sgRNA counting pipeline I

First things first, sharing is an excellent way to tap everyone's potential by lowering the barriers.

Authors: **Haitao Wang**, **Jimmy Zeng**, **Lakhansing Pardeshi**

Step one: Software installation

Description:

cutadapt for cutting **barcode** or **adaptor** in .fastq or .fq file

mageck for counting **activtion/inhibition/knowout** sgRNA reads

```
# Before install cutadapt you should install Conda, Conda can help you install bio
software automatically, to make life easy.
# Conda installation
https://conda.io/docs/user-guide/install/macos.html#install-macos-silent

wget http://repo.continuum.io/miniconda/Miniconda3-3.7.0-Linux-x86_64.sh -O
~/miniconda.sh
bash ~/miniconda.sh -b -p $HOME/miniconda
export PATH="$HOME/miniconda/bin:$PATH"

conda config --add channels bioconda
# Cutadapt installation
conda install -c bioconda cutadapt
# MAGeCK installation
conda install -c bioconda mageck

# test whether installed or not
cutadapt -h # help info to check useage
mageck -h # help info to check useage
```

Step two: Demultipex

Description:

based on **index** or **barcodes** to deconstruct from a mix fastq file

`demultiplex.pl` this tool created by **Lakhansing Pardeshi from Chris lab**

Usage

```
perl demultiplex.pl --barcodes barcodes.txt --1 R1.fq --2 R2.fq --suffix <suffix to add>
```

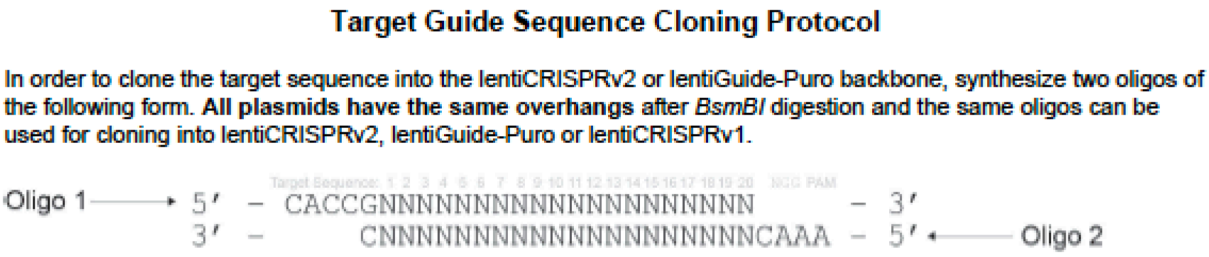
```
# prepare barcodes.txt look like this
TAAGTAGAG    HK112
ATACACGATC   HK113
GATCGCGCGGT  HK114
CGATCATGATCG     HK115
TCGATCGTTACCA    HK116
ATCGATTCCTTGGT   HK117
```

Step three trimme each fastq file for future sg counting

Description:

LentiCRISPRv2 All plasmids have the same overhangs after BsmBI digestion and the same oligos can be used for cloning into lentiCRISPRv2, lentiGuide-Puro or lentiCRISPRv1.

