because this is borderline when we use the limit 0.05. When we correct for multiple testing using the Bonferroni method (with m = 57), only $\beta_{diseaseUCA:snp}$ for rs3087438 is significant.

SNP	Data	$\boldsymbol{C}\,\hat{\boldsymbol{\beta}}$	Wald test
rs11150589	CDA has no change in SNP status	0.43962416 0.0463003	0.1293071 0.7558989
rs34166708	All SNP status equals 0	0.5386151 0	0.02750116 NA
rs1064524	CDA and CDU have no change in SNP status	0.4634648 0.5691513	0.04251931 0.002489612
rs11574941	CDA has no change in SNP status	0.6216958 -0.2402065	0.01137627 0.2589941
rs2285459	CDA has no change in SNP status	0.43250705 0.05709455	0.1389713 0.701793
rs3087438	CDA has no change in SNP status	0.2780598 0.2891405	0.2568068 0.03036489
rs2230433	CDA has no change in SNP status	0.5841934 -0.1252623	0.01582864 0.3832292

Table 4.5: Results for the microarray data. The SNPs rs34166708, rs146094039 and rs34838942 have identical results because only SNP status 0 is observed for all samples, so only results for rs34166708 are shown. For each SNP there are two rows, where the first row shows $\boldsymbol{C}_0$ and the second shows $\boldsymbol{C}_3$. Wald test is performed and presented with 7 significant digits.

4.3.2 RNASeq data

For the RNASeq data, the $\boldsymbol{C}\hat{\boldsymbol{\beta}}$ and results from the Wald test are shown in Table 4.6, using the contrast vectors $\boldsymbol{C}_0$ and $\boldsymbol{C}_3$ for all SNPs. None of the effects of the interaction between SNP status and disease status are significant here, even if we do not correct for multiple testing (where we would have used the Bonferroni method with m = 13).

The plots and results for the RNASeq data are presented in Section 3.3 and Appendix C. There are no significant SNPs with the interaction term, even when we do not correct for multiple testing (where we would have used the Bonferroni method with m = 51).

SNP	Data	$C\,\hat{\beta}$	Wald test
rs11150589	Complete data set	2.0948389	1.522387E-08
		-0.2974784	0.382014
rs34166708	All SNP status equals 0	1.864325	6.73849E-16
		0	NA
rs1064524	CDU has no change in SNP status	1.9754792	4.501684E-17
		-0.8165941	0.1427063
rs11574941	Complete data set	1.8223896	5.091212E-13
		0.3318308	0.6185999
rs2285459	Complete data set	2.0511837	4.897462E-08
		-0.2316014	0.4829886
rs3087438	Complete data set	1.9975641	7.031259E-10
		-0.2010314	0.5453224
rs2230433	Complete data set	1.7956275	4.50224E-07
		0.1263684	0.7529079

Table 4.6: Results for the RNASeq data. The SNPs rs34166708, rs146094039 and rs34838942 have identical results because only SNP status 0 is observed for all samples, so only results for rs34166708 are shown. For each SNP there are two rows, where the first row shows C_0 and the second shows C_3. Wald test is performed and presented with 7 significant digits.

4.4 SNPs within a distance from ITGAL

As explained in Section 1.5, investigation of the SNPs around ITGAL is called local and distant cis-acting. In our case, the local cis-acting, from 29 472 658 to 31 523 185, includes 419 SNPs. The distant cis-acting, from 25 472 658 to 35 523 185, includes 1044 SNPs. In this section, we look at the distant cis-acting SNPs.

We will just look at the SNPs where our data has at least SNP status 0 and 1 for all disease statuses (CDA, CDU, UCA and UCU). The microarray data contains only one sample with CDA, so we have chosen to only perform analysis from the RNASeq data. Out of our 1044 SNPs, there are 468 SNPs which fulfill the restrictions. Using the Wald test with significance level 0.05, we have 36 significant SNPs. When we correct for multiple testing with Bonferroni method (with $m = 468$), there are no significant SNPs. The positions of the 36 SNPs are presented in Figure 4.4, and their correlation is presented in Figure 4.5.

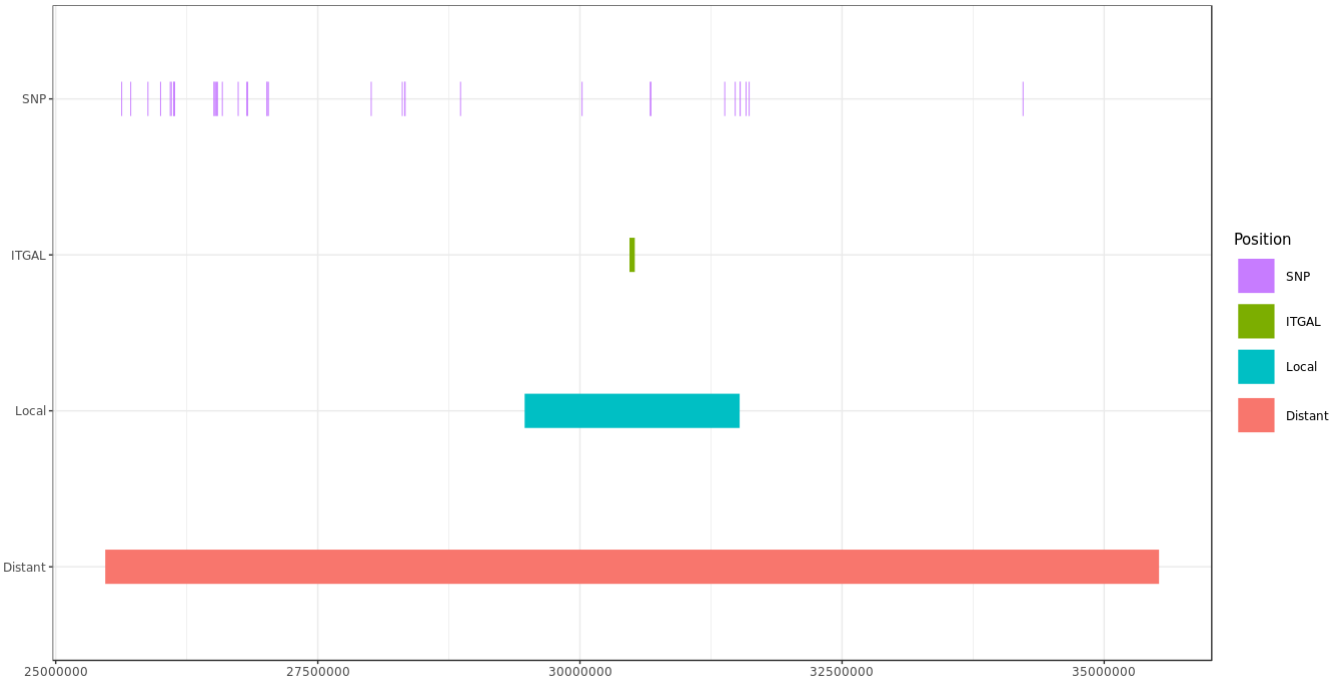


Figure 4.4: Figure showing the positions of the 36 SNPs with p-value below 0.05, the gene ITGAL and local and distant cis-acting regions.

