Transcriptome-Wide Association Study Pipeline (TWAS-pipeline)

INSTALLATIONS

First, we need to install TWAS by the developers

TWAS

http://sashagusev.github.io/TWAS/

weight files

https://data.broadinstitute.org/alkesgroup/TWAS/

z-score clean program

https://data.broadinstitute.org/alkesgroup/TWAS/ETC/CLEAN_ZSCORES.tar.bz2

are required and be unpacked. In addition, lists of genes in the three populations are made through the following scripts,

```
TWAS=/genetics/bin/TWAS
cd $TWAS
for pop in MET NTR YFS
do
  ls WEIGHTS_$1 | sed 's\/\\g' > $pop.lst
done
```

TWAS-pipeline is installed as follows,

```
git clone https://github.com/jinghuazhao/TWAS-pipeline
```

On our system, `TWAS.sh` and `TWAS_get_weights.sh` for `TWAS` and `twas.sh`, `twas2.sh`, `twas2-collect.sh` and `twas2-1.sh` for `TWAS-pipeline` have symbolic links under `/genetics/bin` and available from the $PATH environment. A Stata equivalent has been developed by Dr Jian'an Luan.

To accommodate the suggestion of p value in accordance with the Z-score in the output, `pnorm.c` is included which can be compiled as follows,

```
gcc pnorm.c -lm -o pnorm
```

and a call `pnorm z_score` yields a p value with more decimal places.

GNU Parallel. Further information is available from here.

RUNNING THE PIPELINE

Suppose you have a file containing GWAS summary statistics, you cna run the pipeline as follows,

```
twas.sh input_file
```

where the `input_file` is in tab-delimited format containing SNP_name, SNP_pos, Ref_allele, Alt_allele, Beta and SE. The output will be contained in `<input file>.imp`.

This assumes that ssh can access nodes in a clusters freely and in case this has not been done, a single node mode is more appropriate,

```
twas-single.sh input_file
```

BUILDING REFERENCE PANEL

The weights have to be generated in general. The software TWAS contains two command files:

- `TWAS_get_weights.sh`, which obtains weights (.ld, .cor, .map) from PLINK map/ped pair given a particular locus. It actually wraps up a program in R.
- `TWAS.sh`, which conducts imputatation as reported in the Gusev et al. (2016).

Minor changes to the scripts may be required for your own data. The tasks involved are to

- extract SNPs in a gene from 1000Genomes imputed data into PLINK map/ped files
- obtain .ld, .cor and .map with `TWAS_get_weights.sh` for that gene
- select summary statistics (.zscore) for the gene
- conduct imputation with `TWAS.sh` into file .imp
- repeat above steps for all genes and collect results

From UCSC, you obtain the gene bounaries as follows,

```
mysql --user=genome --host=genome-mysql.cse.ucsc.edu -A -D hg19 -e 'select *
from refGene' > refGene.txt
```

However, it is often necessary to define a region using a list of SNPs. In this regard, tables such as `snp146` in `hg19` above are needed. From `locuszoom-1.3` (Pruim, et al. 2010) we can extract `refFlat.txt` and `snp_pos.txt` (see `lz.sql`) to build a list of SNP-gene pairs, as with (UK BioBank Axiom chip) `Axiom_UKB_WCSG.na34.annot.csv.zip`. Their chromosome-specific counterparts as with SNPs under all genes can also be derived. A `Stata` program `lz.do` which calls `refGene.do` is developed in collaboration with Dr Jian'an Luan to faciliate handling of gene boundaries.

An example is provided on a recent study of body bone mineral density (TBBMD). The relevant files all have prefix bmd- and some are listed as follows,

Files	Description
`bmd.sh`	to generate chromosome-specific z-scores
`bmd.do`	Stata program to flag non-missing individuals
`bmd/TBBMD.gz`	the GWAS summary statistics
`bmd-twas.sh`	script for TWAS by SNP

`bmd-twas2.sh` region selection based on position rather than rsid

`bmd-summary.sh` To put together all imputation results into bmd.imp

The automation would involve `bmd-twas.sh` and `bmd-twas2.sh`.