5' arm: ......GGACGAAACACCG **20bp-sg-sequence** GTTTTAGAGCTAG...... 3' arm



Target Guide Sequence Cloning Protocol

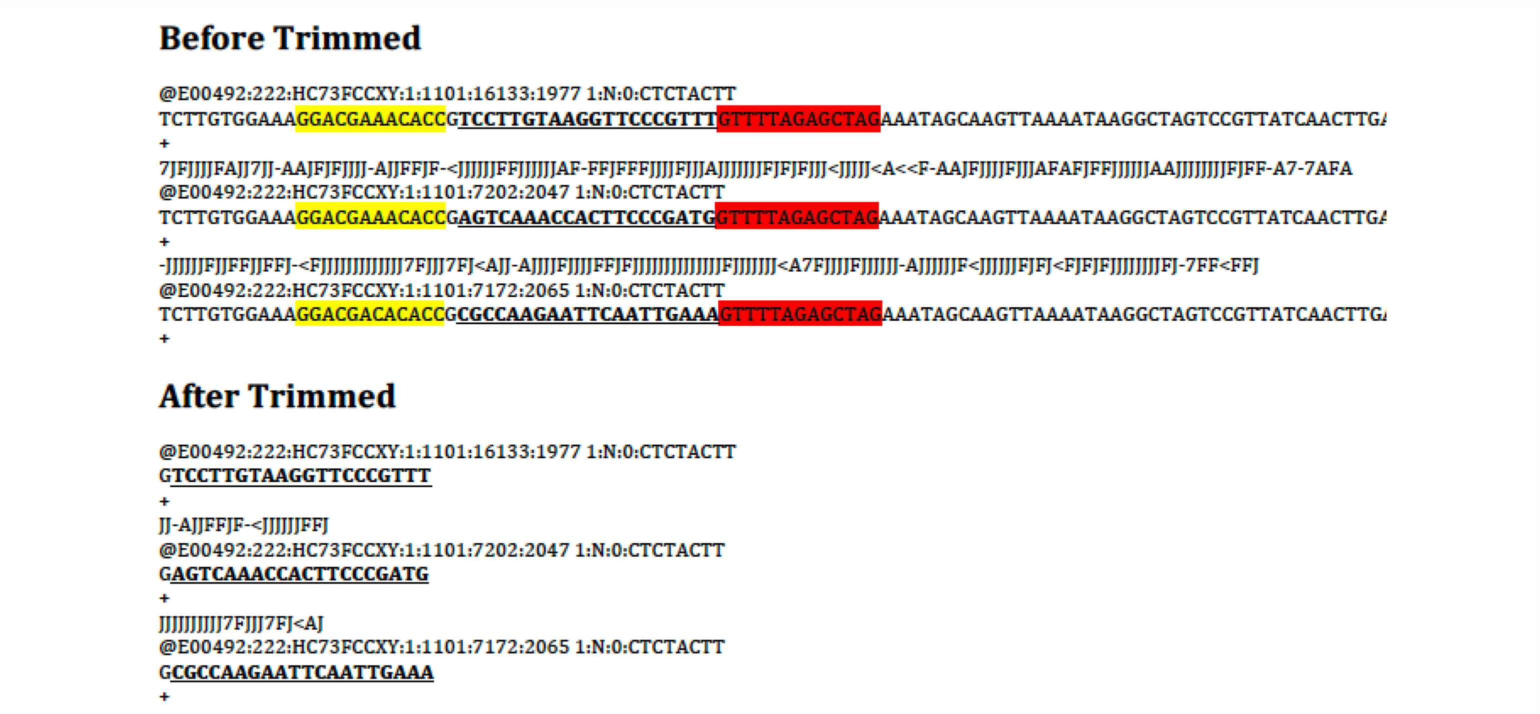
In order to clone the target sequence into the lentiCRISPRv2 or lentiGuide-Puro backbone, synthesize two oligos of the following form. **All plasmids have the same overhangs after** ***BsmBI*** **digestion and the same oligos can be used for cloning into lentiCRISPRv2, lentiGuide-Puro or lentiCRISPRv1.**

Target Sequence: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 NGG PAM
Oligo 1
5′ – CACCGNNNNNNNNNNNNNNNNNNNN – 3′
3′ – CNNNNNNNNNNNNNNNNNNNNCAAA – 5′
Oligo 2

```
# Check the CRISPR-Cas9 seq reads using 5' and 3' sequence beside sg sequence
grep --color=auto GGACGAAACACC CC1-F01_1_R1.fastq | head
grep --color=auto GTTTTAGAGCTAG CC1-F01_1_R1.fastq | head
```

Trimming seq CRISPR-Cas9 knockout screening (**usually we only use R1 in sg sequencing data**)

(compare R1 and R2, R1 have more target info than R2.)



