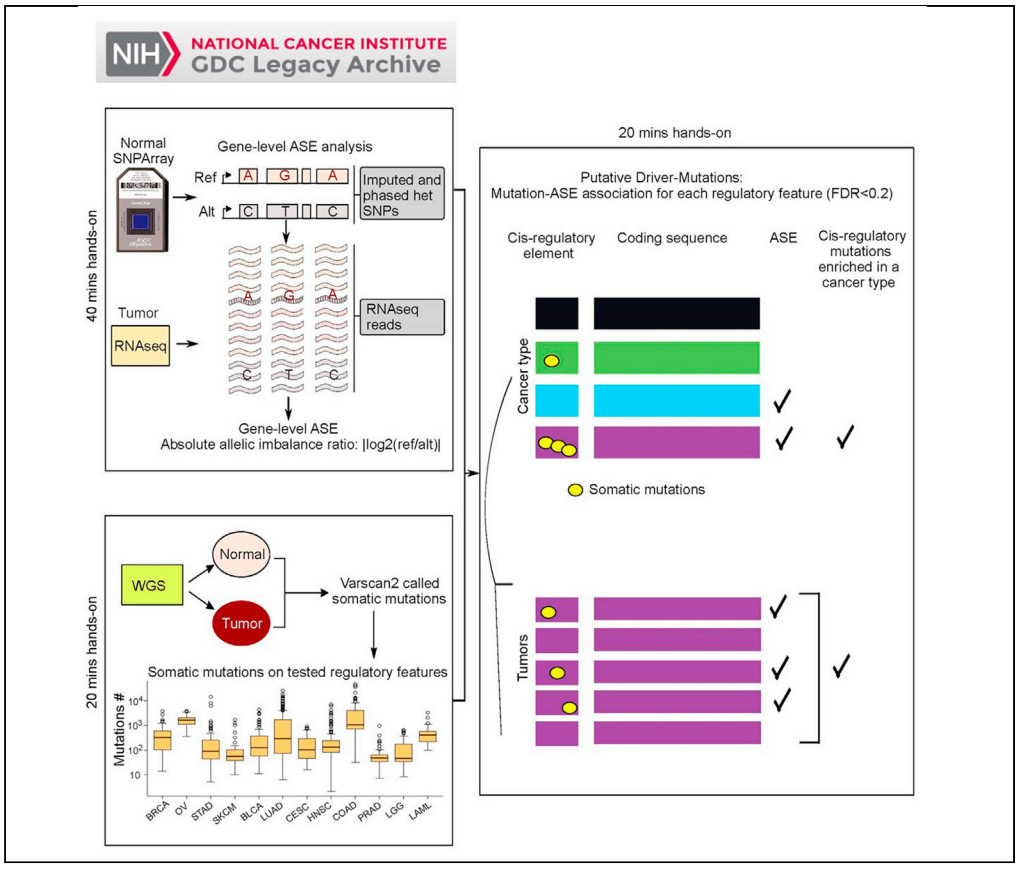
STAR Protocols

CellPress
OPEN ACCESS

Protocol

Identifying tumorigenic non-coding mutations through altered *cis*-regulation



Identification of non-coding mutations driving tumorigenesis requires alternative approaches to coding mutations. Enriched associations between mutated regulatory elements and altered cis-regulation in tumors are a promising approach to stratify candidate non-coding driver mutations. Here we provide a bioinformatics pipeline to mine data from the Cancer Genomic Commons (GDC) for such associations. The pipeline integrates RNA and whole-genome sequencing with genotyping data to reveal putative non-coding driver mutations by cancer type.

Zhongshan Cheng, Michael Vermeulen, Micheal Rollins-Green, Tomas Babak, Brian DeVeale

cheng.zhong.shan@gmail.com (Z.C.)
brian.deveale@ucsf.edu (B.D.)

Highlights

Integration of expression, genotyping, and whole-genome sequencing modalities

RNA-seq profiles resolved to gene-level allele-specific expression (ASE).

Mutated cis-regulatory features associated with gene-level ASE by cancer type

Cheng et al., STAR Protocols *2*, 100934
December 17, 2021
https://doi.org/10.1016/j.xpro.2021.100934

STAR Protocols

CellPress
OPEN ACCESS

Protocol

Identifying tumorigenic non-coding mutations through altered *cis*-regulation

Zhongshan Cheng,[1,3,6,*] Michael Vermeulen,[1,4] Micheal Rollins-Green,[1] Tomas Babak,[1,5] and Brian DeVeale[2,7,*]

[1]Department of Biology, Queen's University, Kingston, ON K7L 3N6, Canada

[2]The Eli and Edythe Broad Center of Regeneration Medicine and Stem Cell Research, University of California, San Francisco, San Francisco, CA 94143, USA

[3]Present address: Applied Bioinformatics Center, St. Jude Children's Research Hospital, Memphis, TN 38105, USA

[4]Present address: Department of Medical Genetics, University of British Columbia, Vancouver, BC V6T 1Z4, Canada

[5]Present address: Leapfrog Bio, 319 N Bernardo Avenue, Mountain View, CA, 94043743, USA

[6]Technical contact

[7]Lead contact

*Correspondence: cheng.zhong.shan@gmail.com (Z.C.), brian.deveale@ucsf.edu (B.D.)
https://doi.org/10.1016/j.xpro.2021.100934

SUMMARY

Identification of non-coding mutations driving tumorigenesis requires alternative approaches to coding mutations. Enriched associations between mutated regulatory elements and altered cis-regulation in tumors are a promising approach to stratify candidate non-coding driver mutations. Here we provide a bioinformatics pipeline to mine data from the Cancer Genomic Commons (GDC) for such associations. The pipeline integrates RNA and whole-genome sequencing with genotyping data to reveal putative non-coding driver mutations by cancer type.

For complete information on the generation and use of this protocol, please refer to Cheng et al. (2021).

BEFORE YOU BEGIN

Obtain a GDC Key and install the Driver-ASE Docker image. Alternatively, install Driver-ASE using conda in a local Linux system or online computational cluster.

Obtain GDC key

Timing: [~10 min]

1. Apply for GDC access and register a GDC account to obtain a downloading key (https://portal.gdc.cancer.gov/). The key requires periodic renewal.

   Note: A GDC Key is required to download SNP array data, RNA-Seq BAMs and whole-genome sequencing (WGS) BAMs from the GDC database (Gao et al., 2019).

   a. Login to GDC.
   b. Click on your account name (top right), and click the download token from the drop-down menu.
   c. Rename the downloaded token file to 'gdc.key' on the local computer.

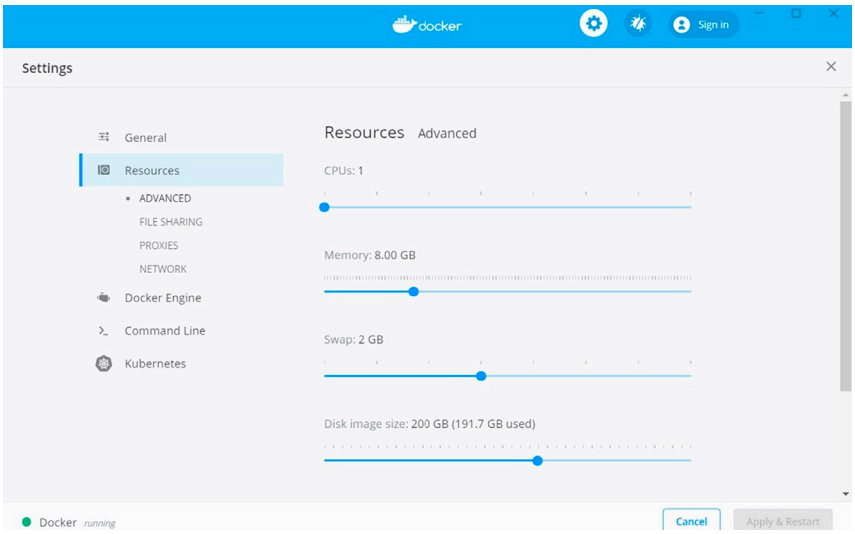


Figure 1. Recommended resource settings for the Driver-ASE Docker image

Install Driver-ASE via docker

Timing: [~30 min]

2. Install Docker (https://www.docker.com/products/docker-desktop) using the default settings.
   a. After Docker is installed, open Terminal and pull the Driver-ASE Docker image with the command:

```
> docker pull mikevermeulen/driver-ase
```

   b. Place the token file 'gdc.key' in the 'Database' directory, '/Driver-ASE/Database/gdc.key'. The Driver-ASE scripts to download GDC data require that the path to the GDC key is specified.
   c. Adjust the amount of CPU and RAM that the Driver-ASE Docker container uses by going to Docker settings and selecting 'Resources' and modify the settings (Figure 1).

