

 [google / deepvariant](#) Public

DeepVariant is an analysis pipeline that uses a deep neural network to call genetic variants from next-generation DNA sequencing data.

 BSD-3-Clause license 2.9k stars  695 forks  Activity Star Notifications

<> Code

 Issues 3 Pull requests 2 Actions Projects Security Insights r1.5 ▾

Go to file



pichuan and Copybara-Service Update README.md ...

✓ on Feb 27 ⌚ 2,087

[View code](#)

DeepVariant

 release **v1.5.0** [announcements](#) [blog](#)

DeepVariant is a deep learning-based variant caller that takes aligned reads (in BAM or CRAM format), produces pileup image tensors from them, classifies each tensor using a convolutional neural network, and finally reports the results in a standard VCF or gVCF file.

DeepVariant supports germline variant-calling in diploid organisms.

- NGS (Illumina) data for either a [whole genome](#) or [whole exome](#).
- [RNA-seq Case Study](#) for Illumina RNA-seq.
- PacBio HiFi data, see the [PacBio case study](#).
- Oxford Nanopore R10.4.1 Simplex or Duplex data, see the [ONT R10.4.1 Simplex case study](#) and [ONT R10.4.1 Duplex case study](#)

 README.md

- Oxford Nanopore R9.4.1 data by using [PEPPER-DeepVariant](#).

PACIFIC BIOSCIENCES OF CALIFORNIA, INC. EXHIBIT - 1036

Please also note:

- For somatic data or any other samples where the genotypes go beyond two copies of DNA, DeepVariant will not work out of the box because the only genotypes supported are hom-alt, het, and hom-ref.
- The models included with DeepVariant are only trained on human data. For other organisms, see the [blog post on non-human variant-calling](#) for some possible pitfalls and how to handle them.

DeepTrio

DeepTrio is a deep learning-based trio variant caller built on top of DeepVariant. DeepTrio extends DeepVariant's functionality, allowing it to utilize the power of neural networks to predict genomic variants in trios or duos. See [this page](#) for more details and instructions on how to run DeepTrio.

DeepTrio supports germline variant-calling in diploid organisms for the following types of input data:

- NGS (Illumina) data for either [whole genome](#) or whole exome.
- PacBio HiFi data, see the [PacBio case study](#).

Please also note:

- All DeepTrio models were trained on human data.
- It is possible to use DeepTrio with only 2 samples (child, and one parent).
- External tool [GLnexus](#) is used to merge output VCFs.

How to run DeepVariant

We recommend using our Docker solution. The command will look like this:

```
BIN_VERSION="1.5.0"
docker run \
  -v "YOUR_INPUT_DIR":"/input" \
  -v "YOUR_OUTPUT_DIR:/output" \
  google/deepvariant:"${BIN_VERSION}" \
  /opt/deepvariant/bin/run_deepvariant \
  --model_type=WGS \
  **Replace this string with exactly one of the following
[WGS,WES,PACBIO,ONT_R104,HYBRID_PACBIO_ILLUMINA]**
  --ref=/input/YOUR_REF \
  --reads=/input/YOUR_BAM \
```



```
--output_vcf=/output/YOUR_OUTPUT_VCF \  
--output_gvcf=/output/YOUR_OUTPUT_GVCF \  
--num_shards=$(nproc) \ **This will use all your cores to run make_examples.  
Feel free to change.**  
--logging_dir=/output/logs \ **Optional. This saves the log output for each  
stage separately.  
--dry_run=false **Default is false. If set to true, commands will be printed  
out but not executed.
```

To see all flags you can use, run: `docker run google/deepvariant:"${BIN_VERSION}"`

If you're using GPUs, or want to use Singularity instead, see [Quick Start](#) for more details or see all the [setup options](#) available.

For more information, also see:

- [Full documentation list](#)
- [Detailed usage guide](#) with more information on the input and output file formats and how to work with them.
- [Best practices for multi-sample variant calling with DeepVariant](#)
- [\(Advanced\) Training tutorial](#)
- [DeepVariant's Frequently Asked Questions, FAQ](#)

How to cite

If you're using DeepVariant in your work, please cite:

[A universal SNP and small-indel variant caller using deep neural networks. *Nature Biotechnology* 36, 983–987 \(2018\).](#)

Ryan Poplin, Pi-Chuan Chang, David Alexander, Scott Schwartz, Thomas Colthurst, Alexander Ku, Dan Newburger, Jojo Dijamco, Nam Nguyen, Pegah T. Afshar, Sam S. Gross, Lizzie Dorfman, Cory Y. McLean, and Mark A. DePristo.

doi: <https://doi.org/10.1038/nbt.4235>

Additionally, if you are generating multi-sample calls using our [DeepVariant and GLnexus Best Practices](#), please cite:

[Accurate, scalable cohort variant calls using DeepVariant and GLnexus. *Bioinformatics* \(2021\).](#)

Taedong Yun, Helen Li, Pi-Chuan Chang, Michael F. Lin, Andrew Carroll, and Cory Y. McLean.

doi: <https://doi.org/10.1093/bioinformatics/btaa1081>

Why Use DeepVariant?

