Full title:

yQTL Pipeline: a structured computational workflow for large scale quantitative trait loci discovery and downstream visualization

Short title:

Quantitative trait loci discovery analysis pipeline.

Authors:

Mengze Li [1 2], Zeyuan Song [3], Anastasia Gurinovich [4 5], Nicholas Schork [7], Paola Sebastiani [4 5 6], Stefano Monti* [1 2 3]

1: Bioinformatics Program, Faculty of Computing & Data Sciences, Boston University, Boston, MA, USA.

2: Section of Computational Biomedicine, School of Medicine, Boston University, Boston, MA, USA.

3: Department of Biostatistics, School of Public Health, Boston University, Boston, MA, USA.

4: Institute for Clinical Research and Health Policy Studies, Tufts Medical Center, Boston, MA, USA.

5: Department of Medicine, School of Medicine, Tufts University, Boston, MA, USA.

6: Data Intensive Study Center, Tufts University, Boston, MA, USA.

7: Quantitative Medicine and Systems Biology, The Translational Genomics Research Institute, Phoenix, AZ, USA.

1 Abstract

Quantitative trait loci (QTL) denote regions of DNA whose variation is associated with variations in quantitative traits. QTL discovery is a powerful approach to understand how changes in molecular and clinical phenotypes may be related to DNA sequence changes. However, QTL discovery analysis encompasses multiple analytical steps and the processing of multiple input files, which can be laborious, error prone, and hard to reproduce if performed manually. In order to facilitate and automate large-scale QTL analysis, we developed the *yQTL Pipeline*, where the '*y*' indicates the dependent quantitative variable being modeled.

Prior to genome-wide association test, the pipeline supports the calculation or the direct input of pre-defined genome-wide principal components and genetic relationship matrix when applicable. User-specified covariates can also be provided. Depending on whether familial relatedness exists among the subjects, genome-wide association tests will be performed using either a linear mixed-effect model or a linear model. Using the workflow management tool Nextflow, the pipeline parallelizes the analysis steps to optimize run-time and ensure results reproducibility. In addition, a user-friendly R Shiny App is developed to facilitate result visualization. Upon uploading the result file, it can generate Manhattan plots of user-selected phenotype traits and trait-QTL connection networks based on user-specified p-value thresholds.

We applied the *yQTL Pipeline* to analyze metabolomics profiles of blood serum from the New England Centenarians Study (NECS) participants. A total of 9.1M SNPs and 1,052 metabolites across 194 participants were analyzed. Using a p-value cutoff 5e-8, we found 14,983 mQTLs cumulatively associated with 312 metabolites. The built-in parallelization of our pipeline reduced the run time from ~90 min to ~26 min. Visualization using the R Shiny App revealed multiple mQTLs shared across multiple

metabolites. The *yQTL Pipeline* is available with documentation on GitHub at https://github.com/montilab/yQTL-Pipeline.

2 Introduction

Genetic association studies aim to test the correlation between disease risks or other phenotypes and genetic variation, with single-nucleotide polymorphisms (SNPs) the most widely used markers of such variation (1) (2). Quantitative trait loci (QTL) refer to those genetic variations that influence the level of a quantitative trait, for example, expression of a given gene (3).

Several analytical approaches for QTL discovery have been developed to date, examples including *Hail* (4), *MatrixeQTL* (5) and *QTLtools* (6). However, these tools do not fully account for familial relatedness, which is an essential component in many genetic association studies. *GENESIS* (7) is a package in R that performs genetic association tests while taking into account of familial relatedness, and has been extensively used in GWAS studies (8). Nevertheless, it can only accommodate one genotype input file and one phenotype at a time, thus its application to QTL discovery becomes inconvenient when faced with a large number of phenotypes and multiple input genotype files.

In addition to the association test, the complete QTL discovery workflow encompasses several preprocessing and post-analysis steps, including conversion of the input genotype file to the correct format, extraction of SNP missingness and frequency information, calculation of genetic principal components (PCs) and genetic relationship matrix (GRM), and merging and visualization of the QTL results. These steps require the execution of multiple commands implemented in different software packages, and can be error prone, time consuming, and difficult to reproduce. We previously developed a Nextflow-based pipeline that incorporates all these steps in a single, reproducible workflow (9).