Note: To calculate gene-level ASE or call somatic mutations using the Driver-ASE workflow, >8 GB of memory and 100 GB of free disk space are recommended. The optimal settings depend on the available RAM and memory. GDC sample information can be queried with ~1 GB of memory.

   d. Install the pipeline within the Docker image using the git command:

```
> git clone <hyperlink refid="https://github.com/michaelcvermeulen/driver-ase.-
git">https://github.com/michaelcvermeulen/driver-ase.git
```

3. Download and mount the 1000 Genome Reference database in Docker for genotype imputation.

Note: This database is required by the Driver-ASE pipeline, ~16 GB and shared through MEGA as 'Database.zip'.

a. Install MEGASync (https://mega.nz/sync) and then use it to download 'Database.zip' from https://mega.nz/file/VsRTzYRJ#Bi-a1lgVkvzX-MGkNRhINNHu9nGY9owew4cTB4aAWys
b. Unzip and place the 'Database.zip' files in the 'Driver-ASE' directory.
c. Mount the directory 'Driver-ASE' to the Docker image and change the absolute path to the full path to the 'Driver-ASE' directory by running the following Docker command:

```
#run Docker image in background

#replace path of Driver-ASE in host computer with its

#absolute path in local computer

> docker run -it -d –rm –name driver-ase –memory=8g \

> -v AbsolutePathInLocalComputer/Driver-ASE:/Driver-ASE \

> mikevermeulen/driver-ase:latest /bin/bash

#Initiate bash within Docker

> docker exec -it driver-ase /bin/bash
```

4. Make the perl scripts executable using the following commands:

```
> cd /Driver-ASE/Driver_ASE_Lib/Driver_ASE_Scripts

> chmod a+x *
```

Note: Third party software required by the pipeline including samtools (Li et al., 2009), overlapSelect (Kent et al., 2002), shapeit (Delaneau et al., 2011), impute2 (Marchini et al., 2007), and VarScan2 (Reble et al., 2017), are preinstalled and located in the directory '/usr/local/bin/anaconda2/bin' of the Docker image.

5. Save these changes by running the command below in the local computer terminal before exiting the Driver-ASE Docker image. This 'commit' prevents loss of changes when Docker is closed, and renders the pipeline executable from the Docker image.

```
> docker commit driver-ase driver-ase
```

Install driver-ASE using conda

Timing: [~60min]

6. Install Driver-ASE, its dependencies and databases in Linux terminal with bash. Installation is automated via conda on local or online Linux clusters.
   a. Download the shell script from github: https://raw.githubusercontent.com/michaelcvermeulen/driver-ase/main/Driver_ASE_Installation_with_conda.sh
   b. Type 'bash ./Driver_ASE_Installation_with_conda.sh' in Linux terminal from within the directory containing the shell script.

Note: The shell script will determine whether conda, the conda environment "driverase," its dependencies and databases are available. If conda is not preinstalled, the script will

download conda and install it before installing Driver-ASE. The script will stop installation if the conda driverase environment has already been created. To force a new installation to over-write a previous one, enter "1". The following message will be produced:

```
> bash ./Driver_ASE_Installation_with_conda.sh

Usage: Driver_ASE_Installation_with_conda.sh

(1) Dir4InstallationFullpath (enter ''.'' to install in current directory)

(2) Force (force installation: default is ''0'', enter ''1'' to force installation)

Note: wget needs to be installed for running this shell script to download data!
```

c. Run the shell script with the Driver-ASE Docker image or a Linux environment with the commands below.

Note: Minconda2 is preinstalled in the Driver-ASE Docker image, so the user only needs to create the conda environment "driverase", then install Driver-ASE and its dependencies.

```
#create driverase environment if necessary

#conda create -n driverase

> conda activate driverase

#set the last parameter to ''1'' to force installation of #Driver-ASE and its dependencies

> ./Driver_ASE_Installation_with_conda.sh . 1
```

d. Activate the conda environment "driverase" before running any Driver-ASE scripts as follows (See 'potential problem 1' if the environment is not activated):

```
> conda activate driverase
```

e. Save changes made to the Driver-ASE docker image during installation or modification by running the following command:

```
> docker commit driver-ase driver-ase
```

Driver-ASE workflow

Timing: [~30 min]

The Driver-ASE workflow involves downloading GDC data, then formatting, processing and ultimately integrating it for the association between gene-level ASE and somatic mutations. Figure 2 summarizes the relationship between the various Driver-ASE functions in a flowchart.

The pipeline uses different perl scripts to download SNP array, RNA-Seq and WGS BAMs, and then calculate gene-level ASE and call somatic mutations; these results are then converted to associate somatic mutations in different regulatory elements and gene-level ASE in MATLAB (at least version 2014b or using version 90 of the freely available MATLAB runtime (https://www.mathworks.com/products/compiler/matlab-runtime.html).

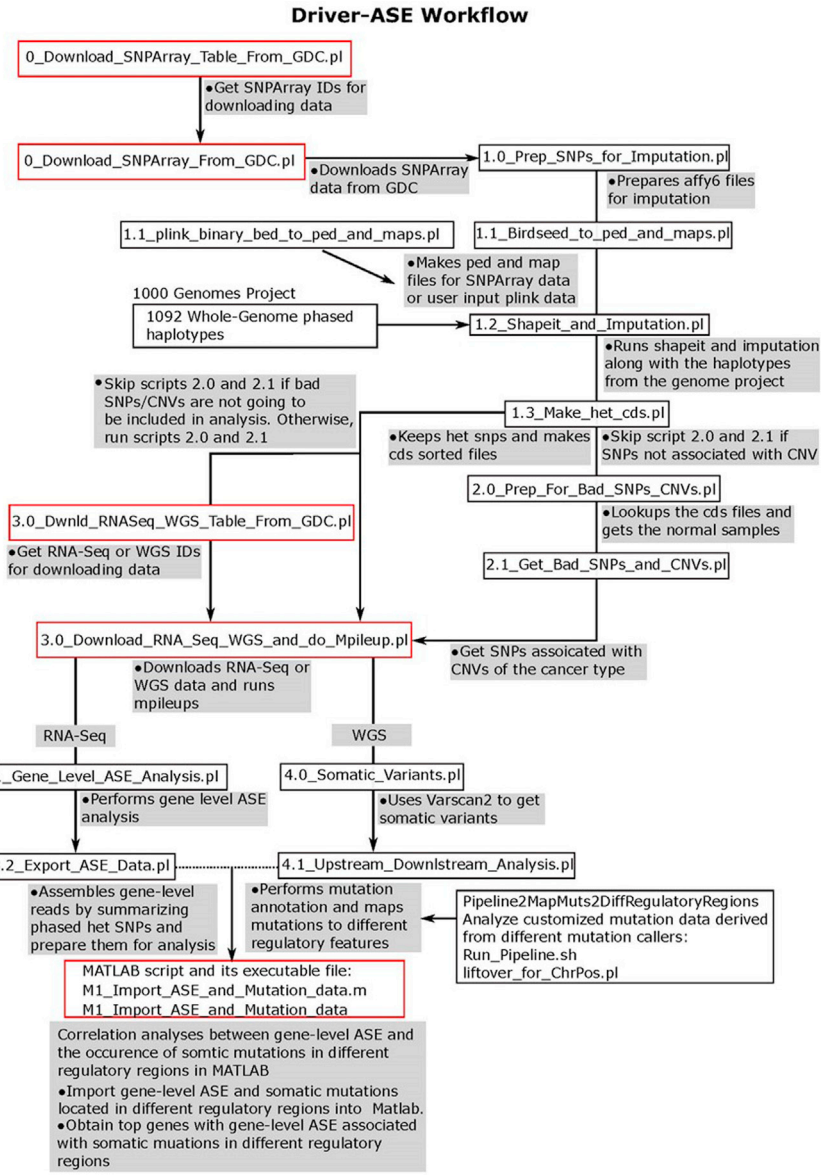


Figure 2. Driver-ASE workflow

The flowchart illustrates the scripts to identify putative non-coding driver mutations. The key Driver-ASE steps of calculating gene-level ASE, calling somatic mutations, and associating mutations with gene-level ASE by regulatory element are outlined in red.

