

NovoBarCode V1.00

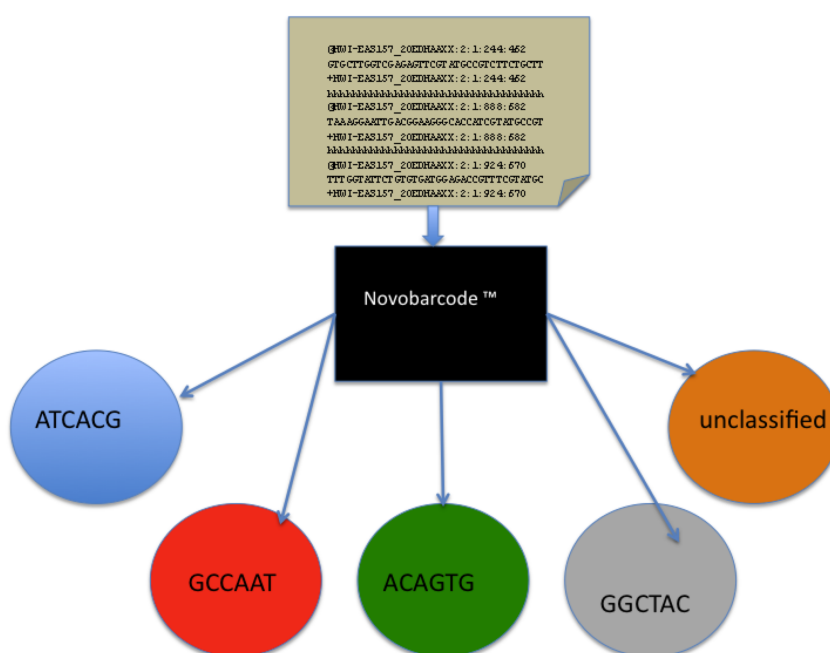
Synopsis

Novobarcodes demultiplexes Illumina/Solexa GAI™ reads based on a set of tag/index sequences, it can process

- Illumina indexed reads with index tag sequence in the header.
- Tag sequence embedded in 5' or 3' end of reads
- Tag sequence on one or both reads of a pair
- Multiple file formats including prb.txt
- Mismatches in tags with classification to most probable tag
- Uses base quality scores in calculating tag alignment scores

In this version classification is based on an ungapped alignment against the tag sequence. The alignment uses base qualities and will classify the read against the most probable tag. Reads with low quality tag alignments are written to a catch-all file with the tag sequence intact.

Except in Illumina format, the tag sequence is stripped from classified reads.



The Tag File (Mandatory)

The tag file defines the tag sequences; the approx distance between tags in terms of bp difference; and the location of the tags on the read. You should include every tag that was used in your sample preparation. Leaving tags out of the tag file can result in misclassification of reads as Novobarcode will classify a read according to tag that it aligns to with the lowest score and subject to the quality threshold specified with the -t option.

Distance 9	<p>Specifies the approx distance in bp differences between a tag and its nearest neighbour. This is used in establishing a quality value for exact tag matches and for when only one tag is specified.</p> <p>Example</p> <p>Distance 3</p> <p>Will mean that tags that match exactly are given an alignment quality of $3 \times 30 = 90$.</p> <p>If only one tag is specified the quality would be $3 \times 30 - 2 \times Sc$ where Sc is the alignment score.</p>												
Format <i>r1</i> [<i>r2</i>]	<p>Specifies the location of the tag within the read. This is ignored if the tags are in the FASTQ headers using Illumina format.</p> <p><i>r1</i> indicates position of the tag on single end reads or the first read of a pair. Values are:</p> <table> <tr> <td>5</td><td>Barcode is in 5' end of read 1</td></tr> <tr> <td>3</td><td>Barcode is in 3' end of read 2</td></tr> <tr> <td>H</td><td>Barcode is in Header record using Illumina Casava (pre V1.8) pipeline format.</td></tr> <tr> <td>N</td><td>There is no barcode on Read 1 of a pair, in this case read 2 must have barcode on 5' end.</td></tr> </table> <p><i>r2</i> specifies tag location on second read of a pair. Legal values are:</p> <table> <tr> <td>5</td><td>Barcode is in 5' end of read 2</td></tr> <tr> <td>N</td><td>There is no barcode on Read 2 of a pair, in this</td></tr> </table>	5	Barcode is in 5' end of read 1	3	Barcode is in 3' end of read 2	H	Barcode is in Header record using Illumina Casava (pre V1.8) pipeline format.	N	There is no barcode on Read 1 of a pair, in this case read 2 must have barcode on 5' end.	5	Barcode is in 5' end of read 2	N	There is no barcode on Read 2 of a pair, in this
5	Barcode is in 5' end of read 1												
3	Barcode is in 3' end of read 2												
H	Barcode is in Header record using Illumina Casava (pre V1.8) pipeline format.												
N	There is no barcode on Read 1 of a pair, in this case read 2 must have barcode on 5' end.												
5	Barcode is in 5' end of read 2												
N	There is no barcode on Read 2 of a pair, in this												

