eQTL Detector; an automated pipeline for eQTL mapping and downstream analysis

Diego M. Martínez R.

6/24/2020

Título del trabajo:	eQTL Detector; an automated pipeline for eQTL mapping and downstream analysis
Nombre del autor:	Diego Maximiliano Martínez Rosales
Nombre del consultor/a:	Diego Garrido Martín
Nombre del PRA:	Diego Garrido Martín
Fecha de entrega (mm/aaaa):	Junio/2020
Titulación:	Máster en Bioinformática y Bioestadística
Área del Trabajo Final:	Area 3. Subarea 9: QTL mapping: genetic effects on gene expression and alternative splicing
Idioma del trabajo:	Inglés
Palabras clave	GWAS, QTL, Pipeline, Docker

1 Abstract

Access to both DNA and RNA genotyping technologies have opened a new paradigm in the mapping of complex characters, these are based on the evaluation of the interaction of the genotype and other variables such as methylations, proteins or levels of gene expression, in relation to this last variable, expression QTLs (eQTL) try to determine the level of expression of each gene based on each of the variants of the organisms in whole genome basis, the variants can be close to the gene of interest, in a range of 1Mb, *cis*-eQTL, or distant in a range greater than 1 Mb, called *trans*-eQTL. The amount of data necessary to perform this type of studies requires the use of tools that are easy to implement and that guarantee reproducible results. The use of containers in bioinformatics has been one of the solutions to this problem, then Docker is one of the most widely used. On the other hand, different tools that can map eQTL have existed for more than a decade, but there is one that stands out above the others, QTLtools, an application easy to implement, efficient in the use of hardware and also contains different modules that goes from quality control of samples to the co-localization of eQTL with GWAS hits. In this work we present eQTL_Detector, an automated pipeline that uses the QTLtools modules as a base and integrates them into a container that generates a report using the statistical software R and R Markdown functionalities for the creation of the report. In addition, in this work we describes each of the steps of this pipeline describing the inputs and outputs for each step together with their statistical foundations.

2 Resumen

El acceso a tecnologías de genotipado tanto de ADN como ARN han abierto una nuevo paradigma en el mapeo de caracteres complejos, estas están basadas en la evaluación de la interacción del genotipo y otras variables como metilaciones, proteínas o niveles de expresión genética, en relación a esta última variable, los denominados QTL de expresión (eQTL) tratan de determinar el nivel de expresión de cada gen en función de cada una de las variantes del genoma del organismo, las variantes pueden ser cercanas al gen de interés, en un rango de 1Mb *cis*-eQTL, o distantes en un rango mayor a 1 Mb , denominados *trans*-eQTL. La cantidad de datos necesarios para realizar este tipo de estudios requiere del uso de herramientas que sean fáciles de implementar y que garanticen resultados reproducibles, el uso de contenedores en bioinformática ha sido una de las soluciones a este problema, siendo Docker uno de los más utilizados. Por otra parte, diferentes herramientas que tienen la capacidad de realizar mapeo de eQTL existen desde hace más de una década, pero existe una que sobresale sobre sobre las otras, QTLtools, una aplicación fácil de implementar, eficiente en el uso del hardware y además contiene diferentes módulos que van desde el control de calidad de las muestras hasta la co-localización de eQTL con hits de GWAS. En este trabajo presentamos eQTL_Detector un pipeline automatizado que utiliza como base los módulos de QTLtools y los integra en un contenedor que genera un reporte utilizando el software estadístico R y funcionalidades de R Markdown para la creación del reporte, además en el presente trabajo se describen cada uno de los pasos de este pipeline describiendo las entradas y salidas de cada paso junto con los fundamentos estadísticos de estos.

Contents

List of Figures

List of Tables

3 Introduction

Life and biological science are experiencing a *data tsunami* that is changing the way research is conducted, researchers are dealing with large amount of data provided from different sources and data analysis by integrating different techniques and tools can be challenge [40],expression quantitative trait locus analysis (eQTL) is one of the techniques that integrate the use of genotyping data (GBS) and mRNA expression data to describe the phenotype variations, the idea of this analysis is to use genotypes as a predictor of the expression levels, then the process of identify eQTL's ends with certain variable regions highly correlated with the expression levels, those regions can be annotated as a single nucleotide polymorphism (SNPs) or variants [34].

Is important to mention that genome wide association studies (GWAS) can also detect SNP's highly associated with complex traits,in fact this method was developed before [42, 24] eQTL mapping as a reult of the human genome project[25], but GWAS has shown that in most of the cases the trait associated to variants are commonly located on non-coding regions [30, 33] and almost 1 third of GWAS hits are described as eQTL's [44]

Regarding the mode of action of eQTL's, the use of 1± Megabase of genomic sequencing is a common practice to separate two type of eQTL's effect [19], then as definition, a *cis*-eQTL are the ones with a distance of 1± Megabase from the gene and *trans*-eQTL are the ones outside of that window or even farther, in another chromosome. eQTL analysis can be applied to different scenarios, from controlled populations with few recombinant events, like in plant breeding [12], or large heterozygous populations for diseases [14], these two types of experiments design refer to two different analysis approach, *linkage-based* eQTL mapping and *association-based* eQTL mapping respectively, both approaches have advantages and disadvantages, in the case of the linkage-based technique, as the strategy utilize inbred strains to map eQTL's with few recombinant regions, the result may suffer by missing some common variant not present on the selected inbred used for the study [19], in the case of association-base studies, they required large populations sizes that increase computer requirements and costs Nodzak [34].

3.0.1 eQTL mapping and methods

The most common model used for eQTL analysis is the lineal regression [39, 35], where each gene SNP's are encoded as 0,1 and 2 according to their alleles frequency,

then the gene expression g is predicted by:

$$g = \alpha + \beta s + \epsilon$$

Where $\epsilon \sim N(0, \sigma^2)$, β will be the slope and s the estimated value for each call, for α and s the Best Linear Unbiased Estimator (BLUE) is searched, on the line fitted the R^2 value gives the goodness-of-fit of this line that is going from 0 to 1 depending on how well the line fit on the real values, on this procedure the hypothesis test to perform will be:

$$H_0 : \beta = 0$$

$$H_1 : \beta \neq 0$$

A t-test can be performed to evaluate this hypothesis, then if the predicted lines have an slope close to zero then line will be horizontal, meaning that the values of the gene expression will not depend on the different allele frequencies, in the contrary case, when we reject the null hypothesis it mean that the values of gene expression will depend on the allele frequencies. There is a direct relationship with lineal regression and the Pearson correlation of the data (ρ), where the goodness-of-fit value (R^2) is equal to the parson correlation between g and s, called by $\rho_{g,s}$.

$$\rho_{g,s} = \frac{cov(g, s)}{\sigma_g \sigma_s}$$

The Pearson correlation can be also tested as:

$$H_0 : \rho_{g,s} = 0$$

$$H_1 : \rho_{g,s} \neq 0$$

A common practice in eQTL analysis is to use covariates to see population stratification like gender, age or any variable related with the population under study, a simple way to add that into the model is by using:

$$g = \alpha + yx + \beta s + \epsilon$$

3.1 Context and work justification

eQTL studies start from obtaining whole genome sequences and RNA-Seq data from populations, both type of data needs to be manipulated using terminal environment, those manipulations can include, data quality of sequences (QC) and alignment with the reference genome, from the whole genome sequence *variant calls* can be obtained for each sample, this variant calls files include the SNP's detected with their physical location obtained from the reference genome, from the RNA-Seq data, a gene expression quantification for each sample and gene is obtained, after that, a data normalization of the expression values is performed to avoid any known or unknown artifacts in the data [2]. With the expression values and the SNP matrix for each sample the eQTL analysis can be performed, many tools are available for eQTL detection (Table 1.1) and most of them are implemented in R [36] and the other ones are developed to be used on the command line.

Table 3.1: List of tools available for eQTL analysis

Tool	Year of release	Platform
Merlin	2007	Command Line
Pseudomarker	2008	Matlab
snpMatrix	2009	R
eMap	2010	R
J/qtl	2013	GUI
MapQTL	2013	MS-Windows
GridQTL	2013	Browser
QTLMap	2013	Command Line
Matrix eQTL	2014	R
PLINK	2014	Command Line
FastQTL	2015	Command Line
R/qtl	2016	R
QTLtools	2016	Command Line

3.2 Objectives

3.2.1 General Objectives

- Develop an automated pipeline for eQTL mapping and downstream analysis.
- The pipeline must run through a Docker container in order to generate repro-

ducible results and make it easy to implement.

- The pipeline must generate a report with the intermedia results for each step.

3.2.2 Specifics Objectives

- The amount of coding steps for the end-user must be reduce to the minimum or order to make the end product more accessible.
- The pipeline must be publish in Git-hub to keep track of the versions and check updates.
- The Report components will be:
  - QC on RNA-Sequence
  - Check on sequence and genotyping match
  - Quantification of gene expression
  - PCA analysis results for genotyping and gene expression
  - *cis* _eQTL mapping using nominal pass
  - *cis* _eQTL mapping using permutation pass
  - *trans* eQTL mapping using nominal pass
  - *trans* eQTL mapping using permutation pass
  - Density analysis
  - Enrichment analysis
  - Co-localization with GWAS hits

3.3 Focus and methodoloy

3.3.1 About the pipeline components and the development process

Today many bioinformatic analyses implemented as pipelines are published on web services where end users can get access to powerful tools with well established work flows, these analyses can be manage in all terms by the provider of the service, EMBL-EBI [29] (https://www.ebi.ac.uk/),so data sources (sequences and references) and analysis the *perse*, and services where the end user can upload their own sequence data and run multiple available analysis or setup their own pipeline, Galaxy [3] (https://usegalaxy.org/).

Containers that include all the required dependencies for a analysis are currently used in order to increase the reproducibility and simplify installation and configuration steps, normally bioinformatic analyses required more than 15 tools and at the same time these tools require their own dependencies [11], one of the most popular

implementation software for containerization of bioinformatic pipelines is Docker [31], Docker offer to developers the possibility to setup a configuration file (Dockerfile) which include a human readable instructions to install all the required dependencies, this Dockerfile can be moved from one machine to another and implement the same process without install any other software [15], containers images normally doesn't include any dependencies and the process to install all the required tools may take some time, to avoid wasting time on compiling containers, Docker offer the Docker-hub (https://hub.docker.com/), a repository to upload to upload containers associated to a Dockerfile with the instructions to deploy specific environments with the required dependencies, apart from the required tools that a container must have, some other configurations on the container or the way it works can be administrated by using Docker-Compose [17], Docker-Compose consist in a .YML file that contain all the configurations, like enabling ports between the host machine and the container machine.

The Rocker Project [7] is a project that has publish in Docker hub different type of containers with the statistical software R [36], one of the container of this project has publish is **Rstudio** (https://hub.docker.com/r/rocker/rstudio/),this container include an R Studio Server [37] configured that can be accessed by enabling a port, this R Studio versions include R Markdown [5, 43], a powerful tool for create reproducible reports in R.

Bash (Bourne-again shell) is the Unix shell language developed by Brian Fox and the GNU project, this language can be used to build Bash scripts `.sh` that is normally used in combination with other application, like AWK [4], to create pipelines.

QTLtools [10] is a powerful tools for eQTL analysis that include some other functionalities like QC on sequence data and some downstream analysis, as this tool works on command line environment (Table 1.1) the development of pipelines using BASH scripts can be more easy and reproducible, also we need consider that in some cases an output from one step is the input for another, then using only one tool can avoid to have issues on recognizing formats.

3.3.2 Pipeline developement

eQTL_Detector was developed using two different containers, one container (eQTL_Detector_Base https://github.com/diegommartinezr/eQTL_Detector_Base) that contain the necessary tools and their dependencies, including R packages, and another container (eQTL_Detector https://github.com/diegommartinezr/eQTL_Detector) that contain the

`.sh` script that run the workflow and the `.rmd` which is the R markdown script that compile a report (Figure 1.1).

