DPAC Manual

Written by Andrew Routh 2018-9.

Please contact alrouth@utmb.edu or @andrewrouth with questions.

All scripts have been successfully executed on Cygwin platforms and on Linux server.

Quick Start:

1. **Obtain raw data from NCBI SRA database (fastq-dump or download directly) (e.g.):**

```
for i in SRR5262326 SRR5262328 SRR5262330 SRR5262332 SRR5262334 SRR5262336;
do fastq-dump –A $i --gzip --origfmt; done
```

2. **(If not already present) Obtain hg19 hisat index from HiSat2 website:**
   https://ccb.jhu.edu/software/hisat2/index.shtml

3. **(If not already present) Generate or download hg19 genome fasta file (e.g.):**

```
Hisat2-inspect-s /data/Indexes/HISAT2/hg19/genome > hg19.fa
```

4. **Perform raw data prep and Map to human genome (assign permission of script first):**

```
wget 'https://sourceforge.net/projects/dpac-seq/files/DPAC_1.0.zip/download'

unzip download

chmod 755 ./DPAC_1.0/

./DPAC_1.0/DPAC -t 4 ./DPAC_1.0/Test_Data/MetaData.HeLa.txt
/data/Indexes/HISAT2/hg19/genome
./DPAC_1.0/Test_Data/hg19_HeLa_PASs.20Dec18.bed Ctrl-vs-CF1KD_test
```

Dependencies

Fastp:	https://github.com/OpenGene/fastp
Cutadapt:	https://cutadapt.readthedocs.io/en/stable/installation.html In python3 for multi-threading
Python2:	2.7 as default, numpy must be installed
Hisat2:	https://ccb.jhu.edu/software/hisat2/index.shtml
Samtools:	http://www.htslib.org/ version 1.9 or higher
Bedtools:	https://bedtools.readthedocs.io/en/latest/
R:	version >3.5
DESeq2:	https://bioconductor.org/packages/release/bioc/html/DESeq2.html

IHW: http://bioconductor.org/packages/release/bioc/html/IHW.html

Required files:

- **Raw poly(A)-tail target data (single end only)**

  e.g.: https://www.ncbi.nlm.nih.gov/sra/?term=pac-seq

- **Meta Data file (example provide in Test_Data):**

  Meta.Data file must be table delimited with following columns:

  Column 1 = Root for all files to be associated with a specific dataset
  Column 2 = Condition/Treatment for experiment
  Column 3 = path to raw data (if full not give, this is set using the –d argument)
  Column 4 = Comment

  For example: (MetaData.HeLa.txt, provided in Test_Data/)

```
#sample      treatment    datafile            comment

Ctrl_1       Wt           SRR5262330.fastq    MiSeq
Ctrl_2       Wt           SRR5262328.fastq    MiSeq
Ctrl_3       Wt           SRR5262326.fastq    MiSeq
CF1_1        CF1KD        SRR5262336.fastq    MiSeq
CF1_2        CF1KD        SRR5262334.fastq    MiSeq
CF1_3        CF1KD        SRR5262332.fastq    MiSeq
```

- **Hisat2 index for genome:**

  Build or download from hisat2 website: https://ccb.jhu.edu/software/hisat2/index.shtml

- **Poly(A)-Cluster file (example provide in Test_Data):**

  This is a bed file containing the output from a previous poly(A)-targeted sequencing experiment. This must contain the naming convention described in the accompanying manuscript in order to determine PAS usage, terminal exon usage and gene expression levels. We will deposit databases from different model organisms in due course, which will be available from the source-forge DPAC website.

Optional files if performing de novo PAS cluster annotation:

- **Reference Genome in fasta format**
- **Annotations file (example provided in Test_Data):**
  Download exon annotations for genome of interest from UCSC Genome Browser's Table Browser (https://genome.ucsc.edu/cgi-bin/hgTables); described below.
- **Accession Names file (example provided in Test_Data):**
  Download exon annotations for genome of interest from UCSC Genome Browser's Table Browser (https://genome.ucsc.edu/cgi-bin/hgTables); described below.

DPAC – Overview of operation:

```
USAGE: ./DPAC [OPTIONS] MetaData INDEX PASCLUSTER OUTPUTROOT

e.g.:

./DPAC -t 4 /Test_Data/MetaData.HeLa.txt /data/Indexes/HISAT2/hg19/genome
/data/Indexes/HISAT2/hg19/hg19.fa Test_Data/hg19_HeLa_PASs.28Dec18.bed Ctrl-vs-
CF1KD_test
```

Required Arguments:

`MetaData`

MetaData file containing root names, conditions and raw data path and an optional comment column. E.g. (Ctrl_1 Wt SRR5262330.fastq MiSeq). A full example table is given in the Test_Data folder and described above.