However, this pipeline is limited to the analysis of one phenotype trait at a time. QTL analysis is often performed over multiple phenotypic traits and processes multiple genotype input files, and visualization of the results can be challenging since the relationship of a large number of genomic loci with multiple traits cannot be easily summarized.

To address these challenges, we developed the *yQTL Pipeline* to incorporate all the analysis steps into a single pipeline. It uses the workflow management tool Nextflow (10) to automate the entire workflow and enables the parallel execution of multiple processes whenever possible.

3 *Methods: yQTL Pipeline* Design

To ensure modularity, to minimize storage requirements and execution time, and to maximize user control of the analysis steps to be executed, the *yQTL Pipeline* workflow consists of three separate components (shown in **Fig 1**): *Prepare.nf, Analysis.nf, and Report.nf*.

Fig 1. The *yQTL Pipeline* workflow. The pipeline is split into three Nextflow steps: *Prepare.nf*, *Analysis.nf*, and *Report.nf*. Two alternative workflows are available for the cases when familial relatedness is present or not. Grey: inputs. Blue: analysis steps and intermediate outputs. Green: final outputs.

Prepare.nf performs any data pre-processing when needed, including the conversion of VCF genotype files to GDS format, and obtaining genetic PCs and GRM. Information about the genetic variants, including the allele information, allele frequency and missingness, are also extracted from the genotype data. Next, *Analysis.nf* can be invoked to perform the association test based on the input files either directly provided by the user or from the output of *Prepare.nf*. Finally, *Report.nf* merges the QTL results and generates the plots.

This modular design was in part adopted to take full advantage of Nextflow's features. Each Nextflow process first creates a copy of all input files into a "work" directory, which ensures reproducibility, but significantly increases the total execution time as well as the storage requirements, which can become a bottleneck when analyzing large datasets. This is particularly the case in QTL analysis, which takes large genotype input files, executes multiple steps, and generates large-sized result files. Splitting the workflow into three components significantly reduces the storage and execution footprints, since the input files can be submitted as "values" corresponding to their file paths, rather than actual "files" to be copied.

Throughout the entire pipeline, processes are executed in parallel whenever possible. Parallelization is an essential feature when analyzing a large number of quantitative traits, and/or when the genotype data is provided as multiple files. In large studies, it can translate into hundreds or thousands of independent batch jobs being submitted, which can be executed in parallel and thus highly decrease the run time.

For the configuration and execution of the *yQTL Pipeline,* all that is needed is for the user to specify a configuration file listing the input files and parameters, and to submit three command lines to invoke the entire pipeline. The *yQTL Pipeline* is released under a General Public License 3.0 license. It is publicly available at https://github.com/montilab/yQTLpipeline, including comprehensive documentations of the configuration setup. It supports Linux and OS X operating systems.

3.1 Input, configuration, and preparation

The required inputs for the *yQTL Pipeline* include genotype and phenotype data. Optionally, covariates, genetic PCs and GRM can also be included. A more detailed description of each of the input parameters is provided in the GitHub documentation.

3.1.1 Genotype data

The pipeline supports either VCF or GDS input format for genotype data. If VCF files are provided, these will be converted to GDS format by running *Prepare.nf*. In addition, the user can specify whether to use the imputed dosage entry or the genotype count entry.

3.1.2 Phenotype and covariates data

Phenotype and covariates data should be entered as a data frame in either RDS (R Data Serialization), CSV (comma separated text file) or TXT (tab separated text file) format, with rows denoting samples and columns denoting the phenotypes to identify QTLs from (i.e., the 'y' in the model). There should be a column named "sample.id" to be matched with sample ids in the genetic data files. In addition, the user needs to input a text file that contains all the phenotype trait names to analyze, corresponding to the column names in the phenotype file. The user can specify both numerical and categorical covariates to include.

Genetic PCs, as well as GRM when familial relatedness is presented in the data, can be estimated using different types of computational tools. The *yQTL pipeline* applies PC-AiR (11) and PC-Relate (12) to perform the tasks and is achieved by running *Prepare.nf*. Alternatively, if pre-calculated genetic PCs and GRM are available, they can be provided as RDS-formatted input files.