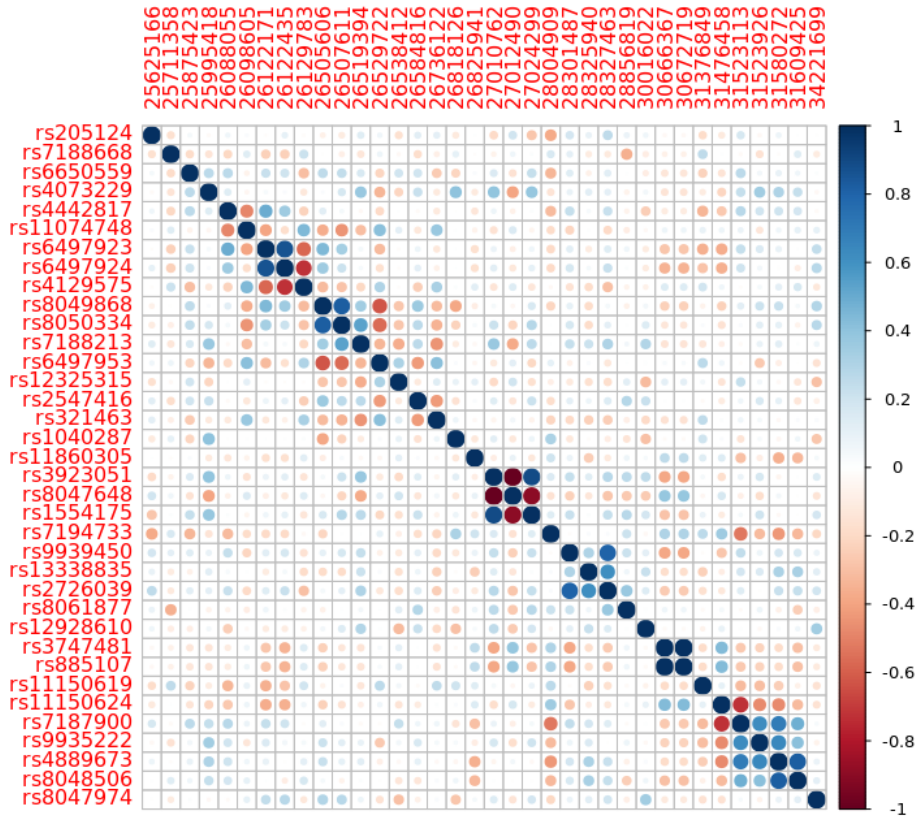


Figure 4.5: Correlation plot showing the correlation between the 36 SNPs. The column names are their position on chromosome 16, and the row names are their corresponding rs scheme number.

Chapter 5

Discussion and conclusion

This thesis is a pilot study to investigate how a change in SNP status affects the gene expression of ITGAL for different inflammable bowel diseases. We have fitted different models to describe this relationship. These models can be used to investigate gene expressions and SNPs inside and within a distance from other genes as well. We now present some of the challenges in the data analysis and the interpretation of the results, conclusion and further work.

Microarray and RNASeq data The microarray data and the RNASeq data come from different technologies and have different samples. The results can therefore not be compared directly. It would be interesting if a method could be developed so that all data - from both platforms - could be analysed together. At present, we have just compared the p-values from similar tests.

Independent measurements In the microarray dataset, there are 8 out of 54 persons with two observations, and in the RNASeq dataset, 22 out of 53 persons have two observations. When there are two observations from the same person, there is one observation from inflamed tissue and one observation from uninflamed tissue. In these cases, the observations are related. For the microarray data we fitted linear mixed effects models with person as random intercept. This gave close to identical results as using LM, so we will not use the random intercept model. For microarray data the intra class correlation was estimated to 0.21-0.37, see Table 4.1 in Mathisen (2018). For the RNASeq data we fitted a GLMM, which gave similar results. Based on this, we concluded that we could perform analysis with LM and GLM and assume that observations from the same person can be considered independent. However, this might not always be the case and care

should be taken to address this problem. To our knowledge a mixed effects version of DESeq is not available.

Interpretation Our main task in this project is to look for any relationship between change in the SNP status and the gene expression of ITGAL for the different diseases. We have considered models with an interaction term. This means that the effect is depending on both SNP status and disease status, so the effect of a change in SNP status is different for each disease. The use of plots of the type in Figure 3.2 and 3.6 have helped in presenting the results.

Sample size Even though our data set is large compared to other data sets in genomics we consider it small from a statistical perspective. The methods to acquire the data are expensive, so the sample size is affected by economical limitations. Small data sets (from a statistical perspective) makes it hard to fit precise models. For the microarray data, we have only one patient with the disease CDA, so we could not estimate any effect of the interaction between SNP status and the disease CDA. Some of the genotyped SNPs in this project have low MAF-values, which means that the least common allele is rarely observed, and in a small data set they might not even occur at all. For the genotyped SNPs rs34166708, rs146094039 and rs34838942 there are only observations with SNP status 0. For the microarray data, the genotyped SNP rs1064524 have no observations with SNP status 2, while there are 55 observations with SNP status 0 and 7 observations with SNP status 1. For the RNASeq data, rs1064524 and rs11574941 have no observations with SNP status 2. Overall, we have to take care when interpreting the results.

Multiple testing In this thesis, we have used $\alpha = 0.05$ as significance level for all tests before correcting for multiple testing. In Section 4.4 we looked at $m = 468$ hypothesis tests for the 468 SNPs which fulfilled our restrictions. Using the Bonferroni method to calculate a local significance level, we get $\alpha_{loc} = \frac{\alpha}{m} = \frac{0.05}{468} = 1.0684 \cdot 10^{-4}$. With this local significance level, no SNPs are significant. In general when we use multiple testing, we have to be precise on what we are inspecting. Considering the Wald tests for $\boldsymbol{C}_3\boldsymbol{\beta}$ for all SNPs from RNASeq, we have 1044 hypothesis tests, which gives $\alpha_{loc} = \frac{0.05}{1044} = 4.7893 \cdot 10^{-5}$. If we include $\boldsymbol{C}_0\boldsymbol{\beta}$ we have 2088 hypothesis tests, and if we include microarray data as well, we have 4176 hypothesis tests. The local significance level will be respectively $\alpha_{loc} = 2.3946 \cdot 10^{-5}$ and $\alpha_{loc} = 1.1973 \cdot 10^{-5}$. This means that the number of significant covariates depends on which hypothesis tests we decide to look at. When we analyse more than one gene, this problem will be even more prominent and has to be addressed properly.

Running time To choose whether we wanted to use the function `glm` or `nBinomWaldTest`, we investigated the running time. For one SNP, the function `nBinomWaldTest` used 222.370 seconds (which equals 3 minutes and 42 seconds). This is in large contrast to `glm`, which only used 0.007 seconds. Considering few SNPs, both functions could have been used. However, in Section 4.4 there are 1044 SNPs, and performing the `nBinomWaldTest` on all of them would have used almost 2.7 days. Due to this and the relatively large sample size for RNASeq data, and because of the similar results for these functions (presented in Section 3.3), we have chosen to use `glm` in this thesis.

Conclusion and further work We have fitted multiple linear models and generalized linear models for investigating the relation between gene expression, SNP status and disease status. When we look at how the gene expression of ITGAL is affected by the SNP status for different diseases, we have some significant results for the microarray data. For the RNASeq data there are no significant results, even if we do not correct for multiple testing using Bonferroni method. We have also compared the two groups of inflamed and uninflamed tissues, regardless of disease. We have significant results for microarray data here as well, but only if we do not correct for multiple testing. For the RNASeq data, there are no significant results at all. For further work, we could look at gene expressions and corresponding SNPs for other genes.

Bibliography

Aabakken, L., 2016. Inflammatorisk Tarmsykdom. Store Medisinske Leksikon. https://sml.snl.no/inflammatorisk_tarmsykdom, Online; accessed 9-November-2018.

