Biological Sequence Analysis

Final Report

23.01.2022 Osman Batuhan Şahin – 170709003

ABOUT RNA SEQUENCE

Beta cells are cells that make insulin, a hormone that controls the level of glucose (a type of sugar) in the blood. Type 1 diabetes disease occurs when humans immune system destroys these cells mistakenly. However, type 2 diabetes was handling in this study. Saturated fatty acids cause beta-cell failure and causes type 2 diabetes if the individual has a genetic predisposition. In this study, the researchers used RNA sequencing to map the transcripts expressed for five palmitate-treated human islets. For type 2 diabetes analysis, the application of palmitate treatment to beta cells was followed for 48 hours. https://teaching.healthtech.dtu.dk/teaching/images/6/6d/1978.full.pdf.

WHY THIS PROJECT?

Diabetes, one of the most common chronic diseases of our time, is increasing day by day with the changing life habits. Although type 1 diabetes occurs at an early age due to genetic disorders, type 2 diabetes occurs due to reasons such as malnutrition and inactivity. Type 2 diabetes can be prevented with a conscious and healthy diet. In this project, I wanted to analyze the effect of fatty acids on beta cells that secrete insulin. Thus, I will be able to observe the effect of a bad diet on type 2 diabetes.

DATA

https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE53949 is url of my data. It has 10 samples and total size is 8.9 gb. For download the data, i used sra toolkit. I used sra toolkit with “prefix” command and list of samples like “SRR1105566”. Original file format was sra. I converted datas from sra format to fastq format with “fastq-dump –split-files” command. I used –split-files command because at the first convert, fastqc quality scores were bad at the 50th position. Once i change the format of the files, they become human readable. Here is first entry ıf my SRR1105566_1.fastq file:

```
@SRR1105566.1 1 length=51

CTTCTGCTGACAATTTCCAACNGGCATGAGATTGCTGAGCTGANGTGGACT

+SRR1105566.1 1 length=51

B7CCCBCC@BCBBBB7ACBC=%9@BBBA=ABAA?BB?BABB<;%<<A;>@B
```

First line is a sequence identifier with information about the sequencing run and the cluster. Second line is the sequence. Third line is a separator. Fourth line calls quality scores.

When I changed the format of the files, their size increased about 4 times. This came as a surprise to me as I am working in a virtual machine, but I solved the problem by increasing the space I allocated to the disk.

FASTQC – MULTIQC

I applied “fasqc” command to all fastq files to analysis. This tool used to provide an overview of basic quality control metrics for sequencing data. I installed this tool to my global environment with “sudo apt-get install fastqc” command.

MultiQC is a reporting tool that parses summary statistics from results and log files generated by fastqc and log files. I installed this tool with pip. Then i get a HTML formatted multiqc report from all fastqc reports before data cleaning.

CLEANING

I had to trim the data because the data had adapter contents and quality scores were not good enough. For trimming i used flexbar tool. I used this command to clean data. flexbar -q TAIL -qf i1.8 --adapters trimrefs/illumina_multiplex.fa --threads 4 --zip-output GZ --reads srrdata/sra/SRR1105572_1.fastq --reads2 srrdata/sra/SRR1105572_2.fastq --target trim2/SRR1105572. I used Quality-based trimming for getting better quality scores. I gave adapter sequences in a .fa formatted file with ” --adapters” command. My adapters sequences are:

>Multiplexing_Read_1_Sequencing_Primer_3_to_5

AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT

>Multiplexing_Read_2_Sequencing_Primer_3_to_5

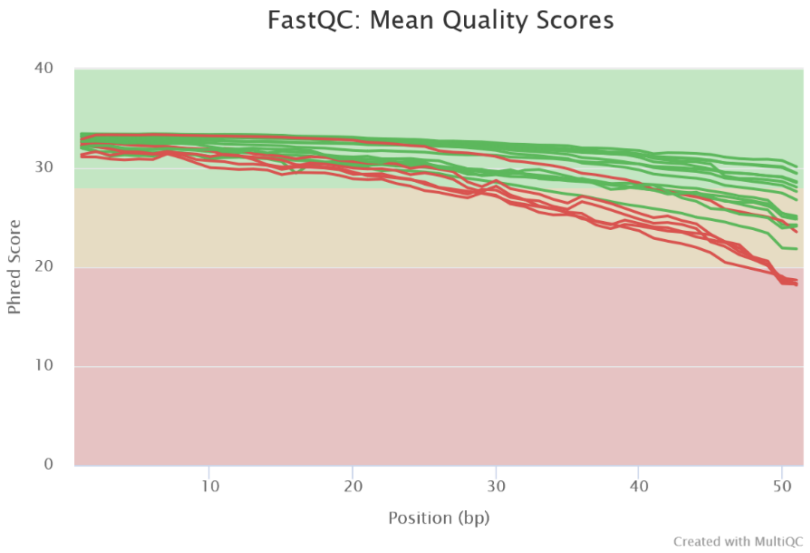
AGATCGGAAGAGCACACGTCTGAACTCCAGTCAC

>Illumina Small RNA Adapter 2

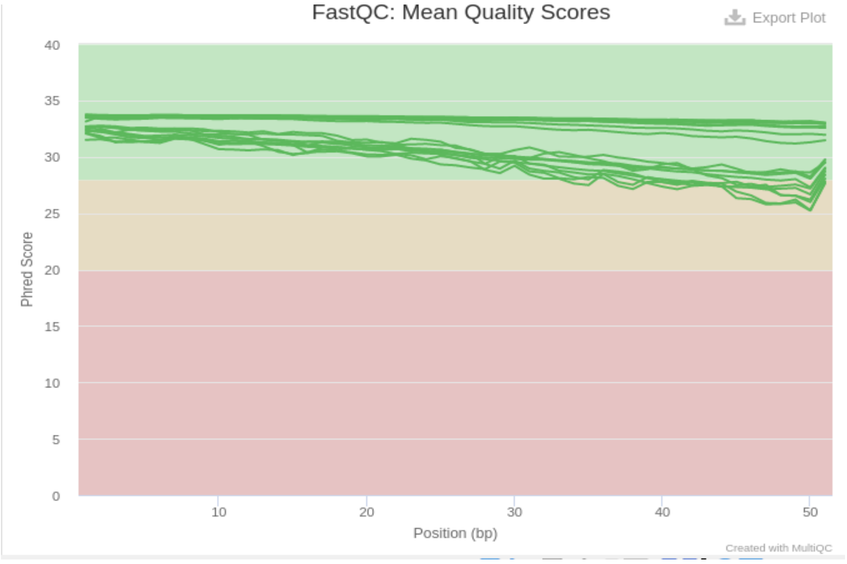
TCGTATGCCGTCTTCTGCTTGT

Per Base Sequence Quality:

Before:

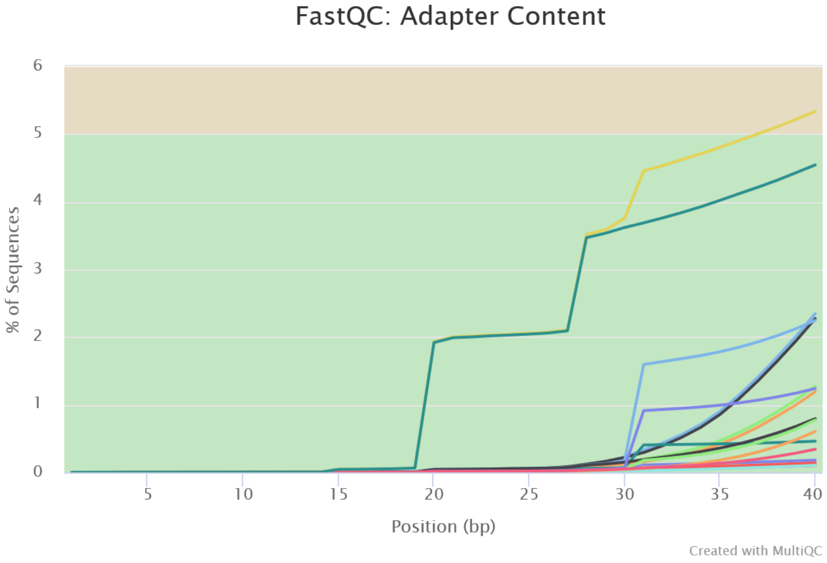


After:

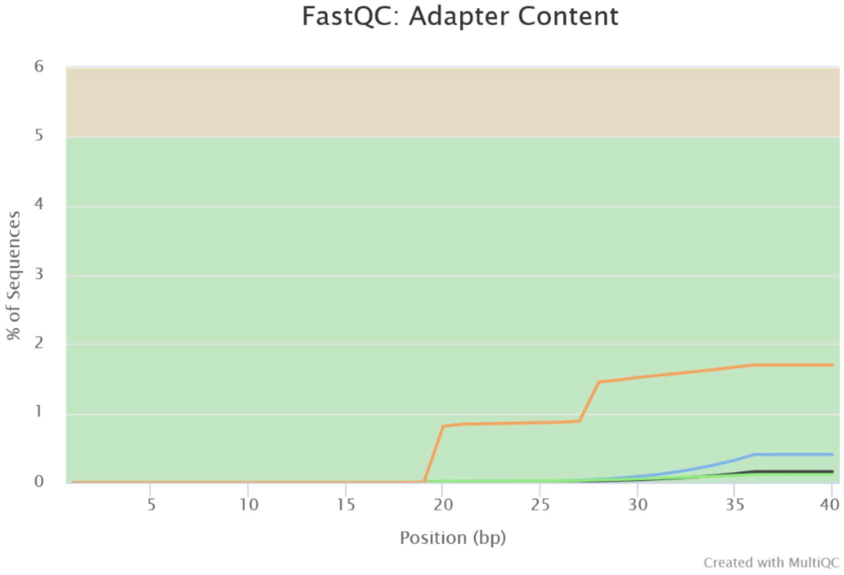


Adapter Content:

Before:



After:



Alignment

HISAT2

HISAT2 is an alignment program for mapping sequencing reads. With this tool, i mapped rna sequences with human genome. Here is a hisat2 command for one of my palmitate reads.

```
“hisat2 -p 8 --rg-id=SRR1105571 --rg SM:PALMITATE --rg LB:SRR1105571-PALMITATE --summary-file SRR1105571.out --rg PL:ILLUMINA -x refs/genome_snp_tran --dta --rna-strandness RF -1 trim2/SRR1105571_1.fastq.gz -2 trim2/SRR1105571_2.fastq.gz -S align/SRR1105571.sam”
```

I gave path to hisat2 index with -x option. I downloaded ENSEMBL indexed hg38 human genome and human genome gtf files with ”wget --content-disposition https://cloud.biohpc.swmed.edu/index.php/s/grch38_snp_tran/download” command.

Here is one of my hisat2 outputs:

```
6607142 reads; of these:
  6607142 (100.00%) were paired; of these:
    474976 (7.19%) aligned concordantly 0 times
    5559599 (84.15%) aligned concordantly exactly 1 time
    572567 (8.67%) aligned concordantly >1 times
    ----
    474976 pairs aligned concordantly 0 times; of these:
      31126 (6.55%) aligned discordantly 1 time
    ----
    443850 pairs aligned 0 times concordantly or discordantly; of these:
      887700 mates make up the pairs; of these:
        516873 (58.23%) aligned 0 times
        310939 (35.03%) aligned exactly 1 time
        59888 (6.75%) aligned >1 times
96.09% overall alignment rate
```

Aligned concordantly exactly 1 time line told us quality of alignment. It should be bigger than 70%. This percentage is around 85 on all of my alignments.

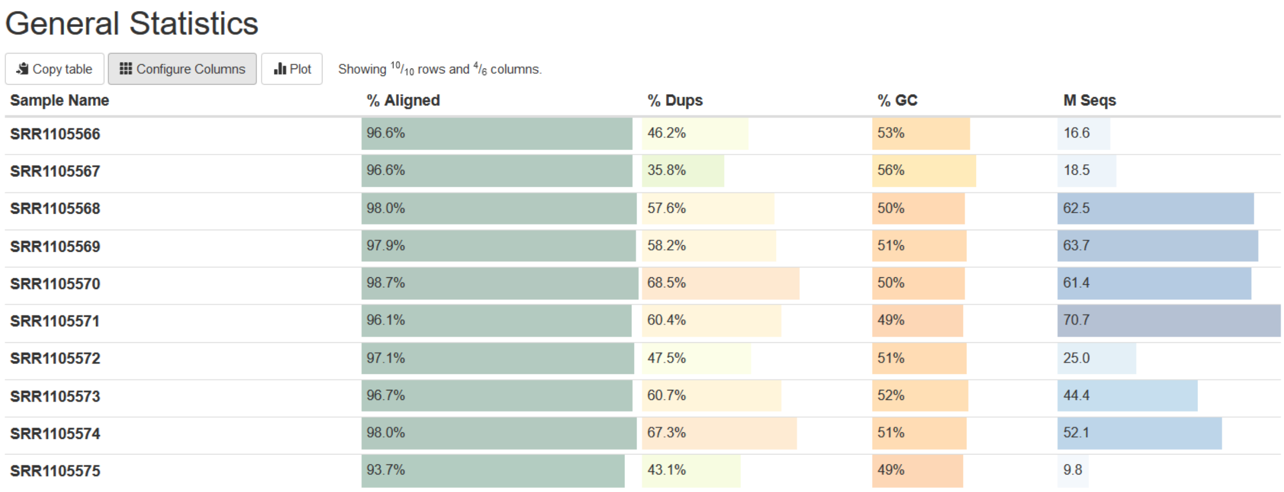
For sam to bam convertion and sorting by aligned position ”samtools sort -@ 8 -o SRR1105566.bam SRR1105566.sam” command used to each sam file. BAM files are smaller and more efficient for software to work with than SAM files, saving time and reducing costs of computation and storage.