7. View the parameters of each perl script (Figures 3 and 4):
   a. Navigate to the directory containing the perl scripts in either the Docker or Linux terminal according to where Driver-ASE is installed and list the scripts:

```
> cd /Driver-ASE/Driver_ASE_Lib/Driver_ASE_Scripts

# list perl scripts

> ls
```

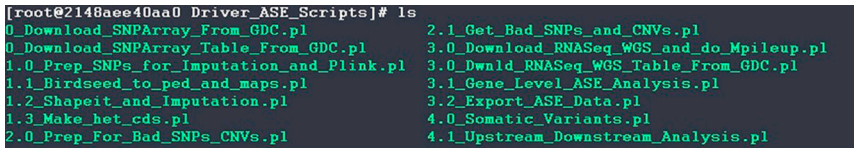


Figure 3. The perl scripts in the Driver-ASE pipeline

b. Enter the name of a script to access its usage and parameter details as shown in the example below.

Note: The script is correctly installed if no errors appear after displaying its details.

```
#If the perl script is not executable, use 'chmod a+x
#script' to make it executable.
#If running the script in a local Linux computer or
#online computational cluster, need to activate conda
#environment
#conda activate driverase
> ./0_Download_SNPArray_Table_From_GDC.pl
```

To save user time, the sample information for >30 cancer types from GDC legacy portal is included in "/Driver_ASE/Analysis/". Please see 'potential problem 2' for detail on how to over-write existing tables.

c. View the target directory of Database to confirm that the 1000 Genome Reference database required by Driver-ASE is available.

```
> cd /Driver-ASE/Database
> ls
```

d. The default output for all Driver-ASE scripts is within the following directory:

```
> cd /Driver-ASE/Analysis
> ls
```

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Software and algorithms		
Driver-ASE	This paper	https://github.com/mikevermeulen/driver-ase
Miniconda	Anaconda, Inc.	https://docs.conda.io/en/latest/miniconda.html
impute2	Marchini and Howie, 2010	https://mathgen.stats.ox.ac.uk/impute/impute_v2.html#download
Shapeit	Delaneau et al., 2013	https://mathgen.stats.ox.ac.uk/genetics_software/shapeit/shapeit.html

(*Continued on next page*)

Continued

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Samtools	Li et al. (2009)	http://www.htslib.org/download/
plink1.9	Chang et al. (2015)	https://www.cog-genomics.org/plink2
overlapSelect	UCSC Genome Browser	http://hgdownload.soe.ucsc.edu/admin/exe/linux.x86_64/overlapSelect
VarScan2	Koboldt et al. (2012)	http://varscan.sourceforge.net/
Deposited data		
Processed Driver-ASE data	Cheng et al. (2021)	https://data.mendeley.com/datasets/4kx5sfx9vz/2
shapeit2 phased 1000 Genome Project reference files	Delaneau et al., 2013	https://mathgen.stats.ox.ac.uk/impute/ALL.integrated_phase1_SHAPEIT_16-06-14.nomono.tgz
1000 Genome Project reference haplotype files	Marchini and Howie, 2010	https://mathgen.stats.ox.ac.uk/impute/1000GP_Phase3.tgz https://mathgen.stats.ox.ac.uk/impute/1000GP_Phase3_chrX.tgz

STEP-BY-STEP METHOD DETAILS

Evaluate gene-level ASE and identify somatic mutations

Timing: [1-h hands-on; computational time scales with sample number and resources]

Outline of workflow for the major Driver-ASE scripts:

1. Use **0_Download_SNPArray_Table_From_GDC.pl** to download the annotation table without the genotyping SNP array data.
   a. Download the SNP array table using the commands below, specify the cancer types to download with '-c' (e.g., BRCA, OV, PRAD), and the data type with '-a' (either "Genotypes" or "Copy number estimate").

Use the shell script '0_Download_ALL_SNPArray_Table_From_GDC.sh', to download SNPArray sample information for all GDC legacy cancer types. Download by wrapping the following commands. See 'potential problem 3' to resolve download issues.

```
Cancer type and/or array type  was not entered!
#################################################################################################
usage: program [--cancer|-c cancer_type (e.g. PRAD)] [--Expstrategy|-E Experimental strategy (Genotyping array)] [--array
type|-a array data type (e.g. Genotypes or "Copy number estimate")] [--help|-h]
Any names with spaces must be wraped in DOUBLE QUOTES or have back slashes to escape spaces.
Experimental Strategies used:
        Copy number estimate
        Genotypes
All available TCGA cancer types:
Breast invasive carcinoma [BRCA]                Glioblastoma multiforme [GBM]
Ovarian serous cystadenocarcinoma [OV]          Lung adenocarcinoma [LUAD]
Uterine Corpus Endometrial Carcinoma [UCEC]     Kidney renal clear cell carcinoma [KIRC]
Head and Neck squamous cell carcinoma [HNSC]    Brain Lower Grade Glioma [LGG]
Thyroid carcinoma [THCA]                        Lung squamous cell carcinoma [LUSC]
Prostate adenocarcinoma [PRAD]                  Stomach adenocarcinoma [STAD]
Skin Cutaneous Melanoma [SKCM]                  Colon adenocarcinoma [COAD]
Bladder Urothelial Carcinoma [BLCA]             Liver hepatocellular carcinoma [LIHC]
Kidney renal papillary cell carcinoma [KIRP]    Sarcoma [SARC]
Esophageal Carcinoma [ESCA]                     Pancreatic Adenocarcinoma [PAAD]
Pheochromocytoma and Paraganglioma [PCPG]       Rectum Adenocarcinoma [READ]
Testicular Germ Cell Tumors [TGCT]              Acute Myeloid Leukemia [LAML]
Thymoma [THYM]                                  Kidney Chromophobe [KICH]
Adernocortical Carcinoma [ACC]                  Mesothelioma [MESC]
Uterine Carcinosarcoma [UCS]                    Cholangiocarcinoma [CHOL]
Cervical squamous cell carcinoma and            Lymphoid Neoplasm Diffuse
endocervical adenocarcinoma [CESC]              Large B-cell Lymphoma [DLBC]
Uveal Melanoma [UVM]
#################################################################################################
```

Figure 4. A typical help message generated after execution *of the perl script* '0_Download_SNPArray_Table_From_GDC.pl'

```
> cd /Driver-ASE/Driver_ASE_Lib/Driver_ASE_Scripts
#download sample information of BRCA SNPArrays from GDC
#run it in docker image or conda drivease environment
> ./0_Download_SNPArray_Table_From_GDC.pl \
> -c BRCA -a Genotypes
# Displays GDC sample IDs as well as corresponding GDC
# uuids for downloaded SNPArrays
>
head ../../Analysis/BRCA/BRCA_tables/BRCA.Genotypes.id2uui
d.txt
#download sample information of BRCA copy number estimate
from GDC
> ./0_Download_SNPArray_Table_From_GDC.pl \
> -c BRCA -a "Copy number estimate"
# Displays GDC sample IDs and corresponding GDC uuids for
# downloaded Copy Number Estimates
> head ../../Analysis/BRCA/BRCA_tables/BRCA.Copy\ number\
estimate.id2uuid.txt
```

2. Download SNP-Array Genotyping or "Copy number estimate" using **0_Download_SNPArray_From_GDC.pl**
   a. Run the script by specifying the cancer type '-c', the data type '-a' (either "Genotypes" or "Copy number estimate"), the full path to the GDC key '-k' and the use of downloading software curl or aria2c using '-d'.

The script downloads and generates GDC tables if it cannot locate preexisting tables of targeted samples. If tables already exist, the script will begin downloading the files.