3.1.3 An option to analyze a subset of samples and/or SNPs

By default, the pipeline will perform the analysis using all the samples and all SNPs available in the intersection of all input data files. Alternatively, the analysis can be restricted to a subset of samples and/or a subset of SNPs as specified in user-provided input text files listing the sample and SNP IDs.

3.1.4 Control Nextflow processes

Nextflow supports the dispatch of multiple processes in parallel, a feature that can significantly reduce execution time. The user can control the maximum number of processes to run concurrently in the configuration file. When running the pipeline on a high-performance shared computer cluster, the user can also specify distinct resource allocation requirements for each of the pipeline steps in the *SGE* (Sun Grid Engine) configuration file. This is an important feature, as different steps may require drastically different computational resources, and the tailored resource allocation ensures the efficient use of computational (memory and CPU) resources.

3.1.5 Plotting parameters

Following the completion of QTL analysis, the *yQTL Pipeline* will generate the Manhattan plots and QQ (quantile-quantile) plots for each of the phenotypes. The user can specify the minor allele count (MAC) threshold for the SNPs to be included, as well as the resolution and size of the plots. This MAC threshold only affects the plotting and will not filter any of the output QTL results.

3.2 QTL analysis workflows

The *yQTL Pipeline* supports two alternative analysis modalities implemented in separate workflows, with the choice to be specified in the parameter “params.pipeline_engine”. Available options are “genesis” and “matrixeqtl”. The details of each are discussed next.

3.2.1 Workflow 1: data with familial relatedness

When there is known familial relatedness, the user can select *workflow 1* (**Fig 1**, left side), by setting params.pipeline_engine = “genesis” or “g”, which is based on *GENESIS*, and uses a two-step procedure. First, it estimates a “null model” representing the fixed effect of all covariates provided. It then performs association testing for each SNP using a linear mixed-effect model.

GENESIS takes a single phenotype, a single genotype file, covariates and a GRM as input. Thus, the pipeline first splits the one multi-phenotype input file into as many single phenotype files, then submits multiple jobs in parallel corresponding to each of the phenotypes and each of the input genotype data files. For instance, if the user wishes to analyze 100 phenotypes and the genotype data is provided as 22 GDS files, corresponding to as many chromosomes, then 2,200 processes will be automatically submitted and run in parallel. The same covariates, PCs and GRM are used across all those processes.

3.2.2 Workflow 2: data without familial relatedness

When the genotype data represent profiles from unrelated samples, the user can opt for *workflow 2* (**Fig 1**, right side), achieved by setting params.pipeline_engine = “matrixeqtl” or “m”, to take advantage of *MatrixeQTL*’s greater efficiency (5). *MatrixeQTL* performs the association test of all input phenotypes with each genetic input file using a linear model. Although there is no set upper limit on how many phenotypes *MatrixeQTL* can handle at once, as the number of phenotypes and the size of the genotype data increase, the required memories increase substantially and may exceed the available resources. To circumvent this problem, the phenotype file will be split into multiple “chunks”, with each chunk containing a subset of phenotypes. The user can control the number of phenotypes included in each chunk to balance the memory requirement and total analysis time. The pipeline will then apply

MatrixeQTL to each phenotype chunk with each genotype input file in parallel. For example, if there are 100 phenotypes, 22 genotype data input files, and a user-specified chunk size of 30 (i.e., 30 phenotypes in each chunk), there would be 4 chunks in total with one chunk containing the last 10 phenotypes, and 88 parallel processes would be submitted. The same covariates are used with all those processes.

3.3 Outputs

The intermediate results and the final outputs of the pipeline are saved to separate folders. Log files of all analysis steps are also saved.

1. "1_data" and "1_phenotype_data" (or "1_phenotype_data_chunk") folders contain all data used, including the GDS version of the genotype data if the original inputs were VCF files, and covariate and phenotype data, respectively.
2. "2_SNP_info" folder contains the SNP information, such as allele, missingness and frequency.
3. "3_individual_results" folder contains the QTL results of each phenotype with each genetic data file.
4. "4_ individual_results_SNPinfo" folder is the combination of the two intermediate results above.
5. "5_Results_Summary" folder contains the final output, which includes the merged version of all the QTL results of each of the phenotype including SNP information, a summary table of the number of QTLs identified, as well as the QQ plots and Manhattan plots of each of the phenotype traits. Since QTL results are often large data frames, the results are output in RDS format. In addition, the user can setup the configuration file to output QTL results in comma separated text files (CSV format) besides RDS files.

