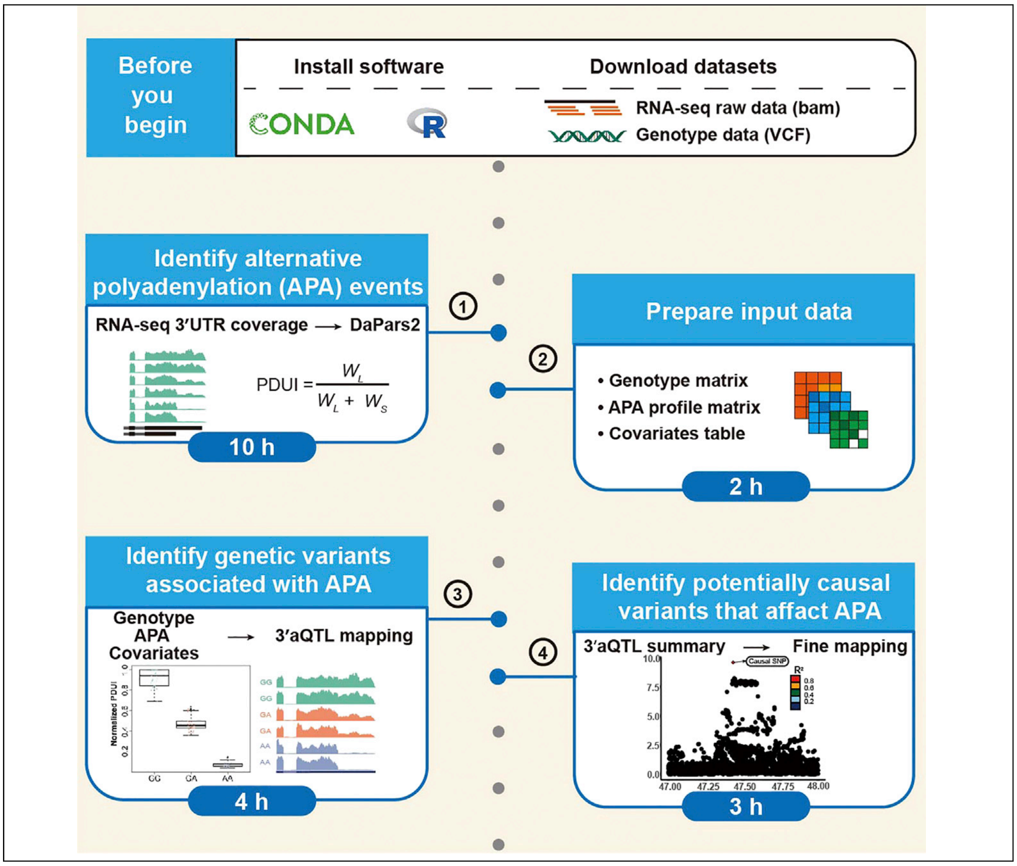
STAR Protocols

Protocol

# Using population-scale transcriptomic and genomic data to map 3′ UTR alternative polyadenylation quantitative trait loci



Xudong Zou, Ruofan Ding, Wenyan Chen, ..., Qin Wang, Wei Li, Lei Li

wei.li@uci.edu (W.L.)
lei.li@szbl.ac.cn (L.L.)

## Highlights

Identification and quantification of APA events across multiple RNA-seq samples

3aQTL-pipe facilitates identification of APA-associated genetic variants

Identification of potential causal SNPs for dynamics APA events by fine mapping

3′ UTR alternative polyadenylation (APA) quantitative trait loci (3′aQTL) can explain approximately 16.1% of trait-associated non-coding variants and is largely distinct from other molecular QTLs. Here, we describe a bioinformatic protocol for identifying 3′aQTLs through standard RNA-seq and matched genomic data. This protocol allows users to analyze dynamic APA events, identify common genetic variants associated with differential 3′ UTR usage, and predict the potential causal variants that affect APA.

Publisher's note: Undertaking any experimental protocol requires adherence to local institutional guidelines for laboratory safety and ethics.

Zou et al., STAR Protocols 3, 101566
September 16, 2022 © 2022 The Author(s).
https://doi.org/10.1016/j.xpro.2022.101566

CellPress
OPEN ACCESS

Protocol

# Using population-scale transcriptomic and genomic data to map 3′ UTR alternative polyadenylation quantitative trait loci

Xudong Zou,[1,5] Ruofan Ding,[1,5] Wenyan Chen,[1] Gao Wang,[2] Shumin Cheng,[1] Qin Wang,[3] Wei Li,[4,*] and Lei Li[1,6,7,*]

[1]Institute of Systems and Physical Biology, Shenzhen Bay Laboratory, Shenzhen 518055, China

[2]The Gertrude H. Sergievsky Center and the Department of Neurology, Columbia University, New York, NY 10032, USA

[3]Shenzhen Bay Laboratory Supercomputing Center, Shenzhen 518055, China

[4]Division of Computational Biomedicine, Department of Biological Chemistry, School of Medicine, The University of California, Irvine, CA 92697, USA

[5]These authors contributed equally

[6]Technical contact

[7]Lead contact

*Correspondence: wei.li@uci.edu (W.L.), lei.li@szbl.ac.cn (L.L.)

https://doi.org/10.1016/j.xpro.2022.101566

## SUMMARY

**3′ UTR alternative polyadenylation (APA) quantitative trait loci (3′aQTL) can explain approximately 16.1% of trait-associated non-coding variants and is largely distinct from other molecular QTLs. Here, we describe a bioinformatic protocol for identifying 3′aQTLs through standard RNA-seq and matched genomic data. This protocol allows users to analyze dynamic APA events, identify common genetic variants associated with differential 3′ UTR usage, and predict the potential causal variants that affect APA.**

**For complete details on the use and execution of this protocol, please refer to Li et al. (2021).**

## BEFORE YOU BEGIN

Here we describe the basic hardware requirements, installation of software and tools, and the test dataset. The protocol is not limited to the current human genome, but is also open to other human reference genome if available.

