








SOFTWARE TOOL ARTICLE

# REVISIED ICR142 Benchmarker: evaluating, optimising and benchmarking variant calling performance using the ICR142 NGS validation series [version 2; referees: 2 approved]

Previously titled: ICR142 Benchmarker: evaluating, optimising and benchmarking variant calling using the ICR142 NGS validation series

Elise Ruark <sup>1</sup>, Esty Holt <sup>1</sup>, Anthony Renwick<sup>1</sup>, Márton Münz<sup>1</sup>,  
Matthew Wakeling <sup>2</sup>, Sian Ellard <sup>2</sup>, Shazia Mahamdallie<sup>1</sup>, Shawn Yost<sup>1</sup>,  
Nazneen Rahman <sup>1,3</sup>

<sup>1</sup>Division of Genetics & Epidemiology, The Institute of Cancer Research, London, SM2 5NG, UK

<sup>2</sup>Institute of Biomedical and Clinical Science, University of Exeter Medical School, Exeter, EX2 5DW, UK

<sup>3</sup>Cancer Genetics Unit, Royal Marsden NHS Foundation Trust, London, SM2 5PT, UK

**v2** **First published:** 31 Aug 2018, 3:108 (<https://doi.org/10.12688/wellcomeopenres.14754.1>)  
**Latest published:** 31 Oct 2018, 3:108 (<https://doi.org/10.12688/wellcomeopenres.14754.2>)

## Abstract

Evaluating, optimising and benchmarking of next generation sequencing (NGS) variant calling performance are essential requirements for clinical, commercial and academic NGS pipelines. Such assessments should be performed in a consistent, transparent and reproducible fashion, using independently, orthogonally generated data.





Here we present ICR142 Benchmarker, a tool to generate outputs for assessing germline base substitution and indel calling performance using the ICR142 NGS validation series, a dataset of Illumina platform-based exome sequence data from 142 samples together with Sanger sequence data at 704 sites. ICR142 Benchmarker provides summary and detailed information on the sensitivity, specificity and false detection rates of variant callers. ICR142 Benchmarker also automatically generates a single page report highlighting key performance metrics and how performance compares to widely-used open-source tools.




We used ICR142 Benchmarker with VCF files outputted by GATK, OpEx and DeepVariant to create a benchmark for variant calling performance. This evaluation revealed pipeline-specific differences and shared challenges in variant calling, for example in detecting indels in short repeating sequence motifs. We next used ICR142 Benchmarker to perform regression testing with DeepVariant versions 0.5.2 and 0.6.1. This showed that v0.6.1 improves variant calling performance, but there was evidence of minor changes in indel calling behaviour that may benefit from attention. The data also allowed us to evaluate filters to optimise DeepVariant calling, and we recommend using 30 as the QUAL threshold for base substitution calls when using DeepVariant v0.6.1.

Finally, we used ICR142 Benchmarker with VCF files from two commercial

## Open Peer Review

Referee Status:  

	Invited Referees	
	1	2
<b>version 2</b> published 31 Oct 2018	 report	 report
<b>version 1</b> published 31 Aug 2018	 report	 report

- Oliver Hofmann** , University of Melbourne, Australia
- Birgit Sikkema-Raddatz** , University of Groningen, University Medical Center Groningen, The Netherlands  
**Lennart F. Johansson** , University of Groningen, University Medical Center Groningen, The Netherlands

## Discuss this article

Comments (0)

variant calling providers to facilitate optimisation of their in-house pipelines and to provide transparent benchmarking of their performance.

ICR142 Benchmark consistently and transparently analyses variant calling performance based on the ICR142 NGS validation series, using the standard VCF input and outputting informative metrics to enable user understanding of pipeline performance. ICR142 Benchmark is freely available at [https://github.com/RahmanTeamDevelopment/ICR142\\_Benchmark/releases](https://github.com/RahmanTeamDevelopment/ICR142_Benchmark/releases).

### Keywords

Variant calling, next generation sequencing, benchmarking, specificity, sensitivity, false detection rate, GATK, OpEx, DeepVariant



This article is included in the [Transforming Genetic Medicine Initiative \(TGMI\) gateway](#).

**Corresponding author:** Nazneen Rahman ([rahmanlab@icr.ac.uk](mailto:rahmanlab@icr.ac.uk))

**Author roles:** **Ruark E:** Conceptualization, Formal Analysis, Methodology, Software, Writing – Original Draft Preparation, Writing – Review & Editing; **Holt E:** Data Curation, Formal Analysis, Software; **Renwick A:** Formal Analysis; **Münz M:** Data Curation; **Wakeling M:** Formal Analysis; **Ellard S:** Formal Analysis; **Mahamdallie S:** Data Curation; **Yost S:** Data Curation, Formal Analysis; **Rahman N:** Conceptualization, Funding Acquisition, Writing – Original Draft Preparation, Writing – Review & Editing

**Competing interests:** NR is a Non-Executive Director of AstraZeneca. ER is a Product Manager for Foresite Capital.

**Grant information:** The work was supported by the Wellcome Trust [200990].

*The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.*

**Copyright:** © 2018 Ruark E *et al.* This is an open access article distributed under the terms of the [Creative Commons Attribution Licence](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

**How to cite this article:** Ruark E, Holt E, Renwick A *et al.* **ICR142 Benchmark: evaluating, optimising and benchmarking variant calling performance using the ICR142 NGS validation series [version 2; referees: 2 approved]** Wellcome Open Research 2018, 3:108 (<https://doi.org/10.12688/wellcomeopenres.14754.2>)

**First published:** 31 Aug 2018, 3:108 (<https://doi.org/10.12688/wellcomeopenres.14754.1>)

**REVISED Amendments from Version 1**

We have updated ICR142 Benchmarker so that it can now be used with hg19/GRCh37 or hg38/GRCh38 data. We have also made other minor changes to the paper to enhance clarity following helpful comments from the reviewers. This includes a slight change to the title to include the word 'performance'.

See referee reports

**Introduction**

Variant calling from next generation sequencing (NGS) data is a highly active area of bioinformatics, important to many clinical, commercial and academic applications. Several open-source tools are available and have been integrated into variant calling pipelines by many laboratories<sup>1-6</sup>. Commercial solutions and/or in-house proprietary tools are also increasingly being used by NGS analysis providers. Evaluations of pipeline performance are often based on internal data. This makes comparison, standardisation and regulation of NGS variant calling performance difficult<sup>7</sup>.

Assessment of variant calling performance is vital for improvement and optimisation of NGS variant calling. Comparative performance across pipelines is also of increasing importance, as the number of different analysis tools and providers continues to expand. The availability of benchmarking datasets with orthogonally confirmed positive and negative sites are required for optimal independent assessment of sensitivity, specificity and false detection rates (FDR).

We previously made available the ICR142 NGS validation series that includes NGS and Sanger data from 142 samples<sup>8</sup>. To construct the ICR142 NGS validation series we analysed exome sequence data from the 142 samples with multiple variant callers and undertook Sanger sequencing analysis at 704 sites to generate a dataset useful for systematic, transparent variant calling assessment and comparison<sup>8</sup>.