3.4 Downstream Visualization

We developed an R Shiny App to facilitate post-analysis visualization. In the R Shiny App interface, the user can upload the RDS file generated by the pipeline, or an RDS file in a similar format, i.e., a data frame reporting the phenotype trait names, QTL names, their chromosomal coordinates, and their p-values.

The Manhattan plot is one of the most used visualization methods for GWAS analysis since it enables the intuitive identification of significant genetic associations. After uploading the QTL results file, the Manhattan plot of a specific phenotype trait is generated by selecting a phenotype trait name from the drop-down menu in the R Shiny App interface. In addition, the user can specify a list of SNP IDs in a text input area, separated by comma, to obtain the filtered QTL result table of those SNPs. If a specific phenotype is selected, only results from this phenotype will be returned. Alternatively, the user can select the option “All phenotypes” in the drop-down menu to display results from all phenotypes.

Manhattan plots can only visualize the results for a single phenotype trait, thus making the comparison across phenotypes difficult. To compare QTL results between multiple phenotype traits, the R Shiny App can also visualize a trait-QTL network. The nodes in the network represent phenotype traits, QTL names (e.g., SNP IDs), and chromosome names. The edges represent significant associations between traits and their QTLs, and top QTLs’ co-localization within the same chromosome. The user can specify a p-value threshold, and the trait-QTL network will be generated including only QTLs reaching the threshold. For each phenotype trait, given the large number of adjacent genetic loci in high linkage disequilibrium (LD) with each other, only the most significant genetic locus on each chromosome will be included in the network plot. The resulting network thus displays which phenotype traits have QTLs identified at the selected p-value threshold, which chromosomes those QTLs are in, and whether phenotype traits are sharing (some of) the same QTLs.

4 Results and Discussions: A Metabolomics Use Case of the *yQTL Pipeline*

We will illustrate the application of the *yQTL Pipeline* to paired metabolomics and genotype datasets from the New England Centenarians Study (NECS). These datasets profiled 194 NECS participants described in (13). Age, gender, and years of education were used as covariates. 1,052 metabolites with less than 20% missing values were selected and their expression values were natural log transformed. 9.1M SNPs in the genotype data were used. Since the participants are not genetically related, the pipeline was setup to run with *workflow 2*, in which the linear model implemented in *MatrixeQTL* was applied and samples were considered as independent. The p-value cutoff was set to 1e-3. Since the dataset had previously estimated genetic PCs and the genotype data was already in GDS format, only *Analysis.nf* and *Report.nf* were executed.

Although all 9.1M SNPs were analyzed, to avoid artifacts caused by extremely rare SNPs, only the results from the 3.2M SNPs that have MAC ≥ 3 were considered in the following post-GWAS analysis. At the relaxed p-value threshold of 1e-3, all 1,052 metabolites had mQTLs identified. At the genome wide significance threshold (p-value < 5e-8), the list reduced to 312 metabolites. The latter threshold yielded 14,983 mQTLs, including 11,931 unique SNPs, with 3,052 of them being mQTLs shared by at least two metabolites.

Fig 2 shows the Manhattan plot of metabolite N2-acetyl,N6-methyllysine, which is part of the *yQTL Pipeline* output, but can also be generated using the companion R Shiny App. Two genomic loci at chromosomes 2 and 10 were identified at genome-wide significance level (p < 5e-8).

Fig 2. Example Manhattan plot from the R Shiny App. Manhattan plot of N2-acetyl,N6-methyllysine mQTL analysis based on the New England Centenarians Study (NECS) dataset. Minor allele count (MAC) cutoff ≥ 3 was applied to avoid artifacts caused by rare SNPs. Two genome-wide signals on chromosome 2 and chromosome 10 are clearly visible.