Before Trimmed

```
@E00492:222:HC73FCCXY:1:1101:16133:1977 1:N:0:CTCTACTT
TCTTGTGGAAAGGACGAAACACCGTCCTTGTAAGGTTCCCGTTTGTTTTAGAGCTAGAAATAGCAAGTTAAAATAAGGCTAGTCCGTTATCAACTTGA
+
7JFJJJJFAJJ7JJ-AAJFJFJJJJ-AJJFFJF-<JJJJJJFFJJJJJJAF-FFJFFFJJJJFJJJAJJJJJJJFJFJFJJJ<JJJJJ<A<<F-AAJFJJJJFJJJAFAFJFFJJJJJJAAJJJJJJJJFJFF-A7-7AFA
@E00492:222:HC73FCCXY:1:1101:7202:2047 1:N:0:CTCTACTT
TCTTGTGGAAAGGACGAAACACCGAGTCAAACCACTTCCCGATGGTTTTAGAGCTAGAAATAGCAAGTTAAAATAAGGCTAGTCCGTTATCAACTTGA
+
-JJJJJFJJFFJJFFJ-<FJJJJJJJJJJJJ7FJJJ7FJ<AJJ-AJJJJFJJJJFFJFJJJJJJJJJJJJJJFJJJJJJJ<A7FJJJJFJJJJJJ-AJJJJJJF<JJJJJJFJFJ<FJFJFJJJJJJJJJFJ-7FF<FFJ
@E00492:222:HC73FCCXY:1:1101:7172:2065 1:N:0:CTCTACTT
TCTTGTGGAAAGGACGACACACCGCGCCAAGAATTCAATTGAAAGTTTTAGAGCTAGAAATAGCAAGTTAAAATAAGGCTAGTCCGTTATCAACTTGA
+
```

After Trimmed

```
@E00492:222:HC73FCCXY:1:1101:16133:1977 1:N:0:CTCTACTT
GTCCTTGTAAGGTTCCCGTTT
+
JJ-AJJFFJF-<JJJJJJFFJ
@E00492:222:HC73FCCXY:1:1101:7202:2047 1:N:0:CTCTACTT
GAGTCAAACCACTTCCCGATG
+
JJJJJJJJJJ7FJJJ7FJ<AJ
@E00492:222:HC73FCCXY:1:1101:7172:2065 1:N:0:CTCTACTT
GCGCCAAGAATTCAATTGAAA
+
# Trimming seq CRISPR-Cas9 knockout screening
### CRISPR-Cas9 knockout system (All plasmids have the same overhangs after BsmBI
digestion and the same oligos can be used for cloning into lentiCRISPRv2,
lentiGuide-Puro or lentiCRISPRv1)

### fq.R1 TCTTGTGGAAAGGACGAAACACCG - xxxxxxxxxxxxxx -
GTTTTAGAGCTAGAAATAGCAAGTTAAAATAAGGCTAGTCCGTTATCAACTTGAAAAAGTGGCACCGAGTCGGTGCTTTTTT
GAATTCGCTAGCTAG

### fq.R2
CTAGCTAGCGAATTCAAAAAAGCACCGACTCGGTGCCACTTTTTCAAGTTGATAACGGACTAGCCTTATTTTAACTTGCTAT
TTCTAGCTCTAAAAC - xxxxxxxxxxxxxx - CGGTGTTTCGTCCTTTCCACAAGA

# cutadapt to cut and trimme the data from 5'(only R1)
cd /PATH/
cutadapt -f fastq -q 10 \
-g GGACGAAACACC \
-o sample_1_trimmed.fastq.gz \
/PATH/sample_R1.fastq # only R1

# cutadapt to cut and trimme the data from 3' (only R1)
cutadapt -f fastq -q 10 \
-a GTTTTAGAGCTAG \
-o /PATH/sample_2_trimmed.fastq.gz \
/PATH/sample_1_trimmed.fastq.gz

cat /PATH/sample_2_trimmed.fastq.gz > /PATH/sample_trimmed.fastq.gz
rm /PATH/sample_1_trimmed.fastq.gz
rm /PATH/sample_2_trimmed.fastq.gz
```

Step four: sg counting

Method I:

1. Raw Count sgRNA

`fastqgz_to_counts.py` this tool created by mhorlbeck

Usage (note: under python2.7; have to put required input files order like: 1. library; 2. output-path, 3. fastq files)

`python fastqgz_to_counts.py --trim_start 1 --trim_end 21 <library.fasta> <output/path> <all/sg.fastq.gz>`

library

```
>0610007P14Rik_TCCTGAATGTGTTACGAAGC
tcctgaatgtgttacgaagc
>0610007P14Rik_GGTCGGGCTCCGGTACCTAG
ggtcgggctccggtacctag
>0610007P14Rik_GCCAGCTTCGTAACACATTC
gccagcttcgtaacacattc
>0610009B22Rik_TCATCATGCTGCATGACGTG
tcatcatgctgcatgacgtg
>0610009B22Rik_ATTCAATGAGTGGTTCGTCT
attcaatgagtggttcgtct
>0610009B22Rik_TCGTCACGGCTGGGCACATG
tcgtcacggctgggcacatg
>0610009D07Rik_TACACTCTGATTTGACGAAT
tacactctgatttgacgaat
```