Aabakken, L., 2018. Crohns Sykdom, store Medisiske Leksikon. https://sml.snl.no/Crohns_sykdom, Online; accessed 9-November-2018.

Aabakken, L., Halstensen, T., 2018. Ulcerøs Kolitt, Store Medisinske Leksikon. https://sml.snl.no/ulcer%C3%B8s_kolitt, Online; accessed 9-November-2018.

Agresti, A., 2015. Foundations of Linear and Generalized Linear Models. John Wiley and Sons, Inc.

Alberts, B., Bray, D., Hopkin, K., Johnson, A., Lewis, J., Raff, M., Roberts, K., Walter, P., 2014. Essential Cell Biology. Garland Science.

Benjamini, Y., Hochberg, Y., 1995. Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. Journal of the Royal Statistical Society. Series B (Methodological), vol. 57, no. 1, 1995, pp. 289–300. JSTOR, JSTOR, www.jstor.org/stable/2346101.

Danecek, P., Auton, A., Abecasis, G., Albers, C. A., Banks, E., DePristo, M. A., Handsaker, R. E., Lunter, G., Marth, G. T., Sherry, S. T., McVean, G., Durbin, R., 2011. The variant call format and VCFtools. Bioinformatics.

Das, K., Imon, A. H. M. R., 2016. A Brief Review of Tests for Normality. American Journal of Theoretical and Applied Statistics. Vol. 5, No. 1, 2016, pp. 5-12. doi: 10.11648/j.ajtas.20160501.12.

de Jong, P., Heller, G. Z., 2008. Generalized Linear Models for Insurance Data. Cambridge University Press.

Dunn, P. K., Smyth, G. K., 2018. Generalized Linear Models with Examples in R. Springer Texts in Statistics.

Durinck, S., Moreau, Y., Kasprzyk, A., Davis, S., De Moor, B., Brazma, A., Huber, W., 2005. Biomart and Bioconductor: a powerful link between biological databases and microarray data analysis. Bioinformatics 21, 3439–3440.

Fahrmeir, L., Kneib, T., Lang, S., Marx, B., 2013. Regression. Springer.

Gross, J., Ligges, U., 2015. nortest: Tests for Normality. R package version 1.0-4. URL `https://CRAN.R-project.org/package=nortest`

Holmes, S., Huber, W., 2018. Modern Statistics for Modern Biology. Cambridge University Press, http://web.stanford.edu/class/bios221/book/Chap-CountData.html.

Jostins, L., Ripke, S., Weersma, R., et al., 2012. Host-microbe interactions have shaped the genetic architecture of inflammatory bowel disease. Nature. 2012;491(7422):119-24.[10.1038/nature11582].

Love, M. I., Huber, W., Anders, S., 2014a. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. Genome Biology, 15:550 doi:10.1186/s13059-014-0550-8.

Love, M. I., Huber, W., Anders, S., 2014b. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. Genome Biology 15, 550.

Martinsen, L., 2019. DNA. Store norske leksikon. `https://snl.no/DNA`, Online; accessed 9-June-2019.

Mathisen, K. L., 2018. Identifying expression quantitative trait loci in patients with inflammatory bowel disease. Unpublished project thesis TMA4500.

Nica, A., Dermitzakis, E., 2012. Expression quantitative trait loci: present and future. Phil Trans R Soc B 368: 20120362. http://dx.doi.org/10.1098/rstb.2012.0362.

Quarteroni, A., Sacco, R., Saleri, F., 2007. Numerical mathematics. Springer.

Robinson, M. A., 2004. Linkage disequilibrium. `https://doi.org/10.1006/rwei.1999.0406`, Online; accessed 9-June-2019.

Stephens, M. A., 1974. EDF Statistics for Goodness of Fit and Some Comparisons. Journal of the American Statistical Association, 69(347), 730-737. doi:10.2307/2286009.

Venables, W. N., Ripley, B. D., 2002. Modern Applied Statistics with S, 4th Edition. Springer, New York, ISBN 0-387-95457-0.
URL http://www.stats.ox.ac.uk/pub/MASS4

Appendix A

R code

A.1 Linear models for microarray data

```
fileloc #change to your directory
#now reading 012-files made with vcftools

filepref="ITGALout.012"
mat=read.table(paste(fileloc,filepref,sep=""))[,-1]
dim(mat)
#432   9
#We have 432 samples with SNP status for the 9 SNPs

pasnames=rownames(mat)=scan(paste(fileloc,filepref,".indv",sep="")
    ,what="s")
snpids=colnames(mat)=read.table(paste(fileloc,filepref,".pos",sep=
    ""),header=FALSE)[,2]

#We have 432 ids, and only a few of those are IBDs
#Working to split names of persons
df=data.frame("id"=pasnames,stringsAsFactors = FALSE)
df$id
df %>% dplyr::rowwise() %>% dplyr::mutate(new_id = substring(id,
    first=nchar(id)-2,last=nchar(id)),
                                          batch = substring(id,
    first=nchar(id)-3,last=nchar(id)-3)
) -> snp.ids
snp.ids$id0=NA
snp.ids$id0[snp.ids$batch==0]=snp.ids$new_id[snp.ids$batch==0]
snp.ids
table(snp.ids$batch)
table(snp.ids$id0)
#batch and newid together defines persons, same newid is not same
```

```
    person if not batch is the same

any(table(snp.ids$new_id,snp.ids$batch)>1)
# we only consider batch=0 for these analyses
# FINAL snp.ids to be used further

# now reading gene expression
ge=read.csv(paste(fileloc,"expressionSet_ITGAL_IFNG.csv",sep=""))
# dummy variable coding for the 5 groups, we also make a factor

# making class for patients
apply(ge[,4:8],1,sum)
ge.class=rep("N",dim(ge)[1])
ge.class[ge[,5]==1]="CDA"
ge.class[ge[,6]==1]="UCA"
ge.class[ge[,7]==1]="CDU"
ge.class[ge[,8]==1]="UCU"
ge.class
table(ge.class)
apply(ge[,4:8],2,sum)

# patient id from ge
df=data.frame(id=as.character(ge[,1]),stringsAsFactors = FALSE)
df %>% dplyr::rowwise() %>% dplyr::mutate(new_id = substring(id,
    first=1,last=3)) -> ge.ids
ge.ids
# here 108S and 108F is the same person, but two tissues -
    inflamed (S) and uninflamed (F)

# matching ge and snp patients
idge<-match(ge.ids$new_id,snp.ids$id0)
idge
sum(!is.na(idge))
length(ge.ids$new_id)

"SNP"=gt.df$new_gt[idge[!is.na(idge)]]

snpge=data.frame("ITGAL"=ge$ITGAL[!is.na(idge)],mat[idge[!is.na(
    idge)],],"PAS.ID"=snp.ids$id0[idge[!is.na(idge)]],
                 "PAS.SNP"=snp.ids$id[idge[!is.na(idge)]],"PAS.GE"
    =ge.ids$id[!is.na(idge)],"CLASS"=ge.class[!is.na(idge)])


#Change from chromosomal position to rs scheme number
variation = useEnsembl(biomart = "snp", dataset = "hsapiens_snp",
    host="grch37.ensembl.org")
snp_library = getBM(attributes = c('refsnp_id','allele','chrom_
    start','chrom_strand'),
                    filters = c('chr_name','start','end'),
```

```
                    values= list(16, 30482400, 30518060),
                    mart = variation)
cbind(snp_library$refsnp_id,snp_library$chrom_start)
colnames(snpge)[2:10]= snp_library$refsnp[match(colnames(snpge)
    [2:10],snp_library$chrom_start)]


snp_analysis = data.frame("itgal" = snpge$ITGAL/log2(exp(1)), "snp
    " = snpge[i+1], "disease" = snpge$CLASS, "pas" = snpge$PAS.SNP
    )
snp_analysis$disease <- relevel(snp_analysis$disease, ref = "N")
colnames(snp_analysis)[2] = "snp"
best_model = lm(itgal~disease-1+disease:snp,data=snp_analysis)
```

