EditExpress User Guide

Eric Marshall

Package version: EditExpress 1.0.0

**Contents**

1. Quick Start Guide

* What is EditExpress?
* Installation instructions
* Reference file formatting
* How to run
* Quickstart

1. EditExpress Configurable Parameters
2. EditExpress Results
3. FAQ
4. **Start Guide**

**What is EditExpress?**

EditExpress is a general tool to analyze amplicon sequencing data of CRISPR-mediated (works on ZFN editing too at least) genome editing. It works an experimental design with a single small guide RNA (sgRNA) or multiple sgRNAs.

**Installation Instructions**

EditExpress can be installed by downloading or cloning from **GitHub repo link**

Pre-requisites:

* Programs
  + Samtools (tested w/ version 1.3.1)
    - <http://samtools.sourceforge.net/>
  + BWA (tested w/ version 0.7.15)
    - <http://bio-bwa.sourceforge.net/>
  + Emboss Needle (tested w/ version 6.6.0)
    - <http://emboss.sourceforge.net/>
  + FLASH (tested w/ version 1.2.11)
    - <http://ccb.jhu.edu/software/FLASH/>
  + PANDASeq (tested w/ version 2.10)
    - <https://github.com/neufeld/pandaseq>
  + Seqtk (tested w/ version 1.0-r82-dirty)
    - <https://github.com/lh3/seqtk>
  + FeatureCounts from Subread (tested w/ version 1.5.0)
    - <http://subread.sourceforge.net/>
  + Bedtools (tested w/ version 2.25.0, only required for the in progress offtarget tool)
    - <http://bedtools.readthedocs.io/en/latest/>
* Python 2.7 or Python 3+ and dependencies:
  + pysam
  + pandas
  + numpy
  + matplotlib
  + reportlab
  + svgwrite
  + Xslxwriter
  + pillow (cannot coexist with PIL)
  + biopython (for *offtarget* only)

**Reference File Formatting**

For the standard and gapped, EditExpress requires a tab or space-delimited reference.txt file formatted as follows:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **target** | **primer1** | **primer2** | **Amplicon** | **sgRNA** |
| target\_A | primer1\_A | primer2\_A | seq\_A | sgRNA\_A |
| target\_B | primer1\_B | primer2\_B | seq\_B | sgRNA\_B |

**Reference Fields** (all are case-insensitive)

* Target - The name of your amplicon/target sequence, for example EGF, SMN-promoter, etc.
* Primer1/Primer2 - EditExpress requires primer sequences to be listed. primer1 is the 5’ primer and primer2 is the 3’ primer. If your experimental design uses single-end reads, then primer2 should be omitted. These are used to filter reads after assembly for paired end reads or generally in the case of single end reads. Ideally, these primer sequences should already be in your amplicon sequence.
* Amplicon - Your amplicon sequence. As mentioned above, ideally it will contain your primer sequences, but they will be added if necessary. Note that variant positions are annotated relative to the amplicon sequence with primers included.
* sgRNAs(optional) - You can specify your guide RNA sequences (one or more). These are only used for generating the SVG alignment plot and (optionally) restricting the alignment region for the mutation tables, in tandem with the 'rel\_cleave\_sites' parameter, explained later. As with the primer sequences, what you label the sgRNA fields doesn't really matter so long as they contain 'sgRNA'. Unlike the primer sequences, the order the guides are given in should not matter.

**Running EditExpress**

EditExpress offers three different workflows - standard, gapped and offtarget. The *standard* workflow is applicable for an experimental design with one or more sgRNAs for which paired-end reads (or single-end reads) **cover the** **whole** amplicon sequence. *Gapped* is intended for paired-end experiments where R1 and R2 reads **do not** **cover the** **whole** amplicon sequence, such as whole gene knockouts through large deletions. *Offtarget* is for exploring off-target effects of Crispr (or other gene editing technologies), but is not yet fully developed and tested.