Here we present ICR142 Benchmarker<sup>9</sup>, a tool to generate outputs for assessing variant calling performance using the ICR142 NGS validation series. We used ICR142 Benchmarker with VCF files from GATK, OpEx and DeepVariant to provide guidance on expected variant caller performance compared to three open-source pipelines<sup>1,10,11</sup>. We then used ICR142 Benchmarker with VCF files from two commercial NGS variant calling providers, to provide comparison data to facilitate optimisation of their in-house pipelines, and to help give transparency of performance for their customers.

**Methods****ICR142 NGS validation series**

The ICR142 NGS validation series is a dataset that includes high-quality exome sequence data from 142 samples together with Sanger sequence data at 704 sites; 416 sites with variants and 288 sites at which variants were called by a variant caller, but no

variant is present in the corresponding Sanger sequence. The exome sequence data was generated using the Illumina TruSeq Exome and a HiSeq2000 sequencer. Full details of the ICR142 series are given in Ruark *et al.*<sup>8</sup>. In total, the ICR142 NGS validation series includes 704 sites, comprised of 123 base substitution variants, 293 insertions and/or deletion (indel) variants, 41 negative base substitution sites and 247 negative indel sites (Figure 1)<sup>8</sup>.

To determine if a variant was present we examined each Sanger sequence with Chromas software v2.13. For each site we selected an ENST from release 65 as the reference sequence. We analysed a region of interest of at least 100 base pairs (bp) of sequence flanking each variant site to allow for position/annotation errors.

We considered a base substitution to be confirmed if the correct variant was called at the exact position and the variant base signal was accompanied by a corresponding reduction in the reference base signal. We considered an indel variant to be confirmed if an indel variant was present in the region of interest and the indel variant allele signal was present along the complete length of the region of interest.

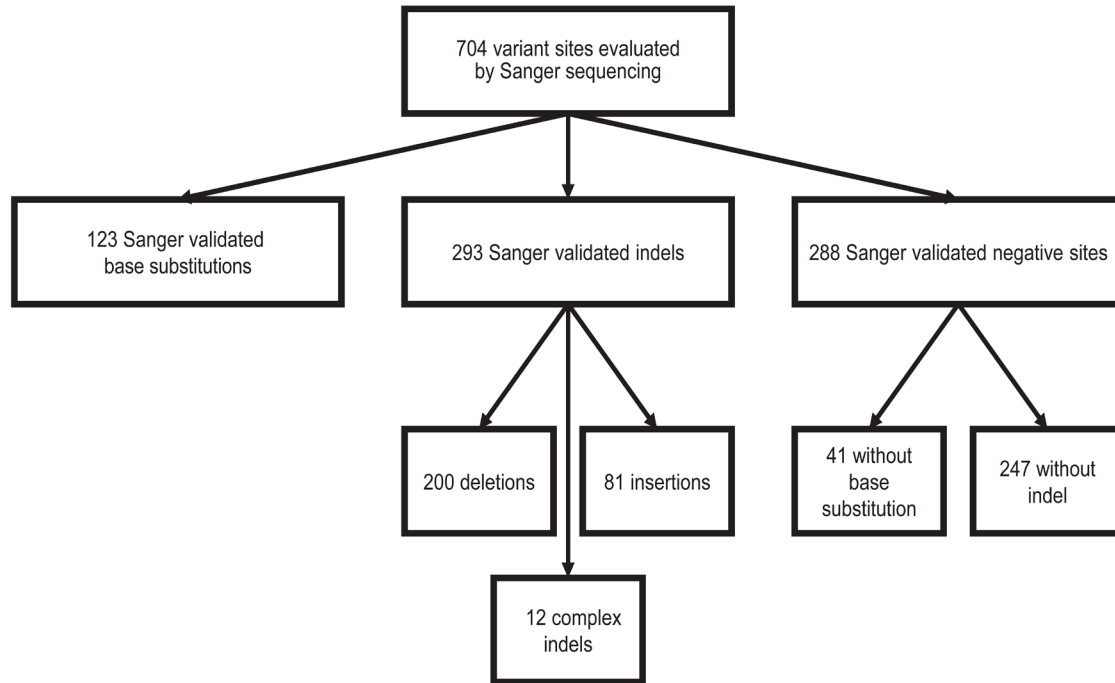
We considered a site negative for a base substitution if the exact base substitution was not present. We considered a site negative for an indel if no indel was detected in the 200bp region of interest.

**ICR142 Benchmarker**

**Implementation.** ICR142 Benchmarker<sup>9</sup> is implemented as an easy-to-use tool for assessing variant calling performance using the ICR142 NGS validation series. It can be used with hg19/GRCh37 or hg38/GRCh38 data. The tool includes an analysis script and three supporting files: the Sanger data file, the Report template file and the descriptive ColumnHeadings.txt file. ICR142 Benchmarker provides a series of informative metrics with increasing levels of detail from overall calling performance to per site profiles together with a one page report summarising both standalone performance and comparative performance against widely-used open-source pipelines. ICR142 Benchmarker is implemented in R and is publicly available at [https://github.com/RahmanTeamDevelopment/ICR142\\_Benchmarker/releases](https://github.com/RahmanTeamDevelopment/ICR142_Benchmarker/releases).

ICR142 Benchmarker requires an input file containing the paths to VCF version 4.X files. The VCF files must each represent a single sample. The script expects the ALT column to contain only one call. Any base substitution calls are expected to have REF and ALT values of length one, e.g. REF / ALT of GTCA / ATCA should be trimmed to G / A. Multi-sample VCF or gVCF files should be parsed to fulfil the above criteria.

At each site, ICR142 Benchmarker assesses both variant detection and accuracy of variant representation, with missing genotypes allowed. Base substitution variants are both detected and accurately represented if the correct variant is called at



**Figure 1. Layout of the 704 Sanger validated variant sites.** Breakdown of the 704 Sanger validated base substitutions, insertion and/or deletions (indels), and negative sites from 142 samples. The diagram shows the exact number of base substitutions, deletions, insertions, complex indels, and sites without a base substitution or indel.

the exact position. If an incorrect base substitution is called at that position it is considered a missed variant. For negative base substitution sites a false positive base substitution call is assigned if any base substitution call is made at the exact position. Due to the more complex nature of indel detection and representation, a stringent exact matching approach is not appropriate. We thus report indel detection as the number of indel calls within a 200bp window centred on the site position, for both true indel and negative indel sites. An indel variant is considered to be both detected and accurately represented if an exact match is found. For negative indel sites a false indel call is assigned if the indel detection value is greater than 0. Summary metrics are calculated from the detection values. Any missing values are treated as ‘no call’ in the metric calculations.