`Index`

Provide full path of the HISAT2 index.
e.g.: /data/Indexes/HISAT2/hg19/hg19

`Poly(A) Cluster Database`

Provide the full path of an existing Poly(A) cluster database. One is provided in the Test_Data folder, or new clusters can be downloaded from the DPAC sourceforge site.

Alternatively, if new PACs are going to be generated; the optional arguments –x and –y must be provided. In this case, this will generate a new PAC file with the given PAC cluster name.

`Output Root Names`

Enter a string with no whitespaces for a root of all the output files.
e.g.: Wt-vs-KD

Optional Arguments:

`-h` `show this help text`

`-p` `Perform custom stages;`

This options allow individual stages of the whole DPAC process to be selected in a custom fashion. Select a combination of P, M, C, and/or D. No whitespaces! Default = “PMD”: Preprocess, map and Differential Poly(A) Cluster analysis.

P = Only perform data preprocessing
M = Only map data
B = Make individual Bedgraphs for each input sample (requires –g GENOME is also selected)
C = force new PAS cluster generation (default = off).
Requires -g Genome, -x Annotation and -y Names (see manual).

D = Only perform Differential Poly(A) Cluster analysis
For example;

- If Novel Cluster generation and subsequent Differential PAS usage is require (data has already been preprocessed and BAM files are provided from a previous run) select "-p CD".
- If only data preprocessing is requested, select "-p P".
- To re-map data and then perform Differential PAS usage: select "-p MD"

`-t` set threads (default: 1)

`-a` set number of A's required in poly(A) tail (default: 5)

If generating new PAS clusters, this set the number of additional non-primer derived A's required to be found in each mapped read in order to be assaigned as mapping to a authentic poly(A) tail. Default value is 5 A's (which corresponds to 26 A's in total) See Routh et al NAR 2017 for further details.

`-r` set number of reads required per poly(A) event (default: 5)

If generating new PAS clusters, this set the number of reads required to be found in each Poly(A) site in order to be annotated as an authentic PAS.. Default value is 5 reads. See Routh et al NAR 2017 for further details.

`-d` Raw data directory if absent from MetaData file.

Provide full path to directory containing raw read data if not already present in Meta.Data.txt and the raw data is not in the same folder as the folder from which DPAC is called.

For example, if raw data MetaData states: `[Ctrl_1 Wt SRR5262330.fastq MiSeq]`
but the raw-data is saved to an external folder, add (e.g.):

```
-d /data/RawData/HeLa_Datasets/
```

`-o` set distance allowed for read mapping upstream of Poly(A) Cluster (default: 10)

The three-prime end of a Poly(A) targeted reads is expected to map either within as poly(A) cluster. However, to also count reads that may map shorty upstream of a PAC used this option (e.g. to include reads that did not contain a poly(A) tail, but were indeed derived from a poly(A)-tailed mRNA). This option maybe helpful when mapping poly(A)-targeted reads using strategies that do not focus perfectly on the junction of the 3'UTR/poly(A) tail or when sequenced reads lengths are too short to reach the poly(A) tail.

`-n` set number of replicates in experiment (default: 3).

Most often, three replicate of each condition are chosen. Number of replicates must be the same for each condition (i.e. not support for (e.g.) 4 Wt vs 3 KD.).

Options for PAS cluster generation:

These following options force new Poly(A) cluster generation, but must all be present otherwise DPAC will exit.

-g Genome

Provide full path the Genome in fasta format.
e.g. /data/Indexes/HISAT2/hg19/hg19.fa

-x Annotations, forces new cluster generation.

If generating new poly(A) Clusters, this option must be selected (also requires –y). Annotations can be downloaded in bed format for specific genome from UCSC as described above. Clustering will be skipped if Annotations is not set.

-y Gene Names, forces new cluster generation.

If generating new poly(A) Clusters, this option must be selected (also requires –x). Gene names can be downloaded in bed format for specific genome from UCSC as described above. Clustering will be skipped if Gene Names is not set.

Output

After running DPAC, a summary if output to the stdout/screen. For example:

```
Total genes detected: 10332
Number of genes with multiple terminal exons:3058
Number of exons with multiple poly(A) sites: 4889
Number of differentially expressed genes (padj<0.1): 462
Number of APA events: 146
         of which 49 are Lengthening, 83 are Shortening and 14 are both.
Number of alternative terminal exons (TE): 49
Number genes with both APA and TE: 9
```

DPAC returns three csv files directly from DESeq2 for Gene, Exons and Poly(A)-sites (PASs) respectively. A final compiled table for each gene, with gene exons, and PASs therein is the main output of DPAC. If a gene only has one PASs site and thus one terminal exon, only the gene information is output. Genes with differential expression (FoldChange > 1.5; padj > 0.1) are annotated with 'DOWN' or 'UP'. If there is no significant change, genes are labels as 'NC' (No Change).