Note: The path to the GDC downloading key must also be specified with '-k' if it is moved from the default path of ('/Driver-ASE/Database/gdc.key'). See 'potential problem 4' to resolve GDC authentication errors.

```
#download raw SNPArray data from GDC
> ./0_Download_SNPArray_From_GDC.pl \
> -c BRCA -a Genotypes \
> -k /Driver-ASE/Database/gdc.key -d curl
#download raw SNPArray copy number estimates from GDC
> ./0_Download_SNPArray_From_GDC.pl \
> -c BRCA -a "Copy number estimate" \
> -k /Driver-ASE/Database/gdc.key -d curl
```

3. Prepares the SNP-Array genotype annotation data for imputation using **1.0_Prep_SNPs_for_Imputation_and_Plink.pl**
   a. Unzip 'GenomeWideSNP_6.na35.annot.csv.zip' from the Database directory into a new directory called 'affy6'.
   b. Run the script, specifying the cancer type with the parameter '-c' as illustrated below:

```
> ./1.0_Prep_SNPs_for_Imputation_and_Plink.pl -c BRCA
```

Note: The script first parses annotation information from the file 'GenomeWideSNP_6.na35.annot.csv' and outputs data into a file called 'snp6.anno.txt'. Next, the script outputs annotated SNPs to a file called 'snp6.cd.txt'. If these files already exist, the script will not execute and it will bypass this step.

4. Parse the genotype files and create the plink1.9 formatted (.map and .ped) files for input into plink1.9 (Chang et al., 2015) using **1.1_Birdseed_to_ped_and_maps.pl**
   a. Run the script, specifying the sample type with '-s' (0 for normal, 1 for tumor and 2 for both), the cancer type '-c' and/or enter the path to the plink1.9 '-p'.

```
> ./1.1_Birdseed_to_ped_and_maps.pl \
> -p /usr/local/bin/anaconda2/bin/plink \
> -s 0 -c BRCA
```

Note: plink1.9 performs quality control by keeping SNPs with minor allele frequency >0.01 on these data, which is required by shapeit. If users prefer genotypes called from WGS data, supply these genotypes in plink binary bed format, using the sister Perl script '1.1_plink_binary_bed_to_ped_and_maps.pl' to prepare the input data. This sister script uses the same parameters, except for the '-b' parameter for supplying plink binary bed file.

5. Phase genotyping data using **1.2_Shapeit_and_Imputation.pl**
   a. Phase genotypes as shown below, specifying the cancer type after the parameter '-c'.

```
> ./1.2_Shapeit_and_Imputation.pl -c BRCA
```

Note: Shapeit uses the genotyping data in the 'peds' and 'maps' directories and outputs the phased result to a new directory called 'phased' (Delaneau et al., 2011).

Shapeit only extracts chromosome sizes, generating a file called 'chr_lens_grep_chr' from which it parses only chr1-22 and X for imputation. The imputation results, including SNP calls and phased haplotype data are printed to the directory 'phased_imputed_raw_out', and unnecessary files removed. See 'potential problem 5' to resolve issues accessing plink format data of peds and maps.

6. Extract heterozygous SNPs and make a heterozygous SNP Bed file for gene-level ASE calculation using **1.3_Make_het_cds.pl**
   a. Make a heterozygous SNP Bed file using the command below, ensuring that the cancer type is specified after the parameter '-c'.

```
> ./1.3_Make_het_cds.pl -c BRCA
```

7. Extract copy number estimates to call CNVs using **2.0_Prep_For_Bad_SNPs_CNVs.pl** (optional).

   Note: Copy number estimate files must be downloaded as described above using the script '0_Download_SNPArray_From_GDC.pl' before proceeding.

   a. Extract copy number estimates using the command below and specifying the cancer type after the parameter '-c'.

```
> ./2.0_Prep_For_Bad_SNPs_CNVs.pl -c BRCA
```

8. Identify SNPs that overlap CNVs for ASE evaluation using **2.1_Get_Bad_SNPs_and_CNVs.pl**
   a. Identify SNPs that overlap CNVs with the command below, ensuring that the cancer type is specified after the parameter '-c'.

```
> ./2.1_Get_Bad_SNPs_and_CNVs.pl -c BRCA
```

   Note: This script requires outputs from scripts 1.1, 1.2, 1.3 and 2.0 to run.

9. Download RNA-Seq or WGS data and run mpileup on them to aggregate reads by genomic co-ordinates using **3.0_Download_RNASeq_WGS_and_do_Mpileup.pl**
   a. Download data and run mpileup as shown below, specifying:
      i. cancer type with '-c'
      ii. data type with '-E' (e.g., 'RNA-Seq' or 'WGS')
      iii. path to the gdc key with '-k'
      iv. total number of BAMs for downloading with the parameter '-n'
      v. '-i' 'yes' to download samples with available RNA-Seq and WGS data (optional)
      vi. Use '-o download' to download BAM files without running mpileups or '-o all' to download BAMs and do mpileups (optional)
      vii. '-d aria' to download with it instead of the default, '-d curl' (optional)

```
#download RNA-Seq BAMs and do mpileup

> ./3.0_Download_RNASeq_WGS_and_do_Mpileup.pl -E RNA-Seq \

> -c BRCA -o all -i yes -n 50 -d curl \

> -k /Driver-ASE/Database/gdc.key

#download WGS BAMs and call somatic mutations

> ./3.0_Download_RNASeq_WGS_and_do_Mpileup.pl -E WGS \

> -c BRCA -o all -i yes -n 10 -d curl -k /Driver-ASE/Database/gdc.key
```

   Note: The script uses preexisting sample tables or downloads the sample table from GDC for the specified cancer type, and downloads BAM files for the samples included in the table.

After downloading and indexing the specified BAM files, the script will run samtools mpileups on them.

b. Create an RNA-Seq or WGS annotation table of GDC sample ids and corresponding BAM ids using 3.0_Dwnld_RNASeq_WGS_Table_From_GDC.pl as shown below. Specify:
   i. cancer type with '-c' (or multiple types separated by commas: e.g., OV, PRAD)
   ii. experimental strategy with '-E' (either 'RNA-Seq' or 'WGS')
   iii. gdc key path with '-k' (optional)
   iv. '-d aria' to download with it instead of the default, '-d curl' (optional)
   v. '-o' to restrict analysis to samples with overlapping WGS and RNA-seq (optional)

Note: The wrapper shell script, '3.0_Download_RNAseq_WGS_Tables_From_GDC.sh', will run the commands of '3.0_Dwnld_RNASeq_WGS_Table_From_GDC.pl' for all cancer types from GDC legacy portal.

```
#download RNA-Seq sample table from GDC

> ./3.0_Dwnld_RNASeq_WGS_Table_From_GDC.pl -E RNA-Seq \

> -c BRCA -d curl -o no -k /Driver-ASE/Database/gdc.key

#download WGS sample table from GDC and overlap WGS table with RNA-Seq samples

> ./3.0_Dwnld_RNASeq_WGS_Table_From_GDC.pl -E WGS -c BRCA \

> -d curl -o yes -k /Driver-ASE/Database/gdc.key
```

10. Perform gene-level ASE analysis using **3.1_Gene_Level_ASE_Analysis.pl**
    a. Run gene-level ASE analysis using the command below and specifying the cancer type after the parameter '-c'. Restrict the gene-level ASE evaluation to only samples with overlapping RNA-Seq, WGS and Genotype TCGA IDs using '-o'.

```
#download RNA-Seq sample table from GDC

> ./3.1_Gene_Level_ASE_Analysis.pl -c BRCA -o yes
```

11. Prepare the data for analysis in MATLAB using **3.2_Export_ASE_Data.pl**
    a. Prepare for MATLAB analysis using the command below, ensuring that the cancer type '-c' is specified. If intersecting RNA-Seq, WGS, and Genotype data tables, specify '-o yes' and ensure the TCGA sample IDs are present in each dataset.
    b. Include annotation of SNPs associated with CNV by running scripts 0, 2.0 and 2.1 to download and process GDC copy number data.

```
> ./3.2_Export_ASE_Data.pl -c BRCA
```

12. Filter the Varscan2 mutation calls for somatic mutations using **4.0_Somatic_Variants.pl**
    a. Filter the Varscan2 mutation calls using the command below and specifying the cancer type '-c'. Optionally, specify:
       i. read cutoff with '–readcutoff'
       ii. tumor frequency with '–tfreq'
       iii. normal frequency of mutated allele with '–nfreq'

iv. restrict filtering of somatic mutations to only samples with overlapping RNA-Seq, WGS and Genotypes TCGA IDs, use '-o'.

```
> ./4.0_Somatic_Variants.pl -c BRCA –readcutoff 20 –tfreq 0.2 –nfreq 0.02 -o yes
```

13. Map somatic mutations to different regulatory regions, including TF-binding site, CpG islands, DNase sensitive sites, broad histone modification regions, 3'UTR, 5'UTR, coding regions, intronic regions from UCSC or generated based on ENCODE data using **4.1_Upstream_Downstream_Analysis.pl**
    a. Map somatic mutations to different regulatory regions using the command below ensuring the cancer type is specified with '-c'.

```
> ./4.1_Upstream_Downstream_Analysis.pl -c BRCA
```

Note: The bed files (hg19) are pre-built for each feature and included in the directory '/Driver-ASE/Database/reg' from 'Database.zip'. Users can replace these bed files with custom files as necessary.

