RESEARCH ARTICLE
JIVA 22 (3) December 2024
https://doi.org/10.55296/JIVA/22.3.2024.88-93

Received on 27.09.2024
Accepted on 16.10.2024

EVALUATION OF THE EFFICIENCY AND SPEED OF POLY-G TAIL TRIMMING BY DIFFERENT TOOLS FROM NEXT-GENERATION SEQUENCING DATA

R. Valarmathi[1], S. Pramod[2*], Marykutty Thomas[3], T.V. Aravindakshan[4], Justin Davis[5], A. Prasad[6] and M.K. Muhammad Aslam[7]

[1]M.V. Sc Scholar, [3]Assistant Professor, [4] Senior Professor and Head,
Department of Animal Genetics and Breeding, College of Veterinary and Animal Sciences,
Mannuthy, Thrissur -680 651,
[2]Assistant Professor, [6]Associate Professor and Head, Livestock Research Station,
Thiruvazhamkunnu, Palakkad -678 601,
[5] Associate Professor, Department of Livestock Production Management, College of
Veterinary and Animal Sciences, Pookode, Wayanad -678 642,
[7]Assistant Professor, Base Farm, Kolahalamedu, Idukki -685 501
Kerala Veterinary & Animal Sciences University, Pookode

*Corresponding author: pramod.s@kvasu.ac.in

ABSTRACT

Next-generation sequencing (NGS) using two-dye chemistry has reduced DNA sequencing costs but introduced challenges, such as overrepresented poly-guanine (poly-G) tails, especially in reverse strands. Poly-G artifacts often result in erroneous high-confidence G bases at the ends of reads, complicating downstream analyses. This study evaluated the efficiency and speed of three popular trimming tools *viz*. BBDuk, Cutadapt, and Fastp in removing poly-G artifacts from NGS datasets. A sample dataset generated using the Illumina NovaSeq 6000 platform from crossbred cattle with 26.32 million reads and 6.79 per cent poly-G content was used for the study. Quality was assessed with FastQC, and trimming was performed using BBDuk, Fastp, and Cutadapt. Post-trimming, datasets were re-evaluated using FastQC and metrics like poor quality sequences, GC content, and trimming time were recorded. Results indicated that the tool BBDuk was the fastest (8.42 seconds), followed by Fastp (9.50 seconds) and Cutadapt (24.42 seconds). All the tools efficiently trimmed poly-G tails, with BBDuk and Cutadapt retaining more sequences than Fastp.

Keywords: Poly G-tail trimming, BBDuk, Cutadapt, Fastp

INTRODUCTION

Next-generation sequencing using two-dye chemistry has significantly reduced sequencing costs, but it also introduced challenges, like the over representation of poly-guanine (poly-G) tails, especially in reverse strands (Chen *et al*., 2018). These poly-G artifacts occur in two-channel sequencing systems when the dark base 'G' is incorrectly called after synthesis termination. It accumulates erroneous high-confidence calls of G bases in the ends of affected reads. In contaminated samples, numerous affected reads can map to reference regions with high G content, complicating downstream processing.

Poly-G tails, the stretches of guanine nucleotides often found at the ends of DNA or RNA sequences, are common artifacts in NGS platforms, notably Illumina systems which use two-colour chemistry. Trimming these poly-G tails from sequencing reads is crucial for improving the accuracy and reliability of results. This step enhances read alignment to reference genomes and reduces bias in downstream analyses. Quality control and pre-processing of sequence data could be considered as resolved, given the availability of several relevant tools. For instance, Cutadapt (Martin, 2011) is a commonly used adapter trimmer, which also provides some read-filtering features. Another one is fastp which is a C++ based tool for quality control and adapter trimming (Chen *et al*., 2018). Removal of unwanted sequences can also be done by BBduk, a member of BBTools package (Singer *et al*., 2016). In this study, we compared the efficiency in terms of speed and accuracy of three trimming tools *viz*; BBDuk, Cutadapt, and Fastp in removing poly-G artifacts. By evaluating these tools, we aim to identify the most effective approach for removing poly-G tails from NGS datasets.

MATERIALS AND METHODS

A next-generation sequencing dataset obtained from Illumina NovaSeq 6000 platform from a crossbred cattle DNA sample, generated after a double digest restriction-site associated DNA (ddRAD) experiment was used for the study. The quality of the reads was assessed with FastQC (Andrews, 2010). The raw dataset contained 26.32 million sequences, with a total size of 418.4 MB and had a high poly-G content of 6.79%.

Poly-G tail trimming was performed using BBDuk (Singer *et al*., 2016), fastp (Chen *et al*., 2018), and Cutadapt (Martin, 2011). The tools were run on a desktop computer with Windows 11 equipped with an Intel i5 processor and 8 GB RAM. The Windows Subsystem for Linux (WSL) was utilized to run the aforementioned software. The specific commands used for each program are provided in Table 1.