Fig 3 illustrates an example of the trait-QTL network generated by the R Shiny App using the most significant mQTLs obtained (p < 1e-17), which reveals information that would not be easily captured by single phenotype trait visualization methods, such as Manhattan plots. For instance, the network visualization makes it clear that while rs4539242 (bottom of **Fig 3**) is one of the top QTL associations of N2-acetyl,N6,N6-dimethyllysine, it is also the top QTL of N6-methyllysine. Meanwhile, orotidine (right of **Fig 3**) has QTLs with p < 1e-17 on both chromosome 14 (top QTL rs192581407) and chromosome 20 (top QTL rs541005701). On the chromosome level, rs768854100 (middle left of **Fig 3**) on chromosome 10 is the top QTL of undecanedioate, while a few other SNPs on the same chromosome are also the top QTL of other metabolites.

Fig 3. Example network plot from the R Shiny App. Results of the New England Centenarians Study (NECS)-based mQTL analysis using a p < 1e-17 threshold are shown. Shared top mQTLs between different metabolites, as well as top mQTLs from different metabolites on the same chromosome are displayed.

If running all analyses sequentially, the total execution time for this example exceeded 90 minutes. Thanks to *the yQTL Pipeline*’s built-in parallelization, the total run time was reduced to 26 minutes, achieving a ~3.5-fold speed-up. In this example, the memory of the compute nodes ranges from 4GB to

32GB, tailored to the requirements of each of the processes. With larger datasets, and when modeling familial relatedness, the execution time reduction would be substantially larger.

5 Conclusions

The tools described and results presented provide strong evidence for the usefulness of the *yQTL Pipeline*. By streamlining the analysis process, increasing parallelization, and improving reproducibility of results, and by incorporating multiple steps into rigorously tested and well-documented wrapper workflows, the pipeline will contribute to lowering the barrier to the wide adoption of QTL analysis tools by the research community.

6 Acknowledgements

We express our sincere gratitude for the funding provided by the National Institute on Aging (NIA), including: U19-AG063893 (PS), UH3-AG064704 (PS, SM), UH3-AG064706 (NS), and U19-AG023122 (PS, SM, NS).

7 References

1. Lewis CM, Knight J. Introduction to Genetic Association Studies. Cold Spring Harb Protoc. 2012 Mar 1;2012(3):pdb.top068163-pdb.top068163.

2. Tam V, Patel N, Turcotte M, Bossé Y, Paré G, Meyre D. Benefits and limitations of genome-wide association studies. Nat Rev Genet. 2019 Aug;20(8):467–84.

3. Montgomery SB, Dermitzakis ET. From expression QTLs to personalized transcriptomics. Nat Rev Genet. 2011 Apr;12(4):277–82.

4. Hail Team. Hail 0.2.13-81ab564db2b4. https://github.com/hail-is/hail/releases/tag/0.2.13.

5. Shabalin AA. Matrix eQTL: ultra fast eQTL analysis via large matrix operations. Bioinformatics. 2012 May 15;28(10):1353–8.

6. Delaneau O, Ongen H, Brown AA, Fort A, Panousis NI, Dermitzakis ET. A complete tool set for molecular QTL discovery and analysis. Nat Commun. 2017 Aug;8(1):15452.

7. Gogarten SM, Sofer T, Chen H, Yu C, Brody JA, Thornton TA, et al. Genetic association testing using the GENESIS R/Bioconductor package. Valencia A, editor. Bioinformatics. 2019 Dec 15;35(24):5346–8.

8. Gurinovich A, Song Z, Zhang W, Federico A, Monti S, Andersen SL, et al. Effect of longevity genetic variants on the molecular aging rate. GeroScience [Internet]. 2021 May 4 [cited 2021 May 16]; Available from: https://link.springer.com/10.1007/s11357-021-00376-4

9. Song Z, Gurinovich A, Federico A, Monti S, Sebastiani P. nf-gwas-pipeline: A Nextflow Genome-Wide Association Study Pipeline. J Open Source Softw. 2021 Mar 2;6(59):2957.

10. Di Tommaso P, Chatzou M, Floden EW, Barja PP, Palumbo E, Notredame C. Nextflow enables reproducible computational workflows. Nat Biotechnol. 2017 Apr;35(4):316–9.

11. Conomos MP, Miller MB, Thornton TA. Robust Inference of Population Structure for Ancestry Prediction and Correction of Stratification in the Presence of Relatedness. Genet Epidemiol. 2015 May;39(4):276–93.

12. Conomos MP, Reiner AP, Weir BS, Thornton TA. Model-free Estimation of Recent Genetic Relatedness. Am J Hum Genet. 2016 Jan;98(1):127–48.