For merging “control” bam files to a single file, i used picard tool.

```
java -Xmx2g -jar $RNA_HOME/student_tools/picard.jar MergeSamFiles OUTPUT=control.bam INPUT=SRR1105566.bam INPUT=SRR1105568.bam INPUT=SRR1105570.bam INPUT=SRR1105572.bam INPUT=SRR1105574.bam
```

I also used this tool to create a single “palmitate” file. At the and i had 2 bam files: “control.bam” and “palmitate.bam”

I ran FastQC on my bam files and multiqc on FastQC reports. Result is below.



General Statistics

Showing 10/10 rows and 4/6 columns.

Sample Name	% Aligned	% Dups	% GC	M Seqs
SRR1105566	96.6%	46.2%	53%	16.6
SRR1105567	96.6%	35.8%	56%	18.5
SRR1105568	98.0%	57.6%	50%	62.5
SRR1105569	97.9%	58.2%	51%	63.7
SRR1105570	98.7%	68.5%	50%	61.4
SRR1105571	96.1%	60.4%	49%	70.7
SRR1105572	97.1%	47.5%	51%	25.0
SRR1105573	96.7%	60.7%	52%	44.4
SRR1105574	98.0%	67.3%	51%	52.1
SRR1105575	93.7%	43.1%	49%	9.8

Samtools

I used “samtools flagstat control.bam” to get a basic summary of control sequence alignment. Output is below.

```
217585138 + 0 in total (QC-passed reads + QC-failed reads)

26531524 + 0 secondary

0 + 0 supplementary

0 + 0 duplicates

213737055 + 0 mapped (98.23% : N/A)

191053614 + 0 paired in sequencing

95526807 + 0 read1

95526807 + 0 read2

183026382 + 0 properly paired (95.80% : N/A)

184374410 + 0 with itself and mate mapped

2831121 + 0 singletons (1.48% : N/A)

579996 + 0 with mate mapped to a different chr

428366 + 0 with mate mapped to a different chr (mapQ>=5)
```

I used “samtools flagstat palmitate.bam” to get a basic summary of palmitate sequence alignment. Output is below.

```
207149089 + 0 in total (QC-passed reads + QC-failed reads)
```

```
30586231 + 0 secondary
0 + 0 supplementary
0 + 0 duplicates
201403711 + 0 mapped (97.23% : N/A)
176562858 + 0 paired in sequencing
88281429 + 0 read1
88281429 + 0 read2
163941890 + 0 properly paired (92.85% : N/A)
166490564 + 0 with itself and mate mapped
4326916 + 0 singletons (2.45% : N/A)
1463572 + 0 with mate mapped to a different chr
1225590 + 0 with mate mapped to a different chr (mapQ>=5)
```

Expression

Stringtie

I used stringtie to generate expression estimates from the RNA-Seq alignments generated by HISAT2 in the previous module. Output transcripts is a gtf file. I used stringtie with the command below.

```
stringtie -p 8 -G ../../../refs/Homo_sapiens.GRCh38.99.gtf -e -B -o SRR1105566/transcripts.gtf -A SRR1105566/gene_abundances.tsv ../../../align/SRR1105566.bam
```

I created tidy expression matrix files for the StringTie results. With --expression_metric option expression measures changed as coverage, FPKM (Fragments Per Kilobase Million), and TPM (Transcripts Per Kilobase Million) for gene and transcript level.

```
./stringtie_expression_matrix.pl --expression_metric=Coverage --result_dirs='SRR1105566,SRR1105567,SRR1105568,SRR1105569,SRR1105570,SRR1105571,SRR1105572,SRR1105573,SRR1105574,SRR1105575' --transcript_matrix_file=transcript_coverage_all_samples.tsv --gene_matrix_file=gene_coverage_all_samples.tsv
```

Here is a row from gene_tpm_all_samples.tsv file:

```
Gene_ID	SRR1105566	SRR1105567	SRR1105568	SRR1105569	SRR1105570	SRR1105571	SRR1105572	SRR1105573	SRR1105574	SRR1105575

ENSG00000000003	5.707782	5.069633	26.946156	22.382566	10.630384	10.464552	11.543087	3.830255	28.493668	8.769938
```

Here is a row from gene_fpkm_all_samples.tsv file:

```
Gene_ID        SRR1105566   SRR1105567   SRR1105568   SRR1105569   SRR1105570
         SRR1105571   SRR1105572   SRR1105573   SRR1105574   SRR1105575

ENSG00000000003   2.941525      2.628471      13.866009      11.494489      5.975628
         5.454505      5.786224      2.038644      14.692317      4.039905
```

Htseq-count

HTSeq is a Python package for analysis of high-throughput sequencing data. Given a file with aligned sequencing reads and a list of genomic features, htseq-count counts how many reads map to each feature. I ran htseq-count on alignments instead to produce raw counts instead of FPKM/TPM values for differential expression analysis. Htseq-count command for SRR1105566 is below.

```
htseq-count --format bam --order pos --mode intersection-strict --stranded reverse --
minaqual 1 --type exon --idattr gene_id ../../align/SRR1105566.bam
../../refs/Homo_sapiens.GRCh38.99.gtf > SRR1105566_gene.tsv
```

Default format is sam, i changed it to bam for using bam files. Previously i generated position sorted BAM files so i used 'pos' for '--order'. My data is stranded so i used –stranded reverse. I downloaded gtf file from http://ftp.ensembl.org/pub/release-99/gtf/homo_sapiens/Homo_sapiens.GRCh38.99.gtf.gz.

I merged results files into a single matrix. The following joins the results for each replicate together, adds a header, reformats the result as a tsv file.

```
echo "GeneID SRR1105566 SRR1105567 SRR1105568 SRR1105569 SRR1105570 SRR1105571
SRR1105572 SRR1105573 SRR1105574 SRR1105575" > header.txt

cat header.txt gene_read_counts_table_all.tsv | grep -v "_" | perl -ne 'chomp $; $_ =~
s/\s+/\t/g; print "$_\n"' > gene_read_counts_table_all_final.tsv

rm -f gene_read_counts_table_all.tsv header.txt
```

3 rows of tsv file are below.

```
ENSG00000000419   48      47      145     159     182     302     75      101     129     38

ENSG00000000457   51      44      223     220     165     203     75      117     149     25

ENSG00000000460   32      10      73      67      73      90      34      53      57      5
```

Differantial Expression

Ballgown DE Analysis

First i created a csv file with my 10 expression file name with ids, type and path columns. Then i started R code. I created a ballgown instance with 227818 transcripts and 10 samples. In this R code i created 4 tsv files with performing differential expression (DE) analysis. First two is without filtering transcript and gene results, other two is filtered transcript and gene results. Difference of my code from tutorial code is at row 55 and 56.

```
sig_transcripts = subset(results_transcripts,
results_transcripts$pval<0.05&results_transcripts$fc>1.5)
```

```
sig_genes = subset(results_genes, results_genes$pval<0.05&results_genes$fc>1.5)
```

With the code above i gave threshold for fold change bigger than 1.5. I did this for getting more significant genes. After reading this article, I decided on thresholds: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2654802/.