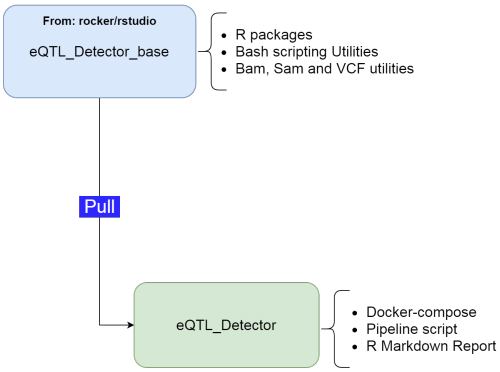

Figure 3.1: In eQTL_Detector the use of a base container that include all the required dependencies and R packages help on reduce the time on building the container for testing and development.

3.4 Workplan

This work was organized for a period of one academic semester, then the development of the pipeline was divided in 3 steps and one more for writing a report, the steps consist in one for the container design, which include the revision and decision of the elements (tools and R packages) that will be included in the containers, the other two steps are related to: the BASH scripting, mainly for QTLtools and file formats, and finaly the R report coding in R Markdown.

Table 3.2: Milesontes for the pipeline developmet, including the number of days dedicated

Milestone Description	Category	Start	No. Days
Container Design	*	NA	*
Define Pipeline	Low Risk	2020-03-17	3
List of dependencies	Low Risk	2020-03-20	2
List of R packages	Low Risk	2020-03-22	2
Design workflow	Low Risk	2020-03-24	4
	*	NA	*
Build Container I	*	NA	*
Create a GitHub Repository	High Risk	2020-03-28	1
Build a Docker container	High Risk	2020-03-29	2
Install dependecies on Docker Container	High Risk	2020-03-31	5
Create BASH scripts	High Risk	2020-04-05	10
Test BASH scripts	High Risk	2020-04-15	7
Deliver first report	Milestone	2020-04-22	1
	*	NA	*
Build Container II	*	NA	*
Create R scripts	Med Risk	2020-04-23	7
Test R scripts	Med Risk	2020-04-30	7
Test all components (interaction)	High Risk	2020-05-07	7
Test complete pipeline	High Risk	2020-05-14	10
Deliver second report	Milestone	2020-05-24	1
	*	NA	*
Documents	*	NA	*
Documentation I	Low Risk	2020-06-02	10
Documentation I Review	Low Risk	2020-06-12	2
Documentation II	Low Risk	2020-06-14	10

Milestone Description	Category	Start	No. Days
Documentation II Review	Low Risk	2020-06-24	4
Presentation	Low Risk	2020-06-28	3

3.5 Brief summary of the obtained product

eQTL_Detector is a pipeline that use as a main tool QTLtools [10] for eQTL mapping, the pipeline run the whole set of functionalities integrated in QTLtools [10] and it takes all the the outputs to build a report, the pipeline is implemented by using a Docker container that included a RStudio Server and the end user just need to provide the RNA-Seq data, the genotyping data and the annotation reference, this last part make this pipeline useful to work with different species. On the Figure 1.2 we can see how the pipeline work on *cis*-eQTL mapping and how some outputs from one step are the input in others.

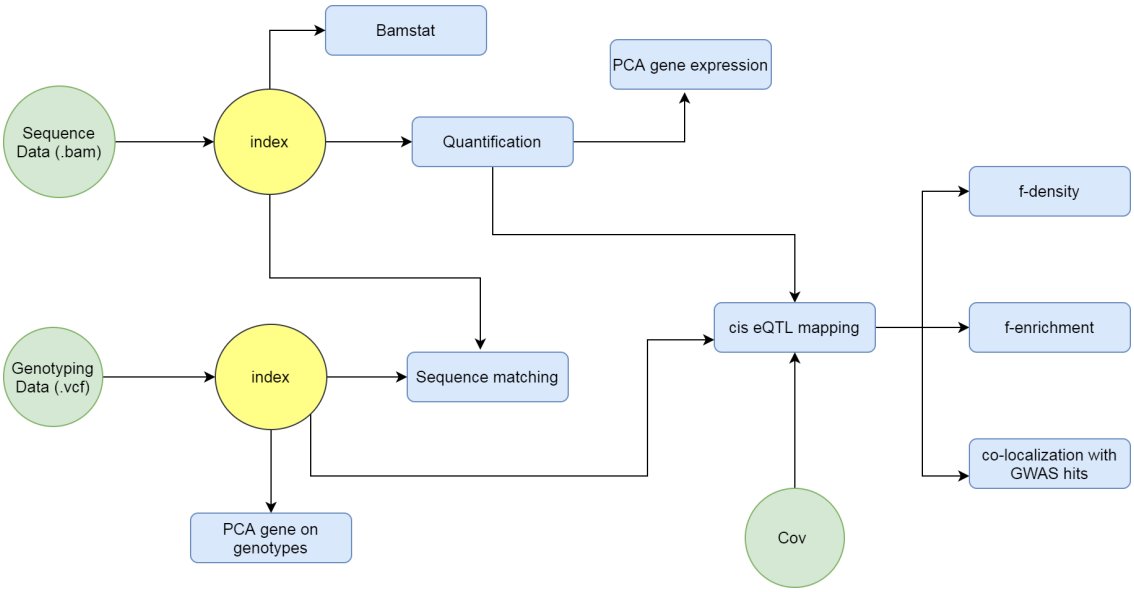

Figure 3.2: eQTL_Detector pipeline for cis eQTL mapping, in green circles the input data required, in blue squared boxes the steps performed by QTLtools and in yellow the circles the indexing steps for sequences and genotyping data.

3.6 Brief description of the chapters of this memory

The follow chapters will describe each of the steps that the pipeline execute with an explanation on the statistical methods behind, also we will explain all the outputs that the pipeline give to end user for each step.

The data used to test the pipeline are two, one set was used to check indexing, sequence QC, matching, quantification and PCA, this data consist in 39 samples randomly picked from the 1000 Genome Browser Portal (https://www.internationalgenome.org/1000-genomes-browsers/), for the case of RNA-seq data we use the Geuvadis consortium [**Lappalainen2013**] and the genotype data from the 1000 Genomes Project Consortium [1], the script that download, align and filter the RNA-Seq data on the chromosome 22 is available here https://github.com/diegommartinezr/eQTL_Detector/blob/master/GetSamples.sh. A second set was used to test; *cis* nominal, *cis* permutation pass, *trans* full pass, fdensity, fenrich and RTC, this set was obtained from the QTLtools git repository (https://qtltools.github.io/qtltools/) and consist in a set of 358 samples for genotyping and expression data from the chromosome 22.

4 Indexing VCF and BAM files

To ensure the pipeline will run without problems due issues with data format, eQTL_Detector will create the required index for VCF and BAM files, the indexing process will run before any analysis. For indexing RNA-Seq data (BAM) we use SAMtools [28], a sequence alignment tools that can index sequence data for BAM and SAM formats. For VCF we use Tabix [27], Tabix will create the index file for the VCF and with this procedure all the index files required for the follow steps will in the same folder that contain the data, while the pipeline is running the end user can access to the specific folder and check if there are `.bai` files with the sames names as the sequence and genotyping data.

5 Sequence QC [bamstat]

The first step in the pipeline is the quality control of the sequence data `[bamstat]`, QTLtools takes each of the BAM files for each sample and perform a counting of the number of alignment that pass the follow list of criteria:

- The alignment is not tagged as unmapped
- The alignment is not tagged as a secondary alignment
- The alignment is not tagged as "*failing QC*"
- The alignment has a mapping quality above the threshold

Then QTLtools will use an annotation file (provided from the end user) to check the number of reads falling in a know annotation,the preferred source for the annotation

file can be, in the case of RNA-seq, from ENCODE [9]. The output generated will include a table with the QC data for each sample with the follow fields:

- A. Total number of sequencing reads.
- B. Number of mapped sequence that pass the filter quality (default value used).
- C. Number of mapped sequences falling within the annotations specified with.
- D. Total number of annotations.
- E. Number of annotations covered by one or more sequence.

With each of the QC outputs that QTLtools create for each sample, eQTL Detector consolidate all of them in order to generate some visualization of all samples together .The Figure 3.1 show the values in a barplot for each column and the Figure 3.2 the stacked barplot for each samples, to evaluate if the QC looks similar across samples, the author also recommend to evaluate the follow relations:

$$\frac{B}{A} ; \frac{C}{A} ; \frac{E}{D}$$

The first and the second terms are making reference to the number of sequence mapped passing the QC threshold and the reads falling in a annotation over the total number of sequences reads, the third terms makes reference to number of annotations covered for at least one read over all annotations, to simplify this a plot with all these relations are included on the output of bmastat (Figure 3.3).

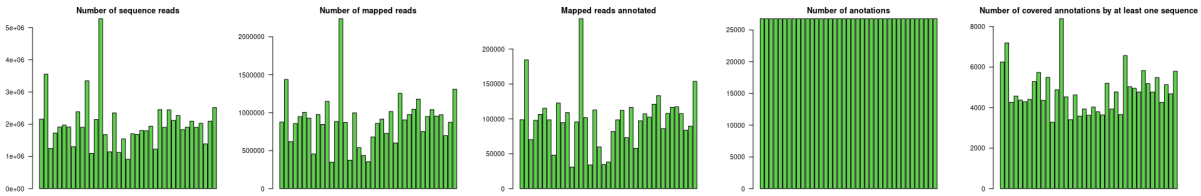

Figure 5.1: **eQTL Detector bamstat output 1:** from left to rigth, number of sequence reads, number of mapped reads, mappped reads annotated, number of anotations, number of covered annotations by at least one sequence.

6 Sequence to genotype matching [mbv]

The sequence genotype matching analysis [16] is performed to check if there are mislabeling or contamination issues, the analysis is tacking the VCF file with the genotyping data and a BAM file from one sample, then comparing the mapped

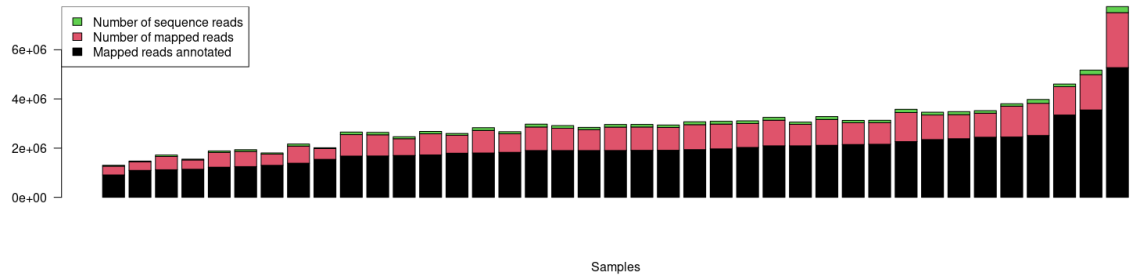

Figure 5.2: **eQTL Detectorbamstat output 2:** for all samples analyzed, the number os sequence reads (green), number of mapped reads (red) and the number of mapped reads annotated.