Additionally, EditExpress uses three modules relevant to the *standard* and *gapped* workflows – QC, alignment and mutation calling, which may be turned off and on. **Alignment** merges paired-end reads, filters out reads with a poor match to the primer sequences, and runs Needle (*standard*) or bwa mem (*gapped*) to produce bam files. **Mutation calling** parses bam files and generates mutation tables, pie charts and alignment plots (alignment plots are *standard* only). **QC** performs an alignment to a genomic sequence to get an overview of where the reads are mapped to, produces a sorted count table and a pdf file summarizing some read quality metrics.

The last important point to touch on is the three available runmodes - project, single\_sample and summary. **Project** mode will perform sample level analysis on every sample and generate a cross-sample summary. If you have a multi-core computer with a fair bit of RAM, then you can increase the running speed by increasing the n\_processes (see **[multiprocess]** below). **Single\_sample** mode runs EditExpress only on one sample, which must be specified by additional command line parameters (i.e. python editexpress.py --pipeline\_config pipeline.conf --fastq1 f1 --fastq2 f2). This is only intended for users who want to parallelize the run for each sample via submitting jobs in an HPC environment by qsub or bsub. **Summary** mode performs only the summary steps to create the cross-sample mutation table and QC pdf and must be used after running EditExpress on **single\_sample** mode to complete a run. The **summary** mode can be used to tweak summary thresholds and produce new output files from **project** runs as well.

**Quickstart**

At minimum to run EditExpress, you will need to create a reference file, as specified above, and a short configuration file. It should look as follows:

[i/o]

fastq\_directory = /path/to/fastqs

output\_directory = /desired/output/path/

[reference\_files]

amplicon\_reference = /path/to/reference.txt

genome\_fasta = /path/to/genome.fasta

genome\_gtf = /path/to/genome.gtf

[processing]

R1\_suffix = \_L001\_R1\_001.fastq

R2\_suffix = \_L001\_R2\_001.fastq

The above can then be saved to a text file (named, say, editexpress.conf). EditExpress can then be run as below:

python editexpress.py –-pipeline\_config /path/to/editexpress.conf

In this case, EditExpress will use default values for parameters and run QC, alignment, mutation calling in **project** mode with 1 process.

**Note: the quickstart assumes the following programs are not only installed, but are in your environment** (e.g. you could run bwa mem just by typing ‘bwa mem’ on the command line).

* + bwa
  + needle
  + samtools
  + featureCounts
  + flash
  + seqtk

If this is not the case, then the paths must be specified in section **[programs]** of the config file.

1. **Complete List of Configurable Parameters**

All the following parameters are configurable by the user. Lines can be commented out by using ‘#’ at the beginning of the line. Multiple instances of the same argument will use whichever occurs last.

**[i/o]**

fastq\_directory - The directory where all your fastqs (both R1 and R2 if running in paired end mode) are located. Gzipped fastqs are also supported, but require a ‘.gz’ extension to be added to the R1/R2 suffix parameters. (Required)

output\_directory - The directory where EditExpress creates its run folder. (Default: './')

static\_config - An optional parameter. Path to an additional config file containing all the parameters you aren’t likely to change run-to-run but differ from the default parameters (e.g. customized FLASH settings). **Values specified in the static config have priority over the standard config file. Be careful about specifying the same parameter in both.** (No default)

**[reference\_files]**

amplicon\_reference - Path to your reference.txt file. (Required)

genome\_fasta - Path to a fasta format reference file, preferably one that is already indexed with bwa. (No default; required for running QC)

genome\_gtf - Path to a genome gtf. (No default; required for running QC)

**[qc\_parameters]**

These are all straighforward percent thresholds used in running the QC. Samples that fail to meet at least one of these parameters are flagged and output in the QC\_summary.pdf file. (Default: 50)

percent\_assembled - Threshold for read assembly/merging (via flash or pandaseq)

percent\_primered - Threshold for primer rate, checked in the R1/R2 reads individually and post read assembly.

percent\_mapped\_amplicon - Threshold for essentially the remaining reads post assembly and primer-filtering

percent\_mapped\_genome - Threshold for reads mapping to whatever genome\_fasta was given.

percent\_max\_rl - Threshold for the R1/R2 reads over 90% of the maximum read length (e.g. given have 150bp reads, how many are at least 135 bp long?)