	<p>case read 2 must have barcode on 5' end.</p> <p>Examples:</p> <p>Format 5 5 Paired end with tag on 5' end of both reads</p> <p>Format 3 Single end where tag is on 3' end of the first read 1</p> <p>Format H Single or paired end with barcode in the sequence header</p> <p>Format N 5 Paired end with barcode on 5' end of read2</p> <p>Format 5 Single or paired end with tag on 5' end of read 1.</p>
Tag Line	<p>The Distance and Format records are followed by tag lines. Each tag line has two fields, a tag identifier and the tag sequence.</p> <p>Example:</p> <pre> 1 ATCACG 2 CGATGT 3 TTAGGC 4 TGACCA 5 ACAGTG 6 GCCAAT 7 CAGATC 8 ACTTGA 9 GATCAG 10 TAGCTT 11 GGCTAC 12 CTTGTA </pre>

Fields in above records should be separated by a single white space character.

Command Line Options

Usage:

novobarcodes options

Options:

-help Print command line usage options.

-b <i>tagfile</i>	Required. Specifies the bar code tag file as described above.																						
-d <i>folder</i>	Sets the folder name for demux'd read files. Default is current folder.																						
-f <i>file1</i> [<i>file2</i>]	Required. Specifies read file(s). Two filenames if paired end. : FASTA, PRB, FASTQ, Solexa FASTQ files are supported.																						
-F <i>format</i> [<i>option</i>]	Specifies the <i>format</i> of the read file. Normally Novobarcoder can detect the format of read files and this option is not required. However starting with Illumina pipeline version 1.3 the scale for quality values has been changed. If you are using the new format Illumina *_sequence.txt files with index tags embedded in the reads you need to add the option '-F ILMFQ' to ensure correct interpretation of quality values. Other values for the -F option are: <table> <tr> <td>FA</td><td>Fasta format read files with no qualities.</td></tr> <tr> <td>SLXFQ</td><td>Fastq format with Solexa style quality values. $10\log_{10}(P/(1-P)) + '@'$</td></tr> <tr> <td>STDFQ</td><td>Fastq format with Sanger coding of quality values. $-10\log_{10}(Perr) + '!''$</td></tr> <tr> <td>ILMFQ</td><td>Fastq with Illumina coding of quality values. $-10\log_{10}(Perr) + '@'$</td></tr> <tr> <td>PRB</td><td>Illumina _prb.txt format.</td></tr> <tr> <td>PRBnSEQ</td><td>Illumina _prb.txt with _seq.txt files.</td></tr> <tr> <td>QSEQ</td><td>Illumina QSEQ files</td></tr> <tr> <td>ILM1.8</td><td>Illumina Casava V1.8 fastq files with Sanger coding of quality values. $-10\log_{10}(Perr) + '!''$.</td></tr> </table> <p>The [<i>option</i>] applies to QSEQ and ILM1.8 format files and species how reads flagged as low quality by Illumina base caller will be processed.</p> <table> <tr> <td>--ILQ_SKIP</td><td>Flagged reads are not classified (i.e are written to the NC folder)</td></tr> <tr> <td>--ILQ_USE</td><td>Flag is ignored and reads are treated as per any other read.</td></tr> <tr> <td>--ILQ_QC</td><td>Same as ILQ_SKIP</td></tr> </table>	FA	Fasta format read files with no qualities.	SLXFQ	Fastq format with Solexa style quality values. $10\log_{10}(P/(1-P)) + '@'$	STDFQ	Fastq format with Sanger coding of quality values. $-10\log_{10}(Perr) + '!''$	ILMFQ	Fastq with Illumina coding of quality values. $-10\log_{10}(Perr) + '@'$	PRB	Illumina _prb.txt format.	PRBnSEQ	Illumina _prb.txt with _seq.txt files.	QSEQ	Illumina QSEQ files	ILM1.8	Illumina Casava V1.8 fastq files with Sanger coding of quality values. $-10\log_{10}(Perr) + '!''$.	--ILQ_SKIP	Flagged reads are not classified (i.e are written to the NC folder)	--ILQ_USE	Flag is ignored and reads are treated as per any other read.	--ILQ_QC	Same as ILQ_SKIP
FA	Fasta format read files with no qualities.																						
SLXFQ	Fastq format with Solexa style quality values. $10\log_{10}(P/(1-P)) + '@'$																						
STDFQ	Fastq format with Sanger coding of quality values. $-10\log_{10}(Perr) + '!''$																						
ILMFQ	Fastq with Illumina coding of quality values. $-10\log_{10}(Perr) + '@'$																						
PRB	Illumina _prb.txt format.																						
PRBnSEQ	Illumina _prb.txt with _seq.txt files.																						
QSEQ	Illumina QSEQ files																						
ILM1.8	Illumina Casava V1.8 fastq files with Sanger coding of quality values. $-10\log_{10}(Perr) + '!''$.																						
--ILQ_SKIP	Flagged reads are not classified (i.e are written to the NC folder)																						
--ILQ_USE	Flag is ignored and reads are treated as per any other read.																						
--ILQ_QC	Same as ILQ_SKIP																						
--GZIP	If this option is set the demuxed read files will be gzip'd and be given a .gz file extension.																						
-i <i>qseqtagfile</i>	If the index tags for the reads are in a separate QSEQ format file (rather than embedded in 5' or 3' of read) then it should be specified here. Use of -i option is only available if all the read files are in QSEQ format.																						