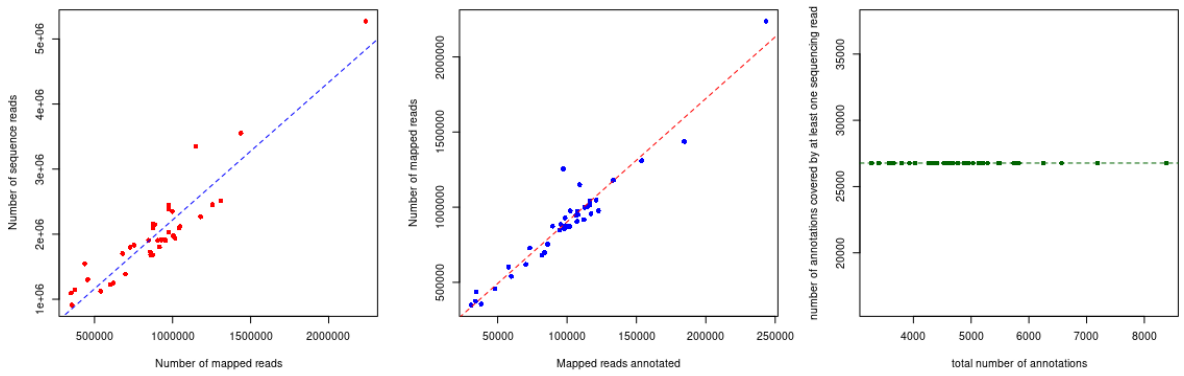

Figure 5.3: **eQTL Detectorbamstat output 3:** B/A relation for all samples (left), C/A relation for all samples (midle) and E/D realtion for all samples (rigth).

regions of the BAM files and the concordance of the variants from the VCF the poorly covered variants are discarded and an estimation of the heterozygous level are calculated, then calculate (from complementary notes of [10]):

- A. The number of homozygous genotypes REF/REF covered by the reads carrying only the reference allele.
- B. The number of homozygous genotypes REF/REF covered by at least one read carrying only the alternative allele.
- C. The number of homozygous genotypes ALT/ALT covered by the reads carrying only the alternative allele..
- D. The number of homozygous genotypes ALT/ALT covered by tat least one read carrying only the reference allele.
- E. The number of heterozygous genotypes REF/ALT covered by reads carrying either the REF or ATL alleles.
- F. The number of heterozygous genotypes REF/ALT covered by reads carrying REF **or** ATL alleles.

Here REF = Reference Genome and ALT = alternative allele, with this QTLtools calculate the concordance at homozygous (C_0) by using:

$$C_0 = \frac{A + B}{A + B + C + D}$$

And at heterozygous (C_1) by using:

$$C_1 = \frac{E}{E + F}$$

Finally with C_0 and C_1 eQTL Detector report make an scatter plot (Figure 4.1) for each sample, compared again the others included in the VCF file, a file for each sample is stored on the mbv folder in the Result directory.

7 Quantification of gene expression [quan]

QTLtools include a program that quantify gene expression from RNA-Seq BAM files using a gene annotation file (GTF format), QTLtools quantify gene expression for each exon by counting the reads that fall on each region, then it generate BED files with row data (counts) and counts per kilobase per million reads (RPKM), then the list of outputs for each sample will be:

- Exon counts

- Exon RPKM
- Gene counts
- Gene RPKM

Then eQTL Detector takes the RPKM for genes and each samples and merge them to create the expression matrix that will be used on the follow steps, a visualization of all samples gene expression is presented in the report (Figure 5.1).

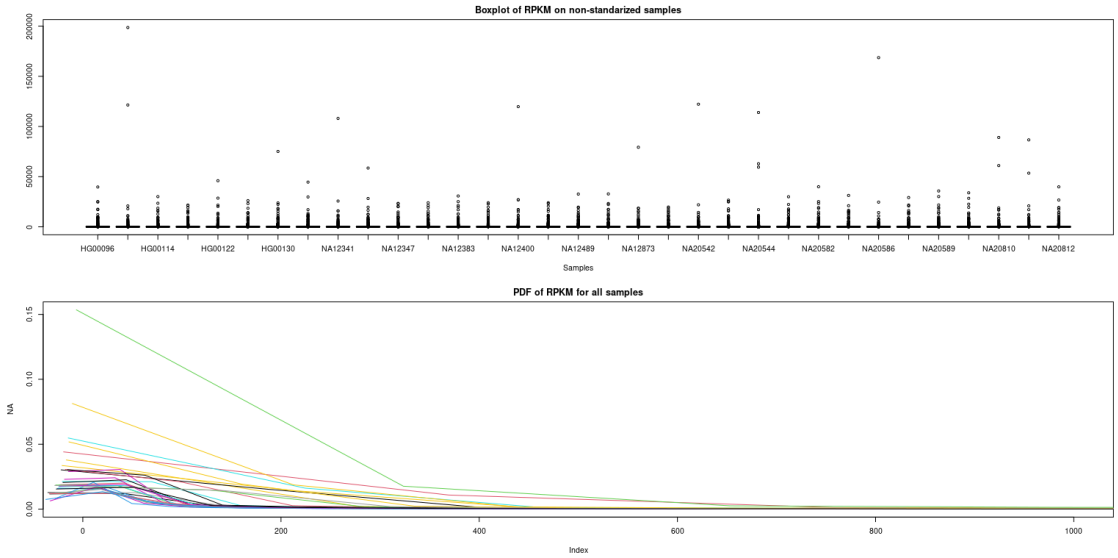

Figure 7.1: **eQTL Detector Quantification output: With all the expression data merged eQTL Detector create some plots to see if there are some outliers on the data, using linear scale and logaritmit scale**

8 Principal component analysis [pca]

Study populations stratification is an important step in any QTL study, normally this analysis is performed before any other analysis as a check step, principal component analysis (PCA) is a well established method to evaluate stratification or any structure that need to be capture and taking in count for any further step [23]. QTLtools include this analysis for gene expression and genotype data based on variants. PCA analysis on genotype data is normally used to see if there is any population structure that can enforce a QTL hits due differences on allele frequencies, it means that if there are two distant population (different co-founders) the alleles frequencies will be because of different co-founder and not because a of QTL, then correct mapped QTL need to be across both populations (). PCA on gene expression is used to capture technical variance, then the first component is used to eliminate outliers

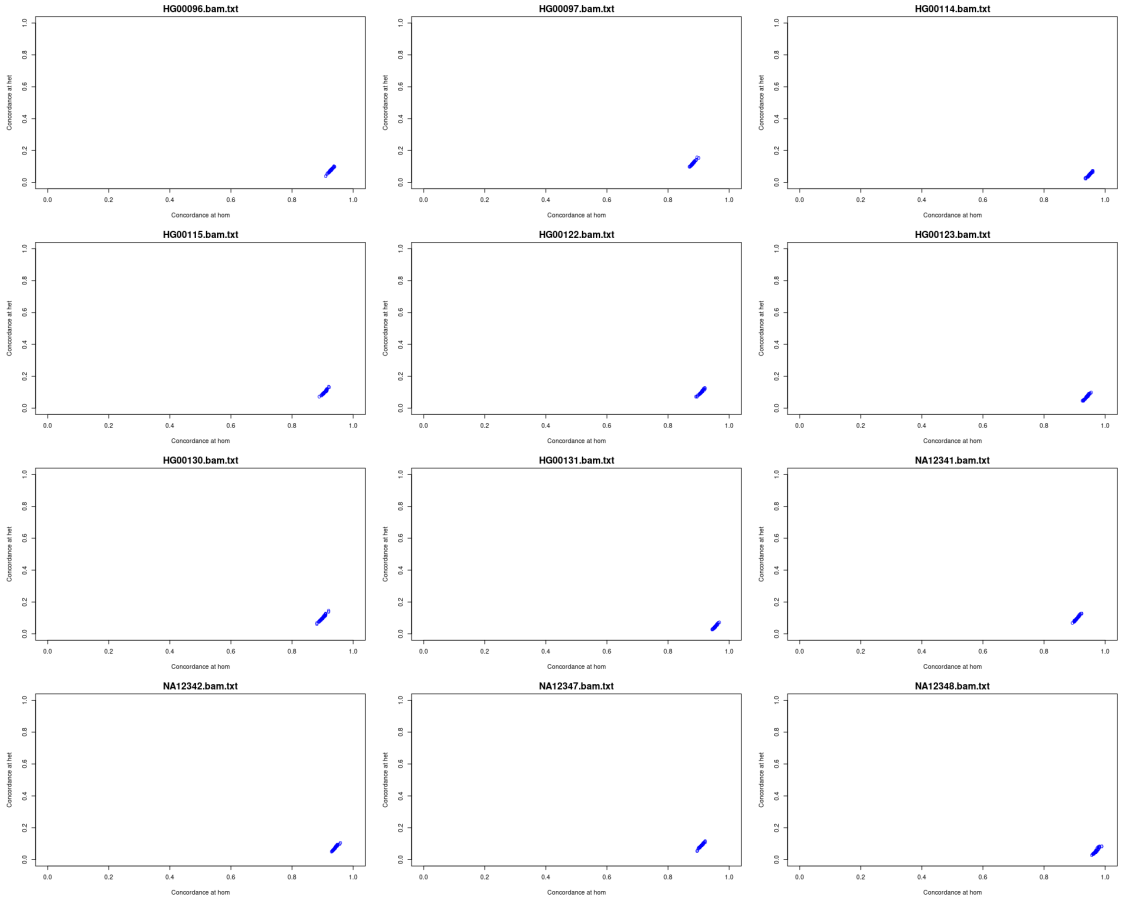

Figure 7.2: **eQTL Detector mbv output:** for aeach sample the estiamtion of the match at heterozygous and homozygous, the axis on these plots are fixed from 0 to 1 in order to compare the distribution of the values for all samples.

(Figure 6.1 and 6.2)

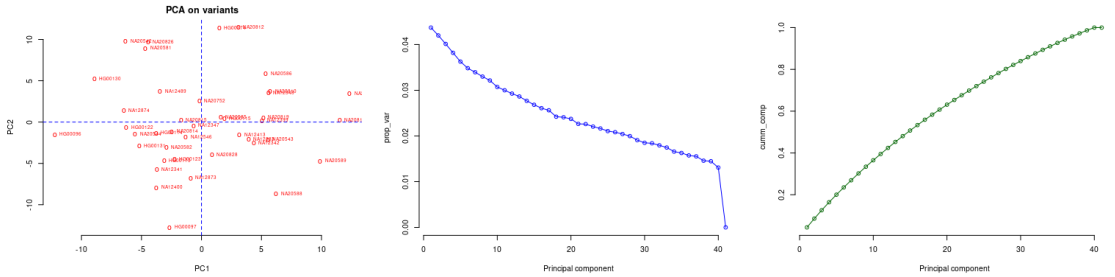

Figure 8.1: **eQTL Detector PCA on variants output:** first and second component for all samples (left), variability explanation for all compoenents (midle) and accumulated explained variability for each component (rigth).

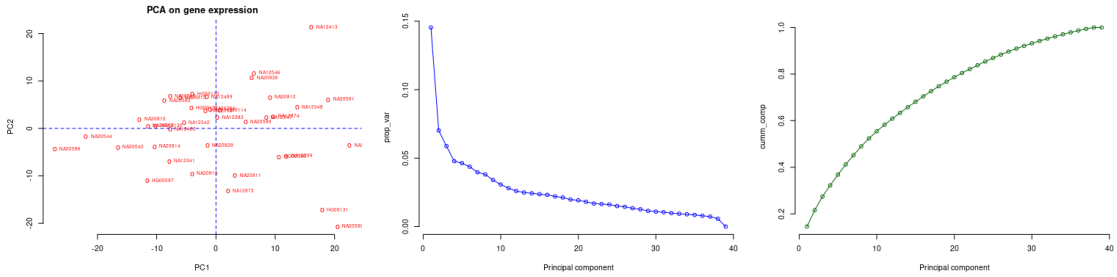

Figure 8.2: **eQTL Detector PCA on genotypes output:** first and second component for all samples (left), variability explanation for all compoenents (midle) and accumulated explained variability for each component (rigth).