Table 1: Commands used to run the programmes

Sl. No	Name of the tool	Command used
1	BBDuk	bbduk.sh in=N1_R2.fq out=TrimN1_R2.fq ref=polyG.fa ktrim=r k=10 mink=5 hdist=1 tpe tbo
2	Cutadapt	cutadapt -a "G{10}" -o output.fq N1_R2.fq
3	fastp	fastp --in1 N1_R2.fq --out1 N1_R2fastp.fq --trim_poly_g

The tools were run four times and the time taken for the task was noted, Post-trimming, the quality of datasets were re-evaluated using FastQC (Chetwynd *et al*., 2023), a Java-based tool for assessing the quality of NGS data. Metrics such as sequences flagged as poor quality, GC content, and the time taken for trimming were recorded for each of the three tools. Statistical analysis was performed using ANOVA and Tukey's test in SPSS version 24.

RESULTS AND DISCUSSION

Differences were noted in the time required to complete the trimming process between runs of all the tools. The characteristics of the raw data along with the performance metrics for BBDuk, Fastp, and Cutadapt are summarised in Table 2. On an average, the tool BBDuk processed the data in 8.42 seconds, while Fastp took 9.50 seconds. Cutadapt was the slowest and required 24.42 seconds to complete the task. This variation could be due to changes in the background processes running on the computer. The outputs were consistent between runs with respect to other attributes studied (trimmed reads, deleted reads, total data retained, sequences flagged as poor quality and GC content).

Table 2: Performance of different tools in trimming poly G tails

Raw Data		Name of tools		
Attributes	Quantity	BBDuk	Fastp	Cutadapt
Total Sequences	26,32,074	26,32,072	23,81,306	2632074
Trimmed reads	-	432693	131	326856
Deleted reads	-	0	250,768	0
Total Bases	418.4 Mbp	381.1 Mbp	378.2 Mbp	384.7 Mbp
Sequences flagged as poor quality	-	0	90724	0
% GC	58	55	55	55
Time taken**	-	8.42±0.18^a s	9.50±0.29^a s	24.42±0.80^b s

**Means with different superscripts in a row differ significantly (p<0.01)

Despite the significant difference (p<0.01) in processing times, all three tools efficiently trimmed poly-G tails, as evident from the comparison charts. The FastQC reports before and after trimming for each tool are presented in Figures 1 to 4.

The GC content of the sequences decreased from 58% to 55%, post trimming in all the cases which indicated the effective trimming of poly-G artifacts. The tools BBDuk and Cutadapt removed poly-G tails more efficiently than Fastp, with minimal deletion of sequences. On the other hand, Cutadapt, although slower, also did not delete sequences and provided the highest retention of reads, making it a reliable option for preserving data integrity. The tool Fastp, in contrast, removed approximately 250,000 sequences but completed the poly-G trimming in just 9.50 seconds. This tool also removed short reads and those with low quality which might produce a dataset