**[runtime]**

runmode - Choose how you want EditExpress to run. Project mode runs both the sample level steps and summary steps, and allows you to parallelize with multiprocessing (the 'n\_processes' option) if you have multiple cores available. This is the preferred runmode. The other two, single\_sample and summary divide the pipeline steps into their namesakes. Single sample mode requires you additional parameters '--fastq1 /path/to/R1.fastq' and (if paired end) '--fastq2 /path/to/R2.fastq' (or only the fastq name if the path is fastq\_directory is given in **[i/o]**) when calling EditExpress. The summary runmode completes a set of single\_sample runs, or if you want to regenerate your QC/topSeqs summary files with different parameters. (Default: project)

**[modules]**

#Note that if you choose offtarget, the other three parameters in this section are ignored.

workflow - Choose between one of three analysis workflows. Default is the standard pipeline, gapped is for the detection of large deletions, and offtarget is for finding offtarget effects (yet untested). (Default: standard, choices: standard, gapped, offtarget)

alignment - Run the alignment module. (Default: yes, choices: yes/no or true/false)

run\_qc - Run the QC module. This requires output from the alignment module. (Default: yes, choices: yes/no or true/false)

mutation\_calling = Run the mutation\_calling module. This requires output from the alignment module. (Default: yes, choices: yes/no or true/false)

**[processing]**

paired\_end - Flag for paired end reads. A value of 'no' will assume single end reads. (Default: yes, choices: yes/no or true/false)

read\_merger - As of now, a choice between two read merging tools – flash and pandaseq. Used in the 'standard' workflow. (Default: flash, choices: flash, pandaseq)

R1\_suffix - Used to match fastqs in the given 'fastq\_directory'. Whatever is specified will be trimmed off and the rest interpreted as sample name. If files are gzipped, the suffix must end in ‘.gz’. (Default: \_L001\_R1\_001.fastq)

R2\_suffix - Used to match fastqs in the given 'fastq\_directory'. Optional if running in single sample mode. (Default: \_L001\_R2\_001.fastq)

merge\_mut\_tables - Optional. Merge sample level mutation tables into a single .xlsx file. Sample names cannot exceed 32 characters for this to function (hard cap for Excel sheet names). Default: no, choices: yes/no or true/false)

clean\_intermediates - Delete intermediate folders and files, leaving mostly output files. (Default: no, choices: yes/no or true/false)

**[mutation\_calling]**

restrict\_alignment - Given specified sgRNAs/cutting sites, restrict the alignment reported in the mutation tables to match that shown in the alignment plot. This is controlled by the same section that handles alignment plots, **[alignment\_plotting\_restriction]**, with parameters 'seq\_range', 'rel\_cleave\_sites', and 'abs\_cleave\_sites'. Does not affect the 'gapped' workflow. (Default: no, choices: yes/no or true/false)

bwa\_min\_mapq - For the gapped workflow, the quality filter for mapped reads. (Default: 15)

summary\_threshold - Threshold percentage for variant coverage to be included in the summary file. If no variants for a sample pass this threshold, then nothing from that sample will appear in the topSeqs file. (Default: 3.0)

topSeqs\_min\_hits - An absolute threshold for the number of hits a variant must have to be included in the topSeqs file (Default: 10)

variant\_filter - A filter applied to variants to determine whether they should be kept. This looks at each variant within a hit. For example, let the variant\_filter be set to its default value, 0.5%. During analysis, the filter comes across two variants: one with a 10bp deletion and single nt substitution, with a frequency of 10%, and another with the same 10bp deletion but no single nt substitution, and a frequency of 0.1%. Since the 10bp deletion appears across all reads at a rate of 10.1%, the rarer variant is retained. The sequence is regenerated based on what (if any) variants remain. **This filter can reclassify variants as WT. Depending on your preference, this may not be ideal. Disable the filter by setting its value to 0.** (Default: 0.25)

parse\_substitutions - Choice to keep substitutions in any variant containing output or not. Sequences/mutation strings are regenerated based on what (if any) variants remain. (Default: yes, choices: yes/no or true/false)