ICR142 Benchmark generates five output files, four tab-separated .txt files and one Word document .docx. The Summary.txt file provides summary performance metrics for the evaluated method, specifically the overall sensitivity, specificity and false detection rate (FDR) values. These three metrics are also separately calculated for base substitutions or indels. The FullResults.txt file contains all of the Sanger validation information from the ICR142 dataset and information on site-specific performance at each of the 704 sites. The FalsePositives.txt and TruePositives.txt files contain the relevant lines of the input VCF files for false positive and true positive variant calls, respectively. Detailed description of all columns in the .txt files is provided in the ColumnHeaders.txt supporting file.

The Report.docx file provides a summary variant calling analysis report of performance using the ICR142 dataset. This single page document is directly constructed from the Summary.txt and FullResults.txt files and thus is transparent and reproducible. Key points from the detailed outputs are highlighted to the user, including information about performance compared to widely-used open-source variant callers.

**Operation.** ICR142 Benchmark can be installed by running a simple Bash script. Installation requires R version 3.1.2 or later and a capacity to build packages from source. ICR142 Benchmark implements full version control using packrat<sup>12</sup>. This approach ensures ICR142 Benchmark implementation will not be affected by future changes of incorporated packages or their dependencies. Once installed, the tool can be run from a Linux/Unix command line. The ICR142 Benchmark documentation is available at GitHub: [https://github.com/Rahman-TeamDevelopment/ICR142\\_Benchmark/](https://github.com/Rahman-TeamDevelopment/ICR142_Benchmark/).

#### Assessing variant calling performance

To evaluate the utility of the ICR142 NGS validation series and ICR142 Benchmark we analysed data from three different open-source variant callers, GATK, OpEx and DeepVariant<sup>1,10,11</sup> and two commercial variant callers from Company A and Company B.

To generate BAM files for the GATK analysis, we aligned the ICR142 FASTQ files with BWA-MEM v0.7.12 and removed

duplicates using [Picard](#) v1.129<sup>13</sup>. We ran a GATK v3.4-46 analysis on the 142 BAM files to create a multi-sample VCF file ([Supplementary File 1](#)). We then applied standard additional filters of  $AB > 0.2$ ,  $DP \geq 10$  and  $GQ \geq 20$  and included the remaining variants as the GATK set.

We ran OpEx v1.0.0, which uses [Platypus](#) v0.1.5 as its variant caller, with default settings to generate 142 single sample VCF output files from the ICR142 FASTQ files<sup>11</sup>. Variants flagged as “high” by OpEx were included as the OpEx set.

We ran DeepVariant<sup>10</sup> with default settings using the OpEx BAM files to generate 142 gVCF output files. Two versions of DeepVariant were run; v0.5.2 and v0.6.1.

Two commercial variant calling providers, referred to as Company A and Company B, supplied data for the ICR142 validation series, Company A supplied 142 individual VCF files and Company B supplied a multi-sample gVCF file.

We pre-processed all multi-sample and gVCF files to ensure compatibility with the script. Multi-sample files were split into 142 single sample files with the `vcf-subset` command in `vcftools` v0.1.14<sup>14</sup>. For each gVCF file, we analysed the variant call subset to generate an initial result for each site. The reference call subset was then used to assign a missing value at sites with no call.

## Results

### Variant detection performance

We used the data from GATK, OpEx and DeepVariant v0.6.1 to provide a baseline for expected variant calling performance ([Table 1](#) and [Supplementary File 2](#)). Comparison of the three pipelines showed concordance at 92% of sites, both positive and negative. Because the same alignment files were used for both OpEx and DeepVariant (BWA), with a related but

different aligner used for GATK (BWA-MEM), one might have expected the OpEx and DeepVariant results to be more similar to one another than to GATK. However, the aligner used did not seem to have a strong impact, as GATK and DeepVariant showed similar performance, while OpEx had a better false detection rate but lower sensitivity. This was expected, as the OpEx “high” quality filter was designed to achieve exactly this balance for its first-pass exome sequence analysis<sup>11</sup>.

Sites where all three pipelines called false positives highlight common challenges in variant calling. False positive base substitutions in *POTEH* and *CHEK2* are likely due to non-specific capture of sequences derived from homologous genomic sequences. For example, the region surrounding the false positive position on chromosome 22 in *POTEH* has 90–95% homology with other paralogs of the *POTE* gene, with the allele at the exact position varying between them. False positive indels called in *MUC13* and *SLC39A14* provide examples of the challenges of variant calling in short repeating sequence motifs in NGS data. Although it is possible that the Sanger result is a false negative at these sites, we consider this to be unlikely.

The discordant false positive values reveal pipeline-specific differences in variant calling performance. GATK had seven unique sites with false positives, i.e. not called by either OpEx or DeepVariant, all of which were indels. These included two sites with a long insertion (SiteIDs 31, 290) and one site with a cluster of multiple calls (SiteID 75) ([Supplementary File 2](#)). Within the cluster, there would be no overall change in length if one could assume that all calls occurred on the same allele. However, phasing information was not provided by GATK until v3.3, and is only run automatically under specific conditions in more recent versions. Users of GATK should be cautious when multiple indels are called in close proximity in the same sample and consider visual inspection of the BAM file to check phasing, if phasing was not performed automatically. DeepVariant

**Table 1. Performance of multiple variant callers based on the ICR142 dataset.** Performance metrics were calculated as: Sensitivity =  $TP/(TP+FN)$ , Specificity =  $TN/(TN+FP)$  or False detection rate =  $FP/(FP+TP)$ , where TP = true positive sites; TN = true negative sites; FP = false positive sites; FN = false negative sites as described in Methods. The ICR142 dataset was generated using the Illumina TruSeq exome.

	Variant type	BWA + GATK	OpEx (Stampy + Platypus)	Stampy + DeepVariant
Sensitivity	Overall	404/416 (97%)	391/416 (94%)	405/416 (97%)
	Base substitutions	123/123 (100%)	118/123 (96%)	123/123 (100%)
	Indels	281/293 (96%)	273/293 (93%)	282/293 (96%)
Specificity	Overall	266/288 (92%)	279/288 (97%)	270/288 (94%)
	Base substitutions	39/41 (95%)	39/41 (95%)	35/41 (85%)
	Indels	227/247 (92%)	240/247 (97%)	235/247 (95%)
False detection rate	Overall	22/426 (5%)	9/400 (2%)	18/423 (4%)
	Base substitutions	2/125 (2%)	2/120 (2%)	6/129 (5%)
	Indels	20/301 (7%)	7/280 (2%)	12/294 (4%)

had four unique sites with false positives, all of which were base substitutions with QUAL value ranging from 9.3–20.3<sup>15</sup>. OpEx did not have any unique sites with false positives, consistent with the priority of the high quality OpEx filter to limit the false detection rate.

