Institute of Medical Genetics and Applied Genomics

SeqPurge: highly-sensitive adapter trimming for paired-end short read data

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

Marc Sturm[1*], Christopher Schroeder[1], Peter Bauer[1]

[1] Institute of Medical Genetics and Applied Genomics, University of Tuebingen, Germany.
*marc.sturm@med.uni-tuebingen.de

Abstract

Trimming adapter sequences from short read data is a common preprocessing step in most DNA/RNA sequence analysis pipelines. For amplicon-based approaches, which are mostly used in clinical diagnostics, sensitive adapter trimming is of special importance. Untrimmed adapters can be located at the same genomic position and can lead to spurious variant calls. Shotgun approaches are more robust towards adapter contamination because untrimmed adapters are randomly distributed over the target region. This reduces the probability of spurious variant calls.

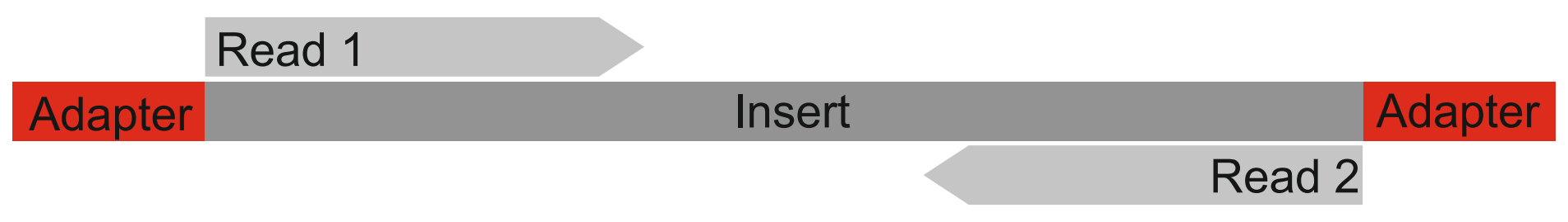
When performing paired-end sequencing, the overlap between forward and reverse read can be used to identify excess adapter sequences. This is exploited by several published adapter trimming tools. However, in our evaluations on amplicon-based paired-end data we found that these tools fail to remove all adapter sequences and that adapter contamination leads to spurious variant calls.

Here we present SeqPurge, a highly-sensitive adapter trimmer that uses a probabilistic approach to detect the overlap between forward and reverse reads of paired-end Illumina sequencing data. The overlap information is then used to remove adapter sequences, even if only one base long. Compared to other adapter trimmers specifically designed for paired-end data, we found that SeqPurge achieves a higher sensitivity. The number of remaining adapters after trimming is significantly reduced compared to other tools. The specificity of SeqPurge is comparable to that of the competing tools. In addition to adapter trimming, SeqPurge also offers quality-based trimming, trimming of no-call (N) stretches, raw read quality-control and error-correction. SeqPurge is available at https://github.com/imgag/ngs-bits

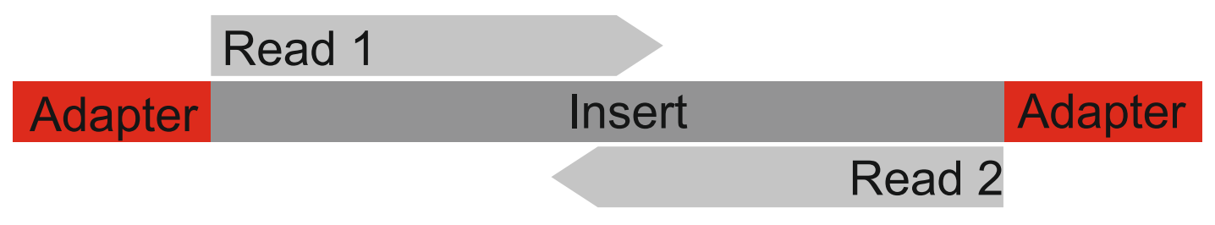
(1) What is adapter contamination

Adapter sequences are present in the read if the insert length is smaller than the read length (case c).

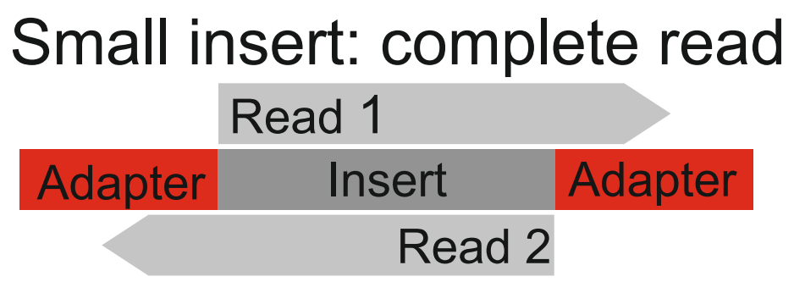
(a) Large insert: no overlap, no adapter contamination