13. Sebastiani P, Song Z, Ellis D, Tian Q, Schwaiger-Haber M, Stancliffe E, et al. A metabolomic signature of the APOE2 allele. GeroScience. 2023 Feb;45(1):415–26.

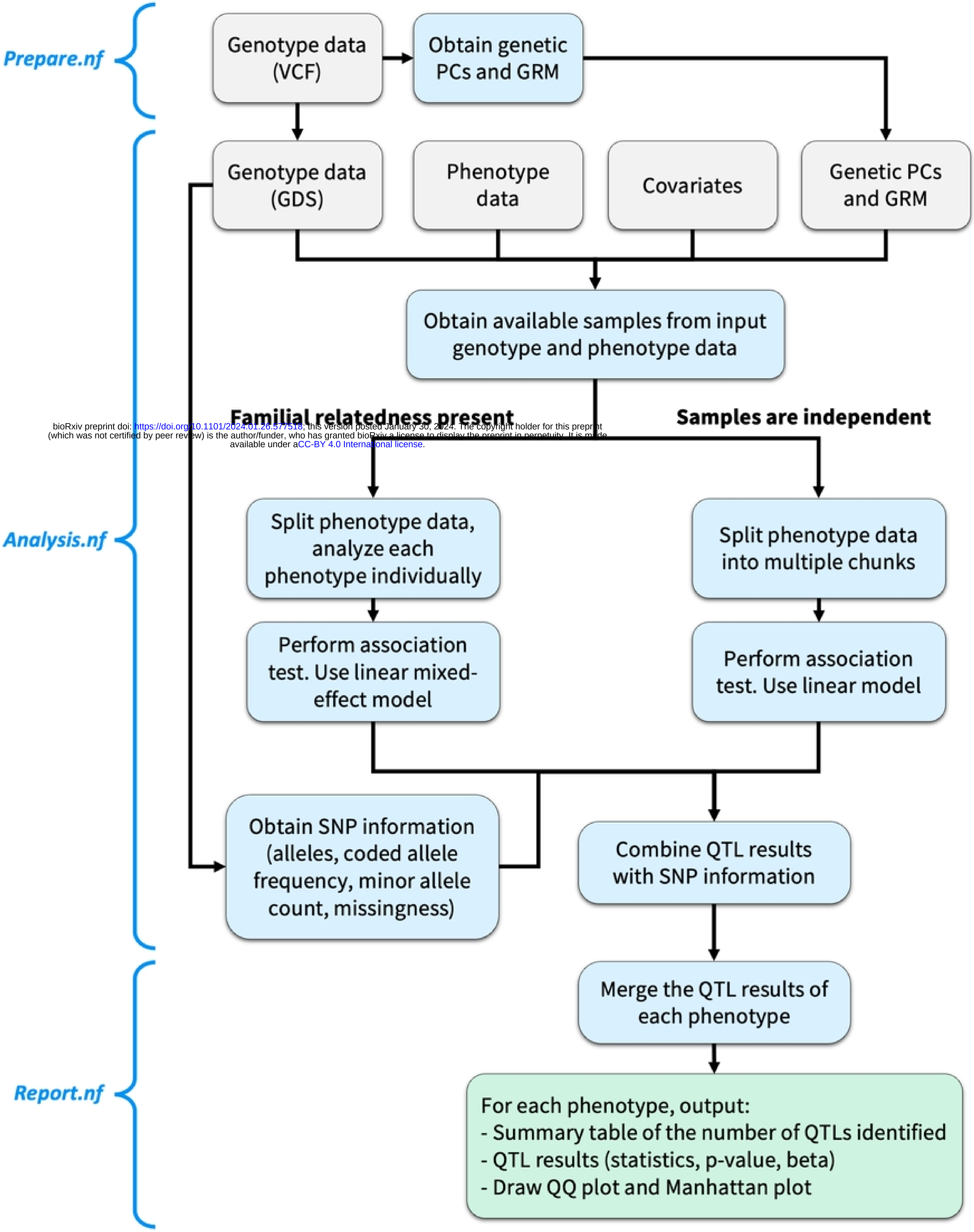
Prepare.nf
Genotype data (VCF)
Obtain genetic PCs and GRM
Analysis.nf
Genotype data (GDS)
Phenotype data
Covariates
Genetic PCs and GRM
Obtain available samples from input genotype and phenotype data
Familial relatedness present
Samples are independent
Split phenotype data, analyze each phenotype individually
Split phenotype data into multiple chunks
Perform association test. Use linear mixed-effect model
Perform association test. Use linear model
Obtain SNP information (alleles, coded allele frequency, minor allele count, missingness)
Combine QTL results with SNP information
Merge the QTL results of each phenotype
Report.nf
For each phenotype, output:
- Summary table of the number of QTLs identified
- QTL results (statistics, p-value, beta)
- Draw QQ plot and Manhattan plot