Associate gene-level ASE with somatic mutations

Timing: [20 mins]

Gene-level ASE and somatic mutations in different regulatory regions can be correlated with M1_Import_ASE_and_Mutation_data.m or its executable file. MATLAB version 2014b or more recent versions will run the script. Alternatively, the executable file 'M1_Import_ASE_and_Mutation_data' can be run with MATLAB Runtime version 90 in a local Linux system or online cluster.

14. Download the MATLAB scripts ('Driver_ASE_MatLab_Lib') and its corresponding database ('MatLab_Analysis.zip') from GitHub (https://github.com/mikevermeulen/Driver-ASE/). If users installed Driver-ASE with the shell script, 'Driver_ASE_Installation_with_conda.sh', these directories and data are automatically available.
    a. Decompress the association scripts to a new directory 'MatLab_Analysis' by double clicking and extracting all files or the 'unzip MatLab_Analysis.zip' command in Linux.
    b. Create a subdirectory for each cancer type (e.g., 'BRCA' was created for demo; n=46) with required nested directories of 'somatic_calls', 'mutations', 'annotations' and 'matrix' (Figure 5).
15. Make Driver_ASE functions and scripts accessible to MATLAB by adding the directory path containing relevant subfolders ('Driver_ASE_MatLab_Lib', 'MatLab_Variables', 'Driver_ASE_MatLab_Scripts') to the MATLAB environment variable path.
    a. In MATLAB, click 'Home' -> 'Set Path' -> 'Add folder' sequentially to add the directory 'Driver_ASE_MatLab_Lib' with default setting. These steps are illustrated in Figure 6.
16. Start MATLAB, navigate to the Driver_ASE_MatLab_Scripts' directory, and run the function: 'M1_Import_ASE_and_Mutation_data.m'.
    a. Specify the cancer type ('cancer_type') and regulatory feature ('rgx4feature') parameters for the ASE-Mut association analysis. Supply a custom FDR cutoff if desired, the default is 0.2. To evaluate all 18 regulatory or genomic features from '4.1_Upstream_Downstream_Analysis.pl', the user can specify '.*' (see Figure 7).

```
%In matlab
> cd 'your_fullpath_to_the_dir_Driver_ASE_MatLab_Scripts'
> cancer_type='BRCA';
> rgx4feature='.*';
> fdrcutoff=0.2;
%FDR cutoff can be supplied after rgx4feature, with the
default value being 0.2.
>
[All_Assoc,Top_assoc]=M1_Import_ASE_and_Mutation_data(canc
er_type,rgx4feature,fdrcutoff);
```

Note: For the association results, MATLAB outputs all ('All_Assoc') and the top-ranked ('Top_assoc') ASE-Mut association results separately for each tested feature (FDR <= user-defined FDR cutoff in each tested feature).

17. Run the MATLAB executable function to associate gene-level ASE with somatic mutations.
    a. Export the global variable 'LD_LIBRARY_PATH' for MATLAB runtime.
    b. Navigate to the working directory 'Driver_ASE_MatLab_Scripts' where the executable is located and follow the commands below to perform the association.

```
> export
LD_LIBRARY_PATH=$MCR_v90/runtime/glnxa64:$MCR_v90/bin/glnxa64:$MCR_v90/sys/os/glnxa6
4:$MCR_v90/sys/opengl/lib/glnxa64
> cd
/Driver_ASE/Driver_ASE_MatLab/Driver_ASE_MatLab_Lib/Driver_A
SE_MatLab_Scripts
#Parameters required by the executable file.
#./M1_Import_ASE_and_Mutation_data <cancer type, such as
BRCA> <feature_rgx, such as chipseq> <FDR cutoff, e.g., 0.2>
#This will print help information for the MATLAB executable
file;
> ./M1_Import_ASE_and_Mutation_data
#When running the following command in a Linux terminal, the
MATLAB executable will generate the same results as that
executing in MATLAB shown in the previous section of
'Expected Outcomes'.
#Note: the regular expression '.*'' will match all 18
regulatory features
> ./M1_Import_ASE_and_Mutation_data BRCA '.*' 0.2
```

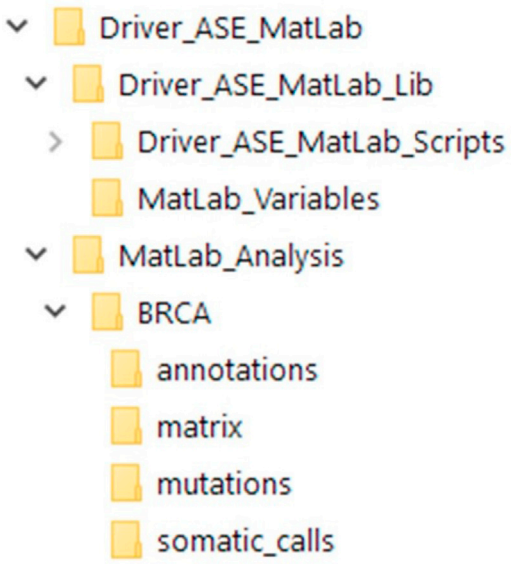


Figure 5. Directory tree of Driver-ASE to associate gene-level ASE with somatic mutations

Note: MATLAB runtime version 90 must be installed in a local Linux system or online cluster, the installation of which should be completed when running the shell script for the installation of Driver-ASE in the system via conda. See 'potential problem 6' to resolve issues in running the MATLAB executable file in a local Linux system.

EXPECTED OUTCOMES

Summary figures and MATLAB datasets are automatically generated by running 'M1_Import_ASE_and_Mutation_data.m' (Figures 8, 9, 10, and 11).

LIMITATIONS

Computational Resources. The Docker pipeline requests >8 GB memory and >100 GB disk space to perform imputation, calculate gene-level ASE and call somatic mutations from the respective SNP genotype data, RNA-Seq and WGS BAMs. As these calculations are computationally intensive we recommend running the pipeline on a high-performance cluster. For additional detail on how to install Driver-ASE and its dependencies onto a cluster via conda, please see the Driver-ASE github homepage (https://github.com/michaelcvermeulen/driver-ase).

Data access. A downloading key is a prerequisite for downloading controlled GDC data. Instructions on how to apply for GDC access can be found at the GDC website (https://portal.gdc.cancer.gov/). To determine if samples of interest are available from GDC before applying for GDC access, users can search sample numbers and types by running the scripts '**0_Download_SNPArray_Table_From_GDC.pl**' and '**3.0_Dwnld_RNASeq_WGS_Table_From_GDC.pl**'. A toy GDC key is available in the 'Database' directory to assist the downloading of these sample tables.

Computational time. Downloading GDC data and calling somatic mutations both consume considerable time and computational resources. We recommend processing a small number of WGS and RNA-Seq files to guide the resource requirements for future analyses.

Mutations and ASE inputs. The default Driver-ASE pipeline screens for driver mutations by integrating orthogonal GDC datasets, including SNPArray data, RNA-Seq, and WGS. To input gene-level ASE and somatic mutations into the Driver-ASE pipeline, we offer a separate package ('Pipeline2MapMuts2DiffRegulatoryRegions') to prepare gene-level ASE and map somatic mutations called by alternative software; the package is available via Mendeley Data: https://doi.