Here is top 20 genes sorted by p value with fc>1.5 threshold with the following column name order “gene id, feature, fc, p value, q value, gene_name”.

```
grep -v feature control_vs_palmitate_gene_results_sig.tsv | sort -k 4 | head -n 20
```

gene id	feature	fc	p value	q value	gene_name
ENSG00000198431	gene	1.814966770226	0.0002785718438100	0.9911893088703	TXNRD1
ENSG00000136244	gene	4.961289746239	0.0006801239964603	0.9911893088703	IL6
ENSG00000110090	gene	1.696156744485	0.0008358710099133	0.9911893088703	CPT1A
ENSG00000170345	gene	1.730196365263	0.0008630102641956	0.9911893088703	FOS
ENSG00000151726	gene	1.739518333235	0.0009241538100712	0.9911893088703	ACSL1
ENSG00000001084	gene	1.930508406247	0.0010598969505082	0.9911893088703	GCLC
ENSG00000151012	gene	2.334912435695	0.0010832061537988	0.9911893088703	SLC7A11
ENSG00000235899	gene	4.252019180605	0.0011630738380532	0.9911893088703	LINC01564
ENSG00000109321	gene	4.653830726109	0.0014333335336698	0.9995565730675	AREG
ENSG00000197279	gene	1.637989106056	0.0019461009583068	0.9995565730675	ZNF165
ENSG00000164038	gene	1.650919886095	0.0023834418788982	0.9995565730675	SLC9B2
ENSG00000130164	gene	1.923184772377	0.0024964250147260	0.9995565730675	LDLR
ENSG00000169429	gene	2.448638532707	0.0025130005508116	0.9995565730675	CXCL8
ENSG00000073756	gene	4.312707764786	0.0030235785734603	0.9995565730675	PTGS2
ENSG00000162772	gene	2.347403547334	0.0034588290866266	0.9995565730675	ATF3
ENSG00000095794	gene	1.830527618953	0.0045729542390642	0.9995565730675	CREM
ENSG00000270299	gene	4.301509699933	0.0049176994861062	0.9995565730675	AL121758.1
ENSG00000141526	gene	1.636865249945	0.0052033951802258	0.9995565730675	SLC16A3
ENSG00000113739	gene	1.794174294766	0.0054325352033646	0.9995565730675	STC2
ENSG00000167772	gene	4.317703930553	0.0060131644862807	0.9995565730675	ANGPTL4

EdgeR Analysis

In this part, first i create a mapping file to go from ENSG IDs (which htseq-count output) to Symbols:

```
perl -ne 'if ($_ =~ /gene_id\s\"(ENSG\S+)\"\;/) { $id = $1; $name = undef; if ($_ =~
/gene_name\s\"(\S+)"\;/) { $name = $1; }; }; if ($id && $name) {print "$id\t$name\n";} if
($_=~/gene_id\s\"(ERCC\S+)\"/){print "$1\t$1\n";}' ../../refs/Homo_sapiens.GRCh38.99.gtf |
sort | uniq > ENSG_ID2Name.txt
```

While using edgeR library, i used rawdata that i generated above using htseq-count. With R code new datas require at least 25% of samples to have count > 25. After necessary codes below, dimention (dim(rawdata)) reduced from 60676 10 to 14878 10. I used estimate dispersion, TMM normalization and differential expression test. Number of up/down significant genes at FDR = 0.05 significance level:

```
> summary(de <- decideTestsDGE(et, p=.05))

       palmitate-control

Down              1

NotSig          14869

Up               8
```

DE_genes.txt table:

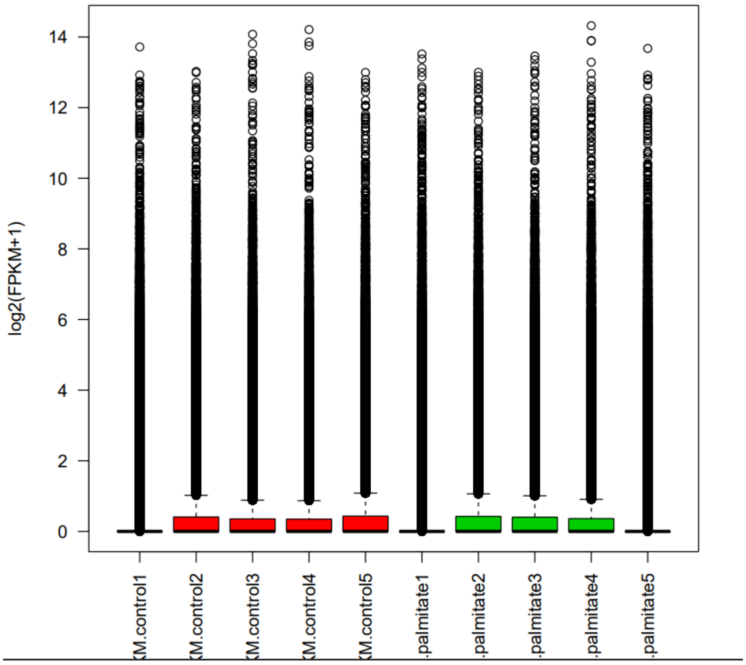
Gene	Gene_Name	Log10_Pvalue	Log_fold_change
ENSG00000167772	ANGPTL4	-6.273	2.023
ENSG00000109321	AREG	-7.176	2.453
ENSG00000136244	IL6	-7.803	2.545
ENSG00000147872	PLIN2	-15.093	2.491
ENSG00000162772	ATF3	-4.720	1.426
ENSG00000073756	PTGS2	-5.186	2.113
ENSG00000235899	LINC01564	-8.686	3.157
ENSG00000151012	SLC7A11	-7.432	1.781
ENSG00000261713	SSTR5-AS1	-4.927	-1.696

DE Visualization

1)With Ballgown

In this section, graphics are plotted with Ballgown package. https://github.com/griffithlab/rnaseq_tutorial/blob/master/scripts/Tutorial_Part2_ballgown.R this R codes has guided me. In this code i used most significant gene from differential expression part. I changed gene stable id's to entrezgene id with biomart tool of ensembl.org

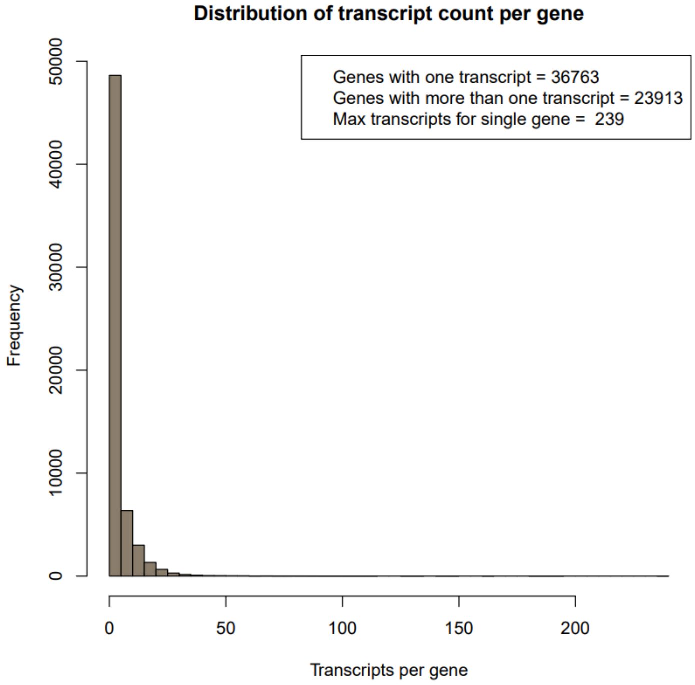
Boxplot of FPKM Values:



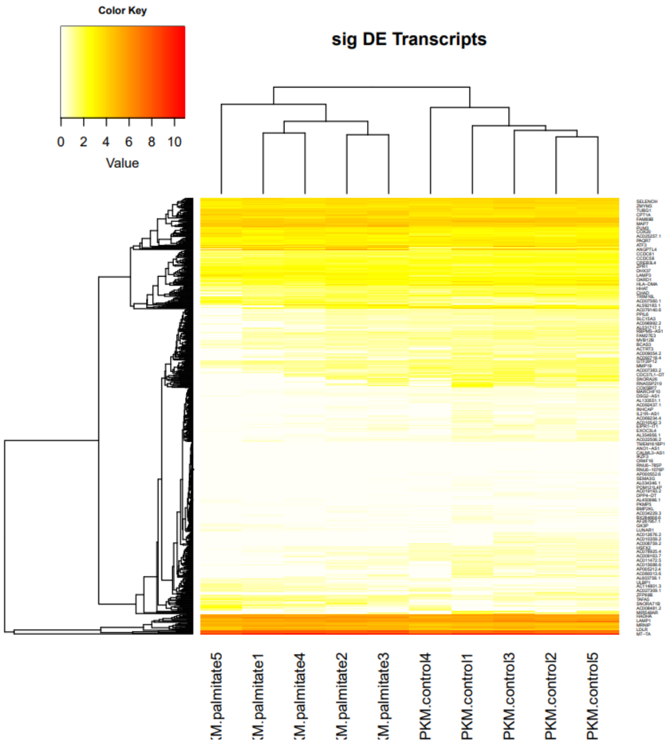
Other plots can be viewed from this link : https://drive.google.com/file/d/1WmjaXJmd1v2n7CJQc-BLWElmNAoPUe3f/view?usp=sharing.

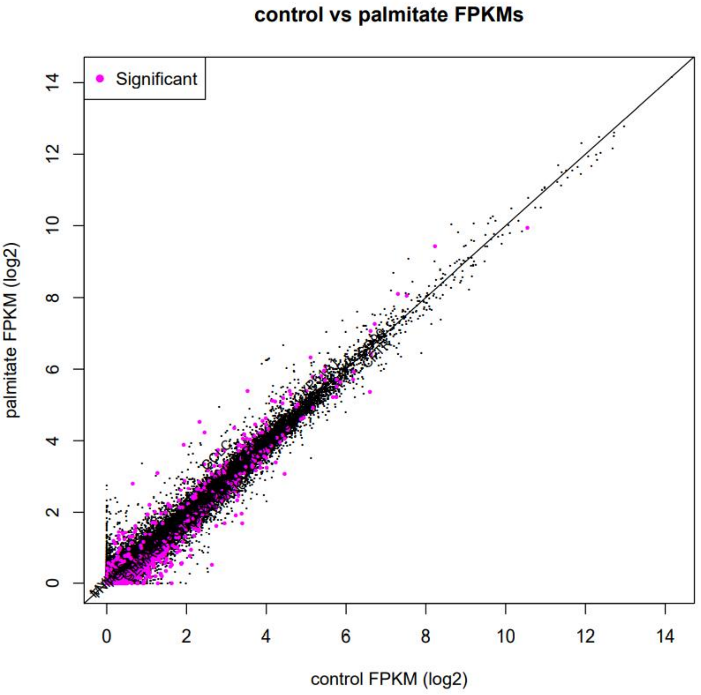
2) Without Ballgown

In this section, graphics are plotted without Ballgown package. https://github.com/griffithlab/rnaseq_tutorial/blob/master/scripts/Tutorial_Supplementary_R.R this R codes has guided me.



Heatmap shows the most significant DE transcripts:





Other plots can be viewed from this link: https://drive.google.com/file/d/10nOjOJlSiTJaidkT3rbIvPKJP27oJyvA/view?usp=sharing.

Variation

Rnaseqmut

To variant detection from BAM files, i used rnaseqmut. I downloaded rnaseqmut from https://github.com/davidliwei/rnaseqmut. For compilation and installation, i followed the steps in the link. Then i ran rnaseqmut on the bam files and produced the vcf files with rundemo.sh. At first, i did cleaning:

```
#echo "####### cleaning ##########"

#if [ ! -d results ]; then

#  mkdir results

#fi

#rm -rf results/.txt results/.vcf
```

Step 1 is de-novo mutation calling. $BAMFILELIST is BAM files produced in the previous steps:

```
#for file in $BAMFILELIST; do

 # filebase=`basename $file`

 # CMD="rnaseqmut $file > results/$filebase.1st.txt"

 # echo "#### COMMAND LINE: $CMD"

 # eval $CMD

#done
```

At Step 2, mutations merged in Step 1 into a candidate mutation list.

```
list  ##########"

CMD="merge1stfile results/*.1st.txt > results/ALLMUTLIST.txt"

echo "#### COMMAND LINE: $CMD"

eval $CMD
```

Step 3 is mutation calling from the merged lists.

```
for file in $BAMFILELIST; do

  filebase=`basename $file`

  CMD="rnaseqmut -l results/ALLMUTLIST.txt $file > results/$filebase.2nd.txt"

  echo "#### COMMAND LINE: $CMD"

  eval $CMD

done
```

Here are first lines of output from step 3 for one of my bam files.

```
BAM file:/home/batu/workspace/rnaseq/student_tools/align/SRR1105574.bam

Mut span:4
```

```
Mutation list:results/ALLMUTLIST.txt

Min read:1

Max mismatch:1

Reference genome:

Reading 9232554 lines, 102 chromosomes.

Switching to chromosome 1, length:248956422

Reading 9232554 lines, 1: 899158 records.
```

At step 4, merged the second pass of mutations into a big table. My labels are: “control1,palmitate1,control2,palmitate2,control3,palmitate3,control4,palmitate4,control5, palmitate5”

```
CMD="python3 ~/workspace/rnaseq/rnaseqmut/script/merge2ndvcf.py -l $LABELS results/*.2nd.txt > results/ALLMUT.txt"

echo "#### COMMAND LINE: $CMD"

eval $CMD
```