A.2 Generalized linear models for RNASeq data

```

pas_info_tot <- read.csv(paste(fileloc,'sampleSheetColon.csv',sep=
    ""))
pas_info <- data.frame("pas_id" = pas_info_tot$X, "disease" = pas_
    info_tot$Sample_Group, "biosource" = pas_info_tot$Sample_
    Biosource)
#extract relevant columns

pas_info$pas_id <- as.character(pas_info$pas_id)

pas_info$ibdnumber <- substr(str_split_fixed(pas_info$pas_id, "_",
     2)[,1], 1, nchar(str_split_fixed(pas_info$pas_id, "_", 2)
    [,1])-1)
pas_info$letter <- substr(str_split_fixed(pas_info$pas_id, "_", 2)
    [,1], nchar(str_split_fixed(pas_info$pas_id, "_", 2)[,1]),
    nchar(str_split_fixed(pas_info$pas_id, "_", 2)[,1]))
pas_info$add_info <- str_split_fixed(pas_info$pas_id, "_", 2)[,2]


for (i in 1:nrow(pas_info)){
  if(nchar(pas_info$ibdnumber[i]) == 3){
    pas_info$ibdnumber[i] <- paste0('IBD0', pas_info$ibdnumber[i])
  } else {
    pas_info$ibdnumber[i] <- paste0('IBD', pas_info$ibdnumber[i])
  }
}

#change order of columns
pas_info <- pas_info[,c(1,4,5,6,2,3)]
pas_info

#> dim(pas_info)
#[1] 113   6
```

```
27
28
29 ##SNP status
30 gene_name = "ITGAL"
31 filepref="ITGALny.012"
32 snp_status=read.table(paste(fileloc,filepref,sep=""))[,-1]
33 pasnames=rownames(snp_status)=scan(paste(fileloc,filepref,".indv",
       sep=""),what="s")
34 snpids=colnames(snp_status)=read.table(paste(fileloc,filepref,".
       pos",sep=""),header=FALSE)[,2]
35 row.names(snp_status) <- substring(row.names(snp_status), nchar(
       row.names(snp_status))-6, nchar(row.names(snp_status)))
36 #snp_status
37
38 #> dim(snp_status)
39 #[1] 432   9
40
41
42 ensembl = useMart("ensembl", dataset = "hsapiens_gene_ensembl")
43 gene_ensg <- getBM(attributes = c("ensembl_gene_id"), filters = "
       hgnc_symbol", values = gene_name, mart = ensembl)$ensembl_gene
       _id
44 # now gene_ensg is the name for ITGAL
45
46 snps = colnames(snp_status)
47 idmatch=match(pas_info$ibdnumber,rownames(snp_status))
48
49 pas_info_new=cbind(pas_info[idmatch,],snp_status[idmatch,])
50
51 for (i in snps){
52   pas_info <- cbind(pas_info, snp_status[,i][match(pas_info$
       ibdnumber, rownames(snp_status))])
53   names(pas_info)[length(names(pas_info))] <- i
54 }
55
56
57 pas_info_tot <- pas_info[complete.cases(pas_info), ]
58 id=(1:dim(pas_info_tot)[1])[pas_info_tot$biosource == "Colon
       tissue"]
59 pas_info_tot <- pas_info_tot[id,]
60
61
62 #change to rs scheme number
63
64 #listEnsembl()
65 variation = useEnsembl(biomart = "snp", dataset = "hsapiens_snp",
       host="grch37.ensembl.org")
66 snp_library = getBM(attributes = c('refsnp_id','allele','chrom_
       start','chrom_strand'),
```

```
                    filters = c('chr_name','start','end'),
                    values= list(16, 30482400, 30518060),
                    mart = variation)
cbind(snp_library$refsnp_id,snp_library$chrom_start)
#variation = useEnsembl(biomart = "snp", dataset = "hsapiens_snp")
colnames(pas_info_tot)[7:15]= snp_library$refsnp[match(colnames(
    pas_info_tot)[7:15],snp_library$chrom_start)]


# no over to RNAseq data

counts2 <- counts
names(counts2) <- substring(names(counts2), 2)
counts2 <- counts2[,colnames(counts2)%in%pas_info_tot$pas_id]

#> dim(counts2)
#[1] 58051    75

# we only want ITGAL now, and the name is in gene_ensg

############## for SNP9 (we can do this for all SNPs)

dd9 <- DESeqDataSetFromMatrix(countData = counts2,
                              colData = pas_info_tot, design = ~
    disease-1+disease:rs2230433)
dd9=estimateSizeFactors(dd9)
dd9=estimateDispersions(dd9)
dd9=nbinomWaldTest(dd9)
results(dd9)
dd9raw=mcols(dd9)
names(dd9raw)

idrow=na.omit(match(gene_ensg,rownames(dd9raw)))
# 108 - this is the only row we need

dd9raw[idrow,]
results(dd9)[gene_ensg,]

alpha=dispersions(dd9)[idrow]
ydata=unlist(counts2[idrow,])
data=data.frame(y=ydata,s=sizeFactors(dd9),
                disease=pas_info_tot$disease,
                snp=pas_info_tot$rs2230433,
                patient=pas_info_tot$ibdnumber)

fit=glm(y~disease-1+disease:snp,offset=log(s),
        family=negative.binomial(theta=1/alpha,link="log"),data=
    data)
summary(fit, dispersion = 1) #We set dispersion to 1
```

A.3 Wald test

```
#This code is for microarray data, for RNASeq data the code is slightly adjusted
snpge=data.frame("ITGAL"=ge$ITGAL[!is.na(idge)],mat[idge[!is.na(idge)],],"PAS.ID"=snp.ids$id0[idge[!is.na(idge)]],
                 "PAS.SNP"=snp.ids$id[idge[!is.na(idge)]],"PAS.GE"=ge.ids$id[!is.na(idge)],"CLASS"=ge.class[!is.na(idge)])

pvalueMain <- NA
pvalueSNP <- NA
isNAvec <- NA

for(i in 1:1044){
  snp_analysis = data.frame("itgal" = snpge$ITGAL/log2(exp(1)), "snp" = snpge[i+1], "disease" = snpge$CLASS, "pas" = snpge$PAS.SNP)
  snp_analysis$disease <- relevel(snp_analysis$disease, ref = "N")
  colnames(snp_analysis)[2] = "snp"


  fit = lm(itgal~disease-1+disease:snp,data=snp_analysis)


#create constrast vectors C0 and C3
convec <- matrix(NA, ncol = 10, nrow = 2)
convec[1,1] <- 0
convec[2,1] <- 0
convec[1,6] <- 0
convec[2,6] <- 0
for (j in 1:10){
  if(is.na(coefficients(fit)[j])){
    isNAvec[i] <- 1
    convec[1,j] <- NA
    convec[2,j] <- NA
  }
  else{
    if(j == 2 | j == 4){
      convec[1,j] <- 1
      convec[2,j] <- 0
    }
    else if(j==3|j==5){
      convec[1,j] <- -1
      convec[2,j] <- 0
    }
    else if(j==7|j==9){
      convec[1,j] <- 0
      convec[2,j] <- 1
    }
```

```
    else if(j==8|j==10){
      convec[1,j] <- 0
      convec[2,j] <- -1
    }

  }
}

#Calculate p-values
AvsU <- convec[,colSums(is.na(convec)) != nrow(convec)]
AvsUcoeff=AvsU%*%coefficients(fit)[which(convec[1,]!="NA")]
AvsUvar=AvsU%*%vcov(fit)%*%t(AvsU)
WaldmainAvsU=AvsUcoeff[1]/sqrt(AvsUvar[1,1])
WaldsnpAvsU=AvsUcoeff[2]/sqrt(AvsUvar[2,2])
pvalueMain[i] <- 2*pnorm(abs(WaldmainAvsU),lower.tail=FALSE)
pvalueSNP[i]<- 2*pnorm(abs(WaldsnpAvsU),lower.tail=FALSE)
}
```