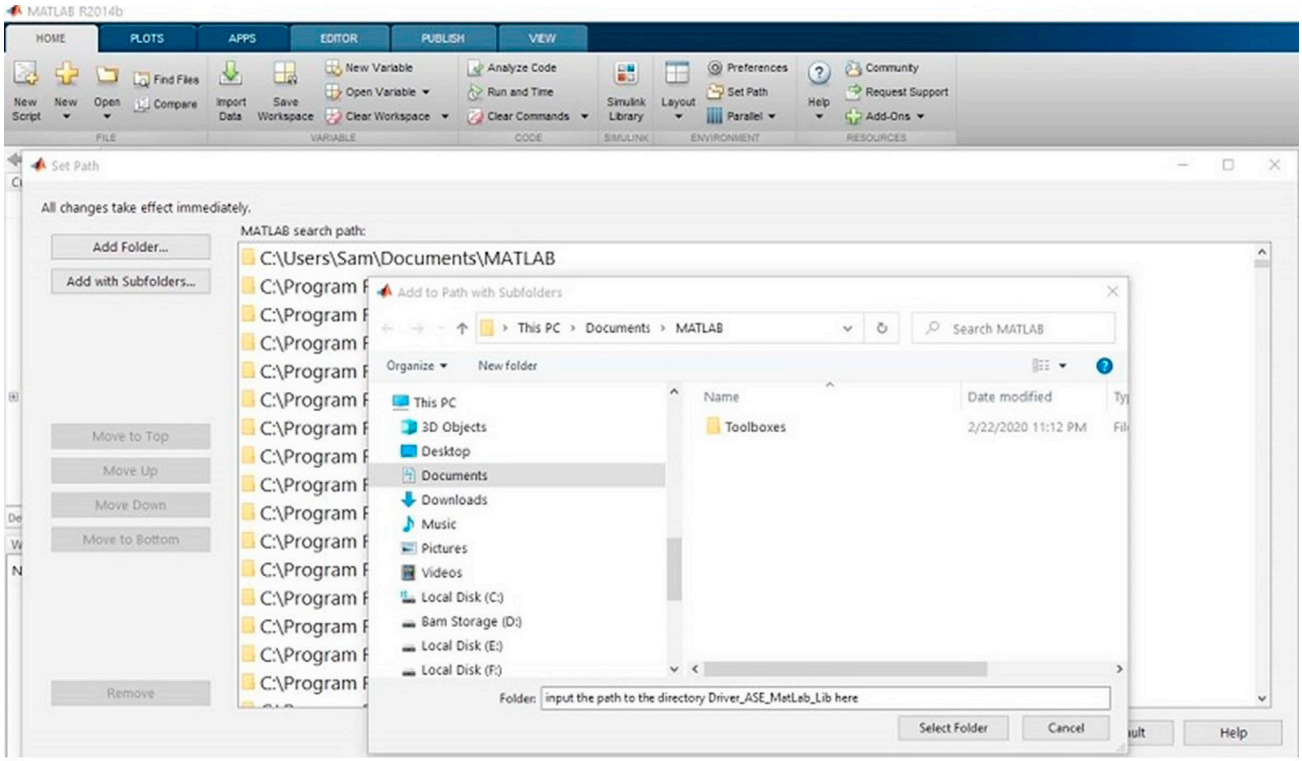


Figure 6. Screenshot of how to add the Driver_ASE scripts into the MATLAB global path

org/10.17632/8x3y5swppw.4. Driver-ASE only process data mapped to human hg19 reference. For hg38 input data, we recommend liftover into hg19 using CrossMap (Zhao et al., 2014).

Operating system compatibility. We have confirmed that the Driver-ASE Docker image runs on Windows, Mac and Linux operating systems but have not evaluated others. We also automated the installation of Driver-ASE via conda for local or online Linux computational clusters.

TROUBLESHOOTING

Problem 1

An error may occur when activating the driverase conda environment if conda versions prior to version 4.4 were used for the installation (corresponding protocol step: Install Driver-ASE using conda 6d). The error might be as follows:

CommandNotFoundError: Your shell has not been properly configured to use 'conda activate'.

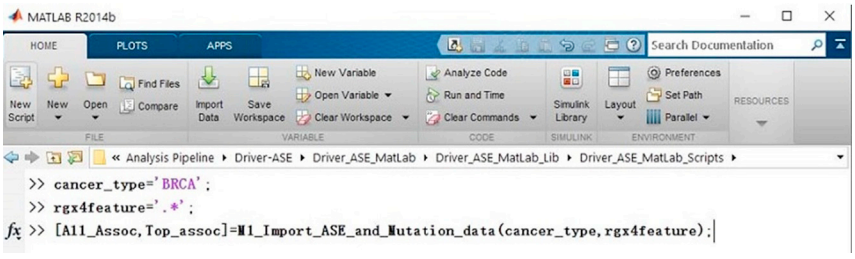


Figure 7. Sample MATLAB commands to run ASE-Mut association on BRCA data

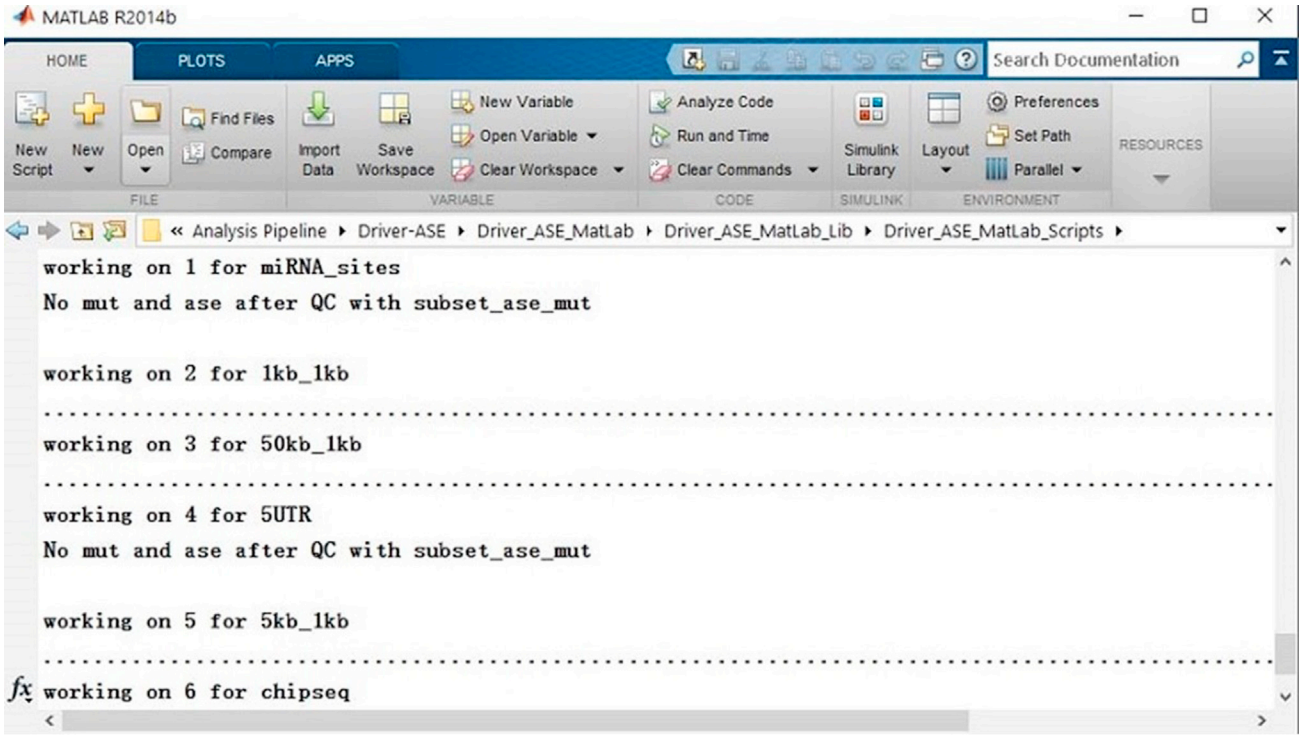


Figure 8. Sample update messages after executing ASE-Mut association across different regulatory features

Potential solution

Use the following approach to address the issue:

(1) Run the command below in docker image to enable all users miniconda access:

```
> sudo ln -s /usr/local/bin/anaconda2/etc/profile.d/conda.sh

/etc/profile.d/conda.sh
```

(2) Add the bin path of miniconda, such as '/usr/local/bin/anaconda2/bin' in our Docker image, to the global $PATH in the file '~/.bashrc':

```
> export PATH=/usr/local/bin/anaconda2/bin:$PATH
```

(3) Write the command 'conda activate' at the end of the file '~/.bashrc'.

(4) Exit Linux terminal and rerun the command 'conda activate driverase'.