**[alignment\_plotting\_restriction]**

seq\_range - A range of positions to plot or use for alignment restriction. The range is specified via ‘-‘, e.g. n1-n2. Ranges for multiple values can be provided, separated by commas. If no value is provided, then by default EditExpress will plot 2/3 of the length of the amplicon from its’ center, up to a maximum of 80bp. This default range will only be used for plotting purposes, and can not be used to produce restricted mutation tables. (No default value, default range up to 40bp around the center of the amplicon)

rel\_cleave\_sites - An integer specifying the location of the cut site relative to the sgRNA(s). A negative number N means that the guide's cutting site is located N bp upstream from the 3' end; a positive number M means that the cutting site is located M bases downstream from the 5' end. For typical experimental designs using CAS9, the number will be -3. If this does not cover where your cutting site is, then it must be specified with abs\_cleave\_sites. If this parameter and abs\_cleave\_sites are blank or invalid, then the alignment plot will be produced without annotating cut sites. (No default)

abs\_cleave\_sites - Coordinates on your amplicon (with primers included) where your cut site is located. Multiple cut sites can be given as a comma delimited list, e.g. posA, posB. For multiple amplicon sequences, each set must be wrapped in parentheses, e.g. (posA, posB), (posC, posD). If this parameter and rel\_cleave\_sites are blank or invalid, then the alignment plot will be produced without annotating cut sites. (No default)

**[offtarget]**

bwa\_min\_mapq - For the offtarget workflow, the quality filter for mapped reads. (Default: 15)

percent\_cutoff = 0

n\_support = 1

**[multiprocess]**

n\_processes - The number of processors available for multiprocessing. This is checked against the actual number as reported by your system, so it cannot exceed that. Only functional in the **project** runmode. (Default: 1)

* **Memory usage and multiprocessing**

Increasing the number of processes increases the amount of memory consumed accordingly. My own testing puts EditExpress sans-QC using around 50-75 MB per thread; the QC is much costlier, using about 5 GB per thread. This increased memory usage stems entirely from using bwa mem to perform genome alignment. If these alignment files (e.g. **out\_dir/genome\_bams/**sample\_name**.genome.sam** or **.genome.bam**) exist already, then peak memory usage decreases to normal levels. Depending on your system, it may be desirable to run QC separately, or create the sam files yourself.

**[programs]**

#These may be relative or absolute paths (i.e. if you can just type 'samtools' then that will work as the path). Relative paths are default. If you do not have execute access to most of these programs, then EditExpress will not run successfully. Only one of flash or pandaseq is required.

featureCounts - The path to featureCounts from Subread. (Default: featureCounts)

samtools - The path to samtools. (Default: samtools)

needle - The path to needle in the emboss suite. (Default: needle)

bwa - The path to the bwa aligner. (Default: bwa)

pandaseq - The path to the pandaseq read assembly tool. (Default: pandaseq)

seqtk - The path to the seqtk read processing tool. (Default: seqtk)

bedtools - The path to bedtools (only required for offtarget). (Default: bedtools)

**[flash]**

#options given for the FLASH read assembly tool, one of three supported. See <https://ccb.jhu.edu/software/FLASH/> for more details.

min\_overlap = 5

max\_overlap = 65

max\_mismatch\_density - Maximum allowed ratio between the number of mismatched base pairs and the overlap length. (Default: 0.25)

allow\_outies – Allow read combination in “outie” as well as “innie” orientation. (Default: no)

Outie -

Read 1: <-----------

Read 2: ------------>

Innie –

Read 1: <------------

Read 2: ----------->

phred\_offset – Which phred offset the fastqs use (33 or 64). (Default: 33)