(b) Medium insert: partial read overlap, no adapter contamination

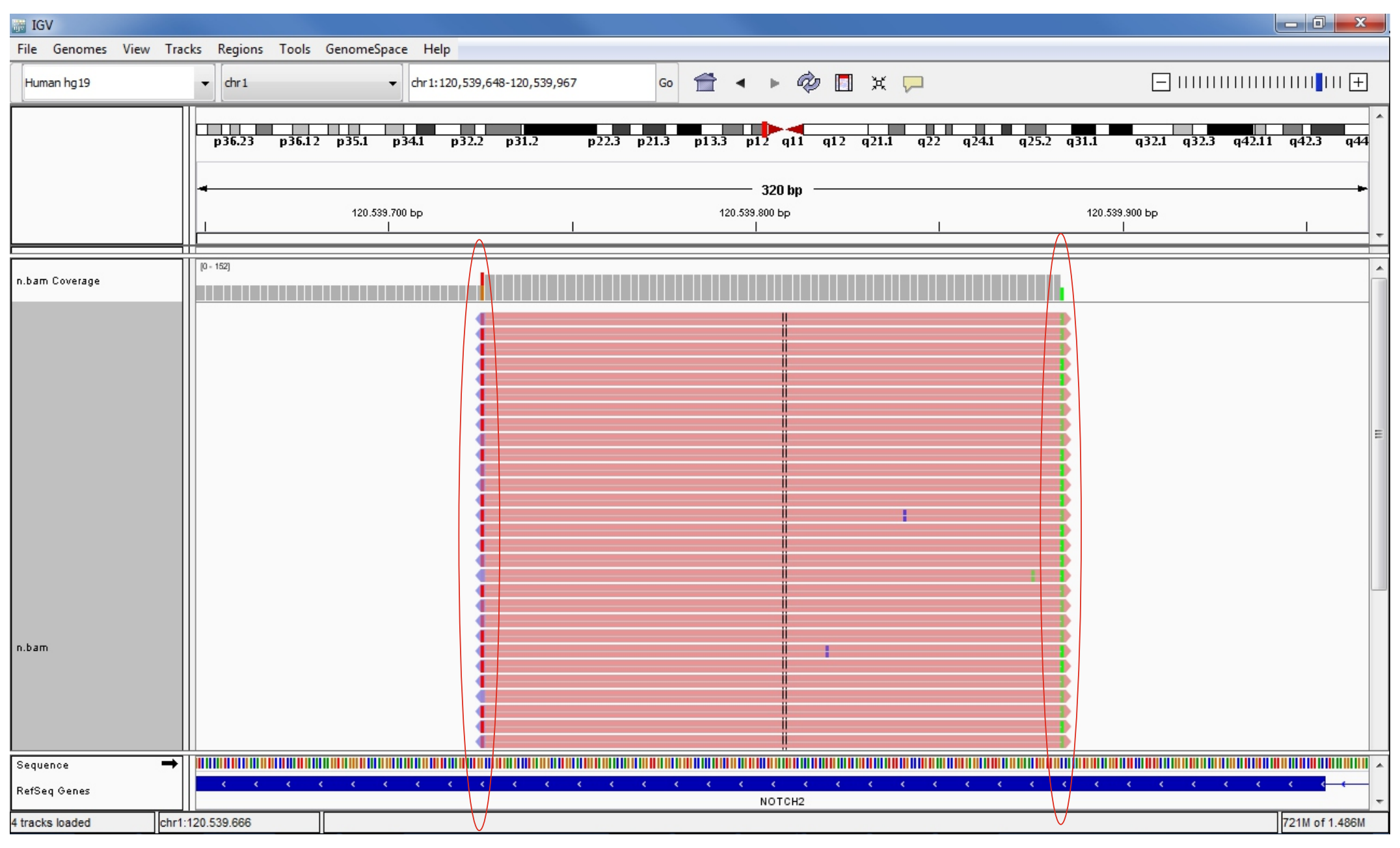


(c) Small insert: complete read overlap, adapter contamination



(2) How does is cause FP variant

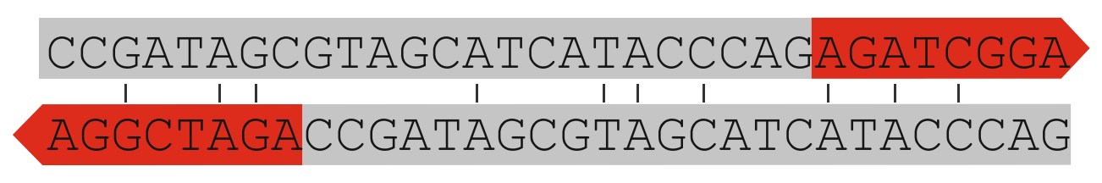
Untrimmed adapters lead to spurious variant calls in amplicon sequencing, especially when the only few adapter bases are present in the read:



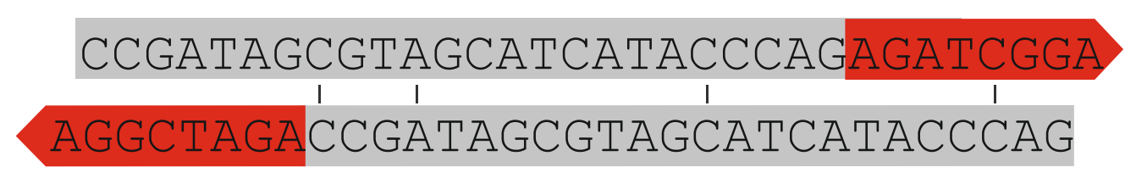
(3) Our algorithm for paired-end data

Insert matches of read 1 and read 2 are detected by shifting the reads against each other, looking at all possible offsets:

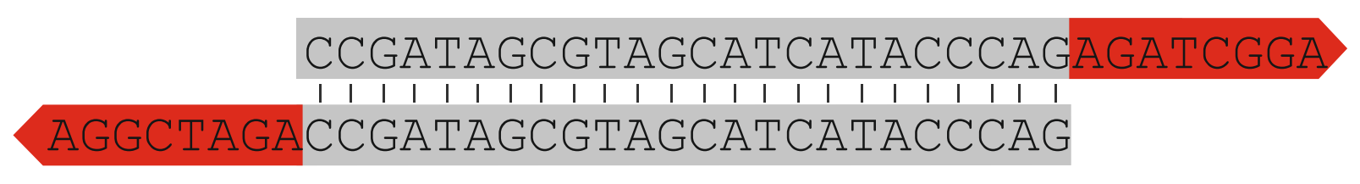
(a) offset 0: 10 matching bases, 22 mismatching bases


CCGATAGCGTAGCATCATACCCAGAGATCGGA
AGGCTAGACCGATAGCGTAGCATCATACCCAG

(b) offset 1: 4 matching bases, 27 mismatching bases


CCGATAGCGTAGCATCATACCCAGAGATCGGA
AGGCTAGACCGATAGCGTAGCATCATACCCAG

(c) offset 8: 24 matching bases, 0 mismatching bases


CCGATAGCGTAGCATCATACCCAGAGATCGGA
AGGCTAGACCGATAGCGTAGCATCATACCCAG

For each offset, the probability that the number of matches occurs by chance is calculated. If the probability is below a given threshold, an insert match was detected. In addition to this basic algorithm, special handling of repeat regions and reads with excessive errors is done.

(4) Performance evaluation

To evaluate the performance of trimming tools, we created an amplicon library with a HaloPlex custom kit (hereditary breast and ovarian cancer) using DNA from the HapMap reference sample NA12878. The library was sequenced on an Illumina MiSeq in 158 bases paired-end mode. Reads were mapped to the hg19 reference genome using the „BWA mem" algorithm. Variants were called using „freebayes" (results for „samtools" or „GATK" are similar). Reads of length 15 or less after trimming were discarded. All tools were configured to trim adapters only, using default parameters. No quality-based trimming was performed. The following tables show the performance and resource benchmarks:

	trimming			mapping			variant calling
	bases remaining	adapter 20mers remaining	reads remaining	reads paired	bases overtrimmed	bases undertrimmed	#variants
no trimming	168509790	414254	1059810	1022432	0	21920176	178
SeqPurge 0.1-319	**142570414**	**0**	**1037070**	**1021315**	**1732**	**33664**	**156**
AdapterRemoval 1.5.4	142864048	1450	1037444	1021336	25	224132	156
Flexbar 2.5	142323263	7	1036962	1021270	4540004	62970	175
PEAT 1.2.2	146433832	24298	1059810	1022947	69298	354531	155
SeqPrep 1.2	142069127	0	1033052	1018828	53	34949	156
Skewer 0.1.123	142664746	425	1037026	1021369	240	95279	157
Trimmomatic 0.32	142881154	824	1037474	1021427	1144	244456	179

	trimming		mapping	variant calling
	time [s]	memory [MB]	time [s]	time [s]
no trimming	n/a	n/a	327	71
SeqPurge 0.1-319	**41**	**7.4**	**252**	**69**
AdapterRemoval 1.5.4	431	3.1	280	70
Flexbar 2.5	152	3.1	271	67
PEAT 1.2.2	47	168.3	272	66
SeqPrep 1.2	297	1.5	253	68
Skewer 0.1.123	28	1.4	269	67
Trimmomatic 0.32	95	60.0	283	70