9 *cis*-eQTL mapping using nominal pass [cis]

In this chapter we will explain how QTLtools do the *cis*-eQTL mapping using the nominal pass. QTLtools use the base code implemented in FastQTL [35] to perform eQTL mapping. As we mention before *cis*-eQTL analysis consist in finding statistically significant association between the gene expression (specific locus) and variants located in a window of 1 Mb or a different distance as the user decide, as this procedure need to be performed for all genes it can implicate millions of linear regressions,is important to clarify at this point that if we run 1000 linear regressions we will get 50 significant P-values that can be false positives, so in cases when we run millions of linear regressions this number is even higher, one solution for this problem was to apply the Bonferroni correction method [20], but this method can fall on creating even more false positives [35], in the case of QTLtools and FastQTL

the use of the False Discovery Rate (FDR) [6], a method that control the number of false positives tacking in count all *P*-values and their distribution, this correction method is applied to all the obtained p-vales in order to get corrected *P*-values (q-values), also this method will be applied on the follow eQTL mapping procedures described on the next chapters. In the nominal pass analysis QTLtools calculate the nominal P-value of association between the variant and the phenotype, then all the nominal *P*-values with the information about the locations of the variants and phenotypes are storage in a `.txt` file with the follow fields:

- 1. Phenotype ID
- 2. Chromosome ID of the phenotype
- 3. Phenotype start
- 4. Phenotype end
- 5. Phenotype strand orientation
- 6. Number of variants tested in cis
- 7. SNP-phenotype distance
- 8. ID of the tested variant
- 9. Tested variant chromosome ID
- 10. Tested variant start position
- 11. Tested variant end position
- 12. Nominal P-value
- 13. Regression slope
- 14. Flag equal to 1 for top variants

Then eQTL Detector sort this file and make a list with 30 most significant hits (Table 6.1), then eQTL Detector estimate the q-values for the set of the *P*-values obtained using the R package *qvalue* [41], then we present a q-value versus *P*-values scatter plot (Figure 7.1) to evaluate the distribution of the q-values obtained from the FRD correction, also an expected versus the obtained *P*-values, also distribution of the *P*-values and the distribution of the *P*-values for the hits (Figure 7.2), to see how the hits are distributed, in terms of distance from the location of the gene, we check the frequency (Figure 7.3 A) we also check the values of the slopes of the linear regressions and their influence on the nominal *P*-values (Figure 7.3 B) and

finally the distribution of the q-values. To see how the gene expression values are affected by the allele frequencies a boxplot for 16 hits are showed on the Figure 7.4.

Table 9.1: List of top 30 cis eQTL mapped using nominal pass, the columns are, 1. Phenotype ID 2. Chromosome ID of the phenotype 3. Phenotype start 4. Phenotype end 5. Phenotype strand orientation 6. Number of variants tested in cis 7. SNP-phenotype distance 8. ID of the tested variant 9. Tested variant chromosome ID 10. Tested variant start position 11. Tested variant end position 12. Nominal P-value 13. Regression slope 14. Flag equal to 1 for top variants

1	2	3	4	5	6	7	8	9	10	11	12	13	14
ENSG00000237438.2	chr22	17517460	17517460	+	5095	25350	22_17542810	chr22	17542810	17542810	0	-0.7518040	1
ENSG00000177663.9	chr22	17565844	17565844	+	5193	-52141	22_17513703	chr22	17513703	17513703	0	-0.2096170	1
ENSG00000069998.8	chr22	17646177	17646177	-	5291	25748	22_17620429	chr22	17620429	17620429	0	-1.5147200	1
ENSG00000131100.8	chr22	18111584	18111584	-	5272	33944	22_18077640	chr22	18077640	18077640	0	-1.4004200	1
ENSG00000269220.1	chr22	18260088	18260088	+	5389	1456	22_18261544	chr22	18261544	18261544	0	-0.7672050	1
ENSG00000093100.13	chr22	18294263	18294263	-	5397	-189125	22_18483388	chr22	18483388	18483388	0	0.5563130	1
ENSG00000243156.3	chr22	18507325	18507325	-	5030	23937	22_18483388	chr22	18483388	18483388	0	0.2009580	1
ENSG00000273032.1	chr22	19005347	19005347	+	4810	95396	22_19100743	chr22	19100743	19100743	0	-0.0135338	1
ENSG00000273311.1	chr22	19035888	19035888	-	4824	2923	22_19032965	chr22	19032965	19032965	0	0.0705657	1
ENSG00000100056.7	chr22	19132197	19132197	-	4726	-128	22_19132325	chr22	19132325	19132325	0	0.1145910	1
ENSG00000100075.5	chr22	19166343	19166343	-	4791	-337	22_19166680	chr22	19166680	19166680	0	-3.1124500	1
ENSG00000070371.11	chr22	19279239	19279239	-	4848	18	22_19279221	chr22	19279221	19279221	0	0.0329442	1
ENSG00000185608.4	chr22	19419425	19419425	+	4643	-76877	22_19342548	chr22	19342548	19342548	0	1.4967700	1
ENSG00000215012.4	chr22	19842419	19842419	-	4278	8393	22_19834026	chr22	19834026	19834026	0	-0.2711870	1
ENSG00000185838.9	chr22	19842462	19842462	-	4278	7352	22_19835110	chr22	19835110	19835110	0	0.0897887	1
ENSG00000093010.7	chr22	19929130	19929130	+	4214	-1038	22_19928092	chr22	19928092	19928092	0	0.7575610	1
ENSG00000184470.15	chr22	19929341	19929341	-	4215	41195	22_19888146	chr22	19888146	19888146	0	-0.2073360	1
ENSG00000265087.1	chr22	19951276	19951276	+	4217	-5	22_19951271	chr22	19951271	19951271	0	-2.7795700	1
ENSG00000183597.11	chr22	20004537	20004537	+	4234	21988	22_20026525	chr22	20026525	20026525	0	-0.2916570	1
ENSG00000128185.5	chr22	20307603	20307603	-	4113	13978	22_20293625	chr22	20293625	20293625	0	-0.7206240	1
ENSG00000206176.5	chr22	20378814	20378814	+	4278	111847	22_20490661	chr22	20490661	20490661	0	-0.1847910	1
ENSG00000215513.7	chr22	20427835	20427835	-	4361	-45583	22_20473418	chr22	20473418	20473418	0	-0.5856190	1
ENSG00000099940.7	chr22	21213271	21213271	+	3204	-170302	22_21042969	chr22	21042969	21042969	0	-0.9612090	1
ENSG00000241973.6	chr22	21213705	21213705	-	3203	146775	22_21066930	chr22	21066930	21066930	0	-0.2037210	1
ENSG00000272600.1	chr22	21245502	21245502	-	3175	55125	22_21190377	chr22	21190377	21190377	0	-1.5867100	1
ENSG00000099949.14	chr22	21333751	21333751	+	3211	2798	22_21336549	chr22	21336549	21336549	0	-0.2135130	1
ENSG00000183506.12	chr22	21871822	21871822	-	4681	49460	22_21822362	chr22	21822362	21822362	0	-0.1185060	1
ENSG00000100023.13	chr22	22006559	22006559	+	4899	43224	22_22049783	chr22	22049783	22049783	0	0.3608720	1
ENSG00000100030.10	chr22	22221970	22221970	-	5215	16617	22_22205353	chr22	22205353	22205353	0	-0.5178000	1
ENSG00000224086.3	chr22	22292609	22292609	+	5323	21854	22_22314463	chr22	22314463	22314463	0	-0.0408298	1

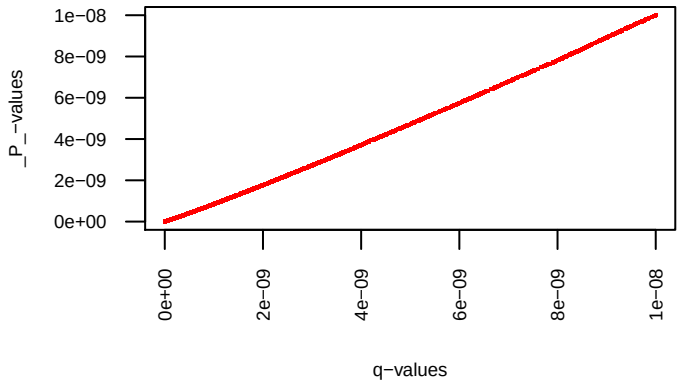

Figure 9.1: Cis nominal pass output; P-values again q-values obtained from the FDR correction method using the qvalue package

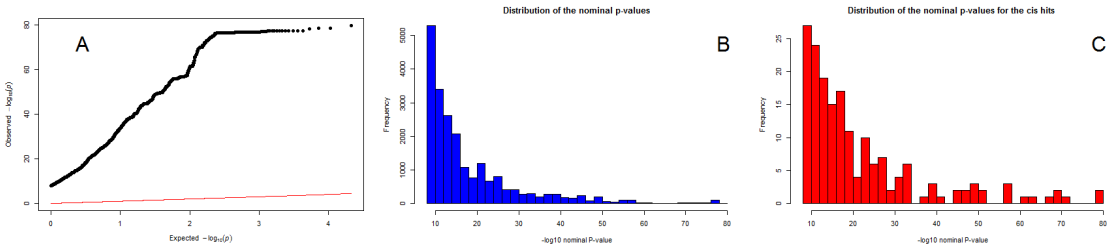

Figure 9.2: **eQTL Detectorcis nominal pass output: A:** Expected versus observed p-vales from nominal pass, **B:** distribution of the P-values and **C:** distribution of the hits selection by QTLtools.

10 Cis eQTL mapping using permutation pass [cis]

In the permutation pass analysis in QTLtools takes a number of permutations to adjusts nominal P-values from multiple testing to characterize the null distribution of the P-values [32], to then utilize the beta distribution to characterize the null distribution of the P-values.

The beta distribution is build typically from 100 to 1000 permutations using the maximum likelihood estimation, then in order to adjusted P-values at any significance level without run all possible permutations, an approximation method based on the

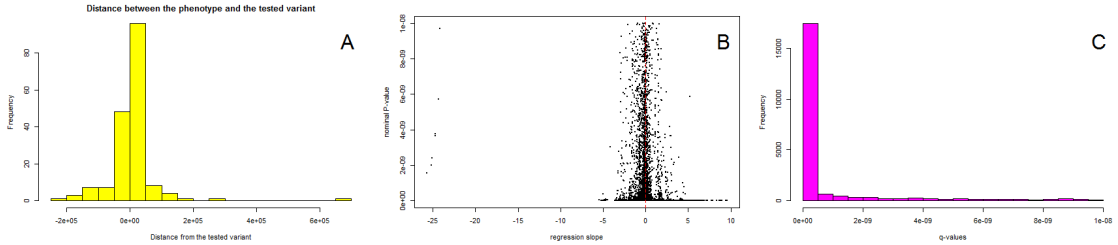

Figure 9.3: **eQTL Detector cis nominal pass output: A:** The distance between the phenotype and the tested variant for hits eQTL in cis nominal pass, **B:** slope of the regression versus the nominal p-value and **C:** the distribution of the q-values.