Problem 2

The following error message may emerge when running the perl script '0_Download_SNPArray_Table_From_GDC.pl' (corresponding protocol step: Driver-ASE workflow 7b):

It seems that there is a table in the /Driver_ASE/Analysis/BRCA/BRCA_tables directory already. Table: BRCA.Genotypes.id2uuid.txt

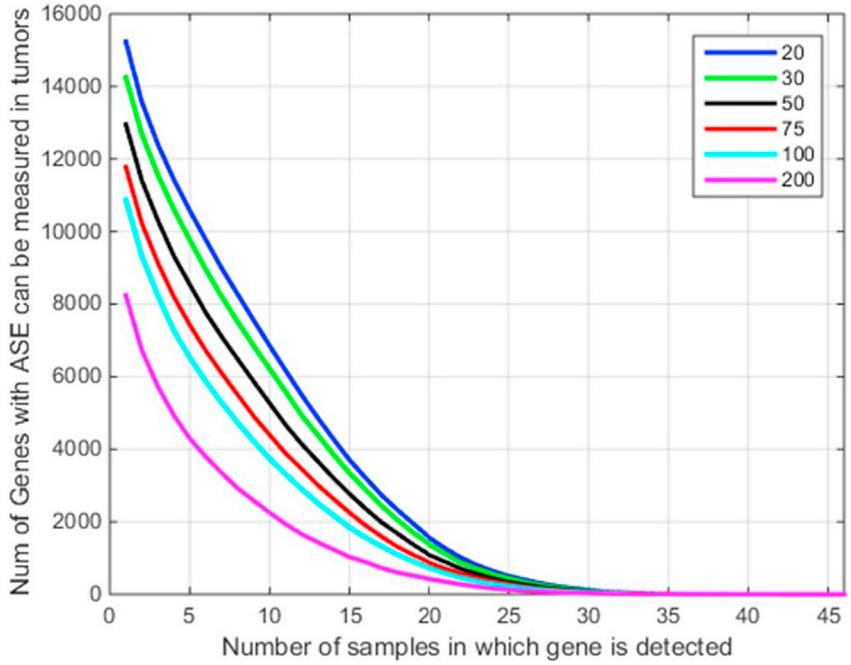


Figure 9. Output of '*M1_Import_ASE_and_Mutation_data.m*' applied to BRCA demonstration data

The plot illustrates the number of genes for which ASE that can be measured in tumors grouped by sample size and gene-level ASE read counts of heterozygous SNPs.

Potential solution

The problem is due to pre-downloaded table residing in the directory '/Driver_ASE/Analysis/BRCA' that we provided to save user time. If users want to download these tables again, please delete the table as show below and rerun the perl download script.

```
> ls /Driver_ASE/Analysis

# remove existing tables

> rm -rf

/Driver_ASE/Analysis/BRCA/BRCA_tables/BRCA.Genotypes.id2u

uid.txt
```

Problem 3

Unable to download the sample information from GDC (corresponding protocol step: Evaluate gene-level ASE and identify somatic mutations 1a).

The following is a typical error:

```
> ./0_Download_SNPArray_Table_From_GDC.pl -c BRCA -a Genotypes
```

```
######################################################################

/Driver-ASE/ Database does not exist! It was either deleted, moved or renamed.
```

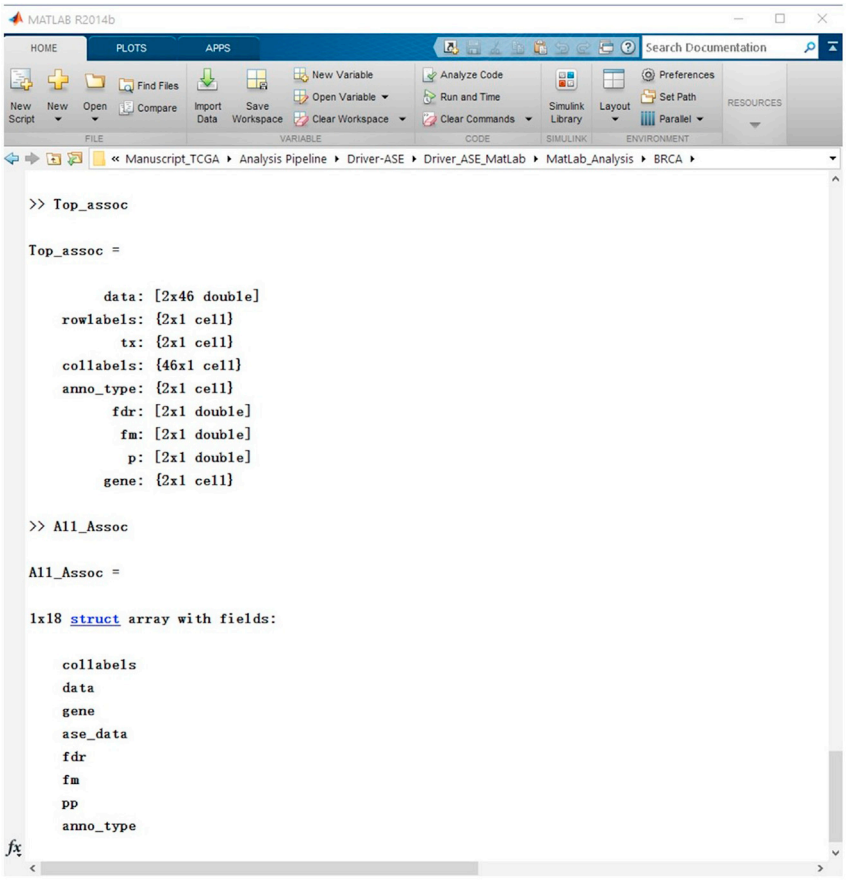


Figure 10. Evaluation of ASE-Mutation associations using the MATLAB function *'M1_Import_ASE_and_Mutation_data.m'*

Typing 'Top_assoc' in MATLAB terminal shows the top associations (false discovery rate [FDR]<=0.2 and raw association p value [p]<=0.05) across 18 different regulatory or genomic features after running the scripts in this vignette with default settings. All association results are also included in the MATLAB structure 'All_Assoc'.

If the missing directory is the Database directory, it may not have been downloaded or the name was changed. Check the README.md file on the github page to find out where to get the Database directory.

##

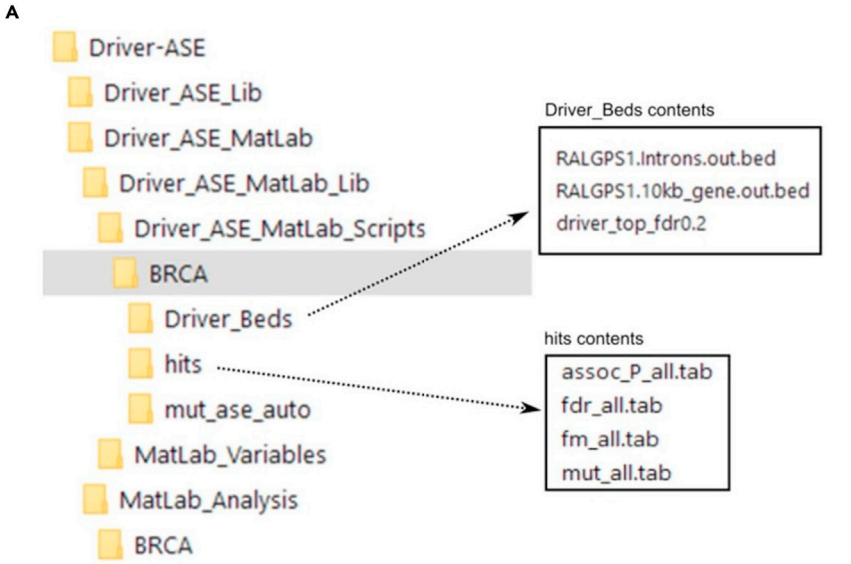
Potential solution

The perl script cannot find the 'Database' directory. Go to Driver-ASE github page, download 'Database.zip', and decompress 'Database' in the path '/Driver-ASE/Database' of the Docker image. Alternatively, please run the shell script 'Driver_ASE_Installation_with_conda.sh' in the force mode from the docker terminal to install the 'Database' automatically (see installation section).