At step 5, mutations filtered based on user-defined parameters.

```
# defining the two normal samples as control groups

CONTROLGROUP="0,2,4,6,8"

# the following command keep mutations that occur in at least 1 non-control sample with at least 10 alternative read support.

# By default, filtermut.py will only keep mutations that occur in at least 1 non-control sample (-t option) with 20% frequency (-f) and 10 alternative read support (-d) , excluding those that also occur in control samples (-a) or does not have enough read coverage in control samples (-b)

CMD="python3 ~/workspace/rnaseq/rnaseqmut/script/filtermut.py -d 10 -f 0.0 -b 0 -c $CONTROLGROUP  -l $LABELS < results/ALLMUT.txt > results/ALLMUT_FILTERED.vcf"

echo "#### COMMAND LINE: $CMD"

eval $CMD
```

Output was:

```
1 1 14480

Number of samples:10

CONTROL group definition:0,2,4,6,8

TREATMENT group definition:1,3,5,7,9
```

```
100001 15 44715453

200001 20 44432751

300001 8 38757371

Found variations saved into nano/results/ALLMUT_FILTERED.vcf
```

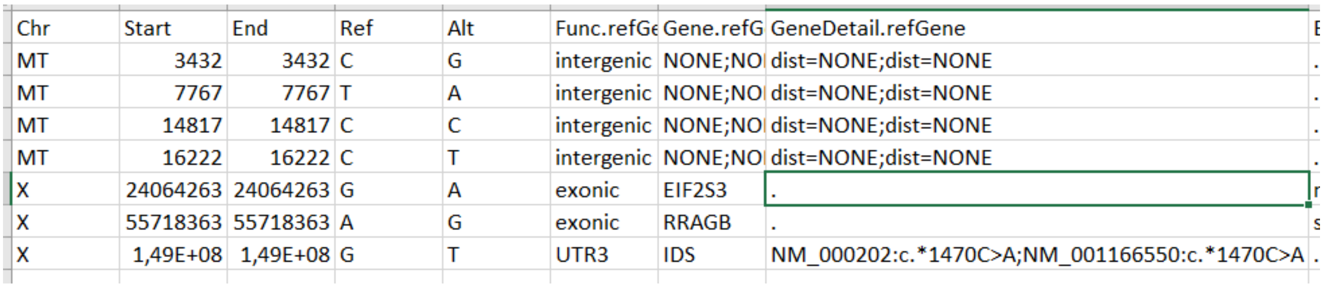
Annovar

Annovar is an efficient software tool to utilize update-to-date information to functionally annotate genetic variants detected from diverse genomes. For annovar, first i downloaded a few annotation databases into annovar directory.

```
perl annotate_variation.pl -buildver hg38 -downdb -webfrom annovar refGene humandb/

perl annotate_variation.pl -buildver hg38 -downdb cytoBand humandb/

perl annotate_variation.pl -buildver hg38 -downdb genomicSuperDups humandb/

perl annotate_variation.pl -buildver hg38 -downdb -webfrom annovar esp6500siv2_all humandb/

perl annotate_variation.pl -buildver hg38 -downdb -webfrom annovar 1000g2015aug humandb/

perl annotate_variation.pl -buildver hg38 -downdb -webfrom annovar exac03 humandb/

perl annotate_variation.pl -buildver hg38 -downdb -webfrom annovar avsnp150 humandb/

perl annotate_variation.pl -buildver hg38 -downdb -webfrom annovar dbnsfp30a humandb/

perl annotate_variation.pl -buildver hg38 -downdb -webfrom annovar clinvar_20200316 humandb/

perl annotate_variation.pl -buildver hg38 -downdb -webfrom annovar cosmic70 humandb/
```

Some of these databases downloaded as compressed file. We have to extract them for running. After the commands below, i get an tsv file as output.

```
perl table_annovar.pl  ../rnaseqmut/demo/results/ALLMUT_FILTERED.filtercoladded.vcf humandb/ -buildver hg38 -out myanno -remove -protocol refGene,cytoBand,genomicSuperDups,esp6500siv2_all,1000g2015aug_all,1000g2015aug_eur,exac03,avsnp150,dbnsfp30a,cosmic70,clinvar_20200316 -operation g,r,r,f,f,f,f,f,f,f,f -nastring . -vcfinput

cp myanno.hg38_multianno.txt myanno.hg38_multianno.tsv

libreoffice myanno.hg38_multianno.tsv
```



Chr	Start	End	Ref	Alt	Func.refGe	Gene.refG	GeneDetail.refGene	E
MT	3432	3432	C	G	intergenic	NONE;NO	dist=NONE;dist=NONE	.
MT	7767	7767	T	A	intergenic	NONE;NO	dist=NONE;dist=NONE	.
MT	14817	14817	C	C	intergenic	NONE;NO	dist=NONE;dist=NONE	.
MT	16222	16222	C	T	intergenic	NONE;NO	dist=NONE;dist=NONE	.
X	24064263	24064263	G	A	exonic	EIF2S3	.	r
X	55718363	55718363	A	G	exonic	RRAGB	.	s
X	1,49E+08	1,49E+08	G	T	UTR3	IDS	NM_000202:c.*1470C>A;NM_001166550:c.*1470C>A	.

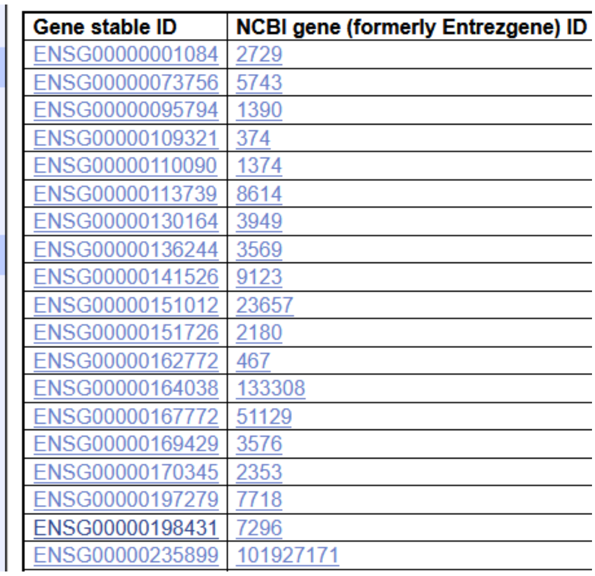
When i examined variations, i saw that four of them are on the mitochondirial chromosome, three of them are on the X chromosome. “EIF2S3” and “RRAGB” are exonic.

