main

September 13, 2021

1 Summary of interacting cis-eQTL analysis

[1]:

```
import functools
import pandas as pd
```

[ ]:

```

```

1.1 Functions

1.1.1 Cached functions

[2]:

```
@functools.lru_cache()
def get_mashr_eqtls(feature, tissue):
    cols = ["effect", "gene_id", "variant_id", tissue]
    df = pd.read_csv("../../_m/%s/lfsr_allpairs_3tissues.txt.gz" % feature,
                     sep='\t').loc[:, cols]
    return df[(df[tissue] < 0.05)]


@functools.lru_cache()
def annotate_eqtls(feature, tissue):
    config = {
        "genes": "/ceph/projects/v4_phase3_paper/inputs/counts/
 ↪text_files_counts/_m/%s/gene_annotation.tsv" % tissue.lower(),
        "transcripts": "/ceph/projects/v4_phase3_paper/inputs/counts/
 ↪text_files_counts/_m/%s/tx_annotation.tsv" % tissue.lower(),
        "exons": "/ceph/projects/v4_phase3_paper/inputs/counts/
 ↪text_files_counts/_m/%s/exon_annotation.tsv" % tissue.lower(),
        "junctions": "/ceph/projects/v4_phase3_paper/inputs/counts/
 ↪text_files_counts/_m/%s/jxn_annotation.tsv" % tissue.lower(),
    }
    annot = pd.read_csv(config[feature], sep='\t').loc[:, ["names", "seqnames",␣
 ↪"gencodeID"]]
    return get_mashr_eqtls(feature, tissue).merge(annot, left_on="gene_id",
                                                  right_on="names").
 ↪drop(["names"], axis=1)
```

```python
@functools.lru_cache()
def load_pgc2():
    pgc2_file = '/ceph/projects/v4_phase3_paper/inputs/sz_gwas/'+\
                'pgc2_clozuk/map_phase3/_m/libd_hg38_pgc2sz_snps_p5e_minus8.tsv'
    return pd.read_csv(pgc2_file, sep='\t', low_memory=False, index_col=0)


@functools.lru_cache()
def merge_pgc2_N_eqtl(feature, tissue):
    return load_pgc2().merge(annotate_eqtls(feature, tissue), how='inner',
                             left_on='our_snp_id', right_on='variant_id',
                             suffixes=['_PGC2', '_eqtl'])
```

1.1.2 Simple functions

[3]:

```python
def extract_features(tissue, fnc):
    ## Extract significant eQTL using mashr
    genes = fnc("genes", tissue).rename(columns={tissue: "lfsr"})
    trans = fnc("transcripts", tissue).rename(columns={tissue: "lfsr"})
    exons = fnc("exons", tissue).rename(columns={tissue: "lfsr"})
    juncs = fnc("junctions", tissue).rename(columns={tissue: "lfsr"})
    return genes, trans, exons, juncs


def output_summary(tissue, fnc, variable):
    ## Extract eQTL using mashr
    genes, trans, exons, juncs = extract_features(tissue, fnc)
    ## Total significant eQTLs
    gg = len(set(genes[variable]))
    tt = len(set(trans[variable]))
    ee = len(set(exons[variable]))
    jj = len(set(juncs[variable]))
    print("\neGene:\t\t%d\neTranscript:\t%d\neExon:\t\t%d\neJunction:\t%d" %
          (gg, tt, ee, jj))


def get_eQTL_result_by_tissue(tissue, fnc):
    genes, trans, exons, juncs = extract_features(tissue, fnc)
    genes["Type"] = "Gene"
    trans["Type"] = "Transcript"
    exons["Type"] = "Exon"
    juncs["Type"] = "Junction"
    df = pd.concat([genes, trans, exons, juncs])
    df["Type"] = df.Type.astype("category").cat.reorder_categories(["Gene",␣
 ↪"Transcript", "Exon", "Junction"])
    df["Tissue"] = tissue
    return df
```

1.2 Caudate

1.2.1 Summarize results mashr (local false sign rate < 0.05)

```
[4]: tissue = "Caudate"
     ## significant eQTLs
     output_summary(tissue, annotate_eqtls, "effect")
     ## significant eFeatures
     output_summary(tissue, annotate_eqtls, "gene_id")
     ## significant eGenes
     output_summary(tissue, annotate_eqtls, "gencodeID")
     caudate = get_eQTL_result_by_tissue(tissue, annotate_eqtls)
```

```
eGene:          5785
eTranscript:    8525
eExon:          11364
eJunction:      2479

eGene:          950
eTranscript:    1381
eExon:          2098
eJunction:      502

eGene:          950
eTranscript:    1313
eExon:          1154
eJunction:      358
```

1.2.2 Summarize results eQTL analysis overlapping with PGC2+CLOZUK SNPs

```
[5]: ## significant eQTLs
     output_summary(tissue, merge_pgc2_N_eqtl, "effect")
     ## significant eFeatures
     output_summary(tissue, merge_pgc2_N_eqtl, "gene_id")
     ## significant eGenes
     output_summary(tissue, merge_pgc2_N_eqtl, "gencodeID")
     caudate_pgc2 = get_eQTL_result_by_tissue(tissue, merge_pgc2_N_eqtl)
```

```
eGene:          29
eTranscript:    29
eExon:          39
eJunction:      16

eGene:          3
```

```
eTranscript:    9
eExon:          7
eJunction:      10

eGene:          3
eTranscript:    9
eExon:          6
eJunction:      6
```