Run

```
# count sgRNA python fastqgz_to_counts.py library-fasta output-path sg-fastq
python fastqgz_to_counts.py -p 16 \
--trim_start 1 --trim_end 21 \
GeckoV2_MGLib_A.fasta \
/output \
*trimmed.fastq.gz
```

Output

```
# count file have two columns, symbol and count
0610009O20Rik_CTGTGCCAAGAGCGTTCAGC  0
0610009O20Rik_TGGGTTTGGGCGTTATCCCA  0
0610010B08Rik_CGTGCATGTGAACTTCACTC  22
0610010B08Rik_GACTTCTAGAAGTTTGAAAA  6
0610010B08Rik_TATAGCTGTGAGATTCTTAT  0
```

2. Raw Count merge to table

`merged_table.pl` this tool created by Jimmy and haitao

Usage (note: can only merge files which include same row names and numbers)

`perl merged_table.pl <file1> <file2> <file3>...`

```
# merge selected files
perl merged_table.pl <file1> <file2> <file3>
# or All files
perl merged_table.pl /output/*
```

Output

```
# count file
sgRNA   Gene    CC1-F01 CC1-F02 CC1-F03 CC1-F04 CC1-F05
MGLibA_57638    Usp50   3   2   245 15  4
MGLibA_2481 Ablim1  10  12  5   18  6
MGLibA_18840    Foxo6   14  24  24  98  32
MGLibA_20534    Gm13040 1   505 3   7   2
MGLibA_42555    Prame   1   0   2   5   0
MGLibA_10739    Clec3b  18  621 32  122 38
MGLibA_36174    Olfr127 9   16  11  61  23
MGLibA_58314    Vmn1r238    10  6   6   69  23
MGLibA_24769    Hp1bp3  2   5   5   19  3
```

3. Normalize counts table (by total counts - size factor)

```
# add one more column named 'gene' for annotation
# sgRNA Gene sample1 sample2 ...
mageck count \
-k merged.txt \
-n /output/merged \
--norm-method total
```

Output

```
# Normalized count file by size factor
sgRNA   Gene    CC1-F01 CC1-F02 CC1-F03 CC1-F04 CC1-F05
MGLibA_57638    Usp50   13.88454104 5.200697141 893.8208094 9.225439438
8.905259711
MGLibA_02481    Ablim1  46.28180348 31.20418285 18.24124101 11.07052733
13.35788957
MGLibA_18840    Foxo6   64.79452487 62.40836569 87.55795684 60.27287099
71.24207769
MGLibA_20534    Gm13040 4.628180348 1313.176028 10.9447446  4.305205071
4.452629855
MGLibA_42555    Prame   4.628180348 0    7.296496403 3.075146479 0
MGLibA_10739    Clec3b  83.30724627 1614.816462 116.7439425 75.0335741
84.59996725
MGLibA_36174    Olfr127 41.65362313 41.60557713 40.13073022 37.51678705
51.20524334
MGLibA_58314    Vmn1r238    46.28180348 15.60209142 21.88948921 42.43702141
51.20524334
MGLibA_24769    Hp1bp3  9.256360696 13.00174285 18.24124101 11.68555662
6.678944783
```

Method II:

Using mageck **one step** `count` function to count, merge and normalize data

Usage

`mageck count -l library.txt -n <output> --sample-label name1,name2,name3 –fastq 01_trimmed.fastq 02_trimmed.fastq 03_trimmed.fastq`

Library

```
#  MGLibA.txt mouse library A looks like:
MGLibA_00001    TCCTGAATGTGTTACGAAGC    0610007P14Rik
MGLibA_00002    GGTCGGGCTCCGGTACCTAG    0610007P14Rik
MGLibA_00003    GCCAGCTTCGTAACACATTC    0610007P14Rik
MGLibA_00004    TCATCATGCTGCATGACGTG    0610009B22Rik
MGLibA_00005    ATTCAATGAGTGGTTCGTCT    0610009B22Rik
```

Run

```
mageck count -l MGLibA.txt \
-n /Users/haitao/Desktop/sgRNA/fq \
--sample-label CC1-F01,CC1-F02,CC1-F03 \
--fastq CC1-F01_trimmed.fastq CC1-F02_trimmed.fastq CC1-F03_trimmed.fastq
```