Enrichment Analysis

Gene set enrichment analysis (GSEA) is a method to identify classes of genes or proteins that are over-represented in a large set of genes or proteins, and may have an association with disease phenotypes. For enrichment analysis first i downloaded "c7.all.v7.1.entrez.gmt” and "c6.all.v7.1.entrez.gmt” files. Then i installed clusterProfiler, GSEABase and org.Hs.eg.db libraries with BiocManager. Then i ran following commands to get csv files.

```
filename <- "c7.all.v7.1.entrez.gmt"

gmtfile <- system.file(filename)

c6 <- read.gmt(gmtfile)

yourEntrezIdList<- c(7296,3569,1374,2353,2180,2729,23657,101927171,374,7718,133308,3949,3576,5743,467,1390,9123,8614,51129) #ENTREZID of DE genes

ImmunSigEnrich <- enricher(yourEntrezIdList, TERM2GENE=c6, pvalueCutoff = 0.01)

ImmunSigEnrich <- setReadable(ImmunSigEnrich, OrgDb = org.Hs.eg.db, keyType = "ENTREZID")

write.csv(ImmunSigEnrich,"MyImmunePathwayRelatedGenes.csv")

goEnrich<-enrichGO(gene= yourEntrezIdList,OrgDb= org.Hs.eg.db, ont= "ALL",pAdjustMethod="BH",pvalueCutoff = 0.01,readable= TRUE)

write.csv(goEnrich,"MyGORelatedGenes.csv")

keggEnrich<-enrichKEGG(gene= yourEntrezIdList,organism= "hsa",pAdjustMethod="BH", pvalueCutoff = 0.01)

write.csv(keggEnrich,"MyKEGGRelatedGenes.csv")

# Exit the R session

quit(save="no")
```

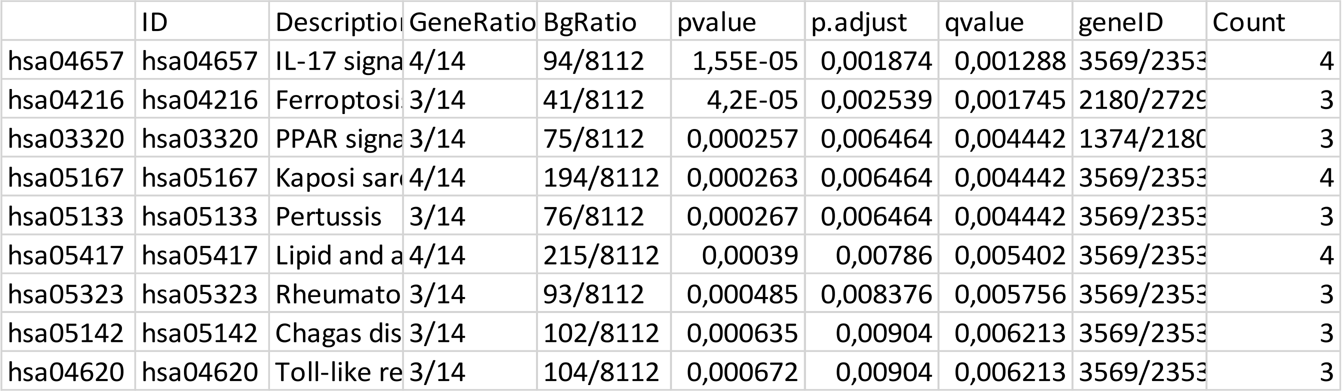
I used top 20 Differentialy expressed genes for EntrezIdList. I changed gene stable id’s to entrezgene id with biomart tool of ensembl.org



Gene stable ID	NCBI gene (formerly Entrezgene) ID
ENSG00000001084	2729
ENSG00000073756	5743
ENSG00000095794	1390
ENSG00000109321	374
ENSG00000110090	1374
ENSG00000113739	8614
ENSG00000130164	3949
ENSG00000136244	3569
ENSG00000141526	9123
ENSG00000151012	23657
ENSG00000151726	2180
ENSG00000162772	467
ENSG00000164038	133308
ENSG00000167772	51129
ENSG00000169429	3576
ENSG00000170345	2353
ENSG00000197279	7718
ENSG00000198431	7296
ENSG00000235899	101927171

KEGGRelatedGenes:

Descriptions are IL-17 signaling pathway, Ferroptosis, PPAR signaling pathway, Kaposi sarcoma-associated herpesvirus infection, Pertussis, Lipid and atherosclerosis, Rheumatoid arthritis, Chagas disease and Toll-like receptor signaling pathway. When we examine geneID 3569, this gene encodes a cytokine that functions in inflammation and the maturation of B cells.



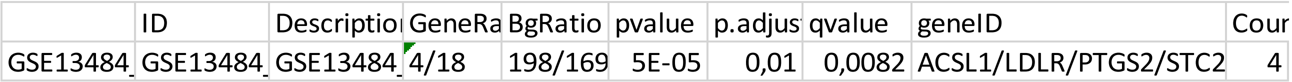
	ID	Descriptio	GeneRatio	BgRatio	pvalue	p.adjust	qvalue	geneID	Count
hsa04657	hsa04657	IL-17 signa	4/14	94/8112	1,55E-05	0,001874	0,001288	3569/2353	4
hsa04216	hsa04216	Ferroptosi	3/14	41/8112	4,2E-05	0,002539	0,001745	2180/2729	3
hsa03320	hsa03320	PPAR signa	3/14	75/8112	0,000257	0,006464	0,004442	1374/2180	3
hsa05167	hsa05167	Kaposi sar	4/14	194/8112	0,000263	0,006464	0,004442	3569/2353	4
hsa05133	hsa05133	Pertussis	3/14	76/8112	0,000267	0,006464	0,004442	3569/2353	3
hsa05417	hsa05417	Lipid and a	4/14	215/8112	0,00039	0,00786	0,005402	3569/2353	4
hsa05323	hsa05323	Rheumato	3/14	93/8112	0,000485	0,008376	0,005756	3569/2353	3
hsa05142	hsa05142	Chagas dis	3/14	102/8112	0,000635	0,00904	0,006213	3569/2353	3
hsa04620	hsa04620	Toll-like re	3/14	104/8112	0,000672	0,00904	0,006213	3569/2353	3

ImmunePathwayRelatedGenes:

The protein encoded by ACSL1 gene is an isozyme of the long-chain fatty-acid-coenzyme.

The LDLR gene provides instructions for making a protein called the low-density lipoprotein receptor.

STC2 gene encodes a secreted, homodimeric glycoprotein that is expressed in a wide variety of tissues and may have autocrine or paracrine functions.



	ID	Descriptio	GeneRa	BgRatio	pvalue	p.adjus	qvalue	geneID	Cour
GSE13484_	GSE13484_	GSE13484_	4/18	198/169	5E-05	0,01	0,0082	ACSL1/LDLR/PTGS2/STC2	4

GORelatedGenes:

All 50 genes ontologies are Biological Process. These genes are mostly about palmitate operation like response to nutrient levels, response to fatty acid, response to oxidative stress, cellular response to chemical stress, peptide transport, response to nutrient…