**[pandaseq]**

#options given for the PANDASeq read assembly tool, one of three supported. See <https://github.com/neufeld/pandaseq> for more details. Tested version - 2.10

phred64 - Whether fastqs use phred64. Derived from the '-6' parameter. (Default: no, i.e. not phred64)

algorithm - A choice between multiple algorithms. Derived from the '-A' parameter. (Default: flash, choices: pear, simple\_bayesian, ea\_util, flash, rdp\_mle, stich, uparse)

no\_uncalled\_bases - Eliminate (yes) or retain (no) uncalled bases. Derived from the '-N' flag. (Default: no, choices: yes/no or true/false)

min\_overlap - Minimum overlap size. Derived from the '-o' parameter. 5 max\_assembly\_length - Max assembly length. Derived from the '-L' parameter. (Default: 499)

min\_assembly\_length - Min assembly length. Derived from the '-l' parameter. (Default: 120)

threshold\_score - A score between 0 and 1 that a sequence must keep to be retained in the output. Derived from the '-t' parameter. (Default: 0.6)

1. **Results**

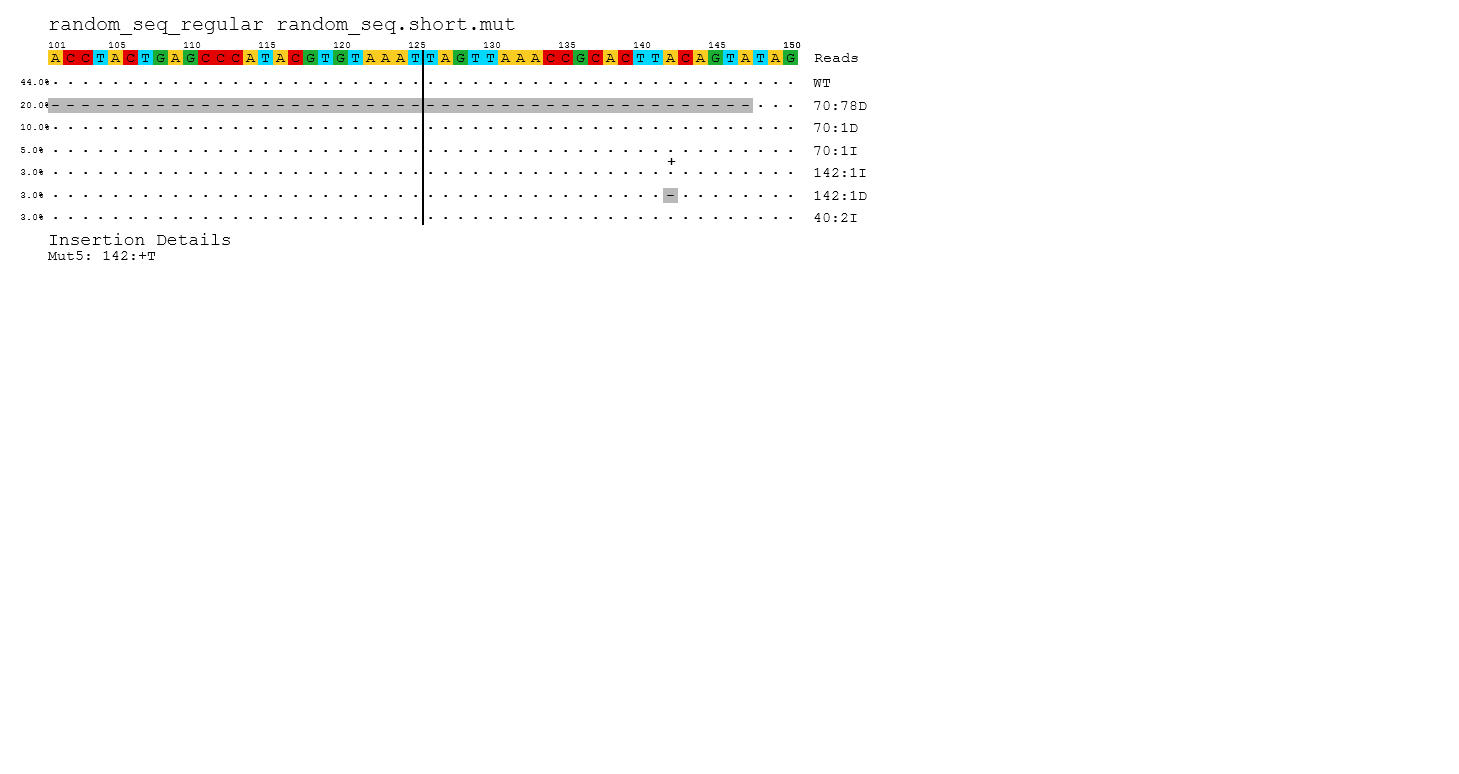
**Simulated Data**

Two simulated datasets are provided in **[Github].** Each set contains a total of 10,000 reads, derived by generating a random fasta sequence using FaBox (<http://users-birc.au.dk/biopv/php/fabox/>), then simulating mutations via ART (<https://www.niehs.nih.gov/research/resources/software/biostatistics/art/>). Each variant has approximately 1.5% sequence variability.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | **Start** | **End** | **Mutation** | **Frequency** | **N\_Reads** |
| *Long Seq* | |  |  |  |  |  |
| WT | | 1 | 2000 | None | 0.51 | 5100 |
| 50:3I(ACC) | | 50 | 50 | Ins | 0.05 | 500 |
| 100:1751D | | 100 | 1850 | Del | 0.2 | 2000 |
| 99:1745D | | 99 | 1844 | Del | 0.1 | 1000 |
| 99:3D | | 99 | 101 | Del | 0.05 | 500 |
| 1852:2I(AC) | | 1852 | 1852 | Ins | 0.03 | 300 |
| 1852:5D | | 1852 | 1856 | Del | 0.03 | 300 |
| 1875:T | | 1875 | 1875 | Mut | 0.03 | 300 |
| *Short Seq* | |  |  |  |  |  |
| WT | | 1 | 286 | None | 0.44 | 4400 |
| 40:2I (GT) | | 40 | 40 | Ins | 0.03 | 300 |
| 70:1D | | 70 | 70 | Del | 0.1 | 1000 |