- **High accuracy** - DeepVariant won 2020 [PrecisionFDA Truth Challenge V2](#) for All Benchmark Regions for ONT, PacBio, and Multiple Technologies categories, and 2016 [PrecisionFDA Truth Challenge](#) for best SNP Performance. DeepVariant maintains high accuracy across data from different sequencing technologies, prep methods, and species. For [lower coverage](#), using DeepVariant makes an especially great difference. See [metrics](#) for the latest accuracy numbers on each of the sequencing types.
- **Flexibility** - Out-of-the-box use for [PCR-positive](#) samples and [low quality sequencing runs](#), and easy adjustments for [different sequencing technologies](#) and [non-human species](#).
- **Ease of use** - No filtering is needed beyond setting your preferred minimum quality threshold.
- **Cost effectiveness** - With a single non-preemptible n1-standard-16 machine on Google Cloud, it costs ~\$11.8 to call a 30x whole genome and ~\$0.89 to call an exome. With preemptible pricing, the cost is \$2.84 for a 30x whole genome and \$0.21 for whole exome (not considering preemption).
- **Speed** - See [metrics](#) for the runtime of all supported datatypes on a 64-core CPU-only machine. Multiple options for acceleration exist.
- **Usage options** - DeepVariant can be run via Docker or binaries, using both on-premise hardware or in the cloud, with support for hardware accelerators like GPUs and TPUs.

(1): Time estimates do not include mapping.

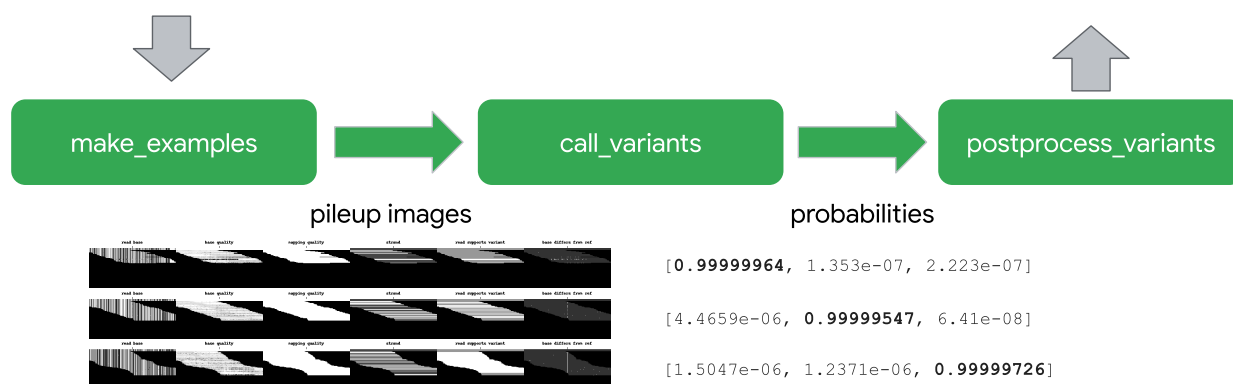
How DeepVariant works

Inputs:

- Alignments (BAM or CRAM)
- Reference (FASTA)

Outputs:

- Variant calls (VCF)
- (optional) gVCF



For more information on the pileup images and how to read them, please see the ["Looking through DeepVariant's Eyes" blog post](#).

DeepVariant relies on [Nucleus](#), a library of Python and C++ code for reading and writing data in common genomics file formats (like SAM and VCF) designed for painless integration with the [TensorFlow](#) machine learning framework. Nucleus was built with DeepVariant in mind and open-sourced separately so it can be used by anyone in the genomics research community for other projects. See this blog post on [Using Nucleus and TensorFlow for DNA Sequencing Error Correction](#).

DeepVariant Setup

Prerequisites

- Unix-like operating system (cannot run on Windows)
- Python 3.8

Official Solutions

Below are the official solutions provided by the [Genomics team in Google Health](#).

Name	Description
Docker	This is the recommended method.
Build from source	DeepVariant comes with scripts to build it on Ubuntu 20.04. To build and run on other Unix-based systems, you will need to modify these scripts.
Prebuilt Binaries	Available at gs://deepvariant/ . These are compiled to use SSE4 and AVX instructions, so you will need a CPU (such as Intel Sandy Bridge) that supports them. You can check the <code>/proc/cpuinfo</code> file on your computer, which lists these features under "flags".

Contribution Guidelines

Please [open a pull request](#) if you wish to contribute to DeepVariant. Note, we have not set up the infrastructure to merge pull requests externally. If you agree, we will test and submit the changes internally and mention your contributions in our [release notes](#). We apologize for any inconvenience.

If you have any difficulty using DeepVariant, feel free to [open an issue](#). If you have general questions not specific to DeepVariant, we recommend that you post on a community discussion forum such as [BioStars](#).

License

[BSD-3-Clause license](#)

Acknowledgements

DeepVariant happily makes use of many open source packages. We would like to specifically call out a few key ones:

- [Boost Graph Library](#)
- [abseil-cpp](#) and [abseil-py](#)
- [CLIF](#)
- [GNU Parallel](#)
- [htslib & samtools](#)
- [Nucleus](#)
- [numpy](#)
- [SSW Library](#)
- [TensorFlow and Slim](#)

We thank all of the developers and contributors to these packages for their work.

Disclaimer

This is not an official Google product.

NOTE: the content of this research code repository (i) is not intended to be a medical device; and (ii) is not intended for clinical use of any kind, including but not limited to diagnosis or prognosis.

Releases 19

 **DeepVariant 1.5.0** Latest
on Feb 28

[+ 18 releases](#)

Packages

No packages published

Contributors 22

[+ 11 contributors](#)

Languages

● Python 67.9% ● C++ 24.7% ● Shell 3.8% ● Starlark 3.2% ● Dockerfile 0.4%