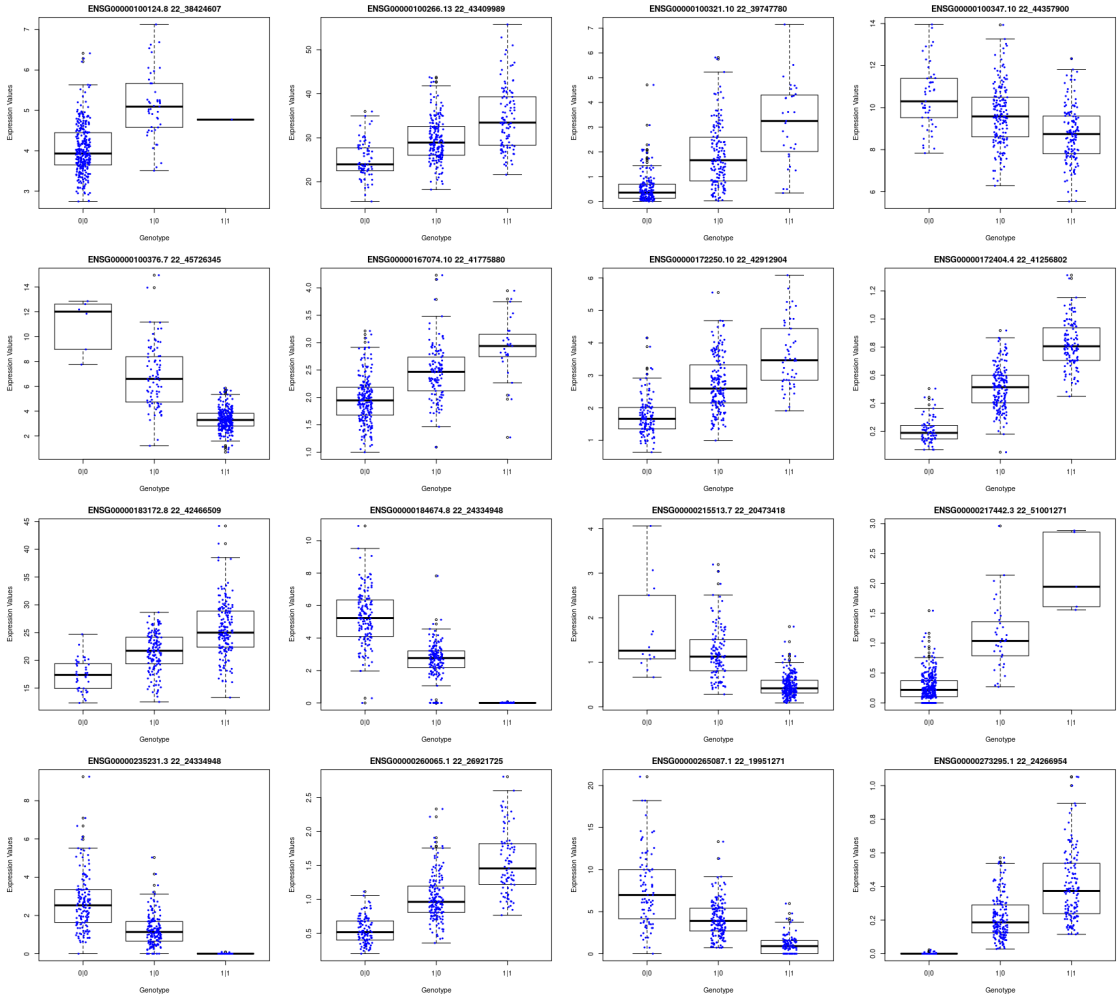

Figure 9.4: **eQTL Detector cis nominal pass output:** Gene expression for the differet allele frequencies on the to 16 eQTTL's mapped using nominal pass.

beta distribution is used as it is well know that the distribution of P-values are beta distributed [13]. The k^{th} smallest P-value from n permutations is distributed as:

$$U \sim Beta(k, n)$$

Then they propose to model the smallest P-value from L tests with a random beta distributed ($K = 1$ and $n = L$), but as we are looking into regions that are in fact correlated, due the linkage disequilibrium (**LD**), then independence of the L test performed is not, so QTLtools does not fix n and k *a priori*, instead of that they use a maximum likelihood estimation for those parameters [18], so basically they run R permutations to a null vector of P-values $p_1, ..., p_R$, to do that they use:

$$L(k, n|p_1, ..., p_R) = (k-1)\sum_{r=1}^{R} lnp_r + (n-1)\sum_{r=1}^{R} ln(1-p_r) - Rln\left[\frac{\Gamma(k)\Gamma(n)}{\Gamma(kn)}\right]$$

Then an approximation of an adjusted P-value (p_b) from the best nominal P-value p_n fit the beta distribution by:

$$p_b = P(U \leq p_n)$$

With the beta p-values QTL tools retrieve a `.txt` file with the follow fields:

- 1. Phenotype ID
- 2. Chromosome ID
- 3. Phenotype start
- 4. Phenotype end position
- 5. Strand orientation of the phenotype
- 6. Number of variants tested
- 7. Distance between the phenotype and variant
- 8. Top variant ID
- 9. Top variant chromosome ID
- 10. Top variant start position
- 11. Top variant end position

- 12. Degrees of freedom used to compute the P-values
- 13. Dummy field
- 14. First parameter of beta distribution
- 15. Second parameter beta distribution
- 16. Nominal P-value
- 17. Regression slope
- 18. Eempirical P-value
- 19. P-value of association

Then eQTL Detector make some visualization (Figure 8.1) and present the most relevant hits the beta distributed p-values are also passed through the FDR correction (Table 8.1).

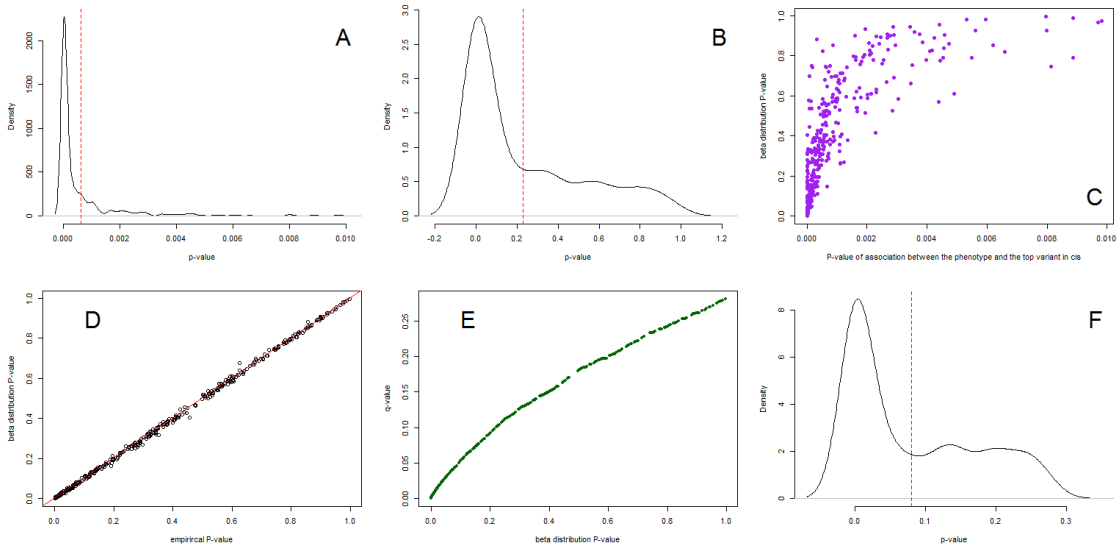

Figure 10.1: **eQTL Detector cis permutation pass output: A:** Distribution of the P-values for the association between the expression and the variant call. **B:** Distribution of the Beta P-values. **C:** Beta P-values versus the p-values from the gene expression due the different allele frequences. **D:** Empirical P-values versus the beta P-values, **E:** beta P-values versus the q-values. **F:** Distribution of the q-values. All the distribution plots include a red line that indicate the mean ov the p or q values vector.

Table 10.1: List of top 30 cis eQTL mapped using permutation pass, the columns are: 1. Phenotype ID 2. Phenotype start 3. End poss of the phenotype 4. End position of the phenotype 5. Strand orientation of the phenotype 6. Number of variants tested in cis 7. Distance between the phenotype and the tested variant 8. ID of the top variant 9. Chromosome ID of the top variant 10. Start position of the top variant 11. End position of the top variant 12. Number of degrees of freedom used to compute the P-values 13. Dummy field 14. First parameter beta distribution 15. Second parameter value of the fitted beta distribution 16. Nominal P-value 17. Regression slope 18. Eempirical P-value 19. P-value of association 20. q-value.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
ENSG00000183172.8	chr22	42475695	42475695	+	4119	-9186	22_42466509	chr22	42466509	42466509	356	298.011	1.036160	180.995	0	3.5837700	0.000999	0	0
ENSG00000100376.7	chr22	45704849	45704849	+	6049	21496	22_45726345	chr22	45726345	45726345	356	284.974	1.056220	468.049	0	-3.1419900	0.000999	0	0
ENSG00000260065.1	chr22	26921002	26921002	-	5286	-723	22_26921725	chr22	26921725	26921725	356	321.878	1.024560	801.387	0	0.3929310	0.000999	0	0
ENSG00000172404.4	chr22	41258130	41258130	-	2636	1328	22_41256802	chr22	41256802	41256802	356	304.199	0.997099	115.152	0	0.2458230	0.000999	0	0
ENSG00000184674.8	chr22	24384680	24384680	-	4896	49732	22_24334948	chr22	24334948	24334948	356	315.672	0.988374	400.765	0	-2.0635400	0.000999	0	0
ENSG00000273295.1	chr22	24249255	24249255	-	4934	-17699	22_24266954	chr22	24266954	24266954	356	300.358	1.036130	347.193	0	0.1629070	0.000999	0	0
ENSG00000100321.10	chr22	39745930	39745930	+	3433	1850	22_39747780	chr22	39747780	39747780	356	287.512	1.084030	219.861	0	1.1476000	0.000999	0	0
ENSG00000100347.10	chr22	44351301	44351301	+	6398	6599	22_44357900	chr22	44357900	44357900	356	324.137	1.037070	857.089	0	-0.9103620	0.000999	0	0
ENSG00000217442.3	chr22	51001334	51001334	-	3359	63	22_51001271	chr22	51001271	51001271	356	279.672	0.965100	207.236	0	0.7993940	0.000999	0	0
ENSG00000100266.13	chr22	43411151	43411151	-	5283	1162	22_43409989	chr22	43409989	43409989	356	309.188	1.023270	395.046	0	3.8690900	0.000999	0	0
ENSG00000215513.7	chr22	20427835	20427835	-	4361	-45583	22_20473418	chr22	20473418	20473418	356	303.886	0.983164	415.235	0	-0.5856190	0.000999	0	0
ENSG00000235231.3	chr22	24373117	24373117	+	4902	-38169	22_24334948	chr22	24334948	24334948	356	308.875	1.010680	416.029	0	-1.0746000	0.000999	0	0
ENSG00000167074.10	chr22	41763337	41763337	+	3002	12543	22_41775880	chr22	41775880	41775880	356	294.465	1.066250	110.783	0	0.4133690	0.000999	0	0
ENSG00000172250.10	chr22	42896585	42896585	+	4670	16319	22_42912904	chr22	42912904	42912904	356	281.115	1.102030	244.422	0	0.8074690	0.000999	0	0
ENSG00000100124.8	chr22	38245334	38245334	-	3988	-179273	22_38424607	chr22	38424607	38424607	356	294.569	1.080950	335.842	0	0.8222770	0.000999	0	0
ENSG00000265087.1	chr22	19951276	19951276	+	4217	-5	22_19951271	chr22	19951271	19951271	356	301.453	0.988548	348.622	0	-2.7795700	0.000999	0	0
ENSG00000184164.10	chr22	50311815	50311815	+	6751	1544	22_50313359	chr22	50313359	50313359	356	307.554	1.058640	637.892	0	-2.2898800	0.000999	0	0
ENSG00000100147.9	chr22	42196683	42196683	+	3695	-75819	22_42120864	chr22	42120864	42120864	356	297.211	1.075480	183.074	0	-0.6426900	0.000999	0	0
ENSG00000272666.1	chr22	50981335	50981335	-	3388	3073	22_50978262	chr22	50978262	50978262	356	273.136	0.990839	184.063	0	7.8536800	0.000999	0	0
ENSG00000128394.12	chr22	39436609	39436609	+	3669	6558	22_39443167	chr22	39443167	39443167	356	311.517	1.025820	304.732	0	-0.7258750	0.000999	0	0
ENSG00000260708.1	chr22	47158460	47158460	-	5361	848	22_47157612	chr22	47157612	47157612	356	323.946	1.013440	637.517	0	-1.0353400	0.000999	0	0
ENSG00000100075.5	chr22	19166343	19166343	-	4791	-337	22_19166680	chr22	19166680	19166680	356	327.510	0.992360	633.868	0	-3.1124500	0.000999	0	0
ENSG00000273272.1	chr22	50981079	50981079	+	3390	-2817	22_50978262	chr22	50978262	50978262	356	274.075	0.953626	189.399	0	9.4835000	0.000999	0	0
ENSG00000070371.11	chr22	19279239	19279239	-	4848	18	22_19279221	chr22	19279221	19279221	356	297.020	0.925251	399.250	0	0.0329442	0.000999	0	0
ENSG00000128408.7	chr22	45809572	45809572	+	5877	52	22_45809624	chr22	45809624	45809624	356	288.694	1.082090	451.627	0	-0.4098270	0.000999	0	0
ENSG00000131100.8	chr22	18111584	18111584	-	5272	33944	22_18077640	chr22	18077640	18077640	356	301.018	1.127820	559.314	0	-1.4004200	0.000999	0	0
ENSG00000167077.8	chr22	42095503	42095503	+	3561	82396	22_42177899	chr22	42177899	42177899	356	292.031	1.057160	142.479	0	0.6071980	0.000999	0	0
ENSG00000100023.13	chr22	22006559	22006559	+	4899	43224	22_22049783	chr22	22049783	22049783	356	310.631	1.049400	473.121	0	0.3608720	0.000999	0	0
ENSG00000179750.11	chr22	39378352	39378352	+	3631	-20315	22_39358037	chr22	39358037	39358037	356	298.706	1.036270	253.090	0	-3.6277600	0.000999	0	0
ENSG00000100304.12	chr22	43583139	43583139	-	5481	-459	22_43583598	chr22	43583598	43583598	356	319.948	1.029560	588.165	0	-1.9265500	0.000999	0	0