e.g. :

Gene	baseMean	FoldChange	padj	GeneResult
ZNF280B	137.379667402	-9.73693676955	0.0055022202852	DOWN

If there are two or more PASs found in a gene, these are next scrutinized for the presence of terminal exon usage (annotated as TE) and alternative polyadenylation (annotated as APA) within each detected exon (if multiple). APA events are further classified as either resulting in 3'UTR lengthening or shortening, or as 'both' in the case that multiple PASs flank the moving PAS. These events are described with one line per gene with the following columns:

Gene	baseMean	FoldChange	padj	GeneResult	
Exons	baseMean	FoldChange	padj	ExonResult	TE_Result
1PASs	baseMean	FoldChange	padj	PASResult	APA_Result

If there are multiple exons per gene, or multiple PASs per exon, these are presented individually but clustered using square brackets. If there are multiple exons and multiple PASs, there will be multiple PASs entries, each denoted with their own squared brackets with each PASs inside. The column 'ExonResult' or 'PASResult' gives the index of the exons and/or PASs that have a significant change. If at least one PASs changes with FC > 1.5 and padj < 0.1, then TE or APA will be reported.

e.g.:

```
TMEM43  381.65680241    1.69461853268   0.98671549346   NC
TMEM43_exon_chr3:14183093       0       0       0       NC      NA
['TMEM43_exon_chr3:14183093_PAS-3', 'TMEM43_exon_chr3:14183093_PAS-4']
[140.889592570943, 225.98508486179]     [-0.48286631272982028, 0.52089939585848344]
[0.0890469008375832, 0.955783810145684] PAS#:1-only_UP  APA
```

If there are multiple exons and multiple PASs, there will be multiple PASs entries, each denoted with their own squared brackets with each PASs inside.

e.g.:

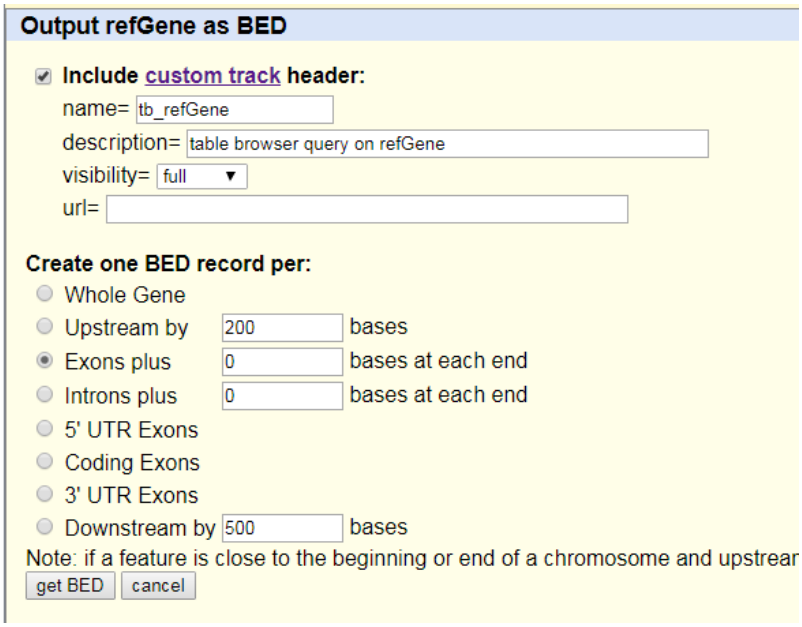
```
VSIR    772.320233928   -0.112229880001 1.0     NC      ['VSIR_exon_chr10:73507314',
'VSIR_intron_chr10:73515224']      [733.157745225319, 39.6499172866183]    [-
0.096484954417287994, 0.096484954417288021]   [1.0, 0.213295144795313]        NC
NotSig  ['VSIR_exon_chr10:73507314_PAS-2', 'VSIR_exon_chr10:73507314_PAS-1',
'VSIR_exon_chr10:73507314_PAS-4']  [494.116184254653, 65.1036529934643,
132.606901476511]  [0.11016894419876144, 0.16957572762968129, -0.36131516911544692]
[0.986895372302997, 0.444950345020261, 0.0937575102841144]      PAS#:3-only_UP  APA
```

De Novo **Cluster generation:**

To perform, de novo cluster generation, the "–p C" option must be invoked. To perform entire analysis (i.e. data processing, mapping cluster generation and differential PAS usage), select "-p PMCD". Two extra files are require in order to generate PACs dataset fresh. Examples for the human genome are provided in Test_Data. For up-to-date files and/or other model organisms, obtain these files from the UCSC genome browser as follows:

- **Reference Genome file in fasta format (required to mask A-rich regions)**
  FASTA file can be generated from provide HISAT2 index:
  hisat2-inspect INDEX > Genome.fa

- **Annotations file (example provide in Test_Data):**
  Download exon annotations for genome of interest from UCSC Genome Browser's Table Browser (https://genome.ucsc.edu/cgi-bin/hgTables) Described below.
  1. Select Clade and genome of interest (e.g. Mammal and Human)
  2. Select relevant database (e.g. Group = 'Genes and Gene Predictions'; Track = 'NCBI RefSeq'; Table = 'UCSC RefSeq (refGene).
  3. Select output format as 'BED – browser extensible data'
  4. Specific output file name (e.g. 'hg19_exons.bed')
  5. Click 'Get Output'
  6. In next dialogue box, specify 'Exons' (N.B. do not select 'whole gene' or 'Coding Exons').



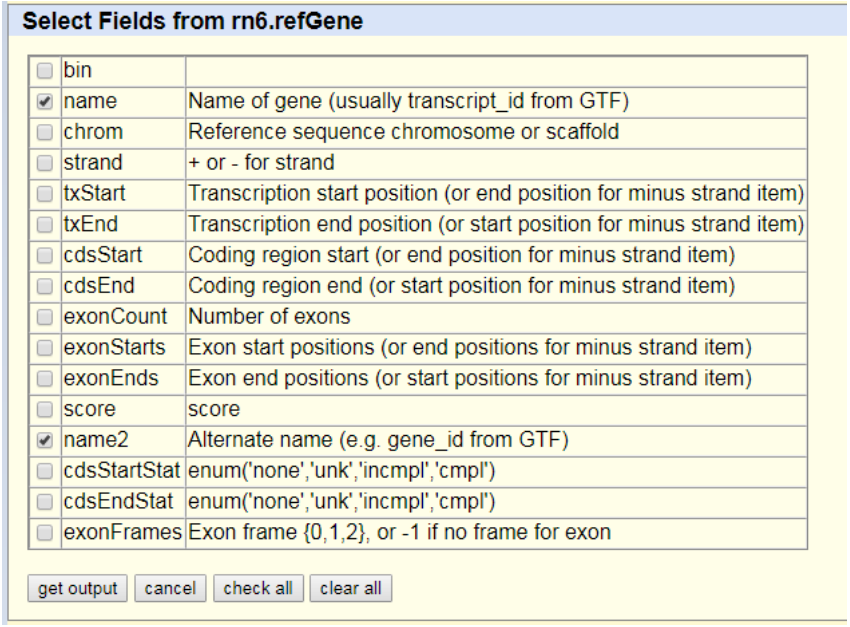
  7. Click 'get BED' to download BED file. In
  8. Repeat steps **a-g** above, except in step **d**, set save file as "hg19_introns.bed'" and in step **f**, select "Introns".
  9. Repeat steps **a-g** above, except in step **d**, set save file as "hg19_DS250.bed'" and in step **f**, select "Downstream by 250 bases".
  10. Combined 3 bed files into one:

```
cat hg19_exons.bed hg19_introns.bed hg19_DS250.bed >
hg19_Exons-introns-DS250.bed
```

- **Accession Names file (example provide in Test_Data):**

Download exon annotations for genome of interest from UCSC Genome Browser's Table Browser (https://genome.ucsc.edu/cgi-bin/hgTables);

a. Repeat steps **a-b** above, and in output format select: "selected fields from primary and related tables"
b. In output file box, name file (e.g. "hg19_names.txt")
c. Click 'get output'
d. Select check boxes for only "name" and "name2".



Select Fields from rn6.refGene

☐	bin	
☑	name	Name of gene (usually transcript_id from GTF)
☐	chrom	Reference sequence chromosome or scaffold
☐	strand	+ or - for strand
☐	txStart	Transcription start position (or end position for minus strand item)
☐	txEnd	Transcription end position (or start position for minus strand item)
☐	cdsStart	Coding region start (or end position for minus strand item)
☐	cdsEnd	Coding region end (or start position for minus strand item)
☐	exonCount	Number of exons
☐	exonStarts	Exon start positions (or end positions for minus strand item)
☐	exonEnds	Exon end positions (or start positions for minus strand item)
☐	score	score
☑	name2	Alternate name (e.g. gene_id from GTF)
☐	cdsStartStat	enum('none','unk','incmpl','cmpl')
☐	cdsEndStat	enum('none','unk','incmpl','cmpl')
☐	exonFrames	Exon frame {0,1,2}, or -1 if no frame for exon

get output | cancel | check all | clear all

e. Click "get output" to download names.