Output

```
# Normalized count file by size factor
sgRNA    Gene     CC1-F01 CC1-F02 CC1-F03 CC1-F04 CC1-F05
MGLibA_57638     Usp50    13.88454104 5.200697141 893.8208094 9.225439438
8.905259711
MGLibA_02481     Ablim1   46.28180348 31.20418285 18.24124101 11.07052733
13.35788957
MGLibA_18840     Foxo6    64.79452487 62.40836569 87.55795684 60.27287099
71.24207769
MGLibA_20534     Gm13040 4.628180348 1313.176028 10.9447446   4.305205071
4.452629855
MGLibA_42555     Prame    4.628180348 0    7.296496403 3.075146479 0
MGLibA_10739     Clec3b   83.30724627 1614.816462 116.7439425 75.0335741
84.59996725
MGLibA_36174     Olfr127 41.65362313 41.60557713 40.13073022 37.51678705
51.20524334
MGLibA_58314     Vmn1r238     46.28180348 15.60209142 21.88948921 42.43702141
51.20524334
MGLibA_24769     Hp1bp3   9.256360696 13.00174285 18.24124101 11.68555662
6.678944783
```

Step five: group sample comparison

Description:

Comparison between samples

MAGeCK has different commands:

`test` (if you already have count tables)

`count` (if you want to generate count tables from fastq files)

`run` (combine both test and count)

`pathway` (if you want to do the pathway test)

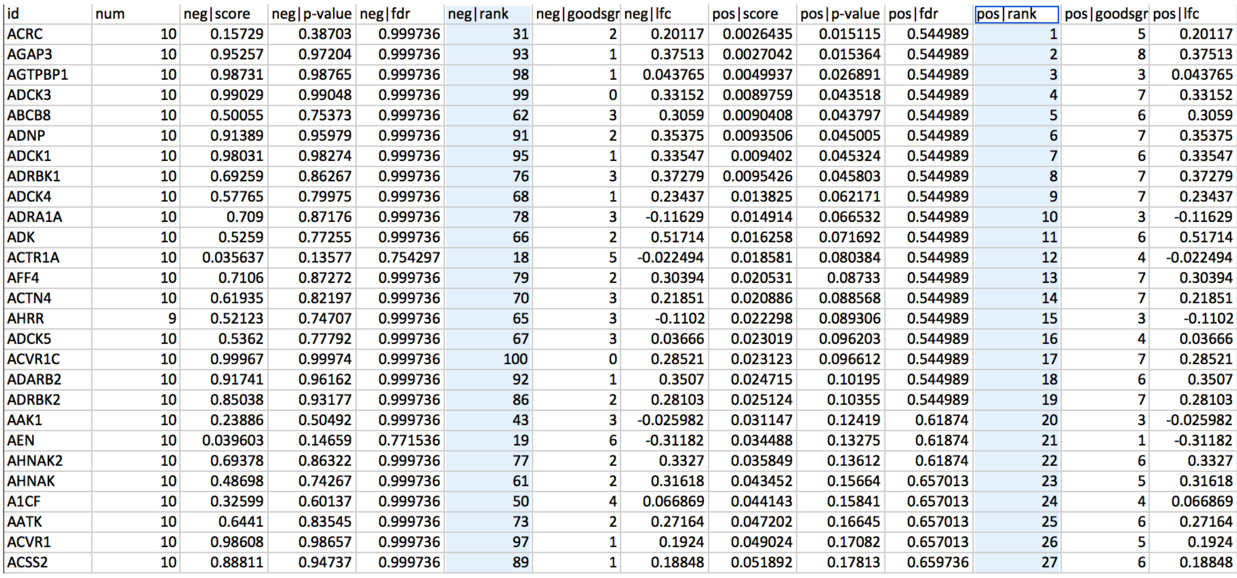
count file: sample.txt

```
sgRNA    Gene     initial1         initial2         final1   final2
A1CF_m52595977  A1CF     213      274      883      175
A1CF_m52596017  A1CF     294      412      1554     1891
AAAS_m53714382  AAAS     704      671      799      1426
AAAS_m53715169  AAAS     651      627      797      1690
AAAS_m53715176  AAAS     545      89       392      664
AAK1_m69870049  AAK1     364      465      693      2006
AATF_m35306444  AATF     449      456      1396     1402
AATF_m35306475  AATF     493      612      1102     537
```