Fig 1

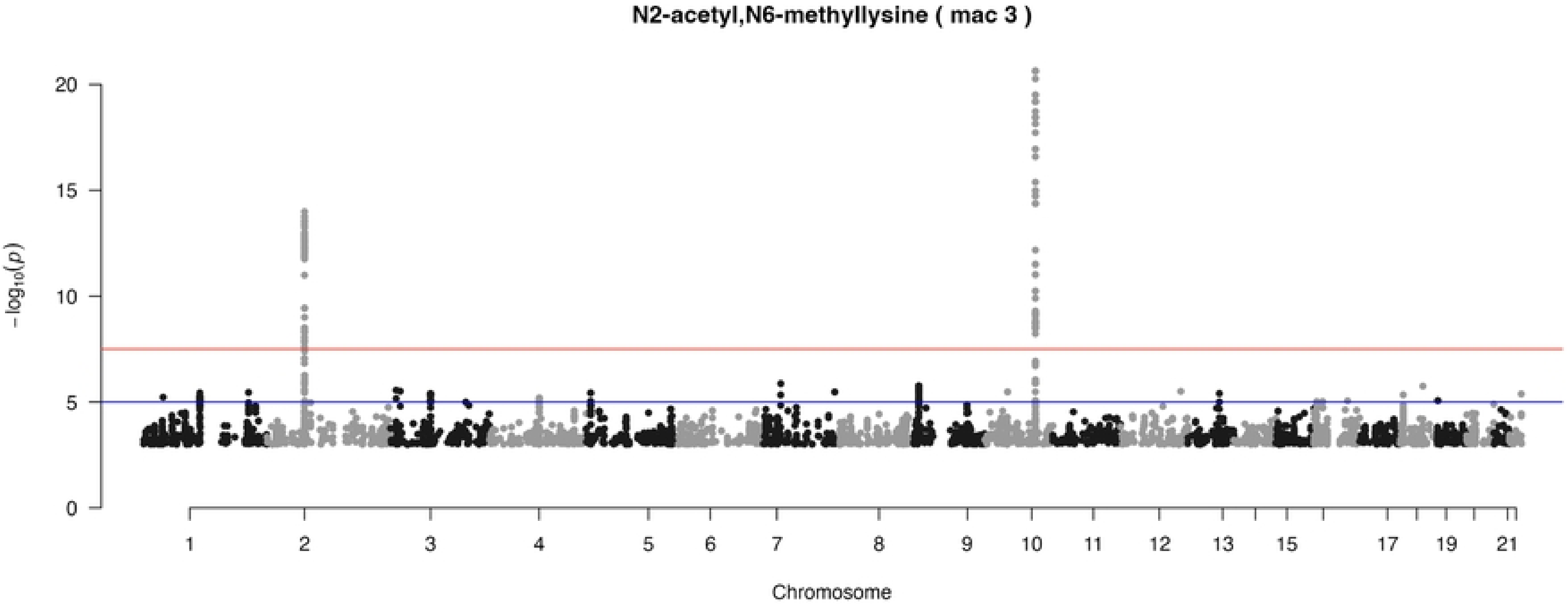
N2-acetyl,N6-methyllysine (mac 3)
-log10(p)
0
5
10
15
20
1
2
3
4
5
6
7
8
9
10
11
12
13
15
17
19
21
Chromosome