> *Note:* Throughout this protocol, each line of executable code is preceded by a greater than sign (>).

### Hardware

A workstation or computer cluster running a POXIS system (Unix, Linux, or macOS) is required (we used the Linux distribution Ubuntu 18.04.6 LTS). The minimum requirements for the test dataset are an 8-core processor, 16 GB of RAM, and 1 TB of hard disk space.

### Download and install software and tools

⏱ Timing: 30 min

The following steps are to download and install software and tools for executing our pipeline. If users have no permission or other reasons that cannot install any software/tools on the machine, we recommend contacting the machine administrator. Alternatively, we provided a docker image that includes all required software/tools at https://hub.docker.com/r/3utr/3aqtl_pipe/tags.

1. Download and install Anaconda3 from https://repo.anaconda.com/archive/.

```
> bash Anaconda3-2021.11-Linux-x86_64.sh
```

*Note:* The installer script "Anaconda3-2021.11-Linux-x86_64.sh" is available at https://repo.anaconda.com/archive/. Anaconda3 will be installed in $HOME/anaconda3/ by default.

*Note:* Python3 and related packages like NumPy, Scipy, and Conda are involved in Anaconda3.

2. Install required software and tools with conda.

```
> conda install -c conda-forge r-base

> conda install -c conda-forge r-dplyr

> conda install -c bioconda r-optparse

> conda install -c bioconda Bioconductor-impute

> conda install -c bioconda r-peer

> conda install -c bioconda r-matrixeqtl

> conda install -c bioconda bedtools

> conda install -c bioconda plink=1.90

> conda install -c bioconda samtools

> conda install -c bioconda vcftools
```

3. Install susieR with the R command below.

```
> install.packages(‘‘susieR’’)
```

*Alternatives:* Install the latest version of susieR from GitHub:

```
> remotes::install_github("stephenslab/susieR")
```

4. Download the source code of 3aQTL-pipe from GitHub.

```
> git clone https://github.com/3UTR/3aQTL-pipe.git
```

*Note:* This command will copy a directory "3aQTL-pipe". All scripts related to this pipeline can be found in the subdirectory "src".

*Note:* Scripts for running Dapars2 are also involved in directory "src".

*Alternatives:* Users can also use the 3′aQTL pipeline through our prebuilt docker image https://hub.docker.com/r/3utr/3aqtl_pipe/tags. A simple tutorial can be found on the wiki page: https://github.com/3UTR/3aQTL-pipe/wiki.

## Initial setup

**Timing: >1 d (depends on the download speed of the test dataset)**

This protocol was tested on an example dataset containing RNA sequencing and genotype data of 89 samples from 1000 Genomes Project (see "Key resources table"). The following steps describe downloading the test dataset and preparing a working directory.

5. Set up a working directory and copy source codes of the 3′aQTL pipeline into the directory.

```
> work_directory=/PATH/TO/WORK_DIRECTORY

> mkdir -p $work_directory && cd $work_directory

> mkdir -p ${work_directory}/data

> cp -r /PATH/TO/3aQTL-pipe/src ./
```

6. Download the test dataset and save them to ${work_directory}/data directory.

```
> cd ${work_directory}/data

> wget -i test_data_RNA.links.txt

> wget -i test_data_genotype.links.txt
```

*Note:* The text files "test_data_RNA.links.txt" and "test_data_genotype.links.txt" contain links for RNA-seq bam files and genotype (VCF) files. Both files are available at https://github.com/3UTR/3aQTL-pipe.

7. Create two files, "sample_list.txt" and "vcf_list.txt," that contain all bam files and genotype file(s).

```
> cd ${work_directory}

>   for   bam   in   ‘ls   ./data/*.bam‘;do   f=‘basename   $bam‘;s=${f%%.*};echo   -e
‘‘${f}\t$bam’’;done > sample_list.txt

> ls ./data/*.vcf* > vcf_list.txt
```

## Vocabularies and abbreviations

In this protocol, APA is the abbreviation of "alternative polyadenylation" and the 3′UTR alternative polyadenylation quantitative trait loci (3′aQTL) represents common genetic variants associated with the 3′UTR usage of target genes.