In this case demuxed files will be written in QSEQ format and adapter trimming will not function.

- QSEQ_OUT** Forces output in QSEQ format when input is QSEQ and tags are in 5' or 3' of a read. Without this option the reads are written in fastq format.
- l 9** Sets index/barcode read length if shorter than the tag length. It is used to control how many bases are trimmed off the read.
- t 99** Specifies a minimum tag alignment score difference (quality) between best tag and next best tag. Alignment scores are calculated using base qualities as per novoalign. A mismatch at a high quality base (phred quality > 30) will score 30.
If the quality, or score difference, is less than this reads are written to the catch all file.
Default is 30 * Distance/2
- a [adapter sequence]** Designed for use with 3' index tags on single end reads, this option allows DNA fragments that are shorter than the read length. Both the tag and adapter are trimmed from the read.
The -a option can also be used with 5' tags and with tags in the Illumina read header.
At the moment it does not work in conjunction with the -i option.
- NC_OFF** Turns off creation of NC folder and the writing of the unclassified reads. This is useful if you are extracting a single index tag from a set of reads and have no interest in the unclassified reads..

Example:

```
novobarcodes -b tags.txt -f s_1_0001_prb.txt -t30 -l 7
```

This will read the tag file, create a folder and file for each tag and then classify the reads. A tag alignment quality of 30 is required. Quality is calculated as in Novoalign and a value of 30 corresponds to approx 1 mismatch or probability of 0.999 that tag aligned is correct. (I.e. the next best tag has at least one more mismatch)

The -l 7 option specifies that the sequence read for the tag is 7 bp long. The actual tag sequences may be shorter.

Output:

```
novobarcodes -b tags.txt -f s_4_0050_prb.txt -t 30 -l 7
```

ID	Tag	Count
5	ACAGTG	8388
8	ACTTGA	10317
1	ATCACG	8050
7	CAGATC	44278
2	CGATGT	8487
12	CTTGTA	8062
9	GATCAG	11
6	GCCAAT	6194
11	GGCTAC	12369
10	TAGCTT	0
4	TGACCA	5082
3	TTAGGC	3266
NC	NC	4537

A simple count of matching reads to each tag is produced.

Output pathnames for demultiplexed reads are created by pre pending the tag sequence as a folder name to the input read filename. So for command such as novobarcode -f <filename> the classified reads will be written to files <tag>/<filename> and unclassified reads to NC/<filename>

When a folder name is specified using the -d <folder> option then classified reads are written to files <folder>/<tag>/<filename>

For Solexa *_prb.txt read files, Novobarcode will look for a corresponding *_seq.txt file in the same folder as the prb file. If found the _seq.txt records will be classified along with the prb records.

For FASTA format read files Novobarcode will look for a corresponding Phred quality file and, if found, the quality values will be used during tag alignment and the quality records will be classified along with the fasta sequences.

Performance

Tests on Mac OS 10.5 and Linux x86_64 for classifying 119,041 length 43 reads are shown below:

Operating System	Elapsed Time (seconds)	%CPU	No. Threads
------------------	---------------------------	------	-------------

Linux x86_64	5.94	99%	1
Mac OS 10.5	6.73	86%	1

Please send feedback to support@novocraft.com

Classification Process

When there is more than one tag in the tag file Novobarcodes will assign reads to the tag with the best alignment, in this case it's important that all tags used in the experiment are included in the tag file.