Run

```
mageck test \
# Raw Count tables
-k sample.txt \
# Treatment sample labels
-t final1,final2 \
# Control sample labels
-c initial1,initial2 \
-n Output # Output labels
```

Output



id	num	neg|score	neg|p-value	neg|fdr	neg|rank	neg|goodsgr	neg|lfc	pos|score	pos|p-value	pos|fdr	pos|rank	pos|goodsgr	pos|lfc
ACRC	10	0.15729	0.38703	0.999736	31	2	0.20117	0.0026435	0.015115	0.544989	1	5	0.20117
AGAP3	10	0.95257	0.97204	0.999736	93	1	0.37513	0.0027042	0.015364	0.544989	2	8	0.37513
AGTPBP1	10	0.98731	0.98765	0.999736	98	1	0.043765	0.0049937	0.026891	0.544989	3	3	0.043765
ADCK3	10	0.99029	0.99048	0.999736	99	0	0.33152	0.0089759	0.043518	0.544989	4	7	0.33152
ABCB8	10	0.50055	0.75373	0.999736	62	3	0.3059	0.0090408	0.043797	0.544989	5	6	0.3059
ADNP	10	0.91389	0.95979	0.999736	91	2	0.35375	0.0093506	0.045005	0.544989	6	7	0.35375
ADCK1	10	0.98031	0.98274	0.999736	95	1	0.33547	0.009402	0.045324	0.544989	7	6	0.33547
ADRBK1	10	0.69259	0.86267	0.999736	76	3	0.37279	0.0095426	0.045803	0.544989	8	7	0.37279
ADCK4	10	0.57765	0.79975	0.999736	68	1	0.23437	0.013825	0.062171	0.544989	9	7	0.23437
ADRA1A	10	0.709	0.87176	0.999736	78	3	-0.11629	0.014914	0.066532	0.544989	10	3	-0.11629
ADK	10	0.5259	0.77255	0.999736	66	2	0.51714	0.016258	0.071692	0.544989	11	6	0.51714
ACTR1A	10	0.035637	0.13577	0.754297	18	5	-0.022494	0.018581	0.080384	0.544989	12	4	-0.022494
AFF4	10	0.7106	0.87272	0.999736	79	2	0.30394	0.020531	0.08733	0.544989	13	7	0.30394
ACTN4	10	0.61935	0.82197	0.999736	70	3	0.21851	0.020886	0.088568	0.544989	14	7	0.21851
AHRR	9	0.52123	0.74707	0.999736	65	3	-0.1102	0.022298	0.089306	0.544989	15	3	-0.1102
ADCK5	10	0.5362	0.77792	0.999736	67	3	0.03666	0.023019	0.096203	0.544989	16	4	0.03666
ACVR1C	10	0.99967	0.99974	0.999736	100	0	0.28521	0.023123	0.096612	0.544989	17	7	0.28521
ADARB2	10	0.91741	0.96162	0.999736	92	1	0.3507	0.024715	0.10195	0.544989	18	6	0.3507
ADRBK2	10	0.85038	0.93177	0.999736	86	2	0.28103	0.025124	0.10355	0.544989	19	7	0.28103
AAK1	10	0.23886	0.50492	0.999736	43	3	-0.025982	0.031147	0.12419	0.61874	20	3	-0.025982
AEN	10	0.039603	0.14659	0.771536	19	6	-0.31182	0.034488	0.13275	0.61874	21	1	-0.31182
AHNAK2	10	0.69378	0.86322	0.999736	77	2	0.3327	0.035849	0.13612	0.61874	22	6	0.3327
AHNAK	10	0.48698	0.74267	0.999736	61	2	0.31618	0.043452	0.15664	0.657013	23	5	0.31618
A1CF	10	0.32599	0.60137	0.999736	50	4	0.066869	0.044143	0.15841	0.657013	24	4	0.066869
AATK	10	0.6441	0.83545	0.999736	73	2	0.27164	0.047202	0.16645	0.657013	25	6	0.27164
ACVR1	10	0.98608	0.98657	0.999736	97	1	0.1924	0.049024	0.17082	0.657013	26	5	0.1924
ACSS2	10	0.88811	0.94737	0.999736	89	1	0.18848	0.051892	0.17813	0.659736	27	6	0.18848

Step Six**: sg downstream analysis

coming soon