## KEY RESOURCES TABLE

| REAGENT or RESOURCE | SOURCE | IDENTIFIER |
|---|---|---|
| Deposited data | | |
| RNA-seq data of lymphoblastoid cell lines (LCL) from 89 individuals (Lappalainen et al., 2013) | Geuvadis RNA-seq Project | https://www.ebi.ac.uk/arrayexpress/experiments/E-GEUV-1/files/processed/?ref=E-GEUV-1 |

(*Continued on next page*)

*Continued*

| REAGENT or RESOURCE | SOURCE | IDENTIFIER |
|---|---|---|
| Individual genotype data (1000 Genomes Project Consortium, 2015) | 1000 Genomes Project | https://www.ebi.ac.uk/arrayexpress/experiments/E-GEUV-1/files/genotypes/?ref=E-GEUV-1 |
| 3′UTR reference annotation | UCSC build 37 | http://genome.ucsc.edu/ |
| Software and algorithms | | |
| R | CRAN | 3.6.1 |
| Python | Anaconda3 | 3.9.11 |
| Dapars2 (Feng et al., 2018) | 2.1 | |
| 3aQTL-pipe | https://github.com/3UTR/3aQTL-pipe;<br>https://doi.org/10.5281/zenodo.6658555 | v1.1 |
| Matrix eQTL (Shabalin, 2012) | http://www.bios.unc.edu/research/genomic_software/Matrix_eQTL/ | 2.3 |
| susieR (Wang et al., 2020) | https://github.com/stephenslab/susieR | 0.11.42 |
| impute | Bioconductor | 1.60.0 |
| dplyr | CRAN | 1.0.8 |
| optparse | CRAN | 1.7.1 |
| PLINK (Purcell et al., 2007) | https://www.cog-genomics.org/plink/1.9 | 1.90 |
| vcftools (Danecek et al., 2011) | https://github.com/vcftools/vcftools | 0.1.16 |
| bedtools (Quinlan and Hall, 2010) | https://github.com/arq5x/bedtools2/releases | 2.30.0 |
| samtools (Danecek et al., 2021) | http://www.htslib.org/ | 1.7 |
| PEER (Stegle et al., 2012) | https://github.com/downloads/PMBio/peer/R_peer_source_1.3.tgz | 1.3 |
| Other | | |
| Recommended hardware: workstation or computer cluster with 8 physical core system; 16 GB of RAM; 1TB of hard disk; Ubuntu 18.04 operating system. | N/A | N/A |

## STEP-BY-STEP METHOD DETAILS

Herein we describe a step-by-step pipeline for analyzing APA events across multiple samples and mapping the associations between genetic variants and APA usage (3′aQTLs). We will start by preparing input data, then perform multi-samples APA quantification, covariates analysis, 3′aQTL mapping, and finally, the fine mapping of 3′aQTL. To illustrate these steps, we use the RNA-seq data from Geuvadis Project (Lappalainen et al., 2013) and matched genotype data from the 1000 Genomes Project (1000 Genomes Project Consortium, 2015) as an example. The dataset contains 89 samples from CEU populations.

We prepared a GitHub repository (https://github.com/3UTR/3aQTL-pipe) containing all the required scripts.

### Prepare required input data for Dapars2

> ⏱ **Timing: 6 h when using eight threads to process 89 bam files from the CEU population (the run time depends on sample size and the number of threads used)**

The following steps prepare the required input data for analyzing APA with Dapars2 (as shown in the table below).

**Required input data for running Dapars2**

| Data description | Example(s) | Location |
|---|---|---|
| bedgraph files of all samples | HG00123.wig; HG00124.wig | ./wig/ |
| Annotated 3′UTR region in bed format | refseq_3utr_annotation.bed | current directory |

(*Continued on next page*)

*Continued*

| Data description | Example(s) | Location |
|---|---|---|
| a tab-delimited including the name of bedgraph files and sequencing depth | wigFile_and_readDepth.txt | current directory |
| DaPars2 configure file | Dapars2_running_configure.txt | current directory |

1. Extract the RefSeq gene annotation and symbol from the UCSC table browser (Figure 1).

   *Note:* The two files generated in this step, as shown in Figure 1, are "hg19_refseq_whole_gene.bed" and hg19_refseq_id_to_symbol.txt".

   ⚠ CRITICAL: This protocol takes human reference genome hg19 for the example dataset; if users choose other genome versions or genomes for other species, please replace "GRCh37/hg19" (see "assembly" in the highlighted box in Figure 1) as their chosen genome version. One thing that needs to be noted is that the selected genome version should be consistent with the one used in RNA-seq reads alignment.

2. Generate all required input files for running Dapars2.

```
> bash ./src/prepare_inputs_for_apa_quant.sh -s sample_list.txt \

-g hg19_refseq_whole_gene.bed \

-r hg19_refseq_id_to_symbol.txt
```

   *Note:* This command will generate multiple bedgraph files with names suffixed by ".wig" in the "./wig" directory, a file containing annotated 3′UTR regions named "refseq_3utr_annotation.bed", and a file named "wigFile_and_readDepth.txt" includes the list of bedgraph files, and a configure file named "Dapars2_running_configure.txt".

   ⚠ CRITICAL: This step may take a long time if hundreds of samples need to be processed. Users can assign a larger number for the option "-t" (which was 8 in default).

### Identify and quantify APA events with DaPars2

⏱ Timing: 4 h

Depends on the size of each chromosome, the number of chromosomes, and the number of threads. This run time was estimated by analyzing 24 human chromosomes with eight threads.