Appendix B

VCFtools

VCFtools is a program package for working with VCF files (Danecek et al., 2011). The VCF files store gene information, and VCFtools is used to filter out specific variants. In this project, we use it to get the genotyped SNPs in a specific interval and the SNP status for patients with different diseases. We also get a MAF-value, which explains how rare this different SNPs are.

The query used for getting the SNP statuses as 0,1,2 within a region is:

```
vcftools --vcf file.vcf --out ITGALout --012 --chr 16
--to-bp 30523185 --from-bp 30472658 --keep-filtered GENOTYPED
```

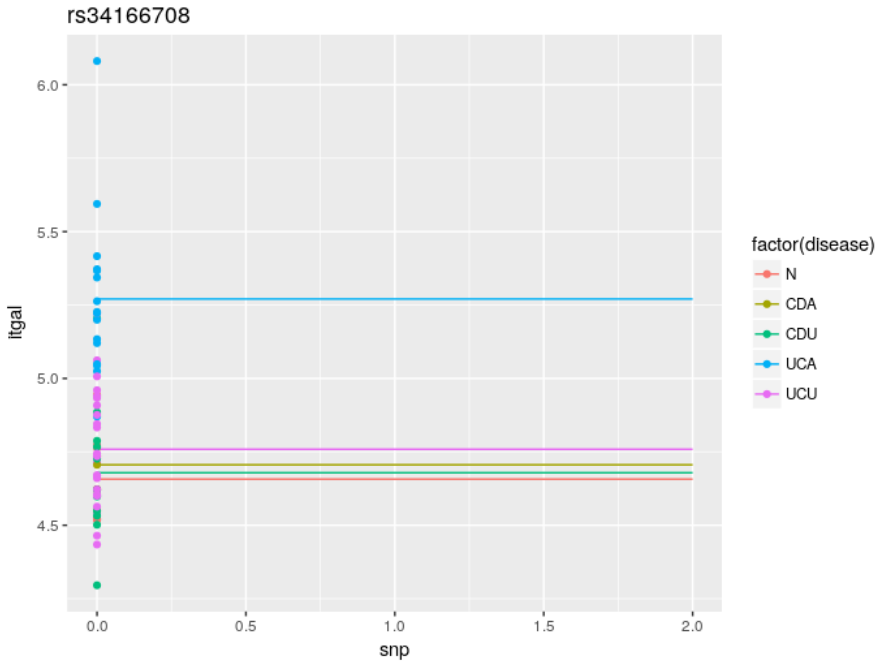
The query used for getting the MAF values is:

```
vcftools --vcf file.vcf --out ITGALoutMAF --recode
--recode-INFO-all --chr 16 --to-bp 30523185 --from-bp 30472658
--keep-filtered GENOTYPED
```

Appendix C

Results

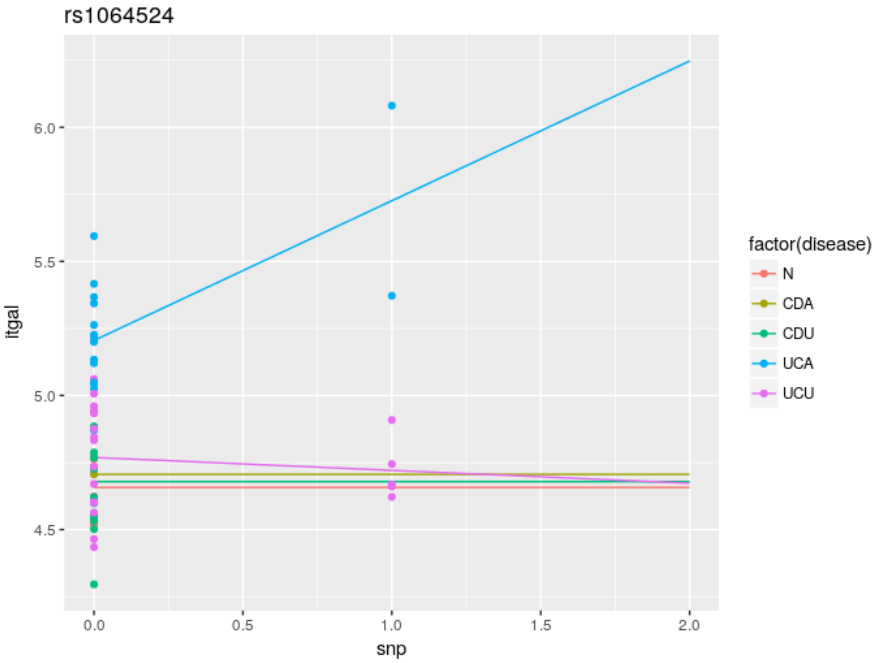
C.1 Results for microarray data



	Estimate	Std. Error	t value	Pr(> |t|)
diseaseN	4.657109	0.070681	65.888887	1.71E-55
diseaseCDA	4.705833	0.223514	21.053884	1.45E-28
diseaseCDU	4.678848	0.067392	69.427414	9.00E-57
diseaseUCA	5.270399	0.055878	94.319002	2.72E-64
diseaseUCU	4.758769	0.045625	104.302793	9.08E-67

$$W_{disease} = 0.02750116$$

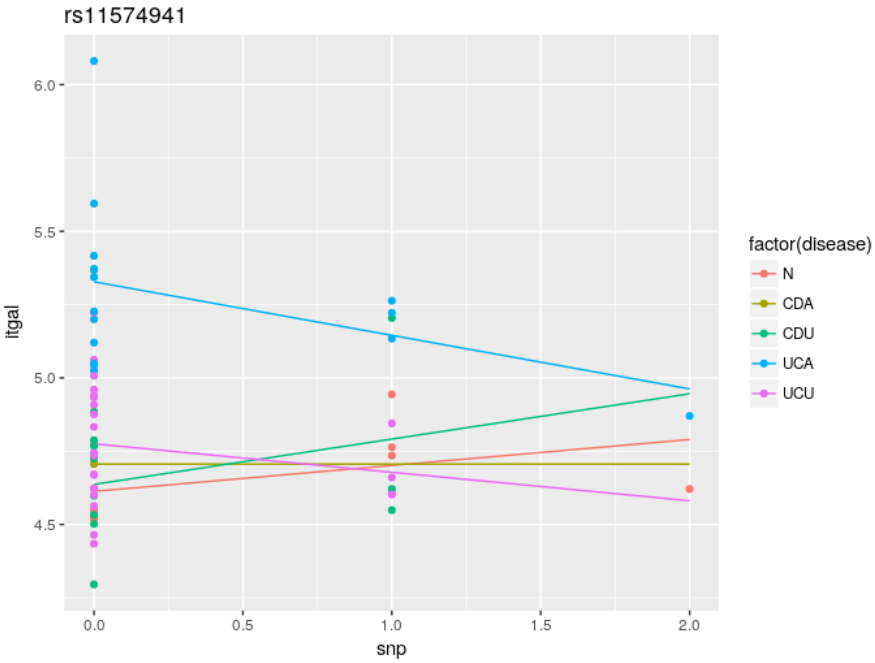
$$W_{SNPi} = \text{NA}$$



	Estimate	Std. Error	t value	Pr(> |t|)
diseaseN	4.657109	0.065551	71.045446	8.47E-56
diseaseCDA	4.705833	0.207291	22.701591	1.31E-29
diseaseCDU	4.678848	0.062501	74.860904	4.91E-57
diseaseUCA	5.205265	0.055401	93.956457	2.02E-62
diseaseUCU	4.768785	0.047556	100.277691	5.75E-64
diseaseUCA:snp	0.521074	0.156697	3.325356	1.58E-03
diseaseUCU:snp	-0.048077	0.104190	-0.461442	6.46E-01

