This common genes python script takes in Genelists for two individual traits files and compares them to find the number of common genes and the gene names. By utilizing the bash script to automate this process for all traits, the outputs are stored in an output txt file.

Starting with

import sys, os

import pandas as pd

inFile1=sys.argv[1]

inFile2=sys.argv[2]

THIS\_FOLDER = os.path.dirname(os.path.abspath(\_\_file\_\_))

inFile1t= os.path.join(THIS\_FOLDER, inFile1)

inFile2t= os.path.join(THIS\_FOLDER, inFile2)

Sys.argv[1] and Sys.argv[2] are connected with the bash script which can automate the process of all traits by replacing sys.argv[1] and sys.argv[2] with the two gene-list files.

with open(inFile1t,"r") as f:

lista = f.readlines()

lista = [x.strip() for x in lista]

splitlist1=[]

for i in lista:

splitlist1 += i.split(" ")

with open(inFile2t,"r") as f:

listb = f.readlines()

listb = [x.strip() for x in listb]

splitlist2=[]

for i in listb:

splitlist2 += i.split(" ")

Opens the two gene-list files and sorts the genes into two lists for processing

nodup1 = []

for i in splitlist1:

if i not in nodup1:

nodup1.append(i)

nodup2 = []

for i in splitlist2:

if i not in nodup2:

nodup2.append(i)

Sorts through the two lists and get rid of any gene duplicates that will mess up the result

commons = [element for element in nodup1 if element in nodup2]

Finds the common genes in the two traits.

num\_commons = len(commons)

print(sys.argv[1] + "--" +sys.argv[2] + " common number: " + str(num\_commons))

print(commons)

Num\_commons prints the number of common genes, while ‘commons’ prints the gene names.