3. Run Dapars2 analysis on a specific chromosome.

```
> python ./src/Dapars2_Multi_Sample.py Dapars2_running_configure.txt chr1
```

   *Note:* This command only analyzes chromosome 1. To complete the whole-genome analysis, repeat this command for all chromosomes by changing the second parameter accordingly. This will output the directories with the suffix "_chr1" and a file "Dapars2_result_temp.chr1.txt" which contains all estimated APA events in chromosome 1. Each row of this result file denotes one transcript, and column 1 to 5 involves transcript information and APA information. Column 6 to the last column represents the APA usage across samples.

*Alternatives:* Users can also analyze multiple chromosomes automatically using the script "DaPars2_Multi_Sample_Multi_Chr.py".

4. Merge the Dapars2 results.

```
> cat refseq_3utr_annotation.bed | cut -f 1|sort|uniq |grep -v ‘‘MT’’ > chrList.txt

> Rscript ./src/merge_apa_quant_res_by_chr.R -c chrList.txt
```

*Note:* This R script generates a file named "Dapars2_res.all_chromosomes.txt" by default.

## Prepare input data for 3′aQTL mapping

**Timing: 2 h**

The following step prepares the required input data (as shown in the table below) for 3′aQTL mapping.

| Data description | Example | Location |
|---|---|---|
| Phenotype data that includes APA usage of all transcripts across samples. | Phenotype_matrix.txt | ./Matrix_eQTL |
| Genotype data that includes genotype value of all SNPs across samples. | Genotype_matrix.txt | ./Matrix_eQTL |
| Covariates that contain known and inferred technical covariates | Covariate_matrix.txt | ./Matrix_eQTL |
| 3′UTR location file | 3UTR_location.txt | ./Matrix_eQTL |
| SNP location file | snp_location.txt | ./Matrix_eQTL |

Example of phenotype data. The format of the APA usage profile should also be a *G* × *N* matrix, where G denotes the number of transcripts and N is the number of individuals.

| id | NA06984 | NA06985 | NA06986 | NA06989 | NA06994 |
|---|---|---|---|---|---|
| NM_001256456.2\|INTS11\|chr1\|- | 0.6 | 0.82 | 0.98 | 1.00 | 0.98 |
| NM_006694.4\|JTB\|chr1\|- | 0.27 | 0.34 | 0.31 | 0.29 | 0.31 |
| NM_001319245.2\|MAST2\|chr1\|+ | 0.97 | 1.00 | 0.88 | 1.00 | 1.00 |
| NM_001282861.2\|GON4L\|chr1\|- | 0.31 | 0.37 | 0.27 | 0.42 | 0.54 |
| … | … | … | … | … | … |

Example of genotype data. The format should be a *S* x *N* matrix, where *S* denotes the number of SNPs and *N* denotes the number of individuals. The genotype was encoded as "0" or "1" or "2".

| id | NA06984 | NA06985 | NA06986 | NA06989 | NA06994 |
|---|---|---|---|---|---|
| chr1_52238_T_G | 2 | 2 | 2 | 2 | 2 |
| chr1_57952_A_C | 2 | 2 | 2 | 2 | 2 |
| chr1_69511_A_G | 2 | 2 | 1 | 2 | 2 |
| chr1_233474_C_G | 0 | 0 | 0 | 0 | 1 |
| … | … | … | … | … | … |

5. Generate input files for 3′aQTL mapping.

```
> bash ./src/prepare_inputs_for_3aQTL_mapping.sh -c known_covariates.txt
```

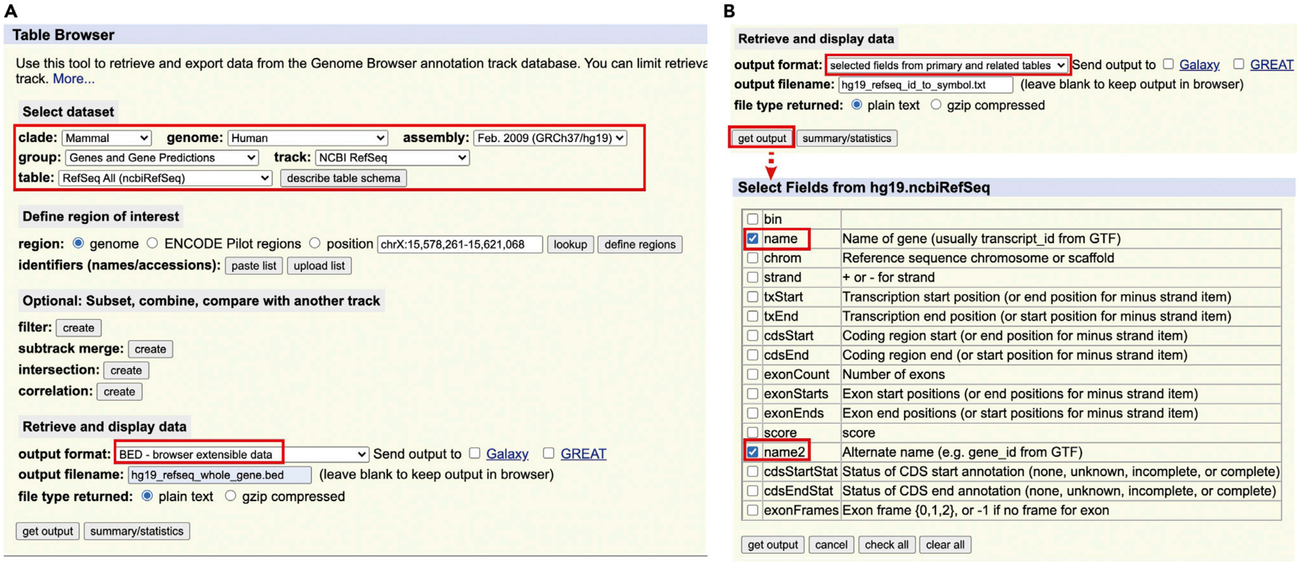


**Table Browser**

Use this tool to retrieve and export data from the Genome Browser annotation track database. You can limit retrieva
track. More...