| 70:1I (G) | | 70 | 70 | Ins | 0.05 | 500 |
| 70:78D | | 70 | 147 | Del | 0.2 | 2000 |
| 69:C | | 69 | 69 | Mut | 0.02 | 200 |
| 72:5D | | 72 | 76 | Del | 0.02 | 200 |
| 72:5I (GCCTA) | | 72 | 72 | Ins | 0.02 | 200 |
| 142:1D | | 142 | 142 | Del | 0.03 | 300 |
| 142:1I (T) | | 142 | 142 | Ins | 0.03 | 300 |
| 141:5D | | 141 | 145 | Del | 0.02 | 200 |
| 141:5I (TATCG) | | 141 | 141 | Ins | 0.02 | 200 |
| 160:G | | 160 | 160 | Mut | 0.02 | 200 |

**Example Results**

**Alignment Plot**



An example alignment plot using the simulated short sequence, a cutting site at position 200 and 50 bases flanking either direction. The segment of the reference sequence is detailed at the top; variants are listed by decreasing abundance. A match to the reference is represented as •. Deletions are represented with a grey background and a **-**. Insertions receive a **+** above the position where they occur, and a note below the plot giving the inserted sequence. Substitutions (none pictured) are simply given as the alternate base, in color.

**Sample level mutation table**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Ref** | **Mut** | **Frameshift** | **N\_mut** | **N\_mapped** | **Mut %** | **Alignment** |
| random.seq.regular | WT | no | 4401 | 10000 | 44.01 | [Sequence] |
| random.seq.regular | 70:78D | no | 2000 | 10000 | 20 | [Sequence] |
| random.seq.regular | 70:1D | yes | 1000 | 10000 | 10 | [Sequence] |
| random.seq.regular | 70:1I | yes | 500 | 10000 | 5 | [Sequence] |
| random.seq.regular | 142:1D | yes | 300 | 10000 | 3 | [Sequence] |
| random.seq.regular | 142:1I | yes | 300 | 10000 | 3 | [Sequence] |
| random.seq.regular | 40:2I | yes | 300 | 10000 | 3 | [Sequence] |
| random.seq.regular | 160:G | no | 200 | 10000 | 2 | [Sequence] |
| random.seq.regular | 141:5D | yes | 200 | 10000 | 2 | [Sequence] |
| random.seq.regular | 71:G | no | 200 | 10000 | 2 | [Sequence] |
| random.seq.regular | 141:5I | yes | 200 | 10000 | 2 | [Sequence] |
| random.seq.regular | 73:5D | yes | 200 | 10000 | 2 | [Sequence] |
| random.seq.regular | 72:5I | yes | 199 | 10000 | 1.99 | [Sequence] |