Fig 2

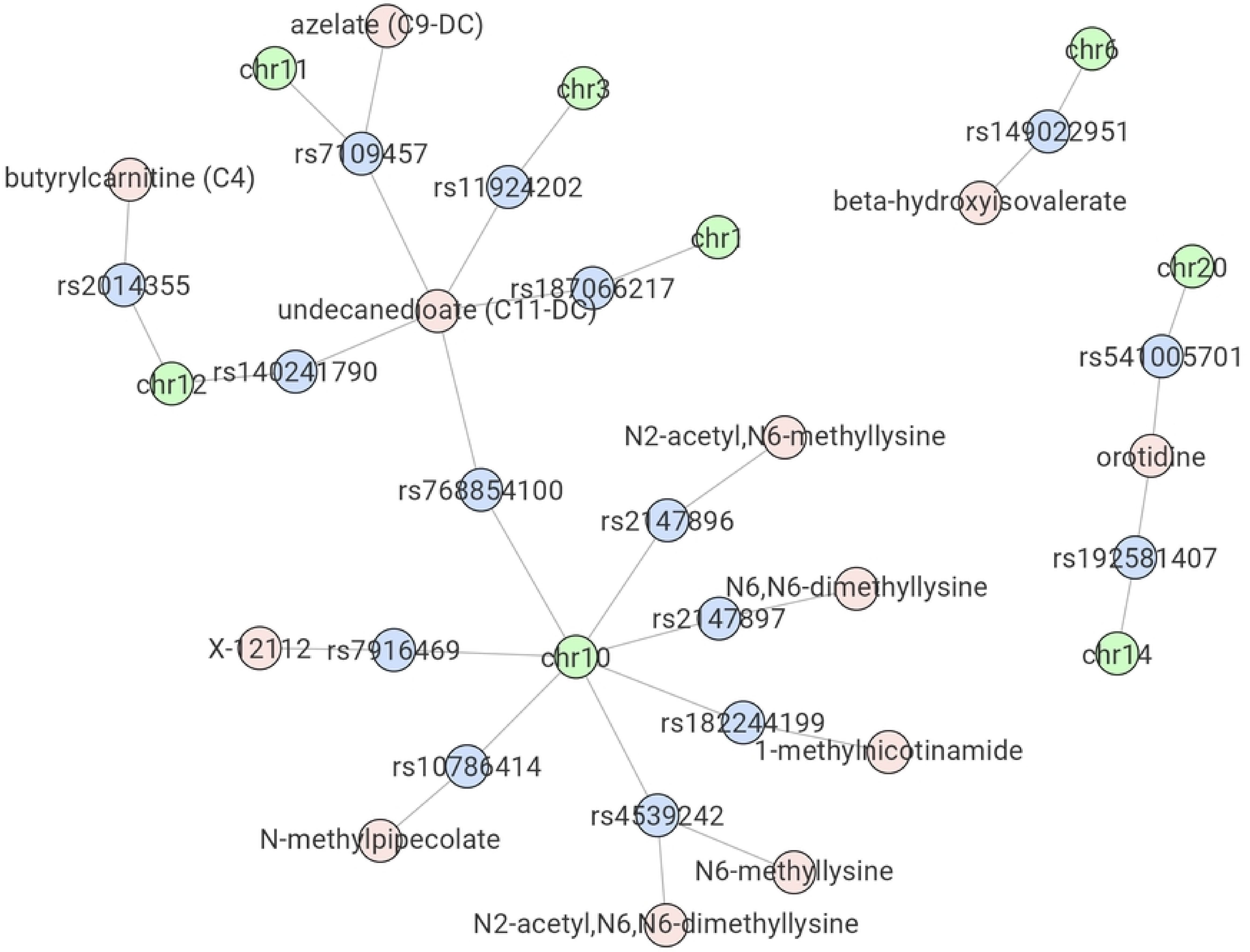
azelate (C9-DC)
chr11
chr3
chr6
rs149022951
rs7109457
rs11924202
butyrylcarnitine (C4)
beta-hydroxyisovalerate
chr1
chr20
rs2014355
rs187066217
undecanedioate (C11-DC)
rs541005701
chr12
rs140241790
N2-acetyl,N6-methyllysine
orotidine
rs768854100
rs2147896
rs192581407
N6,N6-dimethyllysine
rs2147897
X-12112
rs7916469
chr10
chr14
rs182244199
1-methylnicotinamide
rs10786414
rs4539242
N-methylpipecolate
N6-methyllysine
N2-acetyl,N6,N6-dimethyllysine

Fig 3