Problem 4

GDC authentication error (corresponding protocol step of Evaluate gene-level ASE and identify somatic mutations 2a).

```
> ./0_Download_SNPArray_Table_From_GDC.pl -c BRCA -a Genotypes
```



B

Data in 'assoc_P_all.tab'

Rowlabels	-LOG10(p)
RALGPS1-10kb_gene	5.9066
RALGPS1-Introns	5.9066
KIAA1462-10kb_gene	3.6134
KIAA1462-Introns	3.6134
TIPRL-50kb_1kb	3.3594
TIPRL-10kb_gene	3.3594
C6orf62-50kb_1kb	3.3478
C6orf62-10kb_gene	3.3478
EDIL3-DNAseClustered	3.2668
NAV1-10kb_gene	3.2164

Data in 'fdr_all.tab'

Rowlabels	FDR
RALGPS1-Introns	0.13508
RALGPS1-10kb_gene	0.17644
DNAL4-10kb_1kb	0.25841
MIR15A-1kb_1kb	0.33871
ZRANB2-5kb_1kb	0.3392
NM_001243900-Exons	0.37756
MED14-50kb_1kb	0.42689
C6orf192-50kb_1kb	0.43783
CENPP-50kb_1kb	0.44198
ALPK2-50kb_1kb	0.44254

Data in 'fm_all.tab'

Rowlabels	FDR-mut
PTPRD-10kb_gene	0.001278
CNTNAP2-10kb_gene	0.001278
PTPRD-Introns	0.001468
CNTNAP2-Introns	0.001468
CENPP-50kb_1kb	0.00271
GPC6-Introns	0.002937
CSMD1-Introns	0.002937
DMD-10kb_gene	0.003195
GPC6-10kb_gene	0.003195

Data in 'mut_all.tab'

Rowlabels	TCGA-A1-A0SM	TCGA-A2-A04P	TCGA-A2-A04Q	TCGA-A2-A04X	TCGA-A2-A0D1	TCGA-A2-A0EY	TCGA-A2-A3Y0	TCGA-A7-A0CE	TCGA-A7-A0D9	TCGA-A7-A13D
AAK1-10kb_gene	0	1	0	0	1	1	0	0	0	1
AAK1-DNAseClustered	0	1	0	0	0	1	0	0	0	0
AAK1-Introns	0	1	0	0	1	1	0	0	0	1
AATF-10kb_gene	0	0	0	0	0	0	1	0	0	1
AATF-Introns	0	0	0	0	0	0	1	0	0	1
ABAT-10kb_gene	1	0	0	1	0	0	0	0	0	0
ABAT-Introns	0	0	0	1	0	0	0	0	0	0
ABCA4-10kb_gene	0	1	0	0	0	0	0	0	0	1
ABCA4-Introns	0	1	0	0	0	0	0	0	0	1
ABCC1-10kb_gene	0	0	0	0	0	1	0	0	0	1

Figure 11. The directory tree of the final ASE-Mutation results

(A) The ASE-Mutation associations are saved in the 'BRCA' directory that includes 3 subdirectories: 'Driver_Beds', 'hits', and 'mut_ase_auto'. After running the pipeline the 'Driver_beds' directory will contain one text file of all associations FDR<0.2 (driver_top_fdr0.2), and a bed file for each association between a mutated cis-regulatory element and gene-level ASE. For example, the upper box shows an association between *RALGPS1* and mutations within 10kb of its TSS and gene body that is found using the demo data of 46 BRCA samples. The bed files of putative driver mutations can be visualized with the UCSC or alternate genome browsers (hg19). The directory 'hits' will contain all ASE-Mut association results as shown in the lower panel.

(B) The raw association p-values (assoc_P_all.tab), FDR values (fdr_all.tab), mutation enrichment p-values for each feature with each gene (fm_all.tab), and information for samples harboring these regulatory mutations (mut_all.tab) are output into the 'hits' directory. The 'mut_ase_auto' directory contains the 'mutation x regulatory-feature' MATLAB matrix.

The error is as follows:

```
curl -header 'X-Auth-Token: ############put your key to replace this line!
```

Potential solution

Replace the toy GDC key with the user key 'gdc.key' in the '/Driver-ASE/Database' directory. Keep the same key filename.

Problem 5

Missing plink format data of peds and maps. After execution of the '1.2_Shapeit_and_Imputation.pl' script (corresponding protocol step of Evaluate gene-level ASE and identify somatic mutations 5a), the following error appears:

```
/Driver-ASE/Analysis/BRCA/RNA_Seq_Analysis/maps does not exist! It was either deleted, moved or renamed.

If /Driver-ASE/Analysis/BRCA/RNA_Seq_Analysis/maps is missing, It is likely that it was deleted, moved, renamed or the scripts were ran out of order.
```

Potential solution

This error occurs when the required plink formatted data is not in the 'peds' and 'maps' directories. Run the perl script '1.1_Birdseed_to_ped_and_maps.pl' before executing the perl script '1.2_Shapeit_and_Imputation.pl'.

Problem 6

A typical error may occur when running the MATLAB executable file in a Local Linux system (corresponding protocol step: Associate gene-level ASE with somatic mutations 17b), such as Ubuntu 20.04 and CentOS 8, which is listed here:

./M1_Import_ASE_and_Mutation_data: error while loading shared libraries: libncurses.so.5: cannot open shared object file: No such file or directory.

Potential solution

This is due to the MATLAB runtime v90 requires the library libncurses5 to be installed in the Linux system. In Ubuntu or CentOS, it can be addressed by installing the dependency via the following command:

```
#In Ubuntu Linux system

> sudo apt-get install libncurses5

#In CentOS Linux system

> yum install libncurses5
```

RESOURCE AVAILABILITY

Lead contact

Further information and resource requests should be directed to and will be fulfilled by the lead contact, Brian DeVeale (brian.deveale@ucsf.edu).

Materials availability

No newly generated data were produced by this protocol.

Data and code availability

The protocol includes all datasets generated or analyzed during this study that are accessible at: https://github.com/michaelcvermeulen/driver-ase.

ACKNOWLEDGMENTS

We thank the Canadian Cancer Society for funding this research (CBCF grant BC-RG-15-2, PI: T.B.).

AUTHOR CONTRIBUTIONS

Z.C., M.V., M.R.G., T.B, and B.D. assembled the Driver-ASE pipeline and wrote the manuscript.

DECLARATION OF INTERESTS

The authors declare no competing interests.

REFERENCES

Chang, C.C., Chow, C.C., Tellier, L.C., Vattikuti, S., Purcell, S.M., and Lee, J.J. (2015). Second-generation PLINK: rising to the challenge of larger and richer datasets. Gigascience *4*, 7.

Cheng, Z., Vermeulen, M., Rollins-Green, M., DeVeale, B., and Babak, T. (2021). Cis-regulatory mutations with driver hallmarks in major cancers. iScience *24*, 102144.

Delaneau, O., Marchini, J., and Zagury, J.F. (2011). A linear complexity phasing method for thousands of genomes. Nat. Methods *9*, 179–181.

Delaneau, O., Zagury, J.F., and Marchini, J. (2013). Improved whole-chromosome phasing for disease and population genetic studies. Nature Methods *10*, 5–6.

Gao, G.F., Parker, J.S., Reynolds, S.M., Silva, T.C., Wang, L.B., Zhou, W., Akbani, R., Bailey, M., Balu, S., Berman, B.P., et al. (2019). Before and after: Comparison of legacy and Harmonized TCGA genomic data Commons' data. Cell Syst. *9*, 24–34 e10.

Kent, W.J., Sugnet, C.W., Furey, T.S., Roskin, K.M., Pringle, T.H., Zahler, A.M., and Haussler, D. (2002). The human genome browser at UCSC. Genome Res. *12*, 996–1006.

Koboldt, D.C., Zhang, Q., Larson, D.E., Shen, D., McLellan, M.D., Lin, L., Miller, C.A., Mardis, E.R., Ding, L., and Wilson, R.K. (2012). VarScan 2: somatic mutation and copy number alteration discovery in cancer by exome sequencing. Genome Res. *22*, 568–576.

Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., Marth, G., Abecasis, G., Durbin, R., and Genome Project Data Processing, S. (2009). The sequence Alignment/map format and SAMtools. Bioinformatics *25*, 2078–2079.

Marchini, J., and Howie, B. (2010). Genotype imputation for genome-wide association studies. Nature Review Genetics *11*, 499–511.

Marchini, J., Howie, B., Myers, S., McVean, G., and Donnelly, P. (2007). A new multipoint method for genome-wide association studies by imputation of genotypes. Nat. Genet. *39*, 906–913.

Reble, E., Castellani, C.A., Melka, M.G., O'Reilly, R., and Singh, S.M. (2017). VarScan2 analysis of de novo variants in monozygotic twins discordant for schizophrenia. Psychiatr. Genet. *27*, 62–70.

Zhao, H., Sun, Z., Wang, J., Huang, H., Kocher, J.P., and Wang, L. (2014). CrossMap: a versatile tool for coordinate conversion between genome assemblies. Bioinformatics *30*, 1006–1007.