A sample level mutation table has the following columns:

* Ref – name of the reference sequence
* Mut – A mutation string representing the variants or lack thereof (WT, or Exp\_WT for EditExpress *gapped*). Mutations are formatted as **pos:[indel length]variant.** That is, a 1bp deletion at position 100 would look like 100:1D, while a substitution to an A at position 50 would look like 50:A
* Frameshift – A yes or no value indicating whether the mutation implies a frameshift.
* N\_mut – Number of this variant counted
* N\_mapped – Count across all variants
* Mut\_% - Percent occurrence of this variant
* Alignment – The sequence for this alignment (EditExpress *gapped*)

**Multi-sample mutation table**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Sample** | **Ref** | **N.mapped** | **WT** | **WT%** | **Mut** | **N.mut** | **Mut %** | **Frameshift** | **Alignment** |
| Samp2 | Amplicon | 82946 | 52155 | 62.88 | 214:11D | 29626 | 35.72 | yes | [Sequence] |
| Samp3 | Amplicon | 94935 | 57835 | 60.92 | 213:1I | 20191 | 21.27 | yes | [Sequence] |
| Samp4 | Amplicon | 86631 | 56512 | 65.23 | 210:2D | 29658 | 34.23 | yes | [Sequence] |
| Samp5 | Amplicon | 130164 | 121130 | 93.06 | 210:2D | 7557 | 5.81 | yes | [Sequence] |
| Samp6 | Amplicon | 78454 | 39438 | 50.27 | 212:1I | 23973 | 30.56 | yes | [Sequence] |
| Samp7 | Amplicon | 84759 | 60982 | 71.95 | 210:4D;214:C;  216:C | 22272 | 26.28 | yes | [Sequence] |
| Samp8 | Amplicon | 71819 | 22482 | 31.3 | 210:2D | 18850 | 26.25 | yes | [Sequence] |
| Samp10 | Amplicon | 50006 | 46777 | 93.54 | 214:T;216:T;  218:4D;222:A | 3168 | 6.34 | yes | [Sequence] |
| Samp12 | Amplicon | 46110 | 23463 | 50.88 | 210:C;212:13D;  228:21D;249:T | 22528 | 48.86 | yes | [Sequence] |
| Samp13 | Amplicon | 54347 | 12147 | 22.35 | 213:3D;218:T | 24425 | 44.94 | no | [Sequence] |

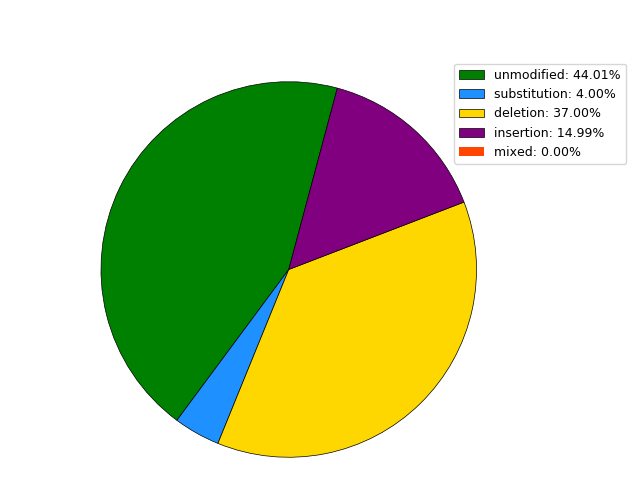
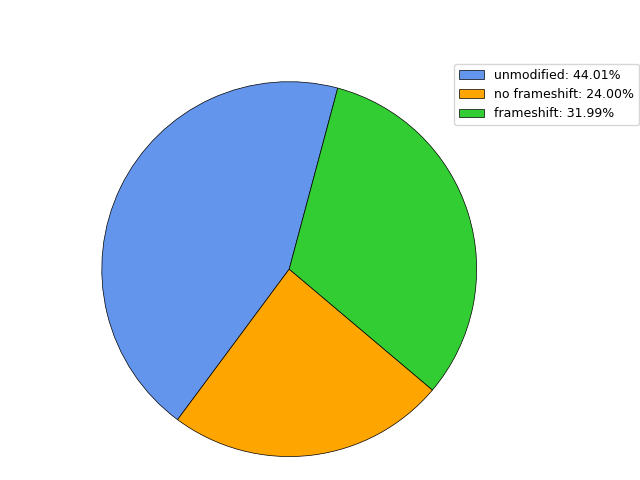
This file contains variant information for multiple samples. Samples which do not have any variants passing the summary threshold (default: 3.0%) will not be present