$$W_{disease} = 0.04251931$$

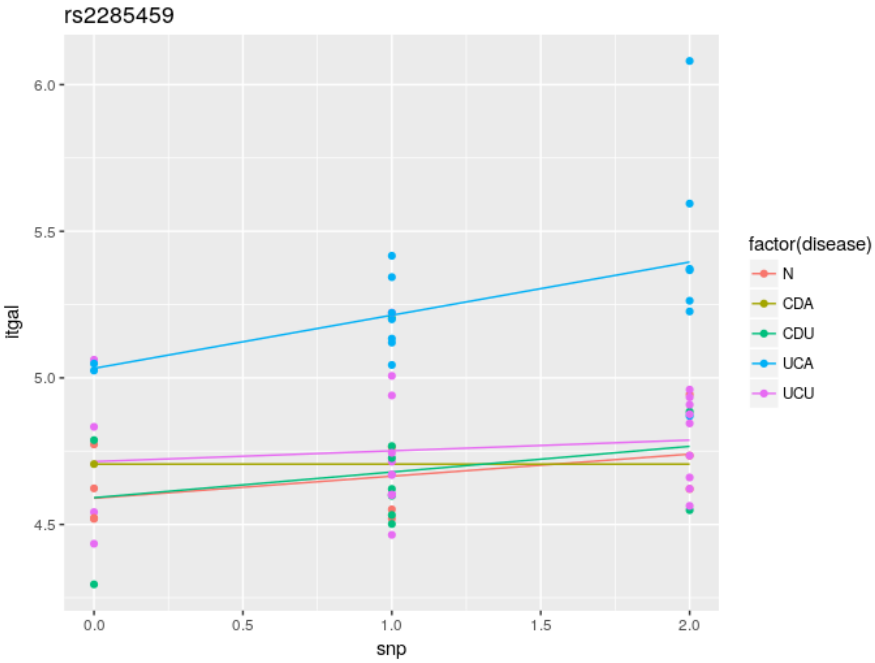
$$W_{SNPi} = 0.002489612$$



	Estimate	Std. Error	t value	Pr(> |t|)
diseaseN	4.612872	0.086465	53.349586	9.65E-48
diseaseCDA	4.705833	0.219229	21.465398	8.29E-28
diseaseCDU	4.636677	0.077509	59.821085	2.46E-50
diseaseUCA	5.327499	0.062185	85.671505	1.62E-58
diseaseUCU	4.774958	0.049021	97.406299	1.88E-61
diseaseN:snp	0.088476	0.103345	0.856115	3.96E-01
diseaseCDU:snp	0.154626	0.148419	1.041826	3.02E-01
diseaseUCA:snp	-0.182719	0.094015	-1.943506	5.73E-02
diseaseUCU:snp	-0.097139	0.120077	-0.808975	4.22E-01

$$W_{disease} = 0.01137627$$

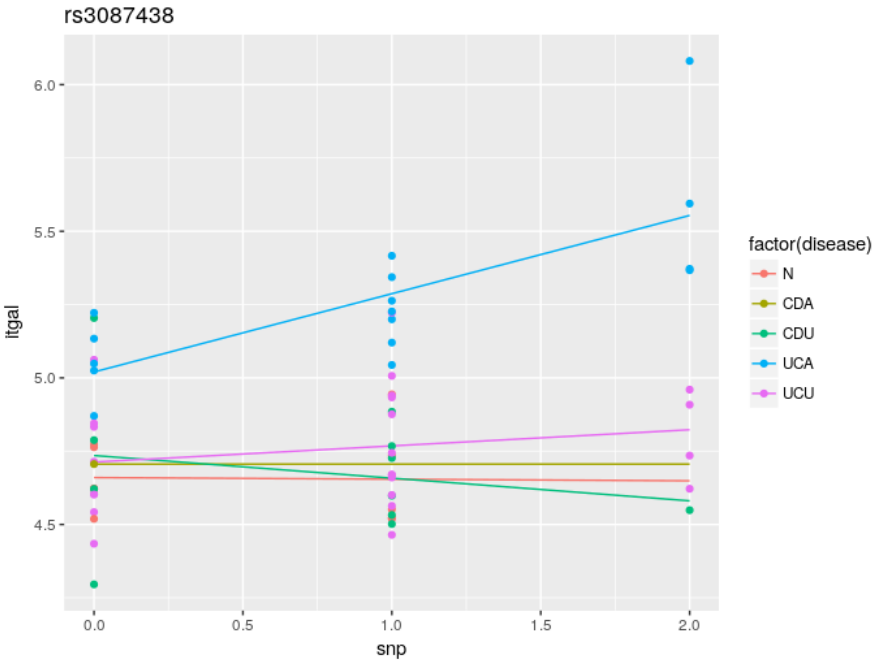
$$W_{SNPi} = 0.2589941$$



	Estimate	Std. Error	t value	Pr(> |t|)
diseaseN	4.589380	0.101618	45.163106	5.47E-44
diseaseCDA	4.705833	0.217946	21.591727	6.27E-28
diseaseCDU	4.591303	0.127253	36.080088	5.56E-39
diseaseUCA	5.032506	0.118198	42.576957	1.15E-42
diseaseUCU	4.714528	0.088198	53.454112	8.72E-48
diseaseN:snp	0.075255	0.082971	0.907009	3.69E-01
diseaseCDU:snp	0.087544	0.108973	0.803358	4.25E-01
diseaseUCA:snp	0.181252	0.079916	2.268021	2.74E-02
diseaseUCU:snp	0.036613	0.063025	0.580927	5.64E-01