#### Indel detection and representation accuracy

ICR142 Benchmarker makes a distinction between variant detection and accurate variant representation since indels detected by NGS are often validated by an orthogonal technique such as Sanger sequencing, and the call amended if required. There were ten variants which were detected by all three pipelines but not correctly represented by at least one pipeline. Nine variants were incorrectly represented by all three pipelines. These were all complex indels, indicating the need for improvement or further standardisation in the representation of this important class of variant. The final variant, an inframe deletion of 24bp in *GPRINI* (SiteID 607), was correctly represented by OpEx but both GATK and DeepVariant represented this variant as two separate frameshifting deletions of 13bp and 11bp. This is a crucial difference, as the functional impact of inframe and frameshifting variants is often markedly different. Excluding complex indels, all three methods had greater than 98% accuracy, only OpEx achieved 100% accuracy, with all of the 264 detected insertions or deletions correctly represented.

#### Utility in variant calling regression testing

Variant calling pipelines are frequently updated. Regression testing ensures previously developed and tested pipelines still perform in the same way after updates have been implemented. The ICR142 NGS validation series allows independent regression testing, and we believe it could be usefully incorporated into variant calling development processes, particularly in the clinical setting.

To investigate this we performed regression testing by performing the same ICR142 Benchmarker analysis with DeepVariant v0.5.2 and v0.6.1. We found that v0.6.1 improved on v0.5.2 across all metrics, for both base substitutions and indels<sup>15</sup>. Comparison of the site-specific performance allowed us to draw more detailed insights. There were ten sites with a change in performance (Supplementary File 3). For two sites (SiteID 34, 666) with a single correctly detected indel, v0.6.1 detected an additional indel, an unexpected change in indel calling behaviour. For seven sites the calling performance improved, with one false positive base substitution and two false positive indels no longer called and four previously undetected indel variants now called by v0.6.1. However, one site had decreased performance, with three false positive indels newly called by v0.6.1 at a site in *PABPC3* (SiteID 114). As one of the four newly detected indel variants occurred at a nearby site in *PABPC3* in a different sample, this indicates that the improved calling performance comes at the cost of additional false positives at one site. Taken together, these data indicate that v0.6.1 provides validated improvement on v0.5.2, with some caveats that may inform future updates.

#### Utility in creation of variant detection filters

Many variant callers apply filters of the raw calls to improve performance. We believe the ICR142 validation series and ICR142 Benchmarker can be used to inform optimal filter creation. To evaluate this we investigated the performance of DeepVariant v0.6.1. We found that while sensitivity was excellent for both base substitutions and indels, specificity was surprisingly low for base substitutions at 85% (Table 1). We looked at the quality information returned by DeepVariant for all base substitution positive and negative sites in the ICR142 series<sup>15</sup>. We found that imposing a threshold of 30 on the QUAL column for base substitution calls reduced the false detection rate from 5% to 2%, increased the specificity to 95%, and did not greatly reduce the sensitivity, as only one variant was excluded, which had a QUAL value of 29.3, resulting in a sensitivity of 99%. We thus recommend using a filter of QUAL threshold of 30 for base substitution calls when using DeepVariant v0.6.1.

#### Benchmarking variant detection performance

Using the concordant data from the open-source pipelines allowed us to describe the expected baseline performance for variant calling (Table 2 and Supplementary File 2). There were 387 variants detected by GATK, OpEx and DeepVariant, which we call Group A variants. Any method seeking to perform as well as these open-source pipelines should be able to detect all variants in Group A. There were 261 sites where no variant was detected, which we call Group B. Methods aiming to have equivalent performance to open-source pipelines should avoid making variant calls at all Group B sites. Failure to detect a Group A variant or calling a variant at a Group B site indicates substandard performance and warrants further investigation at the algorithmic and/or filtering stage.

To demonstrate the utility of ICR142 Benchmarker to provide useful comparative performance information, we assessed variant calling by two commercial pipelines, which we call Company A and Company B (Supplementary File 4 and Supplementary File 5).

**Table 2. Expected baseline performance for variant calling.** Group A – variants that should be detected by any variant calling pipeline; Group B – sites in which a base substitution or insertion and/or deletion should not be called.

	Number of Sites
<b>Group A</b>	
- Base substitution variants	118
- Deletion variants	186
- Insertion variants	74
- Complex indel variants	9
<b>Group B</b>	
- No base substitution	35
- No indel	226

Company A showed overall good sensitivity (96%), specificity (95%) and false detection rate (4%) ([Supplementary File 4](#)). However, six Group A variants were not called and a false positive was called at one Group B site. The lower than typical ability to detect Group A variants indicates further work should be performed to understand why these variants were missed.

Company B also showed overall good sensitivity (98%), specificity (94%) and false detection rate (4%) ([Supplementary File 5](#)). However, one Group A variant was not called and false positives were called at three Group B sites. Deeper evaluation of these results by the companies could help them improve their variant calling performance.

The ICR142 Benchmark report for each company provides a clear summary of performance overall and for indels and base substitutions separately. The report also gives specific benchmarking information about Group A variants and Group B sites in a simple, clear and concise fashion ([Supplementary File 4](#) and [Supplementary File 5](#)). The report further highlights if missing data prevents assessment at any given site, as was the case for one negative site in the data from Company B ([Supplementary File 5](#)).

## Conclusion

Evaluation, optimisation and benchmarking of performance, including comparison with current widely-used pipelines, is essential for clinical, commercial and academic NGS variant calling applications. We have developed a tool, ICR142 Benchmark, to achieve these essential requirements in a consistent, reproducible and transparent fashion, using the ICR142 NGS validation dataset. ICR142 Benchmark returns useful outputs with various levels of detail to allow both broad and deep understanding of variant calling performance. ICR142 Benchmark can be applied to VCF files generated by any variant caller, allowing intra- and inter-pipeline comparison. ICR142 Benchmark is also useful in the optimisation of variant calling algorithms and outputs, including regression testing and filter creation. The ICR142 Benchmark report provides simple, clear and concise summary statements about a pipeline's performance, including comparison with the performance of widely-used open-source pipelines. Use of the ICR142

NGS validation series and ICR142 Benchmark can therefore facilitate in-house optimisation and direct comparison of variant calling methods for NGS data.

## Data availability

The FASTQ files for the ICR142 validation series are available from the European Genome-phenome archive (EGA). The accession number is [EGAS00001001332](#).

## Software availability

ICR142 Benchmark is available at: [https://github.com/RahmanTeamDevelopment/ICR142\\_Benchmark/releases](https://github.com/RahmanTeamDevelopment/ICR142_Benchmark/releases)

Latest source code: [https://github.com/RahmanTeamDevelopment/ICR142\\_Benchmark](https://github.com/RahmanTeamDevelopment/ICR142_Benchmark)

Archived source code as at time of publication: <http://doi.org/10.5281/zenodo.1469013><sup>9</sup>

Software license: MIT

The full ICR142 Benchmark documentation is given in [Supplementary File 6](#) and is available at: [https://github.com/RahmanTeamDevelopment/ICR142\\_Benchmark/](https://github.com/RahmanTeamDevelopment/ICR142_Benchmark/)

Supporting data files of GATK, OpEx, DeepVariant v0.5.2 and v0.6.1 input and output files have been archived as a single project file on Open Science Framework: <http://doi.org/10.17605/OSF.IO/H3ZR9><sup>15</sup> under a CC0 1.0 Universal licence.