AN EXPOSITION WITH GIANT DATA

The example shows details of the implementation (see `giant.sh`). The GIANT consortium study of BMI on Europeans led to the following tab-delimited summary statistics, sorted by SNPs, as in Locke, et al. (2015), called BMI-EUR.gz in brief,

```
SNP A1  A2  Freq1.Hapmap    b   se  p   N
rs1000000   G   A   0.6333  1e-04   0.0044  0.9819  231410
rs10000010  T   C   0.575   -0.0029 0.003   0.3374  322079
rs10000012  G   C   0.1917  -0.0095 0.0054  0.07853 233933
rs10000013  A   C   0.8333  -0.0095 0.0044  0.03084 233886
...
```

from which we generated the following z-score file `EUR/bmi.txt`:

```
rs10    C   A   -0.571429
rs1000000   G   A   0.0227273
rs10000010  T   C   -0.966667
rs10000012  G   C   -1.75926
rs10000013  A   C   -2.15909
...
```

Now that the GWAS summary statistics file contains no SNP positions, but has already been sorted by SNP id and aligned by strand, we can then call `twas2.sh` as follows,

```
mkdir -p EUR/MET
ln -sf EUR/bmi.txt EUR/MET/twas2.txt
dir=`pwd`
twas2.sh $TWAS $TWAS2 $dir/EUR MET 1
```

where MET specifies weights from METSIM population as in Gusev et al. (2016) and we start from block 1 of the gene list involving 25 genes.

Again we resort to parallel computing for all blocks,

```
parallel -j8 twas2.sh {1} {2} {3} {4} {5} ::: $TWAS ::: $TWAS2 ::: $dir/EUR
::: MET ::: $(seq 1000)
```

where we iterate through all sets of weight (MET, NTR and YFS) using 8 CPUs.

If we provide `ALL/bmi.txt` based on all population results, called BMI-ALL.gz in brief, and create all the necessary links as bove, then we simply replace $dir/EUR with $dir/EUR $dir/ALL in the call to parallel above.

The imputation resuls are available from

```
twas2-collect.sh EUR
twas2-collect.sh ALL
```

In particular, imputation can also be done for a specific gene, e.g., BRCA1 and YFS:

```
twas2-1.sh $TWAS $TWAS2 $dir/EUR YFS BRCA1
```

so the results are written into BRCA1/YFS/BRCA1.imp. Note that by doing so, intermediate files with extensions `.join`, `.sort`, `.zscore` are available for check

TWAS using GTEx

This is achieved with `gtex.sh` and `gtex.subs` using weights from the GTEx project. File `GTEX_WEIGHTS.bim.gz` was created by `GTEX_WEIGHTS.sh` in accordance with reference `.bim` files used by `TWAS` and `GTEX_WEIGHTS.lst` was created to facilitate the imputation.

REFERENCES

Locke AM, et al.(2015). Genetic studies of body mass index yield new insights for obesity biology. Nature, 518, 197-206

Gusev A, et al. (2016). Integrative approaches for large-scale transcriptome-wide association studies. Nature Genetics, 48, 245-252

Mancuso N, et al. (2017). Integrating gene expression with summary association statistics to identify susceptibility genes for 30 complex traits. American Journal of Human Genetics, 2017, 100, 473-487, http://www.cell.com/ajhg/fulltext/S0002-9297(17)30032-0. See also http://biorxiv.org/content/early/2016/09/01/072967 or http://dx.doi.org/10.1101/072967.

Pruim RJ, et al. (2010). LocusZoom: regional visualization of genome-wide association scan results. Bioinformatics, 26,2336-2337

EPIGENOMEWIDE ASSOCIATION

This is furnished with `ewas.sh` and `ewas.subs`, along with a few other files as follows,

Files	Description
ewas.sh	EWAS imputation
ewas.subs	subroutine called by ewas.sh
get_weight.qsub	SGE script for EWAS weight generation
get_weight_subs	subroutine callable from `get_weight.qsub` and `parallel`
CpG.lst	list of probe IDs with weights
weights/	directory

	containing weights for all probes as specfied in `CpG.lst`
EWAS/	directory containing `PLINK` binary files for each probe
EWAS.pheno	`PLINK` phenotype file with header for all probes
EWAS.bim	`PLINK` .bim file sorted by SNP IDs

This implementation used the same idea as TWAS. Data from 1000Genomes imputation were scaled down to those in HapMap II to make the weight generation more tenable to sample size. Note that weights were obtained for all probes so it is possible to impute for only subset(s) of them. The file `EWAS.bim` was generated in order to make it easier to align strands for SNPs as in GWAS with those in the reference panel.

FUSION pipeline

This follows Mancuso N, et al. (2017) to use FUSION for gene expressioin analysis, whose associate software available from https://github.com/gusevlab/fusion_twas.

Files	Description
GE.runlist	detailed list of jobs
ge-fusion.sh	driver
ge-fusion.qsub	qsub script
ge-fusion.subs	subroutine
GTEx.runlist	detailed list of jobs
gtex-fusion.sh	driver
gtex-fusion.qsub	sge routine
gtex-fusion.subs	sge subroutine
gtex-fusion.sge	non-array version
gtex-fusion.awk	utility
fusion.R	utility
fusion.sh	utility

Note the ge- prefix indicates weights as in the original TWAS paper and the non-array version is meant to cover the array counterpart but hardly used.

ACKNOWLEDGEMENTS

The work is possible with an EWAS project within the MRC Epidemiology Unit, for which colleagues and collaborators have contributed.