**Select dataset**

**clade:** Mammal **genome:** Human **assembly:** Feb. 2009 (GRCh37/hg19)
**group:** Genes and Gene Predictions **track:** NCBI RefSeq
**table:** RefSeq All (ncbiRefSeq) describe table schema

**Define region of interest**

**region:** ● genome ○ ENCODE Pilot regions ○ position chrX:15,578,261-15,621,068 lookup define regions
**identifiers (names/accessions):** paste list upload list

**Optional: Subset, combine, compare with another track**

**filter:** create
**subtrack merge:** create
**intersection:** create
**correlation:** create

**Retrieve and display data**

**output format:** BED - browser extensible data Send output to □ Galaxy □ GREAT
**output filename:** hg19_refseq_whole_gene.bed (leave blank to keep output in browser)
**file type returned:** ● plain text ○ gzip compressed

get output summary/statistics

**Retrieve and display data**

**output format:** selected fields from primary and related tables Send output to □ Galaxy □ GREAT
**output filename:** hg19_refseq_id_to_symbol.txt (leave blank to keep output in browser)
**file type returned:** ● plain text ○ gzip compressed

get output summary/statistics

**Select Fields from hg19.ncbiRefSeq**

| | |
|---|---|
| □ bin | |
| ☑ name | Name of gene (usually transcript_id from GTF) |
| □ chrom | Reference sequence chromosome or scaffold |
| □ strand | + or - for strand |
| □ txStart | Transcription start position (or end position for minus strand item) |
| □ txEnd | Transcription end position (or start position for minus strand item) |
| □ cdsStart | Coding region start (or end position for minus strand item) |
| □ cdsEnd | Coding region end (or start position for minus strand item) |
| □ exonCount | Number of exons |
| □ exonStarts | Exon start positions (or end positions for minus strand item) |
| □ exonEnds | Exon end positions (or start positions for minus strand item) |
| □ score | score |
| ☑ name2 | Alternate name (e.g. gene_id from GTF) |
| □ cdsStartStat | Status of CDS start annotation (none, unknown, incomplete, or complete) |
| □ cdsEndStat | Status of CDS end annotation (none, unknown, incomplete, or complete) |
| □ exonFrames | Exon frame {0,1,2}, or -1 if no frame for exon |

get output cancel check all clear all

**Figure 1. Extract gene annotation and gene symbol of RefSeq from UCSC Table Browser**

(A) Extract gene annotation of the human genome (hg19) from the NCBI RefSeq database and output them into a BED format. Key options are highlighted in red boxes.

(B) Extract the mapping between gene symbol and RefSeq ID. Select "selected fields from primary and related tables" instead of BED for the option "output format" and choose "name" and "name2" in the pop-up table after clicking on "get output".

*Note:* Option "-c" takes a tab-delimited covariate file (default is "NA" if not available). An example can be found at https://github.com/3UTR/3aQTL-pipe. Other arguments use default values, which can be checked with the option "-h".

⚠ CRITICAL: This step creates intermediate files required for running Matrix eQTL to identify 3′aQTL. Input files include an APA quantile matrix, a genotype matrix, and a known covariate matrix. Sample ids in these three matrices need to be cross-referenced; therefore, please make sure the sample id in the three input matrices is consistent.

## Identify common genetic variants that associated with APA usage

⏲ Timing: 4 h for CEU population of GEUVADIS dataset (the run time depends on the number of genes processed and the number of SNPs, affected by window size)

6. Perform the 3′aQTL mapping.

```
> Rscript ./src/run_3aQTL_mapping.R
```

*Note:* The *cis* 3′aQTLs "Cis_3aQTL_all_control_gene_exprs.txt" and *trans* 3′aQTLs "Trans_3aQTL_all_control_gene_exprs.txt" will be output in the directory "Matrix_eQTL/".

7. Visualize significant associations.

```
> Rscript ./src/QTL_plot.R -s ''chr7_128640188_A_G'' -g ''NM_001347928.2|IRF5|chr7|+''
```

*Note:* The first parameter specifies the SNP for visualization. The second one specifies the related transcript. This command will generate a publication-ready plot named "chr7_128640188_A_G.IRF5.pdf".

### Identify potentially causal SNP

**⏱ Timing: 4 h for CEU population of GEUVADIS dataset, the run time depends on the significant genes detected by Matrix eQTL in step 6 and the number of threads**

Fine mapping can narrow the genetic association signals to a few potential causal variants. We use our Susie (SUm of Single Effects) algorithm (Wang et al., 2020) for fine-mapping analysis. susieR implements the Susie algorithm and requires individual-level genotype and phenotype data. The following codes generate the input data necessary for fine-mapping analysis.