---

## Grant information

The work was supported by the Wellcome Trust [200990].

*The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.*

## Acknowledgements

We acknowledge support from the NIHR RM/ICR Specialist Biomedical Research Centre for Cancer. This work was undertaken as part of the Transforming Genetic Medicine Initiative ([www.thetgmi.org](http://www.thetgmi.org)).

## Supplementary material

**Supplementary File 1.** GATK analysis commands

[Click here to access the data](#)

**Supplementary File 2.** Site-specific ICR142 Benchmark results for GATK, OpEx and DeepVariant

[Click here to access the data](#)

The description of the column headings are given below:

**Sample** – sample name in the ICR142 NGS validation series

**Gene** – HGNC symbol

**SangerCall** – the most 3' representation annotated with CSN v1.0<sup>16</sup> or “No” if no variant present

**Type** – “bs”, “del”, “ins”, “complex”, or “indel” for base substitutions, simple deletions, simple insertions, complex indels, or negative indel sites, respectively

**Transcript** – the ENST ID from Ensembl v65 used to annotate the Sanger call

**CHR** – chromosome

**EvaluatedPosition** - evaluated hg19 site position, centre of designed amplicon

**POS** – the left-aligned position in hg19 coordinates for variants or “.” if no variant present

**REF** – the reference allele in hg19 for variants or “.” if no variant present

**ALT** – the alternate allele for variants or “.” if no variant present

**SiteID** – site ID in the ICR142 NGS validation series

**Zygosity** – “heterozygous” a variant that is present on only one allele or “homozygous” a variant that is present on both alleles

**OpEx** – “.” if there is a missing genotype, 0 if site is not called by OpEx, 1 if a base substitution is called when Type = “bs”, or integer value X if X indels are called when Type = “del”, “ins”, “complex”, or “indel”

**OpExExactFinalMatch** – “yes” if CHR, POS, REF and ALT all match when SangerCall is not “No” and OpEx > 0, “no” if CHR, POS REF and ALT do not match when SangerCall is not “No” and OpEx > 0, “.” otherwise

**GATK** – “.” if there is a missing genotype, 0 if site is not called by GATK, 1 if a base substitution is called when Type = “bs”, or integer value X if X indels are called when Type = “del”, “ins”, “complex”, or “indel”

**GATKExactFinalMatch** – “yes” if CHR, POS, REF and ALT all match when SangerCall is not “No” and GATK > 0, “no” if CHR, POS REF and ALT do not match when SangerCall is not “No” and GATK > 0, “.” otherwise

**DeepVariant** – “.” if there is a missing genotype, 0 if site is not called by DeepVariant, 1 if a base substitution is called when Type = “bs”, or integer value X if X indels are called when Type = “del”, “ins”, “complex”, or “indel”

**DeepVariantExactFinalMatch** – “yes” if CHR, POS, REF and ALT all match when SangerCall is not “No” and DeepVariant > 0, “no” if CHR, POS REF and ALT do not match when SangerCall is not “No” and DeepVariant > 0, “.” otherwise

**Group** – “A” if SangerCall is not “No” and GATK, OpEx and DeepVariant are all > 0, “B” if SangerCall is “No” and GATK, OpEx and DeepVariant are all 0, “.” otherwise

**Supplementary File 3.** Variant calling regression testing of DeepVariant v0.5.2 and v0.6.1

[Click here to access the data](#)

The description of the column headings are given below:

**Sample**– sample name in the ICR142 NGS validation series

**Gene**– HGNC symbol

**SangerCall** – the most 3' representation annotated with CSN v1.0<sup>16</sup> or “No” if no variant present

**Type**– “bs”, “del”, “ins”, “complex”, or “indel” for base substitutions, simple deletions, simple insertions, complex indels, or negative indel sites, respectively

**Transcript** – the ENST ID from Ensembl v65 used to annotate the Sanger call

**CHR** – chromosome

**EvaluatedPosition** - evaluated hg19 site position, centre of designed amplicon

**POS** – the left-aligned position in hg19 coordinates for variants or “.” if no variant present

**REF** – the reference allele in hg19 for variants or “.” if no variant present



**ALT** – the alternate allele for variants or “.” if no variant present

**SiteID** – site ID in the ICR142 NGS validation series

**Group** – “A” if SangerCall is not “No” and GATK, OpEx and DeepVariant are all > 0, “B” if SangerCall is “No” and GATK, OpEx and DeepVariant are all 0, “.” Otherwise

**DeepVariant v0.5.2** – “.” if there is a missing genotype, 0 if site is not called by DeepVariant v0.5.2, 1 if a base substitution is called when Type = “bs”, or integer value X if X indels are called when Type = “del”, “ins”, “complex”, or “indel”

**DeepVariant v0.6.1** – “.” if there is a missing genotype, 0 if site is not called by DeepVariant v0.6.1, 1 if a base substitution is called when Type = “bs”, or integer value X if X indels are called when Type = “del”, “ins”, “complex”, or “indel”

**ConcordantFinalResult v0.5.2** – “no” if either SangerCall is “No” and DeepVariant v0.5.2 is >0 or SangerCall is not “No” and DeepVariant v0.5.2 is “0” or “.”, “yes” if SangerCall and DeepVariant v0.5.2 are concordant

**ConcordantFinalResult v0.6.1** – “no” if either SangerCall is “No” and DeepVariant v0.6.1 is >0 or SangerCall is not “No” and DeepVariant v0.6.1 is “0” or “.”, “yes” if SangerCall and DeepVariant v0.6.1 are concordant

**ExactFinalMatch v0.5.2** – “yes” if CHR, POS, REF and ALT all match when SangerCall is not “No” and DeepVariant v0.5.2 > 0, “no” if CHR, POS REF and ALT do not match when SangerCall is not “No” and DeepVariant v0.5.2 > 0, “.” otherwise

**ExactFinalMatch v0.6.1** – “yes” if CHR, POS, REF and ALT all match when SangerCall is not “No” and DeepVariant v0.6.1 > 0, “no” if CHR, POS REF and ALT do not match when SangerCall is not “No” and DeepVariant v0.6.1 > 0, “.” otherwise

**Supplementary File 4.** ICR142 Benchmark Report for Company A.

[Click here to access the data](#)

**Supplementary File 5.** ICR142 Benchmark Report for Company B.

[Click here to access the data](#)

**Supplementary File 6.** ICR142 Benchmark v1.0.1 documentation.

[Click here to access the data](#)