* Sample – Sample name from which all variants in this line are from
* Ref – Reference name
* N\_mapped – Total variant/WT count for this sample
* WT – Number of WT reads from this sample (Exp\_WT for EditExpress *gapped*)
* WT\_% - Percent of WT reads for this sample
* Mut – A variant from this sample passing the summary threshold
* N\_mut – Number of reads for this variant
* Mut\_% - Variant percentage
* Frameshift – yes or no indicating whether a mutation implies a frameshift
* Alignment – The aligned sequence (not output for EditExpress *gapped*)

If a sample contains multiple mutations above the summary threshold, then the column values for Mut, N\_mut, Mut\_%, Frameshift and Alignment are repeated (the column names, however, are not).

**Pie Charts**

EditExpress *standard* and *gapped* also output 2 pie charts – one summarizing variant types, and the other summarizing frameshifts.

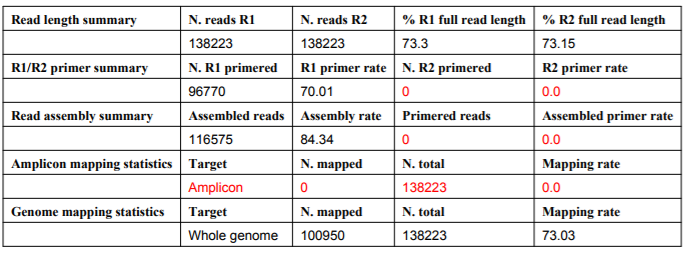
 

**QC Summary Table**

The QC module checks sample quality against several straightforward metrics.

* Read length summary: checks how many reads are approximately full length
* R1/R2 primer summary (*standard* only): checks primer rates (% of reads containing primer sequence, allowing for 1 mismatch) for R1 and R2 reads separately (5’ primer for R1, 3’ primer for R2)
* Read assembly summary (*standard* only): checks assembly rate (% of reads successfully merged via FLASH/PANDASeq) and assembled primer rate (% of assembled reads containing primer sequences)
* Amplicon mapping statistics: mapping rate to amplicon
* Genome mapping statistics: mapping rate to genome fasta

Below is an example of these statistics for a run given an incorrect 3’ primer sequence.



1. **FAQ**

**When should I use Gapped versus Standard?**

Standard should be applicable in any situation where your reads span most of the length of the amplicon sequence, be it through long or short read sequencing.

**With multiple options for read assembly tools available, which should I choose?**

Both should be comparable in most situations. FLASH is a bit easier to install, in my opinion, hence why it is the default option. PANDASeq has a few different algorithms available, which may be useful if your favorite read merger is not supported.

**How do I open the alignment plot?**

The easiest way to open the alignment plot or any SVG file is with your browser. Also, recent versions of Microsoft Office (since 2016) natively support displaying and converting them to other image formats.

**What does ‘Exp\_WT’ mean when I use EditExpress *gapped*?**

This means that neither R1/R2 reads contained any mutations, but we may lack justification to call them ‘WT’, presuming R1/R2 do not completely cover the amplicon sequence.