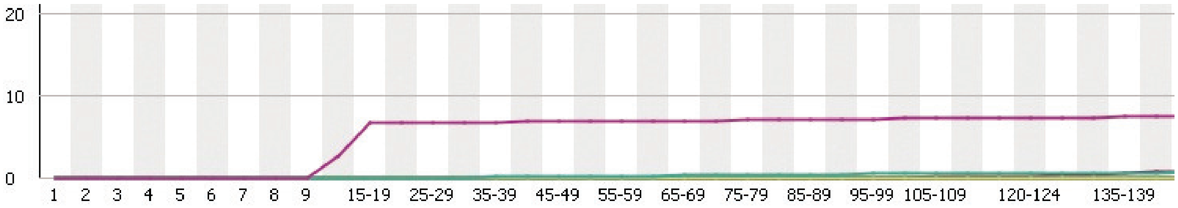

Fig 1: Adapter content before trimming

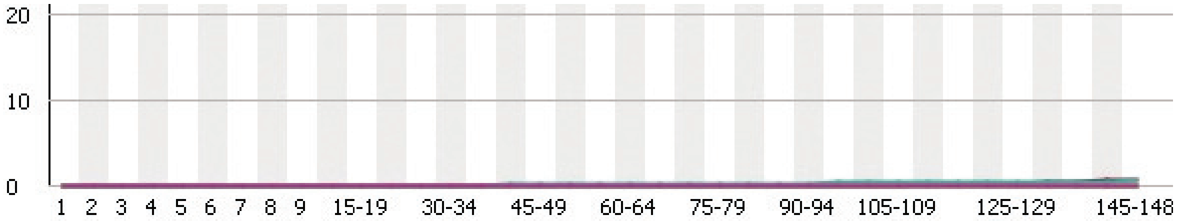

Fig 2: Trimmed with BBDuk

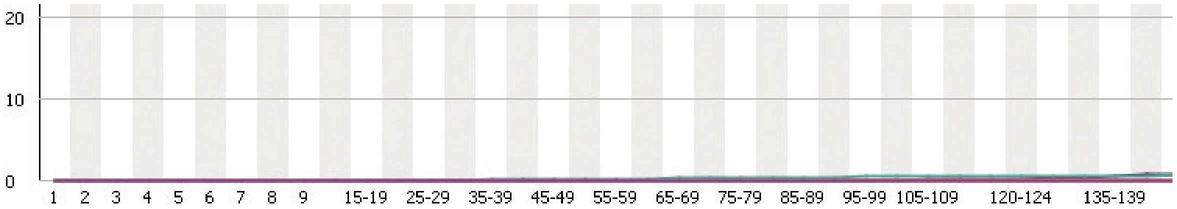

Fig 3: Trimmed with Cutadapt

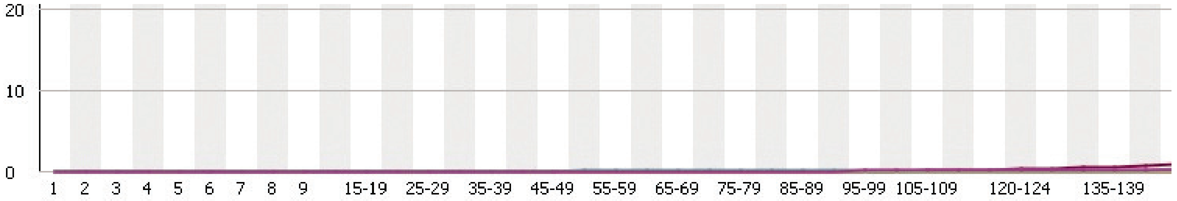

Fig 4: Trimmed with Fastp

X axis: Position in bp **Y Axis:** Poly g tail content (%)

more suitable for downstream processing by eliminating potential sources of error (Chen *et al*., 2018). Despite the deletion of some sequences, Fastp's speed and thoroughness in quality filtering makes it a valuable tool in scenarios where processing time and data quality are critical.

The tool Cutadapt emerged as the one which retained the maximum number of reads for downstream processes, maintaining data volume while reducing poly-G tails (Martin, 2011). This retention is crucial for applications requiring comprehensive data analysis, such as variant calling and genome assembly. However, the slower processing time of Cutadapt might be a limitation in high-throughput settings where speed is essential.

Each tool had its strengths and weaknesses. BBDuk and Fastp took significantly less time ($p<0.01$) than cutadapt for trimming poly-G tails. Fastp provided a balance between speed and data quality by removing low-quality reads, making it suitable for downstream analyses where read quality is of paramount importance. Cutadapt, though slower, excels in retaining the maximum number of reads, making it ideal for comprehensive data analysis. Future studies could focus on optimizing these tools to combine the best attributes of each, potentially developing a hybrid approach that maximizes efficiency, speed, and data retention (Singer *et al*., 2016; Chen *et al*., 2018). Furthermore, exploring the impact of these trimming tools on different types of sequencing data and experimental setups would provide deeper insights into their applicability and performance across diverse NGS applications.

SUMMARY

The study compared the performance of three popular tools *viz.* BBDuk, Cutadapt, and Fastp; focusing on their speed and efficiency in removing poly-G tails from NGS datasets. All three tools successfully eliminated poly-G tails, but their performance varied significantly. BBDuk was the fastest, while Fastp was particularly efficient in removing short and low-quality sequences. Although Cutadapt was the slowest, it effectively removed poly-G tails without deleting any sequences. In conclusion, both BBDuk and Fastp are effective for trimming large NGS datasets contaminated with poly-G tails. BBDuk's speed is advantageous for high-throughput environments, while Fastp's ability to enhance data quality by removing low-quality reads is beneficial for downstream applications. Despite being slower, Cutadapt is ideal for retaining the maximum number of reads and ensuring accurate removal of poly-G tails. Future research should focus on optimizing these tools or developing new hybrid approaches that combine their

strengths, offering a balanced solution that maximizes efficiency, speed, and data quality.

ACKNOWLEDGEMENT

The study was supported by the State Plan Project (2022-23) Augmenting Productivity and Conservation of Livestock.

REFERENCES

Andrews, S. 2010. FastQC: A Quality Control Tool for High Throughput Sequence Data [Online]. Available:http://www.bioinformatics.babraham.ac.uk/projects/fastqc/.

Chen, S. F., Zhou, Y. Q., Chen, Y. R. and Gu, J., 2018. fastp: An ultra-fast all-in-one FASTQ preprocessor. *Bioinformatics*. **34**: 884–890.

Chetwynd, S. A., Andrews, S., Inglesfield, S., Delon, C., Ktistakis, N. T. and Welch, H. C. E. 2023. Functions and mechanisms of the GPCR adaptor protein Norbin. *Biochem. Soc. Trans*. **51**: 1545-1558.

Martin, M., 2011. Cutadapt removes adapter sequences from high-throughput sequencing reads EMB. *Net*. *J*. **17**: 10-12.

Singer, E., Andreopoulos, B., Bowers, R. M., Lee, J., Deshpande, S., Chiniquy, J., Ciobanu, D., Klenk, H. P., Zane, M., Daum, C., Clum, A., Cheng, J. F., Copeland, A. and Woyke, T. 2016. Next generation sequencing data of a defined microbial mock community. *Sci Data*. **27**:160081.