8. Prepare input files for fine-mapping.

```
> bash ./src/prepare_inputs_for_finemapping.sh
```

*Note:* This shell script requires "3UTR_location.txt", genetic associations result (e.g. "Cis_3aQTL_all_control_gene_exprs.txt") that generated in step 5 and step 6, respectively. And genotype data ("Genotype_matrix.txt") are generated in step 5. It generates a directory for each significant transcript with two files named "3aQTL.vcf" and "expr.phen".

9. Fine-mapping analysis.
   a. Perform fine-mapping analysis on all significant transcripts.

```
> bash ./src/run_fine_mapping.sh -t 8
```

*Note:* The option "-t" specifies the number of threads used for parallel analysis; the default value is 1.

*Note:* SusieR will generate three files with names suffixed by ".pdf", ".rds", and ".txt" in each transcript directory. One plot describes the independent signals (Figure 2), an R binary file contains the results of susieR fine mapping, and a text file lists all independent signals.

   b. Summarize the susieR output.

```
> Rscript ./src/merge_finemap_results.R
```

*Note:* This command will summarize the fine-mapping results of each significant transcript and generate two files with the names "susieR_res.all_genes.txt" and "susieR_res.stat.txt", respectively.

## EXPECTED OUTCOMES

This protocol describes a pipeline named 3aQTL-pipe which identifies the associations between common genetic variants and APA changes from population-scale RNA-seq and genotype data. We took 89 RNA-seq samples from the Geuvadis RNA sequencing project and matched genotype data as an example to present the usage of 3aQTL-pipe step-by-step. Before testing the

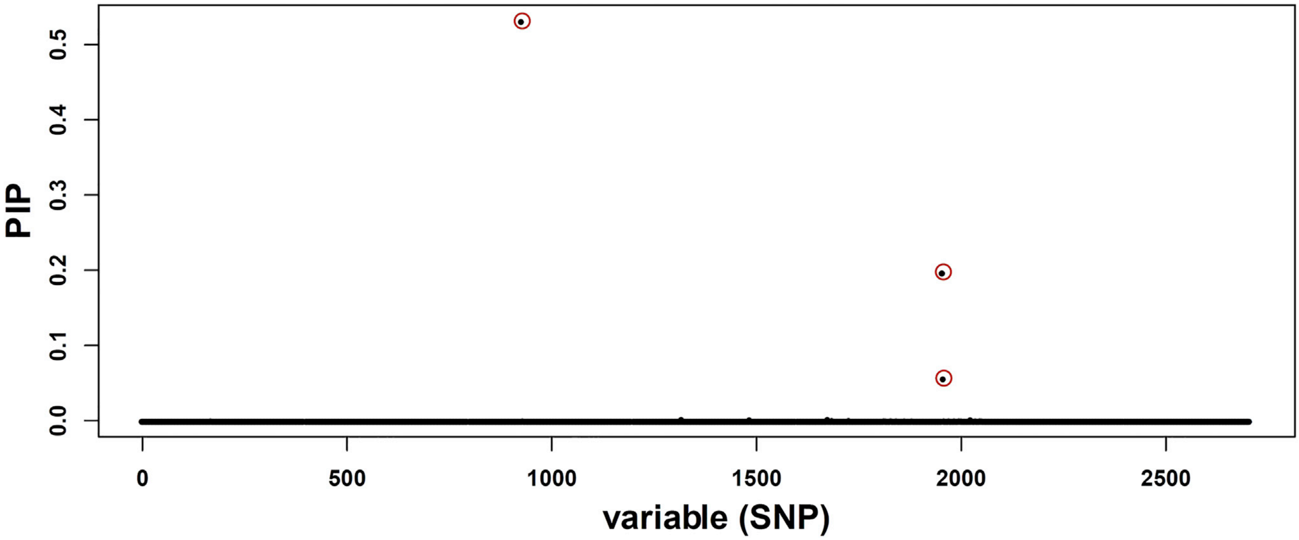


**Figure 2. Three independent signals identified by susieR in gene *ACTR1B***

PIP: Posterior inclusion probability.

association between genetic variants and APA, the APA quantitative profile across all samples in a population was estimated by Dapars2. As described in steps 6 and 7 of this protocol, a matrix contains each transcript's PDUI (Percentage of Distal polyA site Usage Index) across samples. After required quality filtering and formatting, the association test is performed as described in step 6. This step will identify significant associations (FDR<0.05) between common genetic variants and APA usage. Significant and non-significant associations will be summarized, respectively. Figure 3 presents a significant 3′aQTL that regulates the alternative 3′UTR length of *IRF5*. The final