1.3 DLPFC

1.3.1 Summarize results mashr (local false sign rate < 0.05)

[6]:

```
tissue = "DLPFC"
## significant eQTLs
output_summary(tissue, annotate_eqtls, "effect")
## significant eFeatures
output_summary(tissue, annotate_eqtls, "gene_id")
## significant eGenes
output_summary(tissue, annotate_eqtls, "gencodeID")
dlpfc = get_eQTL_result_by_tissue(tissue, annotate_eqtls)
```

```
eGene:          5071
eTranscript:    5276
eExon:          11331
eJunction:      2437

eGene:          867
eTranscript:    982
eExon:          2085
eJunction:      489

eGene:          867
eTranscript:    940
eExon:          1137
eJunction:      342
```

1.3.2 Summarize results eQTL analysis overlapping with PGC2+CLOZUK SNPs

[7]:

```
## significant eQTLs
output_summary(tissue, merge_pgc2_N_eqtl, "effect")
## significant eFeatures
output_summary(tissue, merge_pgc2_N_eqtl, "gene_id")
## significant eGenes
output_summary(tissue, merge_pgc2_N_eqtl, "gencodeID")
dlpfc_pgc2 = get_eQTL_result_by_tissue(tissue, merge_pgc2_N_eqtl)
```

```
eGene:          29
eTranscript:    25
eExon:          40
eJunction:      16

eGene:          3
eTranscript:    7
eExon:          8
eJunction:      10

eGene:          3
eTranscript:    7
eExon:          6
eJunction:      6
```

1.4 Hippocampus

1.4.1 Summarize results mashr (local false sign rate < 0.05)

[8]:

```
tissue = "Hippocampus"
## significant eQTLs
output_summary(tissue, annotate_eqtls, "effect")
## significant eFeatures
output_summary(tissue, annotate_eqtls, "gene_id")
## significant eGenes
output_summary(tissue, annotate_eqtls, "gencodeID")
hippo = get_eQTL_result_by_tissue(tissue, annotate_eqtls)
```

```
eGene:          4821
eTranscript:    5290
eExon:          9867
eJunction:      2336

eGene:          830
eTranscript:    985
eExon:          1847
eJunction:      458

eGene:          830
eTranscript:    943
eExon:          1031
eJunction:      326
```

1.4.2 Summarize results eQTL analysis overlapping with PGC2+CLOZUK SNPs

[9]:
```
## significant eQTLs
output_summary(tissue, merge_pgc2_N_eqtl, "effect")
## significant eFeatures
output_summary(tissue, merge_pgc2_N_eqtl, "gene_id")
## significant eGenes
output_summary(tissue, merge_pgc2_N_eqtl, "gencodeID")
hippo_pgc2 = get_eQTL_result_by_tissue(tissue, merge_pgc2_N_eqtl)
```

```
eGene:          29
eTranscript:    25
eExon:          39
eJunction:      15

eGene:          3
eTranscript:    7
eExon:          7
eJunction:      10

eGene:          3
eTranscript:    7
eExon:          6
eJunction:      6
```

1.5 Save significant results

1.5.1 All associations

[10]:
```
pd.concat([caudate, dlpfc, hippo])\
  .sort_values(["Tissue", "Type", "gene_id", "lfsr"])\
  .loc[:, ["Tissue", "gene_id", "gencodeID", "variant_id", "seqnames", "lfsr",␣
↪"Type"]]\
  .to_csv("BrainSeq_sexGenotypes_4features_3regions.txt.gz", sep='\t',␣
↪index=False)
```

1.5.2 PGC2+CLOZUK associations

[11]:
```
pd.concat([caudate_pgc2, dlpfc_pgc2, hippo_pgc2])\
  .loc[:, ["Tissue", "gene_id", "gencodeID", "variant_id", "rsid", "seqnames",␣
↪"lfsr", "A1",
           "A2", "OR", "SE", "P", "pgc2_a1_same_as_our_counted",␣
↪"is_index_snp", "Type"]]\
  .sort_values(["Tissue", "Type", "gene_id", "lfsr", "P"])\
  .to_csv("BrainSeq_sexGenotypes_4features_3regions_pgc2.txt.gz", sep='\t',␣
↪index=False)
```

[12]:

```
caudate_pgc2.loc[(caudate_pgc2["Type"] == "Gene"), ["gene_id", "variant_id"]].
 ↪groupby("gene_id").size()
```

[12]:

```
gene_id
ENSG00000182600.9     1
ENSG00000227262.3    26
ENSG00000244731.7     2
dtype: int64
```

[13]:

```
dlpfc_pgc2.loc[(dlpfc_pgc2["Type"] == "Gene"), ["gene_id", "variant_id"]].
 ↪groupby("gene_id").size()
```

[13]:

```
gene_id
ENSG00000182600.9     1
ENSG00000227262.3    26
ENSG00000244731.7     2
dtype: int64
```

[14]:

```
hippo_pgc2.loc[(hippo_pgc2["Type"] == "Gene"), ["gene_id", "variant_id"]].
 ↪groupby("gene_id").size()
```

[14]:

```
gene_id
ENSG00000182600.9     1
ENSG00000227262.3    26
ENSG00000244731.7     2
dtype: int64
```

[ ]:

```

```