$$W_{disease} = 0.1389713$$

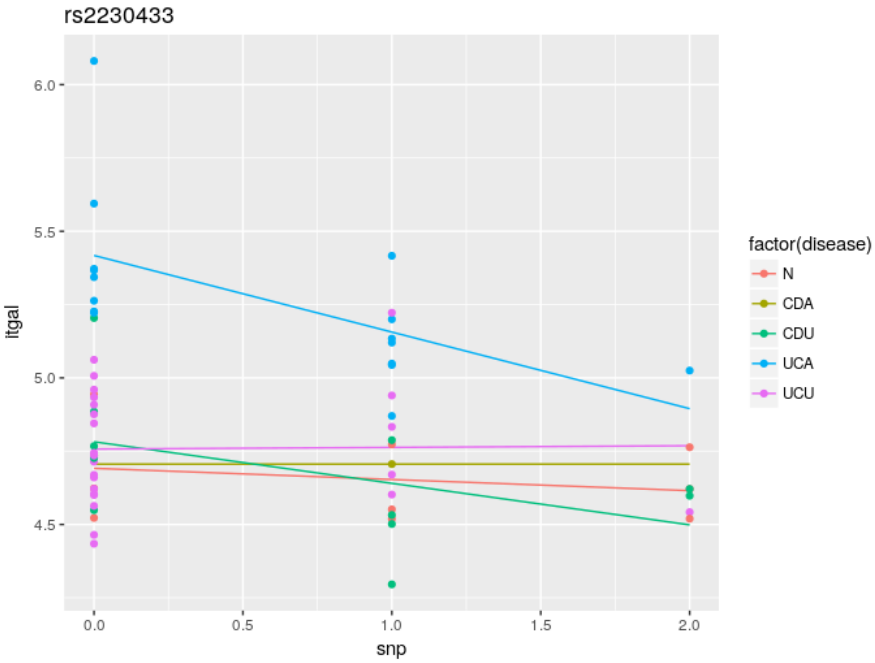
$$W_{SNPi} = 0.701793$$



	Estimate	Std. Error	t value	Pr(> |t|)
diseaseN	4.659853	0.090105	51.715849	4.87E-47
diseaseCDA	4.705833	0.201481	23.356239	1.44E-29
diseaseCDU	4.735062	0.093941	50.404698	1.85E-46
diseaseUCA	5.020173	0.080803	62.128268	3.41E-51
diseaseUCU	4.712884	0.064644	72.905223	7.81E-55
diseaseN:snp	-0.005488	0.127428	-0.043064	9.66E-01
diseaseCDU:snp	-0.077295	0.098526	-0.784510	4.36E-01
diseaseUCA:snp	0.266907	0.067395	3.960358	2.25E-04
diseaseUCU:snp	0.055061	0.059849	0.920005	3.62E-01

$$W_{disease} = 0.2568068$$

$$W_{SNPi} = 0.03036489$$

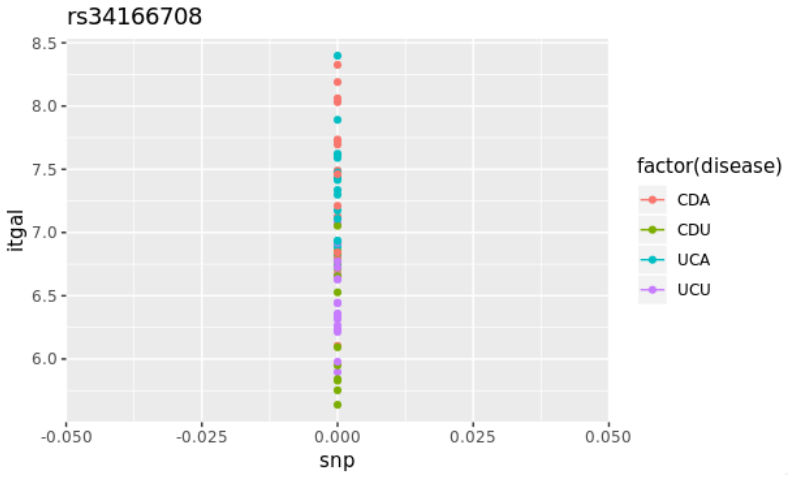


	Estimate	Std. Error	t value	Pr(> |t|)
diseaseN	4.691395	0.097297	48.217032	1.85E-45
diseaseCDA	4.705833	0.208680	22.550491	7.83E-29
diseaseCDU	4.781888	0.087663	54.548487	3.03E-48
diseaseUCA	5.417362	0.071009	76.290912	7.20E-56
diseaseUCU	4.757113	0.048444	98.197405	1.22E-61
diseaseN:snp	-0.038096	0.079443	-0.479533	6.34E-01
diseaseCDU:snp	-0.141680	0.083931	-1.688049	9.73E-02
diseaseUCA:snp	-0.261267	0.085640	-3.050748	3.56E-03
diseaseUCU:snp	0.005675	0.079109	0.071732	9.43E-01

$$W_{disease} = 0.01582864$$

$$W_{SNPi} = 0.3832292$$

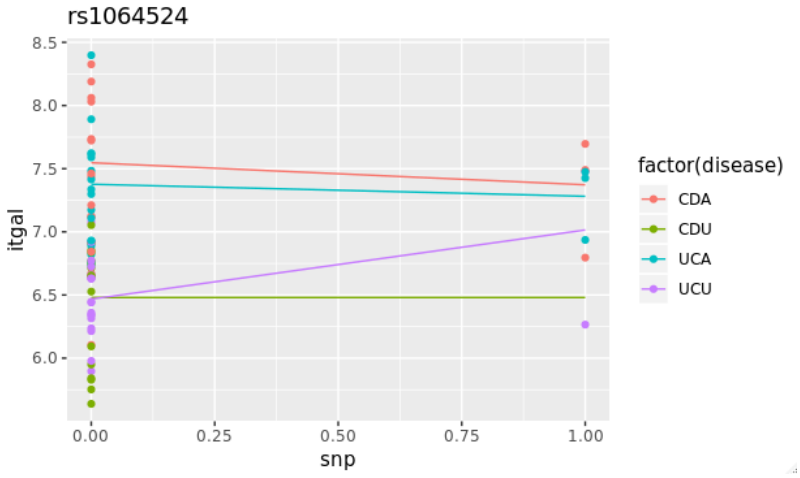
C.2 Results for RNASeq data



	Estimate	Std. Error	z value	Pr(> |z|)
diseaseCDA	7.519584	0.110749	67.897675	0.00E+00
diseaseCDU	6.480014	0.114212	56.736856	0.00E+00
diseaseUCA	7.362919	0.105090	70.062676	0.00E+00
diseaseUCU	6.538164	0.105352	62.060009	0.00E+00

$$W_{disease} = 6.73849E - 16$$

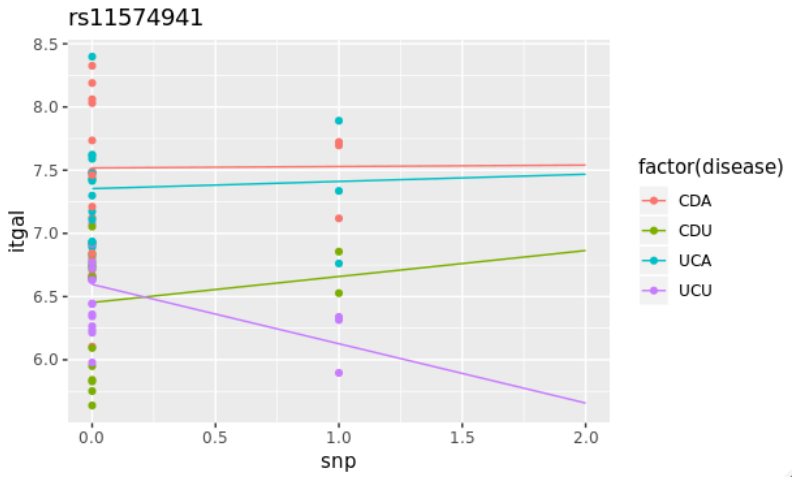
$$W_{SNPi} = \text{NA}$$



	Estimate	Std. Error	z value	Pr(> |z|)
diseaseCDA	7.546609	0.121316	62.206108	0.00E+00
diseaseCDU	6.480014	0.114212	56.736853	0.00E+00
diseaseUCA	7.376645	0.113986	64.715262	0.00E+00
diseaseUCU	6.467760	0.111090	58.220940	0.00E+00
diseaseCDA:snp	-0.174317	0.297205	-0.586521	5.58E-01
diseaseUCA:snp	-0.095268	0.294318	-0.323692	7.46E-01
diseaseUCU:snp	0.547009	0.350611	1.560160	1.19E-01