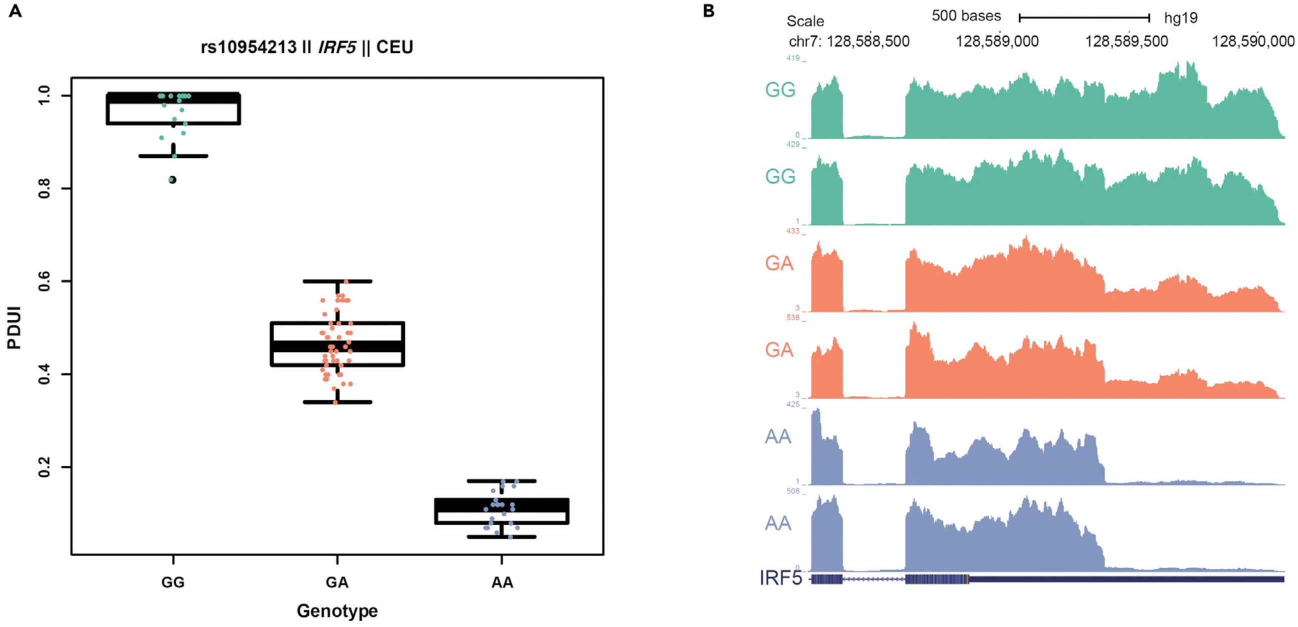


**Figure 3. An example of 3′aQTL describes the regulation of 3′UTR in *IRF5* by the genotype of rs10954213 in European individuals**

(A) Boxplot shows different PDUI values in individuals with three genotypes.

(B) Reads coverage at 3′UTR of *IRF5* in samples with different genotypes, consistent with the PDUI values estimated by Dapars2.

step is fine-mapping of *cis*-3′aQTL, which tries to find potential causal SNPs for each 3′aGene (the transcripts with at least one significant 3′aQTL). susieR may detect one or multiple causal SNPs for a 3′aGene, and the final results are summarized. Figure 2 in step 9 listed an example of three potential causal SNPs identified for *ACTR1B*. 3aQTL-pipe is well developed and can be easily generalized to other datasets.

## QUANTIFICATION AND STATISTICAL ANALYSIS

Dapars2 estimates APA usage depending on the mRNA 3′UTR read coverage. The APA events detected with insufficient reads are more likely due to random sampling errors and limited statistical power. Therefore, APA events with read coverage lower than the threshold defined by users in the configuration file will be excluded. The current protocol defines the threshold of read coverage as 15.

Matrix eQTL provides three regression models ("modelLINEAR", "modelANOVA", and "modelLINEAR_CROS"). In this protocol, we use the "modelLINEAR" model, which only considers the additive effect of genotype on the APA level (as shown in the equation below).

$$expression\ =\ \alpha + \sum_k \beta_k \cdot Covariate_k + \gamma \cdot genotype\_additive$$

After identifying all significant associations, Matrix eQTL will calculate an FDR for each association. In the current protocol, we defined a significant association with an FDR of less than 0.05.

## LIMITATIONS

Our protocol was tested on Ubuntu (tested on Ubuntu 18.04.6) and macOS (tested on Big Sur) systems. For windows users, third-party software, such as a virtual machine or Cygwin, may be needed. Another limitation is that the APA analysis by Dapars2 is restricted to the 3′UTR region, and the APA events that occur within intronic regions are excluded.

## TROUBLESHOOTING

### Problem 1

If the users cannot install PEER and Dapars2 correctly.

### Potential solution

We have provided a docker image for the whole pipeline. See the usage at the GitHub repository: https://github.com/3UTR/3aQTL-pipe/wiki.

### Problem 2

If users run this pipeline through our prebuilt docker image, it will default initiate a container as root (uid:0). This may result in all generated files with the root ownership.

### Potential solution

Users can avoid this problem by specifying the same user id as the localhost (use command "id") through the option "–user".

### Problem 3

The filesystem of the docker container is not reachable by the host. This may result in failure to share files.

### Potential solution

Users can avoid this by mounting the host directory into the container. The command below presents one example.

```
> local_workdir=/PATH/TO/LOCAL_DIR

> container_workdir=/PATH/TO/CONTAINER_DIR

> docker run -it –name=‘‘3aqtl_container’’ \

–mount type=bind,source=$local_workdir,target=‘‘$container_workdir’’
```

### Problem 4

Users may encounter errors like ‘‘[E::hts_open_format_impl] Failed to open file’’ when executing the script ‘‘prepare_inputs_for_apa_quant.sh’’, This may be due to the incorrect bam file path.