11 Trans full pass [trans]

For *trans*-e_QTL's, nominal pass and permutation pass need to performed, these two procedures produce two different sets of outputs:

- nominal pass
  - trans_nominal_best
    - 1. Phenotype ID
    - 2. Dummy field
    - 3. Nominal P-value
  - trans_nominal_bins
    - 1. Index
    - 2-3. Dummy fields
    - 4. Upper boundary of the p-value
    - 5. Lower boundary of the p-value
    - 6. Number of tests performed in the bin
  - trans_nominal_hits
    - 1. Phenotype ID
    - 2. Chromosome ID of the phenotype
    - 3. Phenotype start
    - 4. Variant ID
    - 5. Chromosome ID of the variant
    - 6. Variant physical position
    - 7. Nominal P-value
    - 8. Dummy here
    - 9. Regression slope
- permutation pass
  - trans_permutation_best
    - 1. Phenotype ID
    - 2. Dummy field
    - 3. Nominal P-value
  - trans_permutation_bins
    - 1. Index
    - 2-3. Dummy fields
    - 4. Upper boundary of the p-value
    - 5. Lower boundary of the p-value
    - 6. Number of tests performed in
  - trans_permutation_hits

* 1. Phenotype ID
* 2. Chromosome ID of the phenotype
* 3. Phenotype start
* 4. Variant ID
* 5. Chromosome ID of the variant
* 6. Variant physical position
* 7. Nominal P-value
* 8. Dummy here
* 9. Regression slope

Then *best* is an output that contain the top hit per phenotype, *bins* contain all the hits that are bellow the threshold (we use the default `--threshold`=1e-5), then one of the outputs is the qqplot of the *p*-values obtained from both methods (Figure 11.1 B), then the FDR correction is applied on both methods, the Table 11.1 shows the 30 first hits obtained.

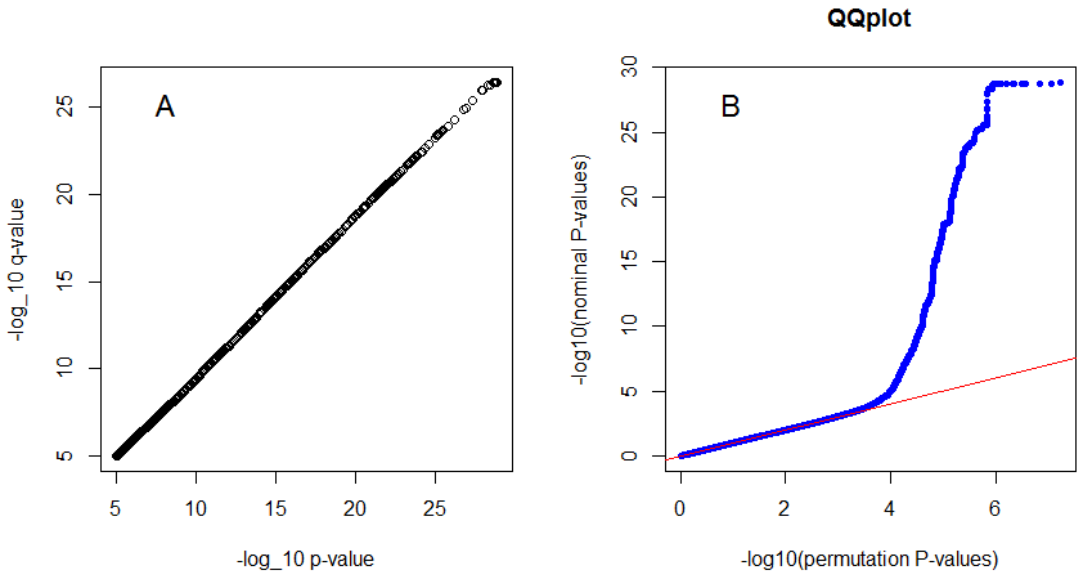

Figure 11.1: **eQTL Detector trans full pass output: A:** p-values again the qvalues from the FDR correction. *B:* Permutation versus nominal pass p-values.

Table 11.1: 30 first hist obtained from trans full pass and the FDR correction:. 1. Phenotype ID 2. Phenotype chrID 3. Phenotype start 4. Variant ID 5. Variant chrID 6. Variant position 7. Nominal P-value of association 8. Dummy field 9. Regression slope 10. Beta p-value.

	1	2	3	4	5	6	7	8	9	10
1405	ENSG00000100077.10	chr22	25960816	22_39794241	chr22	39794241	0	-1	0.549199	0
1417	ENSG00000100077.10	chr22	25960816	22_39806153	chr22	39806153	0	-1	-0.548572	0
1410	ENSG00000100077.10	chr22	25960816	22_39797779	chr22	39797779	0	-1	0.548009	0
1413	ENSG00000100077.10	chr22	25960816	22_39798449	chr22	39798449	0	-1	0.548009	0
1414	ENSG00000100077.10	chr22	25960816	22_39799789	chr22	39799789	0	-1	0.548009	0
1418	ENSG00000100077.10	chr22	25960816	22_39809820	chr22	39809820	0	-1	0.548009	0
1419	ENSG00000100077.10	chr22	25960816	22_39810379	chr22	39810379	0	-1	0.548009	0
1420	ENSG00000100077.10	chr22	25960816	22_39812409	chr22	39812409	0	-1	0.548009	0
1422	ENSG00000100077.10	chr22	25960816	22_39819008	chr22	39819008	0	-1	0.548009	0
1423	ENSG00000100077.10	chr22	25960816	22_39819049	chr22	39819049	0	-1	0.548009	0
1425	ENSG00000100077.10	chr22	25960816	22_39820885	chr22	39820885	0	-1	0.548009	0
1426	ENSG00000100077.10	chr22	25960816	22_39821319	chr22	39821319	0	-1	0.548009	0
1427	ENSG00000100077.10	chr22	25960816	22_39821536	chr22	39821536	0	-1	0.548009	0
1428	ENSG00000100077.10	chr22	25960816	22_39821641	chr22	39821641	0	-1	0.548009	0
1430	ENSG00000100077.10	chr22	25960816	22_39822116	chr22	39822116	0	-1	0.548009	0
1431	ENSG00000100077.10	chr22	25960816	22_39823015	chr22	39823015	0	-1	0.548009	0
1435	ENSG00000100077.10	chr22	25960816	22_39825492	chr22	39825492	0	-1	0.548009	0
1436	ENSG00000100077.10	chr22	25960816	22_39826788	chr22	39826788	0	-1	0.548009	0
1437	ENSG00000100077.10	chr22	25960816	22_39827553	chr22	39827553	0	-1	0.548009	0
1438	ENSG00000100077.10	chr22	25960816	22_39829736	chr22	39829736	0	-1	0.548009	0
1411	ENSG00000100077.10	chr22	25960816	22_39797987	chr22	39797987	0	-1	0.547374	0
1412	ENSG00000100077.10	chr22	25960816	22_39798429	chr22	39798429	0	-1	0.547374	0
1415	ENSG00000100077.10	chr22	25960816	22_39800704	chr22	39800704	0	-1	0.547374	0
1433	ENSG00000100077.10	chr22	25960816	22_39824707	chr22	39824707	0	-1	0.547027	0
1434	ENSG00000100077.10	chr22	25960816	22_39825322	chr22	39825322	0	-1	0.546120	0
1409	ENSG00000100077.10	chr22	25960816	22_39797178	chr22	39797178	0	-1	0.544748	0
1394	ENSG00000100077.10	chr22	25960816	22_39790191	chr22	39790191	0	-1	0.544714	0
1395	ENSG00000100077.10	chr22	25960816	22_39790987	chr22	39790987	0	-1	0.544714	0
1396	ENSG00000100077.10	chr22	25960816	22_39791491	chr22	39791491	0	-1	0.544714	0
1397	ENSG00000100077.10	chr22	25960816	22_39792943	chr22	39792943	0	-1	0.544714	0

12 fdensity

The fdensity mode is used to check the density of annotations around the position of the detected *cis*-eQTL's, for this case we will use the permutation pass results. In this case also the user needs to provide a functional annotation for few Transcription Factor Binding sites file. The first step is the run the FDR correction on the *cis* permutation pass results, then from the `.txt` a reduction, using AWK, of the number of fields that will be used for this analysis leave files with these fields:

- 1. Chromosome ID
- 2. Start position of the QTL
- 3. End position of the QTL
- 4. QTL ID
- 5. Targeted phenotype ID
- 6. Strand orientation

Then after perform the `[fdensity]`analysis we get a file with:

- 1. start position
- 2. end position
- 3. Bin number of annotations

Then eQTL Detector present a plot how the number of annotations are distributed in terms of distance (Figure 12.1).

13 fenrich

, to commit this QTLtools make a list of all the annotations close to a locus (1 Mb range),then if we assume the distribution of the of the p-values is non-uniform, due the existence of a QTL's in that range,this analysis consist in take a QTL and make a list of the phenotypes around (1Mb), then for X phenotypes there is a list of X annotations, for each annotation there is a list of Y significant variants that overlap an annotation, with this a frequency of variants falling in an annotation can be calculated (Y/X), the output of QTLools is a table that gives this frequency as follow:

observed number of QTLs falling within the functional annotations	8909.0000

total number of QTLs	10774.0000
mean expected number of QTLs falling within the functional annotations.	8763.3000
standard deviation of the expected number of QTLs falling within the functional annotations.	37.9972

On the Figure 13.1 (A) we can see how variants fall on annotations regions for the case of the chromosome 22, then if we zooming on a region with significant *cis* eQTL's we can see how the p-values for an specific phenotype get out of the rest of the groups (Figure 13.1 B) and how the ranges for the values of this phenotype also change (Figure 13.1 C), the opposite case for a non hit location is showed on the Figure 13.1 D and F.