The error-tolerance of adapter trimmers cannot be easily tested on real data, thus we simulated reads using out simulator PERsim. Five million read pairs for the coding region of the genome (CCDS) were simualted with varying error rates of 0 to 4%. The 100 bp read pairs were created based on a theoretical library with mean insert size of 100 bp and a standard deviation of 50 bp. This results in a dataset where 50% of the reads contain adapter contamination and need to be trimmed. The trimming results are shown here:

	bases overtrimmed					bases undertrimmed				
	0% error	0.5% error	1% error	2% error	4% error	0% error	0.5% error	1% error	2% error	4% error
SeqPurge 0.1-319	**4226**	**8274**	**7682**	**6070**	**6192**	**0**	**8**	**88**	**346**	**6316**
AdapterRemoval 1.5.4	386	332	218	152	28	0	0	220	44324	6245484
Flexbar 2.5	2838601	2840129	2841875	2851984	2872533	223310	226897	232141	244677	282048
PEAT 1.2.2	2721166	2603976	2575188	2454360	2225584	70706866	70887130	70934088	70823364	71010930
SeqPrep 1.2	1546	1708	1122	933	1437	0	864906	2034199	3364496	4254094
Skewer 0.1.123	20	18	8	14	12	0	2887192	9351780	36053692	109590338
Trimmomatic 0.32	16	0	0	0	0	2229820	2240634	2399286	4316106	25884992

The benchmark results above clearly demonstrate that proper adapter trimming reduces the runtime of the data analysis and removes noise and, thus, spurious variant calls.
However, the results vary depending on the tool used for adapter trimming:
(1) SeqPurge has the highest sensitivity while maintaining a good specificity. It also has highest error-tolerance and is among the fastest tools.
(2) AdapterRemoval shows a decent performance, but is not very sensitive - many untrimmed bases remain in the data. It is too slow for routine application in a high-throughput setting.
(3) Flexbar has a low sensitivity and specificity and fails to trim short adapter remains (<3 bases), which leads to spurious variant calls. It is too slow for routine application in a high-throughput setting.
(4) PEAT has a low sensitivity and specificity, fails to remove reads without insert and has a high memory usage.
(5) SeqPrep is not specific enough - it removed many reads that all other tools do not trim (see paper). It is too slow for routine application in a high-throughput setting.
(6) Skewer shows a good performance both in terms of trimming and speed, but is not very error-tolerant.
(7) Trimmomatic fails to remove short (<8 bases) adapter residues from the data, which leads to spurious variant calls. Additionally, it has a high memory usage.

Quality-based trimming:

All benchmarks above were performed without quality-based trimming. A benchmark of SeqPurge and Skewer with quality-based trimming showed that moderate quality trimming (<Q20) silightly improves the variant quality. Increasing the quality cutoff did not improve the results any further.

(5) Conclusion

(1) Amplicon data
Adapter trimming is a crucial step in the data analysis of amplicon-based short-read data. It significantly reduces the number of spurious variant calls. In our comparison of adapter trimming tools designed specifically for paired-end reads, SeqPurge is the most sensitive tool. Thus, it is most suited for amplicon data.

(2) Shotgun data
For shotgun data, sensitivity of adapter trimming is not as crucial because adapter residues in different reads are generally not placed at the sample genomic position. Here, Skewer is a viable alternative to SeqPurge, when the data has a low error rate. All other tools in the comparison are either too slow or show a worse performance than SeqPrep and Skewer.

(3) Runtime
Adapter removal improves the overall runtime of data analysis, because mapping can be done more efficiently when less adapters are present in the data. In this comparison we only used „BWA mem". When using more sophisticated read mappers (e.g. „stampy") or indel-realignment tools (e.g. „GATK" or „ABRA"), the beneficial effect of adapter trimming on the overall runtime can be expected to be even greater.

(6) Availability

SeqPurge is available under the „GPL Version 2" license as part to the **ngs-bits** project from GitHub:
https://github.com/imgag/ngs-bits

A paper with more details was published recently in BMC Bioinformatics:
http://www.ncbi.nlm.nih.gov/pubmed/27161244

References:

ABRA	https://github.com/mozack/abra
AdapterRemoval	https://github.com/MikkelSchubert/adapterremoval
Flexbar	https://sourceforge.net/projects/flexbar/
Freebayes	https://github.com/ekg/freebayes
GATK	https://www.broadinstitute.org/gatk/
PEAT	https://github.com/jhhung/PEAT
samtools	http://www.htslib.org/
SeqPrep	https://github.com/jstjohn/SeqPrep
Skewer	http://sourceforge.net/projects/skewer/
Stampy	http://www.well.ox.ac.uk/project-stampy
Trimmomatic	http://www.usadellab.org/cms/?page=trimmomatic