$$W_{disease} = 4.501684E - 17$$

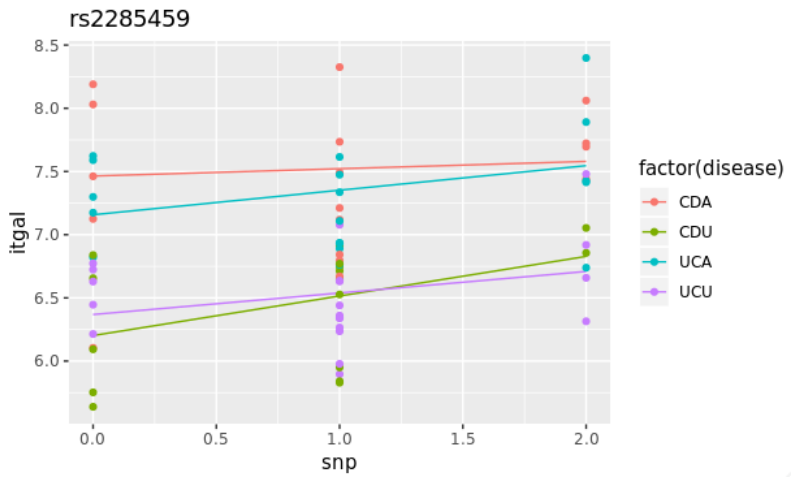
$$W_{SNPi} = 0.1427063$$



	Estimate	Std. Error	z value	Pr(> |z|)
diseaseCDA	7.517705	0.121319	61.966235	0.00E+00
diseaseCDU	6.453529	0.121603	53.070317	0.00E+00
diseaseUCA	7.354198	0.113993	64.514271	0.00E+00
diseaseUCU	6.595984	0.114246	57.734711	0.00E+00
diseaseCDA:snp	0.011221	0.297169	0.037761	9.70E-01
diseaseCDU:snp	0.205417	0.354245	0.579873	5.62E-01
diseaseUCA:snp	0.056723	0.294236	0.192782	8.47E-01
diseaseUCU:snp	-0.469303	0.295516	-1.588080	1.12E-01

$$W_{disease} = 5.091212E - 13$$

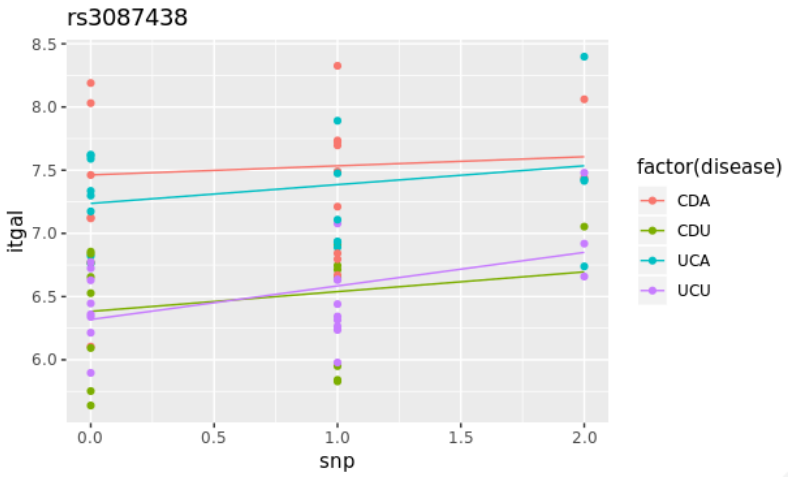
$$W_{SNPi} = 0.6185999$$



	Estimate	Std. Error	z value	Pr(> |z|)
diseaseCDA	7.464120	0.185182	40.306983	0.00E+00
diseaseCDU	6.202188	0.190655	32.530903	3.90E-232
diseaseUCA	7.157768	0.182094	39.308119	0.00E+00
diseaseUCU	6.368517	0.183214	34.759928	9.81E-265
diseaseCDA:snp	0.057709	0.157115	0.367307	7.13E-01
diseaseCDU:snp	0.313053	0.185146	1.690843	9.09E-02
diseaseUCA:snp	0.194090	0.148644	1.305739	1.92E-01
diseaseUCU:snp	0.170348	0.157602	1.080875	2.80E-01

$$W_{disease} = 4.897462E - 08$$

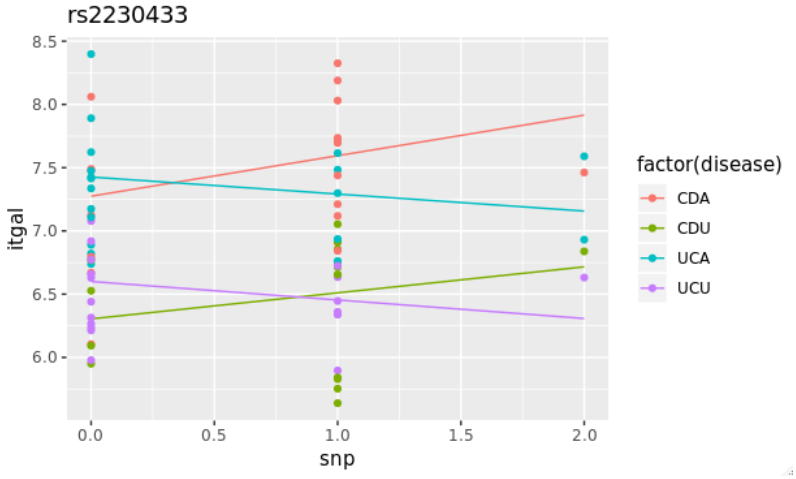
$$W_{SNPi} = 0.4829886$$



	Estimate	Std. Error	z value	Pr(> |z|)
diseaseCDA	7.462710	0.176223	42.348032	0.00E+00
diseaseCDU	6.383177	0.160014	39.891322	0.00E+00
diseaseUCA	7.237075	0.153867	47.034521	0.00E+00
diseaseUCU	6.319044	0.154771	40.828292	0.00E+00
diseaseCDA:snp	0.071664	0.176197	0.406726	6.84E-01
diseaseCDU:snp	0.156032	0.190356	0.819683	4.12E-01
diseaseUCA:snp	0.148711	0.140435	1.058927	2.90E-01
diseaseUCU:snp	0.265375	0.150886	1.758774	7.86E-02

$$W_{disease} = 7.031259E - 10$$

$$W_{SNPi} = 0.5453224$$



	Estimate	Std. Error	z value	Pr(> |z|)
diseaseCDA	7.274596	0.181127	40.163044	0.00E+00
diseaseCDU	6.304883	0.216202	29.161993	5.89E-187
diseaseUCA	7.426022	0.131058	56.662155	0.00E+00
diseaseUCU	6.600107	0.132539	49.797608	0.00E+00
diseaseCDA:snp	0.320317	0.198369	1.614754	1.06E-01
diseaseCDU:snp	0.205647	0.222765	0.923158	3.56E-01
diseaseUCA:snp	-0.134593	0.156660	-0.859143	3.90E-01
diseaseUCU:snp	-0.146292	0.178719	-0.818555	4.13E-01

$$W_{disease} = 4.50224E - 07$$

$$W_{SNPi} = 0.7529079$$