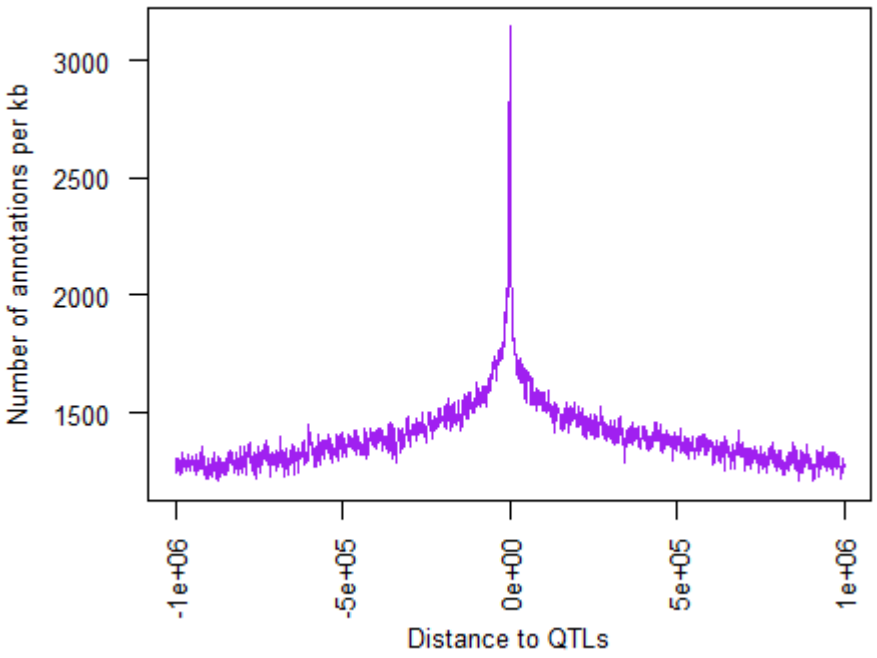

Figure 12.1: **eQTL Detector fdensity output:** Number of functional annotations from the mid point of the bin

14 GWAS hits colocalization [rtc]

The last step in the pipeline will be RTC, Regulatory Trait Concordance [33], the aim of this analysis is to asses if a GWAS hit is tagging the same variant detected with *cis*-eQTL's, even when we have a high linkage disequilibrium it may may not necessary mean we are underlying a functional variant, but in a scenario where this association is real, then if we remove the GWAS effect the eQTL association will be considerable affected by decreasing their association, this evaluation of the reduction on the significance by removing the GWAS effect can be estimated by the follow steps.

- 1. A new molecular phenotype is created by removing the GWAS effect using residual estimated from the GWAS hit, it mean removing the values on the expression given by the residuals estimated on the GWAS hit.
- 2. A new test for the association between the new phenotype and the variant is estimated
- 3. A *hot spot* region that contain eQTL's and GWAS hits is defined, then this region has N_{SNPs} variants.

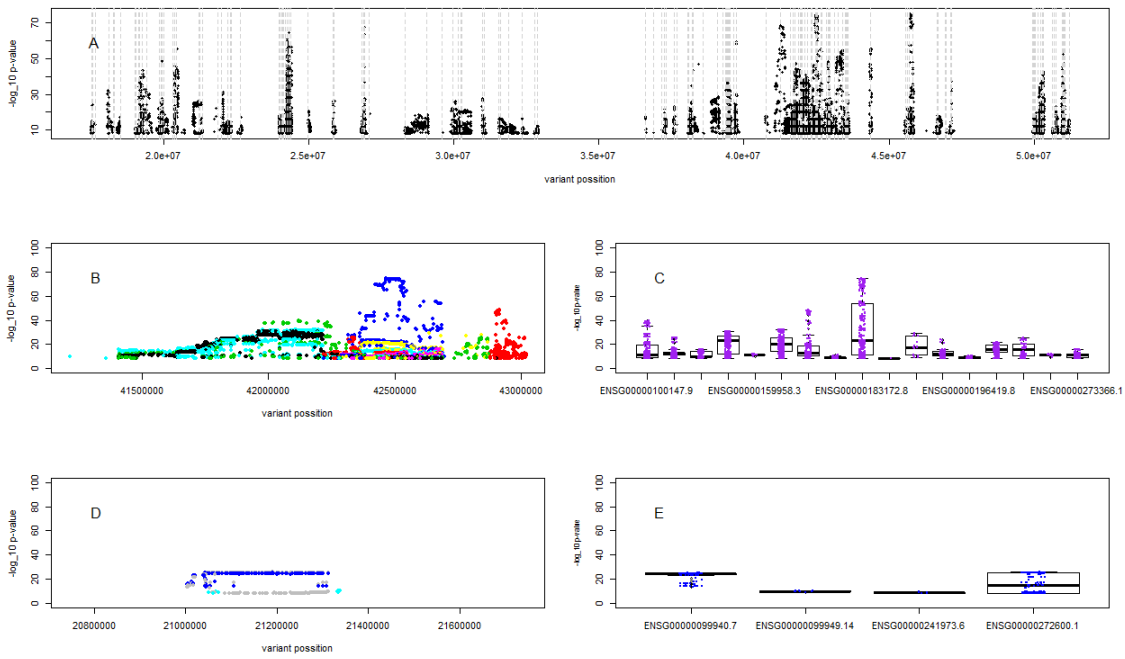

Figure 13.1: **eQTL Detector fenrich output: A:** The *p*-values and the location of the variants for the whole set on the chromosome 22, the vertical grey lines represent the location of a phenotype. **B** and **C:** in a region of 1 Mb where we have a *cis*-eQTL ,**B** is showing how the *p*-values by phenotype (dots colored) and the possition of the variant and **C** a boxplot of the *p*-values for the diferent phenotype. **D** and **F:** a region (1 Mb) without *cis*-eQTL hits, ,**D** is showing how the *p*-values by phenotype (dots colored) and the possition of the variant and **E** a boxplot of the *p*-values for the diferent phenotype

- 4. N_{SNPs} pseudophenotypes are created using the residuals from the GWAS hits
- 5. A for the association between the pseudophenotype and eQTL is perform then a vector of p-values is obtained and sorted
- 6. From the sorted vector with the ranked of each variant $Rank_{GWAS\ \ SNP}$ the RTC score is calculated by:

$$RTC = \frac{N_{SNPs} - Rank_{GWAS\ \ SNP}}{N_{SNPs}}$$

With the obtained output from QTLtools `rct` eQTL Detector takes the data from the GWAS catalog [8] and all the variants with RTC score over 0.9 are shown on the Table with the correspond information on the GWAS hit and the Figure 14.1 show also how rtc are distributed and how is the relation RTC distance from the variant.

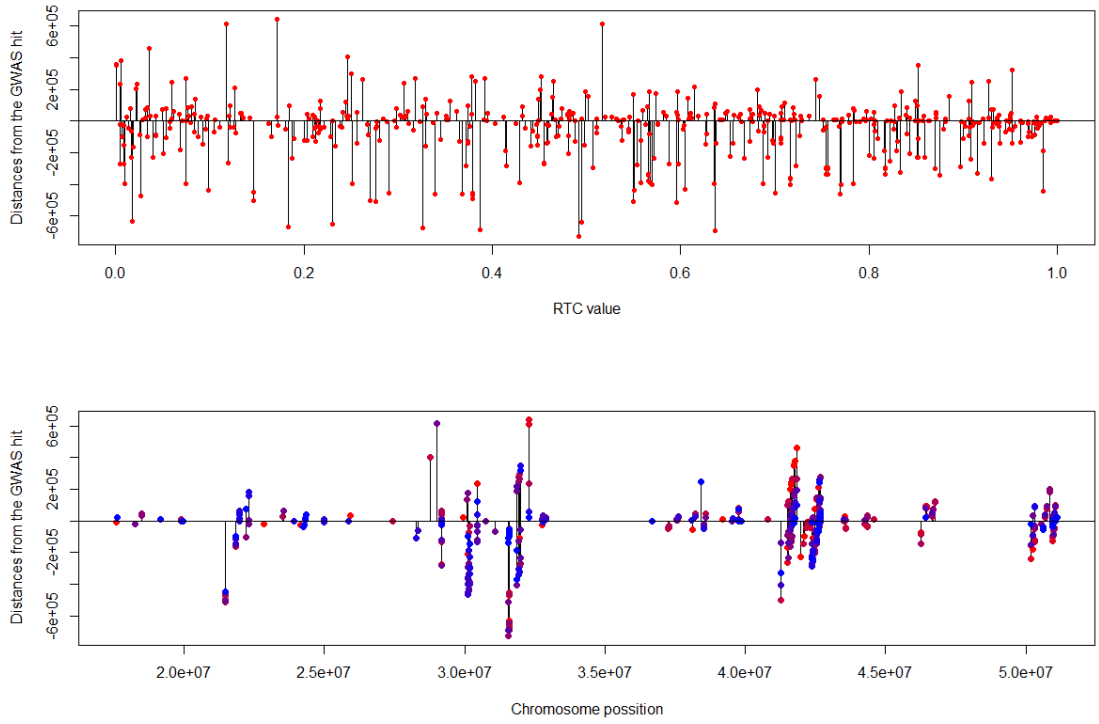

Figure 14.1: **eQTL Detector rtc output:** On the top, the distance fromthe GWAS hit to the location of the tested variant. On the bottom, how the RTC is distributed on the chromosome 22, the red dots denotate the hits with an RTC score value onver 0.9.

15 Discusion

Mapping eQTL's is a process that required multiple computation steps, from quality check of the sequence data to co-localization with GWAS hits, all these steps generate multiple outputs and all of them required inputs from a previous step or from a external source, the main idea of eQTL Detector is to take care of all these connections and automatize the process of eQTL mapping, this way of pipeline implementation reduce the workload associated to the step by step process of the eQTL mapping, but it can also fall into low interaction with the end user on the analysis as many steps required some parameters that in our case are setup as default (*i.e.* p-values thresholds or permutation number), to try to avoid this type issues the pipeline is available in Git so the end user can always modify those parameters according to their requirements.

The last step in our pipeline is the RTC analysis, as we mention before the aim of this analysis is to co-localize eQTL with GWAS hits from the GWAS catalog, this step can help on understand previous GWAS hits cataloged, but this may be a good end point of this pipeline, in that sense adding one more step that can help on finding causative SNP's from *cis*-eQTL may help on selecting SNP's that can make better prediction in a particular phenotype, but as this pipeline is implemented in a container that use a base Rocker-rstudio the possibilities for the end user to go into an R studio session and try run any extra step or modify any output from the present pipeline is available. We need to add also that using two different data sources did not test the whole pipeline as using the first data (39 samples) we didn't find significant eQTL to continue with the next steps on the analysis.

16 Conslusion

The implementation of the pipeline in a Docker container that run automatically all the steps and create an `rmd` report cover the general objectives of the present work, in terms of specific objectives we cover almost all of them with one last observation related with the fine mapping analysis, adding some extra steps that use tools different than QTLtools like CAVIAR [22, 21].

17 Glossary

- **BAM (*file format*):** Binary Alignment Map is a type of row data format of genome sequencing, it consist on a representation without losses of the Sequence Alignment Map-files (SAM) [28].
- **BED (*file format*):** BED file, Browser Extensive Data, develop by the Human Genome Project [26], is a text format separated by spaces or tabs that contain genomic regions a coordinates (first 3 columns are: chrom, chromStart and chromEnd) with annotations associated, is one of the mos common type of format in bioinformatics, the advantage of having these 3 coordinates helps on data manipulation and optimize computation timing.
- **GTF (*file format*):** The GTF, General Feature Format,it use one line per feature using 9 columns, the file is separated by tabs and each line should have an empty value denoted by ".". the fields for this format are:
  - seqname
  - source
  - start
  - end
  - score
  - strand
  - frame
  - attribute
- **GWAS:** A GWAS (Genome Wide Association) is a method to detect SNPs that are associated with some particular trait, the method looks allele frequencies related to a control case type of experiment.
- **Linkage Disequilibrium (LD):** The linkage disequilibrium is the non randomize association between the allele frequencies and the phenotype in a population [38], meainig that two alleles are in equilibrium when their frequencies are randomly distributed, in the opposite case the are in disequilibrium when their frequencies are not randomly distributed due some mutation event, genetic distances or some historical event on the population.
- **QTL:** Quantitative trait *locus* or *loci*, refer to an specific region or many of them that are correlated to a certain quantitative phenotype.
- **SNP:** Single Nucleotid Polymorfism, is a genetic variation on one specific base (A, C, T or G), they occur in a frequency of 1 in 1000 nucleotides, this

type of variation are extremely polymorphic as they can be detected from one individual to another, also from one generation to another.

References

[1] "A global reference for human genetic variation". In: *Nature* 526.7571 (Sept. 2015), pp. 68–74. DOI: 10.1038/nature15393. URL: https://doi.org/10.1038/nature15393.