## References

- DePristo MA, Banks E, Poplin R, *et al.*: **A framework for variation discovery and genotyping using next-generation DNA sequencing data.** *Nat Genet.* 2011; 43(5): 491–8.  
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Fang H, Bergmann EA, Arora K, *et al.*: **Indel variant analysis of short-read sequencing data with Scalpel.** *Nat Protoc.* 2016; 11(12): 2529–2548.  
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Koboldt DC, Zhang G, Larson DE, *et al.*: **VarScan 2: somatic mutation and copy number alteration discovery in cancer by exome sequencing.** *Genome Res.* 2012; 22(3): 568–76.  
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Rimmer A, Phan H, Mathieson I, *et al.*: **Integrating mapping-, assembly- and haplotype-based approaches for calling variants in clinical sequencing applications.** *Nat Genet.* 2014; 46(8): 912–918.  
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Li D, Kim W, Wang L, *et al.*: **Comparison of INDEL Calling Tools with Simulation Data and Real Short-Read Data.** *IEEE/ACM Trans Comput Biol Bioinform.* 2018.  
[PubMed Abstract](#) | [Publisher Full Text](#)
- Sandmann S, de Graaf AO, Karimi M, *et al.*: **Evaluating Variant Calling Tools for Non-Matched Next-Generation Sequencing Data.** *Sci Rep.* 2017; 7: 43169.  
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Roy S, Coldren C, Karunamurthy A, *et al.*: **Standards and Guidelines for Validating Next-Generation Sequencing Bioinformatics Pipelines: A Joint Recommendation of the Association for Molecular Pathology and the College of American Pathologists.** *J Mol Diagn.* 2018; 20(1): 4–27.  
[PubMed Abstract](#) | [Publisher Full Text](#)
- Ruark E, Renwick A, Clarke M, *et al.*: **The ICR142 NGS validation series: a resource for orthogonal assessment of NGS analysis [version 1; referees: 2 approved].** *F1000Res.* 2016; 5: 386.  
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Holt E, Ruark E: **ICR142 Benchmark v1.0.2.** *Zenodo.* 2018.  
<http://www.doi.org/10.5281/zenodo.1469013>
- Poplin R, Chang PC, Alexander D, *et al.*: **Creating a universal SNP and small indel variant caller with deep neural networks.** *bioRxiv.* 2018; 092890.  
[Publisher Full Text](#)
- Ruark E, Münz M, Clarke M, *et al.*: **OpEx - a validated, automated pipeline optimised for clinical exome sequence analysis.** *Sci Rep.* 2016; 6: 31029.  
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Ushey K, McPherson J, Cheng J, *et al.*: **A Dependency Management System for Projects and their R Package Dependencies.** 2016.  
[Reference Source](#)
- Stals KL, Wakeling M, Baptista J, *et al.*: **Diagnosis of lethal or prenatal-onset autosomal recessive disorders by parental exome sequencing.** *Prenat Diagn.* 2018; 38(1): 33–43.  
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Danecek P, Auton A, Abecasis G, *et al.*: **The variant call format and VCFtools.** *Bioinformatics.* 2011; 27(15): 2156–8.  
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Rahman N: **ICR142 Benchmark Supporting material.** *Open Science Framework.* 2018.  
<http://www.doi.org/10.17605/OSF.IO/H3ZR9>
- Munz M, Ruark E, Renwick A, *et al.*: **CSN and CAVA: variant annotation tools for rapid, robust next-generation sequencing analysis in the clinical setting.** *Genome Med.* 2015; 7: 76.  
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)

# Open Peer Review

Current Referee Status:  

---

## Version 2

Referee Report 09 November 2018

<https://doi.org/10.21956/wellcomeopenres.16253.r34217>

 **Birgit Sikkema-Raddatz** , **Lennart F. Johansson** 

Department of Genetics, University of Groningen, University Medical Center Groningen, Groningen, The Netherlands

Thank you for the second version. All points are discussed. Together with the comments it will enable to further value the benchmarker and allow readers to use the benchmarker for several comparisons.

**Competing Interests:** No competing interests were disclosed.

**We have read this submission. We believe that we have an appropriate level of expertise to confirm that it is of an acceptable scientific standard.**

Referee Report 01 November 2018

<https://doi.org/10.21956/wellcomeopenres.16253.r34143>

 **Oliver Hofmann** 

Centre for Cancer Research, University of Melbourne, Melbourne, Vic, Australia

Thank you for the quick response. All my (minor) comments have been addressed. Brad Chapman also has additional resources related to ICR142 available at <https://github.com/bcbio/icr142-validation> which may be of use to other researchers.

**Competing Interests:** No competing interests were disclosed.

**I have read this submission. I believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard.**

---

## Version 1

Referee Report 12 October 2018

<https://doi.org/10.21956/wellcomeopenres.16077.r34022>



**Birgit Sikkema-Raddatz**  , **Lennart F. Johansson** 

Department of Genetics, University of Groningen, University Medical Center Groningen, Groningen, The Netherlands

The manuscript presented by Ruark et al. describes the ICR142 tool that can be used to calculate sensitivity, specificity and false detection rates of NGS variant callers. Making such open source tools available to the community is very valuable. In our opinion the overall quality of the manuscript is good, however, the claims made in the title are too strong.

#### Major points:

1. **Table 1 and in general:** It should be specified that the benchmarker ICR142 is only for Illumina data, enriched with a TruSeq kit.

It is mentioned that the values are calculated using the ICR142 dataset. However, the use of the terms sensitivity and specificity imply that these numbers are generalizable to the entire exome. However, the sites are selected using data Illumina TruSeq Exome enrichment procedure in combination with the Illumina HiSeq 2000 sequencing. Difficult positions could differ for different enrichment procedures or sequencing platforms, even between Illumina machines. A variant caller may for instance have a good performance on an Illumina dataset but less on an IonTorrent dataset.

2. **The title** suggests that variant calling is optimised. However, the manuscript does not discuss optimization of variant calling. It only states that it can be done. In my opinion this is not enough to warrant the statement in the title.

It would be interesting to further discuss reasons for lower sensitivity and specificity. Only in the Deep Variant tool settings were changed. A more systematic approach for all the tools would be useful. Are there clusters of (types of) variants being missed or called together at different settings. Such a discussion could warrant the optimisation claim.

3. It would be interesting to discuss how the benchmarking results for one or more of the aligner/variant calling combinations compares to other benchmarking datasets. For instance on high-confidence variants in a Genome In A Bottle dataset.

#### Minor points

**The title** suggests that variant calling in general is evaluated. However, only germline SNV and Indel variants are taken into account. Copy number variant calls or somatic/mosaic variants are not benchmarked. The types of variants evaluated should be specified more clearly.

#### Page 3:

**Introduction:** we agree with the comment of reviewer 1 that evaluation of published pipelines is not limited.

**Method:** The selection criteria of the 704 variant sites can be stated more clearly.

**Figure 1** states that there are 288 Sanger validated negative sites. This is further specified in 41 sites without a base substitution and 247 sites without an indel. This suggests that the 41 sites do have an indel and the 247 sites do have a SNV. Is this the case or are the negative sites negative for both indels and variants. If so, specify.

**Table 1** should include the used aligner as well as the variant caller in the column headers. As is stated correctly in the methods and results section, the variant callers aren't isolated units and the aligner could influence the result (although the results show this is not the case here). A general remark: the data are not easy to read and understand.