### Potential solution

Users can avoid this by creating a ‘‘sample_list.txt’’ file listing all samples and bam files with the correct path.

### Problem 5

Step 2 (prepare_inputs_for_apa_quant.sh) may take a long time to finish if large samples are analyzed parallelly. This may be due to small threads for converting bam to bedgraph format.

### Potential solution

Users can avoid this by setting the option ‘‘-t’’ with a larger threads number.

## RESOURCE AVAILABILITY

### Lead contact

Further information and requests for resources and reagents should be directed to and will be fulfilled by the lead contact, Lei Li (lei.li@szbl.ac.cn).

### Materials availability

This study did not generate new unique reagents.

### Data and code availability

The RNA-seq data of the Geuvadis RNA-seq Project and corresponding genotype data are publicly available at EBI ArrayExpress (https://www.ebi.ac.uk/arrayexpress/experiments/E-GEUV-1/). All codes mentioned in this protocol are available at GitHub: https://github.com/3UTR/3aQTL-pipe, or docker hub: https://hub.docker.com/r/3utr/3aqtl_pipe/tags.

## ACKNOWLEDGMENTS

We thank members of the Li laboratory for helpful discussions. This work was supported by the National Natural Science Foundation of China (no. 32100533) to L.L., NIH National Institute on Aging grant (no. U01AG072572) and funding from the Thompson Family Foundation (TAME-AD) to G.W., and we also thank Shenzhen Bay Laboratory supercomputing center for high-computing support.

## AUTHOR CONTRIBUTIONS

X.Z. designed the protocol and wrote the manuscript. R.D. built the docker image for the pipeline. W.C., G.W., and S.C. edited the protocol, and Q.W. provided high-computing support. W.L. supervised and edited the manuscript. L.L. supervised, wrote, and revised the manuscript.

## DECLARATION OF INTERESTS

The authors declare no competing interests.

## REFERENCES

1000 Genomes Project Consortium (2015). A global reference for human genetic variation. Nature *526* (7571), 68–74. https://doi.org/10.1038/nature15393.

Danecek, P., Auton, A., Abecasis, G., Albers, C.A., Banks, E., DePristo, M.A., Handsaker, R.E., Lunter, G., Marth, G.T., Sherry, S.T., et al. (2011). The variant call format and VCFtools. Bioinformatics *27*, 2156–2158. https://doi.org/10.1093/bioinformatics/btr330.

Danecek, P., Bonfield, J.K., Liddle, J., Marshall, J., Ohan, V., Pollard, M.O., Whitwham, A., Keane, T., McCarthy, S.A., Davies, R.M., and Li, H. (2021). Twelve years of SAMtools and BCFtools. GigaScience *10*, giab008. https://doi.org/10.1093/gigascience/giab008.

Feng, X., Li, L., Wagner, E.J., and Li, W. (2018). TC3A: the cancer 3′ UTR atlas. Nucleic Acids Res. *46*, D1027–D1030. https://doi.org/10.1093/nar/gkx892.

Lappalainen, T., Sammeth, M., Friedländer, M.R., 't Hoen, P.A.C., Monlong, J., Rivas, M.A., Gonzàlez-Porta, M., Kurbatova, N., Griebel, T., Ferreira, P.G., et al. (2013). Transcriptome and genome sequencing uncovers functional variation in humans. Nature *501*, 506–511. https://doi.org/10.1038/nature12531.

Li, L., Huang, K.L., Gao, Y., Cui, Y., Wang, G., Elrod, N.D., Li, Y., Chen, Y.E., Ji, P., Peng, F., et al. (2021). An atlas of alternative polyadenylation quantitative trait loci contributing to complex trait and disease heritability. Nat. Genet. *53*, 994–1005. https://doi.org/10.1038/s41588-021-00864-5.

Purcell, S., Neale, B., Todd-Brown, K., Thomas, L., Ferreira, M.A., Bender, D., Maller, J., Sklar, P., de Bakker, P.I., Daly, M.J., and Sham, P.C. (2007). PLINK: a tool set for whole-genome association and population-based linkage analyses. Am. J. Hum. Genet. *81*, 559–575. https://doi.org/10.1086/519795.

Quinlan, A.R., and Hall, I.M. (2010). BEDTools: a flexible suite of utilities for comparing genomic features. Bioinformatics *26* (6), 841–842. https://doi.org/10.1093/bioinformatics/btq033.

Shabalin, A.A. (2012). Matrix eQTL: ultra fast eQTL analysis via large matrix operations. Bioinformatics *28*, 1353–1358. https://doi.org/10.1093/bioinformatics/bts163.

Stegle, O., Parts, L., Piipari, M., Winn, J., and Durbin, R. (2012). Using probabilistic estimation of expression residuals (PEER) to obtain increased power and interpretability of gene expression analyses. Nat. Protoc. *7*, 500–507. https://doi.org/10.1038/nprot.2011.457.

Wang, G., Sarkar, A., Carbonetto, P., and Stephens, M. (2020). A simple new approach to variable selection in regression, with application to genetic fine mapping. J. Roy. Stat. Soc. B *82*, 1273–1300. https://doi.org/10.1111/rssb.12388.