1. Tag sequence from read is checked against all tags for an exact match. If an exact match is found then assignment is made and quality is given as $30 \times \text{distance}$
2. Else tag sequence from the read is aligned against each tag using ungapped alignment with match/mismatch scoring based on qualities as in Novoalign. The read is assigned to the tag with the lowest alignment score and given a quality equal to difference between the lowest score and next lowest score.
3. For paired end with tag in both reads both reads must align to the same tag or one read of the pair is not classified.
4. If the quality of the classification is above the threshold (-t option) the tag is trimmed from the read and the read output to the appropriate tag file else it is written to the NC folder.

When there is a single tag in the tag file we assume that you are extracting just this tag from a file of reads that includes other tags.

1. The tag sequence from the read is checked against the tag for an exact match. If found then assignment is made and a quality of $30 \times \text{distance}$ is given
2. Else the tag sequence from the read is aligned against the tag using ungapped alignment with match/mismatch scoring based on qualities as in Novoalign. Alignment quality is calculated as $30 \times \text{Distance} - 2 \times \text{Score}$. In addition read will not be classified if the number of mismatches is greater than $\text{Distance}/2$
3. For a paired end with tag in both reads the classifications must be to same tag or one read must have failed classification and the other passed.
4. If quality of classification is above the threshold (-t option) the tag is trimmed from the read and the read output to the appropriate tag file.

Appendix

Tag File Example

```
Distance 3
Format 5 5
1 ATCACG
2 CGATGT
3 TTAGGC
4 TGACCA
5 ACAGTG
6 GCCAAT
7 CAGATC
8 ACTTGA
9 GATCAG
10 TAGCTT
11 GGCTAC
12 CTTGTA
```

Read File Formats

Illumina Cassava

The index tag is in the header line. This format is specified using the 'H' option on the **Format** line of the tag file.

```
@HWI-EAS334:8:1:0:281#TCCCTT/1
TGCAGGAGGTGCTTACACATGTTTGTTCCTTCGCTGCCGTCTTCC
+HWI-EAS334:8:1:0:281#TCCCTT/1
abbbbabbbb`bbbbbbbbbbbbabbbbabbbbbbbbababb`babba
@HWI-EAS334:8:1:0:248#ATACGT/1
GCTCCCCGCGTGGCCCTGCACCAGCAGCTCCTACACCGCGCGGCC
+HWI-EAS334:8:1:0:248#ATACGT/1
_aaaaaaaaaa`_aaaaaaaaaaaaaaaaaaaaaaa`aaa`aaa`
```

FASTQ with tag on 5' end of read.

```
@HWI-EAS334:8:1:0:281/1
TCCCTTTGCAGGAGGTGCTTACACATGTTTGTTTCTTCGCTGCCGTCTTCC
+HWI-EAS334:8:1:0:281/1
IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
@HWI-EAS334:8:1:0:248/1
ATACGTGCTCCCCGCGTGGCCCCTGCACCAGCAGCTCCTACACCGGCGGCC
+HWI-EAS334:8:1:0:248/1
IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
```



Disclaimer

THE SOFTWARE IS PROVIDED "AS IS." NOVOCRAFT TECHNOLOGIES SDN BHD DISCLAIMS ALL OTHER WARRANTIES, WHETHER EXPRESS, IMPLIED, OR STATUTORY, REGARDING THE SOFTWARE AND THE DOCUMENTATION, INCLUDING ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE, AND NON INFRINGEMENT OF THIRDPARTY RIGHTS.

YOU ACKNOWLEDGE THAT THE SOFTWARE AND ITS STRUCTURE, ORGANIZATION, AND SOURCE CODE CONTAIN VALUABLE TRADE SECRETS OF NOVOCRAFT. ACCORDINGLY, YOU AGREE NOT TO (A) SUBLICENSE, LEASE, RENT, LOAN, OR OTHERWISE TRANSFER THE SOFTWARE TO ANY THIRD PARTY; (B) REVERSE ENGINEER, DECOMPILE, DISASSEMBLE, OR OTHERWISE ATTEMPT TO DERIVE THE SOURCE CODE FOR THE SOFTWARE; (C) USE THE SOFTWARE IN ANY SERVICE BUREAU OR TIME SHARING ARRANGEMENT WITHOUT EXPRESS PERMISSION OF NOVOCRAFT TECHNOLOGIES SDN BHD.