**Page 5:** "regression testing" should be explained, also how to perform.

**Page 5/6:** It would be interesting to discuss if the 4% FDR in companies A and B and DeepVariant are constituted by the same variants/regions and how this compares to the 2% and 5% of OpEx and GATK.

**Page 6:** It is stated that the information of Companies A and B is shown in table 1. This is not the case.

**Page 6:** utility in creation of variant detection filters:

The filter setting to improve performance was tested for DeepVariant. What about the other caller? A more systematic approach is required.

We could not locate the 3 Vcf files to reproduce the output.

**Is the rationale for developing the new software tool clearly explained?**

Partly

**Is the description of the software tool technically sound?**

Yes

**Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?**

Yes

**Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?**

Yes

**Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?**

Partly

**Competing Interests:** No competing interests were disclosed.

**We have read this submission. We believe that we have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however we have significant reservations, as outlined above.**

Author Response 25 Oct 2018

**Nazneen Rahman**, The Institute of Cancer Research and The Royal Marsden NHS Foundation Trust, UK

## General Response

We thank the reviewers for their comments. We believe they may have a slight misunderstanding of our intentions in writing this paper and in making ICR142 Benchmarker. In 2016 we made the ICR142 dataset available (Ruark et al 10.12688/f1000research.8219.2). We did this simply because we had been using the dataset in-house and had found it useful and we thought others might also find it useful. Many have - the paper has been read >1000x and downloaded >200x. Following discussions and feedback with ICR142 users we made ICR142 Benchmarker. We believe/hope it makes using the ICR142 dataset easier and more useful.

We have included some simple use cases in this paper to highlight potential uses, but naturally there will be situations where ICR142 and hence ICR142 Benchmarker will not be suitable.

## Response to reviewers comments:

### Major points

- **Table 1 and in general:** It should be specified that the benchmarker ICR142 is only for Illumina data, enriched with a TruSeq kit. It is mentioned that the values are calculated using the ICR142 dataset. However, the use of the terms sensitivity and specificity imply that these numbers are generalizable to the entire exome. However, the sites are selected using data Illumina TruSeq Exome enrichment procedure in combination with the Illumina HiSeq 2000 sequencing. Difficult positions could differ for different enrichment procedures or sequencing platforms, even between Illumina machines. A variant caller may for instance have a good performance on an Illumina dataset but less on an IonTorrent dataset.

Response: Thank you. We have added to the abstract, paper and Table 1 legend that these are Illumina data. In the paper, we have added 'The exome sequence data was generated using the Illumina TruSeq Exome and a HiSeq2000 sequencer. Full details of the ICR142 series are given in Ruark et al'. ICR142 data can, and has, been used with data from other capture kits. All the ICR142 sites have been orthogonally validated so the results are not dependent on the enrichment process or sequencer. Indeed we believe that the ICR142 dataset and ICR142 Benchmarker have particular utility in helping to uncover this type of performance variability.

- **The title** suggests that variant calling is optimised. However, the manuscript does not discuss optimization of variant calling. It only states that it can be done. In my opinion this is not enough to warrant the statement in the title.

Response: Thank you. We had inadvertently omitted the word 'performance' from the title, which we have now added.

- It would be interesting to further discuss reasons for lower sensitivity and specificity. Only in the Deep Variant tool settings were changed. A more systematic approach for all the tools would be useful. Are there clusters of (types of) variants being missed or called together at different settings. Such a discussion could warrant the optimisation claim.

Response: Thank you. These are exactly the types of questions we hope people will find ICR142 Benchmarker useful in highlighting, evaluating and optimizing. We have extensively used ICR142 dataset in the optimization of OpEx, as described in the OpEx paper.

- It would be interesting to discuss how the benchmarking results for one or more of the aligner/variant calling combinations compares to other benchmarking datasets. For instance on high-confidence variants in a Genome In A Bottle dataset.

Response: Thank you. We agree these types of comparisons would be interesting. We hope that people will be inclined to do them, and as more people use (and hopefully make available) their ICR142 Benchmarker outputs we believe there may be several opportunities for interesting

comparisons.

### Minor points

- **The title** suggests that variant calling in general is evaluated. However, only germline SNV and Indel variants are taken into account. Copy number variant calls or somatic/mosaic variants are not benchmarked. The types of variants evaluated should be specified more clearly.

Response: Thank you. This is important. We have added to the abstract the following. 'Here we present ICR142 Benchmark, a tool to generate outputs for assessing germline base substitution and indel calling performance using the ICR142 NGS validation series'

### Page 3:

- **Introduction:** we agree with the comment of reviewer 1 that evaluation of published pipelines is not limited.

Response: We agree with you both. We have changed this sentence to. 'Evaluations of pipeline performance are often based on internal data. This makes comparison, standardisation and regulation of NGS variant calling performance difficult.'

- **Method:** The selection criteria of the 704 variant sites can be stated more clearly.

Response: Details for this are given in the original ICR142 publication. We have included in the revised paper 'Full details of the ICR142 series are given in Ruark et al'.

- **Figure 1** states that there are 288 Sanger validated negative sites. This is further specified in 41 sites without a base substitution and 247 sites without an indel. This suggests that the 41 sites do have an indel and the 247 sites do have a SNV. Is this the case or are the negative sites negative for both indels and variants. If so, specify.

Response: Each site was inspected and negative or positive for the specified type of variation. No other assumptions should be made. i.e. it should not be assumed that the 41 sites without a base substitution have an indel.

- **Table 1** should include the used aligner as well as the variant caller in the column headers. As is stated correctly in the methods and results section, the variant callers aren't isolated units and the aligner could influence the result (although the results show this is not the case here). A general remark: the data are not easy to read and understand.

Response: Thank you for this suggestion. We have included this.

- **Page 5:** "regression testing" should be explained, also how to perform.

Response: We have explained the term in the paper as follows: "Regression testing ensures previously developed and tested pipelines still perform in the same way after updates have been implemented." We don't feel it is appropriate to state how it should be performed, because there are many different ways to do regression testing, dependent on many factors. The aim of our paper is simply to provide a dataset and tool that people might find useful when doing regression testing.

- **Page 5/6:** It would be interesting to discuss if the 4% FDR in companies A and B and DeepVariant are constituted by the same variants/regions and how this compares to the 2% and 5% of OpEx and GATK.

Response: We agree that these types of comparisons would be interesting. We didn't have permission from Companies A and B to use their data in this way.

- **Page 6:** It is stated that the information of Companies A and B is shown in table 1. This is not the case

Response. Thank you for bringing this to our attention. We have removed this for Companies A and B.

- **Page 6:** utility in creation of variant detection filters: The filter setting to improve performance was tested for DeepVariant. What about the other caller? A more systematic approach is required.

Response: Thank you. In line with the recommendations for software tool articles we have included representative 'use cases'. The filter setting for DeepVariant is an example of one of these use cases. It was not our aim to perform a systematic approach to filter setting. Though we would be delighted if the ICR142 dataset and ICR142 Benchmark were used in this way.