	ONTOL(	ID	Description	GeneRa	BgRatio	pvalue	p.adjust	qvalue	geneID	Coun
GO:00316(	BP	GO:00316(	response to nutrient levels	7/18	474/18723	1,59E-07	0,000185	9,01E-05	CPT1A/ACS	7
GO:00705	BP	GO:00705	response to fatty acid	4/18	64/18723	3,66E-07	0,000213	0,000104	CPT1A/ACS	4
GO:00069	BP	GO:00069	response to oxidative stress	6/18	446/18723	2,57E-06	0,000994	0,000485	IL6/FOS/G	6
GO:01500	BP	GO:01500	regulation of neuroinflammat	3/18	40/18723	7,21E-06	0,001609	0,000784	IL6/LDLR/I	3
GO:01500	BP	GO:01500	neuroinflammatory response	3/18	44/18723	9,64E-06	0,001609	0,000784	IL6/LDLR/I	3
GO:00513	BP	GO:00513	response to glucocorticoid	4/18	148/18723	1,05E-05	0,001609	0,000784	IL6/FOS/A	4
GO:19050	BP	GO:19050	carboxylic acid transmembran	4/18	149/18723	1,08E-05	0,001609	0,000784	CPT1A/ACS	4
GO:19038	BP	GO:19038	organic acid transmembrane t	4/18	150/18723	1,11E-05	0,001609	0,000784	CPT1A/ACS	4
GO:00319(	BP	GO:00319(	response to corticosteroid	4/18	167/18723	1,69E-05	0,002184	0,001064	IL6/FOS/A	4
GO:00075	BP	GO:00075	response to nutrient	4/18	174/18723	1,99E-05	0,00231	0,001126	ACSL1/GCL	4
GO:00065	BP	GO:00065	cysteine metabolic process	2/18	12/18723	5,73E-05	0,005684	0,00277	GCLC/SLC7	2
GO:00094	BP	GO:00094	response to xenobiotic stimul	5/18	462/18723	5,88E-05	0,005684	0,00277	CPT1A/FOS	5
GO:00076	BP	GO:00076	learning or memory	4/18	255/18723	8,85E-05	0,007534	0,003671	FOS/SLC7A	4
GO:00096	BP	GO:00096	response to toxic substance	4/18	262/18723	9,83E-05	0,007534	0,003671	TXNRD1/F	4
GO:00158	BP	GO:00158	peptide transport	4/18	264/18723	0,000101	0,007534	0,003671	IL6/CPT1A	4
GO:00067	BP	GO:00067	glutathione biosynthetic proce	2/18	16/18723	0,000104	0,007534	0,003671	GCLC/SLC7	2
GO:00066	BP	GO:00066	triglyceride metabolic process	3/18	100/18723	0,000114	0,007586	0,003697	CPT1A/ACS	3
GO:00017	BP	GO:00017	neutrophil homeostasis	2/18	17/18723	0,000118	0,007586	0,003697	IL6/SLC7A	2
GO:00191	BP	GO:00191	nonribosomal peptide biosynt	2/18	18/18723	0,000132	0,007676	0,003741	GCLC/SLC7	2
GO:19020(	BP	GO:19020(	fatty acid transmembrane trar	2/18	18/18723	0,000132	0,007676	0,003741	CPT1A/ACS	2
GO:19039	BP	GO:19039	regulation of microglial cell ac	2/18	19/18723	0,000148	0,007981	0,003889	IL6/LDLR	2
GO:00508	BP	GO:00508	cognition	4/18	296/18723	0,000157	0,007981	0,003889	FOS/SLC7A	4
GO:00016	BP	GO:00016	long-chain fatty acid metaboli	3/18	112/18723	0,000159	0,007981	0,003889	CPT1A/ACS	3
GO:00428	BP	GO:00428	amide transport	4/18	301/18723	0,000168	0,007981	0,003889	IL6/CPT1A	4
GO:00158	BP	GO:00158	organic acid transport	4/18	303/18723	0,000172	0,007981	0,003889	CPT1A/ACS	4
GO:00069	BP	GO:00069	inflammatory cell apoptotic pı	2/18	21/18723	0,000181	0,008091	0,003943	IL6/SLC7A	2
GO:00714	BP	GO:00714	cellular response to external s	4/18	320/18723	0,000212	0,008543	0,004163	FOS/GCLC/	4
GO:00066	BP	GO:00066	acylglycerol metabolic proces	3/18	128/18723	0,000236	0,008543	0,004163	CPT1A/ACS	3
GO:00066	BP	GO:00066	neutral lipid metabolic proces	3/18	129/18723	0,000242	0,008543	0,004163	CPT1A/ACS	3
GO:00620	BP	GO:00620	regulation of small molecule n	4/18	334/18723	0,00025	0,008543	0,004163	CPT1A/SLC	4
GO:00466	BP	GO:00466	response to organophosphoru	3/18	131/18723	0,000253	0,008543	0,004163	FOS/AREG/	3
GO:00621	BP	GO:00621	cellular response to chemical s	4/18	337/18723	0,000258	0,008543	0,004163	IL6/FOS/SL	4
GO:00481	BP	GO:00481	astrocyte activation	2/18	25/18723	0,000258	0,008543	0,004163	IL6/LDLR	2
GO:00485	BP	GO:00485	response to steroid hormone	4/18	339/18723	0,000264	0,008543	0,004163	IL6/FOS/A	4
GO:00105(	BP	GO:00105(	regulation of cellular ketone n	3/18	133/18723	0,000264	0,008543	0,004163	CPT1A/SLC	3
GO:00324	BP	GO:00324	response to lipopolysaccharid	4/18	343/18723	0,000276	0,008543	0,004163	IL6/FOS/C	4
GO:00456	BP	GO:00456	positive regulation of osteocla	2/18	26/18723	0,00028	0,008543	0,004163	FOS/SLC9B	2
GO:00466	BP	GO:00466	decidualization	2/18	26/18723	0,00028	0,008543	0,004163	PTGS2/STC	2
GO:00507	BP	GO:00507	positive regulation of inflamm	3/18	142/18723	0,000321	0,008923	0,004348	IL6/LDLR/I	3
GO:00026	BP	GO:00026	positive regulation of acute in	2/18	28/18723	0,000325	0,008923	0,004348	IL6/PTGS2	2
GO:00105	BP	GO:00105	positive regulation of vascular	2/18	28/18723	0,000325	0,008923	0,004348	IL6/PTGS2	2
GO:00705	BP	GO:00705	response to interleukin-1	3/18	143/18723	0,000327	0,008923	0,004348	IL6/GCLC/	3
GO:00076	BP	GO:00076	learning	3/18	144/18723	0,000334	0,008923	0,004348	FOS/SLC7A	3
GO:00022	BP	GO:00022	response to molecule of bacte	4/18	363/18723	0,000343	0,008923	0,004348	IL6/FOS/C	4
GO:00330	BP	GO:00330	myeloid cell apoptotic process	2/18	29/18723	0,000349	0,008923	0,004348	IL6/SLC7A	2
GO:00140	BP	GO:00140	response to purine-containing	3/18	148/18723	0,000362	0,008923	0,004348	FOS/AREG/	3
GO:00442	BP	GO:00442	sulfur compound biosynthetic	3/18	148/18723	0,000362	0,008923	0,004348	ACSL1/GCL	3
GO:00458	BP	GO:00458	positive regulation of lipid met	3/18	149/18723	0,000369	0,008923	0,004348	CPT1A/LDL	3
GO:00487	BP	GO:00487	regulation of astrocyte differe	2/18	31/18723	0,000399	0,009452	0,004606	IL6/LDLR	2