[2] Farnoosh Abbas-Aghababazadeh, Qian Li, and Brooke L. Fridley. "Comparison of normalization approaches for gene expression studies completed with high-throughput sequencing". In: *PLOS ONE* 13.10 (Oct. 2018). Ed. by Honghuang Lin, e0206312. DOI: 10.1371/journal.pone.0206312. URL: https://doi.org/10.1371/journal.pone.0206312.

[3] Enis Afgan et al. "The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2018 update". In: *Nucleic Acids Research* 46.W1 (May 2018), W537–W544. DOI: 10.1093/nar/gky379. URL: https://doi.org/10.1093/nar/gky379.

[4] Alfred V. Aho, Brian W. Kernighan, and Peter J. Weinberger. "Awk — a pattern scanning and processing language". In: *Software: Practice and Experience* 9.4 (Apr. 1979), pp. 267–279. DOI: 10.1002/spe.4380090403. URL: https://doi.org/10.1002/spe.4380090403.

[5] JJ Allaire et al. *rmarkdown: Dynamic Documents for R*. R package version 2.1. 2020. URL: https://CRAN.R-project.org/package=rmarkdown.

[6] Yoav Benjamini and Yosef Hochberg. "Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing". In: *Journal of the Royal Statistical Society Series B (Methodological)* 57.1 (1995), pp. 289–300. DOI: http://dx.doi.org/10.2307/2346101. URL: http://dx.doi.org/10.2307/2346101.

[7] Carl Boettiger and Dirk Eddelbuettel. "An Introduction to Rocker: Docker Containers for R". In: *The R Journal* 9.2 (2017), pp. 527–536. DOI: 10.32614/RJ-2017-065. URL: https://doi.org/10.32614/RJ-2017-065.

[8] Annalisa Buniello et al. "The NHGRI-EBI GWAS Catalog of published genome-wide association studies, targeted arrays and summary statistics 2019". In: *Nucleic Acids Research* 47.D1 (Nov. 2018), pp. D1005–D1012. DOI: 10.1093/nar/gky1120. URL: https://doi.org/10.1093/nar/gky1120.

[9] Carrie A Davis et al. "The Encyclopedia of DNA elements (ENCODE): data portal update". In: *Nucleic Acids Research* 46.D1 (Nov. 2017), pp. D794–D801. DOI: 10.1093/nar/gkx1081. URL: https://doi.org/10.1093/nar/gkx1081.

[10] Olivier Delaneau et al. "A complete tool set for molecular QTL discovery and analysis". In: *Nature Communications* 8.1 (May 2017). DOI: 10.1038/ncomms15452. URL: https://doi.org/10.1038/ncomms15452.

[11] Yanlei Diao, Abhishek Roy, and Toby Bloom. “Building Highly-Optimized, Low-Latency Pipelines for Genomic Data Analysis”. In: *CIDR*. 2015.

[12] Arnis Druka et al. “Expression quantitative trait loci analysis in plants”. In: *Plant Biotechnology Journal* 8.1 (Jan. 2010), pp. 10–27. DOI: 10.1111/j.1467-7652.2009.00460.x. URL: https://doi.org/10.1111/j.1467-7652.2009.00460.x.

[13] Frank Dudbridge and Bobby P.C. Koeleman. “Efficient Computation of Significance Levels for Multiple Associations in Large Studies of Correlated Data, Including Genomewide Association Studies”. In: *The American Journal of Human Genetics* 75.3 (Sept. 2004), pp. 424–435. DOI: 10.1086/423738. URL: https://doi.org/10.1086/423738.

[14] Valur Emilsson et al. “Genetics of gene expression and its effect on disease”. In: *Nature* 452.7186 (Mar. 2008), pp. 423–428. DOI: 10.1038/nature06758. URL: https://doi.org/10.1038/nature06758.

[15] Bjørn Fjukstad and Lars Ailo Bongo. “A Review of Scalable Bioinformatics Pipelines”. In: *Data Science and Engineering* 2.3 (Sept. 2017), pp. 245–251. DOI: 10.1007/s41019-017-0047-z. URL: https://doi.org/10.1007/s41019-017-0047-z.

[16] Alexandre Fort et al. “MBV: a method to solve sample mislabeling and detect technical bias in large combined genotype and sequencing assay datasets”. In: *Bioinformatics* 33.12 (Feb. 2017). Ed. by Oliver Stegle, pp. 1895–1897. DOI: 10.1093/bioinformatics/btx074. URL: https://doi.org/10.1093/bioinformatics/btx074.

[17] Adam Freeman. “Docker Compose”. In: *Essential Docker for ASP.NET Core MVC*. Apress, 2017, pp. 97–117. DOI: 10.1007/978-1-4842-2778-7_6. URL: https://doi.org/10.1007/978-1-4842-2778-7_6.

[18] Nicholas W. Galwey. “A new measure of the effective number of tests, a practical tool for comparing families of non-independent significance tests”. In: *Genetic Epidemiology* 33.7 (Nov. 2009), pp. 559–568. DOI: 10.1002/gepi.20408. URL: https://doi.org/10.1002/gepi.20408.

[19] Yoav Gilad, Scott A. Rifkin, and Jonathan K. Pritchard. “Revealing the architecture of gene regulation: the promise of eQTL studies”. In: *Trends in Genetics* 24.8 (Aug. 2008), pp. 408–415. DOI: 10.1016/j.tig.2008.06.001. URL: https://doi.org/10.1016/j.tig.2008.06.001.

[20] Winston Haynes. “Bonferroni Correction”. In: *Encyclopedia of Systems Biology*. Springer New York, 2013, pp. 154–154. DOI: 10.1007/978-1-4419-9863-7_1213. URL: https://doi.org/10.1007/978-1-4419-9863-7_1213.

[21] Farhad Hormozdiari et al. “Colocalization of GWAS and eQTL Signals Detects Target Genes”. In: *The American Journal of Human Genetics* 99.6 (Dec. 2016),

pp. 1245–1260. DOI: 10.1016/j.ajhg.2016.10.003. URL: https://doi.org/10.1016/j.ajhg.2016.10.003.

[22] Farhad Hormozdiari et al. “Identifying Causal Variants at Loci with Multiple Signals of Association”. In: *Genetics* 198.2 (Aug. 2014), pp. 497–508. DOI: 10.1534/genetics.114.167908. URL: https://doi.org/10.1534/genetics.114.167908.

[23] Sayed M. Hosseini-Vardanjani et al. “Incorporating Prior Knowledge of Principal Components in Genomic Prediction”. In: *Frontiers in Genetics* 9 (Aug. 2018). DOI: 10.3389/fgene.2018.00289. URL: https://doi.org/10.3389/fgene.2018.00289.

[24] Shiro Ikegawa. “A Short History of the Genome-Wide Association Study: Where We Were and Where We Are Going”. In: *Genomics & Informatics* 10.4 (2012), p. 220. DOI: 10.5808/gi.2012.10.4.220. URL: https://doi.org/10.5808/gi.2012.10.4.220.

[25] “Initial sequencing and analysis of the human genome”. In: *Nature* 409.6822 (Feb. 2001), pp. 860–921. DOI: 10.1038/35057062. URL: https://doi.org/10.1038/35057062.

[26] W. J. Kent et al. “The Human Genome Browser at UCSC”. In: *Genome Research* 12.6 (May 2002), pp. 996–1006. DOI: 10.1101/gr.229102. URL: https://doi.org/10.1101/gr.229102.

[27] H. Li. “Tabix: fast retrieval of sequence features from generic TAB-delimited files”. In: *Bioinformatics* 27.5 (Jan. 2011), pp. 718–719. DOI: 10.1093/bioinformatics/btq671. URL: https://doi.org/10.1093/bioinformatics/btq671.

[28] H. Li et al. “The Sequence Alignment/Map format and SAMtools”. In: *Bioinformatics* 25.16 (June 2009), pp. 2078–2079. DOI: 10.1093/bioinformatics/btp352. URL: https://doi.org/10.1093/bioinformatics/btp352.

[29] Fábio Madeira et al. “The EMBL-EBI search and sequence analysis tools APIs in 2019”. In: *Nucleic Acids Research* 47.W1 (Apr. 2019), W636–W641. DOI: 10.1093/nar/gkz268. URL: https://doi.org/10.1093/nar/gkz268.

[30] M. T. Maurano et al. “Systematic Localization of Common Disease-Associated Variation in Regulatory DNA”. In: *Science* 337.6099 (Sept. 2012), pp. 1190–1195. DOI: 10.1126/science.1222794. URL: https://doi.org/10.1126/science.1222794.

[31] Dirk Merkel. “Docker: lightweight Linux containers for consistent development and deployment”. In: *Linux Journal* 2014 (Mar. 2014).

[32] Stephen B. Montgomery et al. “Transcriptome genetics using second generation sequencing in a Caucasian population”. In: *Nature* 464.7289 (Mar. 2010),

pp. 773–777. DOI: 10.1038/nature08903. URL: https://doi.org/10.1038/nature08903.

[33] Alexandra C. Nica et al. "Candidate Causal Regulatory Effects by Integration of Expression QTLs with Complex Trait Genetic Associations". In: *PLoS Genetics* 6.4 (Apr. 2010). Ed. by Greg Gibson, e1000895. DOI: 10.1371/journal.pgen.1000895. URL: https://doi.org/10.1371/journal.pgen.1000895.

[34] Conor Nodzak. "Introductory Methods for eQTL Analyses". In: *Methods in Molecular Biology*. Springer US, Dec. 2019, pp. 3–14. DOI: 10.1007/978-1-0716-0026-9_1. URL: https://doi.org/10.1007/978-1-0716-0026-9_1.

[35] Halit Ongen et al. "Fast and efficient QTL mapper for thousands of molecular phenotypes". In: *Bioinformatics* 32.10 (Dec. 2015), pp. 1479–1485. DOI: 10.1093/bioinformatics/btv722. URL: https://doi.org/10.1093/bioinformatics/btv722.

[36] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria, 2019. URL: https://www.R-project.org/.

[37] Jeffrey S. Racine. "RStudio: A Platform-Independent IDE for R and Sweave". In: *Journal of Applied Econometrics* 27.1 (Oct. 2011), pp. 167–172. DOI: 10.1002/jae.1278. URL: https://doi.org/10.1002/jae.1278.

[38] Manfred Schwab, ed. *Encyclopedia of Cancer*. Springer Berlin Heidelberg, 2011. DOI: 10.1007/978-3-642-16483-5. URL: https://doi.org/10.1007/978-3-642-16483-5.

[39] A. A. Shabalin. "Matrix eQTL: ultra fast eQTL analysis via large matrix operations". In: *Bioinformatics* 28.10 (Apr. 2012), pp. 1353–1358. DOI: 10.1093/bioinformatics/bts163. URL: https://doi.org/10.1093/bioinformatics/bts163.

[40] Xinghua Mindy Shi, ed. *eQTL Analysis*. Springer US, 2020. DOI: 10.1007/978-1-0716-0026-9. URL: https://doi.org/10.1007/978-1-0716-0026-9.

[41] John D. Storey et al. *qvalue: Q-value estimation for false discovery rate control*. R package version 2.14.1. 2019. URL: http://github.com/jdstorey/qvalue.

[42] Duncan C. Thomas, Robert W. Haile, and David Duggan. "Recent Developments in Genomewide Association Scans: A Workshop Summary and Review". In: *The American Journal of Human Genetics* 77.3 (Sept. 2005), pp. 337–345. DOI: 10.1086/432962. URL: https://doi.org/10.1086/432962.

[43] Yihui Xie, J.J. Allaire, and Garrett Grolemund. *R Markdown: The Definitive Guide*. ISBN 9781138359338. Boca Raton, Florida: Chapman and Hall/CRC, 2018. URL: https://bookdown.org/yihui/rmarkdown.

[44] Zhihong Zhu et al. “Integration of summary data from GWAS and eQTL studies predicts complex trait gene targets”. In: *Nature Genetics* 48.5 (Mar. 2016), pp. 481–487. DOI: 10.1038/ng.3538. URL: https://doi.org/10.1038/ng.3538.