- We could not locate the 3 Vcf files to reproduce the output.

Response: We have checked and all files are there and available to download. Of note, there are not three VCF files but rather 3 zipped files containing 142 VCF files each.

**Competing Interests:** No competing interests were disclosed.

Referee Report 24 September 2018

<https://doi.org/10.21956/wellcomeopenres.16077.r33804>



**Oliver Hofmann** 

Centre for Cancer Research, University of Melbourne, Melbourne, Vic, Australia

Ruark *et al* present 'ICR142 Benchmark', a tool to simplify comparing locally generated SNV calls against a Sanger-validated benchmark set using the previously published ICR142 sample set available from EGA. Additional methods to assess variant callers are always needed, and the easy software installation, pre-configured comparisons and result summaries make this task more accessible to groups without dedicated bioinformatics support. The tool is freely available under the MIT license and well-documented.

#### ## Major comments

- I would not agree that the evaluation of published pipelines is limited in the literature; I've lost count of the number of papers assessing variant callers, exome capture methods and other aspects of HTS workflows. That said, ICR142 Benchmark simplifies the comparison for researchers which is worth emphasizing.
- Readers might also benefit from a *brief* comparison to other available benchmarking datasets (Genome in a Bottle, COLO829, ...); the ICR142 set benefits from the breadth of having a large number of different samples, but doesn't cover as much of the genome as some of the WGS-based evaluation sets.
- Likewise, how does ICR142 Benchmark compare to existing frameworks such as RTG's *vcfeval* or Illumina's *hap.py/som.py*? Again, I see the use case for Benchmark but the target audience might not.
- I am not sure I quite understand how InDels are handled. Does Benchmark compare just the *size* of the insertion/deletion (at a given position)? Are InDels left-aligned as part of the VCF parsing step, or is this task left to the user?

**## Minor comments**

- A brief description of the ~700 covered site would be helpful to understand what ICF142 does/does not cover. Do any variants overlap difficult to sequence regions (e.g., from Heng Li's paper<sup>1</sup>) or other low complexity regions?
- It is worth mentioning that the ICR142 set covers germline variant calls from human lymphocyte DNA (apologies if I missed this being pointed out somewhere).
- Based on the GitHub repository it looks like the assessment is limited to hg19. Could benchmark data for GRChr37 and, more importantly, GRCh38 be made available? If not, worth mentioning in the paper.
- Could DeepVariant's low base substitution specificity be due to working with Stampy-generated alignments rather than bwa-mem alignment? I realize the assessment of callers is not the focus of this paper, but this might still be worth testing.
- It might be helpful to have a sample (Word) report in the GitHub repository.

**References**

1. Li H: Toward better understanding of artifacts in variant calling from high-coverage samples. *Bioinformatics*. 2014; **30** (20): 2843-51 [PubMed Abstract](#) | [Publisher Full Text](#)

**Is the rationale for developing the new software tool clearly explained?**

Partly

**Is the description of the software tool technically sound?**

Yes

**Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?**

Yes

**Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?**

Yes

**Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?**

Partly

**Competing Interests:** No competing interests were disclosed.

**I have read this submission. I believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.**



Author Response 25 Oct 2018

**Nazneen Rahman**, The Institute of Cancer Research and The Royal Marsden NHS Foundation Trust, UK

### ## Major comments

- I would not agree that the evaluation of published pipelines is limited in the literature; I've lost count of the number of papers assessing variant callers, exome capture methods and other aspects of HTS workflows. That said, ICR142 Benchmarker simplifies the comparison for researchers which is worth emphasizing.

Response: We agree with you. We have changed this sentence to 'Evaluations of pipeline performance are often based on internal data. This makes comparison, standardisation and regulation of NGS variant calling performance difficult.'

- Readers might also benefit from a *brief* comparison to other available benchmarking datasets (Genome in a Bottle, COLO829, ...); the ICR142 set benefits from the breadth of having a large number of different samples, but doesn't cover as much of the genome as some of the WGS-based evaluation sets. Likewise, how does ICR142 Benchmarker compare to existing frameworks such as RTG's *vcfeval* or Illumina's *hap.py/som.py*? Again, I see the use case for Benchmarker but the target audience might not.

Response: We made ICR142 Benchmarker available following interactions with users of the ICR142 dataset, to enhance its usability and potential utility. We are pleased you can see possible advantages compared with other datasets / frameworks and we would be delighted if someone wanted to evaluate this more formally, but as you say that wasn't the purpose of this paper. Overall, we personally favour using as many as possible – one almost always gains something from such evaluations.

- I am not sure I quite understand how InDels are handled. Does Benchmarker compare just the *size* of the insertion/deletion (at a given position)? Are InDels left-aligned as part of the VCF parsing step, or is this task left to the user?

Response: We considered detection and annotation/representation of indels separately, because there remains considerable variation in how the same indel can be described. If an indel was called within a 200bp window centred on the site position we considered an indel to have been detected and it is counted as a Positive indel call. It is possible that the annotation/representation of the indel might differ. The indels are not left-aligned by ICR142 benchmarker.

### ## Minor comments

- A brief description of the ~700 covered site would be helpful to understand what ICR142 does/does not cover. Do any variants overlap difficult to sequence regions (e.g., from Heng Li's paper<sup>1</sup>) or other low complexity regions?

Response: We have included in the revised paper 'Full details of the ICR142 series are given in Ruark et al'. The original paper describes how we selected the 704 sites. 13 of the sites (SiteIDs: 150, 169, 191, 221, 235, 473, 502, 512, 523, 534, 543, 580, 639) overlap those in Heng Li's paper.

- It is worth mentioning that the ICR142 set covers germline variant calls from human lymphocyte DNA (apologies if I missed this being pointed out somewhere).

Response: Thank you. Yes it is. We have added 'for assessing germline base substitution and indel calling performance....' To the abstract.

- Based on the GitHub repository it looks like the assessment is limited to hg19. Could benchmark data for GRChr37 and, more importantly, GRCh38 be made available? If not, worth mentioning in the paper.

Response: Thank you. Great point. We have updated ICR142 Benchmarker so this is possible. In the paper we have added 'It can be used with hg19/GRCh37 or hg38/GRCh38 data.'

- Could DeepVariant's low base substitution specificity be due to working with Stampy-generated alignments rather than bwa-mem alignment? I realize the assessment of callers is not the focus of this paper, but this might still be worth testing.

Response: Thank you. This is exactly the type of question we hoped ICR142 Benchmarker might stimulate, though it is not a focus of our work to take it any further. In this paper we simply wanted to highlight some possible use cases, to give an idea of how people might find the ICR142 series and ICR12 Benchmarker useful.

- It might be helpful to have a sample (Word) report in the GitHub repository.

Response: Thank you. We have updated the GitHub repository to include a sample (Word) report.

***Competing Interests:*** No competing interests were disclosed.

---